



## FIPS 140-2 Non-Proprietary Security Policy

---

Embedded Module and Embedded Module Lite

Level 2 Validation

Document Version 2.0

August 27, 2021

*Prepared By:*



Persistent Systems, LLC  
303 Fifth Avenue Suite 306  
New York, NY 10016  
[www.persistentsystems.com](http://www.persistentsystems.com)

## **Abstract**

This document provides a non-proprietary FIPS 140-2 Security Policy for the Embedded Module and Embedded Module Lite

## Table of Contents

|       |   |    |
|-------|---|----|
| 1     | Introduction .....  | 5  |
| 1.1   | About FIPS 140 .....  | 5  |
| 1.2   | About this Document .....   | 5  |
| 1.3   | External Resources .....  | 5  |
| 1.4   | Notices .....   | 5  |
| 1.5   | Acronyms.....   | 6  |
| 2     | Persistent Systems Embedded Module and Embedded Module Lite ..... | 7  |
| 2.1   | Wave Relay® Product Overview .....                                | 7  |
| 2.2   | Cryptographic Module Specification .....                          | 7  |
| 2.2.1 | Validation Level Detail.....                                      | 9  |
| 2.2.2 | Algorithm Implementation Certificates .....                       | 9  |
| 2.3   | Module Interfaces .....   | 15 |
| 2.4   | Roles, Services, and Authentication .....                         | 16 |
| 2.4.1 | Operator Services and Descriptions.....                           | 17 |
| 2.4.2 | Operator Authentication.....                                      | 20 |
| 2.5   | Physical Security.....  | 20 |
| 2.6   | Operational Environment.....                                      | 20 |
| 2.7   | Cryptographic Key Management.....                                 | 21 |
| 2.8   | Self-Tests.....   | 26 |
| 2.8.1 | Power-On Self-Tests .....   | 26 |
| 2.8.2 | Conditional Self-Tests.....                                       | 28 |
| 2.9   | EMI/EMC .....   | 28 |
| 2.10  | Mitigation of Other Attacks.....                                  | 29 |
| 3     | Guidance and Secure Operation .....                               | 30 |
| 3.1   | Crypto Officer and User Guidance .....                            | 30 |
| 3.1.1 | Initialization for FIPS Mode of Operation .....                   | 30 |
| 3.1.2 | General Crypto Officer and User Guidance.....                     | 30 |

## List of Tables

|   |    |
|---|----|
| Table 1 – Acronyms and Terms .....  | 6  |
| Table 2 – Validation Level by DTR Section.....                                | 9  |
| Table 3 – Algorithm Certificates for Wave Relay® E2 Cryptographic Engine..... | 9  |
| Table 4 – Algorithm Certificates for Wave Relay® Cryptographic Kernel .....   | 9  |
| Table 5 – Algorithm Certificates for Wave Relay® Cryptographic Library .....  | 15 |
| Table 6 – Logical Interface / Physical Interface Mapping.....                 | 16 |
| Table 7 – Operator Services and Descriptions .....                            | 19 |
| Table 8 – Key/CSP Management Details (also includes public keys).....         | 25 |
| Table 9 – Cryptographic Engine POST .....                                     | 26 |
| Table 10 – Cryptographic Kernel POST.....                                     | 27 |
| Table 11 – Cryptographic Library POST .....                                   | 28 |
| Table 12 – Conditional Self-Tests .....                                       | 28 |

## List of Figures

|  |   |
|--|---|
| Figure 1 – Physical Boundary of Embedded Module .....      | 8 |
| Figure 2 – Physical Boundary of Embedded Module Lite ..... | 8 |

## 1 Introduction

### 1.1 About FIPS 140

Federal Information Processing Standards Publication 140-2 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic products to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Communications Security Establishment (CSE) Cryptographic Module Validation Program (CMVP) owns the FIPS 140 program. The CMVP accredits independent testing labs to perform FIPS 140 testing; the CMVP also validates test reports for all products pursuing FIPS 140 validation. *Validation* is the term given to a product that is documented and tested against the FIPS 140 criteria.

More information is available on the CMVP website at <https://csrc.nist.gov/projects/testing-laboratories>

### 1.2 About this Document

This non-proprietary Cryptographic Module Security Policy for the Persistent Systems Embedded Module and Embedded Module Lite provides an overview of the product and a high-level description of how it meets the security requirements of FIPS 140-2. This document contains details on the module's cryptographic keys and critical security parameters. This Security Policy concludes with instructions and guidance on running the modules in a FIPS 140-2 mode of operation.

The Embedded Module and Embedded Module Lite may also be referred to as the “module” in this document.

### 1.3 External Resources

The Persistent Systems website (<http://www.persistentsystems.com>) contains information on the full line of products from Persistent Systems, including a detailed overview of the Embedded Module and Embedded Module Lite solutions. The Cryptographic Module Validation Program website (<https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program>) contains links to the FIPS 140-2 certificate and Persistent Systems contact information. The Cryptographic Module Validation Program website (<https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program>) contains links to the FIPS 140-2 certificate and Persistent Systems contact information.

### 1.4 Notices

This document may be freely reproduced and distributed in its entirety without modification.

## 1.5 Acronyms

The following table defines acronyms found in this document:

| Acronym | Term   |
|---------|--|
| AES     | Advanced Encryption Standard                   |
| DRBG    | Deterministic Random Number Generator          |
| CSE     | Communications Security Establishment          |
| CSP     | Critical Security Parameter                    |
| DTR     | Derived Testing Requirement                    |
| FIPS    | Federal Information Processing Standard        |
| FTL     | Flash Translation Layer                        |
| GUI     | Graphical User Interface                       |
| HMAC    | Keyed-Hash Message Authentication Code         |
| KAT     | Known Answer Test                              |
| MANET   | Mobile Ad-hoc Network                          |
| MPU     | Man Portable Unit                              |
| NIST    | National Institute of Standards and Technology |
| NDRNG   | Non Deterministic Random Number Generator      |
| PCT     | Pairwise Consistency Test                      |
| SHA     | Secure Hashing Algorithm                       |
| TLS     | Transport Layer Security                       |

Table 1 – Acronyms and Terms

## 2 Persistent Systems Embedded Module and Embedded Module Lite

### 2.1 Wave Relay® Product Overview

The Wave Relay® System is a peer-to-peer wireless MANET networking solution in which there is no master node. If any device fails, the rest of the devices continue to communicate using any remaining connectivity. By eliminating master nodes, gateways, access points, and central coordinators from the design, Wave Relay® delivers high levels of fault tolerance regardless of which nodes might fail. The system is designed to maximize the capacity of the radio frequency (RF) spectrum and to minimize the network overhead. While optimizing efficiency, Wave Relay® also implements techniques that increase multicast reliability. The advanced multicast functionality allows the system to support both multicast voice and video over IP.

