Fortinet Inc.
FortiWeb 5.6

Security Target

Doc No: 1877-000-D102
Version: 1.10
28 November 2017

Fortinet, Incorporated
899 Kifer Road
Sunnyvale, California
94086

Prepared by:
EWA-Canada
1223 Michael Street, Suite 200
Ottawa, Ontario, Canada
K1J7T2

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 SECURITY TARGET INTRODUCTION

This Security Target (ST) defines the scope of the evaluation in terms of the assumptions made, the intended environment for the Target of Evaluation (TOE), the Information Technology (IT) security functional and assurance requirements to be met, and the level of confidence (evaluation assurance level) to which it is asserted that the TOE satisfies its IT security requirements. This document forms the baseline for the Common Criteria (CC) evaluation.

1.1 DOCUMENT ORGANIZATION

Section 1, ST Introduction, provides the Security Target (ST) reference, the Target of Evaluation (TOE) reference, the TOE overview and the TOE description.

Section 2, Conformance Claims, describes how the ST conforms to the Common Criteria and Protection Profile.

Section 3, Security Problem Definition, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.

Section 4, Security Objectives, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment in response to the problem defined by the security problem definition.

Section 5, Extended Components Definition, defines the extended components which are then detailed in Section 6.

Section 6, Security Requirements, specifies the security functional and assurance requirements that must be satisfied by the TOE and the Information Technology (IT) environment.

Section 7, TOE Summary Specification, describes the security functions that are included in the TOE to enable it to meet the IT security functional requirements.

Section 8 Terminology and Acronyms, defines the acronyms and terminology used in this ST.

1.2 SECURITY TARGET REFERENCE

ST Title:	Fortinet Inc. FortiWeb 5.6 Security Target
ST Version:	1.10
ST Date:	28 November 2017

1.3 TOE REFERENCE

TOE Identification:	FortiWeb 5.6 build 6180

TOE Developer: Fortinet Inc.

TOE Type: Network Device (Web Application Firewall)

1.4 TOE OVERVIEW

FortiWeb Web Application Firewalls provide specialized, layered web application threat protection for medium to large enterprises, application service providers, and Security-as-a-Service (SaaS) providers. FortiWeb Web Application Firewalls protect web-based applications and internet-facing data from attack and security breaches. Using advanced techniques, FortiWeb provides bidirectional protection against malicious sources, denial of service attacks and sophisticated threats such as SQL injection, cross-site scripting, buffer overflows, file inclusion, and cookie poisoning. Only the functions described in the collaborative Protection Profile for Network Devices have been evaluated.

The TOE is a combined software and hardware TOE.

1.5 TOE DESCRIPTION

1.5.1 Physical Scope

The TOE is the Fortinet FortiWeb 3000E and 4000E devices, used with the Fortinet Entropy Token. Figure 1 shows deployment in Offline Mode. The FortiAnalyzer unit serves as the audit server in the evaluated configuration.

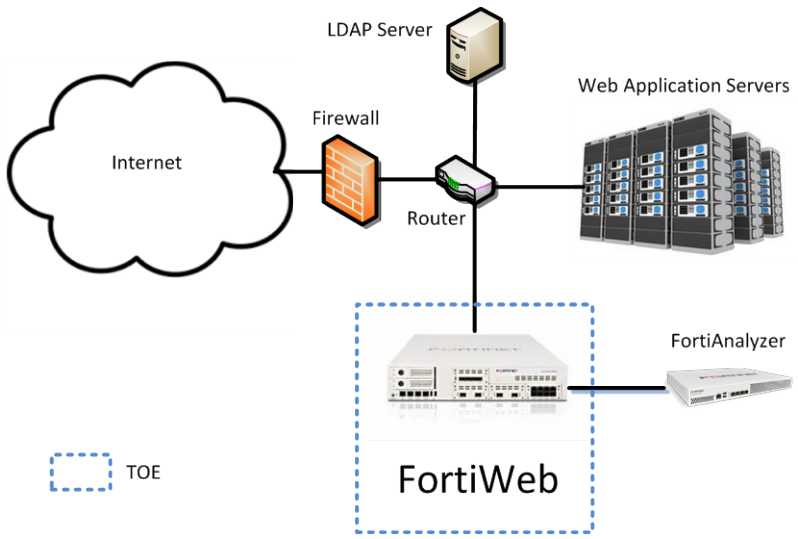

Figure 1 – FortiWeb Deployment Diagram

The physical scope of the TOE includes the FWB-3000E and FWB-4000E appliance with firmware version 5.6, as well as a Fortinet hardware-based noise source. The hardware–based Fortinet Entropy Token noise source is connected to the appliance via a USB port.

Each FortiWeb appliance (together with noise source) is a stand-alone unit. Each of the appliances provides a local interface and network interfaces for administrator access. Both appliances provide the same security functionality. The following table details the evaluated appliance models.

Model	Designation	Details
FortiWeb 3000E	FWB-3000E	Dual Intel Xeon E5-2620v3, 2.4G, 6 cores
FortiWeb 4000E	FWB-4000E	Dual Intel Xeon E5-2650v3, 2.3G, 10 cores

Table 1 –TOE Hardware and Software

1.5.2 TOE Guidance

The TOE includes the following guidance documentation:

- FortiWeb 3000E QuickStart Guide, 22 December 2016
- FortiWeb 4000E QuickStart Guide, 22 December 2016
- FortiWeb Administration Guide Version 5.6, 9 February 2017
- FortiWeb Log Reference Version 5.5, 5 April 2016
- FortiWeb CLI Reference Version 5.6, 23 September 2016
- Common Criteria Compliant Operation for FortiWeb 5.6, 9 November 2017

1.5.3 Logical Scope

The logical boundary of the TOE includes all interfaces and functions within the physical boundary. The logical boundary of the TOE may be broken down by the security function classes described in Section 6. Table 2 summarizes the logical scope of the TOE.

Functional Classes	Description
Security Audit	The TOE generates audit records for administrative activity, security related configuration changes, cryptographic key changes and startup and shutdown of the audit functions. The audit events are associated with the administrator who performs them if applicable. The audit records are transmitted over a TLS-protected link to an external audit server in the IT environment for storage.
Cryptographic Support	The TOE provides cryptographic functions (key generation, key establishment, key destruction, cryptographic operation) to secure remote administrative sessions over Hypertext Transfer Protocol Secure (HTTPS)/Transport Layer Security (TLS) and the connection to the audit server using TLS. The connection to the authentication server is protected using TLS 1.2.

Functional Classes	Description
Identification and Authentication	Users must identify and authenticate prior to TOE access using the TOE's password-based logon mechanism that enforces minimum strength requirements, or an LDAP server. (Both authentication mechanisms are evaluated.) The login process ensures that passwords are obscured when entered. The TOE also validates and authenticates X.509 certificates for all certificate use.
Security Management	The TOE provides management capabilities via a web UI, accessed over HTTPS and a CLI accessed locally. Management functions allow the administrators to configure and update the system, manage users and user access and configure cryptographic functionality.
Protection of the TSF	The TOE prevents the reading of plaintext passwords and keys. The TOE provides a reliable timestamp for its own use. To protect the integrity of its security functions, the TOE implements a suite of self-tests at startup, and ensures that updates to the TOE software may be verified using a digital signature.
TOE Access	The TOE monitors local and remote sessions for inactivity and terminates the session when a threshold time period is reached. An advisory notice is displayed at the start of each session.
Trusted Path/Channel	The TSF provides a TLS protected link for trusted communication between itself and an audit server and between itself and an authentication server. The TOE implements HTTPS for protection of communications between itself and remote users of the web UI.

Table 2 – Logical Scope of the TOE

1.5.4 Evaluated Configuration

The following configuration options were selected for the evaluated configuration:

- A Fortinet Entropy token must be used to provide the required entropy.
- The system must be configured in 'FIPS-CC' mode during installation.
- An LDAP authentication server must be configured and the link to this server must be configured to use TLS. The LDAP authentication server is used for LDAP authentication only, and is not required when using local authentication.
- Both local and remote logging must be enabled. The logging level must be set to 'Information' and the following event types must be selected for logging:
    - When configuration has changed

    - Admin login/logout event
    - System activity event
    - Update
- The remote logging destination must point to the FortiAnalyzer device and must specify a TLS security profile with a minimum encryption strength of 128 bits.
- The Enable Strong Passwords setting is selected for administrative users.
- The TOE access banner is configured by customizing the authentication page for web UI access and enabling the pre-login-banner for CLI access.
- Administrative access is restricted to HTTPS. SSH, SNMP, Telnet and HTTP are not to be selected for administrative access.

