SIEMENS

Security Target

RUGGEDCOM ROS v4.2.2.F

Reference Guide

08/2018
RC1407-EN-01

Disclaimer Of Liability

Siemens has verified the contents of this document against the hardware and/or software described. However, deviations between the product and the documentation may exist.

Siemens shall not be liable for any errors or omissions contained herein or for consequential damages in connection with the furnishing, performance, or use of this material.

The information given in this document is reviewed regularly and any necessary corrections will be included in subsequent editions. We appreciate any suggested improvements. We reserve the right to make technical improvements without notice.

Registered Trademarks

RUGGEDCOM™ and ROS™ are trademarks of Siemens Canada Ltd.

Other designations in this manual might be trademarks whose use by third parties for their own purposes would infringe the rights of the owner.

Third Party Copyrights

Siemens recognizes the following third party copyrights:

- Copyright © 2004 GoAhead Software, Inc. All Rights Reserved.

Open Source

RUGGEDCOM ROS contains Open Source Software. For license conditions, refer to the associated *License Conditions* document.

Security Information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens' products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. For more information about industrial security, visit https://www.siemens.com/industrialsecurity.

To stay informed about product updates as they occur, sign up for a product-specific newsletter. For more information, visit https://support.automation.siemens.com.

Contacting Siemens

Address
Siemens Canada Ltd
Industry Sector
300 Applewood Crescent
Concord, Ontario
Canada, L4K 5C7

Telephone
Toll-free: 1 888 264 0006
Tel: +1 905 856 5288
Fax: +1 905 856 1995

E-mail
ruggedcom.info.i-ia@siemens.com

Web
https://www.siemens.com/ruggedcom

Table of Contents

1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the organization of the ST. The TOE is the Siemens Canada Ltd (Siemens) RUGGEDCOM ROS (Rugged Operating System) v4.2.2.F and will hereafter be referred to as the TOE. The TOE is a proprietary operating system designed for the RUGGEDCOM M2100F, M2200F, M969F, RSG2100F, RSG2100PF, RSG2200F, RSG2300F, RSG2300PF, RSG2488F, RS400F, RS416F, RS416PF, RS900F, RS900GF, RS900GPF, and RS940GF switches developed and built by Siemens. These RUGGEDCOM switches are designed specifically to withstand harsh environmental conditions including temperature and humidity extremes, shock, vibration, and electromagnetic interference. The ruggedized switches, equipped with RUGGEDCOM ROS, provide Ethernet switching capabilities for customer networks in virtually any environment.

CONTENTS

Section 1.1

Purpose

This ST is divided into nine sections, as follows:

- Chapter 1, *Introduction* – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functionality and describes the physical and logical scope for the TOE as well as the ST and TOE references.
- Chapter 2, *Conformance Claims* – Provides the identification of any Common Criteria (CC), Protection Profile (PP), and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Chapter 3, *Security Problem* – Describes the threats, organizational security policies, and assumptions that pertain to the TOE and its environment
- Chapter 4, *Security Objectives for the Operational Environment* – Identifies the security objectives that are satisfied by the TOE and its environment
- Chapter 5, *Extended Components* – Identifies extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs) that are not included in CC Part 2 or CC Part 3
- Chapter 6, *Security Requirements* – Presents the SARs met by the TOE
  - Section 6.3, "Security Functional Requirements" – Presents the SFRs met by the TOE

- Chapter 7, *TOE Summary Specification* – Describes the security functions provided by the TOE that satisfy the security functional requirements and objectives
- Chapter 8, *Conformance Claims Rationale* - Presents the rationale for the SFR dependencies as to their consistency, completeness, and suitability
- Chapter 9, *Acronyms and Terms* – Defines the acronyms and terminology used within this ST

Section 1.2
Security Target and TOE References

Table 1 shows the ST and TOE references.

Table 1: ST and TOE References

ST Title	*Siemens Canada Ltd RUGGEDCOM ROS (Rugged Operating System) v4.2.2.F Security Target*
ST Version	Version 1.3
ST Author	Corsec Security, Inc.
ST Publication Date	August 13, 2018
TOE Reference	Siemens RUGGEDCOM ROS v4.2.2.F running on the M2100F, M2200F, M969F, RSG2100F, RSG2100PF, RSG2200F, RSG2300F, RSG2300PF, RSG2488F, RS400F, RS416F, RS416PF, RS900F, RS900GF, RS900GPF, and RS940GF RUGGEDCOM switches
CAVP Certificates	See Section 7.1.2, "Cryptographic Support"

Section 1.3
Product Overview

The Product Overview provides a high-level description of this evaluation's subject, Siemens RUGGEDCOM ROS running on the RUGGEDCOM switches. The following section, TOE Overview, provides the introduction to the parts of the overall product offering that are specifically being evaluated.

RUGGEDCOM ROS may be deployed on any of the RUGGEDCOM switches listed in Section 1.5.1, "Physical Scope". The RUGGEDCOM switches are highly configurable and can be customized with a number of different line module and power supply combinations. Customers choose a configuration that suits the targeted network and RUGGEDCOM assembles the RUGGEDCOM switches according to the specific configuration.

The RUGGEDCOM switches are therefore able to operate in the most adverse conditions and are primarily deployed in power distribution, refineries, or traffic control systems.

The RUGGEDCOM switches, equipped with the proprietary RUGGEDCOM ROS, provide switching capabilities for customer networks. Administrators provision and configure the switches using the management interfaces made available by RUGGEDCOM ROS described below in Section 1.4.1, "Management".

Section 1.4
TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

The TOE is a hardware and software TOE consisting of the RUGGEDCOM ROS running on the M2100F, M2200F, M969F, RSG2100F, RSG2100PF, RSG2200F, RSG2300F, RSG2300PF, RSG2488F, RS400F, RS416F, RS416PF, RS900F, RS900GF, RS900GPF, and RS940GF RUGGEDCOM switches. The purpose of the TOE is to provide Ethernet switching capabilities in a ruggedized enclosure for customer networks in virtually any environment.

CONTENTS

Section 1.4.1
Management

Administrative access to the TOE is provided locally over the serial port and remotely over the Ethernet port(s). Administrators access the serial port using a terminal emulator over a direct serial connection to the RUGGEDCOM switch. The serial port provides access to a terminal-based menu system that includes the Command Line Interface (CLI), which is used for normal administrative operations. Administrators can also access the terminal-based menu remotely using SSH over an Ethernet connection. Administrators access a Web Graphical User Interface (web interface) using HTTPS over an Ethernet connection for normal administrative operations. Using SFTP, Administrators may remotely read and write configuration files; write the SSL server certificate, SSH host keys, and SSH user public keys; update the Certificate Authority (CA) truststore; and may remotely read the audit log (read-only access).

Section 1.4.2
TOE Environment

The TOE needs the following environmental components in order to function properly:

- The cables and connectors that allow administrators and environmental components to communicate with the TOE
- An audit server with autossh script installed that will securely receive the audit logs from the TOE
- An OCSP server
- A management workstation with a standards-compliant client web browser to access the web interface over HTTPS and the terminal-based menu using SSH
- A firewall between the TOE and the external network, if connected to a public network

The TOE is intended to be deployed in a secure data center that protects physical access to the TOE. The TOE is intended to be interconnected by a back-end private network that does not connect directly to external hosts.

Section 1.5
TOE Description

This section primarily addresses the physical and logical components of the TOE that are included in the evaluation.

CONTENTS

Section 1.5.1
Physical Scope

Figure 1 illustrates the physical scope and the physical boundary of the overall solution and ties together all of the components of the TOE and the constituents of the TOE Environment. The TOE is composed of both the RUGGEDCOM switch hardware and the RUGGEDCOM ROS software.

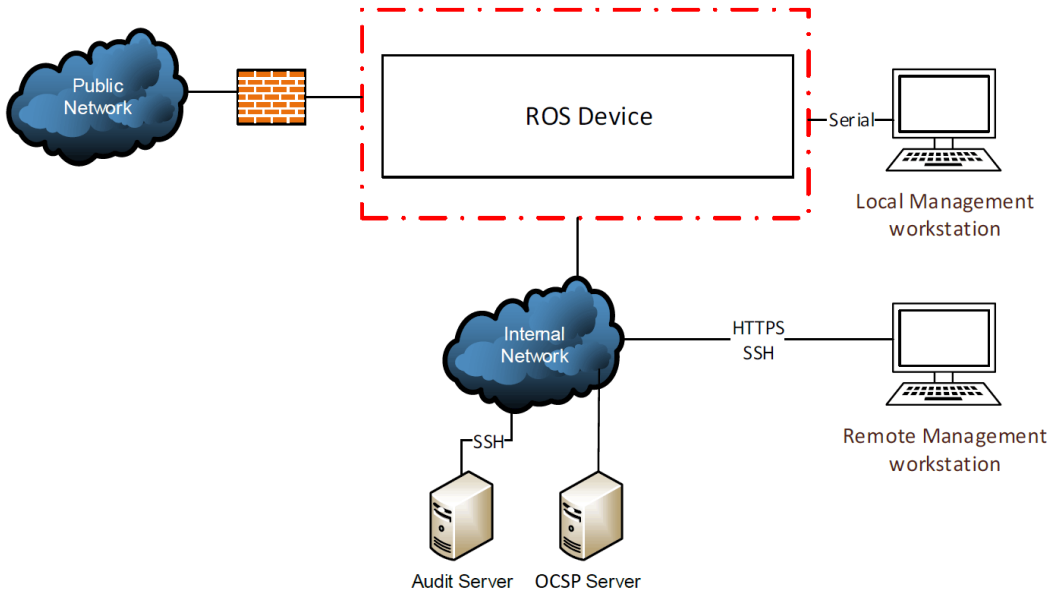

Figure 1: Physical TOE Boundary

CONTENTS

Section 1.5.1.1
TOE Hardware and Software

The TOE is a hardware and software TOE. For the evaluated configuration, the TOE software (RUGGEDCOM ROS v4.2.2.F) were installed and run in the following TOE hardware configurations:

Table 2: TOE Models Tested Configuration

Model	HW ID/CPU/Description	Line Card Configuration
M2100F	RSG2100v2 NXP ColdFire MCF5272 M2100FIPS-CC	2x 10FL – Multimode, 850nm, ST, 2km
		2x 100FX – Multimode, 1300nm, ST, 2km
		2x 100FX – Singlemode, 1310nm, ST, 20km
		2x 10/100TX – Micro-D
RSG2100F	RSG2100v2 NXP ColdFire MCF5272 RSG2100FIPS-CC	2x 10/100TX – RJ45
		2x 10FL – Multimode, 850nm, ST, 2km
		2x 100FX – Multimode, 1300nm, ST, 2km
		2x 100FX – Singlemode, 1310nm, LC, 90km
		2x 10/100TX – Micro-D
		2x 10/100/1000TX – RJ45
		1x 1000SX – Multimode, 850nm, LC, 500m
RSG2100PF	RSG2100v2 NXP ColdFire MCF5272 RSG2100PFIPS-CC	2x 10/100TX – RJ45
		2x 100FX – Multimode, 1300nm, ST, 2km
		2x 100FX – Multimode, 1300nm, MTRJ, 2km
		2x 100FX – Singleimode, 1310nm, SC, 90km
		2x 1000LX – Singleimode, 1310nm, LC, 25km
		1x 1000SX – Multimode, 850nm, LC, 500m
		2x 10/100TX – Micro-D
		2x 10/100TX – Micro-D
M2200F	RSG2200 NXP ColdFire MCF5272 M2200FIPS-CC	2x 10/100/1000TX – Micro-D
		2x 1000SX – Multimode, 850nm, LC, 500m
		2x 1000LX – Singlemode, 1310nm, LC, 25km
		2x 10/100/1000TX – Micro-D
		2x 10/100/1000TX – Micro-D
RSG2200F	RSG2200 NXP ColdFire MCF5272 RSG2200FIPS-CC	2x 10/100/1000TX – RJ45
		2x 1000SX – Multimode, 850nm, LC, 500m
		2x 1000LX – Singlemode, 1310nm, LC, 25 km
		2x 1000SX – Multimode, 850nm, LC, 500m
		1x 100FX – Multimode, 1300nm, MTRJ, 2km
RSG2300F	RSG2300	2x 10/100TX – RJ45