Wave Relay® is designed to maintain high bandwidth connectivity among devices that are on the move. The system is scalable, enabling it to incorporate unlimited meshed devices into the wireless network, where the devices themselves form the communication infrastructure. Even in highly dynamic environments, the system is able to maintain connectivity by rapidly re-routing data as necessary. Wave Relay® is a self-forming and self-healing network where nodes can move freely within the network. Critical information flows reliably throughout the network while individual data paths are able to adapt at sub-second intervals. This unique approach creates an ideal environment for maximizing performance across the available communications medium. Customers leverage Wave Relay®'s straight forward and effective architecture to enable a true "Plug and Play" capability. Deploying a Wave Relay® network is as simple as connecting a standard Ethernet cable; customers are immediately connected to everything on the network.

Wave Relay® is a seamless wireless networking system offering a dynamic and reliable solution for all mobile networking needs. The Persistent Systems Embedded Module and Embedded Module Lite offers the Wave Relay® MANET combined with other leading-edge technologies in a single smart radio.

### 2.2 Cryptographic Module Specification

The module is the Embedded Module HW P/N WR-5200 Versions 4.0, 6.0, 7.0, and 12.B and the Embedded Module Lite HW P/N WR-5250 Version 1.0, 3.0, and 12.B. The Embedded Module Lite is identical to the Embedded Module, except for the following:

- Audio: Codec and connectors for Microphone and Speaker were removed
- Accelerometer/Gyroscope (IMU) were removed
- Flash memory is reduced from 128GB to 32GB
- HDMI Video input chip, supporting hardware and connector were removed
- SDI Video input chip, supporting hardware and MMCX connector were removed
- Analog video input was removed
- GPS chip, supporting hardware and MMCX connector were removed

The module uses FW Versions **19.5.5** and **19.6.4**. Each module is a multiple-chip embedded embodiment.

## FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

The physical cryptographic boundary is defined as the module board with heat-sinks, which includes the Wave Relay® main board, including the hardware cryptographic accelerator chip, drivers, CPU, and on-board flash memory. The boundary does not include the radio module or power supply (not depicted).



Figure 1 – Physical Boundary of Embedded Module



Figure 2 – Physical Boundary of Embedded Module Lite

The module is in FIPS-approved mode of operation when the validated firmware is used and when the guidance in Section 3.1 is adhered to. It does not have any bypass capability. The module does not support a non-Approved mode.



### 2.2.1 Validation Level Detail

The following table lists the level of validation for each area in FIPS 140-2:

| FIPS 140-2 Section Title                                     | Validation 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                                      | N/A              |
| Cryptographic Key Management                                 | 2                |
| Electromagnetic Interference / Electromagnetic Compatibility | 2                |
| Self-Tests   | 2                |
| Design Assurance   | 3                |
| Mitigation of Other Attacks                                  | N/A              |
| <b>Overall Level</b>   | <b>2</b>         |

Table 2 – Validation Level by DTR Section

### 2.2.2 Algorithm Implementation Certificates

The Embedded Module’s cryptographic algorithm implementations have received the following certificate numbers from the Cryptographic Algorithm Validation Program:

| Algorithm Type | Algorithm  | Standard   | CAVP Cert.  | Use                          |
|----------------|--|------------|-------------|------------------------------|
| Symmetric Key  | AES- {128*, 192*, 256} in {CBC*, CTR, ECB*} mode | FIPS 197   | Cert. #4456 | Data encryption / decryption |
| Keyed Hash     | HMAC-SHA-{1, 224*, 256}                          | FIPS 198   | Cert. #2957 | Message integrity            |
| Hashing        | SHA- {1, 224*, 256}                              | FIPS 180-4 | Cert. #3668 | Message digest               |

Table 3 – Algorithm Certificates for Wave Relay® E2 Cryptographic Engine

\* Denotes that the algorithm, mode of operation, and/or key size is not used/accessible

| Algorithm Type | Algorithm  | Standard   | CAVP Cert.  | Use                           |
|----------------|--|------------|-------------|-------------------------------|
| Symmetric Key  | AES- {128*, 192*, 256*} in {CBC*, CTR*, GCM*, ECB*} mode | FIPS 197   | Cert. #4454 | Not currently used/accessible |
| Symmetric Key  | AES- {128*, 256*} in {XTS*} mode                         | FIPS 197   | Cert. #4454 | Not currently used/accessible |
| Keyed Hash     | HMAC-SHA-{1*, 224*, 256*, 384*, 512}                     | FIPS 198   | Cert. #2955 | Message integrity             |
| Hashing        | SHA-{1*, 224*, 256*, 384*, 512}                          | FIPS 180-4 | Cert. #3666 | Message digest                |

