FIPS 140-2 Level 2 Security Policy

for

Mojo Access Point

November 2018

339 N Bernardo Avenue, Suite 200
Mountain View, CA 94043

www.mojonetworks.com

© 2018 Mojo Networks, Inc.
This is non-proprietary document. It may be freely reproduced and distributed in whole without any modification.

Table Of Contents

1. Introduction

This document describes security policy for the Mojo Access Point cryptographic module from Mojo Networks, Inc. The security policy specifies the rules for operation of the module to meet Federal Information Processing Standard (FIPS) 140-2 Level 2 requirements.

FIPS 140-2, *Security Requirements for Cryptographic Modules*, describes the requirements for cryptographic modules. For more information about the FIPS 140-2 standard and the cryptographic module validation process see https://csrc.nist.gov/projects/cryptographic-module-validation-program.

2. Module Specification

Mojo Access Point is a multi-chip standalone hardware cryptographic module. It is referred to as a “Module” throughout this document.

- Hardware Versions C-120 and C-130
- Firmware Version 8.2.1

The Module firmware consists of Linux OS version 3.4.103, device drivers, wireless access application, sensor application, Crypto Cores, CLI application, and utilities. Crypto Core 2.0.16-1-01 and Crypto Core Supplement 2.2-1-00 in the firmware perform all cryptographic functions of the Module, other than inline encryption/decryption of Wireless Client traffic. The inline encryption/decryption is performed by QCA9994 System on Chip (SoC) integrated-circuit radio from Qualcomm. Access to the operating system operations is restricted.

The Module has been tested for FIPS compliance on two different operational environments, namely, C-120 appliance and C-130 appliance. Figure 1 shows physical embodiment of the appliances. They have identical enclosures. The appliances contain the following hardware components: System Processor, Wireless Subsystem, RAM, Flash Memory, Ethernet Network Interface which is PoE input (Power over Ethernet) capable, Console Interface, LED Interface, Power Inlet, USB Port, and Pinhole Reset Button. The wireless subsystem of C-120 appliance contains two radio modules, whereas that of C-130 contains three radio modules.

Figure 2 shows the cryptographic boundary for the Module and the paths of data, control and status information flow across the cryptographic boundary.

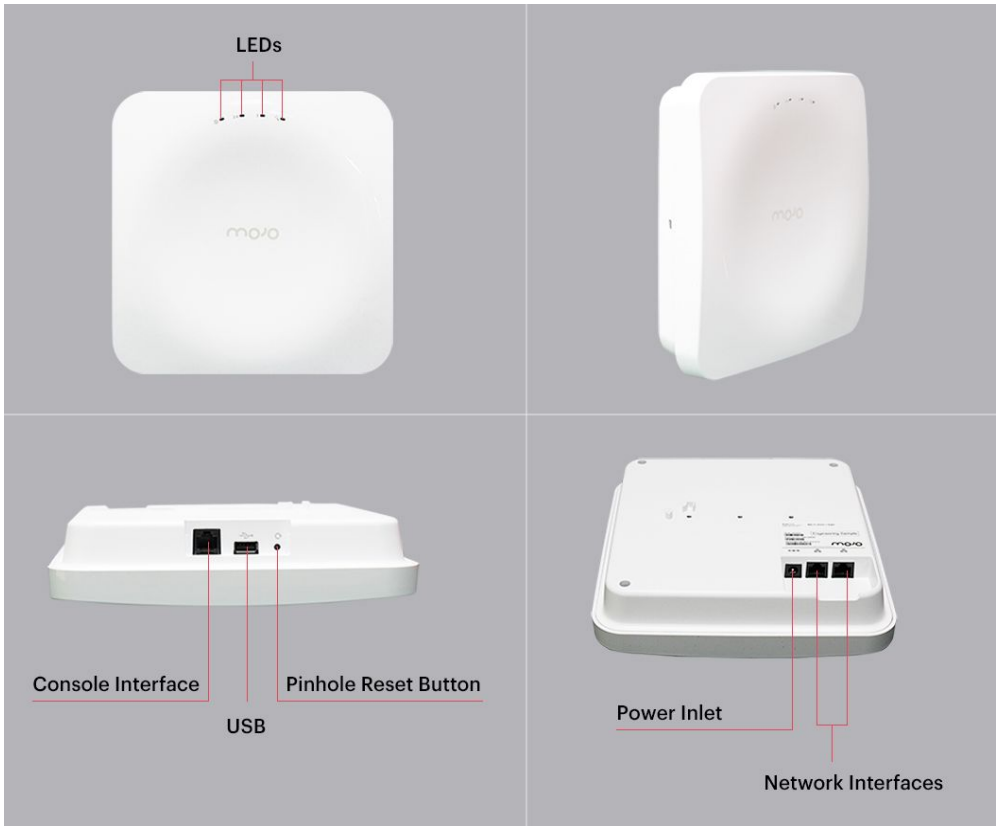

Figure 1: Appliance (C-120, C-130)

