RUCKUS

an ARRIS company

Security Target for Ruckus Solution

Version: 1.2

June 14, 2019

TABLE OF CONTENTS

ABBREVIATIONS

Abbreviation	Description
AEAD	Authenticated Encryption with Associated Data
AES	Advanced Encryption Standard
CA	Certificate Authority
CBC	Cipher Block Chaining
CC	Common Criteria
CLI	Command-line interface
CM	Configuration Management
CMVP	Cryptographic Module Validation Program
CRL	Certificate Revocation List
DH	Diffie-Hellman
DNS	Domain Name System
DSA	Digital Signature Algorithm
ECDH	Elliptic Curve Diffie Hellman
DTLS	Datagram Transport Layer Security
ECDSA	Elliptic Curve Digital Signature Algorithm
EEPROM	Electrically Erasable Programmable Read-Only Memory
FIPS	Federal Information Processing Standards
GCM	Galois Counter Mode
HMAC	Keyed-Hash Message Authentication Code
HTTPS	HyperText Transfer Protocol Secure
IP	Internet Protocol
IPsec	Internet Protocol Security
IT	Information Technology
NAT	Network Address Translation
NIST	National Institute of Standards and Technology
OS	Operating System
OCSP	Online Certificate Status Protocol
PP	Protection Profile
PUBS	Publications
RBG	Random Bit Generator
RSA	Rivest Shamir Adleman Algorithm
SD	Supporting Document
SF	Security Function
SHA	Secure Hash Algorithm
SSH	Secure Shell
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification
VPN	Virtual Private Network

GLOSSARY

Term	Meaning
Administrator	See Security Administrator.
Assurance	Grounds for confidence that a TOE meets the SFRs
Key Chaining	The method of using multiple layers of encryption keys to protect data. A top layer key encrypts a lower layer key which encrypts the data; this method can have any number of layers.
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

1. SECURITY TARGET INTRODUCTION

The Security Target for Ruckus Solution contains the following sections:

- Section 1: Security Target Introduction
- Section 2: CC Conformance Claims
- Section 3: Security Problem Definition
- Section 4: Security Objectives
- Section 5: Security Functional Requirements
- Section 6: Security Assurance Requirements
- Section 7: TOE Summary Specification

1.1 ST and TOE Reference

ST title and version: Security Target for Ruckus Solution Version 1.2

TOE name: Ruckus SmartZone WLAN Controllers & Access Points

TOE version: 5.1.1.3

Hardware versions:

- Wireless Controllers:
  - Smart Zone 100 (includes SZ-104 and SZ-124 models)
  - Smart Zone 300 (SZ 300)
  - Ruckus virtual SmartZone (includes vSZ-E and vSZ-H) running on a specific physical hardware platform (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz with AES-NI)
  - Ruckus virtual SmartZone – Data plane (vSZ-D) running on a specific physical hardware platform (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz with AES-NI)

- Access Points:
  - R610
  - R710
  - R720
  - E510
  - T610 (including T610S)
  - T710 (including T710S)

Software version: 5.1.1.3

1.2 TOE Overview

Ruckus Wireless Controller has been designed to eliminate the difficulties administrators are experiencing with building and managing large-scale WLAN networks, to support several Wi-Fi access points and many concurrent Wi-Fi clients. It offers one of the industry's most scalable WLAN controller architectures.

Ruckus Wireless Controllers can support tens of thousands of Ruckus Smart Wi-Fi APs and hundreds of thousands of concurrent Wi-Fi subscribers. The Ruckus carrier-class management system provides feature-rich management of access points, such as RF management, load balancing, adaptive meshing and backhaul optimization and secure connectivity to all wireless clients.

1.2.1 Deployment Models

Ruckus Wireless Controllers and Ruckus Smart Wi-Fi APs are deployed in two different models; distributed deployment model and centralized deployment model.

DISTRIBUTED DEPLOYMENT MODEL

In distributed deployment model client traffic directly reaches the intended destination from the AP. All Ruckus Wireless Controllers and APs support this deployment model. See figure 1.

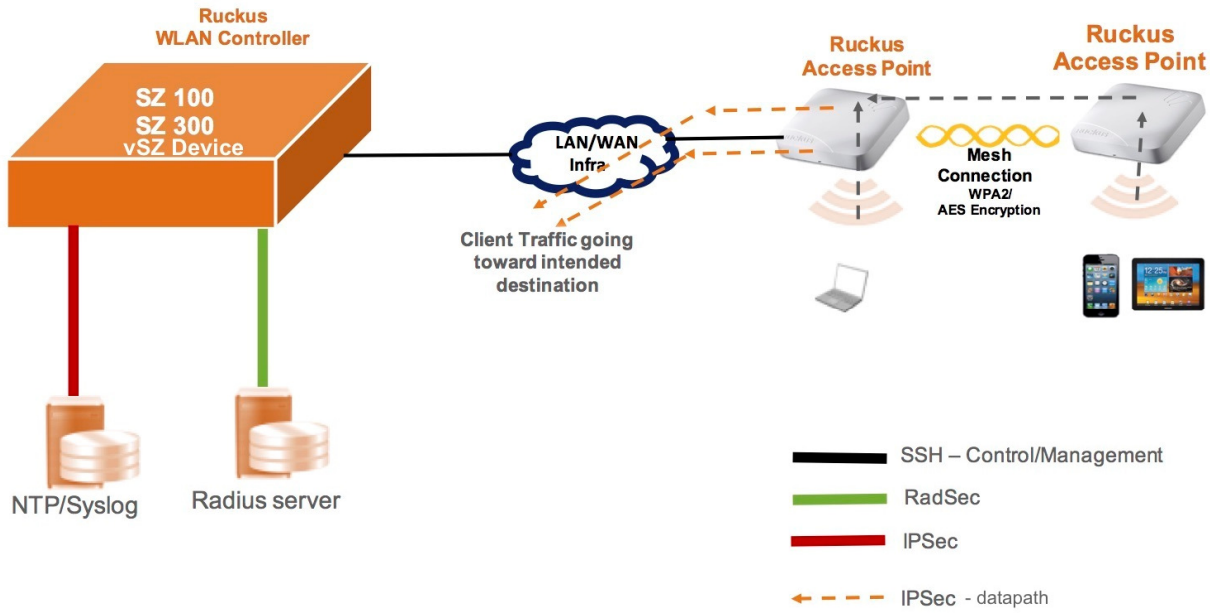

Figure 1: Distributed Deployment Model

CENTRALIZED DEPLOYMENT MODEL

In centralized deployment model client traffic always reaches the WLAN controller first via the AP before going to intended destination. See figure 2 and 3.

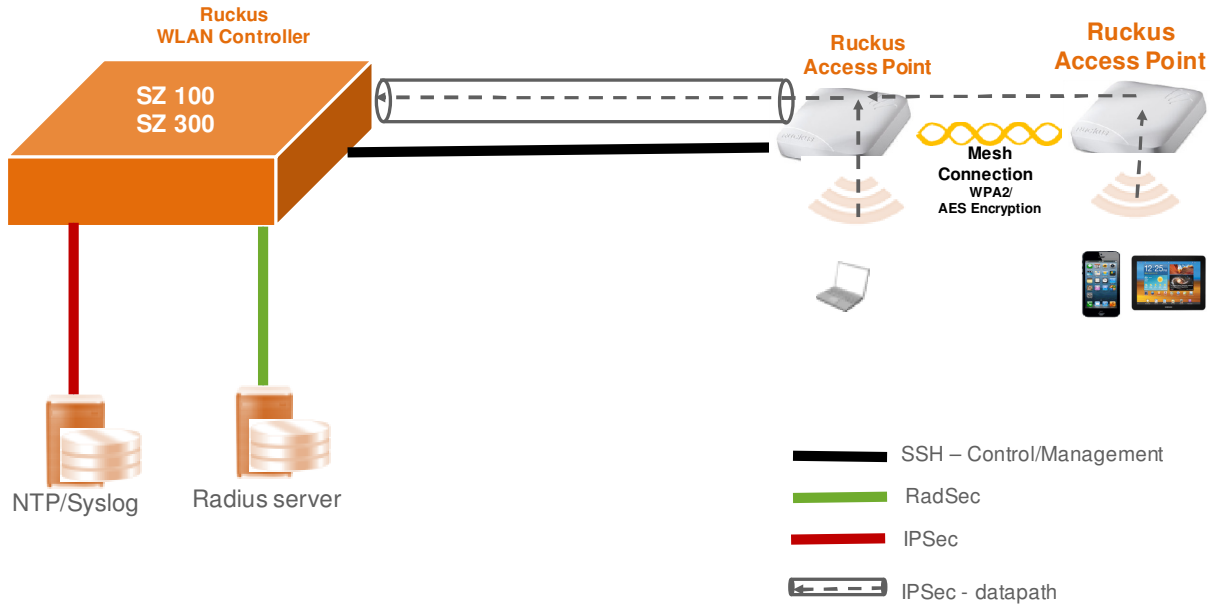

Figure 2: Centralized Deployment Model with hardware

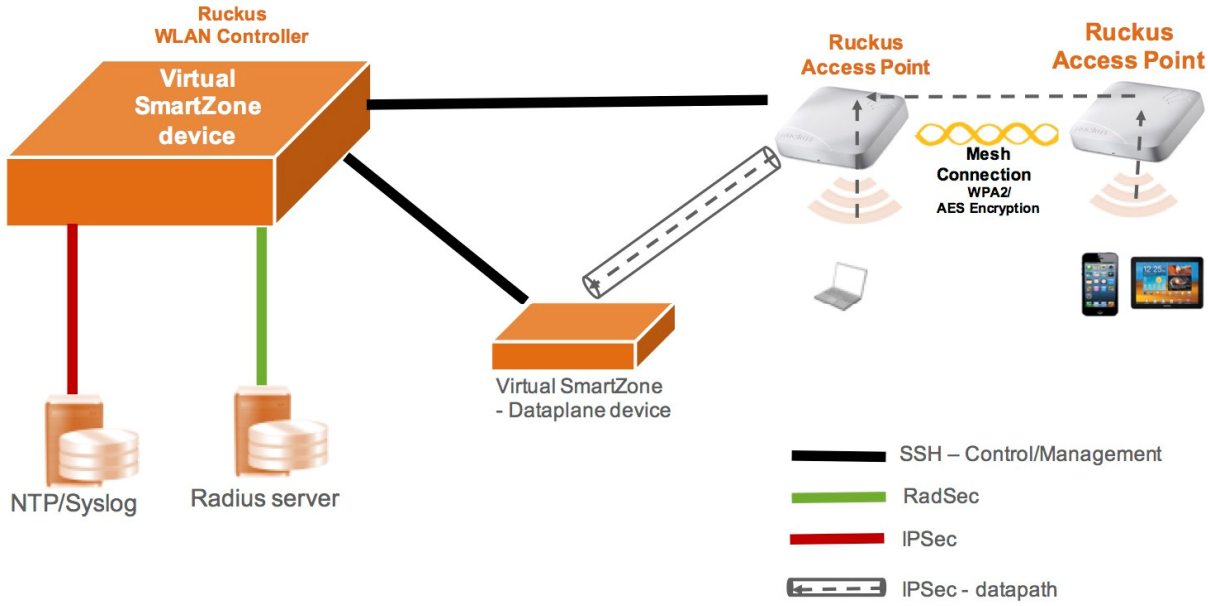

Figure 3: Centralized Deployment Model with Software

Once authenticated as trusted nodes on the wired infrastructure, the access points provide the encryption service on the wireless network between themselves and the wireless client. The APs also communicate directly with the wireless controller for management purposes. The management traffic between Ruckus AP and Ruckus Wireless Controller is encrypted.