Table 4 – Algorithm Certificates for Wave Relay® Cryptographic Kernel

\* Denotes that the algorithm, mode of operation, and/or key size is not used/accessible

| Algorithm Type                 | Algorithm   | Standard                          | CAVP Cert.      | Use                           |
|--------------------------------|---|-----------------------------------|-----------------|-------------------------------|
| Symmetric Key                  | AES- {128, 192, 256} in {CBC, OFB*, CTR, GCM <sup>1</sup> , ECB*, CFB-1*, CFB-8*, CFB-128*} mode  | FIPS 197                          | Cert. #4455     | Data encryption / decryption  |
| CKG                            | Cryptographic Key Generation:<br>Asymmetric signature key generation using unmodified DRBG output<br><br>Direct symmetric key generation using unmodified DRBG output<br><br>Derivation of symmetric keys from a key agreement shared secret. | SP800-133                         | Vendor Affirmed | Key Generation                |
| Asymmetric Key                 | ECDSA SigGen Component - Curves:<br>P-224* P-256* P-384* P-521*<br>K-233 * K-283* K-409* K-571*<br>B-233* B-283* B-409* B-571*  | SP 800-56A                        | CVL #1164       | Not currently used/accessible |
| Transport Layer Security (TLS) | Section 4.2, TLS-<br>TLS (TLS1.0/1.1 TLS1.2 (SHA 256, 384, 512) )<br><br>No parts of this protocol, other than the KDF, have been tested by the CAVP and CMVP.  | SP 800-135<br>Section 4.2         | CVL #1163       | Key Derivation                |
| Asymmetric Key                 | RSA Decryption Primitive - RSADP: (Mod2048)   | SP 800-56B<br>Section 7.1.2       | CVL #1162       | Key Recovery                  |
| KAS ECC                        | ECC CDH -<br>Curves: B-233*, B-283*, B-409*, B-571*, K-233*, K-283*, K-409*, K-571*, P-224*, P-256*, P-384*, P-521*<br><br>KAS ECC –<br>( EC: P-256 SHA256)*<br>( ED: P-384 SHA384 )<br>( EE: P-521 SHA512)*                                  | SP 800-56Arev2<br>Section 5.7.1.2 | CVL #1161       | Key Agreement                 |

<sup>1</sup> The module is compatible with TLSv1.2 and provides support for the acceptable GCM cipher suites from SP 800-52 Rev1, Section 3.3.1. The counter portion of the IV is set by the module within its cryptographic boundary. When the IV exhausts the maximum number of possible values for a given session key, the first party, client or server, to encounter this condition will trigger a handshake to establish a new encryption key. In case the module's power is lost and then restored, a new key for use with the AES GCM encryption/decryption shall be established.

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

| Algorithm Type                  | Algorithm  | Standard   | CAVP Cert.  | Use                           |
|---------------------------------|--|------------|-------------|-------------------------------|
| Deterministic Random Bit (DRBG) | CTR_DRBG - {128, 192, 256}-CTR*,<br>HASH_DRBG - SHA- {1, 224, 256, 384, 512}*,<br>HMAC_DRBG - SHA- {1*, 224*, 256*, 384*, 512}}  | SP 800-90A | Cert. #1443 | Random Bit Generation         |
| Asymmetric Key                  | DSA - PQG(gen):<br>(2048, 224) SHA(224, 256, 384, 512)*<br>(2048, 256) SHA(256, 384, 512)*<br>(3072, 256) SHA(256, 384, 512)*<br>PQG(ver):<br>(1024, 160) SHA(1, 224, 256, 384, 512)*<br>(2048, 224) SHA(224, 256, 384, 512)*<br>(2048, 256) SHA(256, 384, 512)*<br>(3072, 256) SHA(256, 384, 512)*<br>KeyPairGen:<br>(2048, 224)*<br>(2048, 256)*<br>(3072, 256)*<br>SIG(gen):<br>(2048, 224) SHA(224, 256, 384, 512)*<br>(2048, 256) SHA(224, 256, 384, 512)*<br>(3072, 256) SHA(224, 256, 384, 512)*<br>SIG(ver):<br>(1024, 160) SHA(1, 224, 256, 384, 512)*<br>(2048, 224) SHA(1, 224, 256, 384, 512)*<br>(2048, 256) SHA(1, 224, 256, 384, 512)*<br>(3072, 256) SHA(1, 224, 256, 384, 512)* | FIPS 186-4 | Cert. #1191 | Not currently used/accessible |

| Algorithm Type | Algorithm  | Standard   | CAVP Cert.  | Use                                 |
|----------------|--|------------|-------------|-------------------------------------|
| Asymmetric Key | ECDSA Key Pair Gen, Sig Gen, Sig Ver<br>Key Pair Generation: CURVES<br>(P-224* P-256* P-384 P-521* K-233* K-283* K-409* K-571* B-233* B-283* B-409* B-571*)<br>Public Key Validation: CURVES<br>(P-192* P-224* P-256* P-384 P-521* K-163* K-233* K-283* K-409* K-571* B-163* B-233* B-283* B-409* B-571*)<br>SigGen: CURVES<br>(P-224*: (SHA-224*, 256*, 384*, 512*)<br>P-256*: (SHA-224*, 256*, 384*, 512*)<br>P-384: (SHA-224*, 256*, 384, 512*)<br>P-521*: (SHA-224*, 256*, 384*, 512*)<br>K-233*: (SHA-224*, 256*, 384*, 512*)<br>K-283*: (SHA-224*, 256*, 384*, 512*)<br>K-409*: (SHA-224*, 256*, 384*, 512*)<br>K-571*: (SHA-224*, 256*, 512*)<br>B-233*: (SHA-224*, 256*, 384*, 512*)<br>B-283*: (SHA-224*, 256*, 384*, 512*)<br>B-409*: (SHA-224*, 256*, 384*, 512*)<br>B-571*: (SHA-224*, 256*, 384*, 512*))<br>SigVer: CURVES<br>(P-192*: (SHA-1*, 224*, 256*, 384*, 512*)<br>P-224*: (SHA-1*, 224*, 256*, 384*, 512*)<br>P-256*: (SHA-1*, 224*, 256*, 384*, 512*)<br>P-384: (SHA-1*, 224*, 256*, 384, 512*)<br>P-521*: (SHA-1*, 224*, 256*, 384*, 512*)<br>K-233*: (SHA-1*, 224*, 256*, 384*, 512*)<br>K-283*: (SHA-1*, 224*, 256*, 384*, 512*)<br>K-409*: (SHA-1*, 224*, 256*, 384*, 512*)<br>K-571*: (SHA-1*, 224*, 256*, 384*, 512*)<br>B-233*: (SHA-1*, 224*, 256*, 384*, 512*)<br>B-283*: (SHA-1*, 224*, 256*, 384*, 512*)<br>B-409*: (SHA-1*, 224*, 256*, 384*, 512*)<br>B-571*: (SHA-1*, 224*, 256*, 384*, 512*))<br><b>Note: Only P-384 curve with SHA-384 is employed in this module</b> | FIPS 186-4 | Cert. #1085 | Signature Generation & Verification |
| Keyed Hash     | HMAC-SHA-{1, 224*, 256, 384*, 512}   | FIPS 198   | Cert. #2956 | Message integrity                   |