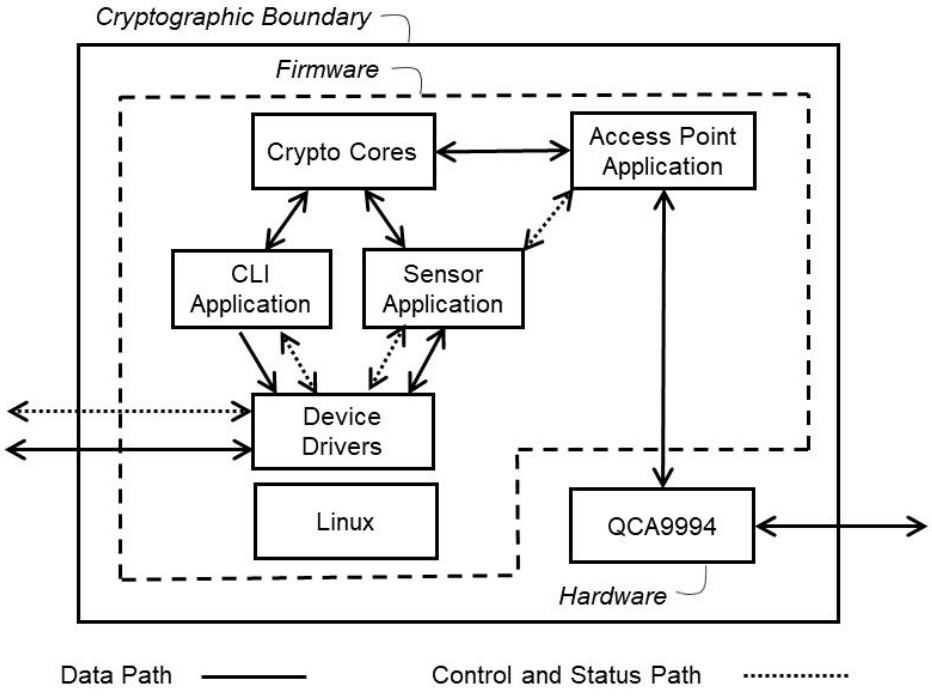

Figure 2: Cryptographic Boundary and Firmware

During operation, the Module transmits and receives wireless signals to provide network connectivity to Wireless Clients. It also receives wireless signals to sniff traffic in the wireless medium. The Module also sniffs traffic on the wired network to perform correlation between the wireless and the wire-side traffic. The Module transmits wireless signals to perform functions such as probing wireless devices and to disable undesirable wireless communications. The Module also transmits probing frames and frames to disable undesirable communications on the wired network. The Module reports information about Wireless Client connectivity, wireless sniffing and wire-side sniffing to the management server (Server) and is also able to accept commands from the Server for its operation. By virtue of these operations, the Module provides network connectivity to Wireless Clients and also ensures that wireless communication activity in its proximity complies with the specified security rules.

a) Ports and Interfaces

Table 1 shows physical ports on the Module and their mapping to FIPS 140-2 logical interfaces.

Physical Port	Logical Port	FIPS 140-2 Logical Interface
Network Interface	Access Point Application, Sensor Application	Data input, data output, control input, status output, power input (PoE).
Console Interface	CLI Application	Data output, control input, status output.
Power Inlet	Device Driver	DC power enters via power inlet. It can also enter via the Network Interface which is PoE (Power Over Ethernet) capable.
LEDs	Device Driver	Status output.
Antennas	Device Driver	Data input, data output.
Pin-hole Reset Button	Device Driver	Control input.
USB Port	NA	Disabled, NA

Table 1: Ports and Interfaces

b) Logical Interfaces

Logical interfaces on the Module are described in Table 2.

Logical Interface	Description
Data Input	Data enters the Module via the Antennas. This data corresponds to traffic received from Wireless Clients and from wireless sniffing.

	Data enters the Module via the Network Interface. This data represents traffic received from the wired network that is to be forwarded to Wireless Clients and from wire-side sniffing.
Data Output	Data exits the Module via the Antennas. This data represents traffic sent to Wireless Clients, wireless probing information, and transmissions used to disable certain wireless communications. Data exits the Module via the Network Interface. This data represents traffic received from Wireless Clients that is to be forwarded to the wired network and reports on connectivity of Wireless Clients, reports from wireless sniffing, reports from wire-side sniffing, wire-side probing information, and transmissions used to block certain communications associated with devices on the subnet(s) where the Network Interface is connected. Data also exits the Module via the Console Interface. This data represents reports on connectivity of Wireless Clients, reports from wireless sniffing, and reports from wire-side sniffing.
Control Input	Control input enters the Module via the Network Interface or the Console Interface. This input comprises of information such as network settings, Server discovery settings, status requests, password change, key change, mode change, configuration for wireless connectivity and sniffing, instructions to take actions on specific devices such as probing, disabling etc. Control input to reset the Module to factory settings can also enter via the Pinhole Reset Button.
Status Output	Status output exits the Module via the Network Interface or the Console Interface. This output comprises of information such as network settings, version number, operation logs, results of self tests, mode of operation, and status of other configuration parameters. Status output can also exit the Module via the LED Interface. This Status output consists of information such as whether the Module is powered up, status of the Module's connectivity to the Server, status of the Wireless Subsystem etc.

Table 2: Logical Interfaces

On a physical port, output (data and status) is logically separated from input (data and control) by way of direction of information flow. Data output is logically separated from status output by way of networking protocols and application level identifiers. Data input

is logically separated from control input by way of networking protocols and application level identifiers.

c) Modes of Operation

The Module is able to operate in FIPS and non-FIPS modes. Factory default setting is non-FIPS mode. The Module can be switched to FIPS mode using following steps:

- User logs into the Module over CLI (Command Line Interface). CLI login can be over console cable or SSH.
- User runs “set FIPS mode” command to invoke “change mode” service. After this command is run, the Module reboots. The User can confirm that the Module has booted in FIPS mode by running the “get FIPS mode” command.
- User sets custom password using “passwd” command.
- User manually enters the new key K using “set communication key” command and permits the module to connect to the Server.
- Crypto Officer applies WiFi profile with WPA2 security mode only.