2 CONFORMANCE CLAIMS

2.1 COMMON CRITERIA CONFORMANCE CLAIM

Where applicable, this Security Target is conformant to Version 3.1 of Common Criteria for Information Technology Security Evaluation according to:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components CCMB-2012-09-003, Version 3.1, Revision 4, September 2012

As follows:

- CC Part 2 extended
- CC Part 3 conformant

The Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012 has to be taken into account.

2.2 ASSURANCE PACKAGE CLAIM

This Security Target claims conformance to the Security Assurance Requirements (SARs) claimed in the collaborative Protection Profile for Network Devices (v1.0, 27-Feb-2015).

2.3 PROTECTION PROFILE CONFORMANCE CLAIM

The TOE for this ST claims exact conformance to the collaborative Protection Profile for Network Devices (v1.0, 27-Feb-2015). The following technical decisions have been considered and applied where required: TD 0090, TD 0094, TD 0095, TD 0096, TD 0111, TD 0112, TD 0113, TD 0114, TD 0115, TD 0116, TD 0117[1], TD 0125, TD 0126, TD 0130, TD 0143, TD 0150, TD 0151, TD 0152, TD 0153, TD 0154, TD 0155, TD 0156, TD0160 TD 0164, TD 0165, TD 0168, TD 0169, TD 0170, TD 0181, TD 0182, TD 0183, TD 0184, TD 0185, TD 0186, TD 0187, TD 0188, TD 0189, TD 0191, TD 0195, TD0199, TD0200, TD 0201, TD0223, TD0224, TD0225, TD0226, TD0227, TD0228, and TD0235.

[1] TD0093 was not considered as it was superseded by TD0117.

3 SECURITY PROBLEM DEFINITION

3.1 THREATS

Table 3 lists the threats addressed by the TOE.

Threat	Description
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.
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.

Threat	Description
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 include 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	A component of the network device may fail during start-up or during operations causing a compromise or failure in the security functionality of the network device, leaving the device susceptible to attackers.

Table 3 – Threats

3.2 ORGANIZATIONAL SECURITY POLICIES

Organizational Security Policies (OSPs) are security rules, procedures, or guidelines imposed on the operational environment. Table 4 lists the OSPs that are presumed to be imposed upon the TOE or its operational environment by an organization that implements the TOE in the Common Criteria evaluated configuration.

OSP	Description
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.

Table 4 – Organizational Security Policies

3.3 ASSUMPTIONS

The assumptions required to ensure the security of the TOE are listed in Table 5.

Assumptions	Description
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 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).

Assumptions	Description
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.
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.

Table 5 – Assumptions

4 SECURITY OBJECTIVES

The purpose of this section is to identify and describe the security objectives that are addressed by the operational environment. Table 6 identifies and describes these objectives.

Security Objective	Description
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	TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.
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.

Table 6 – Security Objectives for the Operational Environment

5 EXTENDED COMPONENTS DEFINITION

This section specifies the extended Security Functional Requirements (SFRs) used in this ST and defined in the PP. The following table identifies the extended SFRs that have been created to address additional security features of the TOE:

Class	Family	Component
FAU: Security Audit	FAU_STG_EXT: Event Storage	FAU_STG_EXT.1: Protected Audit Event Storage
		FAU_STG_EXT.3: Display warning for local storage space
FCS: Cryptographic Support	FCS_HTTPS_EXT: HTTPS Protocol	FCS_HTTPS_EXT.1: HTTPS Protocol
	FCS_RBG_EXT: Random Bit Generation	FCS_RBG_EXT.1: Random Bit Generation
	FCS_TLSC_EXT: TLS Client Protocol	FCS_TLSC_EXT.2: TLS Client Protocol with authentication
	FCS_TLSS_EXT: TLS Server Protocol	FCS_TLSS_EXT.1: TLS Server Protocol
FIA: Identification and Authentication	FIA_PMG_EXT: Password Management	FIA_PMG_EXT.1: Password Management
	FIA_UIA_EXT: User Identification and Authentication	FIA_UIA_EXT.1: User Identification and Authentication
	FIA_UAU_EXT: User Authentication	FIA_UAU_EXT.2: Password-Based Authentication Mechanism
	FIA_X509_EXT: Authentication Using X.509 Certificates	FIA_X509_EXT.1: Certificate Validation
		FIA_X509_EXT.2: Certification Authentication
		FIA_X509_EXT.3: Certificate Requests

Class	Family	Component
FPT: Protection of the TSF	FPT_APW_EXT: Protection of Administrator Passwords	FPT_APW_EXT.1: Protection of Administrator Passwords
	FPT_SKP_EXT: Protection of TSF Data	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)
	FPT_TST_EXT: TSF Self Test	FPT_TST_EXT.1: TSF Testing
	FPT_TUD_EXT: Trusted Update	FPT_TUD_EXT.1: Trusted Update
FTA: TOE Access	FTA_SSL_EXT: TSF-Initiated Session Locking	FTA_SSL_EXT.1: TSF-Initiated Session Locking

Table 7 – Extended Security Functional Requirements

5.1 SECURITY FUNCTIONAL REQUIREMENTS

5.1.1 Class FAU: Security Audit

5.1.1.1 FAU_STG_EXT Protected Audit Event Storage

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 Leveling

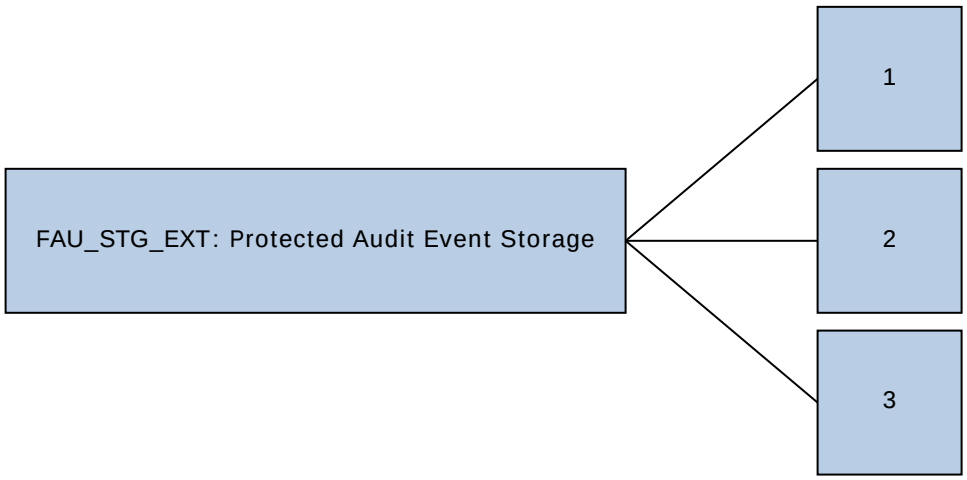

Figure 2 – Protected Audit Event Storage Component Leveling

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 Display warning for local storage space requires the TSF to generate a warning before the audit log becomes full.

Management: FAU_STG_EXT.1

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

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

a) No audit necessary.

Note: FAU_STG_EXT.2 is not being claimed in this ST, therefore the extended component definition has not been provided.

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.

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

FAU_STG_EXT.1.3 The TSF shall [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.

FAU_STG_EXT.3 Display warning for local storage space

Hierarchical to:	No other components
Dependencies:	No dependencies.

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

5.1.2 Class FCS: Cryptographic Support

5.1.2.1 FCS_HTTPS_EXT 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 Leveling

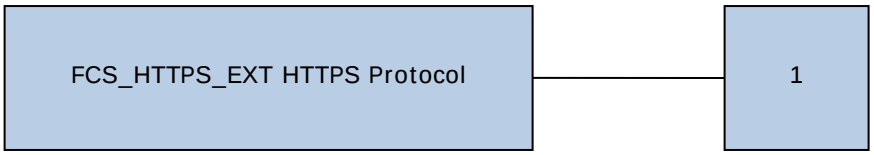

Figure 3 – HTTPS Protocol Component Leveling

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.

FCS_HTTPS_EXT.1 HTTPS Protocol

Hierarchical to:	No other components
Dependencies:	FCS_TLS_EXT.1 TLS 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	The TSF shall establish a connection only if [selection: *the peer presents a valid certificate during handshake, the peer initiates handshake*].