Model	HW ID/CPU/Description	Line Card Configuration
	NXP ColdFire MCF5272 RSG2300FIPS-CC	2x 100FX – Multimode, 1300nm, MTRJ, 2km
		2x 1000SX – Multimode, 850nm, LC, 500m
		2x 10/100/1000TX – RJ45
RSG2300PF	RSG2300 NXP ColdFire MCF5272 RSG2300PFIPS-CC	2x 10/100TX – RJ45
		2x 10/100TX – RJ45
		2x 1000SX – Multimode, 850nm, LC, 500m
		2x 100FX – Multimode, 1300nm, MTRJ, 2km
RSG2488F	RSG2488v2 NXP PowerQUICC MPC8308 RSG2488FIPS-CC	4x 10/100/1000TX – RJ45
		4x 10/100/1000TX – M12 X-Coded
		4x 1000SX – Multimode, 850nm, LC, 500m
		4x 1000SX – Multimode, 850nm, LC, 500m
		4x 1000LX – Singlemode, 1310nm, LC, 25km
		1x Precision Time Protocol (PTP) Module
		2x 10/100/1000TX – RJ45
		2x 10/100/1000TX – M12 X-Coded
RS416F	RSG416v2 NXP ColdFire MCF5272 RS416FIPS-CC	4x 10FL – Multimode, 850nm, ST, 2km
		4x RS232/RS422/RS485 & IRIG-B DB9
		4x RS232/RS422/RS485 & IRIG-B RJ45
		2x 10/100TX – RJ45
		1x IRIG-B in – BNC
		1x IRIG-B out – BNC (Slot 5 only)
RS416PF	RS416v2 NXP ColdFire MCF5272 RS416PFIPS-CC	4x 10FL – Multimode, 850nm, ST, 2km
		4x RS232/RS422/RS485 & IRIG-B DB9
		4x RS232/RS422/RS485 & IRIG-B RJ45
		1x IRIG-B in – BNC
		1x IRIG-B out – BNC (Slot 5 only)
		2x 10/100TX – RJ45
RS400F	RS400 (40-00-0010 Rev C1) NXP ColdFire MCF5272 RS400FIPS-CC	100FX – Multimode, 1310nm, LC, 2km
		1x 100FX – Singlemode, 1300nm, LC, 20km
		4x RS232/RS422/RS485 – DB9
RS940GF	RS940G (40-00-0097-000 Rev A) NXP ColdFire MCF5272 RS940GFIPS-CC	2x 1000SX – Multimode, 850nm, LC, 500m
RS900GF	RS900Gv2 NXP ColdFire MCF5272	2x 1000LX – Singlemode, 1310nm, LC, 25km

Model	HW ID/CPU/Description	Line Card Configuration
	RS900FIPS-CC	
RS900GPF	RS900GP NXP ColdFire MCF5272 RS900GFIPS-CC	2x 1000LX – Singlemode, 1310nm, SC, 25km
RS900F	RS900v3, Fiber NXP ColdFire MCF5272 RS900FIPS-CC	2x 100FX – Multimode, 1300nm, ST, 2km
		1x 100FX – Singlemode, 1310nm, ST, 20km
		1x 100FX – Multimode, 1300nm, SC, 2km
M969F	RS969 (v2, 40-00-0090) NXP ColdFire MCF5272 M969FIPS-CC	2x 1000SX – Multimode, 850nm, LC, 500m

Table 2 lists the RUGGEDCOM switches evaluated configuration. The line models on each switch model are configurable and the tested configurations provide a sample of possible configurations. The line modules provide 10/100/1000BaseTX Ethernet, serial, and fiber interfaces that are used to send and receive user data. The line modules provide nothing more than the physical interface.

Section 1.5.1.2
Guidance Documentation

Table 3 lists the TOE Guidance Documentation to install, configure, and maintain the TOE.

Table 3: Guidance Documentation

Document Name	Description
Siemens RUGGEDCOM M2100F Installation Guide, 05/2018, RC1342-EN-1 *Siemens RUGGEDCOM M2200F Installation Guide*, 05/2018, RC1343-EN-1 *Siemens RUGGEDCOM M969F Installation Guide*, 05/2018, RC1349-EN-1 *Siemens RUGGEDCOM RSG2100F Installation Guide*, 05/2018, RC1344-EN-1 *Siemens RUGGEDCOM RSG2200F Installation Guide*, 05/2018, RC1346-EN-1 *Siemens RUGGEDCOM RSG2100PF Installation Guide*, 05/2018, RC1345-EN-1 *Siemens RUGGEDCOM RSG2300F Installation Guide*, 05/2018, RC1347-EN-1 *Siemens RUGGEDCOM RSG2300PF Installation Guide*, 05/2018, RC1348-EN-1 *Siemens RUGGEDCOM RSG2488F Installation Guide*, 05/2018, RC1351-EN-1 *Siemens RUGGEDCOM RS400F Installation Guide*, 05/2018, RC1341-EN-1 *Siemens RUGGEDCOM RS416F Installation Guide*, 05/2018, RC1354-EN-1 *Siemens RUGGEDCOM RS416PF Installation Guide*, 05/2018, RC1355-EN-1 *Siemens RUGGEDCOM RS900GF Installation Guide*, 05/2018, RC1352-EN-1 *Siemens RUGGEDCOM RS900GPF Installation Guide*, 05/2018, RC1353-EN-1 *Siemens RUGGEDCOM RS900F Installation Guide*, 05/2018, RC1350-EN-1 *Siemens RUGGEDCOM RS940GF Installation Guide*, 05/2018, RC1356-EN-1	Includes steps for the basic initialization and setup of the TOE.
Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RSG2100F, RSG2100PF, M2100F, 05/2018, RC1234-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RSG2200F, M2200F*, 05/2018, RC1235-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS969F, M969F*, 05/2018, RC1233-EN-1	Contains detailed steps for how to properly configure and maintain the TOE.

Document Name	Description
Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RSG2300F, RSG2300PF, 05/2018, RC1236-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RSG2488F*, 05/2018, RC1237-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS400F*, 05/2018, RC1300-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS416F, RS416PF*, 05/2018, RC1228-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS900GF*, 05/2018, RC1229-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS900GPF*, 05/2018, RC1230-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS900F*, 05/2018, RC1231-EN-1 *Siemens RUGGEDCOM ROS v4.2.2.F User Guide for RS940GF*, 05/2018, RC1232-EN-1	
Siemens Canada Ltd RUGGEDCOM ROS (Rugged Operating System) v4.2.2.F Guidance Documentation Supplement v0.6	Contains information regarding specific configuration for the TOE evaluated configuration.
Siemens RUGGEDCOM ROS Devices Security Policy, v0.14, August 16, 2017	Contains details on FIPS mode configurations and taper evidence validation.
FAQ - "How to Implement Secure, Unattended Logging in ROS"	Contains steps to setup and maintain secure audit logging on the TOE.

Section 1.5.2
Logical Scope

The logical boundary of the TOE will be broken down into the following security classes which are further described in Section 6.3, “Security Functional Requirements” and Chapter 7, *TOE Summary Specification* of this ST. The logical scope also provides the description of the security features of the TOE. The security functional requirements implemented by the TOE are usefully grouped under the following Security Function Classes:

CONTENTS

Section 1.5.2.1
Security Audit

The TOE generates audit records for security-relevant actions of the authorized administrators accessing the TOE via the terminal-based menu and web interface. The TOE records the identity of the administrator responsible for the log event, where applicable. To remotely and securely backup the audit logs, an Administrator can configure the syslog server to call into the TOE and request audit log files be printed to the syslog server over an SSH connection. The connection to the audit server is secured using SSH. When logs are filled, the TOE overwrites in

two possible ways: the oldest log record can be overwritten with the new log record or the oldest log file can be overwritten with the new log file.

Section 1.5.2.2
Cryptographic Support

The Cryptographic Support TOE Security Function (TSF) provides cryptographic functions to establish web interface sessions (secured with HTTPS) and terminal sessions (secured with SSH) between an administrator's management workstation and the TOE. Cryptographic functions are also provided when the external audit server requests an SSH session with the TOE to protect the transfer of audit messages. Lastly, cryptographic functions are used to establish SFTP sessions to import keys and certificates. The cryptographic operations necessary to support this TSF are provided by Siemens's proprietary cryptographic module (for list of CAVP Certificates see Section 7.1.2, "Cryptographic Support"). In addition, the TOE provides a variety of cryptographic algorithms for its own use. Keys are zeroized by overwriting with zeros. The TOE is limited by hardware on SSH rekeys. See Section 7.1.2, "Cryptographic Support" for more details.

Section 1.5.2.3
Identification and Authentication

The TOE provides functionality that requires administrators to verify their claimed identity. The Identification and Authentication TSF ensures that only authorized administrators can gain access to the configuration settings and management settings of the TOE. Administrators can only view the access banner prior to authenticating with a valid user name and password. The TOE requires administrators to use strong passwords. The TOE provides no feedback to Administrators when they are entering their passwords at the login prompt of the terminal-based menu for both direct serial remote SSH connections. The TOE does present feedback in the form of bullets ('•') over the web interface and in the form of an 'x' over the terminal interface. Administrators using password-based authentication are locked out after a configurable number of unsuccessful authentication attempts and must wait a configurable period of time before they are unlocked. The TOE can present a certificate to authenticate to external entities and this certificate and the trust anchor certificate are stored within a truststore. Certificate revocation status is verified on certificates uploaded to the TOE using an external OCSP server.

Section 1.5.2.4
Security Management

The TOE provides a web interface and a terminal-based menu for administrators to manage the security functions, configuration, and other features of the TOE. The security management function specifies user roles with defined access for the management of the TOE components. Updating the TOE, modifying the configuration file, configuring the access banner, setting the inactivity timeout, configuring authentication failure parameters, configuring time and re-enabling the Administrator account are all functions restricted to the Security Administrator.

Section 1.5.2.5
Protection of the TSF

The TOE invokes a set of self-tests each time the TOE is powered on to ensure that the TSF operates correctly. The TOE also provides a reliable timestamp for its own use. An Administrator can manually set the time for the TOE. A digital signature using an RSA public key is used to verify all software updates that are applied to the TOE. The

TOE prevents an administrator from reading the keys stored in the TOE. Passwords are stored in obfuscated form to prevent them from being read in plaintext.

Section 1.5.2.6
TOE Access

The TOE terminates local and remote management sessions after an administrator-configurable time period of inactivity. The TOE also provides administrator's the capability to manually terminate the session prior to the inactivity timeout. After an administrator's session is terminated, the administrator must log in again to regain access to TOE functionality. A login banner is displayed at the login screen of the web interface and prior to authentication over the terminal-based menu.

Section 1.5.2.7
Trusted Path/Channels

The cryptographic functionality of the TOE provides the TOE the ability to create trusted paths and trusted channels. The TOE implements a trusted channel using SSH between itself and a remote server in order to protect the audit logs as they are being sent. Additionally, the TOE provides trusted paths between administrators and the web interface via HTTPS and the terminal-based menu via SSH. The management communication channels between the TOE and a remote entity are distinct from network data communication channels and provide mutual identification and authentication. In addition, the communications are protected from modification and disclosure.

Section 1.5.3
Excluded Functionality

The following services were not part of the evaluated configuration and were not tested:

- Virtual Local Area Network configuration
- Port configuration
- Broadcast Storm filtering
- Quality of Service based on part, tag, MAC16, or IP type of service
- Multiple Spanning Tree Protocol
- Rapid Spanning Tree Protocol
- Enhanced Rapid Spanning Tree Protocol

The following services are present in the TOE but are excluded in this evaluation:

- RADIUS
- TACACS+
- RSH
- Telnet
- TFTP
- ModBus management
- Remote Syslog

- Management connections over SNMP15 v1, v2, and v3
- Management via HTTP
- Network Time Protocol (NTP) time synchronization and service
- RUGGEDCOM Discovery Protocol (RCDP)[1]
- IP forwarding

Section 1.5.4
Scope of Evaluation

The evaluation is limited in scope to the secure management features described in collaborative *Protection Profile for Network Devices v2.0 + Errata 20180314, 14-March-2018* (ND cPP) and detailed in Section 1.5.2, "Logical Scope".

[1] RCDP has been disabled and removed from all command and debugging interfaces.

2 Conformance Claims

This section provides the identification for any CC, Protection Profile (PP), Technical Decisions (TD), and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in Chapter 8, *Conformance Claims Rationale*.

Table 4: CC and PP Conformance

Common Criteria (CC) Identification and Conformance	*Common Criteria for Information Technology Security Evaluation*, Version 3.1, Revision 4, September 2012; CC Part 2 extended; CC Part 3 conformant; PP claim to the *Collaborative Protection Profile for Network Devices v2.0 + Errata 20180314*, March 14, 2018 conformant; and no interpretations apply to the claims made in this ST.
PP Identification	Exact Conformance[a] to the collaborative Protection Profile for Network Devices, v2.0 + Errata 20180314.
TD Conformance	Conformance to TD0228, TD0259, TD0260, TD0262, TD0281, TD0290, TD0291, TD0321, and TD0324 are claimed.

[a] *Exact Conformance is a type of Strict Conformance such that the set of SFRs and the SPD/Objectives are exactly as presented within the accepted NDPP without changes.*

3 Security Problem

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed.

CONTENTS

Section 3.1

Threats

The threats for the Network Device are grouped according to functional area of the device in the sections below.

CONTENTS

Section 3.1.1

Communications with the Network Device

A network device communicates with other network devices and other network entities. The endpoints of this communication can be geographically and logically distant and may pass through a variety of other systems. The intermediate systems may be untrusted providing an opportunity for unauthorized communication with the network device or for authorized communication to be compromised. The security functionality of the network device must be able to protect any critical network traffic (administration traffic, authentication traffic, audit traffic, etc.). The communication with the network device falls into two categories: authorized communication and unauthorized communication.

Authorized communication includes network traffic allowable by policy destined to and originating from the network device as it was designed and intended. This includes critical network traffic, such as network device administration and communication with an authentication or audit logging server, which requires a secure channel to protect the communication. The security functionality of the network device includes the capability to ensure that only authorized communications are allowed and the capability to provide a secure channel for critical

network traffic. Any other communication with the network device is considered unauthorized communication. (Network traffic traversing the network device but not ultimately destined for the device, e.g. packets that are being routed, are not considered to be “communications with the network device” – cf. Section 3.2.3, “No Thru Traffic Protection”.)

