Ciena Corporation

Ciena 6500 Flex3 WaveLogic 3e OCLD Encryption Module

Hardware Versions: 2.0, 2.1 and 2.2 with PCB P/N NTK539QS-220
Firmware Versions: 2.01 and 3.60

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level: 2
Document Version: 2.0

Prepared by/for:

Ciena Corporation
7035 Ridge Road

Hanover, Maryland 21076
United States of America

Phone: +1 410 694 5700
www.ciena.com

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Purpose

This is a non-proprietary Cryptographic Module Security Policy for the Ciena 6500 Flex3 WaveLogic 3e (WL3e) OCLD[1] Encryption Module (hardware versions: 2.0, 2.1 and 2.2 with PCB part number NTK539QS-220; firmware versions: 2.01 and 3.60). This Security Policy describes how the Ciena 6500 Flex3 WaveLogic 3e (WL3e) OCLD Encryption Module meets the security requirements of Federal Information Processing Standards (FIPS) Publication 140-2, which details the U.S. and Canadian Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the National Institute of Standards and Technology (NIST) and the Communications Security Establishment (CSE) Cryptographic Module Validation Program (CMVP) website at http://csrc.nist.gov/groups/STM/cmvp.

This document also describes how to run the module in a secure FIPS-Approved mode of operation. This policy was prepared as part of the Level 2 FIPS 140-2 validation of the module. The Ciena 6500 Flex3 WaveLogic 3e OCLD Encryption Module is referred to in this document as the WL3e Encryption Module or the module.

1.2 References

This document deals only with operations and capabilities of the module in the technical terms of a FIPS 140-2 cryptographic module security policy. More information is available on the module from the following sources:

- The Ciena website (www.ciena.com) contains information on the full line of products from Ciena.
- The CMVP website (http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm) contains contact information for individuals to answer technical or sales-related questions for the module.

[1] OCLD – Optical Channel Laser and Detector

2. WL3e Encryption Module

2.1 Overview

As network traffic demands and unpredictability grow, requirements for next-generation networks are rapidly increasing in scope. Ciena has developed its WaveLogic 3 coherent optical technology to help transport systems adapt and meet these requirements. Ciena's customized solution, the 6500 Flex3 WaveLogic 3e OCLD Encryption Module with the WL3e chipset, provides the capacity and flexibility necessary to adapt to unpredictable service growth.

The WL3e is a programmable chipset that provides two modulation schemes:

- Extreme 16QAM[2] – Provides double the capacity and spectral efficiency of 100Gb/s with 200Gb/s per wavelength for all applications.
- Extreme QPSK[3] – Provides strong performance of 100 Gb/s per wavelength for most long-haul and transatlantic submarine distances; also provides enhanced non-linear mitigation for best performance alongside 10G channels.

The WL3e is implemented as components on a circuit board. All traffic entering and exiting the circuit board (also called the "circuit pack") is encrypted/decrypted at wire-speed using AES-256 Counter mode. To provide secure cryptographic services, the circuit pack contains the embedded 6500 Flex3 WL3e OCLD Encryption Module. The WL3e Encryption Module is composed of a Krypto Module (KM) daughtercard, an ASIC[4], the PCB-embedded wire connections between them, and all associated physical security mechanisms (defined in Section 2.5 and illustrated in Section 3.1). The KM (part number NTK53926-501 for Hardware Ver 2.0; part number NTK53926-502 for Hardware Ver 2.1 and Ver 2.2, including an aluminum enclosure) provides the certificate management, Crypto Officer (CO) and User authentication, peer authentication, and key derivation functions of the module. The ASIC (part number 077-0084-017 for Hardware Ver 2.0; part number 077-0084-028 for Hardware Ver 2.1 and Ver 2.2; part number 077-0084-029 for Hardware Ver 2.2) features two data path engines to encrypt/decrypt all Optical channel Data Unit (ODU) 4 traffic. The WL3e Encryption Module is part of the WaveLogic 3 Encryption OCLD circuit pack of the 6500 series Packet-Optical Platform.

The module as it appears on the circuit pack can be seen in Figure 1, while Figure 3 provides the module's block diagram. Both figures surround the module's cryptographic boundary with a dotted red line.