1.3 TOE Description

The Ruckus SmartZone controllers and Access points Solution (TOE) is a Wireless LAN access system (WLAN). The Wireless LAN access system defined in this ST are multiple products operating together to provide secure wireless access to a wired and wireless network. The TOE provides end-to-end wireless encryption, centralized WLAN management, authentication, authorization, and accounting (AAA) policy enforcement. The TOE has the following Access Point TOE components: R610, R710, R720, E510, T610 and T710. The TOE also has the following Wireless Controllers: SmartZone 100 (SZ-104 and SZ-124), SmartZone 300 (SZ 300), virtual SmartZone (vSZ-E and vSZ-H hosted on a physical device), and virtual SmartZone – Data plane (vSZ-D hosted on a physical device).

1.3.1 Wireless Controller

The wireless controller serves client devices using secure authentication protocols, such as 802.1X/EAP. This is combined with policy-based data traffic steering which enterprises can optimize to forward all client traffic appropriately.

The wireless controller can function as a very large-scale WLAN controller that can manage a lot of access points, providing feature-rich management including control over their self-organizing smart networking behaviors such as RF management, load balancing, adaptive meshing, and backhaul optimization.

SZ 100
SmartZone™ 100 (SZ-104 and SZ-124) is a Scalable, Resilient, and High Performing Wireless LAN controller for Enterprises. It manages up to 1,024 Wi-Fi access points, 2,000 WLANs, and 25,000 clients per device. SmartZoneOS' unique architecture enables SZ 100 to be deployed in multiple architectures like centralized and distributed traffic forwarding. Smart licensing allows customers to manage all the licensing needs online. With Smart licensing, customers will have the ability to buy and assign licenses as granular as 1 (one) AP license.

SZ 300
The SmartZone™ 300 (SZ 300) Flagship Large Scale WLAN Controller is designed for Service Provider and Large Enterprises, which prefer to use appliances. The Carrier Grade platform supports, comprehensive integrated management functionality, high performance operations and flexibility to address many different deployment scenarios. The SZ300 supports up to 10,000 AP and 100,000 Clients per unit.

vSZ
The Ruckus Virtual SmartZone™ (vSZ-E and vSZ-H hosted on a physical device) is an NFV-based WLAN controller for service providers and enterprises ready to elevate their WLAN deployment to the next level of flexibility, resiliency, and scale.

vSZ-D
With the Virtual SmartZone Data Plane (vSZ-D hosted on a physical device), the Ruckus Virtual SmartZone platform launches sophisticated data plane capabilities that enable tunneled WLANs architectures. This is an industry-first, truly differentiated and distinguished offering that provides compelling architecture flexibility that translates into business benefits for varied deployment scenarios.

1.3.2 Access Point

The AP components can be centrally managed by the Ruckus Wireless Controller as part of a unified indoor/outdoor wireless LAN. Each AP supports a wide range of value-added applications.

Wireless communications between clients and APs is carried out using the IEEE 802.11 protocol standard. The 802.11 standard governs communication transmission for wireless devices. For this evaluation, the APs use a variation within 802.11a, 802.11ac, 802.11b, 802.11g and 802.11n for wireless communication. The wireless security protocols that are to be used with the APs are 802.1X/802.1i.

The AP components combine patented adaptive antenna technology and automatic interference mitigation to deliver consistent and predictable performance by means of BeamFlex, which is a software-controlled, high gain antenna array that continually forms and directs each 802.11n packet over the best performing signal path. The APs automatically select channels for highest throughput potential using ChannelFly dynamic channel management, adapting to environmental changes. ChannelFly uses actual activity to learn what channels will yield the most capacity to provide the highest client speeds and reduced interference, and selects automatically the best performing channel based on statistical, real-time capacity analysis of all RF channels.

The AP part of the TOE consists of six different component products:

- R610
- R710
- R720
- E510
- T610 (including T610S)
- T710 (including T710S)

Non-TOE hardware/software required by the TOE for operation are the servers (RADIUS, CA, Syslog, NTP, DHCP), wireless client, local and remote management computers.

R610
The Ruckus R610 delivers the right combination of performance and affordability for medium-density locations. It provides fast 802.11ac data rates (up to 1900Mbps), with patented Ruckus Wi-Fi intelligence to support dozens of users with guaranteed throughput.

T610
The Ruckus T610 access point delivers blazing-fast Wi-Fi for medium-density outdoor deployments, with data rates up to 2.5 Gbps—the highest available for Wi-Fi clients. Patented Ruckus adaptive antenna technology improves signal quality for every connected device, everywhere.

R710
The ZoneFlex R710 ensures the most reliable connectivity within challenging and ever-changing RF environments. With BeamFlex+, the ZoneFlex R710 is capable of delivering 6 dB of signal-to-interference-plus-noise (SINR) improvement and up to 15 dB of interference mitigation over other APs. R710 is 4x4:4 and MU-MIMO capable. The ZoneFlex R710 simultaneously supports spatial multiplexing and BeamFlex+ to deliver the best price/performance of any three-stream 802.11ac AP. ZoneFlex R710 is purpose-built for high-capacity, high performance and interference-laden environments. The perfect choice for data-intensive streaming multimedia applications, the ZoneFlex R710 delivers picture perfect HD-quality IP video while supporting VoIP and data applications that have stringent quality of service requirements.

T710
Designed for the highest-density outdoor venues, the Ruckus T710 access point delivers Ruckus' premier Wi-Fi technology in an ultra-lightweight, industrial-grade (IP 67-rated) enclosure. It features patented Ruckus technologies to extend range, mitigate interference, and deliver blazing fast performance—up to 1733Mbps data rate, the highest available for 4 stream MIMO Wi-Fi clients.

R720
The R720 is our highest-capacity four-stream 802.11ac Wave 2 AP. With four streams of MU-MIMO connectivity, it can simultaneously transmit to multiple Wave 2 clients in the widest available channels, drastically improving RF efficiency. And with support for multi-gigabit 2.5 GbE, you can more than double your backhaul capacity without having to add switch ports or run new Ethernet cabling.

E510
The Ruckus E510 802.11ac Wave 2 access point (AP) is designed with a unique, diminutive two element enclosure which separates the RF components from the antenna module. The E510 can be placed unobtrusively inside metal-shielded signage at a bus or train stations, and within a vending machine and display kiosk. The low-profile antenna module can be located externally and linked to the RF module via weather proof cabling.

Logical Boundaries

The following security functions are provided by the TOE:

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

Security audit
The TOE provides auditing capabilities to provide secure and reliable way to trace all changes to the system. Any configuration changes, administrative activities and auditable events are audited both internally and externally over a secure communication channel to an audit server. All audited events have the necessary details like timestamp, event log, event code, identity of the party involved to provide a comprehensive audit trail.

Communication
The distributed TOE offers secure internal TSF communication. Access Points and vSZ-Ds register to the WLAN controller and must be approved by the administrator to communicate with each other as parts of the distributed TOE.

Cryptographic support
The distributed TOE provides cryptographic functions like (but not limited to) for secure administration access via HTTPS and SSH, communication between the distributed parts of the TOE via SSH, TLS and IPSec, wireless communication via WPA2 and communication to external systems

like, NTP, audit log servers via IPSec and RADIUS via TLS. Functions include Key generation, key establishment, key distribution, key destruction, cryptographic operations

Identification and Authentication
The distributed TOE provides secure connectivity to the network for wireless clients via 802.1X authentication. Certificate based authentication is supported via external RADIUS server and password based authentication is supported via strong password requirements covered in the ST. The distributed TOE provides secure password based autehtnication for remote administrators and X.509 certificate based authentication for TOE components. The distributed TOE also provides strong password requirements that can be configured by the admintrator including length, session timeout and password complexity. Consecutive unsuccessful attempts beyond a certain limit will result in locking of the user for a specified duration of time.

Security Management
Administrators of the the TOE can manage the system via secure connection like HTTPS over a web interface including TOE software update. Optionally SSH and console can also be used as a method to configure the system via the SmartZone controller. System provides mechanism to prevent administration from a wireless client. TOE also has ability to configure the session activity timeout of an administrator. TOE provides RBAC (Role Based admin Access Control) to map specific roles for the administrators of the system. TOE provides the ability to configure access banner on the controller.

Protection of TSF
The TOE provides image integrity verification to validate the authenticity of the images before loading them. Upon every boot up, power on self-tests are conducted to validate the integrity of the software components. If power up self tests fail, a quarantine state is entered. All the components of the distributed TOE use X.509 certificates to authenticate and establish a secure connectivity amongst themselves. TOE is a closed system and no external software or hardware components are allowed. TOE also allows configuration of timestamps manually or via the NTP server. TOE protects cryptographic keys and passwords form unauthorized access.