The primary threats to network device communications addressed in this collaborative Protection Profile (cPP) focus on an external, unauthorized entity attempting to access, modify, or otherwise disclose the critical network traffic. A poor choice of cryptographic algorithms or the use of non-standardized tunneling protocols along with weak administrator credentials, such as an easily guessable password or use of a default password, will allow a threat agent unauthorized access to the device. Weak or no cryptography provides little to no protection of the traffic allowing a threat agent to read, manipulate and/or control the critical data with little effort. Non-standardized tunneling protocols not only limit the interoperability of the device but also lack the assurance and confidence standardization provides through peer review.

CONTENTS

Section 3.1.1.1
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.

SFR Rationale:

- The Administrator role is defined in FMT_SMR.2 and the relevant administration capabilities are defined in FMT_SMF.1 and FMT_MTD.1/CoreData
- The actions allowed before authentication of an Administrator are constrained by FIA_UIA_EXT.1, and include the advisory notice and consent warning message displayed according to FTA_TAB.1
- The requirement for the Administrator authentication process is described in FIA_UAU_EXT.2
- Locking of Administrator sessions is ensured by FTA_SSL_EXT.1 (for local sessions), FTA_SSL.3 (for remote sessions), and FTA_SSL.4 (for all interactive sessions)
- The secure channels used for remote Administrator connections is specified in FTP_TRP.1/Admin

Section 3.1.1.2
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.

SFR Rationale:

- Requirements for key generation and key distribution are set in FCS_CKM.1 and FCS_CKM.2 respectively

  Requirements for use of cryptographic schemes are set in FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash

  Requirements for random bit generation to support key generation and secure protocols (see SFRs resulting from T.UNTRUSTED_COMMUNICATION_CHANNELS) are set in FCS_RBG_EXT.1

  Management of cryptographic functions is specified in FMT_SMF.1

Section 3.1.1.3
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.

SFR Rationale:

- The general use of secure protocols for identified communication channels is described at the top level in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the requirements for inter-component communications are addressed by the requirements in FPT_ITT.1
- Requirements for the use of secure communication protocols are set for all the allowed protocols in FCS_HTTPS_EXT.1, FCS_SSHS_EXT.1, FCS_TLSS_EXT.1
- Optional and selection-based requirements for use of public key certificates to support secure protocols are defined in FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

Section 3.1.1.4
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.

SFR Rationale:

- The use of appropriate secure protocols to provide authentication of endpoints (as in the SFRs addressing T.UNTRUSTED_COMMUNICATION_CHANNELS) are ensured by the requirements in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the authentication requirements for endpoints in inter-component communications are addressed by the requirements in FPT_ITT.1.

Section 3.1.2
Valid Updates

Updating network device software and firmware is necessary to ensure that the security functionality of the network device is maintained. The source and content of an update to be applied must be validated by cryptographic means; otherwise, an invalid source can write its own firmware or software updates that circumvents the security functionality of the network device. Methods of validating the source and content of a software or firmware update by cryptographic means typically involve cryptographic signature schemes where hashes of the updates are digitally signed.

Unpatched versions of software or firmware leave the network device susceptible to threat agents attempting to circumvent the security functionality using known vulnerabilities. Non-validated updates or updates validated using non-secure or weak cryptography leave the updated software or firmware vulnerable to threat agents attempting to modify the software or firmware to their advantage.

CONTENTS

Section 3.1.2.1
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.

SFR Rationale:

- Requirements for protection of updates are set in FPT_TUD_EXT.1

  Requirements for management of updates are defined in FMT_SMF.1 and (for manual updates) in FMT_MOF.1/ManualUpdate

Section 3.1.3
Audited Activity

Auditing of network device activities is a valuable tool for administrators to monitor the status of the device. It provides the means for administrator accountability, security functionality activity reporting, reconstruction of events, and problem analysis. Processing performed in response to device activities may give indications of a failure or compromise of the security functionality. When indications of activity that impact the security functionality are not generated and monitored, it is possible for such activities to occur without administrator awareness. Further, if records are not generated and retained, reconstruction of the network and the ability to understand the extent of any compromise could be negatively affected. Additional concerns are the protection of the audit data that is recorded from alteration or unauthorized deletion. This could occur within the TOE or while the audit data is in transit to an external storage device.

Note this cPP requires that the network device generate the audit data and have the capability to send the audit data to a trusted network entity (e.g., a syslog server).

CONTENTS

Section 3.1.3.1
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.

SFR Rationale:

- Requirements for basic auditing capabilities are specified in FAU_GEN.1 and FAU_GEN.2, with timestamps provided according to FPT_STM_EXT.1
- Requirements for protecting audit records stored on the TOE are specified in FAU_STG.1
- Requirements for secure transmission of local audit records to an external IT entity via a secure channel are specified in FAU_STG_EXT.1

Section 3.1.4
Administrator and Device Credentials and Data

A network device contains data and credentials which must be securely stored and must appropriately restrict access to authorized entities. Examples include the device firmware, software, configuration authentication credentials for secure channels, and administrator credentials. Device and administrator keys, key material, and authentication credentials need to be protected from unauthorized disclosure and modification. Furthermore, the security functionality of the device needs to require default authentication credentials, such as administrator passwords, be changed.

Lack of secure storage and improper handling of credentials and data, such as unencrypted credentials inside configuration files or access to secure channel session keys, can allow an attacker to not only gain access to the network device but also compromise the security of the network through seemingly authorized modifications to configuration or through man-in-the-middle attacks. These attacks allow an unauthorized entity to gain access and perform administrative functions using the Security Administrator's credentials and to intercept all traffic as an authorized endpoint. This results in difficulty in detection of security compromise and in reconstruction of the network, potentially allowing continued unauthorized access to administrator and device data.

CONTENTS

Section 3.1.4.1
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.

SFR Rationale:

- Protection of secret/private keys against compromise is specified in FPT_SKP_EXT.1
- Secure destruction of keys is specified in FCS_CKM.4

- If (optionally) management of keys is provided by the TOE then this is specified in FMT_SMF.1, and confining this functionality to Security Administrators is required by FMT_MTD.1/CryptoKeys

Section 3.1.4.2
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.

SFR Rationale:

- Requirements for password lengths and available characters are set in FIA_PMG_EXT.1
- Protection of password entry by providing only obscured feedback is specified in FIA_UAU.7
- Actions on reaching a threshold number of consecutive password failures are specified in FIA_AFL.1
- Requirements for secure storage of passwords are set in FPT_APW_EXT.1.

Section 3.1.5
Device Failure

Security mechanisms of the network device generally build up from roots of trust to more complex sets of mechanisms. Failures could result in a compromise to the security functionality of the device. A network device self-testing its security critical components at both start-up and during run-time ensures the reliability of the device's security functionality.

CONTENTS

Section 3.1.5.1
Security Functionality Failure

An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

SFR Rationale:

- Requirements for running self-test(s) are defined in FPT_TST_EXT.1

Section 3.2
Assumptions

This section describes the assumptions made in identification of the threats and security requirements for network devices. The network device is not expected to provide assurance in any of these areas, and as a result, requirements are not included to mitigate the threats associated.

CONTENTS

Section 3.2.1
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.

[OE.PHYSICAL]

Section 3.2.2
Limited Functionality

The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example the device should not provide a computing platform for general purpose applications (unrelated to networking functionality).

[OE.NO_GENERAL_PURPOSE]

Section 3.2.3
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).

[OE.NO_THRU_TRAFFIC_PROTECTION]

Section 3.2.4
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.

[OE.TRUSTED_ADMIN]

Section 3.2.5
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.

[OE.UPDATES]

Section 3.2.6
Admin Credentials Secure

The administrator’s credentials (private key) used to access the network device are protected by the platform on which they reside.

[OE.ADMIN_CREDENTIALS_SECURE]

Section 3.2.7
Residual Information

The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

[OE.RESIDUAL_INFORMATION]

Section 3.3
Organizational Security Policies

An organizational security policy (OSP) is a set of rules, practices, and procedures imposed by an organization to address its security needs. For the purposes of this cPP, a single policy is described in the section below.

CONTENTS

Section 3.3.1
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.

[FTA_TAB.1]

4 Security Objectives for the Operational Environment

The following subsections describe objectives for the Operational Environment.

CONTENTS

Section 4.1
Physical

Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

Section 4.2
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.

Section 4.3
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.

Section 4.4
Trusted Admin

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

Section 4.5
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.

Section 4.6
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.

Section 4.7
Residual Information

The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

5 Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.2, "Conventions". This section contains the definitions for the extended requirements that are used in the cPP, including those used in Chapter 6, *Security Requirements* and Section 6.3, "Security Functional Requirements".

CONTENTS

Section 5.1

Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE. The extended SFRs are organized by class. Table 5 identifies all extended SFRs implemented by the TOE.

Table 5: Extended TOE Security Functional Requirements

Name	Description
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_RBG_EXT.1	Random Bit Generation
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_SSHS_EXT.1	SSH Server
FCS_TLSS_EXT.1	TLS Server Protocol
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_X509_EXT.1/REV	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Request
FPT_SKP_EXT.1	Protection of TSF data (for reading of all symmetric keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_STM_EXT.1	Reliable time stamps
FPT_TST_EXT.1	TSF Testing

Name	Description
FPT_TUD_EXT.1	Trusted Update
FTA_SSL_EXT.1	TSF-initiated session locking

CONTENTS

Section 5.1.1

Class FAU: Security Audit

Families in this class address the requirements for functions to implement security audit as defined in Chapter 2, *Conformance Claims*.

CONTENTS

Section 5.1.1.1

FAU_STG_EXT: Protected Audit Event Storage

Family Behavior

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 Family Decomposition

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

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 connection to the external IT entity

Audit: FAU_STG_EXT.1

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

a. No audit necessary

FAU_STG_EXT.1 Protected Audit Event Storage

Hierarchical to:	**No other components**
Dependencies:	**AU_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

Section 5.1.2

Class FCS: Cryptographic Support

Families in this class address the requirements for functions to implement cryptographic functionality as defined in Chapter 2, *Conformance Claims*.

CONTENTS

Section 5.1.2.1

FCS_RBG_EXT: Random Bit Generation

Family Behavior

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

Component Leveling

Figure 3: Random Bit Generation Family Decomposition

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 cPP/ST:

a. There are no auditable events foreseen

FCS_RBG_EXT.1 Random Bit Generation

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***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 RBG23 shall be seeded by at least one entropy source that accumulates entropy from [selection: [assignment: *number of software-based sources*] software-based noise source, [assignment: *number of hardware-based sources*] hardware-based noise source] with a minimum of [selection: 128 bits, 192 bits, 256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011, of the keys and CSP that it will generate

Section 5.1.2.2

FCS_HTTPS_EXT.1 HTTPS Protocol

Family Behavior

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 4: HTTPS Protocol Family

FCS_HTTPS_EXT.1 HTTPS requires that HTTPS be implemented according to RFC24 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 cPP/ST:

a. There are no auditable events foreseen

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

Section 5.1.2.3

FCS_SSHS_EXT.1 SSH Server Protocol

Family Behavior

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

Component Leveling

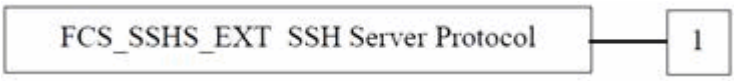

Figure 5: SSH Server Protocol Family Decomposition

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

Management: FCS_SSHS_EXT.1

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

a. There are no management activities foreseen

Audit: FCS_SSHS_EXT.1

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

a. Failure of SSH session establishment

b. SSH session establishment

c. SSH session termination

FCS_SSHS_EXT.1 SSH Server Protocol

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_SSHS_EXT.1.1***
  The TSF shall implement the SSH protocol that complies with [RFC(s) [selection: 4251, 4252, 4253, 4254, *5647, 5656, 6187, 6668*]
- ***FCS_SSHS_EXT.1.2***
  The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based
- ***FCS_SSHS_EXT.1.3***
  The TSF shall ensure that, as described in RFC 4253, packets greater than [assignment: *number of bytes*] bytes in an SSH transport connection are dropped
- ***FCS_SSHS_EXT.1.4***
  The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [assignment: *encryption algorithms*]
- ***FCS_SSHS_EXT.1.5***
  The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: ssh-rsa, ecdsa-sha2-nistp256] and [selection: ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, no other public key algorithms] as its public key algorithm(s) and rejects all other public key algorithms

- ***FCS_SSHS_EXT.1.6***
  The TSF shall ensure that the SSH transport implementation uses [assignment: *: list of MAC algorithms*] as its MAC algorithm(s) and rejects all other MAC algorithm(s)
- ***FCS_SSHS_EXT.1.7***
  The TSF shall ensure that [assignment: *list of key exchange methods*] are the only allowed key exchange methods used for the SSH protocol
- ***FCS_SSHS_EXT.1.8***
  The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed

Section 5.1.2.4

FCS_TLSS_EXT TLS Server Protocol

Family Behavior

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

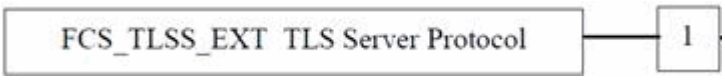

Figure 6: TLS Server Protocol Family Decomposition

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

Management: FCS_SSHS_EXT.1

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

a. There are no management activities foreseen

Audit: FCS_TLSS_EXT.1

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

a. Failure of TLS session establishment

b. TLS session establishment

c. TLS session termination

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)] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:
  - [assignment: *List of optional ciphersuites and reference to RFC in which each is defined]*
- ***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]]

Section 5.1.3
Class FIA: Identification and Authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity as defined in Chapter 2, *Conformance Claims*.