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

| Algorithm Type | Algorithm   | Standard | CAVP Cert.         | Use           |
|----------------|---|----------|--------------------|---------------|
| KTS            | AES Cert. #4455 and HMAC Cert. #2956; key establishment methodology provides between 128 and 256 bits of encryption strength) | IG D.9   | Cert. #4455, #2956 | Key Transport |

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

|                       |   |                                     |                    |  |
|-----------------------|---|-------------------------------------|--------------------|--|
| <p>Asymmetric Key</p> | <p>RSA - 186-2<br/>           Sig Ver 9.31*:<br/>           Modulus lengths (in bits): 1024, 1536, 2048, 3072, 4096<br/>           SHAs: SHA-{1, 256, 384, 512}<br/>           Sig Ver PKCS1.5*:<br/>           Modulus lengths (in bits): 1024, 1536, 2048, 3072, 4096<br/>           SHAs: SHA-{1, 256, 384, 512}<br/>           Sig Ver PSS*:<br/>           Modulus lengths (in bits): 1024, 1536, 2048, 3072, 4096<br/>           SHAs: SHA-{1, 256, 384, 512}</p> <p>RSA - 186-4<br/>           Key Gen 9.31:<br/>           Public Key Exponent: Fixed<br/>           Probable Random Primes:<br/>               Mod lengths (in bits): 2048, 3072*<br/>               Primality Tests: C.2<br/>           Sig Gen 9.31*:<br/>           Mod 2048 SHA: SHA-{1, 256, 384, 512}<br/>           Mod 3072 SHA: SHA-{1, 256, 384, 512}<br/>           Sig Ver 9.31*:<br/>           Mod 1024 SHA: SHA-{1, 256, 384, 512}<br/>           Mod 2048 SHA: SHA-{1, 256, 384, 512}<br/>           Mod 3072 SHA: SHA-{1, 256, 384, 512}<br/>           Sig Gen PKCS1.5:<br/>           Mod 2048 SHA: SHA-{1*, 224*, 256, 384*, 512*}<br/>           Mod 3072* SHA: SHA-{1*, 224*, 256*, 384*, 512*}<br/>           Sig Ver PKCS1.5:<br/>           Mod 1024* SHA: SHA-{1*, 224*, 256*, 384*, 512*}<br/>           Mod 2048 SHA: SHA-{1*, 224*, 256, 384*, 512}<br/>           Mod 3072* SHA: SHA-{1*, 224*, 256*, 384*, 512*}<br/>           Sig Gen PSS*:<br/>           Mod 2048 SHA: SHA-{1, 224, 256, 384, 512}<br/>           Mod 3072 SHA: SHA-{1, 224, 256, 384, 512}<br/>           Sig Ver PSS*:<br/>           Mod 1024 SHA: SHA-{1, 224, 256, 384, 512}</p> | <p>FIPS 186-2</p> <p>FIPS 186-4</p> | <p>Cert. #2433</p> | <p>Key Generation</p> <p>Signature Generation &amp; Verification</p> |
|-----------------------|---|-------------------------------------|--------------------|--|

| Algorithm Type | Algorithm  | Standard   | CAVP Cert.  | Use            |
|----------------|--|------------|-------------|----------------|
|                | Mod 2048 SHA: SHA-{1, 224, 256, 384, 512}<br>Mod 3072 SHA: SHA-{1, 224, 256, 384, 512} |            |             |                |
| Hashing        | SHA-{1, 224*, 256, 384, 512}   | FIPS 180-4 | Cert. #3667 | Message digest |

Table 5 – Algorithm Certificates for Wave Relay® Cryptographic Library

\* Denotes that the algorithm, mode of operation, and/or key size is not used/accessible

The following non-approved, but allowed protocols/algorithms are available in FIPS mode of operation:

- EC Diffie-Hellman (CVL Cert. #1161, key agreement; key establishment methodology provides 192 bits of encryption strength)
- MD5 within TLS only\*
- Hardware non-deterministic random number generator (NDRNG) (allowed for seeding FIPS-approved DRBG)
- RSA (key wrapping; key establishment methodology provides 112 bits of encryption strength)

\* No security is claimed from the use of these protocols/algorithms.

### 2.3 Module Interfaces

The interfaces for the cryptographic boundary include physical and logical interfaces. The physical interfaces provided by the module are mapped to five FIPS 140-2 defined logical interfaces: Data Input, Data Output, Control Input, Status Output, and Power. The mapping of logical interfaces to module physical interfaces is provided in the following table:

| Module Physical Interface (Port)  | FIPS 140-2 Logical Interface                          | Embedded Module | Embedded Module Lite |
|-----------------------------------|---|-----------------|----------------------|
| <b>HDMI Video In</b>              | Data Input  | X               |                      |
| <b>HDMI Video Out</b>             | Data Output, Status Output                            | X               | X                    |
| <b>3G-SDI Video In</b>            | Data Input  | X               |                      |
| <b>Serial (RS-232)</b>            | Data Input, Data Output, Control Input, Status Output | X               | X                    |
| <b>USB</b>                        | Data Input, Data Output, Control Input, Status Output | X               | X                    |
| <b>Ethernet</b>                   | Data Input, Data Output, Control Input, Status Output | X               | X                    |
| <b>Mic</b>                        | Data Input  | X               |                      |
| <b>GPS</b>                        | Data Input  | X               |                      |
| <b>Radio Data</b>                 | Data Input, Data Output, Control Input, Status Output | X               | X                    |
| <b>Speaker</b>                    | Data Output, Status Output                            | X               |                      |
| <b>Power/Zeroize/GPIO Control</b> | Control Input   | X               | X                    |
| <b>Status LED</b>                 | Status Output   | X               | X                    |
| <b>Power Input Port</b>           | Power   | X               | X                    |

Table 6 – Logical Interface / Physical Interface Mapping

## 2.4 Roles, Services, and Authentication

The module is accessed via Web browser over HTTPS/TLS. As required by FIPS 140-2, the module supports a Crypto Officer role and a User role. The module supports role-based authentication, and the respective services for each role are described in the following sections.