TOE Access
Login banner is offered which provides the ability to have a custom warning/access policy message as per the organization needs. TOE is capable of restricting wireless access based on time of the day and SSID. TOE provides ability to configure an inactivity timeout which terminates the session beyond the inactivity period configured. An administrator can also terminate their own session.

Trusted path / Channels
TOE communicates to external components in a secure manner. Following secure channels are used to communicate externally – TLS for RADIUS, HTTPS for WebUI administration, SSH for CLI administration, IPsec for audit and NTP servers, WPA2 for wireless clients.

Evaluated configuration

The following configuration options are outside the evaluated configuration to meet the NDcPP, WLAN PP requirement

1) Internal captive portal
2) Soft-GRE to external gateway
3) FIPS/CC mode disabled
4) 802.11r
5) Clustering
6) API based configuration

7) GTP tunnel
8) SSH based AP administration
9) Encrypted/Ruckus GRE

2. CC CONFORMANCE CLAIMS

This TOE and ST are conformant with the following specifications: CC Part 2: Security functional components, September 2012, Version 3.1, Revision 4, extended. CC Part 3: Security assurance components, September 2012, Version 3.1, Revision 4, conformant.

The TOE and ST are exactly conformant with collaborative Protection Profile for Network Devices, Version. 2.0 + Errata 20180314, 14 March 2018 (CPP_ND_V2.0E), and Extended Package for Wireless LAN Access System, v1.0, May 29, 2015 (PP_WLAN_AS_EP_V1.0) ("Protection Profiles"). The TOE provides all of the functionality at a level of security corresponding to that identified in the Protection Profiles.

The Assumptions, Threats, and Organization Security Policies included in the Security Target correspond to the Assumptions, Threats, and Organization Security Policies specified in the Protection Profiles for which exact conformance is claimed.

The Security Objectives included in the Security Target correspond to the Security Objectives specified in the Protection Profiles.

The Security Functional Requirements included in the Security Target correspond to the Security Functional Requirements specified the Protection Profiles, for which exact conformance is claimed. Security Assurance Requirements specified in this Security Target are identical to the Security Assurance Requirements included in the Protection Profiles.

3. SECURITY PROBLEM DEFINITION

3.1 Threats

The threats to the TOE and TOE Environment are described in the table below:

Threat	Description
T.UNAUTHORIZED_ ADMINISTRATOR_ACCESS	Threat agents may attempt to gain administrator access to the network device by nefarious means such as masquerading as an administrator to the device, masquerading as the device to an administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between network devices. Successfully gaining administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.

T.UNTRUSTED_ COMMUNICATION_ CHANNELS	Threat agents may attempt to target network devices that do not use standardized secure tunneling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the network device itself.
T.WEAK_AUTHENTICATION_ ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the network device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the network device without administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_ COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other network devices.
T.SECURITY_FUNCTIONALITY_ FAILURE	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.
T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress- or egress-based actions.
T.NETWORK_ACCESS	Unauthorized access may be achieved to service on a protected network from outside that network.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.DATA_INTEGRITY	A malicious party attempts to change the data being sent - resulting

	in loss of integrity.
T.REPLAY_ATTACK	If malicious external IT entities are able to gain access to the network, they may have the ability to capture information traversing throughout the network and send them on to the internal server.

3.2 Assumptions

The assumptions made in identification of the threats and security requirements for the TOE are described in the table below:

Assumption	Description	Operational Environment
A.CONNECTIONS	It assumes that the TOE is connected to the distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.	OE.CONNECTIONS
A.PHYSICAL_ PROTECTION	The network device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device.	OE.PHYSICAL
A.LIMITED_ FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example the device should not provide computing platform for general purpose applications (unrelated to networking functionality).	OE.NO_GENERAL_ PURPOSE
A.NO_THRU_TRAFFIC_ PROTECTION	A standard/generic network device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the network device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the network device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs for particular types of network devices (e.g, firewall).	OE.NO_THRU_TRAFFIC _PROTECTION
A.TRUSTED_ ADMINISTRATOR	The Security Administrator(s) for the network device are assumed to be trusted and to act in the best interest of security for the organization. This includes being appropriately trained, following	OE.TRUSTED_ADMIN

	policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/ credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The network device is not expected to be capable of defending against a malicious administrator that actively works to bypass or compromise the security of the device.	
A.REGULAR_UPDATES	The network device firmware and software is assumed to be updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.	OE.UPDATES
A.ADMIN_ CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the network device are protected by the platform on which they reside.	OE.ADMIN_ CREDENTIALS_ SECURE
A.COMPONENTS_ RUNNING	For distributed TOEs it is assumed that the availability of all TOE components is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. It is also assumed that in addition to the availability of all components it is also checked as appropriate that the audit functionality is running properly on all TOE components.	OE.COMPONENTS_ RUNNING
A.RESIDUAL_ INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.	OE.RESIDUAL_ INFORMATION

3.3 Organizational Security Policy

Organizational Security Policy imposed by an organization to address its security needs is described in the table below:

Security Policy	Description
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

4. SECURITY OBJECTIVES

4.1 Security Objectives for the TOE

Security objectives for the TOE are described in the table below:

Security Objectives for the TOE	Description
O.CRYPTOGRAPHIC_	The TOE will provide means to encrypt and decrypt data as a means to

FUNCTIONS	maintain confidentiality and allow for detection and modification of TSF data that is transmitted outside of the TOE.
O.AUTHENTICATION	The TOE will provide means to authenticate the user to ensure they are communicating with an authorized IT entity.
O.FAIL_SECURE	Upon a self-test failure, the TOE will shut down to ensure data cannot be passed while not adhering to the security policies configured by the administrator.
O.SYSTEM_MONITORING	The TOE will provide means to audit events specific to WLAN functionality and security.
O.TOE_ADMINISTRATION	The TOE will provide the functions necessary to address failed authentication attempts by a remote administrator.

4.2 Security Objectives for the Operational Environment

Security objectives for the Operational Environment are described in the table below:

Security Objectives for the OE	Description
OE.CONNECTIONS	TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic flowing among attached networks.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.NO_THRU_TRAFFIC_ PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.
OE.UPDATES	The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_ SECURE	The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.COMPONENTS_RUNNING	For distributed TOEs the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

5. SECURITY FUNCTIONAL REQUIREMENTS

5.1 Conventions

The CC defines the following operations on the Security Functional Requirements: Assignment, Refinement, Selection, Assignment within Selection, and Iteration. 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 or Selection within an Assignment: 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.

5.2 TOE Security Functional Requirements

Functional Class	Component	Component Definition
FAU: Security audit	FAU_GEN.1	Audit Data Generation
	FAU_GEN.2	User Identity Association
	FAU_STG_EXT.1	Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation (Refined)
	FCS_CKM.1(2)	Cryptographic Key Generation (Symmetric Keys for WPA2 Connections)
	FCS_CKM.2	Cryptographic Key Establishment (Refined)
	FCS_CKM.2(2)	Cryptographic Key Distribution (PMK)
	FCS_CKM.2(3)	Cryptographic Key Distribution (GTK)
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/(1) DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption) (Refinement)
	FCS_COP.1(2)/SigGen	Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1(3)/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1(4)/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_IPSEC_EXT.1	Internet Protocol Security (IPsec) Communications
	FCS_SSHC_EXT.1	SSH Client Protocol
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol

	FCS_TLSS_EXT.1	TLS Server Protocol
	FCS_RADSEC_EXT.1	RADIUS over TLS Protocol
	FCS_TLSC_EXT.2/RADSec	TLS Client Protocol with Authentication/RADSec
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Management (Refined)
	FIA_PMG_EXT.1	Password Management
	FIA_PSK_EXT.1	Pre-Shared Key Composition (Extended)
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.6	Re-authenticating
	FIA_UAU.7	Protected Authentication Feedback
	FIA_8021X_EXT.1	802.1X Port Access Entity (Authenticator) Authentication (Extended)
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.1/ITT	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate	Trusted Update - Management of Security Functions behaviour
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
	FMT_SMR.1.3	Security Management Roles
FPT: Protection of the TSF	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_FLS.1	Failure with preservation of secure state
	FPT_ITT.1	Basic Internal TSF Data Transfer Protection
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	Testing (Extended)
	FPT_TUD_EXT.1	Trusted update
FTA: TOE Access	FTA_TAB.1	Default TOE Access Banner
	FTA_SSL_EXT.1	TSF-Initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-Initiated Termination
	FTA_TSE.1	TOE Session Establishment
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF Trusted Channel (Refinement)
	FTP_TRP.1/Admin	Trusted Path (Refinement)
FCO: Communication	FCO_CPC_EXT.1	Component Registration Channel Definition

Table 1: TOE Security Functional Requirements

5.2.1 Security Audit (FAU)

5.2.1.1 FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions comprising:
- Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [no other actions];

d) Specifically defined auditable events listed in Table 2.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 2.

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.1(2)	Failure of key generation activity.	None
FCS_CKM.2	None	None
FCS_CKM.2(2)	Failure of key distribution activity.	None
FCS_CKM.2(3)	Failure of key distribution activity, including failures to wrapping GTK.	Identifier(s) for intended recipients of wrapped key.
FCS_CKM.4	None	None
FCS_COP.1/ DataEncryption	Failure of the WPA2 encryption or decryption	Cryptographic mode of operation, name/identifier of object being encrypted/ decrypted, non-TOE endpoint of connection.
FCS_COP.1/SigGen	None	None
FCS_COP.1/Hash	None	None
FCS_COP.1/KeyedHash	None	None
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure
FCS_RBG_EXT.1	None	None
FSC_IPSEC_EXT.1	Protocol failures. Establishment/ Termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection for both

		successes and failures.
	Failure to establish an IPsec SA.	Reason for failure
FCS_SSHC_EXT.1	Failure to establish an SSH session	Reason for failure
FCS_SSHS_EXT.1.1	Failure to establish an SSH session	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_RADSec_EXT.1	None	None
FCS_TLSC_EXT.2/RADSec	Failure to establish a TLS Session	Reason for failure
FIA_AFL.1	The reaching of the treshold for the unsuccessful authentication attemts and the actions taken (e.g., disabling of the account) and the subsequent, if appropriate, restoration to the normal state.	None
	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PSK_EXT.1	None	None
FIA_PMG_EXT.1	None	None
FIA_UIA_EXT.1	All use of 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.6	Attempts to re-authenticate	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None	None
FIA_8021X_EXT.1	Attempts to access the 802.1X controlled port prior to successful completion of the authentic exchange.	Provided client identity (MAC address)
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.1/ITT	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_SKP_EXT.1	None	None
FPT_APW_EXT.1	None	None

FPT_FLS.1	Failure of the TSF.	Indication that the TSF has failed with the type of failure that occurred.
FPT_ITT.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FPT_TST_EXT.1	Execution of this set of TSF self-tests. Detected integrity violations.	For integrity violations, the TSF code file that caused the integrity violation
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process.	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1 (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
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism.	Reason for denial, origin of establishment attempt.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failure of the trusted path functions.	None
FCO_CPC_EXT.1	Enabling communications between a pair of components. Disabling communications between a pair of components.	Identities of the endpoints pairs enabled or disabled.