CONTENTS

Section 5.1.3.1
FIA_PMG_EXT: Password Management

Family Behavior

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

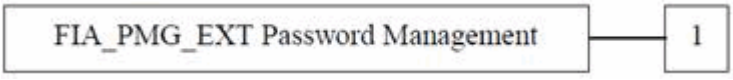

Figure 7: Password Management Family Decomposition

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

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

a. The ability to configure the password length

Audit: FIA_PMG_EXT.1

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

a. There are no auditable events foreseen

FIA_PMG_EXT.1 Password Management

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***FIA_PMG_EXT.1.1***
  The TSF shall provide the following password management capabilities for administrative passwords:
  a. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [selection: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [assignment: *other characters*]]
  b. Minimum password length shall be configurable to [assignment: *minimum number of characters supported by the TOE*] and [assignment: *number of characters greater than or equal to 15*]

Section 5.1.3.2

FIA_UIA_EXT: User Identification and Authentication

Family Behavior

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

Component Leveling

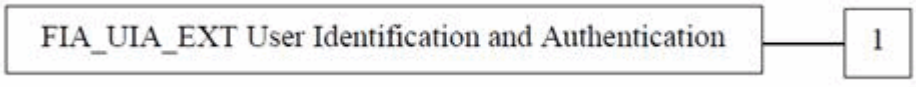

Figure 8: User Identification and Authentication Family Decomposition

FFIA_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 cPP/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

Section 5.1.3.3

FIA_UAU_EXT: Password-based Authentication Mechanism

Family Behavior

Provides for a locally based administrative user authentication mechanism.

Component Leveling

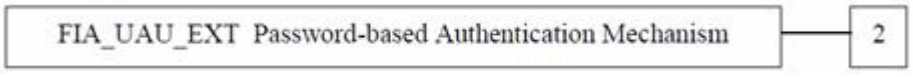

Figure 9: Password-based Authentication Mechanism Family Decomposition

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 cPP/ST:

a. Minimal: All use of the authentication mechanism

FIA_UAU_EXT.2 Password-based Authentication Mechanism

Hierarchical to:	**No other components**
Dependencies:	**No other components**

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

Section 5.1.3.4

FIA_X509_EXT: Authentication Using X.509 Certificates

Family Behavior

This family defines the behavior, 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: User Identification and Authentication Family Decomposition

FIA_X509_EXT.1/REV 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.

IA_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/REV, 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 certificate

b. Approve import and removal of X.509v3 certificates

c. Initiate certificate requests

Audit: FIA_X509_EXT.1/REV, 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 cPP/ST:

a. Minimal: No specific audit requirements are specified

FIA_X509_EXT.1/REV X.509 Certificate Validation

Hierarchical to:	**No other components**
Dependencies:	**No other components**

- ***FIA_X509_EXT.1.1/Rev***
  The TSF shall validate certificates in accordance with the following rules:
  - RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates
  - The certificate path must terminate with a trusted CA certificate
  - The TSF shall validate a 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 6069, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5, no revocation method]
  - The TSF shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field
- ***FIA_X509_EXT.1.2/Rev***
  The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE

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, DTLS], 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

Section 5.1.4

Class FPT: Protection of the TSF

Families in this class address the requirements for functions providing integrity and management of mechanisms that constitute the TSF and of the TSF data as defined in Chapter 2, *Conformance Claims*.

CONTENTS

Section 5.1.4.1
FPT_SKP_EXT: Protection of TSF Data

Family Behavior

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

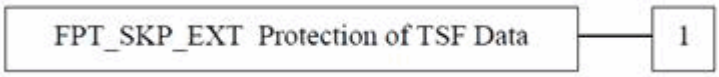

Figure 11: Protection of TSF Data (for Reading of Symmetric Keys) Family Decomposition

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 cPP/ST:

a. There are no auditable events foreseen

FFPT_SKP_EXT.1 Protection of TSF Data (for Reading of All Pre-shared Symmetric and Private Keys)

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

Section 5.1.4.2
FPT_APW_EXT: Protection of Administrator Passwords

Family Behavior

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

Component Leveling

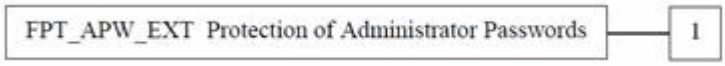

Figure 12: Protection of Administrator Passwords

FPT_APW_EXT.1 Protection of administrator passwords, requires the TOE not to store passwords in plaintext. It was modeled after FPT_SSP.1.

Management: FPT_APW_EXT.1

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

a. There are no management activities foreseen

Audit: FPT_APW_EXT.1

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

a. There are no auditable events foreseen

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

Section 5.1.4.3

FPT_TST_EXT: TSF Testing

Family Behavior

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

Component Leveling

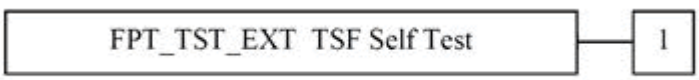

Figure 13: TSF Testing Family Decomposition

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.

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 cPP/ST:

a. Indication that TSF self test was completed

FPT_TST_EXT.1 TSF Testing

Hierarchical to:	**No other components**
Dependencies:	**No other components**

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

Section 5.1.4.4

FPT_TUD_EXT: Trusted Update

Family Behavior

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

Component Leveling

Figure 14: Trusted Update Family Decomposition

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.

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 cPP/ST:

a. Initiation of the update process

b. Any failure to verify the integrity of the update

FPT_TUD_EXT.1 Trusted Update

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

- ***FPT_TUD_EXT.1.1***
  The TSF shall provide [assignment: *Administrators*] the ability to query the currently executing version of the TOE firmware/software and [selection: the most recently installed version of the TOE firmware/software; no other TOE firmware/software version].
- ***FPT_TUD_EXT.1.2***
  The TSF shall provide [assignment: *Administrators*] the ability to manually initiate updates to TOE firmware/software and [selection: support automatic checking for updates, support automatic updates, no other update mechanism]
- ***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

Section 5.1.4.5

FPT_STM_EXT: Reliable Time Stamps

Family Behavior

Components in this family address the requirements for maintaining a reliable time stamp.

Component Leveling

Figure 15: Reliable Time Stamp Family Decomposition

FPT_STM_EXT.1 Reliable Time Stamp requires that the TSF provide a reliable time stamp and allows for Administrator to set time and configure synchronization with an external time source.

Management: FPT_STM_EXT.1

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

a. Ability to set the TOE time

b. Administrator setting of the time

Audit: FPT_STM_EXT.1

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

a. Discontinuous changes to time – either administrator actuated or changed via automated process

FPT_STM_EXT.1 Reliable Time Stamp

Hierarchical to:	**No other components**
Dependencies:	**No other components**

- ***FPT_STM_EXT.1.1***
  The TSF shall be able to provide reliable time stamps for its own use
- ***FPT_STM_EXT.1.2***
  The TSF shall [selection: allow the Security Administrator to set the time, synchronize time with external sources]

Section 5.1.5

Class FTA: TOE Access

Families in this class specify functional requirements for controlling the establishment of a user's session as defined in Chapter 2, *Conformance Claims*.

CONTENTS

Section 5.1.5.1
FTA_SSL_EXT: TSF-initiated Session Locking

Family Behavior

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

The extended FTA_SSL_EXT family is based on the FTA_SSL family.

Component Leveling

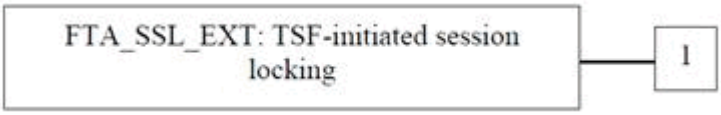

Figure 16: TSF-initiated Session Locking Family Decomposition

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 cPP/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

Section 5.2

Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6 Security Requirements

This cPP identifies the SFRs and SARs to be met by the TOE. These requirements are described in the sections below.

CONTENTS

Section 6.1

Security Assurance Requirements

The SARs identified in the cPP frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

This section lists the set of SARs from CC Part 3 that are required in evaluations against this cPP. Individual Evaluation Activities to be performed are specified in the Supporting Document (SD), Mandatory Technical Document, Evaluation Activities for Network Device cPP.

The TOE security assurance requirements are identified in Table 6.

Table 6: Security Assurance Requirements

Assurance Requirements	
Security Target (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Stated security requirements (ASE_REQ.1)
	Security Problem Definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labeling of the TOE (ALC_CMC.1)
	TOE Configuration Management coverage (ALC_CMS.1)
Tests (ATE)	Independent testing – sample (ATE_IND.1)

Assurance Requirements	
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

Section 6.2

Conventions

The conventions used in descriptions of the SFRs are as follows:

- Assignment: Indicated with *italicized text*
- Refinement made by PP author: Indicated with **bold text** and ~~strikethroughs~~, if necessary
- Selection: Indicated with underlined text
- Assignment within a Selection: Indicated with *italicized and underlined text*
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/"
- Extended SFRs are identified by having a label "EXT" at the end of the SFR name
- Operations such as assignments and selections performed by the PP author are identified as shown above; however, do not appear within brackets. This is done intentionally to delineate between selections/assignments made by the PP author and those made by the ST author. No refinements have been made by the ST author other than grammatical and/or formatting corrections, or in places where a table reference differs from that of the PP.

Section 6.3

Security Functional Requirements

The individual security functional requirements are specified in the sections below. SFRs in this section are mandatory SFRs that any conformant TOE must meet. Based on selections made in these SFRs, it will also be necessary to include some of the selection-based SFRs in Appendix B. Additional optional SFRs may also be adopted from those listed in Appendix A.

The Evaluation Activities defined in [SD] describe actions that the evaluator will take in order to determine compliance of a particular TOE with the SFRs. The content of these Evaluation Activities will therefore provide more insight into deliverables required from TOE Developers.

This section specifies the SFRs for the TOE. This section organizes the SFRs by CC class. Table 7 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement.

Table 7: TOE Security Functional Requirements

Name	Description	S[a]	A[b]	R[c]	I[d]
FAU_GEN.1	Audit data generation	✓	✓		
FAU_GEN.2	User identity association				
FAU_STG_EXT.1	Protected Audit Event Storage	✓	✓		
FCS_CKM.1	Cryptographic Key Generation	✓		✓	
FCS_CKM.2	Cryptographic Key Establishment	✓		✓	

Name	Description	S[a]	A[b]	R[c]	I[d]
FCS_CKM.4	Cryptographic Key Destruction	✓			
FCS_COP.1/ DataEncryption	Cryptographic Operation (AES Data Encryption/ Decryption)	✓			✓
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)	✓	✓	✓	✓
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)	✓		✓	✓
FCS_COP.1/Keyed Hash	Cryptographic Operation (Keyed Hash Algorithm)	✓	✓	✓	✓
FCS_HTTPS_EXT.1	HTTPS Protocol	✓			
FCS_SSHS_EXT.1	SSH Server	✓	✓		
FCS_TLSS_EXT.1	TLS Server Protocol	✓			
FCS_RBG_EXT.1	Random Bit Generation	✓	✓		
FIA_AFL.1	Authentication Failure Management	✓	✓		
FIA_PMG_EXT.1	Password Management	✓	✓		
FIA_UIA_EXT.1	User Identification and Authentication	✓	✓		
FIA_UAU_EXT.2	Password-based Authentication Mechanism	✓	✓		
FIA_UAU.7	Protected Authentication Feedback		✓		
FIA_X509_EXT.1/REV	X.509 Certificate Validation	✓			✓
FIA_X509_EXT.2	X.509 Certificate Authentication	✓	✓		
FIA_X509_EXT.3	X.509 Certificate Requests	✓			
FMT_MOF.1/ ManualUpdate	Management of security functions behavior				✓
FMT_MTD.1/CoreData	Management of TSF data				✓
FMT_SMF.1	Specification of management functions	✓			
FMT_SMR.2	Restrictions on Security Roles				
FPT_APW_EXT.1	Protection of Administrator Passwords				
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)				
FPT_STM_EXT.1	Reliable Time Stamps				
FPT_TST_EXT.1	TSF testing	✓	✓		
FPT_TUD_EXT.1	Trusted Update	✓	✓		
FTA_SSL_EXT.1	TSF-initiated session locking	✓			
FTA_SSL.3	TSF-initiated Termination			✓	
FTA_SSL.4	User-initiated Termination			✓	
FTA_TAB.1	Default TOE access banners			✓	
FTP_ITC.1	Inter-TSF Trust Channel	✓	✓	✓	

Name	Description	S[a]	A[b]	R[c]	I[d]
FTP_TRP.1/Admin	Trusted Path	✓		✓	✓

[a] *S=Selection*

[b] *A=Assignment*

[c] *R=Refinement*

[d] *I=Iteration*

CONTENTS

Section 6.3.1

Class FAU: Security Audit

FAU_GEN.1 Audit Data Generation

Hierarchical to:	**No other components**
Dependencies:	**FPT_STM.1 Reliable Time Stamps**

- ***FAU_GEN.1.1***
  The TSF shall be able to generate an audit record of the following auditable events:
  a. Start-up and shut-down of the audit functions;
  b. All auditable events, for the not specified level of audit; and
  c. *All administrative actions comprising:*
     - *Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).*
     - *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
     - *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
     - *Resetting passwords (name of related user account shall be logged).*
     - *[Starting and stopping services];*
  d. *Specifically defined auditable events listed in Table 8.*

Table 8: FAU_GEN.1 Audit Data Generation

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None
FCS_COP.1/DataEncryption	None	None
FCS_COP.1/SigGen	None	None
FCS_COP.1/Hash	None	None
FCS_COP.1/KeyedHash	None	None
FCS_HTTPS_EXT.1	Failure to establish an HTTPS Session	Reason for failure
FCS_SSHS_EXT.1	Failure to establish an SSH Session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_RBG_EXT.1	None	None
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded	Origin of the attempt (e.g., IP address)
FIA_PMG_EXT.1	None	None
FIA_UIA_EXT.1	All use of the identification and authentication mechanism	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/REV	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None
FMT_MTD.1/CoreData	All management activities of TSF data	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None
FPT_APW_EXT.1	None	None
FPT_SKP_EXT.1	None	None
FPT_STM_EXT.1	Discontinuous changes to time – either administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1.)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).