Both roles can access all services in the module. The module does not support a Maintenance role. The “Unauthenticated” role indicates services that the module performs automatically after POST and services that an operator may perform without authentication.



### 2.4.1 Operator Services and Descriptions

The services available to roles in the modules are as follows:

| Service                        | Description   | Roles                  | CSPs  |
|--------------------------------|---|------------------------|---|
| Power-On                       | Provides power and initializes the module. TLS key pairs are generated if unit was previously zeroized.                             | Unauthenticated        | Use Module Integrity Key<br>Use Store Key<br>Use/Zeroize DRBG entropy input<br>Use/Zeroize DRBG V<br>Use/Zeroize DRBG Key<br>Generate/Use/Zeroize CA Private Key<br>Generate/Use/Zeroize CA Public Key<br>Generate TLS ECDSA Private Key<br>Generate TLS ECDSA Public Key<br>Generate TLS RSA Private Key<br>Generate TLS RSA Public Key<br>Generate TLS Pre-master Secret(All Cases)<br>Generate TLS Master Secret/Traffic Keys <sup>1</sup> |
| Packet Forwarding <sup>2</sup> | Provides packet forwarding and receipt. Forwarded packets are encrypted and signed, and incoming packets are decrypted and verified | Crypto Officer<br>User | Use MANET Encryption Key<br>Use MANET Authentication Key  |

<sup>1</sup> “Traffic Keys” refer to the MANET Encryption Key and MANET Authentication Key listed in Table 8.

<sup>2</sup> This service is authorized on behalf of the Crypto Officer and User, though it does not require the operator to explicitly authenticate for its use. The Packet Forwarding service is only available once the operator has successfully authenticated and configured the MANET Encryption Key and MANET Authentication Key; thereafter, this service may be performed without the operator logging in, since it will only operate with peer devices that have also been configured with the same keys.

| Service              | Description   | Roles                  | CSPs   |
|----------------------|---|------------------------|--|
| Management           | Provides configuration and password management functions over TLS such as setting and deleting a password | Crypto Officer<br>User | Write/Use/Zeroize<br>Operator Passwords  |
| TLS                  | Establishes and maintains TLS connections   | Crypto Officer<br>User | Generate/Use/Zeroize<br>DRBG entropy input<br>Use/Zeroize DRBG V<br>Use/Zeroize DRBG<br>Key<br>Generate/Use/Zeroize<br>TLS Elliptic Curve<br>Diffie-Hellman Private<br>Key<br>Generate/Use/Zeroize<br>TLS Elliptic Curve<br>Diffie-Hellman Public<br>Key<br>Use/Zeroize TLS<br>Elliptic Curve Diffie-<br>Hellman Shared<br>Secret<br>Use TLS ECDSA<br>Private Key<br>Use TLS ECDSA Public<br>Key<br>Use TLS RSA Private<br>Key<br>Use TLS RSA Public<br>Key<br>Use/Zeroize TLS Pre-<br>master Secret<br>Use/Zeroize TLS<br>Master Secret/Traffic<br>Keys |
| Manage<br>MANET Keys | Generates MANET Encryption and Authentication Keys for encrypt/decrypt operations                         | Crypto Officer<br>User | Read/Write/Generate<br>/Zeroize MANET<br>Encryption Key<br>Read/Write/Generate<br>/Zeroize MANET<br>Authentication Key   |

| Service          | Description   | Roles   | CSPs  |
|------------------|---|---|---|
| Firmware Upgrade | Upgrade firmware to newer release<br>Note: If non-FIPS validated firmware is loaded, the module is no longer a FIPS validated module. | Crypto Officer<br>User                        | Use Firmware Upgrade Public Key<br><br>Use Firmware Decryption Key  |
| Self-Test        | Performs self-tests on critical functions of module   | Crypto Officer<br>User<br><br>Unauthenticated | Use Module Integrity Key  |
| Status           | Status of the module  | Crypto Officer<br>User<br><br>Unauthenticated | N/A   |
| Zeroize          | Zeroize keys and CSPs in the module   | Crypto Officer<br>User<br><br>Unauthenticated | Zeroize Operator Passwords<br><br>Zeroize MANET Encryption Key<br><br>Zeroize MANET Authentication Key<br><br>Zeroize TLS ECDSA Private Key<br><br>Zeroize TLS ECDSA Public Key<br><br>Zeroize TLS RSA Private Key<br><br>Zeroize TLS RSA Public Key<br><br>Zeroize Store Key |

**Table 7 – Operator Services and Descriptions**

The module does not support multiple concurrent operators. Each “view” or “set” of configuration by a user is a separate action, and the actual configuration is determined by the latest “set.” The Web GUI will indicate that a User/Crypto Officer role has logged themselves in. Only one operator can configure the module at one time. In the event that two authenticated sessions exist at one time for configuration, the module will save/store the parameters of the last operation. Concurrent sessions are treated as an individual session, but from separate end points.

## 2.4.2 Operator Authentication

Crypto Officer and User password must be a minimum of 8 characters. Legal password characters are the set of all 95 printable ASCII characters. This includes a-z, A-Z, 0-9, space, and these special characters: ! " # \$ % & ' ( ) \* + , - . / : ; < = > ? @ [ \ ] ^ \_ ` { | } ~. Passwords are case-sensitive. Given a random password of eight characters using the full character set, the probability of a successful random attempt is  $1/95^8$ , which is dramatically less than the  $1/1,000,000$  requirement. There is an explicit limit employed by the module to dramatically slow down the effective speed of an online brute force guessing attack. The system keeps tracks of recent failed attempts. If this count reaches ten, the system no longer accepts authentication attempts and the system reduces this count by one every ten seconds. As a result, a maximum of 16 guesses can be attempted in a one minute interval. This assumes that there are no failed guesses in the prior 100 seconds, ten guesses are made immediately at the beginning of the minute, and then followed by one guess every ten seconds for the remainder of the minute. Given a random password of eight characters using the full character set, this reduces the probability of success to  $16/95^8$ , which is dramatically less than  $1/100,000$  requirement.