Steps to switch the Module from FIPS mode to non-FIPS mode are:

- User logs into the Module over CLI, either via console cable or SSH.
- User runs “set FIPS mode” command to invoke “change mode” service. After this command is run, the Module reboots. The User can confirm that the Module has booted in non-FIPS mode by running the “get FIPS mode” command.
- User sets custom password using “passwd” command.
- User manually enters the new key K using “set communication key” command and permits the Module to connect to the Server.

The same security functions and services are provided in both modes. The same cryptographic algorithms are used in FIPS and non-FIPS modes, with the exception of upgrade bundle verification. In FIPS mode, RSA digital signature is used for upgrade bundle verification, while in non-FIPS mode HMAC-SHA-1 is used for the same purpose. Keys/CSPs are not shared between FIPS and non-FIPS modes.

d) Compliance with FIPS Requirements

The Module meets FIPS 140-2 security requirements as shown in Table 3.

Security Requirements Section	**Level**
Cryptographic Module Specification	2
Cryptographic Module Ports and Interfaces	2
Roles, Services, and Authentication	2
Finite State Model	2
Physical Security	2
Operational Environment	Not applicable
Cryptographic Key Management	2
Electromagnetic Interference/Electromagnetic Compatibility	2

Self Tests	2
Design Assurance	2
Mitigation of Other Attacks	Not applicable

Table 3: FIPS Security Requirements

3. Security Functions

The various security functions incorporated in the Module are described below.

a) Roles, Authentication, Services

The Module supports three roles: User, Crypto Officer, and Wireless Client. The User can access the Module over the Console Interface or the Network Interface. The Crypto Officer role is assumed by the Server when the Module connects to the Server. The Wireless Client can connect to the Module through the antenna.

a.1) User

The User accesses the Module over the Console Interface (using console cable) or the Network Interface. Information flowing over the console wire is plaintext. Access over the Network Interface is via SSH. User is authenticated to the Module using a password. Password should be at least 6 characters in length. The maximum password length is limited to 32 characters. This results in at least 308,915,776 combinations for the password (computed as 26 raised to the power 6). Thus, the possibility of correctly guessing the password is less than 1 in 1,000,000.

The login attempt rate of the User is limited to 1 attempt every 3 seconds. In other words, there can be at most 20 login attempts per minute. This, along with the number of possible combinations of the password, ensures that repeated attempts to use the authentication mechanism during a one-minute period have a probability of success less than one in 100,000.

Crypto Officer can change User password when the Module is properly connected to the Server.

When the Module is power-cycled, the User will have to re-authenticate. That is, the authentication state is forgotten after power cycle. When the Module is reset to factory default, the password for the User defaults to factory setting. The User password is also reset to factory default setting upon entering or exiting FIPS mode.

a.2) Crypto Officer

Server assumes the role of Crypto Officer when the Module connects to it. The Crypto Officer accesses the Module over the 128 bit AES-CBC encrypted tunnel. The Crypto Officer authenticates to the Module using the shared secret key (K) which is 128 bit in

length. Since there are 2^{128} combinations for the key, the possibility of correctly guessing the key is less than 1 in 1,000,000.

The authentication between the Module and the Server is mutual authentication using challenge/response procedure. Each side sends a random challenge to the other. The other side encrypts the challenge with key K using AES-CBC encryption, and sends the encrypted response to the issuing side. The issuing side decrypts the response to verify that the originally issued challenge is found therein. The authentication attempt rate is limited to 1 attempt every 2 minutes. This, along with the number of possible combinations of the key, ensures that repeated attempts to use the authentication mechanism during a one-minute period have a probability of success less than one in 100,000.

Key K is manually entered by User when turning the Module to FIPS mode as described above. Crypto Officer can change the key K when the Module is properly connected to the Server. The new key K is sent from Crypto Officer to the Module in a message that is protected by 128 bit AES-CBC encryption and HMAC-SHA1 authentication.

When the Module is power cycled, the Crypto Officer will have to re-authenticate. That is, the authentication state is forgotten after power cycle. When the Module is reset to factory default, the shared secret key K defaults to factory setting. It will also be reset to factory default (zeroized) upon entering and exiting the FIPS mode of operation for the Module.

a.3) Wireless Client

The Wireless Client authenticates to the Module as defined in the IEEE 802.11i standard, also called WPA2. The Pairwise Master Key (PMK), which is set equal to Pre Shared Key (PSK), is used for Wireless Client authentication. Each of PMK and PSK is 256 bits in length. Thus, the possibility of correctly guessing the PMK/PSK is less than one in 1,000,000 per login attempt.

The Wireless Client authenticates to the Module during the 4-way handshake that occurs at the beginning of the wireless connection. At least 2 of the 4 messages (i.e., 2 frames of at least 99 bytes) in the handshake are required for authentication failure to be detected. Also, the wireless link operates at 6 Mbps during the 4-way handshake phase. Hence, the maximum possible number of login attempts in a minute is 238,312 (which is a number less than less than 2^{18}), This, along with the number of possible combinations of the PMK/PSK, ensures that repeated attempts to use the authentication mechanism during a one-minute period have a probability of success less than one in 100,000.

Every time a Wireless Client attempts to connect to the Module, it needs to authenticate. When the Module is power cycled, the Wireless Client will have to re-authenticate. That is, the authentication state is forgotten after power cycle.

Table 4 summarizes strength of authentication for User, Crypto Officer, and Wireless Client roles.

Role	Auth. Credentials	Minimum Length	Maximum Length	Strength per Attempt	Strength per Minute
User	Password	6 characters	32 characters	Probability of success less than 1 in 308,915,776	Probability of success less than 1 in 15,445,788
Crypto Officer	Key (K)	128 bits	128 bits	Probability of success is 1 in 2^{128}	Probability of success is 1 in 2^{128}
Wireless Client	Key (PMK/PSK)	256 bits	256 bits	Probability of success less than 1 in 2^{256}	Probability of success less than 1 in 2^{238}

Table 4: Authentication

Table 5 shows services available to User, Crypto Officer, and Wireless Client roles.