Requirement	Auditable Events	Additional Audit Record Contents
FPT_TST_EXT.1	None	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None
FTA_SSL_EXT.1 (if 'terminate the session' is selected)	The termination of a local session by the session locking mechanism	None
FTA_SSL.3	The termination of a remote session by the session locking mechanism	None
FTA_SSL.4	The termination of an interactive session	None
FTA_TAB.1	None	None
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 attempts
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None

- ***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 8*.

FAU_GEN.2 User Identity Association

Hierarchical to:	**No other components**
Dependencies:	**FAU_GEN.1 Audit Data Generation** **FIA_UID.1 Timing of Identification**

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

FAU_STG_EXT.1 Protected Audit Event Storage

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.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: [

a. *for TOEs with a ColdFire processor, the oldest sector is overwritten with the record*

b. *for TOEs with a PowerPC processor the log file is erased and new logs are written into the erased file*]]

when the local storage space for audit data is full.

Section 6.3.2

Class FCS: Cryptographic Support

FCS_CKM.1 Cryptographic Key Generation

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.2** **FCS_CKM.4**

- ***FCS_CKM.1.1***
  The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [
  - *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
  - *ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;*
  - *FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3*

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

FCS_CKM.2 Cryptographic Key Establishment

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1** **FCS_CKM.4** **FTP_ITC.1**

- ***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 meets the following: NIST Special Publication 800-56B Revision 1, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography";*
  - *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";*
  - *Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3;*

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

FFCS_CKM.4 Cryptographic Key Destruction

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1 Cryptographic Key Generation**

- ***FCS_CKM.4.1***
  The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method
  - *For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes]]*,
  - *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by the TSF that [*
    - *Logically addresses the storage location of the key and performs a [single] overwrite consisting of [zeroes]];*

  ]

  that meets the following: *No Standard*

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

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1 Cryptographic Key Generation** **FCS_CKM.4 Cryptographic Key Destruction**

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

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

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1 Cryptographic Key Generation** **FCS_CKM.4 Cryptographic Key Destruction**

- ***FCS_COP.1.1/SigGen***
  The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm [
  - *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits to 3072 bits]*,

  ] ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~

  that meet the following: [
  - *For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1 5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3*

  ]

FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1 Cryptographic Key Generation** **FCS_CKM.4 Cryptographic Key Destruction**

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

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

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1 Cryptographic Key Generation** **FCS_CKM.4 Cryptographic Key Destruction**

- ***FCS_COP.1.1/Keyed Hash***
  The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*] and cryptographic key sizes [*160 to 320 bits, 256 to 512 bits, 384 to 768 bits, or 512 to 1024*] **and message digest sizes [*160, 256, 384, 512*] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*

FCS_HTTPS_EXT.1 HTTPS Protocol

Hierarchical to:	**No other components**
Dependencies:	**FCS_TLS_EXT.1**

- ***FCS_HTTPS_EXT.1.1***
  The TSF shall implement the HTTPS protocol that complies with RFC 2818
- ***FCS_HTTPS_EXT.1.2***
  The TSF shall implement HTTPS using TLS
- ***FCS_HTTPS_EXT.1.3***
  If a peer certificate is presented, the TSF shall [not require client authentication, [no other action]] if the peer certificate is deemed invalid

FCS_SSHS_EXT.1 SSH Server Protocol

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1** **FCS_CKM.2** **FCS_COP.1/DataEncryption** **FCS_COP.1/SigGen** **FCS_COP.1/Hash** **FCS_COP.1/KeyedHash** **FCS_RBG_EXT.1**

- ***FCS_SSHS_EXT.1.1***
  The TSF shall implement the SSH protocol that complies with RFC(s) [4251, 4252, 4253, 4254, 5647]
- ***FCS_SSHS_EXT.1.2***
  The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based
- ***FCS_SSHS_EXT.1.3***
  The TSF shall ensure that, as described in RFC 4253, packets greater than [*4096*] bytes in an SSH transport connection are dropped
- ***FCS_SSHS_EXT.1.4***
  The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [aes128-cbc, aes256-cbc, aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM]
- ***FCS_SSHS_EXT.1.5***
  The TSF shall ensure that the SSH public-key based authentication implementation uses [ssh-rsa] and [no other public key algorithm] as its public key algorithm(s) and rejects all other public key algorithms
- ***FCS_SSHS_EXT.1.6***
  The TSF shall ensure that the SSH transport implementation uses [hmac-sha1] and [AEAD_AES_128_GCM, AEAD_AES_256_GCM] as its MAC algorithm(s) and rejects all other MAC algorithm(s)
- ***FCS_SSHS_EXT.1.7***
  The TSF shall ensure that [diffie-hellman-group14-sha1, ecdh-sha2-nistp256] and [ecdh-sha2-nistp521, and ecdh-sha2-nistp384] are the only allowed key exchange methods used for the SSH protocol
- ***FCS_SSHS_EXT.1.8***
  The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

FCS_TLSS_EXT.1 TLS Server Protocol

Hierarchical to:	**No other components**
Dependencies:	**FCS_CKM.1** **FCS_CKM.2** **FCS_COP.1/DataEncryption** **FCS_COP.1/SigGen** **FCS_COP.1/Hash** **FCS_COP.1/KeyedHash** **FCS_RBG_EXT.1 Random Bit Generation**

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

  [
  - TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
  - TLS_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_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_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 [none]
- ***FCS_TLSS_EXT.1.3***
  The TSF shall [perform RSA key establishment with key size [2048 bits, 3072 bits]; generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves; generate Diffie-Hellman parameters of size [2048 bits]]

FCS_RBG_EXT.1 Random Bit Generation

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***FCS_RBG_EXT.1.1***
  The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [CTR_DRBG (AES)]
- ***FCS_RBG_EXT.1.2***
  The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[*one*] software-based noise source] with minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and hashes that it will generate

Section 6.3.3

Class FIA: Identification and Authentication

FIA_AFL.1 Authentication Failure Management

Hierarchical to:	**No other components**
Dependencies:	**FIA_UIA_EXT.1**

- ***FIA_AFL.1.1***
  The TSF shall detect when *an Administrator configurable positive integer within* [*1 to 20*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely*
- ***FIA_AFL.1.2***
  When the defined number of unsuccessful authentication attempts has been met, the TSF shall *[prevent the offending remote Administrator from successfully authenticating until [the appliance’s rebooting action] is taken by a local Administrator; prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed]*.

FIA_PMG_EXT.1 Password Management

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***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 configurable to [*1*] and [*17*]

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;
  - [*[ICMP echo response;*
  - *HTTP redirects;*
  - *TCP handshakes]*]
- ***FIA_UIA_EXT.1.2***
  The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user

FIA_UAU_EXT.2 Password-based Authentication Mechanism

Hierarchical to:	**No other components**
Dependencies:	**No other components**

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

FIA_UAU.7 Protected Authentication Feedback

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

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

FIA_X509_EXT.1/Rev X.509 Certificate Validation

Hierarchical to:	**No other components**
Dependencies:	**No other components**

- ***FIA_X509_EXT.1.1/Rev***
  The TSF shall validate certificates in accordance with the following rules:
  - RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**
  - The certificate path must terminate with a trusted CA certificate
  - The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates
  - The TSF shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 6960]
  - The TSF shall validate the extendedKeyUsage field according to the following rules:
    - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field*
    - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field*
    - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field*
    - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field*
- ***FIA_X509_EXT.1.2/Rev***
  The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE

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 [TLS, HTTPS], and [no additional uses]
- ***FIA_X509_EXT.2.2***
  When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [allow the administrator to choose whether to accept the certificate in these cases]

FIA_X509_EXT.3 X.509 Certificate Requests

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 [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

Section 6.3.4
Class FMT: Security Management

FMT_MOF.1/ManualUpdate Management of Security Functions Behaviour

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

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

FMT_MTD.1/CoreData Management of TSF Data

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

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

FMT_SMF.1 Specification of Management Functions

Hierarchical to:	**No other components**
Dependencies:	**FIA_UIA_EXT.1 User Identification and Authentication** **FCS_COP.1/SigGen Cryptographic Operation (for Cryptographic Signature)** **FPT_TUD_EXT.1 Trusted Update**

- ***FMT_SMF.1.1***
  The TSF shall be capable of performing the following management functions:
  - *Ability to administer the TOE locally and remotely;*
  - *Ability to configure the access banner;*
  - *Ability to configure the session inactivity time before session termination or locking;*
  - *Ability to update the TOE, and to verify the updates using [*digital signature*] capability prior to installing those updates;*
  - *Ability to configure the authentication failure parameters for FIA_AFL.1;*
  - *[*

- *Ability to set the time that is used for time-stamps;*
- *Ability to re-enable an Administrator account;*
- *No other capabilities]*

FMT_SMR.2 Restrictions on Security Roles

Hierarchical to:	**No other components**
Dependencies:	**FIA_UID.1 Timing of Identification**

- ***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

Section 6.3.5

Class FPT: Protection of the TSF

FPT_APW_EXT.1 Protection of Administrator Passwords

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

FPT_SKP_EXT.1 Protection of TSF Data (for Reading of All Pre-shared, Symmetric, and Private Keys)

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

FPT_STM_EXT.1 Reliable Time Stamps

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***FPT_STM_EXT.1.1***
  The TSF shall be able to provide reliable time stamps for its own use
- ***FPT_STM_EXT.1.2***
  The TSF shall [allow the Security Administrator to set the time]

FPT_TST_EXT.1 TSF Testing

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***FPT_TST_EXT.1.1***
  The TSF shall run a suite of the following self-tests [during initial start-up (on power on), at the request of the authorised user] to demonstrate the correct operation of the TSF: [
  - *Firmware Integrity Test*
  - *AES encrypt/decrypt CBC Known Answer Test (KAT)*
  - *AES encrypt/decrypt CTR KAT*
  - *AES encrypt/decrypt GCM KAT*
  - *HMAC KAT with SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512*
  - SHA-1 KAT
  - SHA-224, SHA-256, SHA-384, and SHA-512 KATs
  - *CTR DRBG KAT*
  - *RSA signature generation/verification KAT*
  - *ECDSA pair-wise consistency test*
  - *Primitive "Z" Computation KAT*
  - *Continuous Random Number Generator Test for DRBG*
  - *Continuous Random Number Generator Test for NDRNG*
  - *RSA pair-wise consistency test*
  - *ECDSA signature generation/verification test*
  - *ECC full public-key validation test*
  - *EC Diffie-Hellman Public Key Assurance Test*]

FPT_TUD_EXT.1 Trusted Update

Hierarchical to:	**No other components**
Dependencies:	**FCS_COP.1(1) Cryptographic Operation (for Cryptographic Signature), or FCS_COP.1(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 [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

Section 6.3.6

Class FTA: TOE Access

FTA_SSL_EXT.1 TSF-initiated Session Locking

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

- ***FTA_SSL_EXT.1.1***
  The TSF shall, for local interactive sessions, [
  - terminate the session]

  after a Security Administrator-specified time period of inactivity

FTA_SSL.3 TSF-initiated Termination

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

FTA_SSL.4 User-initiated Termination

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

FTA_TAB.1 Default TOE Access Banners

Hierarchical to:	**No other components**
Dependencies:	**No dependencies**

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

Section 6.3.7
Class FTP: Trusted Path/Channels

FTP_ITC.1 Inter-TSF Trusted Channel

Hierarchical to:	**No other components**
Dependencies:	**No other components**

- ***FTP_ITC.1.1***
  The TSF shall **be capable of using [SSH] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [no other capabilities]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**
- ***FTP_ITC.1.2***
  The TSF shall permit **the TSF, or the authorized IT entities** to initiate communication via the trusted channel
- ***FTP_ITC.1.3***
  The TSF shall initiate communication via the trusted channel for [*no services*]

FTP_TRP.1/Admin Trusted Path

Hierarchical to:	**No other components**
Dependencies:	**No other components**

- ***FTP_TRP.1.1/Admin***
  The TSF shall **be capable of using [*SSH, 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/Admin***
  The TSF shall permit remote **Administrators** to initiate communication via the trusted path
- ***FTP_TRP.1.3/Admin***
  The TSF shall require the use of the trusted path for *initial administrator authentication and all remote administration actions*

7 TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

CONTENTS

Section 7.1

TOE Security Functionality

Each of the security requirements and the associated descriptions correspond to the security functions. Hence, each function is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions satisfy the necessary requirements.