## 2.5 Physical Security

The physical security of the cryptographic module meets FIPS 140-2 Level 2 requirements. The cryptographic module consists of production-grade components and includes an opaque enclosure protected by tamper evident seals. The physical boundary of the cryptographic module is the same as the physical boundary of the device.

The module does not include a maintenance interface; therefore, the FIPS-140-2 maintenance mode requirements do not apply.

## 2.6 Operational Environment

The module supports a non-modifiable operational environment. The module's firmware can only be updated with the verification of a digital signature over the firmware to be loaded. The loading of third party applications is procedurally controlled and is disabled by configuration per guidance provided in Section 3 of this Security Policy.

## 2.7 Cryptographic Key Management

The table below provides a complete list of Critical Security Parameters and Public Keys used within the module:

| Key/CSP Name             | Description / Use  | Generation / Establishment                       | Storage   | Import/Export  | Destruction                          |
|--------------------------|--|--|---|--|--------------------------------------|
| MANET Encryption Key     | AES CTR mode with 256-bit key for encryption / decryption of network traffic   | Internal generation by DRBG or imported via TLS  | <b>Storage:</b> Flash in encrypted form by the Store Key<br><br><b>Association:</b> The system is the one and only owner. Relationship is maintained by the operating environment via protected memory. | <b>Agreement:</b> NA<br><b>Entry:</b> via TLS<br><b>Output:</b> via TLS  | Destroyed by zeroizing the Store Key |
| MANET Authentication Key | Minimum key size of 256 bits. Maximum key size is the size of the block algorithm used. HMAC-SHA1 has a block size of 512. HMAC-SHA256 has a block size of 1024 bits. This key is used for message verification and integrity check. | Internal generation by DRBG or imported via TLS  | <b>Storage:</b> Flash in encrypted form by the Store Key  | <b>Agreement:</b> NA<br><b>Entry:</b> via TLS<br><b>Output:</b> via TLS  | Destroyed by zeroizing the Store Key |
| Module Integrity Key     | HMAC SHA-256 key for verifying the integrity of the module. Fixed string of 43 characters.   | Not generated by the module; built into firmware | <b>Storage:</b> Flash in plaintext<br><br><b>Type:</b> Static   | <b>Agreement:</b> NA<br><br><b>Entry:</b> FW upgrade encrypted by Firmware Decryption Key and TLS<br><br><b>Output:</b> NA | Replaced during FW upgrade           |
| Firmware Decryption Key  | AES CTR 256-bit key for decryption of firmware before upgrade  | Not generated by the module; built into firmware | <b>Storage:</b> Flash in plaintext<br><br><b>Type:</b> Static   | <b>Agreement:</b> NA<br><br><b>Entry:</b> FW upgrade encrypted by itself and TLS<br><br><b>Output:</b> NA                  | Replaced during FW upgrade           |

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

| Key/CSP Name                | Description / Use   | Generation / Establishment   | Storage  | Import/Export   | Destruction                          |
|-----------------------------|---|--|--|---|--------------------------------------|
| Firmware Upgrade Public Key | RSA 15360-bit key for verifying firmware signature before upgrading   | Not generated by the module; built into firmware   | <b>Storage:</b> Flash in plaintext<br><b>Type:</b> Static  | <b>Agreement:</b> NA<br><b>Entry:</b> FW upgrade encrypted Firmware Decryption Key and TLS<br><b>Output:</b> NA | Replaced during FW upgrade           |
| Operator Passwords          | Alphanumeric passwords externally generated by a human user for authentication.   | Not generated by the module; imported by the human operator  | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> via TLS.<br><b>Output:</b> NA   | Destroyed by zeroizing the Store Key |
| Store Key                   | AES CBC 256-bit key for encryption of Flash data store  | Internal generation by DRBG  | <b>Storage:</b> Flash (without FTL) in plaintext           | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA   | Zeroize                              |
| DRBG entropy input          | 960-bits of input from the NDRNG. Expected entropy is significantly greater than 512 bits.                                  | Hardware based entropy source used to construct seed   | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA   | Zeroized after use                   |
| DRBG V                      | The DRBG V consists of 512-bits and is part of the internal state upon which the security of this DRBG mechanism depends.   | Generated first during DRBG instantiation and then subsequently updated using the DRBG update function | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA   | Zeroized after use                   |
| DRBG Key                    | The DRBG Key consists of 512-bits and is part of the internal state upon which the security of this DRBG mechanism depends. | Generated first during DRBG instantiation and then subsequently updated using the DRBG update function | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA   | Zeroized after use                   |

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

| Key/CSP Name                                    | Description / Use   | Generation / Establishment             | Storage  | Import/Export   | Destruction                          |
|---|---|--|--|---|--------------------------------------|
| RSA CA Public Key                               | RSA Public 2048-bit certificate signature   | Internal generation by DRBG            | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> via TLS                                  | Destroyed by zeroizing the Store Key |
| RSA CA Private Key                              | RSA Private 2048-bit certificate signature  | Internal generation by DRBG            | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                                       | Zeroized after use                   |
| ECDSA CA Public Key                             | ECDSA Public P-384 certificate signature  | Internal generation by DRBG            | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> via TLS                                  | Destroyed by zeroizing the Store Key |
| ECDSA CA Private Key                            | ECDSA Private P-384 certificate signature   | Internal generation by DRBG            | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                                       | Zeroized after use                   |
| TLS Elliptic Curve Diffie-Hellman Shared Secret | The shared secret used in Elliptic Curve Diffie-Hellman (ECDH) exchange. The size of the shared secret is 384-bits. | Established per the ECDH key agreement | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                                       | Zeroized after use                   |
| TLS Elliptic Curve Diffie-Hellman Private Key   | The private key used in Elliptic Curve Diffie-Hellman (ECDH) exchange. Using the P-384 curve.                       | Internal generation by DRBG            | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                                       | Zeroized after use                   |
| TLS Elliptic Curve Diffie-Hellman Public Key    | The public key used in Elliptic Curve Diffie-Hellman (ECDH) exchange. Using the P-384 curve.                        | Internal generation by DRBG            | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> Part of TLS handshake<br><b>Output:</b> Part of TLS handshake | Zeroized after use                   |