Service	User	Crypto Officer	Wireless Client	Description	CSP Access U: User, CO: Crypto Officer, WC: Wireless Client W: Write, E: Execute, R: Read
Login	Yes	No	No	Log into the Module to access CLI (Command Line Interface).	<u>All Cases:</u> User Password: U (E) <u>Via SSH:</u> SSH-RSA-PRK: U (E) SSH-DH-PRK, Outbound and Inbound SSH-MAK and SSH-MEK: U (W), U (E)
Connect to Server	Yes	No	No	Configure settings to establish connection with Server	<u>Server connection over UDP or TLS:</u> SPT-SK: U (E) Outbound and Inbound SPT-MAK and SPT-MEK: U (W), U (E) PSK: U (W)

					Server connection over TLS: TLS-ECDH-PRK, TLS-PMK, TLS-MK, TLS-MAK, TLS-MEK: U (W), U (E) User Via SSH: Outbound and Inbound SSH-MAK and SSH-MEK: U (E)
Operational settings	Yes	Yes	No	Configure operational settings such as network settings, wireless settings etc.	User Via SSH: Outbound and Inbound SSH-MAK and SSH-MEK: U (E) CO Via SpectraTalk/TLS: Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E) PSK, GMK: CO (W)
Change mode	Yes	No	No	Change mode of operation of the Module between FIPS and non-FIPS.	All Cases: All Passwords, Keys and CSPs in Table 6: U (W) Via SSH: Outbound and Inbound SSH-MAK and SSH-MEK: U (E)
Change shared secret key (K)	Yes	Yes	No	Change shared secret key (K) used between the Module and the Server.	All Cases: Shared Secret Key (K), SPT-SK, Outbound and Inbound SPT-MAK and SPT-MEK: U (W), CO (W) User Via SSH: Outbound and Inbound SSH-MAK and SSH-MEK: U (E) CO Via SpectraTalk/TLS:

					Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E)
Change User password	Yes	Yes	No	Change password used to authenticate the User.	<u>All Cases:</u> User Password: U (W), CO (W) <u>User Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E) <u>CO Via SpectraTalk/TLS:</u> Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E)
View self test result	Yes	No	No	Check result of self tests.	<u>Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E)
Perform on-demand self test (same as Reboot)	Yes	Yes	No	Perform self tests/reboot the Module.	<u>All Cases:</u> SSH-DH-PRK, Outbound and Inbound SSH-MAK and SSH-MEK: U (W), CO (W) TLS-ECDH-PRK, TLS-PMK, TLS-MK, Outbound and Inbound TLS-MAK and TLS-MEK: U (W), CO (W) SPT-SK, Outbound and Inbound SPT-MAK and SPT-MEK: U (W), CO (W) Seed for DRBG and V and Key: U (W), CO (W)

					Preserved State of LRNG: U (W), U (R), CO (W), CO (R) <u>User Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E) <u>CO Via SpectraTalk/TLS:</u> Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E)
Reset factory defaults (same as Zeroize)	Yes	No	No	Restore the Module to factory default state.	<u>All Cases:</u> All Passwords, Keys and CSPs in Table 6: U (W) <u>Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E)
Show status	Yes	Yes	No	View status of operation of the Module.	<u>User Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E) <u>CO Via SpectraTalk/TLS:</u> Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E)
Upgrade firmware	Yes	Yes	No	Upgrade firmware of Module.	<u>User Via SSH:</u> Outbound and Inbound SSH-MAK and SSH-MEK: U (E) <u>CO Via SpectraTalk/TLS:</u>

					Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E)
Change PSK	No	Yes	No	Change PSK used to authenticate Wireless Client and to generate session keys for its traffic.	CO Via SpectraTalk/TLS: Outbound and Inbound TLS-MAK and TLS-MEK: CO (E) Outbound and Inbound SPT-MAK and SPT-MEK: CO (E) PSK: CO (W)
Set up wireless link	No	No	Yes	Connect to the Module.	PMK, PTK: WC (W) GTK, IGTK: WC (R)
Transfer wireless traffic	No	No	Yes	Send and receive traffic over wireless link with the Module.	PTK, GTK, IGTK: WC (E)
Tear down wireless link	No	No	Yes	Disconnect from Module.	PMK, PTK, GTK, IGTK: WC (W) GMK: WC (E)

Table 5: Services

b) Controlling Access to the Module for the First time

Upon switching the Module to FIPS mode, the User is required to:

- Put tamper evident labels on the Module
- Change the User password from its factory default value
- Change the shared secret key (K) from its factory default value
- Permit the Module to connect to the Server

c) Cryptographic Operations

The Module performs the following encryption/decryption functions.

c.1) Communication with Server

The Module communicates with the Server over UDP protocol. Optionally, it can connect to the server over TCP using TLS version 1.2. Messages exchanged between the Module and the Server correspond to data output egressing the Network Interface of the Module and directed to the Server. They also correspond to the control input from Server (Server commands) and the status output from the Module to the Server. These messages are referred herein as SpectraTalk messages. The SpectraTalk messages, whether carried over UDP or within TLS, are afforded cryptographic protection of their own via HMAC-SHA-1 authentication and AES encryption as described below.

Derivation and Transport of Session Key

After successful mutual authentication, the Server randomly generates a session key (SPT-SK) and transports it to the Module. The shared secret key K encrypts the message carrying SPT-SK from the Server to the Module using AES-CBC encryption and authenticates it using HMAC-SHA-1. The session key is 128 bits in length.