Table 2: Security Functional Requirements and Auditable Events

5.2.1.2 FAU_GEN.2 User Identity Association

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

5.2.1.3 FAU_ STG_EXT.1 Protected Audit Event Storage

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

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

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

5.2.2 Cryptographic Support (FCS)

5.2.2.1 FCS_CKM.1 Cryptographic Key Generation (Refinement)

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

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;
- FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3

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

5.2.2.2 FCS_CKM.1(2) Cryptographic Key Generation (Symmetric Keys for WPA2 Connections)

FCS_CKM.1.1(2) Refinement: The TSF shall generate **symmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm [***PRF-384***] and [no other] and specified cryptographic key sizes ***[128 bits]*** **using a Random Bit Generator as specified in FCS_RBG_EXT.1** that meet the following: [IEEE 802.11ac-2013].

5.2.2.3 FCS_CKM.2 Cryptographic Key Establishment (Refinement)

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

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

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

5.2.2.4 FCS_CKM.2(2) Cryptographic Key Distribution (PMK)

FCS_CKM.2.1(2) Refinement: The TSF shall **receive the 802.11 Pairwise Master Key (PMK)** in accordance with a specified cryptographic key distribution method: ***[from 802.1X Authorization Server]*** that meets the following: ***[IEEE 802.11-2012]*** **and does not expose the cryptographic keys**.

5.2.2.5 FCS_CKM.2(3) Cryptographic Key Distribution (GTK)

FCS_CKM.2.1(3) Refinement: The TSF shall distribute **Group Temporal Key (GTK)** in accordance with a specified cryptographic key distribution method: [**AES Key Wrap in an EAPOL-Key frame**] that meets the following: [**NIST SP 800-38F, IEEE 802.11-2012 for the packet format and timing considerations] and does not expose the cryptographic keys.**

5.2.2.6(a) FCS_CKM.4 Cryptographic Key Destruction (AP)

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

- For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [a pseudo-random pattern using the TSF's RBG]];

- For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that

  [logically addresses the storage location of the key and performs a single overwrite consisting of [a pseudo-random pattern using the TSF's RBG]]];

  that meets the following: No Standard.

5.2.2.6(b) FCS_CKM.4 Cryptographic Key Destruction (SZ)

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 a part of the TSF that

  [logically addresses the storage location of the key and performs a single overwrite consisting of [zeroes]]];

  that meets the following: No Standard.

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

FCS_COP.1.1/DataEncryption Refinement: The TSF shall perform **encryption/decryption** in accordance with a specified cryptographic algorithm **AES used in CBC, CCMP** [GCM] mode and

cryptographic key sizes **128 bits** [256 bits] that meet the following: **AES as specified in ISO 18033-3, CCMP as defined in NIST SP 800-38C and IEEE 802.11-2012**, [CBC as specified in ISO 10116, GCM as specified in ISO 19772, CCMP as specified in IEEE 802.11ac-2013].

5.2.2.8 FCS_COP.1(2)/SigGen Cryptographic Operation (Signature Generation and Verification)

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

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

that meet the following: [

- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1 5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,
- For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4].

].

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

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

5.2.2.10 FCS_COP.1(4)/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

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

5.2.2.11 FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

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

5.2.2.12 FCS_RBG_EXT.1 Random Bit Generation

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

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*1*] *[*software-based source*],* [*2*] *[*hardware-based noise sources*]* with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and CSPs that it will generate.

5.2.2.13 FCS_IPSEC_EXT.1 Extended: Internet Protocol Security (IPsec) Communications

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

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

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

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512] and [no other algorithm].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23)], and [RFC 4868 for hash functions]].

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv2] protocol uses the cryptographic algorithms [AES-CBC-128, AES-CBC-256 (specified in RFC 3602)].
FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

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

  o [length of time, where the time values can be configured within *[1 minute to 239976]* hours

];

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

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

  o length of time, where the time values can be configured within *[1 minute to 239976]* hours;

];

].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*384*] bits.

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

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

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s) [14 (2048-bit MODP), 20 (384-bit Random ECP)].

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

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

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

5.2.2.14 FCS_SSHC_EXT.1 SSH Client Protocol

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [4251, 4252, 4253, 4254, 4344, 5656, 6668].

FCS_SSHC_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [no other method].

FCS_SSHC_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*262144*] bytes in an SSH transport connection are dropped.

FCS_SSHC_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: *[aes128-ctr, aes256-ctr, aes256-gcm@openssh.com].*

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

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

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

FCS_SSHC_EXT.1.8 The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

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

5.2.2.15 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [4251, 4252, 4253, 4254, 4344, 5656, 6668].

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

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*262144*] 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-ctr, AES256-ctr, aes256-gcm@openssh.com]*.

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2- nistp384, ecdsa-sha2-nistp521] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [hmac-sha1, hmac-sha2-256, hmac-sha2-512, implicit] 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-nistp384, ecdh-sha2-nistp521]* are the only allowed key exchange methods used for the SSH protocol.

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

5.2.2.16 FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The TSF shall implement [TLS 1.2 (RFC 5246)] 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_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches the reference identifier per RFC 6125 section 6.

FCS_TLSC_EXT.1.3 The TSF shall only establish a trusted channel if the server certificate is valid. If the server certificate is deemed invalid, then the TSF shall [not establish the connection].

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

5.2.2.17 FCS_TLSS_EXT.1 TLS Server Protocol

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

[

- 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
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_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 [TLS 1.1].

FCS_TLSS_EXT.1.3 The TSF shall [perform RSA key establishment with key size [3072 bits]; generate EC Diffie-Hellman parameters over NIST curves [secp384r1] and no other curves; generate Diffie-Hellman parameters of size [3072 bits].

5.2.2.18 FCS_RADSEC_EXT.1 RADIUS over TLS Protocol

FCS_RADSEC_EXT.1.1: The TSF shall implement RADIUS over TLS as specified in RFC 6614 to communicate securely with a RADIUS server.

FCS_RADSec_EXT.1.2: The TSF shall perform peer authentication using [X.509v3 certificates].

5.2.2.19 FCS_TLSC_EXT.2 TLS Client Protocol with Authentication/RADSec

FCS_TLSC_EXT.2.1 The TSF shall implement [TLS 1.2 (RFC 5246)] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:
[
- 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
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289

].

FCS_TLSC_EXT.2.2 The TSF shall verify that the presented identifier matches the reference identifier per RFC 6125 section 6.

FCS_TLSC_EXT.2.3 The TSF shall only establish a trusted channel if the server certificate is valid. If the server certificate is deemed invalid, then the TSF shall [not establish the connection].

FCS_TLSC_EXT.2.4 The TSF shall [present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1, secp521r1] and no other curves] in the Client Hello.

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

5.2.3 Identification and Authentication (FIA)

5.2.3.1.FIA_AFL.1(a) Authentication Failure Management (Refinement)

FIA_AFL.1.1 The TSF shall detect when **an Administrator configurable positive integer within [*1-100*] of successive** unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password.*

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been **met**, the TSF shall [prevent the offending remote administrator from successfully authenticating using a password until *[an Administrator defined time period has elapsed]*].

5.2.3.2 FIA_PMG_EXT.1 Password Management

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")", "_", "`", "~", "[", "]", "{", "}", "|", ";", ":", "'", "", ",", ".", "<", ">", "/", "?", "=", "+", "-", "\"];

b) Minimum password length shall be configurable to *[8]* and *[64]*.

5.2.3.3 FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for [IEEE 802.11 WPA2-PSK, IPsec, *[DPSK]*].

FIA_PSK_EXT.1.2 The TSF shall be able to accept text-based pre-shared keys that:
- are 22 characters and [*[up to 64 characters]*];
- are composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")").

FIA_PSK_EXT.1.3 The TSF shall be able to [accept] bit-based pre-shared keys.

5.2.3.4 FIA_UIA_EXT.1 User Identification and Authentication (Extended)

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

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

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

5.2.3.5 FIA_UAU_EXT.2(a) Password-based Authentication Mechanism

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

5.2.3.6 FIA_UAU.6 Re-authenticating

FIA_UAU.6.1 The TSF shall re-authenticate the administrative user under the conditions: when the user changes their password, [no other conditions].

5.2.3.7 FIA_UAU.7 Protected Authentication Feedback

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

5.2.3.8 FIA_8021X_EXT.1 Extended: 802.1X Port Access Entity (Authenticator) Authentication

FIA_8021X_EXT.1.1 The TSF shall conform to IEEE Standard 802.1X for a Port Access Entity (PAE) in the "Authenticator" role.

FIA_8021X_EXT.1.2 The TSF shall support communications to a RADIUS authentication server conforming to RFCs 2865 and 3579.

FIA_8021X_EXT.1.3 The TSF shall ensure that no access to its 802.1X controlled port is given to the wireless client prior to successful completion of this authentication exchange.

5.2.3.9 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.

- The certificate path must terminate with a trusted CA certificate.

- The TSF shall validate a certificate path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag is set to TRUE.