Table 9: Mapping of TOE Security Functionality to Security Functional Requirements

TOE Security Function	SFR ID	Description
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected audit event storage
Cryptographic Support	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.2	Cryptographic key establishment
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1/DataEncryption	Cryptographic operation (AES data encryption/decryption)
	FCS_COP.1/SigGen	Cryptographic operation (signature generation and verification)
	FCS_COP.1/Hash	Cryptographic operation (hash algorithm)
	FCS_COP.1/Keyed Hash	Cryptographic operation (keyed hash algorithm)
	FCS_HTTPS_EXT.1	HTTPS protocol
	FCS_SSHS_EXT.1	SSH server protocol
	FCS_TLSS_EXT.1	TLS server protocol
	FCS_RBG_EXT.1	Random bit generation
Identification and Authentication	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password management
	FIA_UIA_EXT.1	User identification and authentication

TOE Security Function	SFR ID	Description
	FIA_UAU_EXT.2	Password-based authentication mechanism
	FIA_UAU.7	Protected authentication feedback
	FIA_X509_EXT.1/REV	X.509 Certificate validation
	FIA_X509_EXT.2	X.509 Certificate authentication
	FIA_X509_EXT.3	X.509 Certificate requests
	FMT_MOF.1/ManualUpdate	Management of security functions behavior
	FMT_MTD.1/CoreData	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.2	Restrictions on security roles
Protection of the TSF	FPT_APW_EXT.1	Protection of administrator passwords
	FPT_SKP_EXT.1	Protection of TSF data (for reading of all symmetric keys)
	FPT_STM_EXT.1	Reliable time stamps
	FPT_TST_EXT.1	TSF testing
	FPT_TUD_EXT.1	Trusted update
TOE Access	FTA_SSL.3	TSF-initiated termination
	FTA_SSL.4	User-initiated termination
	FTA_SSL_EXT.1	TSF-initiated session locking
	FTA_TAB.1	Default TOE access banners
Trusted path/channels	FTP_ITC.1	Inter-TSF trust channel
	FTP_TRP.1/Admin	Trusted path

CONTENTS

Section 7.1.1
Security Audit

FAU_GEN.1, FAU_GEN.2

The TOE generates audit records as part of the security audit function. As administrators manage and configure the TOE, their activities are tracked and recorded as audit records and are stored in the file system. The TOE generates audit records for all the required events specified in Section 6.3.1, "Class FAU: Security Audit".

An example audit record is shown below:

- 15/09/15 18:29:54.481 INFO 32C Console user 'admin' logged out

This example shows for the event:

- Date and time: 15/09/15 18:29:54.481
- Level of event: possible levels include DEBUG, INFO, NOTICE, WARN, ERROR, CRITICAL, ALERT
- Interface where event occurred: possible values are Console, SSH, or HTTPS
- Username of user who performed the action: 'admin'
- Description of the action: logged out

For the password authentication mechanism, the User Name is the User ID. Actions performed by the user (admin role) authenticated using the password are identified by the "User Name." For the SSH key-based authentication mechanism, the keys are identified by their fingerprint and the key fingerprint will be the User ID. Actions performed by users authenticated using key-based authentication are identified by the key fingerprints. The user name in that case is the client's user name associated with particular ssh public key and it is used by the SSH service of the TOE to select the respective public key.

Audits of time vary according to the type of time change that is made. When an administrator changes the time, the old and new time are recorded in the audit records. When a time zone change is made, the time is also updated. In this case the old and new time are also recorded. The TOE can be set to automatically account for Daylight Savings Time.

FAU_STG_EXT.1

The TOE securely transfers audit logs via SSH to an external audit server. The TOE maintains 1.5 to 5.5 MB[1] of local audit storage. The TOE can include a ColdFire or PowerPC processor. The action that is taken when the audit storage is full depends on the type of processor. The CPU platforms differ, in that the ColdFire platform's filesystem is directly mapped flash sectors, while the PowerPC platform uses a transparent filesystem (tfs). The flash sectors implement a simple round-robin mechanism for the logs by writing over the first sector after all sectors are filled. On the ColdFire processor the syslog.txt file is allocated the remainder of flash memory after all other files have been allocated. The size can be verified by using the flashfiles command. When the log becomes full on a TOE with a ColdFire processor, the TOE will erase the first sector of the file and begin recording audit records in this sector. On the PowerPC processor the log is file based and the syslog.txt file is 1.75M in size. When the log becomes full on a TOE with a PowerPC processor, the TOE will erase the log file and begin a new log file. Please see Section 1.5.1.1, "TOE Hardware and Software" for a complete list of which model includes which CPU.

A Security Administrator can configure logs to be sent to the external server as they occur so that no log events are lost. All audit logs are protected using operating system file permissions and only authorized Security Administrators can delete these files.

TOE Security Functional Requirements Satisfied:

- FAU_GEN.1

[1] MB - Megabytes

- FAU_GEN.2
- FAU_STG_EXT.1

Section 7.1.2

Cryptographic Support

Details the cryptographic algorithm certificates for the TOE.

Table 10: CAVP-Approved Algorithm Implementations

CAVP Certificate		Algorithm	Standard	Mode/ Method	Key Lengths/ Curves/ Moduli	Use
PowerPC	ColdFire					
4030 [http://csrc.nist.gov/groups/STM/cavp/documents/aes/aesval.html#4030]	4037 [http://csrc.nist.gov/groups/STM/cavp/documents/aes/aesval.html#4037]	AES[a]	FIPS 197 NIST SP 800-38D	CBC, GCM	128, 256	Encryption/ Decryption (FCS_COP.1/ DataEncryption)
			FIPS 197	CTR	256	
2078 [http://csrc.nist.gov/groups/STM/cavp/documents/dss/rsanewval.html#2078]	2072 [http://csrc.nist.gov/groups/STM/cavp/documents/dss/rsanewval.html#2072]	RSA	FIPS 186-4	-	2048, 3072	Key Generation (FCS_CKM.1, bullet 1)
				SHA-256, SHA-384, and SHA-512 (PKCS #1 v1.5)[b]	2048, 3072	Signature Generation and Verification (FCS_COP.1/SigGen)
899 [http://csrc.nist.gov/groups/STM/cavp/documents/dss/ecdsanewval.html#899]	903 [http://csrc.nist.gov/groups/STM/cavp/documents/dss/ecdsanewval.html#903]	ECDSA	FIPS 186-4	-	P-256, P-384, P-521	Public Key Generation and Verification (FCS_CKM.1, bullet 2)
3336 [http://csrc.nist.gov/groups/STM/cavp/documents/shs/shaval.html#3336]	3329 [http://csrc.nist.gov/groups/STM/cavp/documents/shs/shaval.html#3329]	SHS	FIPS 180-4	SHA-1, SHA-256, SHA-384, SHA-512	-	Message Digest (FCS_COP.1/Hash)
2631 [http://csrc.nist.gov/groups/STM/cavp/documents/mac/hmacval.html#2631]	2635 [http://csrc.nist.gov/groups/STM/cavp/documents/mac/hmacval.html#2635]	HMAC	FIPS 198-1	SHA-1, SHA-256, SHA-384, and SHA-512	-	Message Authentication (FCS_COP.1/Keyed Hash)
1204 [http://csrc.nist.gov/groups/STM/cavp/documents/drbg/drbgnewval.html#1204]	1207 [http://csrc.nist.gov/groups/STM/cavp/documents/drbg/drbgnewval.html#1207]	DRBG	NIST SP 800-90A	CTR_based	-	Deterministic Random Bit Generation (FCS_RBG_EXT.1)
858 [http://csrc.nist.gov/groups/STM/cavp/documents/components/componentnewval.html#858]	863 [http://csrc.nist.gov/groups/STM/cavp/documents/components/componentnewval.html#863]	CVL for ECC KAS	NIST SP 800-56A	-	P-256, P-384, P-521	ECC KAS (FCS_CKM.2, bullet 2)

[a] *CAVP testing was performed on AES ECB mode but is not implemented in the module.*

[b] *SHA-1 shall not be used for digital signature generation with the exception as specified in SP 800-52 REV1 and SP 800-57 Part 3 REV1.*

FCS_CKM.1

The TOE is capable of generating asymmetric RSA key pairs. RSA key pairs can be generated for 2048 and 3072-bit sizes and are used for SSH and TLS. The RSA key pairs meet the SP 800-56B section 6.3.1. ECDSA key pairs are generated as a part of the ECDH algorithm using the P-256, P-384, and P-521 curves. The TOE implements DH group 14 as a part of SSH.

FCS_CKM.2

The TOE uses RSA-based key establishment for TLS. In TLS communications the TOE acts as the server and therefore receives the pre-master secret and verifies it using the server's generated RSA key. Both ECDH and DH are used to establish keys in SSH communications. The TOE acts as the server in SSH communications as well and therefore uses ECDH and DH to compute the shared secret. The DH key agreement used for SSH uses 2048-bit MODP group as defined in RFC 3526 Section 3.

FCS_CKM.4

Volatile memory is used for temporary storage of passwords during the authentication process, and session or ephemeral keys are used during the secure session. Volatile memory is destroyed using a single, direct overwrite of zeros.

SSH host keys, SSH public keys, SSL host certificates, and public certificates are stored in non-volatile flash memory. When key destruction is needed, a block erase is performed followed by a read of the memory to confirm the key has been erased. If memory is not all zeros, the block erase is attempted up to five times.

FCS_COP.1/DataEncryption

As a part of the TOE's secure protocols, AES is used for encryption and decryption. The modes CBC or GCM can be used in either SSH or TLS, and key sizes are either 128 bits or 256 bits. AES CTR with 256 bits can also be used for encryption and decryption in SSH. In AES GCM the IV is generated internally using the CTR DRBG.

FCS_COP.1/SigGen

Cryptographic signatures are used by the TOE as a part of SSH and TLS sessions as well as to verify TOE updates. RSA signatures with a key size of 2048 to 3072 bits can be generated and verified for SSH or TLS. Updates to the TOE are signed using RSA.

FCS_COP.1/Hash

In order to perform digital signature operations for TLS, and update verification, the TOE performs cryptographic hashing using Secure Hash Algorithm SHA-256, SHA-384, and SHA-512. In addition, the TOE performs cryptographic hashing using SHA-1 and SHA-256 for key exchange operations in SSH. The TOE also uses a SHA-512 hash to hash the password and a salt value prior to storing or checking the password.

FCS_COP.1/Keyed Hash

Keyed-hash message authentication is used in both TLS and SSH within the TOE. HMAC with SHA-1, SHA-256, SHA-384, and SHA-512 is provided by the TOE for this use. The HMAC SHA-1 algorithm provides key sizes of 160 to 320bits and message digests that are 160 bits. The HMAC SHA-256 algorithm provides key sizes of 256 to 512 bits and message digests that are 256 bits. The HMAC SHA-384 algorithm provides key sizes of 384 to 768 bits and message digests that are 384 bits. Lastly, the HMAC SHA-512 algorithm provides key sizes of 512 to 2014 bits and message digests that are 512 bits.

FCS_HTTPS_EXT.1

HTTPS using TLS is used for management sessions to the TOE's web interface. Certificate-based authentication can be used for these sessions. The TOE does not perform mutual authentication and will not verify the client certificate prior to connection. The HTTPS implementation in the TOE complies with all MUST statements related to the server side implementation of HTTPS in RFC 2818.

FCS_SSHS_EXT.1

The TOE uses SSH for management communications to the terminal-based menu and to send log messages to an external audit server. RSA is the only algorithm used for authentication within SSH. The host-based public key can use either 2048 or 3072 bit keys. Password-based authentication is also permitted via SSH. The TOE will drop packets received over SSH that are larger than 4096 bytes. SSH traffic is encrypted using one of the following algorithms:

- AES CBC with 128 bit key
- AES CBC with 256 bit key
- AES CTR with 256 bit key
- AES in GCM mode with authenticated encryption and 128 bit keys
- AES in GCM mode with authenticated encryption and 256 bit keys

SSH-RSA is used for SSH transport and Diffie-Hellman Group 14 with SHA 1 or ECDH with SHA 2 and NIST curves P-256, P-384, and P-521 are used for key exchange, according to SP 800-135 Rev 1. Data integrity is verified using HMAC SHA-1 or AES GCM with authenticated encryption with 128 or 256 bit keys.