5.1.2.2 FCS_RBG_EXT Random Bit Generation

Family Behaviour

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

Component Leveling

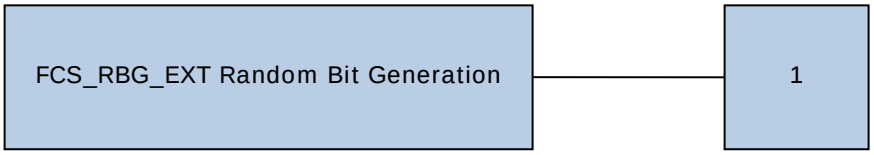

Figure 4 – Random Bit Generation Component Leveling

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

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 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.

5.1.2.3 FCS_TLSC_EXT TLS Client Protocol with authentication

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 Leveling

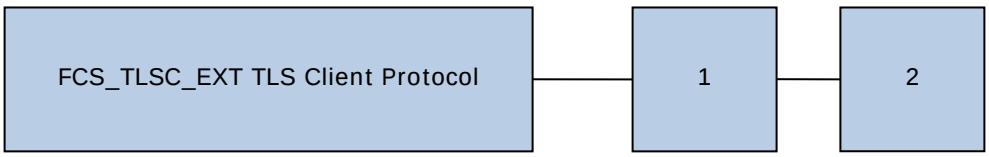

Figure 5 – TLS Client Protocol Component Leveling

FCS_TLSC_EXT.1 TLS Client requires that the client side of TLS be implemented as specified.

FCS_TLSC_EXT.2 TLS Client requires the client side of the TLS implementation include mutual authentication.

Management: FCS_TLSC_EXT.1, FCS_TLSC_EXT.2

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

a) There are no management activities foreseen.

Audit: FCS_TLSC_EXT.1, FCS_TLSC_EXT.2

The following actions should be auditable 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

Note: FCS_TLSC_EXT.1 is not being claimed in this ST, therefore the extended component definition has not been provided.

FCS_TLSC_EXT.2 TLS Client Protocol with Authentication

Hierarchical to: No other components

Dependencies: FCS_COP.1(1) Cryptographic Operation (AES Data encryption/decryption

FCS_COP.1(2) Cryptographic Operation (Signature Verification)

FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)

FCS_RBG_EXT.1 Random Bit Generation

FCS_TLSC_EXT.2.1 The TSF shall implement [selection: TLS 1.2 *(RFC 5246), TLS 1.1 (RFC 4346)*] supporting the following ciphersuites:

[selection:

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*

- o *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- o *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- o *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- o *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- o *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- o *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

].

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

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

FCS_TLSC_EXT.2.4 The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [assignment: *List of supported curves including an option for 'none'*].

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

5.1.2.4 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 Leveling

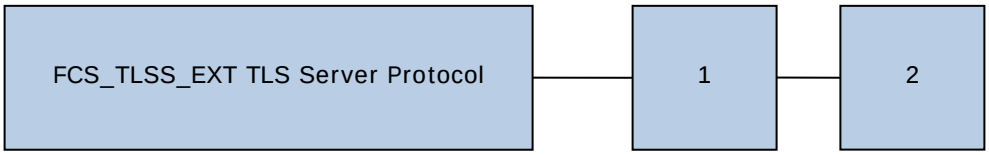

Figure 6 – TLS Server Protocol Component Leveling

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 auditable 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

Note: FCS_TLSS_EXT.2 is not being claimed in this ST, therefore the extended component definition has not been provided.

FCS_TLSS_EXT.1 TLS Server Protocol

Hierarchical to: No other components

Dependencies: FCS_CKM.1 Cryptographic Key Generation
FCS_COP.1(1) Cryptographic Operation (AES Data encryption/decryption
FCS_COP.1(2) Cryptographic Operation (Signature Verification)
FCS_COP.1(3) Cryptographic Operation (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)*] supporting the following ciphersuites:

[selection:

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*

- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289]*.

FCS_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*].

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]].

5.1.3 Class FIA: Identification and Authentication

5.1.3.1 FIA_PMG_EXT Password Management

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 Leveling

Figure 7 – Password Management Component Leveling

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.

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 settable by the Security Administrator, and support passwords of 15 characters or greater.

5.1.3.2 FIA_UIA_EXT User Identification and Authentication

Family Behaviour

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

Component Leveling

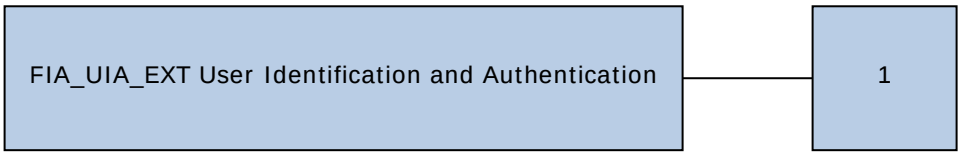

Figure 8 - User Identification and Authentication Component Leveling

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.1

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)

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, [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.

5.1.3.3 FIA_UAU_EXT User Authentication

Family Behaviour

Provides for a locally based administrative user authentication mechanism.

Component Leveling

Figure 9 - Password-Based Authentication Mechanism Component Leveling

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.

FIA_UAU_EXT.2 Password-based Authentication Mechanism

Hierarchical to: No other components

Dependencies: None

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [selection: *[assignment: other authentication mechanism(s)], none*] to perform administrative user authentication.

5.1.3.4 FIA_X509_EXT Authentication Using X.509 Certificates

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 Leveling

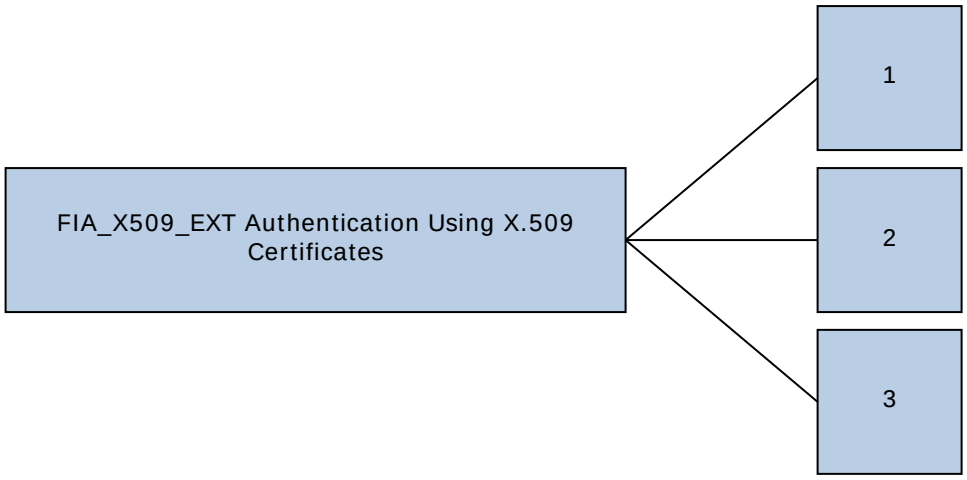

Figure 10 - Authentication Using X.509 Certificates Component Leveling

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.

FIA_X509_EXT.1 X.509 Certificate Validation

Hierarchical to: No other components

Dependencies: No other components

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

- RFC 5280 certificate validation and certificate path validation.

- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [selection: the Online Certificate Status Protocol (OCSP) as specified in RFC 2560, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, a Certificate Revocation List (CRL) as specified in RFC 5759 Section 5].
- 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*].

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

FIA_X509_EXT.2 X.509 Certificate Authentication

Hierarchical to: No other components

Dependencies: No other components

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [selection: *IPsec, TLS, HTTPS, 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*].

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*].

FIA_X509_EXT.3 X.509 Certificate Requests

Hierarchical to: No other components

Dependencies: No other components

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 [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.

5.1.4 Class FPT: Protection of the TSF

5.1.4.1 FPT_APW_EXT Protection of Administrator Passwords

Family Behaviour

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

Component Leveling

Figure 11 - Protection of Administrator Passwords Component Leveling

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.

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 passwords in non-plaintext form.

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

5.1.4.2 FPT_SKP_EXT Protection to TSF Data

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 Leveling

Figure 12 - Protection of TSF Data Component Leveling

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.

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.