- The TSF shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 6960]

- The TSF shall validate the extendedKeyUsage field according to the following rules:

  o 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.

  o 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.

  o 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.

  o OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

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

5.2.3.10 FIA_X509_EXT.1/ITT X.509 Certificate Validation

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

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

- The TSF shall validate the revocation status of the certificate using [no revocation method]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - o 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.
  - o 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.
  - o OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field

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

5.2.3.11 FIA_X509_EXT.2 X.509 Certificate Authentication

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

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

5.2.3.12 FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.2.4 Security Management (FMT)

5.2.4.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

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

5.2.4.2 FMT_MTD.1/CoreData Management of TSF Data

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

5.2.4.3 FMT_SMF.1 Specification of Management Functions

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

[

- Ability to administer the TOE locally and remotely;
- Ability to configure the 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;
- [
  - o Ability to configure audit behavior;
  - o Ability to configure the lifetime for IPsec SAs;
  - o Ability to configure the interaction between TOE components;
  - o Ability to set the time which is used for time-stamps;
  - o Ability to configure the reference identifier for the peer;
  - o Ability to re-enable an Administrator account

  ]

5.2.4.4 FMT_SMR.1 Security Management Roles

FMT_SMR.1.3 The TSF shall ensure that the ability to remotely administer the TOE from a wireless client shall be disabled by default.

5.2.4.5 FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator.*

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.2.5 Protection of the TSF (FPT)

5.2.5.1 FPT_SKP_EXT.1 Protection of TSF Data (Extended)

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

5.2.5.2 FPT_APW_EXT.1 Protection of Administrator Passwords (Extended)

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

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

5.2.5.3 FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: failure of the power-on self-tests.

5.2.5.4 FPT_ITT.1 Basic Internal TSF Data Transfer Protection

FPT_ITT.1.1 Refinement: The TSF shall use [IPsec, SSH, TLS] **with security strength commensurate with all other trusted communications to** protect TSF data from disclosure and modification when it is transmitted between separate parts of the TOE.

5.2.5.5 FPT_TST_EXT.1 TSF Testing (Extended)

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests during initial start-up (on power on and [*at no other times or conditions*] to demonstrate the correct operation of the TSF: *[FIPS 140-2 power-up self-tests and integrity test for software and firmware].*

FPT_TST_EXT.1.2 The TSF shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the TSF-provided cryptographic service specified in FCS_COP.1(2) [FCS_COP.1/SigGen].

5.2.5.6 FPT_TUD_EXT.1 Trusted Update

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

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

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

5.2.5.7 FPT_STM.EXT.1 Reliable Time Stamps (Extended)

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, synchronise time with external time sources].

5.2.6 TOE Access (FTA)

5.2.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking (Extended)

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

- terminate the session]

after a Security Administrator-specified time period of inactivity.

5.2.6.2 FTA_SSL.3 TSF-initiated Termination (Refinement)

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

5.2.6.3 FTA_SSL.4 User-initiated Termination (Refinement)

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

5.2.6.4 FTA_TAB.1 Default TOE Access Banners (Refinement)

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

5.2.6.5 FTA_TSE.1 TOE Session Establishment

FTA_TSE.1.1 Refinement: The TSF shall be able to deny establishment **of a wireless client session** based on **TOE interface, time, day,** *[no other attributes].*

5.2.7 Trusted path/channels (FTP)

5.2.7.1 FTP_ITC.1 Inter-TSF trusted channel (Refinement)

FTP_ITC.1.1 Refinement: The TSF shall be **capable of using IEEE 802.11-2012 (WPA2), IEEE 802.1X, [**IPsec, RADIUS over TLS] **to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: WLAN clients, audit servers, 802.1X authentication servers, and *[NTP server]*** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

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

FTP_ITC.1.3 Refinement: The TSF shall initiate communication via the trusted channel for *[communications with:*

- *802.1X authentication server using Radsec*
- *audit server using IPsec*
- *NTP server using IPsec].*

5.2.7.2 FTP_TRP.1 Trusted Path (Refinement)

FTP_TRP.1.1 Refinement: The TSF shall **be capable of using** [**SSH, 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 Refinement The TSF shall permit remote **Administrators** to initiate communication via the trusted path.

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

5.2.8 Communication (FCO)

5.2.8.1 FCO_CPC_EXT.1 Component Registration Channel Definition

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

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

- A channel that meets the secure channel requirements in FPT_ITT.1], for at least TSF data.

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

6. SECURITY ASSURANCE REQUIREMENTS

6.1 SAR Requirements

The TOE assurance requirements for this ST are listed in the collaborative Protection Profile for Network Devices, v2.0 (NDcPPv2.0) + Errata 20180314, Version 2.0, 14 March 2018 and Extended Package for Wireless LAN Access System, v1.0, May 28, 2015 (pp_wlan_as_ep_v1.0) and correspond to the set of SARs listed in Common Criteria Version 3.1, Revision 4.

Assurance Class	Assurance Components
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 CM coverage (ALC_CMS.1)
Security Target (ASE)	Security Target (ASE_TSS.1)
Tests (ATE)	Independent testing – sample (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

6.2 SAR Rationale

This ST contains the assurance requirements from the collaborative Protection Profile for Network Devices, v2.0. Since the SARs were taken directly from the NDcPPv2.0 + Errata 20180314, Version 2.0, 14 March 2018 and pp_wlan_as_ep_v1.0 which are the approved PPs, the SAR rationale is presumed to be satisfied.

6.3 Assurance Measures

The table below lists the assurance components defined in the Protection Profiles.

Assurance Component	Component Name	Assurance Measures
ADV_FSP.1	Basic functional specification	Security Design for the Ruckus Solution
AGD_OPE.1	Operational user guidance	Operational User Guidance for the Ruckus Solution
AGD_PRE.1	Preparative procedures	Preparative procedures for the Ruckus Solution
ALC_CMC.1	Labeling of the TOE	CM plan for the Ruckus Solution
ALC_CMS.1	TOE CM coverage	CM plan for the Ruckus Solution
ATE_IND.1	Independent testing – sample	EVIT Security Testing of the Ruckus Solution
AVA_VAN.1	Vulnerability survey	Vulnerability survey of the Ruckus Solution
ASE_TSS.1	Security Target	Security Target

7. TOE SUMMARY SPECIFICATION

7.1 TOE Security Functions Specification

This section describes the security functions provided by the TOE to meet the security functional requirements specified for the TOE in section 5.2 TOE Security Functional Requirements (SFRs).

TOE SFRs	Component	How SFR is Met
FAU_GEN.1	All	The TOE generates audit data for the auditable events covered in below table. The table also reflects the module that audits the specific events. Audited events cover timestamps along with details like user identity and type of state or config change that was executed. All systems function behavior like cryptographic key genation/import/change/deletion is logged as well.
FAU_GEN.2	All	Any event that needs to be audited due to a user action is recorded with the identity of the user along with timestamps.
FAU_STG_EXT.1	All	In the distributed TOE, AP and vSZ-D authenticate to SZ and once and administrator allows SZ/vSZ to communicate to the AP and vSZ-D they form a distributed TOE. AP and vSZ-D buffer the logs and sends periodically to controller via SSH. All audit

		messages are propagated via SZ/vSZ to an audit server securely via IPSec in real-time, alternatively, the audit messages can be stored in the controller itself. When stored locally, and local storage is full, oldest audit records are overwritten. Only authorised administrators have access to the audit records. Controller can store 14 archives of application logs with each log size of upto 10 MB.
FCS_CKM.1 FCS_CKM.2	AllAP	For asymmentric keys, TOE supports RSA schemes using cryptographic key sizes of 2048-bit or greater that meet FIPS PUB 186-4, "Digital Signature Standard (DSS)". FFC schemes can use Diffie-Hellman group 14 with 2048-bit. For the TLS connection between SZ and web-client, use the DH-3072 or secp384r1 as the key establishment algorithm. As for the TLS connection between DP and AP, system uses the RSA-3072 as the key establishment algorithm.
FCS_CKM.1(2) FCS_CKM.2(2) FCS_CKM.2(3) FIA_8021X_EXT.1	AP	For all symmetric keys, the TOE supports PRF-384 with 128 keys sizes. In the distributed TOE deployment, AP acts as the authenticator and derives the 802.11 keys. With WPA2-enterprise connections via 802.1X, all communications between the AP and wireless client is encapsulated in EAP and all traffic between AP and the controller is secured via SSH. The controller to RADIUS server communication is secured via RADSEC. TOE supports standard WPA2-enterprise with 802.11i where 4 way EAP handshake is between Authenticator (AP) and the wireless client. AP sends the EAPOL identity to the wireless client, which responds with a radius access request. Upon successful authentication, radius access accept message is recieved by the client. A PMK is then generated by the RADIUS server for Authenticator and client after which a PTK is derived. Authenticator also generates GTK. If the 4-way handshake fails, client will not be allowed to access the network or connect to the AP. Upon successful 4-way handshake,