[2] QAM – Quadrature Amplitude Modulation
[3] QPSK – Quadrature Phase Shift Keying
[4] ASIC – Application-Specific Integrated Circuit

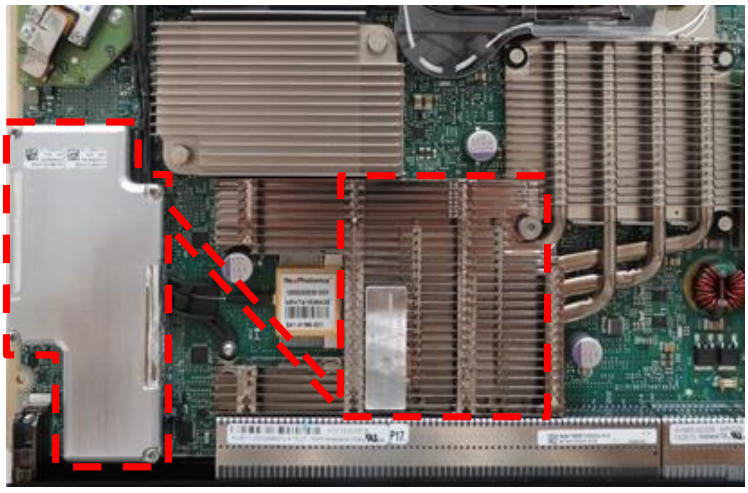

Figure 1 – Module on Circuit Pack (Top View)

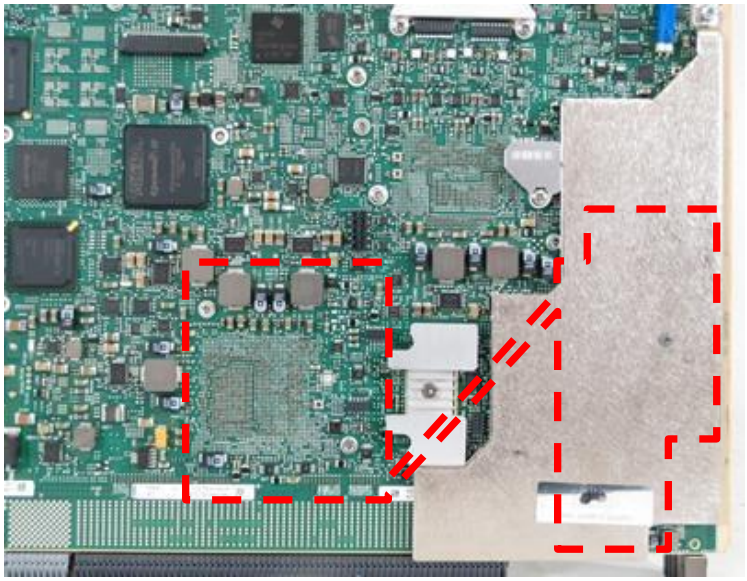

Figure 2 – Module on Circuit Pack (Bottom View)