5.1.4.3 FPT_TST_EXT TSF Self Test

Family Behaviour

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

Component Leveling

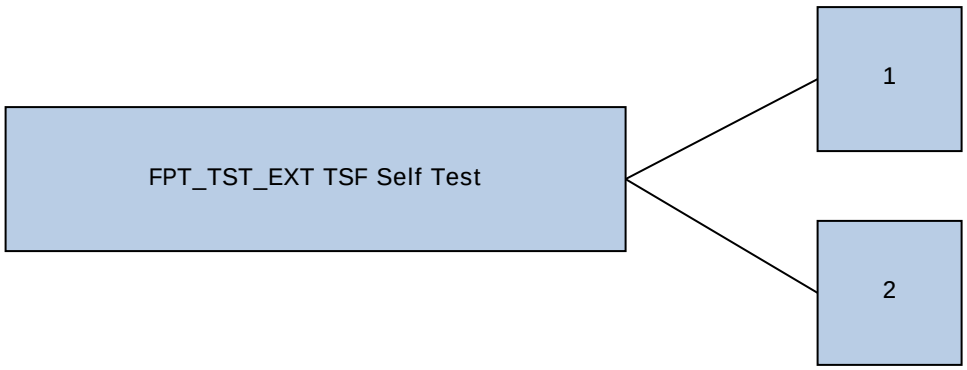

Figure 13 - TSF Self Test Component Leveling

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

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

a) No management functions

Audit: FPT_TST_EXT.1

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

a) Indication that TSF self test was completed

Note: FPT_TST_EXT.2 is not being claimed in this ST, therefore the extended component definition has not been provided.

FPT_TST_EXT.1 TSF Testing

Hierarchical to: No other components

Dependencies: None

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*].

5.1.4.4 FPT_TUD_EXT Trusted Update

Family Behaviour

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

Component Leveling

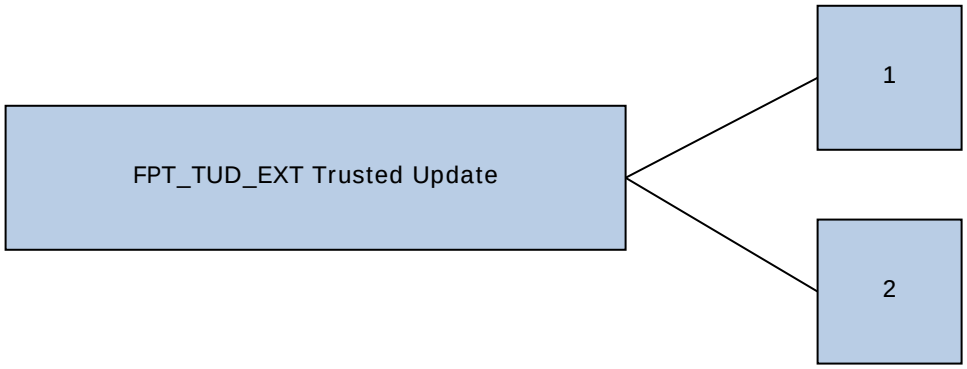

Figure 14 - Trusted Update Component Leveling

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(2)) 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

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

Note: FPT_TUD_EXT.2 is not being claimed in this ST, therefore the extended component definition has not been provided.

FPT_TUD_EXT.1	**Trusted Update**
Hierarchical to:	No other components
Dependencies:	FCS_COP.1(1) Cryptographic Operation (for cryptographic signature), or FCS_COP.3 Cryptographic Operation (for cryptographic hashing)

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 [selection: the most recently installed version of the TOE firmware/software; no other TOE firmware/software version].

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

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.

5.1.5 Class FTA: TOE Access

5.1.5.1 FTA_SSL_EXT TSF-Initiated Session Locking

Family Behaviour

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

Component Leveling

Figure 15 - TSF-Initiated Session Locking Component Leveling

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:

a) 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:

a) Any attempts at unlocking an interactive session.

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 user'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.

5.2 SECURITY ASSURANCE REQUIREMENTS

This ST does not include extended Security Assurance Requirements.

Note that the following refinement has been made to ASE_TSS.1.1:

ASE_TSS.1.1C Refinement: The TOE summary specification shall describe how the TOE meets each SFR. **In the case of entropy analysis the TSS is used in conjunction with, including required supplementary information on Entropy.**

6 SECURITY REQUIREMENTS

Section 6 provides security functional and assurance requirements that must be satisfied by a compliant TOE. These requirements consist of functional components from Part 2 of the CC, extended requirements, and assurance components from Part 3 of the CC.

6.1 CONVENTIONS

The conventions used in the collaborative Protection Profile for Network Devices are described as:

- Assignment: Indicated with italicized text[2]
- Refinement made by PP author: Indicated with bold text and strikethroughs, if necessary[3]
- Selection: Indicated with underlined text[4]
- Assignment within a Selection: Indicated with italicized and underlined text[5]
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/"

Some selections and assignments are bracketed, but only where indicated in the PP.

[2] Where the Protection Profile uses bolded, italicized text to indicate assignments, this has been reflected in the ST.

[3] Some, but not all refinements to SFRs from Common Criteria Part 2 are indicated. Where the Protection Profile uses the word 'Refinement' in bold letters at the beginning of the SFR, this has also been reflected in the ST.

[4] The collaborative Protection Profile for Network Devices and related Technical Decisions use italicized text to indicate most selections; this has been reflected in this ST. Where the Protection Profile uses bolded, italicized text to indicate selections, this has been reflected in the ST. Where the Protection Profile uses bolded, underlined text to indicate selections, this has also been reflected in the ST. Where the Protection Profile uses underlined, italicized text to indicate selections, this has been reflected in the ST. Where the Protection Profile uses underlined text, this has been reflected in the ST.

[5] Most assignments within selections are indicated with italicized text, to be consistent with the collaborative Protection Profile for Network Devices. Where the Protection Profile uses bolded, underlined text to indicate assignments within selections, this has also been reflected in the ST.

6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for this ST consist of the following components from Part 2 of the CC and extended components defined in Section 5, summarized in Table 8 - Summary of Security Functional Requirements.

Class	Identifier	Name
Security Audit (FAU)	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG.1	Protected audit trail storage
	FAU_STG_EXT.1	Protected Audit Event Storage
	FAU_STG_EXT.3	Display warning for local storage space
Cryptographic Support (FCS)	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1(1)	Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1(2)	Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1(3)	Cryptographic Operation (Hash algorithm)
	FCS_COP.1(4)	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_TLSC_EXT.2	TLC Client Protocol with authentication
	FCS_TLSS_EXT.1	TLS Server Protocol
Identification and Authentication (FIA)	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

Class	Identifier	Name
	FIA_X509_EXT.1	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
Security Management (FMT)	FMT_MOF.1(1)/ TrustedUpdate	Management of security functions behaviour
	FMT_MOF.1(2)/ Audit	Management of security functions behaviour
	FMT_MOF.1(1)/ AdminAct	Management of security functions behaviour
	FMT_MOF.1(2)/ AdminAct	Management of security functions behaviour
	FMT_MTD.1	Management of TSF Data
	FMT_MTD.1/ AdminAct	Management of TSF data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF (FPT)	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM.1	Reliable Time Stamps
	FPT_TST_EXT.1(1)	TSF testing
	FPT_TST_EXT.1(2)	TSF testing
	FPT_TUD_EXT.1	Trusted update
TOE Access (FTA)	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
Trusted	FTP_ITC.1	Inter-TSF trusted channel

Class	Identifier	Name
path/channels (FTP)	FTP_TRP.1	Trusted Path

Table 8 – Summary of Security Functional Requirements

6.2.1 Security Audit (FAU)

6.2.1.1 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 shutdown of the audit functions;

b) All auditable events for the not specified level of audit; and

c) *All administrative actions comprising:*

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
- *Security related configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- *Starting and stopping services (if applicable)*
- *[*no other actions*]*;

d) *Specifically defined auditable events listed in Table 9.*

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 9*.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.

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

Requirement	Auditable Events	Additional Audit Record Contents
TrustedUpdate	manual update	
FMT_MOF.1(2)/ Audit	Modification of the behaviour of the handling of audit data.	None.
FMT_MOF.1(1)/ AdminAct	Modification of the behaviour of the TSF.	None.
FMT_MOF.1(2)/ AdminAct	Starting and stopping of services.	None.
FMT_MTD.1	All management activities of TSF data.	None.
FMT_MTD.1(1)/ AdminAct	Modification, deletion, generation/import of cryptographic keys.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	No additional information.
FPT_STM.1	Changes to time.	The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.

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

Table 9 – Security Functional Requirements and Auditable Events

6.2.1.2 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.