Derivation of Message Authentication and Encryption Keys

The message authentication key (SPT-MAK) and message encryption key (SPT-MEK) are derived from the key SPT-SK using SP 800-108. The the SPT-MAK is 160 bits in length and SPT-MEK is 128 bits in length. The SPT-MAK is used for per-message HMAC-SHA-1 authentication and the SPT-MEK is used for per-message AES-CBC encryption between the Module and the Server. There is different pair of (SPT-MAK, SPT-MEK) in each direction – Module to Server and Server to Module.

Shared Secret Key Management

Upon switching the Module to FIPS mode, the User is required to change the K from its factory default value. The key K resets to factory default (zeroized) upon entering or exiting the FIPS mode. The User has to manually input the new value of key K in the Module. When the Module is properly connected to the Server, the Server can electronically change the shared secret key (K). The new key travels from the Server to the Module within a message that is protected via AES encryption and HMAC-SHA-1 authentication with the key SPT-SK. The User is required to zeroize the shared secret key either by resetting the module to factory default or by exiting the FIPS mode when the Module is to be discarded. The plaintext key K is never outputted from the Module.

Use of TLS 1.2 for Communication with Server

When using TLS over TCP transport between the Module and the Server, a TLS tunnel is first established by the Module with the Server and the SpectraTalk messages are carried inside the TLS tunnel. In this case, there are two layers of cryptographic protections - first is by virtue of cryptographic protection of SpectraTalk messages themselves as described above and the other by virtue of cryptographic protection of the TLS tunnel. For the establishment of the TLS tunnel, the Module acts as TLS client. It

supports ECDHE_RSA_WITH_AES_256_CBC_SHA384, with RSA key size of 2048 bits, SHA-2 for digital signature, secp256r1 curve (also known as P-256 curve) for ECDHE.

c.2) SSH Server

SSH server in the Module is used to support remote access for the User role. The RSA public and private key pair (2048 bits) for host authentication is randomly generated when the Module is first booted upon entering/exiting FIPS mode and reset to factory defaults. At the time of establishing SSH session, Diffie-Hellman key exchange (diffie-hellman-group14-sha1, public key size 2048 bits) is performed to establish shared secret. Message authentication key (160 bits) and message encryption key (128 bits) are derived from the shared secret using SSH KDF. The message authentication key is used for HMAC-SHA-1 integrity protection and the message encryption key is used for AES-CBC encryption within the SSH session.

c.3) Communication with Wireless Client

Unicast frames between the Module and the Wireless Client are protected with AES-CCM using 128/256-bit Temporal Key (TK). Broadcast and multicast data-type frames from the Module to the Wireless Clients are protected with AES-CCM using 128/256-bit Group Transient Key (GTK). Broadcast management-type frames from the Module to the Wireless Clients are protected with 16 byte AES-CMAC message integrity code with 128/256-bit Integrity Group Transient Key (IGTK). These keys are derived using EAPOL 4-way handshake at the beginning of the connection, as specified in the IEEE 802.11i standard. When Module connects to Server, PSK travels from the Server to the Module within a message that is protected via AES encryption with the key SPT-MEK and HMAC-SHA-1 authentication with the key SPT-MAK. When Wireless Client initiates connection to the Module, the Module sets PMK for the Wireless Client equal to PSK. The Wireless Client needs to possess the same PSK/PMK for successful connection establishment. The messages exchanged in the 4-way handshake are as follows.

M1: The Module generates a random nonce (called ANonce) and sends it to the Wireless Client.

M2: The Wireless Client generates a random nonce of its own (called SNonce). Using ANonce, SNonce, PMK and some other parameters, the Wireless Client generates a 384/512-bit Pairwise Transient Key (PTK) according to SP 800-108. The PTK is a concatenation of 128-bit Key Confirmation Key (KCK), 128-bit Key Encryption Key (KEK), and 128/256-bit Temporal Key (TK). The Wireless Client sends SNonce to the Module in message M2. KCK is used to generate message integrity code (MIC) of 16 bytes on the entire M2 using HMAC-SHA-1 or AES-CMAC, and also on the messages M3 and M4 described below.

M3: Upon receiving SNonce, the Module generates the 384/512-bit PTK just as the Wireless Client does according to SP 800-108. Additionally, the Module randomly generates 128-bit GMK and from it generates 128/256-bit GTK and 128/256-bit IGTK according to SP 800-108. It encrypts the GTK and IGTK using AES key wrap with KEK according to SP 800-38F and sends them to the Wireless Client in message M3.

M4: The Wireless Client unwraps GTK and IGTK and sends acknowledgement to the Module in message M4. This completes the 4-way handshake process.

c.4) Firmware Upgrade

The integrity of upgrade bundle is established via its SHA-256 hash created at build time. This hash is signed with RSA 2048 bits private key of Mojo development environment. The Module upon downloading the upgrade bundle verifies the signed hash value using RSA 2048 bits public key of the Mojo development environment.

d) Summary of Passwords, Keys and Critical Security Parameters (CSPs)

d.1) Passwords, Keys and CSPs

Passwords, keys and CSPs used in the Module are summarized in Table 6. None of the passwords, keys and CSPs are shared between FIPS and non-FIPS modes. Also, they are never output from the module.