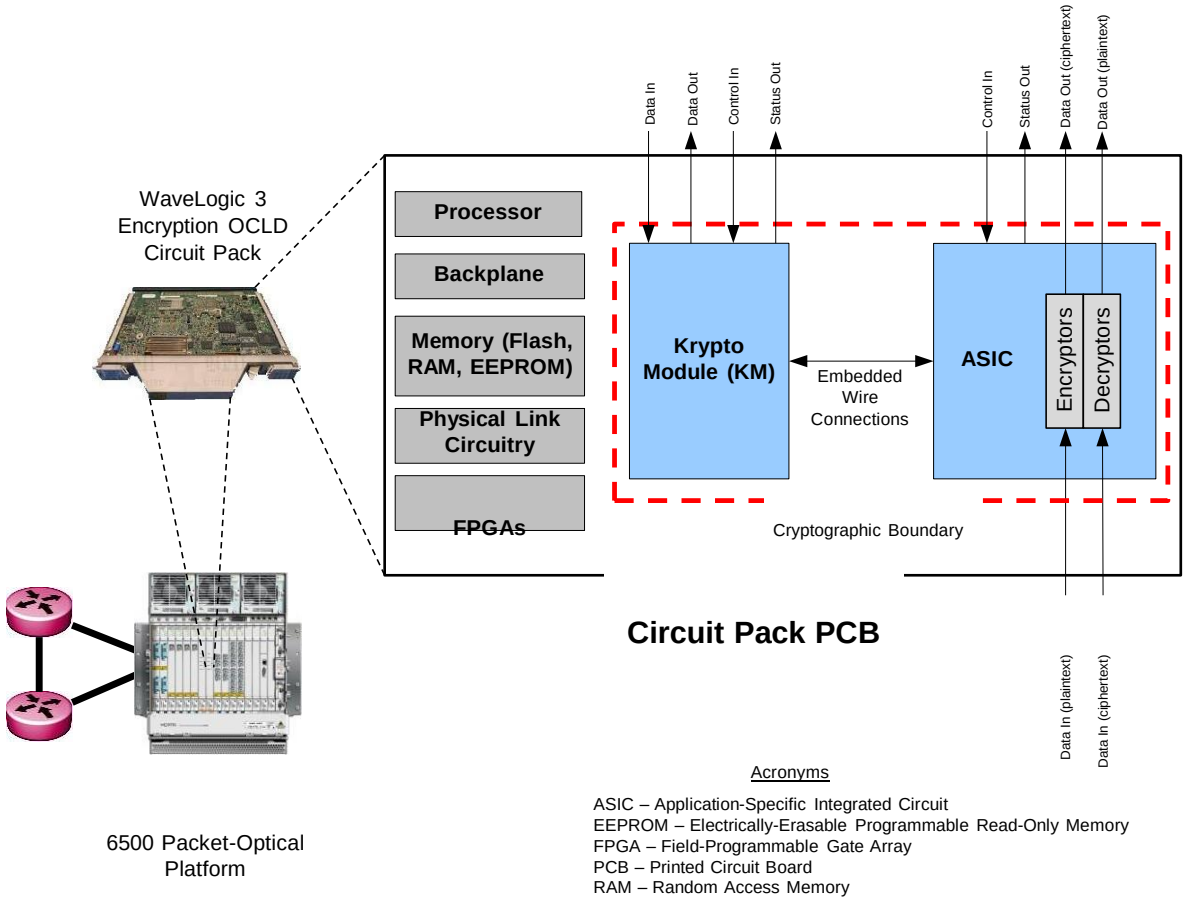

Figure 3 – Module Block Diagram

The WL3e Encryption Module is validated at the FIPS 140-2 Section levels shown in Table 1.

Table 1 – Security Level per FIPS 140-2 Section

Section	Section Title	Level
1	Cryptographic Module Specification	2
2	Cryptographic Module Ports and Interfaces	2
3	Roles, Services, and Authentication	3
4	Finite State Model	2
5	Physical Security	2
6	Operational Environment	N/A
7	Cryptographic Key Management	2

Section	Section Title	Level
8	EMI/EMC[5]	2
9	Self-tests	2
10	Design Assurance	3
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The WL3e Encryption Module is a hardware module with a multiple-chip embedded embodiment. The module consists of two primary components: a KM enclosed in an aluminum enclosure and an ASIC mounted on the motherboard's PCB and covered by a heatsink. These two components communicate via wire connections embedded beneath multiple PCB layers. The KM also contains integrated circuits, processors, Synchronous Dynamic Random Access Memory (SDRAM), flash memories (NOR[6] and EEPROM), and FPGAs[7].

The overall security level of the module is 2. The cryptographic boundary of the WL3e Encryption Module surrounds the KM, ASIC, the portion of the PCB under which the connecting wire traces are embedded, and all physical security mechanisms described in Section 2.5.

The WL3e Encryption Module implements the FIPS-Approved algorithms listed in Table 2 below.

Table 2 – FIPS-Approved Algorithm Implementations

Algorithm	Certificate Number	
	ASIC	KM Firmware
AES[8]-CTR[9] mode with 256-bit keys	# 4231, # 5241, C 651	-
AES-ECB[10] mode (encryption) with 256-bit keys	# 4231, # 5241, C 651	-
AES-CBC[11] mode with 128, 192, and 256-bit keys	-	# 4232, C 649
AES-GCM[12] mode with 128 and 256-bit keys	-	# 4232, C 649
Triple-DES[13]-CBC mode (3-key)	-	# 2291, C 649
SHA[14]-1, SHA-256, SHA-384, and SHA-512	-	# 3469, C 649
SHA-384	# 3468, # 4219, C 650	-
HMAC[15] with SHA-1, SHA-256, SHA-384, and SHA-512	-	# 2770, C 649