All SSH connections are rekeyed after 900MB of traffic has been transmitted using the same key or the session has been active for 3000 seconds. Hardware constraints restrict all models except the RSG2488F from being able to transmit 1GB within an hour. To do so would require a transmission rate of approximately 280 KB/s. The NXP ColdFire CPU is capable of receiving or transmitting at most 60KB/s using aes-128-cbc and is therefore not able to process 1GB within an hour. The RSG2488F model, with the NXP PowerQUICC, is capable of meeting this threshold. It is not possible to change the CPU in the TOE, therefore this limitation will always exist.

FCS_TLSS_EXT.1

The TOE uses TLS v1.1 or 1.2 for secure management connections to the web interface. The SFR lists all ciphersuites implemented by the TOE. If a connection is requested using TLS v1.0, any version of SSL, or a ciphersuite not listed in the SFR, the TOE will reject the connection and a browser error will be seen on the client side. The TLS session establishes keys using one of the following methods:

- RSA with key size of 2048 or 3072 bits
- ECDHE with NIST curves secp256r1, secp384r1, or secp521r1
- Diffie-Hellman with a 2048 bit parameter

Mutual authentication is not supported by the TOE.

FCS_RBG_EXT.1

In order to generate secure keys, the TOE implements a CTR DRBG using AES. The DRBG relies on a software noise source to provide 256 bits of entropy for seeding the DRBG. Prediction resistance is enabled on the DRBG.

TOE Security Functional Requirements Satisfied:

- FCS_CKM.1
- FCS_CKM.2

- FCS_CKM.4
- FCS_COP.1/DataEncryption
- FCS_COP.1/SigGen
- FCS_COP.1/Hash
- FCS_COP.1/Keyed Hash
- FCS_HTTPS_EXT.1
- FCS_SSHS_EXT.1
- FCS_TLSS_EXT.1
- FCS_RBG_EXT.1

Section 7.1.3

Identification and Authentication

FIA_AFL.1

The TOE will only allow 1 Administrator account to use password-based authentication while operating in the evaluated configuration. The Operator and Guest accounts are disabled. The TOE allows the Administrator to configure the number of unsuccessful login attempts to the TOE with a configurable value from 1 to 20 attempts per service, i.e. SSH or Web UI. The TOE will disable the password-based login authentication for a service when the number of successive, unsuccessful authentication attempts has been reached within a defined time window called the “Failed Attempts Windows”. This time period is not associated with the lockout period that RUGGEDCOM ROS imposes when the number of unsuccessful attempts is met. The TOE will also disable the password-based login authentication when the number of unsuccessful authentication attempts has been reached within the “Failed Attempts Window” when the attempts are not successive, meaning that a successful authentication within that time period does not reset the counter. The counter is reset to 0 when the “Failed Attempts Window” expires. If the device is rebooted, the admin account will be reenabled. This means that the TOE will not disable password-based login authentication for a service when the number of unsuccessful authentication attempts happens outside the “Failed Attempts Window” even if the attempts are successive, unsuccessful authentication attempts. The “Failed Attempts Window” can be configured to be between 1 and 30 minutes.

Since the Administrator is the only valid TOE user, all login attempts are to the Administrator account and any unsuccessful attempts are counted against the Administrator’s unsuccessful attempts. Authentication on the service is blocked by disabling the authentication mechanism on that service until the Lockout Time is reached. The Lockout Time can be configured by an Administrator to between 1 to 120 minutes. The service and port remain open. When password-based logins are blocked on SSH, they are still available via the Web UI, the local console, or SSH using public key-based authentication. SSH key-based authentication is a more secure authentication mechanism and it will not be disabled/blocked by unsuccessful authentication attempts. If logins are blocked on both the Web UI and SSH, local console is always available to prevent a denial of service attack. The local console is protected by the secure environment. This will prevent a remote threat from to attempting to access the TOE by brute force attacks.

Only the user using password authentication via a specific service (e.g. SSH) will be blocked for that service (e.g. SSH). Password authentications via other services are not affected, meaning that the same user blocked on password authentication on SSH can still be successfully authenticated using the password via the HTTPS service or local Console.

FIA_PMG_EXT.1

Authorized administrators can configure the password to be from a minimum length of one (1) characters to a maximum length of nineteen (19) characters. Valid passwords can be composed of any combination of upper and lower case letters, numbers, and other printable characters including: !, @, #, $, %, ^, &, *, (, and). Administrators can configure the password requirements from the shell within the CLI using the passwordCfg table.

FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7

All interfaces and access methods show an access banner prior to authentication. The TOE will respond to ICMP echoes and perform HTTP redirect and TCP handshakes prior to authentication. No other actions are allowed prior to authentication. Only authentication to the local file system is provided in the evaluated configuration. On the terminal-based menu, a user can access via serial connection or SSH. When accessing via serial access, the access banner is shown and then the user is prompted for their username. After entering the username, the user is prompted for the user's password. No feedback is shown when entering the password. The TOE verifies the username and password against the local user file. If the password is verified, the user is shown a list of menus available to their role. The same process is used when authenticating to the terminal-based menu via SSH except that an SSH client is required. The SSH client will perform initial key exchange and display the configured banner prior to performing user authentication via either password or public key.

Access via the web interface is similar. A TLS session is established with the TOE via the client's web browser. Once the TLS session is established, the login page is displayed. The login page includes the access banner and fields for username and password. The password is displayed as dots when entering on the web interface. Once the user enters their authentication data and presses submit, the data is sent to the TOE and verified against the local user file. Once the password is verified, the web interface displays the list of menus available to the user's role. This list is similar to the menus available on the terminal-based menu.

FIA_X509_EXT.1/REV, FIA_X509_EXT.2

The TOE provides an X.509 certificate to support authentication in TLS server sessions. A Security Administrator must first provision the public portion of a trusted CA certificate into the sslpub.certs file. This file is in PEM format and is preceded by the following header:

- `<index>, <active|inactive>, <rootca|signingca>`

Where each component in the header indicates the following:

- **Index**
  A unique integer value used to identify the entry
- **Active**
  Means the certificate has been loaded into the truststore
- **Inactive**
  Means the certificate has not been loaded into the trust store
- **Rootca**
  Signifies that this certificate is the certificate of a root CA
- **Signingca**
  Signifies that this certificate is from an intermediate certificate issuer

All certificates are verified against the certificate chain in the ssl.crt file. This chain terminates with the trusted CA certificate. The TOE calls out to the external OCSP server listed in the uploaded chain to verify revocation status when verifying new certificates. The TOE verifies the extendedKeyUsage field within certificates based on the rules outlined in the SFR. If the TOE cannot connect to the OCSP server to check for revocation, the Security Administrator can configure the functionality of the TSF. The TSF can be configured to ACCEPT or REJECT certificates in these cases. If REJECT is selected, entire chain of certificates are rejected. In the case where this a rejection leaves no valid certificates, the TOE web server is disabled. If the OCSP server returns that a certificate

has been revoked, the TOE will destroy the certificate to ensure it can no longer be used. A certificate's status is additionally checked via the OCSP server at regular intervals once it has been added to ensure it remains valid.

» FIA_X509_EXT.3

Security Administrators can create a Certificate Request Message to receive a certificate generated by a trusted CA by specifying the following parameters in the web interface:

- Common Name
- Email Address
- Organization
- Organizational Unit
- Locality
- State or Province
- Country
- PKI key type and size

The TOE then generates a PKI key pair and a csr.txt file. The file contains the CSR and the private key. The Administrator downloads this file and submits the CSR PEM to the appropriate CA for signature. The CA returns the signed certificate and certificate chain. The Administrator loads the trust anchor(s) into the sslpub.certs file via SFTP. These may be root or intermediate CA certificates to which the TLS server certificate chain traces its authenticity. The TLS server certificate chain and TLS server certificate's private key are loaded into the ssl.crt file in PEM format. The server certificate appears first, followed by each issuer CA certificate in the hierarchy. The last PEM object in the ssl.crt file is the TLS server certificate's private key pair. The TOE then validates all uploaded certificates prior to accepting the new certificate. If any certificate within the chain cannot be verified, the certificate is rejected.

TOE Security Functional Requirements Satisfied:

- FIA_PMG_EXT.1
- FIA_UIA_EXT.1
- FIA_UAU_EXT.2
- FIA_UAU.7
- FIA_X509_EXT.1/REV
- FIA_X509_EXT.2
- FIA_X509_EXT.3

Section 7.1.4

Security Management

Security management specifies how the TOE manages several aspects of the TSF including TSF data and security functions. TSF data includes configuration data of the TSF and audit data. The TOE provides authorized administrators over the web interface and terminal-based menu to easily manage the security functions and TSF data of the TOE.

FMT_MOF.1/ManualUpdate, FMT_MTD.1/CoreData

A Security Administrator can update the TOE by transferring the TOE binary over SFTP to the TOE and resetting the device. The device can be reset from the CLI or web interface in the Diagnostics menu. The Diagnostics menu additionally provides the Security Administrator with the ability to configure alarms, view and clear the log, load factory defaults and reset the device. The Administration menu on both interfaces provides the Security Administrator with the ability to manage passwords. Additionally the CLI shell allows the Security Administrator to query system configurations. Security Administrators can import certificates to the TOE and upload new configuration files to modify configurations via SFTP. Only acceptance of the access banner is allowed prior to authenticating. The TOE does not provide any additional options to allow other actions prior to authentication.

FMT_SMF.1, FMT_SMR.2

The TOE provides a role called Administrator. There are also Operator and Guest roles, but these are disabled in the evaluated configuration. The Administrator role holds the Security Administrator privileges discussed in this ST. Administrators can manage the TOE remotely via the web interface over HTTPS/TLS or the terminal-based menu. The terminal-based menu can be accessed locally via RS-232 or remotely over SSH. The TOE supports one user (in admin role) for the password authentication mechanism (via SSH/HTTPS/Console) and additional users (in admin role) authenticated using key-based authentication (via SSH only). There is only one password-based user which is created by default and has a configurable username and password. To allow multiple new users to access the TOE, the password-based user with the role of Administrator can be used to add SSH public keys for specified user names and associate them with the Administrator role. For information on managing this, refer to the "Managing SSH Public Keys" section of the *RUGGEDCOM RUGGEDCOM ROS v4.2.2.F User Guides*. Security Administrators can configure the following on either the web interface or the terminal-based menu:

- Configure session inactivity timeouts
- Reset the device to apply an update
- Configure the authentication failure parameters
- Configure the system time
- Reboot the appliance

Security Administrators can configure the following only through the terminal-based menu:

- Upload a text file that is used to generate an access banner
- Upload the update

Rebooting the appliance will unlock an Administrator account that has been lockout due to authentication failures. Signature verification is an internal process that is automatically triggered by uploading an update. No administrator action is required to perform the verification. The Security Administrator can also configure the system time used for time stamps.

TOE Security Functional Requirements Satisfied:

- FMT_MOF.1/ManualUpdate
- FMT_MTD.1/Coredata
- FMT_SMF.1
- FMT_SMR.2

Section 7.1.5

Protection of the TSF

FPT_APW_EXT.1

A salted hash of the password is stored in the TOE's user configuration file. This hash is created using an SHA-512 hash. Only the Security Administrator has the ability to read the salted and hashed passwords. The Security Administrators are trusted not to read these files.

FPT_SKP_EXT.1

Private keys are stored in the TOE's file system with access controls on the file. No reading of these files is allowed. The files can be zeroized and updated by an authorized Security Administrator.

FPT_STM_EXT.1

The TOE provides a reliable time stamp for use in the audit function. TOE maintains an internal date and time, this system time can be modified by an Administrator in the ***Administration » System Time Manager*** menu. The NTP service is excluded from the TOE and must be disabled..

FPT_TST_EXT.1

The TOE performs FIPS power-up and conditional self-tests for all cryptographic algorithms. The following is a description of each of the start-up tests:

- **Firmware Integrity Test (RUGGEDCOM ROS Firmware Components)**
  The TOE performs a firmware integrity check using a SHA-256 hash to ensure that the firmware binary image is not corrupted or tampered with. At power-up, the modules compute a new hash and compare it to the pre-computed value which is decrypted using an RSA 2048 public key. If the values are same, the test is passed; otherwise, it is failed.
- **AES Encrypt/Decrypt CBC KAT**
  The AES CBC KAT takes a known 256-bit key and plaintext value, which is encrypted and compared to the expected ciphertext value. If the values differ, the test is failed. The AES KAT then reverses this process by taking the ciphertext value and key, performing decryption, and comparing the result to the known plaintext value. If the values differ, the test is failed. If they are the same, the test is passed.
- **AES Encrypt/Decrypt GCM KAT**
  The AES GCM KAT encrypts a known plaintext with known 256-bit key. It then compares the resultant ciphertext with the expected ciphertext hard-coded in the module. If the two values differ, then the KAT is failed. If the two values agree, then the AES-GCM KAT decrypts the ciphertext with the known keys and compares the decrypted text with the known plaintext. If they differ, then the test fails. If they are the same, then the test passes.
- **AES Encrypt/Decrypt CTR KAT**
  The AES CTR KAT takes a known 256-bit key and plaintext value, which is encrypted and compared to the expected ciphertext value. If the values differ, the test fails. The AES CTR KAT the reverses this process by taking the ciphertext value and key, performing decryption, and comparing the results to the known plaintext value. If the values differ, the test is failed. If they are the same, the test passes.
- **HMAC KAT with SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512**
  The HMAC implementation creates a MAC using known input data and known 256-bit key. This MAC value is then compared to the expected MAC value. If the values differ, the test fails. If they are the same, the test passes.