Password/ Key/CSP	Description	Generation	Input/Output	Storage	Zeroization
User Password	Used to authenticate the User. Password should be at least 6 characters in length.	NA	Manually entered in plain text by User from GPC (General Purpose Computer) keyboard. Electronically input by Crypto Officer in message protected with encryption and	SHA-1 hash with salting in non-volatile memory.	Overwritten with factory default value upon change mode and reset factory defaults. Overwritten with new value upon change password.

			authentication.		
Shared Secret Key (K)	Used for HMAC-SHA-1 mutual authentication with the Server and AES encryption of message transporting SPT-SK to the Module. This key is 128 bits in length.	NA	Manually entered in plain text by User from GPC (General Purpose Computer) keyboard. Electronically input by Crypto Officer in message protected with encryption and authentication.	Plain Text in non-volatile memory.	Overwritten with zeros and then replaced with default value upon change mode and reset factory defaults. Overwritten with new value upon change key.
SpectraTalk Session key (SPT-SK)	Used to derive authentication and encryption key pairs (SPT-MAK, SPT-MEK) for communication between the Module and the Server. The key SPT-SK is 128 bits in length.	Generated externally by the Server.	Electronically input by Crypto Officer in message protected with encryption and authentication by key K.	Plaintext in volatile memory.	Overwritten with zeros upon termination of session between the Module and the Server.
Outbound and Inbound Message Authentication Keys (SPT-MAK)	SPT-MAK and SPT-MEK are used for HMAC-SHA-1	Derived from SPT-SK.	NA	Plaintext in volatile memory.	Overwritten with zeros upon termination of session between the

and Message Encryption Keys (SPT-MEK)	authentication and AES encryption of messages between the Module and the Server. There are separate SPT-MAK and SPT-MEK in outbound and inbound directions. Each key SPT-MAK is 160 bits in length and SPT-MEK is 128 bits in length.				Module and the Server.
Elliptic Curve Diffie-Hellman Private Key in TLS Client (TLS-ECDH-PRK)	This is the private key component of the Elliptic Curve Diffie-Hellman (ECDH) key pair which is used for key agreement at the time of establishment of TLS session with the Server. The TLS-ECDH-PRK is 256 bits in length. The key establishment methodology provides 128	Generated internally.	NA	Plaintext in volatile memory.	Overwritten with zeros upon derivation of master secret.

	bits of encryption strength.				
Premaster Secret in TLS Client (TLS-PMK)	Premaster secret is agreed between the Module and the Server at the time of establishment of the TLS session. It is established via ECDH exchange. The premaster secret is used to establish master secrets for TLS connections. It is 256 bits in length.	EC Diffie-Hellman key establishment.	NA	Plaintext in volatile memory.	Overwritten with zeros upon derivation of master secret.
Master Secret (TLS-MK), and Outbound and Inbound Message Authentication (TLS-MAK) and Encryption (TLS-MEK) keys in TLS Client	Master secret (of size 48 bytes) is generated for each TLS connection within the TLS session with the Server, using the premaster secret for the TLS session. The master secret is used to derive HMAC-SHA-	Derived from pre-master secret.	NA	Plaintext in volatile memory.	Overwritten with zeros upon termination of TLS session between the Module and the Server.

	384 message authentication key (384 bits) and AES message encryption key (256 bits). There are separate authentication and encryption keys in each direction (outbound and inbound).				
RSA Private Key for SSH Server (SSH-RSA-PRK)	This is the private key component of the RSA key pair which is used for host authentication in SSH. The RSA private key is 2048 bits in length. It is generated when the Module first boots up.	Generated internally.	NA	Plaintext in non-volatile memory.	Overwritten with zeros upon change mode and reset factory defaults.
Diffie-Hellman Private Key for SSH Server (SSH-DH-PRK)	This is the private key component of the Diffie-Hellman key pair which is used for key agreement at the time of establishment of SSH	Generated internally.	NA	Plaintext in volatile memory.	Overwritten with zeros upon termination of SSH session.

	session. The Diffie-Hellman private key is at least 224 bits in length. The key establishment methodology provides 112 bits of encryption strength.				
SSH Per-session Outbound and Inbound Message Authentication (SSH-MAK) and Encryption (SSH-MEK) Keys	There is one 160-bit key for HMAC-SHA-1 message authentication and one 128-bit key for AES message encryption in each direction in the SSH session. These keys are generated at the time of SSH session establishment.	Diffie Hellman key establishment.	NA	Plaintext in volatile memory.	Overwritten with fixed pattern upon termination of SSH session.

Seed for DRBG (Deterministic Random Bit Generator) and V and Key values	DRBG is seeded with random 256-bit AES key and random 128-bit Nonce (V). These values are obtained by reading bytes from the /dev/random device and feeding them directly to DRBG.	Generated internally.	NA	Plaintext in volatile memory.	Overwritten with zeros upon instantiation of DRBG.
Preserved State of LRNG (Linux Random Number Generator)	During system shutdown, the Module extracts 512 bytes from LRNG output pool and stores them in non-volatile memory in plaintext. During system start, the stored 512 bytes are mixed into the output pools.	Generated internally.	NA	Plaintext in non-volatile memory.	Overwritten with zeros upon change mode and reset to factory default. Overwritten with new value upon reboot.

Pre Shared Key (PSK)	This is 256 bit key.	NA	Electronically input by Crypto Officer in message protected via encryption and authentication.	Plaintext in non-volatile memory.	Overwritten with zeros upon change mode, reset to factory default and removal of WiFi profile. Overwritten with new value upon change PSK.
Pairwise Master Key (PMK)	This is 256 bit key used for derivation of PTK.	Set to value same as PSK by the Module.	NA	Plaintext in volatile memory.	Overwritten with zeros upon change mode,reset to factory default and when Wireless Client disconnects. Overwritten with new value upon change PSK.
Pairwise Transient Key (PTK)	This 384/512-bit key is a concatenation of 128-bit KCK for MIC, 128-bit KEK for AES key wrap, and 128/256-bit TK for AES-CCM.	Derived from PMK using SP 800-108.	NA	Plain text in volatile memory.	Overwritten with zeros upon change mode, reset to factory default, and when Wireless Client disconnects from the Module.