FIPS 140-2 Non-Proprietary Security Policy: Embedded Module and Embedded Module Lite

| Key/CSP Name                              | Description / Use  | Generation / Establishment   | Storage  | Import/Export  | Destruction                                  |
|---|--|--|--|--|--|
| TLS ECDSA Private Key                     | Signature generation. Using the P-384 curve.                                     | Internal generation by DRBG  | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                    | Destroyed by zeroizing the Store Key         |
| TLS ECDSA Public Key                      | Signature verification. Using the P-384 curve.                                   | Internal generation by DRBG  | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> Part of TLS handshake | Destroyed by zeroizing the Store Key         |
| TLS ECDSA Public Key of the remote server | To authenticate the TLS key agreement. Using the P-384 curve.                    | Received as part of the TLS handshake when using ECDSA cipher suites | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> Part of TLS handshake<br><b>Output:</b> NA | Automatically when TLS session is terminated |
| TLS RSA Private Key                       | Identity certificates used in TLS negotiations. 2048 bits in size.               | Internal generation by DRBG  | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> NA                    | Destroyed by zeroizing the Store Key         |
| TLS RSA Public Key                        | Identity certificates used in TLS negotiations. 2048 bits in size.               | Internal generation by DRBG  | <b>Storage:</b> Flash in encrypted form by the Store Key   | <b>Agreement:</b> NA<br><b>Entry:</b> NA<br><b>Output:</b> Part of TLS handshake | Destroyed by zeroizing the Store Key         |
| TLS RSA Public Key of the remote server   | To encrypt the TLS Pre-master Secret using RSA Key Transport. 2048 bits in size. | Received as part of the TLS handshake when using RSA cipher suites   | <b>Storage:</b> RAM in plaintext<br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><b>Entry:</b> Part of TLS handshake<br><b>Output:</b> NA | Automatically when TLS session is terminated |



| Key/CSP Name          | Description / Use  | Generation / Establishment  | Storage  | Import/Export  | Destruction                                  |
|-----------------------|--|---|--|--|--|
| TLS Pre-master Secret | Used to derive the TLS Master Secret and session keys, 384 bits in size    | Establishment depends on cipher suite used and client/server role.<br><br><b>RSA client:</b><br>Internal generation by DRBG<br><br><b>RSA server:</b><br>Decrypted using the TLS RSA Private Key<br><br><b>ECDH:</b><br>Established by ECDH Key Agreement | <b>Storage:</b> RAM in plaintext<br><br><b>Type:</b> Ephemeral | Import/Export depends on cipher suite used and client/server role.<br><br><b>RSA client:</b><br>Output encrypted by the RSA Public Key of the server via RSA Key Transport<br><br><b>RSA server:</b><br>Entered encrypted by the RSA Public Key of the server via RSA Key Transport<br><br><b>ECDH:</b> N/A. Established by ECDH Key Agreement | Automatically when TLS session is terminated |
| TLS Master Secret     | Used in TLS connections to derive session keys. 384 bits in size.          | Established using TLS protocol. This key was derived in the module.   | <b>Storage:</b> RAM in plaintext<br><br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><br><b>Entry:</b> NA<br><br><b>Output:</b> NA  | Automatically when TLS session is terminated |
| TLS Encryption Key    | AES 128, 192, or 256-bit keys in GCM or CBC mode. Used in TLS connections. | Established using TLS protocol. This key was derived in the module.   | <b>Storage:</b> RAM in plaintext<br><br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><br><b>Entry:</b> NA<br><br><b>Output:</b> NA  | Automatically when TLS session is terminated |
| TLS Integrity Key     | HMAC-SHA-1. Used in TLS connections. 160 bits in size.                     | Established using TLS protocol. This key was derived in the module.   | <b>Storage:</b> RAM in plaintext<br><br><b>Type:</b> Ephemeral | <b>Agreement:</b> NA<br><br><b>Entry:</b> NA<br><br><b>Output:</b> NA  | Automatically when TLS session is terminated |

**Table 8 – Key/CSP Management Details (also includes public keys)**

Network Keys can be exported from the physical boundary of the module when the Crypto Officer re-keys the module using the network management feature. The Network Key will be sent to other nodes (modules) on the network encrypted with TLS.

All persistent keys and CSPs are stored in an encrypted store. This store is located in eMMC and is encrypted via an AES 256-bit key (Store Key). The key & IV used to encrypt the store are stored in a

separate flash without FTL. Zeroization has been implemented to ensure no traces are left of the store key & IV. Zeroization is achieved by explicitly erasing the flash sector, containing the key and IV material. The erase operation is at the hardware level and writes a specific value to flash. The Embedded Module can be zeroized by switching the zeroized pin, in the control port, to ground (requires main power to be connected to a power source), or via the management interface by an authorized role (requires unit to be on and operational).

## 2.8 Self-Tests

The module includes an array of self-tests that are run during startup and periodically during operations to prevent secure data from being released and to ensure all components are functioning correctly. In the event of any self-test failure, the module will restart. Self-test Success status is indicated by the status LED as well as via HTTPS. No keys or CSPs will be output when the module is in an error state.

If the self-tests succeed, the operator will be presented with a login screen when accessing the module via HTTPS. Attempts to access it via HTTP will be automatically redirected to HTTPS. If the self-tests fail, any attempt to access the module will fail.

The self-tests are always run. On failure, the module will always be non-operational.

The following sections discuss the module’s self-tests in more detail.