		Authenticator will allow for WLAN data to pass through the system onto contorller in a tunnel architecture or to intended destination in distributed architecture. The PTK(total 384 bits) is derived into three parts. The second part is KEK and used to encrypt GTK to be sent as 3rd message in WPA2 handshake. Third part is TK, which is actually used to encrypt/decrypt communication between both AP and Client. Ruckus implementation of PTK generation complies to following sections of IEEE 802.11-2012: Section 4.10.3.2 --> AKM operations with AS (WPA2 4-way handshake explaining PTK/GTK transfer) Section 11.6.1.3 --> Pairwise key hierarchy (PTK derivation using the PRF function) Ruckus implementation of GTK generation complies to following sections of IEEE 802.11-2012: Section 4.10.3.2 --> AKM operations with AS (WPA2 4-way handshake explaining PTK/GTK transfer) Section 11.6.1.4 --> Group key hierarchy (GTK derivation using the PRF function) Certification testing performed by the Wi-Fi Alliance demonstrates the TOE implements the IEEE 802.11-2012 standard correctly. Refer Wi-Fi Alliance certificates for compliance.
FCS_CKM.4	All	The TOE destroys all the critical security parameters including all private keys and shared secrect keys are specified in below table
FCS_COP.1/DataEncryption	All	TOE performs encryption, decryption with AES with below modes • AES as defined in ISO 18033-3 • CCMP as defined in IEEE 802.11-12 • CBC as defined in ISO 10116 • GCM as defined in ISO 19772 With 128 and 256 bits
FCS_COP.1/SigGen	All	All cryptographic signature services are done in accordance with RSA with key size 2048 bits or higher and ECDSA with key sizes 256

		bits or greater. Both RSA and ECDSA meet requirements specified in FIPS PUB 186-4.
FCS_COP.1/Hash	All	TOE supports SHA-1, SHA-256, SHA-384, SHA-512.
FCS_COP.1/KeyedHash	All	TOE supports keyed hash message authentication that meet ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2" with following algorithms - HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 and cryptographic key sizes of 160, 256, 384, 512 bits.
FCS_HTTPS_EXT.1	Controller	The TOE provides HTTPS (via TLS1.2), as specified in RFC 2818, to provide a secure interface for remote administrative functions, and to support secure exchange of user authentication parameters during login and configuration data .
FCS_RBG_EXT.1	All	The TOE implements a random bit generator (RBG) in accordance with ISO/IEC 18031:2011. RBG is seeded with both hardware and software sources at least 256 bits of entropy to the DRBG.
FCS_IPSEC_EXT.1	All	TOE supports IPSec communication between the TOE components – AP and vSZ-D and also to external components like SYSLOG, NTP via Controller. TOE uses the encapsulating security payload (ESP) to protect the payloads of the traffic. Only IKEv2 is supported in both IPSec tunnels. For IPSec between AP and vSZ-D, TOE uses AES-128 with SHA1 and MODP 2048, AES-256 with SHA384 and ECP384. As part of this negotiation, the TOE verifies that the negotiated phase 2 symmetric algorithm key strength is at most as large as the negotiated phase 1 key strength as configured on the TOE and peer via an explicit check. Security policy is based on the WLAN ie only specific WLAN traffic that the administrator chooses is protected via IPSec tunnel between AP and vSZ-D. All other traffic including WLANs that are not configured for IPSec is bypassed and transmitted locally from AP. All inbound and outbound traffic that is desined to the Wireless client on that SSID is protected via IPSec in transport mode.

		The default time value for Phase 1 SAs is 4 hours, but is configurable from 1 minute to 9999 days. The default time value for Phase 2 SAs is 1 hour, but it is configurable 1 minute to 9999 days. Tunnel between AP and vSZ-D supports X.509v3 based authentication. The IKE protocols implement DH Groups 14 (2048-bit MODP) and 20 (384-bit Random ECP). Administrator selects the DH group. For the peer authentication using X509 certificate, either ECDSA or RSA, the TOE validates one of the following identifiers: - The full qualified domain name (FQDN) in the subject alternative name (SAN) - Full distinguish name (DN). For IPSec between Controller and external component, TOE uses AES-128, AES-192, AES-256 with HMAC-SHA1, HMAC-SHA256, HMAC-SHA384, HMAC-SHA512 and ECP384. As part of this negotiation, the TOE verifies that the negotiated phase 2 symmetric algorithm key strength is at most as large as the negotiated phase 1 key strength as configured on the TOE and peer via an explicit check. The default time value for Phase 1 SAs is 4 hours, but is configurable from 1 minute to 9999 days. The default time value for Phase 2 SAs is 4 hours, but it is configurable 1 minute to 9999 days. Tunnel between Controller and external component supports X.509v3 and PSK based authentication. The IKE protocols implement DH group 20 (384-bit Random ECP). For the peer authentication using X509 certificate, either ECDSA or RSA, the TOE validates one of the following identifiers: - The full qualified domain name (FQDN) in the subject alternative name (SAN) - Full distinguish name (DN).

		Once the IPSec configuration is defined in Controller, it initiates the tunnel to the external component defined in the controller. All inbound and outbound traffic to/from NTP/Audit server is protected by the IPSec in tunnel mode. All other traffic is bypassed. When a packet destined for the Audit server or NTP server is processed by the TOE and it determines it requires IPsec, it uses active SA settings or creates new SAs for initial connections with the IPsec peer. TOE supports both bit based and text based keys for Pre-shared keys. A TOE administrator specifies the PSK in the controller configuration. PSK includes a combination of upper and lower-case letters, numbers, and supported special characters and must be 44-128 for Hex characters and 8-64 for ASCII.
FCS_SSHC_EXT.1	AP, vSZ-D	In the distributed TOE, AP and vSZ-D only are SSH clients which communicate to SSH Server which is the SmartZone controller via only public key auth (No password-based authentication) TOE supports public key based authentication and supports only the following - ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2- nistp384. Packets larger than 256K bytes are dropped. Following encryption algorithms are supported - AES-128-CTR, AES-256-CTR, aes256-gcm@openssh.com The following data integrity algorithms are supported - hmac-sha1, hmac-sha2-256, hmac-sha2-512 and implicit. The following key exchange algorithms are supported • diffie-hellman-group14-sha1 • ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521 When AP or vSZ-D joins the controller, it has the ability to transmit a rekey request to the SSH server. With the RekeyLimit configuration, the ssh session key will be re-

		negotiated by either one of the following conditions: - The maximum amount of data transmitted over the ssh connection exceeds 1GB. - The maximum amount of connection time over 1 hour.
FCS_SSHS_EXT.1	Controller, vSZ-D	TOE supports following public key algorithms - ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2- nistp384, ecdsa-sha2-nistp521 Packets larger than 256K bytes are dropped. The following data integrity algorithms are supported - hmac-sha1, hmac-sha2-256, hmac-sha2-512 and implicit. The following key exchange algorithms are supported • diffie-hellman-group14-sha1 • ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521 The following encryption algorithms are supported - AES128-ctr, AES256-ctr, aes256-gcm@openssh.com The ssh session key will be re-negotiated by either one of the follow two conditions is met: 1. The maximum amount of data transmitted over the ssh connection exceeds 1G bytes 2. The maximum amount of connection time over 1 hour.
FCS_TLSC_EXT.1	AP, vSZ-D	TOE supports TLS client that confrms only to TLS 1.2(RFC 5246). The following cipher suites are supported • *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268* • *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268* • *TLS_DHE_RSA_WITH_AES_128_CBC _SHA as defined in RFC 3268* • *TLS_DHE_RSA_WITH_AES_256_CBC*

		_SHA as defined in RFC 3268 • *TLS_ECDHE_RSA_WITH_AES_128_ CBC_SHA as defined in RFC 4492* • *TLS_ECDHE_RSA_WITH_AES_256_CBC _SHA as defined in RFC 4492* • *TLS_ECDHE_ECDSA_WITH_AES_128 _CBC_SHA as defined in RFC 4492* • *TLS_ECDHE_ECDSA_WITH_AES_256_ CBC_SHA as defined in RFC 4492* • *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246* • *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246* • *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246* • *TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246* • *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288* • *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288* •*TLS_ECDHE_ECDSA_WITH_AES_128_CB C_SHA256 as defined in RFC 5289* •*TLS_ECDHE_ECDSA_WITH_AES_256_CB C_SHA384 as defined in RFC 5289* •*TLS_ECDHE_ECDSA_WITH_AES_128_G CM_SHA256 as defined in RFC 5289* •*TLS_ECDHE_ECDSA_WITH_AES_256_G CM_SHA384 as defined in RFC 5289* • *TLS_ECDHE_RSA_WITH_AES_128_ GCM_SHA256 as defined in RFC 5289* • *TLS_ECDHE_RSA_WITH_AES_256_ GCM_SHA384 as defined in RFC 5289* • *TLS_ECDHE_RSA_WITH_AES_128_ CBC_SHA256 as defined in RFC 5289* • *TLS_ECDHE_RSA_WITH_AES_256_ CBC_SHA384 as defined in RFC 5289].* TLS session is established only when the server certificate is valid. The following elliptic curves are supported in the client hello - secp256r1, secp384r1, secp521r1

		The reference identifier (FQDN) is supplied to the client (AP and vSZ-D) via administrator configuration or via DHCP option 43. Once the reference identifier is received, the client tries to intiate the TLS connection to the server. IP address, wild cards and certificate pinning are not supported. The distributed TOE also requires administrator action to allow data transfer after initial session establishment.
FCS_TLSS_EXT.1	Controller	TOE offers TLS sever via the controller and supports TLS 1.2 with following cipher suites ● 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 ● TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 ●TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 ●TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246 ●TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 ●TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288 ●TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246 ●TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 ●TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288]. TOE will deny connections from clients that request SSL2.0, SSL3.0, TLS1.0 and TLS1.1 TOE performs RSA key establishment with key size 3072, ECDH over secp384r1, and

		uses DH parameters of size 3072 bits.
FCS_RADSec_EXT.1 FCS_TLSC_EXT.2/RADSec	Controller	TOE supports RADIUS over TLS as specified in RFC 6614. TOE performs the function of a RADSEC client and conducts mutual authentication via X.509v3 certificates. TOE supports only TLS 1.2 and rejects all other TLS and SSL versions. The reference identifier is configured by the administrator of the TOE. Controller initiates the RADSEC connection and verifies the presented identifier (FQDN) of the server in the certificate that server presents to it. If the certificate is not valid, the connection is dropped. IP address, wild card is not supported. OCSP is supported to verify the validity of the certificate. The TLS implementation will support the following ciphersuites: ● 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 ●TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 ●TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 TOE will present the supported elliptic curve extensions with following secp256r1, secp384r1, secp521r1.
FIA_AFL.1	Controller, vSZ-D	Controller provides remote administration of the system via secure communication channel (WebGUI via HTTPS and CLI via SSH). vSZ-D provides remote administration of the system via secure communication channel CLI via SSH. Unsuccessful attempts beyond an administrator configured limit will result in access being denied until an administrator

		configured time duration has elapsed. Local system administrator can login into the TOE while remote administrators are blocked.
FIA_PMG_EXT.1	Controller	TOE provides the ability to set administrator passwords which can be composed of any combination of upper and lower case letters, numbers, and the following special characters [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [”_”, ”`”, ”~”, ”[”, ”]”, ”{”, ”}”, ”|”, ”;”, ”:”, ”’”, ”””, ”,”, ”.”, ”<”, ”>”, ”/”, ”?”, ”=”, ”+”, ”-”, ”\”]]. The minimum lenght can be configured by the administrator and can be between 8-64 characters.
FIA_PSK_EXT.1	Controller	TOE supports use of preshared keys for IPSec, 802.11 WPA2-PSK SSID (single PSK and dynamic PSK). TOE accepts text and bit based keys. For text based keys, 22-64 characters are allowed with any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”) TOE does not generate bit based keys. Text based/ASCII WPA2 passwords are converted to 256 bit based on HMAC SHA1 and IPSec passwords are converted using PRF.
FIA_UIA_EXT.1	Controller	TOE requires all administrators of the system to login via credentials (username & password) before granting access to the system. When the login screen is presented, TOE presents a customized banner. Administrators can manage the TOE via the controller. While SSH is available on vSZ-D, it can also be managed via Controller. AP cannot be managed directly.