6.2.1.3 FAU_STG.1 Protected audit trail storage

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

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

6.2.1.4 FAU_STG_EXT.1 Protected Audit Event Storage

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

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

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

6.2.1.5 FAU_STG_EXT.3 Display warning for local storage space

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

6.2.2 Cryptographic Support (FCS)

6.2.2.1 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;***
- ***FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1***

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

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

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

6.2.2.3 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 [single overwrite consisting of [zeroes]];*
- *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 [*
    - o *logically addresses the storage location of the key and performs a [[administrator selectable number between 1 and 10]-pass] overwrite consisting of [zeroes]]*

that meets the following: *No Standard*.

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

FCS_COP.1.1(1) The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in [CBC] mode*] and cryptographic key sizes *[128, 256 bits]* that meet the following: *AES as specified in ISO 18033-3, [CBC as specified in ISO 10116]*.

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

FCS_COP.1.1(2) 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]*

]

that meets 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*,

].

6.2.2.6 FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)

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

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

FCS_COP.1.1(4) The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm *HMAC-SHA-1, HMAC-SHA-256* and cryptographic key sizes [*160, 256*] **and message digest sizes [160, 256] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

6.2.2.8 FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3 The TSF shall establish the connection only if [*the peer initiates handshake*].

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

6.2.2.10 FCS_TLSC_EXT.2 TLS Client Protocol with authentication

FCS_TLSC_EXT.2.1 The TSF shall implement [TLS 1.2 *(RFC 5246), TLS 1.1 (RFC 4346)*] supporting the following ciphersuites:

- o *[TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- o *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- o *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- o *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- o *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- o *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- o *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*
- o *TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246]*.

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

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

FCS_TLSC_EXT.2.4 The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: *[none]* and no other curves.

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

6.2.2.11 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)*] supporting the following ciphersuites:

- o *[TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- o *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- o *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- o *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- o *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- o *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- o *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*
- o *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]].

6.2.3 Identification and Authentication (FIA)

6.2.3.1 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: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”]*;

b) *Minimum password length shall be settable by the Security Administrator, and shall support passwords of 15 characters or greater.*

6.2.3.2 FIA_UIA_EXT.1 User Identification and Authentication

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

- Display the warning banner in accordance with FTA_TAB.1;
- [*no other actions*]

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

6.2.3.3 FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

6.2.3.4 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.

6.2.3.5 FIA_X509_EXT.1 X.509 Certificate validation

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

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*

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

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

6.2.3.6 FIA_X509_EXT.2 X.509 Certificate Authentication

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

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

6.2.3.7 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.

6.2.4 Security Management (FMT)

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

6.2.4.1 FMT_MOF.1(2)/Audit Management of security functions behaviour

FMT_MOF.1.1(2)/Audit The TSF shall restrict the ability to determine the behaviour of, modify the behaviour of the functions *handling of audit data* to *Security Administrators*.

6.2.4.2 FMT_MOF.1(1)/AdminAct Management of security functions behaviour

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

6.2.4.3 FMT_MOF.1(2)/AdminAct Management of security functions behaviour

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

6.2.4.4 FMT_MTD.1 Management of TSF Data

Hierarchical to: No other components.
Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

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

6.2.4.5 FMT_MTD.1/AdminAct Management of TSF data

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

6.2.4.6 FMT_SMF.1 Specification of Management Functions

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

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

6.2.4.7 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator*.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

6.2.5 Protection of the TSF (FPT)

6.2.5.1 FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

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

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

6.2.5.3 FPT_STM.1 Reliable Time Stamps

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

6.2.5.4 FPT_TST_EXT.1(1) TSF testing

FPT_TST_EXT.1.1(1) The TSF shall run a suite of self tests [*during initial start-up*] to demonstrate the correct operation of [

- *Firmware integrity test using RSA signatures*
- *Configuration integrity test using HMAC SHA-256*
- *Triple-DES, CBC mode, encrypt known answer test*
- *Triple-DES, CBC mode, decrypt known answer test*
- *AES, CBC mode, encrypt known answer test*
- *AES, CBC mode, decrypt known answer test*
- *HMAC SHA-1 known answer test*
- *SHA-1 known answer test (tested as part of HMAC SHA-1 known answer test)*
- *HMAC SHA-256 known answer test*
- *SHA-256 known answer test (tested as part of HMAC SHA-256 known answer test)*
- *RSA signature generation known answer test*
- *RSA signature verification known answer test*
- *DRBG known answer test*
- *CPU and BIOS test*
- *Boot loader test*

].

6.2.5.5 FPT_TST_EXT.1(2) TSF testing

FPT_TST_EXT.1.1(2) The TSF shall run a suite of self tests [*periodically during normal operation*] to demonstrate the correct operation of [

- *Continuous NDRNG test*
- *Continuous DRBG test*
- *RSA pairwise consistency test*
- *Configuration integrity test using HMAC SHA-256*
- *Firmware load test using RSA signatures*
- *Firmware integrity test using RSA signatures*
- *Configuration integrity test using HMAC SHA-256*
- *Triple-DES, CBC mode, encrypt known answer test*
- *Triple-DES, CBC mode, decrypt known answer test*
- *AES, CBC mode, encrypt known answer test*
- *AES, CBC mode, decrypt known answer test*
- *HMAC SHA-1 known answer test*
- *SHA-1 known answer test (tested as part of HMAC SHA-1 known answer test)*
- *HMAC SHA-256 known answer test*
- *SHA-256 known answer test (tested as part of HMAC SHA-256 known answer test)*
- *RSA signature generation known answer test*
- *RSA signature verification known answer test*
- *DRBG known answer test*

6.2.5.6 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 [the most recently installed version of the TOE firmware/software].

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

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

6.2.6 TOE Access (FTA)

6.2.6.1 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.

6.2.6.2 FTA_SSL.3 TSF-initiated Termination

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

6.2.6.3 FTA_SSL.4 User-initiated Termination

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

6.2.6.4 FTA_TAB.1 Default TOE Access Banners

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

6.2.7 Trusted Path/Channels (FTP)

6.2.7.1 FTP_ITC.1 Inter-TSF trusted channel

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

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [***transfer of audit records, validation of administrator credentials***].

6.2.7.2 FTP_TRP.1 Trusted Path

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

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

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

6.3 DEPENDENCY RATIONALE

Table 10 identifies the Security Functional Requirements (SFRs) from Part 2 of the CC, extended SFRs identified in Section 5, and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency.

SFR	Dependency	Dependency Satisfied	Rationale
FAU_GEN.1	FPT_STM.1	✓	
FAU_GEN.2	FAU_GEN.1	✓	
	FIA_UID.1	✓	The dependency is satisfied by FIA_UID_EXT.1, which covers the identification requirement for the collaborative Protection Profile for Network Devices
FAU_STG_EXT.1	FAU_GEN.1	✓	
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1	✓	Satisfied by FCS_COP.1
	FCS_CKM.4	✓	
FCS_CKM.2	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Satisfied by FCS_CKM.1
	FCS_CKM.4	✓	
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Satisfied by FCS_CKM.1
FCS_COP.1(1)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	No	This dependency is not met in the claimed Protection Profile. It may be considered to be resolved by the protocol utilizing the session algorithms
	FCS_CKM.4	✓	
FCS_COP.1(2)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Satisfied by FCS_CKM.1
	FCS_CKM.4	✓	

SFR	Dependency	Dependency Satisfied	Rationale
FCS_COP.1(3)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	No	FCS_COP.1(3) has no key management dependencies in the claimed Protection Profile
	FCS_CKM.4	✓	
FCS_COP.1(4)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	No	This dependency is not met in the claimed Protection Profile. It may be considered to be resolved by the protocol utilizing the session algorithms
	FCS_CKM.4	✓	
FCS_HTTPS_ EXT.1	FCS_TLS_ EXT.1	✓	This dependency is met by FCS_TLSC_EXT.2 and FCS_TLSS_EXT.1.
FCS_RBG_EXT.1	None	N/A	
FCS_TLSC_EXT.2	FCS_COP.1(1)	✓	
	FCS_COP.1(2)	✓	
	FCS_COP.1(3)	✓	
	FCS_RBG_ EXT.1	✓	
FCS_TLSS_EXT.1	FCS_CKM.1	✓	
	FCS_COP.1(1)	✓	
	FCS_COP.1(2)	✓	
	FCS_COP.1(3)	✓	
	FCS_RBG_ EXT.1	✓	
FDP_RIP.2	None	N/A	
FIA_PMG_EXT.1	None	N/A	
FIA_UIA_EXT.1	FTA_TAB.1	✓	
FIA_UAU_EXT.1	None	N/A	