Group Master Key (GMK)	GMK is 128-bit randomly generated key.	Generated internally.	NA	Plain text in volatile memory.	Overwritten with zeros upon change mode, reset to factory default and removal of WiFi profile.
Group Transient Key (GTK) and Integrity Group Transient Key (IGTK)	GTK is 128/256-bit key for AES-CCM and IGTK is 128/256-bit key for AES-CMAC.	Derived from GMK using SP 800-108.	Output using AES key wrap.	Plain text in volatile memory.	Overwritten with zeros upon change mode, reset to factory default, and when Wireless Client disconnects from the Module.

Table 6: Passwords, Keys, CSPs

d.2) Public Keys

The following public keys are used in the Module in FIPS mode:

- RSA Public Key for SSH Server (2048 bits): It is the public key counterpart of the private key used for host authentication in SSH. It cannot be shared between non-FIPS and FIPS modes. This key is stored in plaintext in non-volatile memory. The RSA key pair is generated when the Module is first booted, and thereafter on entering or exiting FIPS mode, or on reset to factory default. It is output from the Module to the SSH client, when the client attempts connection to the SSH server in the Module.

- Diffie-Hellman Public Key for SSH Server (2048 bits): It is the public key component of the private key used for key agreement during SSH session establishment. It cannot be shared between non-FIPS and FIPS modes. This key is stored in plaintext in volatile memory. It is deleted upon termination of the SSH session. Power cycle, reboot, entering and exiting FIPS mode, and reset to factory default result in termination of the SSH session and hence deletion of this key. It is output from the Module to the SSH client, when the client connects to the SSH server in the Module.

- ECDH Public Key for TLS Client (256 bits): It is the public key component of the private key used for key agreement during TLS session establishment. It cannot be shared between non-FIPS and FIPS modes. This key is stored in plaintext in volatile memory. It is deleted upon termination of the TLS session. Power cycle, reboot, entering and exiting FIPS mode, and reset to factory default result in termination of TLS session and hence deletion of this key. It is output from the Module to the Server, when the Module connects to the Server over TLS.

The following public keys do not belong to the Module, but are stored in the module and are used in the FIPS mode:

- RSA Public key of certificate authority (CA) for Server certificate validation (2048 bits): It is the public key counterpart of the private key used for signing the Server certificate. This key is input into the Module by the User. It is stored in plaintext in non-volatile memory and it is not output from the module.

- RSA Public key for firmware upgrade bundle signature verification (2048 bits): It is the public key counterpart of the private key used for signing the upgrade bundle at build time. The RSA key pair is generated during the build process and the public key to be used for verification of the next upgrade bundle is included in the current firmware version. This key is stored in plaintext in non-volatile memory. The public key is not output from the module. The key is overwritten during the firmware upgrade with a new RSA public key that is to be used for verification of the next upgrade bundle. The new key may be the same as or different from the present key.

e) Summary of Cryptographic Algorithms

The Module implements FIPS approved algorithms described in Table 7.

Algorithm	**CAVP Certificate**	**Standard**	**Mode/ Method**	**Key length or Moduli**	**Use**
AES	AES #5158	FIPS 197	CBC	128 bits, 256 bits	Encryption/decryption
AES	AES #5412	SP 800-38C	CCM	128 bits, 256 bits	Encryption/decryption
SHA-1	SHS #4165	FIPS 180-4			Message digest and key derivation (SSH)
SHA-256	SHS #4165	FIPS 180-4			Message digest (TLS)

SHA-384	SHS #4165	FIPS 180-4			Key Derivation(TLS)
HMAC-SHA-1	HMAC #3419	FIPS 198		128 bits, 160 bits	Message integrity (SpectraTalk, SSH); key derivation (SpectraTalk)
HMAC-SHA-384	HMAC #3419	FIPS 198		384 bits	Message integrity (TLS)
CMAC	AES #5449	SP 800-38B		128 bits, 256 bits	Message integrity (4-way handshake, broadcast management frame protection)
RSA	RSA #2774	FIPS 186-4	PKCS#1 v1.5	2048 bits	Key generation (SSH), sign (SSH), verify (TLS)
DRBG	DRBG #1936	SP 800-90AR1	AES-CTR	256 bits	Random number generation
KBKDF	KBKDF #174	SP 800-108	Counter, HMAC-SHA-1		Key derivation (SpectraTalk)
KBKDF	KBKDF #219	SP 800-108	Counter, HMAC-SHA-1		Key derivation (PTK, GTK, IGTK)
KBKDF	KBKDF #220	SP 800-108	Counter, HMAC-SHA-256		Key derivation (PTK, GTK, IGTK)
Key Wrap***	AES #5473	SP 800-38F	AES	128 bits	Key transport (GTK, IGTK)

SSH** KDF	CVL #1672	SP 800-135			Key derivation for SSH per-session message authentication and encryption keys
TLS** KDF	CVL #1672	SP 800-135			Key derivation for TLS per-session message authentication and encryption keys
ECC CDH Primitive	CVL #1673	SP 800-56A		P-256	Pre-master secret establishment (TLS)
KTS	AES #5158 and HMAC #3419	SP 800-38F		128 bit AES and 160 bit HMAC	Key transport through SSH and SpectraTalk
Vendor affirmed	CKG*	SP 800-133			Key generation

Table 7: Approved Algorithms

(Note*: Key generation using unmodified output from an approved DRBG as the random seed of FIPS 186-4 key generation).