FIA_UAU_EXT.2	Controller & vSZ-D	TOE requires all administrators of the system to login via credentials (username & password) before granting access to the system. CLI via SSH, WebUI via HTTPS methods are allowed on the controller for a remote administrator and local console access is allowed to the controller via authentication as well. Once an administrator tries to access the system, a login screen is presented where the administrator inputs the credentials. Access to the system is denied if the authentication does not succeed or is not performed.
FIA_UAU.6	Controller	Upon change of an administrator password by the administrator themself, TOE will re-authenticate the administrator.
FIA_UAU.7	Controller, vSZ-D	When an administrator enters the password, entered password is obscured.
FIA_X509_EXT.1/ITT	All	TOE provides X.509 v3 based authentication between the components in accordance to RFC 5280. AP and vSZ-D authenticate the controller and perform the certificate validation when they try to join it. AP and vSZ-D will verify the extendedKeyUsage field of the controller certificate.
FIA_X509_EXT.1/Rev FIA_X509_EXT.2	Controller	TOE provides X.509v3 based authentication for communications with external components via IPSec for NTP/audit server and RADSEC in accordance to RFC 5280. Controller performs the certificate validation when it tries to join an external component. Controller verifies the extendedKeyUsage field. TOE supports OCSP to verify the validity of the certificate presented by the external component. Administrators can upload the CA chain and map which certificates to use for different external connections. TOE rejects the connection if the certificate validation fails.
FIA_X509_EXT.3	Controller	TOE supports the certificate request message generation as specified by RFC 2986 and provides the following information – public key, Common Name, Organization, Organizational Unit and Country.
FMT_MOF.1/ManualUpdate	All	TOE provides centralized management via the controller. AP cannot be managed remotely. Only authorized administrators after successful authentication will gain access to the system via secure channel. Upon access, administrators can update the TOE. TOE supports Role Based Administrator

		Access which can limit privileges to some administrators.
FMT_MTD.1/CoreData	All	Only authenticated administrators are allowed access to the SZ to perform administrative functions via WebUI (HTTPS), CLI (SSH) and Console vSZ-D authenticated administrators are allowed access to perform administrative functions via an SSH secure channel.
FMT_SMF.1	Controller	TOE provides the ability to perform administrative functions. Administrators can control the TOE components via the Controller. Only authenticated administrators are allowed access to the TOE to perform administrative functions via WebUI (HTTPS), CLI (SSH) and Console. All security functions are available through all interfaces. Below administrative functions are supported • Ability to administer the TOE locally and remotely; • 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 configure audit behavior; • Ability to configure the lifetime for IPsec SAs; • Ability to configure the interaction between TOE components; • Ability to set the time which is used for time-stamps; • Ability to configure the reference identifier for the peer; • Ability to re-enable an Administrator account • Configure approval/denial of AP and vSZ-D to communicate/join the controller
FMT_SMR.2	Controller	Only authenticated administrators are allowed access to the TOE to perform administrative functions via WebUI (HTTPS), CLI (SSH)

		and Console. TOE supports Role Based Administrator Access which can limit privileges to administrators.
FMT_SMR.1.3	Controller	TOE prevents the ability to administer it from a wireless client.
FPT_SKP_EXT.1 FPT_APW_EXT.1	All	TOE provides no mechanism to read or disclose private keys, preshared keys, passwords or symmentric keys stored in TOE. Keys are stored in internal storage. Passwords are stored in non-plaintext form and are obscured. Passwords are encrypted in storage.
FPT_FLS.1	All	Upon critical failure, the TOE component either goes into a quarantine state and can be recovered only by a security administrator or it reboots and disables access to its interfaces.
FPT_ITT.1	All	TOE has 3 components, a controller, AP and vSZ-D. All communication between controller and remaining components are secured via SSH and TLS. Communication between AP and vSZ-D is secured via IPSec.
FPT_STM_EXT.1	All	The TOE provides reliable timestamp via configuration by system administrator or syncronising time via NTP securely. The timestamps are used in audit records, certificate validation, session monitoring activity and Wireless client access management.
FPT_TST_EXT.1	All	When the TOE boots up, each component of the TOE performs POST (Power On Self-Test) for all cryptographic modules. The tests include AES Known Answer Test SHA Known Answer Test HMAC Known Answer Test CCM Known Answer Test GCM Known Answer Test DRBG Known Answer Test Software Integrity test (RSA with SHA384) CMAC Known Answer Test RSA Known Answer Test ECDSA Pairwise Consistency Test ECC CDH shared secret computation When any one of the known answer tests fails, the system will enter the quarantine state.

FPT_TUD_EXT.1	All	TOE provides the ability to query a currently running firmware version via WebUI by going to viewing Administration-upgrade after logging in and CLI via executing the command show version after logging in. TOE also provides ability to get a new software version by going to Administration-Upgrade-Upload firmware and subsequently apply the new software version by clicking upgrade on WebUI. Controller in turn will upgrade the APs and vSZ-D once it has applied the software on itself securely via SSH. All components of the TOE are updated via controller. All controller, AP and vSZ-D updates are verified via digital signatures. Software installation will fail if the signature verification fails. All software updates are hosted on Ruckus support portal and can be downloaded securely via HTTPS after authenticating to the server.
FTA_SSL_EXT.1 FTA_SSL.3 FTA_SSL.4	Controller, vSZ-D	TOE security administrator can configure the session inactivity timeout which terminates the sessions of administrators once the session inactivity timeout is reached for both local and remote interactive sessions. Administrator also has the ability to terminate its own session by clicking logout on WebUI and typing exit on CLI.
FTA_TAB.1	Controller	TOE provide the ability for administrators to configure a login banner which will be visible to users when they try to access the controller prior to log in via WebUI or CLI.
FTA_TSE.1	AP	TOE provides ability to deny establishment of a wireless client session based on TOE interface, time, day.
FTP_ITC.1	Controller, AP	The controller uses secure channel to communicate to external components. • Audit server via IPSec • NTP server via IPSec • RADIUS via RADSEC • Wireless client via WPA2
FTP_TRP.1	Controller, vSZ-D	SZ supports all remote administrative functions via secure channels - WebUI (HTTPS) and CLI (SSH) vSZ-D supports all remote administrative function via an SSH secure channel

FCO_CPC_EXT.1	All	When an AP and vSZ-D successfully discovers a controller via manual configuration or optionally via DHCP options, it tries to connect via TLS and SSH. When the AP and vSZ-D successfully connect to controller, no data transfer is allowed between the controller and AP/vSZ-D. An administrator has to manually approve the AP/vSZ-D which enables data transfer between the controller and AP/vSZ-D. An administrator also has the ability to manually revoke the connection between controller and AP/vSZ-D. This process meets FPT_ITT.1.

Auditable Events

Requirement	Implementing Component	Auditable Event	Rationale
FAU_GEN.1	Controller	• Administrative login and logout. • Changes to TSF data related to configuration changes and corresponding user identity • Generating/import of, changing, or deleting of cryptographic keys • Resetting passwords	TOE generates audit logs for all the auditable events via the controller.
FCS_CKM.1(2)	Controller, AP	Failure of key generation activity.	TOE generates audit event when a key generation fails.
FCS_CKM.2(2)	Controller, AP	Failure of key distribution activity.	TOE generates audit event when a key distribution fails.
FCS_CKM.2(3)	Controller, AP	Failure of key distribution activity, including failures to wrapping GTK.	TOE generates audit event when a key distribution and GTK wrap fails.
FCS_COP.1/DataEncryption	Controller, AP	Failure of WPA2 encryption or decryption	TOE generates audit events when WPA2 encryption or decryption fails.
FCS_HTTPS_EXT.1	Controller	Failure to establish HTTPS session	TOE generates an audit record when failure to establish a HTTPS session session occurs.
FCS_IPSEC_EXT.1	Controller, AP, vSZ-D	• Protocol failures. • Establishment/ Termination of an IPsec SA.	TOE generates audit logs for IPSec protcol failures with reason for failure/establishement/termination and details of end

			points of the tunnel.
FCS_SSHS_EXT.1	Controller, vSZ-D	Failure to establishment of session	TOE generates an audit record when failure to establish a SSH session session occurs.
FCS_SSHC_EXT.1	AP, vSZ-D	Failure to establishment of session	TOE generates an audit record when failure to establish a SSH session session occurs.
FCS_TLSC_EXT.1	AP, vSZ-D	Failure to establishment of session	TOE generates an audit record when failure to establish a TLS session session occurs.
FCS_TLSS_EXT.1	Controller, vSZ-D	Failure to establishment of session	TOE generates an audit record when failure to establish a TLS session session occurs.
FCS_TLSC_EXT.2/RADSec	Controller	Failure to establishment of session	TOE generates an audit record when failure to establish a TLS session session occurs.
FIA_AFL.1	Controller, vSZ-D	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken (e.g., disabling of the account) and the subsequent, if appropriate, restoration to the normal state.	TOE generates audit records when threshold for failed authentication is reached. Origin of the attempt is captured in the audit record as well.
FIA_UIA_EXT.1 FIA_UAU_EXT.2	Controller, vSZ-D	All use of identification and authentication mechanism.	TOE generates audit records when any attempt login and re-authenticate occurs. Origin of the attempt is captured in the audit record as well.
FIA_UAU.6	Controller, vSZ-D	Attempts to re-authenticate	TOE generates audit records when any attempt to re-authenticate occurs. Origin of the attempt is captured in the audit record as well.
FIA_8021X_EXT.1	Controller, AP	Attempts to access the 802.1X controlled port prior to successful completion of the authentication exchange.	TOE generates audit records when any attempt to authenticate 802.1X port.
FIA_X509_EXT.1/ Rev	Controller	Failure to validate a certificate	TOE generates an audit record when validation of a certificate fails.
FIA_X509_EXT.1/ITT	All	Failure to validate a certificate	TOE generates an audit record when validation of a certificate fails.
FMT_MOF.1 / ManualUpdate	All	Any attempt to initiate a manual update	TOE generates audit records when any attempt to initiate a manual update occurs.