SFR	Dependency	Dependency Satisfied	Rationale
FIA_UAU.7	FIA_UAU.1	✓	The dependency is satisfied by FIA_UAU_EXT.2, which covers the authentication requirement for the collaborative Protection Profile for Network Devices
FIA_X509_EXT.1	None	N/A	
FIA_X509.EXT.2	None	N/A	
FIA_X509.EXT.3	None	N/A	
FMT_MOF.1(1)	FMT_SMR.1	✓	
	FMT_SMF.1	✓	
FMT_MTD.1	FMT_SMR.1	✓	The dependency is satisfied by FMT_SMR.2, which is hierarchical to FMT_SMR.1
	FMT_SMF.1	✓	
FMT_SMF.1	None	N/A	
FMT_SMR.2	FIA_UID.1	✓	The dependency is satisfied by FIA_UAU_EXT.2, which covers the authentication requirement for the collaborative Protection Profile for Network Devices
FPT_APW_EXT.1	None	N/A	
FPT_SKP_EXT.1	None	N/A	
FPT_STM.1	None	N/A	
FPT_TST_EXT.1	None	N/A	
FPT_TUD_EXT.1	FCS_COP.1(1) or FCS_COP.1(3)	✓	The dependency is satisfied by FCS_COP.1(1) which covers cryptographic signature
FTA_SSL_EXT.1	FIA_UAU.1	✓	The dependency is satisfied by FIA_UAU_EXT.2, which covers the authentication requirement for the collaborative Protection Profile for Network Devices
FTA_SSL_EXT.3	None	N/A	

SFR	Dependency	Dependency Satisfied	Rationale
FTA_SSL_EXT.4	None	N/A	
FTA_TAB.1	None	N/A	
FTP_ITC.1	None	N/A	
FTP_TRP.1	None	N/A	

Table 10 – Functional Requirement Dependencies

6.4 TOE SECURITY ASSURANCE REQUIREMENTS

The TOE security assurance requirements for this ST conform to those described in the claimed Protection Profile.

The security assurance requirements are summarized in the following table:

Assurance Class	Assurance Components	
	Identifier	Name
Development (ADV)	ADV_FSP.1	Basic functional specification
Guidance Documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-Cycle Support (ALC)	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Security Target Evaluation (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

Assurance Class	Assurance Components	
	Identifier	**Name**
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability Assessment (AVA)	AVA_VAN.1	Vulnerability survey

Table 11 – Security Assurance Requirements

Note that the following refinement has been made to ASE_TSS.1.1:

ASE_TSS.1.1C **Refinement:** The TOE summary specification shall describe how the TOE meets each SFR. **In the case of entropy analysis the TSS is used in conjunction with, including required supplementary information on Entropy.**

7 TOE SUMMARY SPECIFICATION

This section provides a description of the security functions and assurance measures of the TOE that meet the TOE security requirements.

7.1 SECURITY AUDIT

When a security related event occurs, an audit record is written to the audit logs. The TOE logs the startup and shutdown of the TOE, which includes the audit services.

In the evaluated configuration, the TOE is configured to simultaneously record audit messages both locally and remotely. The behaviour of the TOE when audit logs become full is configurable. By default, the TOE will log locally and will block further traffic from occurring should the local storage become exhausted. Guidance is provided to the administrator to modify this behavior to overwrite the oldest audit logs when the threshold of memory capacity has been reached. The most recent audit records are stored locally. Only authorized administrators may view these records, and no capability to modify the records is provided. Logs begin to be overwritten when the remaining free disk space reaches 20 per cent of disk capacity. Two 2 terabyte (2TB) drives are available for storage; however, the disk capacity reserved for logging ultimately depends on the configuration of the unit. In the evaluated configuration, the drives are mirrored, so 2 terabytes of storage space is available. The administrator may select a threshold level between 60 and 90 percent, beyond which the log disk usage triggers an event log entry.

Remote storage of audit events is configured in the evaluated configuration. Records are sent in real-time to one or more configured FortiAnalyzer audit servers, and the link to the FortiAnalyzer device is protected using TLS. The audit events are transmitted as they are generated; a cache storing up to 32 kilobytes of data is maintained to address temporary outages in communication with remote audit servers. If the cache is exhausted the oldest records are overwritten by the new records.

The TOE is capable of logging messages to the audit log for interactions which occur via from the local console or over HTTPS. These events include attempts to establish or terminate sessions and errors detected during decryption or validation of data.

TOE Security Functional Requirements addressed: FAU_GEN.1, FAU_GEN.2, FAU_STG.1, FAU_STG_EXT.1, FAU_STG_EXT.3.

7.2 CRYPTOGRAPHIC SUPPORT

7.2.1.1 Cryptographic Key Generation and Key Establishment

The TOE supports Rivest-Shamir-Adleman (RSA) using cryptographic 2048-bit key sizes and Finite Field Cryptography (FFC) schemes using cryptographic 2048-bit key sizes. RSA and FFC are used in support of TLS. For RSA-based key establishment, the TOE acts as a sender when establishing TLS connections with

remote administrators, and acts as a recipient when establishing TLS connections with the audit and authentication servers. When a decryption error occurs, the TOE does not reveal the particular error that occurred, in accordance with NIST SP 800-56B.

Table 12 shows the sections of National Institute of Standards and Technology (NIST) Special Publication (SP) 800-56B that are met by the implementation.

SP 800-56B Section	Requirement	Compliant
5.9 Key Derivation Functions for Key Establishment Schemes	Shall	Yes
6.3.1 RSAKPG1 Family: rsakpg1-basic RSA Key Pair Generation with a Fixed Public Exponent	Shall	Yes
6.3.2 RSAKPG2 Family: rsakpg1-basic RSA Key Pair Generation with a Random PublicExponent	Shall	Yes
6.4 Assurances of Validity	Shall	Yes
6.4.1 Assurance of Key Pair Validity	Shall	Yes
6.4.2 Recipient Assurances of Public Key Validity	Shall	Yes
8 Key Agreement Schemes	Shall	Yes
9 IFC based Key Transport Schemes	Shall	Yes

Table 12 – SP 800-56B Conformance

TOE Security Functional Requirements addressed: FCS_CKM.1, FCS_CKM.2

7.2.2 Cryptographic Key Destruction

Key materials held in volatile memory are zeroized after use by overwriting the key storage area with zeroes. The overwrite is read and verified. Keys held in flash memory may be destroyed using a Command Line Interface (CLI) command to overwrite the entire flash memory an administrator specified number of times (between 1 and 10) with zeroes. This command is used when a device is reset or taken out of operation.

The table below shows the origin, storage location and destruction details for all plaintext keys. Unless otherwise stated, the keys are generated by the TOE.

Type / Description	Generation / Algorithm	Storage	Zeroization

Type / Description	Generation / Algorithm	Storage	Zeroization
RSA private key used for TLS	RSA (2048 bits)	Stored in flash memory Held in RAM in plaintext	The key is overwritten with zeroes on format of the flash storage The plaintext key is overwritten with zeroes upon termination of the session or reboot of the appliance
RSA public key used for TLS	RSA (2048 bits)	Certificate stored in flash memory Held in RAM in plaintext	The plaintext key is overwritten with zeroes upon termination of the session or reboot of the appliance
DH Keys used for TLS	DH (2048 bits)	Held in RAM in plaintext	The plaintext key is overwritten with zeroes upon termination of the session or reboot of the appliance
AES key used for TLS	AES-128 AES-256	Keys are not stored Held in RAM in plaintext	The key is overwritten with zeroes upon termination of the session or reboot of the appliance
Fortinet firmware update key	Generated by Fortinet RSA (2048 bits)	Stored in flash memory	The key may be overwritten when a firmware update is installed
Locally stored administrator passwords	User generated text string	Stored in flash memory	Overwritten by zeroes on factory reset

Table 13 – Key Material

The TOE includes two types of memory: RAM and flash. Ephemeral keys are only held in RAM. Private keys are only held in plaintext in RAM. These keys are zeroized when no longer required by the session, or when the device is rebooted by being overwritten with zeroes. The overwrite is read and verified. Private keys and public key certificates are stored in flash memory using OpenSSL 1.0.2h. Private and public keys are zeroized in RAM after use.

TOE Security Functional Requirements addressed: FCS_CKM.4

7.2.3 Cryptographic Operation

Cryptographic support is provided by cryptographic modules within the TOE devices. The applicable Cryptographic Algorithm Validation Program (CAVP) certificate numbers associated with the claimed functions are shown in Table 14.