(Note**: No parts of the SSH and TLS protocols, other than the KDF, have been tested by the CAVP or CMVP).

(Note***: When a wrapped key is larger in size larger than the encryption key used for wrapping, strength of the encryption is determined by the encryption key).

The Module implements FIPS allowed algorithms described in Table 8.

Algorithm	**Caveat**	**Use**

Diffie-Hellman	Provides 112 bits (Oakley group 14) of encryption strength	Key Establishment in SSH
EC Diffie-Hellman	Provides 128 bits (P-256) of encryption strength	Key Establishment in TLS
MD5	Used for firmware integrity test	Integrity Test
NDRNG	Linux /dev/random device which provides 384 bits of entropy	Seeding for DRBG

Table 8: Allowed Algorithms

The Module implements non-security functions described in Table 9.

Function	**Caveat**	**Algorithm**
Decryption of upgrade bundle	No security claimed for encryption of upgrade bundle	AES-CBC with 128-bit key

Table 9: Additional Cryptographic Mechanisms

4. Self Tests

The Module always reboots when FIPS mode is entered. At boot time, firmware integrity check is done using MD5 checksum of the firmware.

If the firmware integrity test passes, the Module performs power-up self tests shown in Table 10.

Algorithm	**Test**
AES-ECB	Encrypt KAT and Decrypt KAT
AES-ECB (Qualcomm QCA9994)	Encrypt KAT
AES-CMAC	Generate KAT and Verify KAT CBC mode
RSA	Sign KAT and Verify KAT
DRBG	KAT for Instantiate, Reseed and Generate
SHA-1	KAT
SHA-256	KAT
HMAC-SHA-1	KAT
HMAC-SHA-384	KAT
ECC CDH	KAT
KBKDF (SP 800-108)	KAT

Table 10: Power-up Self Tests

If any of the above tests fails, the Module goes to the Error State.

During operation, the module performs conditional self tests shown in Table 11.

Algorithm/Service	Test
RSA	Pairwise Consistency
DRBG	Continuous
NDRNG (/dev/random)	Continuous
Upgrade Firmware	Firmware Load (RSA 2048 SigVer)
Change Shared Secret Key (K)	Manual Key Entry

Table 11: Conditional Self Tests

If conditional test for RSA, DRBG, or NDRNG fails, the Module enters the Error State.

At the time of updating the Module to the new firmware, firmware load test (using RSA 2048 bits digital signature) is performed. If the firmware load test fails, the new firmware image is not loaded onto the Module and the Module continues operation with the existing firmware image. If the test succeeds, the old firmware is replaced with the new firmware.

In the Error State, the Module does not output any data on the Data Output interface. The results of the above tests can be viewed by the User by accessing the Module over the Console Interface.

For the manual key entry test, the key is accepted if the test succeeds. Else, the key is rejected.

It is also possible to perform on-demand self test by rebooting the Module.

5. Physical Security

The appliances containing the Module is made from production-grade components, which are encased within an opaque, hard, production-grade enclosure. The appliances meet commercial-grade specifications for power, temperature, reliability, shock and vibration.

Tamper Evident Labels

Two tamper evident labels shall be applied to appliance by the User to operate the Module in FIPS approved mode. The locations for applying these labels are shown in Figures 3(A) and 3(B). Before applying the labels, the receiver surface of the appliance must be cleaned with dry cloth or tissue paper. Tamper evident labels can be ordered from Mojo Networks (part number C-TPL-A).

Per FIPS 140-2 Implementation Guidance (IG 14.4), the User is responsible for securing and having control at all times of any unused labels. The user is also responsible for the

direct control and observation of any changes to the Module such as reconfigurations where the tamper evident labels or security appliances are removed or installed to ensure the security of the Module is maintained during such changes and the module is returned to a FIPS approved mode.

The User is also required to periodically inspect the Module for evidence of tampering at intervals defined by the organization's security policy. The inspection must include visual examination of tamper evident labels for tears, rips, dissolved adhesive, and other signs of malice. It must also include visual inspection of any enclosure defects such as broken screws, bends, depressions, scratches, and other questionable markings. If evidence of tampering is observed, the User must alert incident response team at the organization. Upon gathering of the evidence, the User must zeroize the Module.

Figure 3(A)

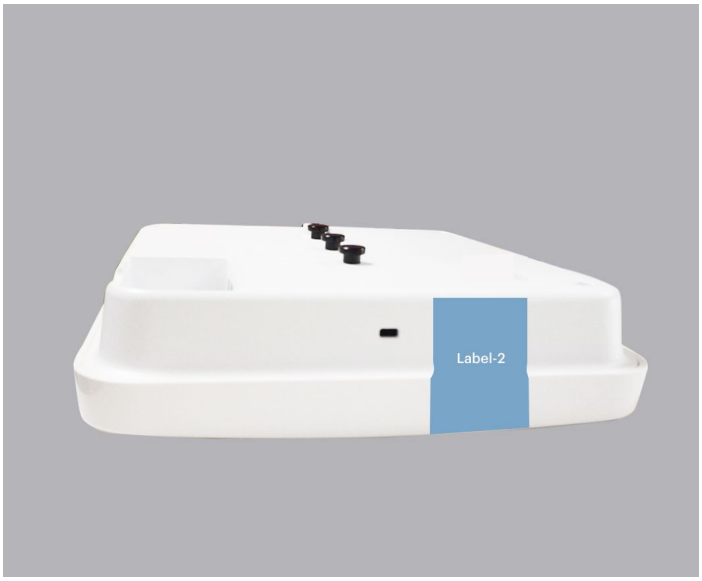

Figure 3(B)

6. Mitigation of Other Attacks

The Module does not mitigate other attacks.