- **SHA KAT with SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512**
  The SHA implementation is further tested in a SHA KAT. Again, a known 256 bit input data is used and a hash is created of the input data. This hash is compared to the expected hash. If the values differ, the test fails. If they are the same, the test passes.
- **CTR DRBG KAT**
  An entropy input string is used to create the seed and after initialization, the DRBG generates random bits by using this seed and provided additional input. These generated random bits are discarded. The DRBG is then reseeded with reseed entropy and an additional reseed input for creating the new seed. This new seed along with the provided second additional input generates the random bits which are compared with the known bits. If there is a match, the test passes; if there is no match, the test fails.
- **RSA Signature Generation/Verification KAT**
  A known private 2048-bit (with SHA-256) key is used to sign a known block of. data, and the resultant value is compared with the expected ciphertext. If they differ, the test fails. If they are the same, then the public key is used to decrypt the ciphertext and the output is compared to the original data. If they are the same, the test passes. Otherwise, it is failed.
- **ECDSA Pair-wise Consistency Test**
  The ECDSA private key is used to sign a block of data. The resulting signature is compared to the original data before it was signed. If the two values are equal, then the test fails. If the two values differ, the ECDSA public key is used to verify the signature and the resulting value is compared to the original data. If they are the same, the test is passed. Otherwise, it is failed.
- **Primitive “Z” Computation KAT (Crypto Library)**
  The EC DH KAT requires two public/private key pairs obtained from one of the NIST-Approved P-521 curves. A shared secret is generated using the private key of the first key pair (derived from the P curve) with the public key of the second key pair (derived from the P curve). A second shared secret is generated using the private key from the second key pair and the public key of the first key pair. If the shared secret values differ, the test fails. If they are the same, the test passes.
- **Continuous Random Number Generator Test for DRBG**
  This test is activated on the DRBG implementation whenever a fresh random value is requested. The new random number returned from the DRBG will be compared with the previous random number from the same DRBG to determine if stuck-at-constant type of failure is occurring.
- **Continuous Random Number Generator Test for NDRNG**
  This test is activated on the NDRNG implementation whenever a fresh random value is requested. The new random number returned from the NDRNG will be compared with the previous random number from the same NDRNG to determine if stuck-at-constant type of failure is occurring.
- **RSA Pair-wise Consistency Test**
  The RSA private key is used to sign a block of data. The resulting signature is compared to the original data before it was signed. If the two values are equal, then the test fails. If the two values differ, the RSA public key is used to verify the signature and the resulting value is compared to the original data. If they are the same, the test is passed. Otherwise, it is failed.
- **ECDSA Signature Generation/Verification Test**
  The generated ECDSA private key is used to sign a known block of data, which outputs two signature components. The public key curve point is used to verify the signature. If the verification matches, the test passes. Otherwise, it fails.
- **ECC Full Public-key Validation Test**
  Whenever a public key is generated for ECC Diffie-Hellman key agreement, the module will implement the elliptical curve cryptography full public key validation routine outlined in Section 5.6.2.3.2 of SP 800-56A.
- **EC Diffie-Hellman Public Key Assurance Test**
  Whenever a public key is generated for Diffie-Hellman key agreement, the module will implement the elliptical curve cryptography full public key validation routine outlined in Section 5.7.1.2 of SP 800-56A.

FPT_TUD_EXT.1

Security Administrators can download firmware updates from https://www.siemens.com/ruggedcom. These files can then be loaded onto the TOE using SFTP. The files are signed using a Siemens key. Once the files are on the TOE, the TOE decrypts the digital signature of the firmware binary using the compiled-in Siemens public key. If the signature is verified the installation can occur. If the signature fails, the installation operation stops and an error is logged. Once the signature is validated, the TOE must be reset to complete the installation. The Security Administrator can query the TOE version through the CLI shell version command.

TOE Security Functional Requirements Satisfied:

- FPT_APW_EXT.1
- FPT_SKP_EXT.1
- FPT_STM_EXT.1
- FPT_TST_EXT.1
- FPT_TUD_EXT.1

Section 7.1.6

TOE Access

FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4

Local and interactive sessions with the TOE are terminated after a period of inactivity. This period can be configured by a Security Administrator using the ***Administration » Configure IP Services » Inactivity Timeout*** menu. This timeout time is valid on all interfaces. Administrators can also terminate their own session by typing "logout" on the CLI or clicking Log Out on the web interface.

FTA_TAB.1

The TOE provides an access banner on the web interface and terminal-based menu prior to accessing the TOE. This banner can be configured by modifying the banner.txt file. This file can be downloaded from the device via SFTP. Once it has been modified, SFTP can again be used to upload the file to the TOE.

TOE Security Functional Requirements Satisfied

- FTA_SSL_EXT.1
- FTA_SSL.3
- FTA_SSL.4
- FTA_TAB.1

Section 7.1.7

Trusted Path/Channels

FTP_ITC.1

Audit events are sent from the TOE to an external audit server over an SSH trusted channel. The channel is initiated by the audit server and follows all FCS_SSHS_EXT.1 requirements. As part of the channel initiation the

audit server sends an SSH public key to identify itself to the TOE. The TOE does not initiate any secure channels for any services.

FTP_TRP.1/Admin

Secure remote management is performed using SSH to the terminal-based menu or HTTPS/TLS to the web interface. These remote sessions conform to the requirements in FCS_SSHS_EXT.1 or FCS_HTTPS_EXT.1 and FCS_TLSS_EXT.1.

TOE Security Functional Requirements Satisfied

- FTP_ITC.1
- FTP_TRP.1/Admin

8 Conformance Claims Rationale

This Security Target conforms to Part 2 extended and Part 3 conformant of the Common Criteria Standard for Information Technology Security Evaluations, Version 3.1 Release 4. This ST conforms to the ND cPP v2.0E.

CONTENTS

Section 8.1

Variance Between the PP and this ST

There is no variance between the ND cPP v2.0E and this ST.

Section 8.2

Security Assurance Requirements Rationale

This ST maintains exact conformance to the ND cPP v2.0E, including the assurance requirements listed in Section 6 of the ND cPP v2.0E.

Section 8.3

Dependency Rationale

This ST does satisfy all the requirement dependencies of the Common Criteria. Table lists each requirement to which the TOE claims conformance with a dependency and indicates whether the dependent requirement was included. As Table 11 indicates, all dependencies have been met.

Table 11: Functional Requirements Dependencies

SFR ID	Dependencies	Dependency Met	Rationale
FAU_GEN.1	FPT_STM.1	✓	FPT_STM_EXT.1 included (which is hierarchic to FPT_STM.1)
FAU_GEN.2	FIA_UID.1	✓	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification timing
	FAU_GEN.1	✓	FAU_GEN.1 included

SFR ID	Dependencies	Dependency Met	Rationale
FAU_STG_EXT.1	FAU_GEN.1	✓	FAU_GEN.1 included
	FTP_ITC.1	✓	FTP_ITC.1 included
FCS_CKM.1	FCS_CKM.2	✓	FCS_CKM.2 included
	FCS_CKM.4	✓	FCS_CKM.4 included
FCS_CKM.2	FCS_CKM.1	✓	FCS_CKM.1 included
	FCS_CKM.4	✓	FCS_CKM.4 included
	FTP_ITC.1	✓	FTP_ITC.1 as a secure channel that is used for import
FCS_CKM.4	FCS_CKM.1	✓	FCS_CKM.1 included
FCS_COP.1/DataEncryption	FCS_CKM.4	✓	FCS_CKM.4 included
	FCS_CKM.1	✓	FCS_CKM.1 included
FCS_COP.1/SigGen	FCS_CKM.4	✓	FCS_CKM.4 included
	FCS_CKM.1	✓	FCS_CKM.1 included
FCS_COP.1/Hash	FCS_CKM.4	✓	This SFR specifies keyless hashing operations, so initialisation and destruction of keys are not relevant
	FCS_CKM.1	✓	
FCS_COP.1/KeyedHash	FCS_CKM.1	✓	FCS_CKM.1 included
	FCS_CKM.4	✓	FCS_CKM.4 included
FCS_HTTPS_EXT.1	FCS_TLSS_EXT.1	✓	FCS_TLSS_EXT.1 included as selection-based SFR
FCS_SSHS_EXT.1	FCS_CKM.1	✓	FCS_CKM.1 included
	FCS_CKM.2	✓	FCS_CKM.2 included
	FCS_COP.1/DataEncryption	✓	FCS_COP.1/DataEncryption included
	FCS_COP.1/SigGen	✓	FCS_COP.1/SigGen included
	FCS_COP.1/Hash	✓	FCS_COP.1/Hash included
	FCS_COP.1/KeyedHash	✓	FCS_COP.1/KeyedHash included
	FCS_RBG_EXT.1	✓	FCS_RBG_EXT.1 included
FCS_TLSS_EXT.1	FCS_CKM.1	✓	FCS_CKM.1 included
	FCS_CKM.2	✓	FCS_CKM.2 included
	FCS_COP.1/DataEncryption	✓	FCS_COP.1/DataEncryption included
	FCS_COP.1/SigGen	✓	FCS_COP.1/SigGen included
	FCS_COP.1/Hash	✓	FCS_COP.1/Hash included
	FCS_COP.1/KeyedHash	✓	FCS_COP.1/KeyedHash included
	FCS_RBG_EXT.1	✓	FCS_RBG_EXT.1 included
FCS_RBG_EXT.1	No dependencies	✓	
FIA_AFL.1	FIA_UAU.1	✓	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication.

SFR ID	Dependencies	Dependency Met	Rationale
FIA_PMG_EXT.1	No dependencies	✓	
FIA_UIA_EXT.1	FTA_TAB.1	✓	FTA_TAB.1 included
FIA_UAU_EXT.2	No dependencies	✓	
FIA_UAU.7	FIA_UAU.1	✓	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication.
FIA_X509_EXT.1/REV	No dependencies	✓	
FIA_X509_EXT.2	No dependencies	✓	
FIA_X509_EXT.3	FCS_CKM.1	✓	FCS_CKM.1 included
FMT_MOF.1/ManualUpdate	FMT_SMR.1	✓	FMT_SMR.2 included
	FMT_SMF.1	✓	FMT_SMF.1 included
FMT_MTD.1/CoreData	FMT_SMF.1	✓	FMT_SMF.1 included
	FMT_SMR.1	✓	FMT_SMR.2 included
FMT_SMF.1	No dependencies	✓	
FMT_SMR.2	FIA_UID.1	✓	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication.
FPT_APW_EXT.1	No dependencies	✓	
FPT_SKP_EXT.1	No dependencies	✓	
FPT_STM_EXT.1	No dependencies	✓	
FPT_TST_EXT.1	No dependencies	✓	
FPT_TUD_EXT.1	FCS_COP.1/Hash	✓	FCS_COP.1/Hash included
FTA_SSL_EXT.1	FIA_UAU.1	✓	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication.
FTA_SSL.3	No dependencies	✓	
FTA_SSL.4	No dependencies	✓	
FTA_TAB.1	No dependencies	✓	
FTP_ITC.1	No dependencies	✓	
FTP_TRP.1/Admin	No dependencies	✓	

9 Acronyms and Terms

This section describes the acronyms and terms used throughout the document.

CONTENTS

Section 9.1

Acronyms

Table 12: Acronyms

Acronym	Definition
AEAD	Authenticated Encryption with Associated Data
AES	Advanced Encryption Standard
°C	Celsius
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CC	Common Criteria
CEM	Common Evaluation Methodology
CLI	Command Line Interface
cPP	Collaborative Protection Profile
CRL	Certificate Revocation List
CSP	Critical Security Parameters
CTR	Counter Mode
CVL	Component Validation List
DHE	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DSS	Digital Signature Standard
EAL	Evaluation Assurance Level
EC	Elliptical Curve

Acronym	Definition
ECDH	Elliptical Curve Diffie-Hellman
ECDSA	Elliptical Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standards
GCM	Galois Counter Mode
HMAC	(Keyed-)Hashed Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IPsec	Internet Protocol Security
ISO	International Standards Organization
IT	Information Technology
MAC	Media Access Control
MB	Megabytes
ND cPP	Collaborative Protection Profile for Network Devices v2.0E
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
OS	Operating System
OSP	Organizational Security Policy
PP	Protection Profile
RADIUS	Remote Authentication Dial-In User Service
RBG	Random Bit Generator
RFC	Request for Comment
RSA	Rivest-Shamir-Adleman
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SFTP	Secure File Transfer Protocol
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TACACS+	Terminal Access Controller Access Control System Plus

Acronym	Definition
TCP	Transmission Control Protocol
TD	Technical Decision
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality

Section 9.2

Terms

Table 13: Terms

Name	Definition
Administrator	See “Security Administrator.”
Assurance	Grounds for confidence that a TOE meets the SFRs.
Security Administrator	The terms “Administrator” and “Security Administrator” are used interchangeably in this document at present.
Target of Evaluation	A set of software, firmware and/or hardware possibly accompanied by guidance.
TOE Security Functionality (TSF)	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the SFRs.
TSF Data	Data for the operation of the TSF upon which the enforcement of the requirements relies.
User	See “Security Administrator”.