Function / Algorithm	Details	CAVP Certificate
Encryption/decryption using AES in CBC mode	128, 256 bit key sizes	4461
RSA digital signature algorithm	2048 bit key size	2437
Component Validation List	TLS 1.0/1.1 (SHA-1, SHA-256)	1169
	KAS FFC	1330
SHA-1 and SHA-256 Cryptographic hashing services	160 bit message digest for SHA-1 256 bit message digest for SHA-256	3673
HMAC-SHA-1 Keyed hash message authentication	Key length: 160 bit Hash function: SHA-1 Block size: 512 bit Output MAC length: 160 bit	2960
HMAC-SHA-256 Keyed hash message authentication	Key length: 256 bit Hash function: SHA-256 Block size: 512 bit Output MAC length: 256 bit	2960
DRBG	SP 800-90A, AES-256 Counter Mode	1434

Table 14 – Cryptographic Functions

The TOE implements a NIST SP 800-90 DRBG with SHA-256 for key generation. The TOE implements a NIST SP 800-56B section 8.2 conformant RSA-based key establishment scheme for asymmetric key establishment. SHA-1 and SHA-256 are used for secure hashing. RSA and SHA-256 are used for digital signatures. SHA-1 is used with HMAC-SHA-1 Keyed hash message authentication and SHA-256 is used with HMAC-SHA-256 Keyed hash message authentication. SHA-256 is used with RSA for the verification of firmware.

TOE Security Functional Requirements addressed: FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4).

7.2.4 Random Bit Generation

The TOE implements an entropy collection system from the hardware based Fortinet Entropy Token. The noise source, which is derived from wide-band radio frequency (RF) white noise, is pooled and conditioned prior to being used. This noise source provides full entropy to the random number generation up to 256 bits.

The Fortinet Cryptographic Module contains a CTR_DRBG and is seeded with a hardware entropy source. Entropy from the noise source is extracted 5120 bits at a time, conditioned and used to seed the DRBG with 256 bits of full entropy. A failure of the entropy source is a blocking event for the cryptographic system and the entropy source is continually monitored for health; this helps to ensure that a catastrophic failure of the noise source will halt operation of the TOE.

TOE Security Functional Requirements addressed: FCS_RBG_EXT.1.

7.2.5 TLS Client Protocol and TLS Server Protocol

The TLS Client protocol is implemented using Open SSL 1.0.2h in support of the connection to the audit server and authentication server. The TLS Server protocol is implemented in support of the HTTPS connection to the administrative interface. In both cases, the following ciphersuites are supported:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256

The TOE supports TLS 1.1 and TLS 1.2. All other protocol requests will be denied. RSA with 2048 bit keys and Diffie-Hellman 2048 bit parameters are implemented in these ciphersuites.

When the TOE is acting as a server, peer authentication is implemented using username and password. When the TOE is acting as a client, peer authentication is implemented using X509 certificate-based mutual authentication. If the peer certificate is deemed invalid for TLS connections from the TOE, the connection will not be established and an error message will be generated.

The TLS client may use one of the following as a reference identifier (in the given order of preference): Host IP, Fully Qualified Domain Name or email address. Subject Alternative Name and the use of wildcards are supported. Certificate pinning is not supported.

TOE Security Functional Requirements addressed: FCS_HTTPS_EXT.1, FCS_TLSC_EXT.2, FCS_TLSS_EXT.1.

7.3 IDENTIFICATION AND AUTHENTICATION

7.3.1 Password Management

The TOE supports a variety of methods of Identification and Authentication to both local and external sources. Regardless of the method of administration that

is chosen by the administrator, no administrative action is possible prior to authentication.

The TOE uses a local password database for all of its locally-defined credentials. Passwords are created using mixed case characters, digits and the special characters “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“ and “)”.

TOE Security Functional Requirements addressed: FIA_PMG_EXT.1.

7.3.2 User Identification and Authentication

The following subsections describe the logon process for each of the supported logon methods.

7.3.2.1 Web/HTTPS Remote Logon

By default the web/HTTPS interface is enabled on the first network port. The TOE may also be configured to allow or disallow access to this interface on a per-network port basis, in either the CLI or the web User Interface (UI). The HTTPS web interface is accessed by pointing to the appliance IP address on port 443. Once the user has connected to the port and the HTTPS session is established, the TOE provides a warning banner which must be accepted prior to proceeding. Next, the user is presented with a username and login screen. During the credential entry, the user’s password appears only as dots in the password input field. When complete, the credentials are sent to the TOE over the TLS protected link. If the supplied username is not a configured administrator account, the login fails. If the supplied username is valid, the authentication mechanism configured for that account is retrieved. For local authentication, the local credential store is consulted; if there is a match, access is granted and the initial Web UI screen is displayed to the user. For LDAP authentication, a TLS-protected LDAP exchange takes place with a configured LDAP server to validate the supplied credentials. A failed login attempt will be met with an error message.

7.3.2.2 Local Console Logon

The local console is only accessible through the use of the dedicated Serial Console management port, if appropriately configured.

Local access is enabled and may not be disabled. The pre-login warning banner is displayed on initial access. The user is then prompted for the username, which is echoed back to the screen. Following identification, the user is prompted for his or her password. Asterisks appear in the password field as this password is entered. Once the credentials have been entered, they are validated. If the supplied username is not a configured administrator account, the login fails. If the supplied username is valid, the authentication mechanism configured for that account is retrieved. For local authentication, the local credential store is consulted; if there is a match, access is granted. For LDAP authentication, a TLS-protected LDAP exchange takes place with a configured LDAP server to validate the supplied credentials. Following successful authentication, the command prompt changes to the hostname followed by ‘#’ or ‘$’. If the supplied credentials are not valid, the user is prompted again for the password. After

three consecutive failed login attempts, a one minute delay is imposed by the TOE before the user may attempt to login again.

TOE Security Functional Requirements addressed: FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7.

7.3.3 Certificate Usage

FortiWeb uses the certificate path validation algorithm described in RFC 5280 to validate certificates. FortiWeb does not include pre-installed Certification Authority (CA) certificates, allowing the user to determine and limit the CAs to be trusted. Therefore, at least one CA certificate must be uploaded to the appliance before any certificates may be validated. Where client certificates are accepted, the client's CA certificate must also be uploaded to the appliance.

Although the FortiWeb appliance is delivered with a certificate that may be used to access the Web UI, in the evaluated configuration, the administrator must create a keypair and upload a certificate for use with HTTPS. Validity checks are performed on the certificate prior to use.

Validity checks are performed on certificates prior to use. To be considered valid, the following must be true:

- The certificate is not expired nor not yet valid
- The certificate has not been revoked, as indicated by an entry on the Certificate Revocation List (CRL)
- The certificate must be signed by a trusted CA
- The certificate must include an entry in the Issuer field whose value matches the entry in the Subject field of one of the trusted CA certificates

If a presented certificate does not meet these criteria, it is considered to be invalid and the FortiWeb appliance does not allow the connection. Additionally, an error is logged.

The Certificate Signing Request allows the user to enter a unique Common Name using the 'Certification Name' field. This may be used to enter the domain name. Optional information includes Organization, Organizational Unit and Country.

TOE Security Functional Requirements addressed: FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3.

7.4 SECURITY MANAGEMENT

Security management functionality is provided through the appliance CLI (accessed via the local console) and the web UI. Both interfaces provide full control over all security management functions. However, the user's functionality will be limited based on the user's role[6]. There is a default

[6] The CC term 'role' is used in this ST; however, in the FortiWeb guidance, the terms 'access profile' and 'administrative domains' are used to describe the allocation of permissions.

administrator account named ‘admin’, which cannot be deleted and has full permission to view and change all FortiWeb configuration options, and perform user management.

Manual update may only be performed by the ‘admin’ user, or a security administrator granted permission to perform this action.

No administrative functionality is accessible through the security management interfaces prior to login. Only administrative users provided with the appropriate permissions are able to manipulate TSF data for the purposes of performing security management functions.

Security management functionality includes the ability to configure the access banner, ability to configure the session inactivity timeout, verify firmware updates prior to installation and configuration of cryptographic functionality.

TOE Security Functional Requirements addressed:
FMT_MOF.1(1)/TrustedUpdate, FMT_MOF.1(2)/Audit, FMT_MOF.1(1)/TrustedUpdate, FMT_MOF.1(1)/AdminAct, FMT_MOF.1(2)/AdminAct, FMT_MTD.1, FMT_SMF.1, FMT_SMR.2.