FMT_MTD.1/CoreData	All	All management activities of TSF data.	TOE generates audit records when any management activites are performed. Identity of administrator is captured in the audit record as well.
FPT_FLS.1	All	Failure of the TSF.	TOE generates a local access console message when POST is performed. If a failure occurs the systems goes into a quarantine state which can be viewed on console. When a image integrity fails, audit records are generated.
FPT_ITT.1	All	Initiation/Termination/Failure of Trusted channel	Identification of the initiator and target of failed trusted channels establishment attempt.
FPT_TST_EXT.1	All	Execution of this set of TSF self-tests. Detected integrity violations.	TOE generates a local access console message when POST is performed. If a failure occurs the systems goes into a quarantine state which can be viewed on console. When an image integrity fails, audit records are generated.
FPT_TUD_EXT.1	All	Initiation of update; result of the update attempt (success or failure)	TOE generates audit records when any attempt to update the system is performed.
FPT_STM_EXT.1	All	Discontinuous changes to time - either Administrator actuated or changed via an automated process.	TOE generates audit records when manual or NTP based discontinuous changes to time occurs. The old and new values for the time and origin of the attempt to change time for success and failure are logged
FTA_SSL_EXT.1	Controller, vSZ-D	The termination of a local session by the session locking mechanism.	TOE generates audit records when the local session is being terminated.
FTA_SSL.3 FTA_SSL.4	Controller, vSZ-D	The termination of a remote session by the session locking mechanism or termination of an interactive session.	TOE generates audit records when a remote session is terminated.
FTA_TSE.1	AP	Denial of a session establishment due to the session establishment mechanism.	TOE generates audit records when session establishment is denied.

FTP_ITC.1	Controller, AP	Initiation, termination and failure of the trusted channel functions.	TOE generates audit records when any attempt to intiate, terminate or failure to establish a trusted channel occurs.
FTP_TRP.1/Admin	Controller, vSZ-D	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	TOE generates audit records when any attempt to initiate, terminate or failure to establish a trusted path occurs.
FCO_CPC_EXT.1	All	Enabling communications between a pair of components. Disabling communications between a pair of components.	TOE generates audit records when communications between a pair of components are enabled or disabled.

Key Zeroization

Zeroize commands:
"fips enable/disable" command on Controller;
"fips zeroization" command on vSZ-D;
“zeroize-all csp” command on AP (via controller)

Name	Description	Type	Storage	Zeroize
TLS Server RSA Private Key	RSA key used to sign server certificate	RSA-3072 Private Key	RAM, Flash	Zeroize command
TLS Server RSA Public Key	RSA public key signed as a server certificate	RSA-3072 Public Key	RAM, Flash	Zeroize command
TLS Client RSA Public Key	RSA public key signed as a client certificate	2048 – n bits; size dependent on cert. loaded into the module	RAM	Zeroize command
TLS Client RSA Private Key	RSA private key	2048 – n bits; size dependent on cert. loaded into the module	RAM	Zeroize command
TLS Pre-Master Secret	ECDH or DH secret value used to establish the TLS Master Secret	TLS key precursor	RAM	Session termination
TLS Master Secret	Secret value used to establish the TLS	TLS key precursor	RAM	Session termination

	Encryption Keys and TLS Authentication Keys			
TLS DH/ ECDH Host Private Key	Ephemeral DH or ECDH private key used to establish the TLS Pre-Master Secret	DHE ECDHE	RAM	Zeroize commands, power cycle
TLS DH/ ECDH Client Private Key	Ephemeral DH or ECDH private key used to establish the TLS Pre-Master Secret	DHE ECDHE	RAM	Zeroize commands, power cycle
TLS DH/ ECDH Host Public Key	Ephemeral DH or ECDH public key sent to the TLS client to establish the TLS Pre-Master Secret	DHE ECDHE	RAM	Zeroize command
TLS DH/ ECDH Client Public Key	Ephemeral DH or ECDH public key used to establish the TLS Pre-Master Secret	DHE ECDHE	RAM	Zeroize command
TLS Encryption Keys	AES keys used to encrypt TLS session data	AES-CBC-128 or AES-CBC-256 or AES-GCM-128 or AES-GCM-256	RAM	Session termination
TLS Authentication Keys	HMAC keys used to authentication TLS session data	HMAC-SHA1 HMAC-SHA256 or HMAC-SHA384	RAM	Session termination
Random Number Generation	Entropy Input for the SP800-90A CTR_DRBG	DRBG Seed material	RAM	Power cycle and Zeroize commands
DRBG Internal States	Internal State of SP800-90A CTR_DRBG	SP800-90A DRBG State	RAM	Power cycle and Zeroize commands
User Password	Password used to authenticate the User (at least 8 characters)	Authentication data	AES encrypted and encoded in Flash, RAM	Zeroize commands
Enable Password	Password used by the Crypto Officer to enable the CLI(at least 8 characters)	Authentication data	SHA512 hashed in Flash	Zeroize commands
Crypto Officer Password	Password used to authenticate the Crypto Officer (at least 8 characters)	Authentication data	AES encrypted and encoded in Flash	Zeroize commands
SSHv2 Host RSA/ECDSA Private Key	Used when host algorithm “ssh-rsa” (RSA) or “ecdsa-sha2” (ECDSA)	RSA-3072 (Controller and vSZ-D) or ECDSA P-384 Private Key (controller & AP)	RAM, Flash	Zeroize commands
SSHv2 Client RSA/ECDSA Private Key	Used when using RSA or ECDSA	RSA or ECDSA Private Key	RAM, Flash	Zeroize commands

SSHv2 Host RSA/ECDSA Public Key	Used to authenticate SSHv2 server to client when host algorithm "ssh-rsa" (RSA) or "ecdsa-sha2" (ECDSA)	RSA or ECDSA Private Key	RAM, Flash	Zeroize commands
SSHv2 Client RSA/ ECDSA Public Key	Used to authenticate client using RSA or ECDSA	RSA-2048 – n bits or ECDSA P-256 - n;	RAM, Flash	Zeroize commands
SSHv2 DH/ ECDH Private Key	DH or ECDH private key used to derive SSH Session and Authentication Keys	DHE: 2048, ECDHE: P-256/384/521	RAM	Session termination
SSHv2 Host DH/ ECDH Public Keys	Key exchange keys	DHE: 2048, ECDHE: P-256/384/521	RAM	Session termination
SSHv2 Client DH/ ECDH Public Keys	Key exchange keys	DHE: 2048, ECDHE: P-256/384/521	RAM	Session termination
SSHv2 Session Keys	AES encryption key used to secure SSHv2	AES-128-CTR or AES-256-CTR or AES-GCM@openssh.com Key	RAM	Session termination
SSHv2 Authentication Key	Session authentication key used to authenticate and provide integrity of SSHv2 session	HMAC-SHA-1 or HMAC-SHA-256 or HMAC-SHA-512	RAM	Session termination
SSHv2 KDF Internal State	Used to generate Host encryption and authentication key	KDF	RAM	Session termination and Zeroize commands
IKEv2/ IPsec Encryption Key	AES keys used to encrypt IKE/ IPsec session data	AES-CBC-128 or AES-CBC-192 (controller & AP) or AES-CBC-256	RAM	Session termination
IKEv2/ IPsec Authentication Keys	Session authentication key used to authenticate and provide integrity of IKE/ IPsec session	HMAC-SHA-1 or HMAC-SHA-256 or HMAC-SHA-384 or HMAC-SHA-512 (controller)	RAM	Session termination
IKEv2/ IPsec DH/ ECDH Private Key	Used to establish the secret keying material for IKE and IPsec	Group-20 (P-384) Group-14 (2048)	RAM	Session termination
IKEv2/ IPsec DH/ECDH Public Key	Used to establish the secret keying material for IKE and IPsec	Group-20 (P-384) Group-14 (2048)	RAM	Session termination
IKEv2/ IPsec Pre-Shared Key	Authenticate the peers to each other	Pre-Shared Key	Flash, RAM	Zeroize commands

IKEv2/ IPsec RSA/ ECDSA Private Key	RSA or ECDSA private key used during the IKE/ IPsec handshake to sign the host certificate	RSA-3072 or ECDSA P-384 private key	RAM	Zeroize commands
IKEv2/ IPsec RSA/ ECDSA Public Key	RSA or ECDSA public key signed as a host certificate	RSA-3072 or ECDSA P-384 public key	RAM	Zeroize commands
Firmware Upgrade Key	RSA key used to sign and verified the integrity of firmware.	RSA-4096 Public Key	Temporary file during firmware upgrade	Zeroized after verification of image
NTP Key	Authenticate with NTP server	Authenticate with sha-1 key Type	RAM, Flash.	Zeroize commands
RADIUS Secret	Authenticate with external radius server	Characters	RAM	Session termination and Zeroize commands
SSHv2 DH Shared Secret Key	2048 bits	DH Shared Secret	RAM	Zeroize command

CAVP Certificates

Smart Zone CAVP Certificates	
AES	5097
Triple-DES	2624
SHA	4145
HMAC	3399
DRBG	1903
ECDSA	1322
RSA	2759
KDF	1647, 1778
KAS-ECC (CVL), KDF, RSA	C706
AES, HMAC, SHA	C707
virtual SmartZone / virtual SmartZone – Data plane CAVP Certificates	
AES	5098
Triple-DES	2625
SHA	4146
HMAC	3400
DRBG	1904
ECDSA	1323
RSA	2760
KDF	1648, 1790
KAS-ECC (CVL), KDF, RSA	C706
AES, HMAC, SHA	C707
Access Points CAVP Certificates	
AES, HMAC, SHA	C708

AES, DRBG, ECDSA, HMAC, KAS-ECC (CVL), KDF, RSA, SHA, TDES	C710
KBKDF	199
AES	5312