### 2.8.1 Power-On Self-Tests

Power-on self-tests are run upon every initialization of the module and if any of the tests fail, the process will be halted and the module will not initialize. In this error state, no services can be accessed by the users. The module implements the following power-on self-tests:

| Hardware Implementation (Cryptographic Engine) |   |
|--|---|
| Test Target                                    | Description   |
| AES  | KATs: Encryption, Decryption<br>Modes: CBC, CTR, ECB<br>Key sizes: 128-bits, 192-bits, 256-bits |
| SHS  | KATs: Output Verification<br>SHA sizes: SHA-1, SHA-224, SHA-256                                 |
| HMAC   | KATs: Generation, Verification<br>SHA sizes: SHA-1, SHA-224, SHA-256                            |

Table 9 – Cryptographic Engine POST

| Firmware Implementation (Cryptographic Kernel) |  |
|--|--|
| Test Target                                    | Description  |
| AES  | KATs: Encryption, Decryption<br>Modes: CBC, CTR, ECB, GCM<br>Key sizes: 128-bits, 192-bits, 256-bits |
| AES  | KATs: Encryption, Decryption<br>Mode: XTS<br>Key sizes: 128-bits, 256-bits                           |
| SHS  | KATs: Output Verification<br>SHA sizes: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512                    |
| HMAC   | KATs: Generation, Verification<br>SHA sizes: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512               |

Table 10 – Cryptographic Kernel POST

| Firmware Implementation (Cryptographic Library) |  |
|---|--|
| Test Target                                     | Description  |
| AES   | KATs: Encryption, Decryption<br>Modes: ECB<br>Key sizes: 128-bits                          |
| AES   | KATs: Encryption, Decryption<br>Mode: XTS<br>Key sizes: 128-bits, 256-bits                 |
| DSA   | PCT: Signature Generation, Signature Verification<br>Key size: 2048-bits                   |
| DRBG  | KATs: HASH_DRBG, HMAC_DRBG, CTR_DRBG<br>Security Strengths: 256-bits                       |
| ECDSA   | PCTs: Key Generation, Signature Generation, Signature Verification<br>Curves: P-224, K-233 |
| ECC CDH   | Shared secret calculation per SP 800-56A §5.7.1.2, IG 9.6<br>Curve: P-224                  |
| ECC KAS   | KAT is performed by ECC CDH KAT  |
| GCM   | KATs: Encryption, Decryption, Generation, Verification<br>Key sizes: 256-bits              |
| RSA   | KATs: Signature Generation, Signature Verification<br>Key sizes: 2048-bits                 |

| Firmware Implementation (Cryptographic Library) |  |
|---|--|
| Test Target                                     | Description  |
| SHS   | KATs: Output Verification<br>SHA sizes: SHA-1  |
| HMAC  | KATs: Generation, Verification<br>SHA sizes: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 |

**Table 11 – Cryptographic Library POST**

The module performs all power-on self-tests automatically when it is initialized. The module also verifies its integrity using HMAC-SHA256. Successful completion of self-tests will be indicated via HTTPS. All power-on self-tests must be passed before a User/Crypto Officer can perform services. The Power-on self-tests can be run on demand by restarting the module.

### 2.8.2 Conditional Self-Tests

Conditional self-tests are run continuously when certain conditions are met during operation of the module. The module performs the following conditional self-tests:

| Conditional Self-Tests             |                                    |                             |
|------------------------------------|------------------------------------|-----------------------------|
| Test Target                        | Condition                          | Description                 |
| RSA                                | On each generation of a key pair   | Pairwise consistency test   |
| DSA                                | On each generation of a key pair   | Pairwise consistency test   |
| ECDSA                              | On each generation of a key pair   | Pairwise consistency test   |
| DRBG                               | On output of DRBG implementation   | Continuous test             |
| NDRNG                              | On output of NDRNG (seed for DRBG) | Continuous test             |
| RSA digital signature verification | Firmware Load / Firmware Upgrade   | Signature verification test |
| DRBG                               | SP800-90A Health Tests             | Health Checks               |

**Table 12 – Conditional Self-Tests**

Note that the module performs conditional tests for firmware and software implementations of the algorithms listed in the Algorithm Implementation Certificates section. If any of these tests fail, the module will enter an error state. The module can be re-initialized to clear the error and resume FIPS mode of operation. While in an error state, no services can be accessed by the operators.

## 2.9 EMI/EMC

The module is designed as a component of a radio, which meets Federal Communications Commission (FCC) FCC Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) requirements for a radio.

## **2.10 Mitigation of Other Attacks**

The module does not mitigate other attacks.

## 3 Guidance and Secure Operation

This section describes how to configure the module for FIPS-approved mode of operation. Operating it without maintaining the following settings will violate the FIPS-approved mode of operation.

### 3.1 Crypto Officer and User Guidance

#### 3.1.1 Initialization for FIPS Mode of Operation

The Crypto Officer or User must configure and enforce the following procedures to maintain the FIPS 140-2 Level 2 configuration:

1. When setting the password, the Crypto Officer or User must specify a password with a minimum length of eight legal characters, which is enforced by the module. Legal password characters are the set of all 95 printable ASCII characters. This includes a-z, A-Z, 0-9, space, and these special characters: !"#%&'()\*+,-./:;<=>?@[\\]^\_`{|}~.

Note: Stronger, more secure passwords should have a combination of letters and numbers and should not contain any recognizable words that may be found in a dictionary. The module does not enforce this; the Crypto Officer or User must follow his/her organization's systems security policies and adhere to the password policies set forth therein.

2. Ensure FW version running is listed in section 2.2 of this document.
3. Ensure User Application loading is disabled. From the management interface, select the "Security" tab and confirm "User App Install" is set to "Disabled"; otherwise, select to "Disable User App Installs". The module will zeroize and erase any installed User Apps when enabling/disabling this option.

#### 3.1.2 General Crypto Officer and User Guidance

After initialization for FIPS mode, the Crypto Officer and User should follow the guidance below:

1. When entering a network key over the configuration GUI, the operator must ensure the key was generated by FIPS-approved methods and that the key was not previously used.
2. The operator must ensure that all Radio MAC addresses used in a network are unique.
3. The Crypto Officer or User must not disclose passwords and must store passwords in a safe location and according to his/her organization's systems security policies for password storage.