7.5 PROTECTION OF THE TSF

7.5.1 Protection of Administrator Passwords and TSF Data

Table 13 provides a list of the keys maintained by the TOE, and how they are protected when stored. No interface for the viewing of keys is provided by the TOE.

Administrative passwords are stored in the configuration file on the flash drive of the TOE and are encoded via a hash function to ensure their confidentiality. These keys are capable of being zeroized either through a format of the flash memory or through a factory reset of the TOE.

Private keys and certificates used in support of TLS are maintained on the flash filesystem. Private keys are not viewable through the TOE interfaces. When these keys are no longer required, they may be removed by formatting of the flash memory.

The TOE stores a number of keys in volatile memory during normal operation of the cryptographic modules. This includes ephemeral keys and copies of persistent keys loaded into memory during normal operation. The TOE maintains these keys in its volatile memory in support of cryptographic operations for TLS and HTTPS. These keys are cleared when the process terminates. Each key is protected from unauthorized access via memory management, which disallows any memory reads from other processes within the operating system ensuring that the keys are only available to the calling application.

TOE Security Functional Requirements addressed: FPT_APW_EXT.1, FPT_SKP_EXT.1.

7.5.2 Timestamps

For the hardware devices, an internal clock source with battery backup in the hardware is used to initialize the time maintained by the operating system at boot. Subsequent hardware interrupts generated at fixed intervals are used to update the time accurately. The time maintained by the operating system is used to generate timestamps for audit records.

Time is used in support of the following security functions noted in Table 15.

Function		Use of Time
Audit	Generation of audit records	Date and time indication
Denial of Service Protection	HTTP Access Limit	Validation Time/Block Period
	HTTP Flood Prevention	Validation Time/Block Period
	TCP Flood Prevention	Block Period
Application Delivery	Web Cache	Cache Timeout
	Application Policy	Cache Timeout/Connection Timeout
Server Objects	Server Session Persistence	Timeout
	Authentication Policy	Connection Timeout/Cache Timeout

Table 15 – Use of Time Function

TOE Security Functional Requirements addressed: FPT_STM.1.

7.5.3 TSF Testing

At startup, the TOE undergoes the following tests:

- Firmware integrity test using RSA signatures
- Configuration integrity test using HMAC SHA-256
- Triple-DES, CBC mode, encrypt known answer test
- Triple-DES, CBC mode, decrypt known answer test
- AES, CBC mode, encrypt known answer test
- AES, CBC mode, decrypt known answer test
- HMAC SHA-1 known answer test
- SHA-1 known answer test (tested as part of HMAC SHA-1 known answer test)
- HMAC SHA-256 known answer test
- SHA-256 known answer test (tested as part of HMAC SHA-256 known answer test)
- RSA signature generation known answer test
- RSA signature verification known answer test
- DRBG known answer test

- Central Processing Unit (CPU) and Memory Basic Input/Output System (BIOS) self-tests – CPU and memory are initialized by exercising a set of known answer tests and the BIOS is compared against a known checksum of the image. The memory is zeroized and then a random pattern is written to and read from the memory.
- Boot loader image verification – the boot loader compares the image of the TOE to a known checksum of the image prior to booting.

These tests ensure the correct operation of the cryptographic functionality of the TOE, the CPU and BIOS and verify that the correct TOE image is being used. The cryptographic functionality will not be available if the tests fail, and any operation of the TOE supported by this functionality will not be available. If the CPU, or BIOS tests fail, the device will not complete the boot up operation. If the boot loader image verification fails, the boot up operation will fail. When the device completes the boot up operation, this is evidence that the self-tests have passed, and that the TOE, and the cryptographic functions are operating correctly.

The cryptographic module executes the following conditional tests when the related service is invoked:

- Continuous NDRNG test
- Continuous DRBG test
- RSA pairwise consistency test
- Configuration integrity test using HMAC SHA-256
- Firmware load test using RSA signatures

The following start-up self-tests can also be initiated on demand using the CLI command ‘execute fips kat all’ (to initiate all self-tests) or ‘execute fips kat <test>’ (to initiate a specific self-test):

- Firmware integrity test using RSA signatures
- Configuration integrity test using HMAC SHA-256
- Triple-DES, CBC mode, encrypt known answer test
- Triple-DES, CBC mode, decrypt known answer test
- AES, CBC mode, encrypt known answer test
- AES, CBC mode, decrypt known answer test
- HMAC SHA-1 known answer test
- SHA-1 known answer test (tested as part of HMAC SHA-1 known answer test)
- HMAC SHA-256 known answer test
- SHA-256 known answer test (tested as part of HMAC SHA-256 known answer test)
- RSA signature generation known answer test
- RSA signature verification known answer test
- DRBG known answer test

These tests ensure the correct operation of the cryptographic functionality of the TOE. If a self-test fails, the device enters error mode and halts system operation. All data output and cryptographic services are inhibited when in the

error state. Continued operation indicates that the tests have passed, and the TOE is operating correctly.

TOE Security Functional Requirements addressed: FPT_TST_EXT.1(1), FPT_TST_EXT.1(2).

7.5.4 Trusted Update

The current firmware version may be queried using 'get system status' in the CLI, or by viewing 'System > System > Status' using the web UI. The administrator may download a firmware update from https://support.fortinet.com using a network workstation. The firmware may then be updated using the web UI by selecting 'System > Status > Status' and selecting 'Update'. The process to upload the firmware image file includes verification of the digital signature using the Fortinet firmware update key, which is held in an encoded format in the current firmware image. If the digital signature on the firmware image cannot be verified, an error message appears. Once uploaded and verified, the administrator may choose to run the new image. This action installs the new firmware. The administrator may query the new version using 'get system status' in the CLI, or by viewing 'System > System > Status' using the web UI.

TOE Security Functional Requirements addressed: FPT_TUD_EXT.1.

7.6 TOE ACCESS

7.6.1 Session Termination

An authorized administrator may configure the TOE to terminate an inactive session following a specified period of time. The timeout value is set to five minutes by default, and applies to both local and remote CLI and web UI sessions. Administrative users may terminate their own sessions at any time.

TOE Security Functional Requirements addressed: FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4.

7.6.2 TOE Access Banners

The TOE displays an administrator configurable message to users prior to login on any of the three administrative interfaces: local console or web UI over HTTPS.

TOE Security Functional Requirements addressed: FTA_TAB.1.

7.7 TRUSTED PATH / CHANNELS

In the evaluated configuration, the TOE appliance sends audit records to a FortiAnalyzer device. This connection is initiated by the TOE and is protected using TLS. The TOE supports multiple authentication mechanisms, including the use of an LDAP authentication server. The connection to the authentication server is initiated by the TOE and is protected using TLS.

The TOE may be administered remotely using an HTTPS protected link to the Web UI. The HTTPS protected link uses TLS to provide cryptographic protection.

TOE Security Functional Requirements addressed: FTP_ITC.1, FTC_TRP.1.

8 ACRONYMS

The following acronyms are used in this ST:

Acronym	Definition
AES	Advanced Encryption Standard
BIOS	Basic Input/Output System
CA	Certification Authority
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CC	Common Criteria
CLI	Command Line Interface
cPP	collaborative Protection Profile
CPU	Central Processing Unit
CRL	Certificate Revocation List
DES	Data Encryption Standard
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DRNG	Digital Random Number Generator
DSS	Digital Signature Standard
FIPS	Federal Information Processing Standards
FFC	Finite Field Cryptography
HMAC	Hash Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IPSec	Internet Protocol Security
ISO/IEC	International Organization for Standardization and the International Electrotechnical Commission
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MAC	Message Authentication Code
ND	Network Device

Acronym	Definition
NIST	US National Institute of Standards and Technology
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OSP	Organizational Security Policy
PKCS	Public-Key Cryptography Standards
PP	Protection Profile
RAM	Random Access Memory
RBG	Random Bit Generator
RF	Radio Frequency
RFC	Request for Comment
RSA	Rivest, Shamir and Adleman
RSASSA-PSS	RSA Signature Scheme with Appendix - Probabilistic Signature Scheme
SaaS	Security-as-a-Service
SAR	Security Assurance Requirments
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SP	Special Publication
SQL	Structured Query Language
SSL	Secure Sockets Layer
ST	Security Target
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TRNG	True Random Number Generator
TSF	TOE Security Functionality
UI	User Interface
USB	Universal Serial Bus

Table 16 – Acronyms