[5] EMI/EMC – Electromagnetic Interference / Electromagnetic Compatibility
[6] NOR – Not Or
[7] FPGA – Field Programmable Gate Array
[8] AES – Advanced Encryption Standard
[9] CTR – Counter
[10] ECB – Electronic Code Book
[11] CBC – Cipher Block Chaining
[12] GCM – Galois/Counter Mode
[13] DES – Data Encryption Standard
[14] SHA – Secure Hash Algorithm
[15] HMAC – (Keyed) Hash Message Authentication Code

Algorithm	Certificate Number	
	ASIC	KM Firmware
NIST SP[16] 800-90A CTR_DRBG[17]	-	# 1315, C 649
ECDSA[18] PKG[19] with NIST-defined P-curves P-224, P-256, P-384, and P-521	-	# 977, C 649
ECDSA PKV[20] with NIST-defined P-curves P-192, P-224, P-256, P-384, and P-521	-	# 977, C 649
ECDSA signature generation with NIST-defined P-curves P-224 (SHA-256, 384, and 512), P-256 (SHA-256, 384, and 512), P-384 (SHA-256, 384, and 512), and P-521 (SHA-256, 384, and 512)	-	# 977, C 649
ECDSA signature verification with NIST-defined P-curves P-192 (SHA-1, 256, 384, and 512), P-224 (SHA-1, 256, 384, and 512), P-256 (SHA-1, 256, 384, and 512), P-384 (SHA-1, 256, 384, and 512), and P-521 (SHA-1, 256, 384, and 512)	-	# 977, C 649
ECDSA signature verification with NIST-defined P-curve P-384 with SHA-384	# 976, # 1363, C 650	-
Section 4.2 TLS[21] v1.2 (NIST SP 800-135)	-	# 980, C 649
Section 4.1.1 IKE[22] v1 (NIST SP 800-135)	-	# 980, C 649
Section 4.1.2 IKE v2 (NIST SP 800-135)	-	# 980, C 649

NOTE: The TLS and IKE protocols have not been reviewed or tested by the CAVP or CMVP.

Additionally, the module implements the following algorithms that are allowed for use in a FIPS-Approved mode of operation:

- Non-Deterministic Random Number Generator (NDRNG)
- Elliptic Curve Diffie-Hellman[23] with NIST-defined P-curve P-384

2.3 Module Interfaces

The module's design separates the physical ports into four logically distinct and isolated interface categories. They are:

- Data Input Interface
- Data Output Interface
- Control Input Interface
- Status Output Interface

Data input/output consists of the data utilizing the services provided by the module. Control input consists of configuration or administration data entered into the module. Status output consists of signals output that are then translated into alarms, LED signals, and log information by the circuit pack.

[16] SP – Special Publication
[17] DRBG – Deterministic Random Bit Generator
[18] ECDSA – Elliptic Curve Digital Signature Algorithm
[19] PKG – Public Key Generation
[20] PKV – Public Key Validation
[21] TLS – Transport Layer Security
[22] IKE – Internet Key Exchange
[23] Caveat: EC Diffie-Hellman (key agreement; key establishment methodology provides 192 bits of encryption strength). Please see NIST Special Publication 800-131A for further details.

The physical ports and interfaces of the WL3e Encryption Module consist of the mezzanine connector and ASIC pin-outs, and are depicted in Figure 4 and Figure 5. Figure 4 shows the KM; Figure 5 shows the ASIC.

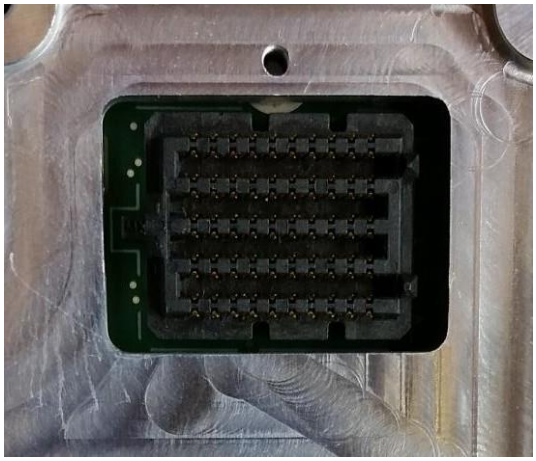

Figure 4 – KM Mezzanine Connector

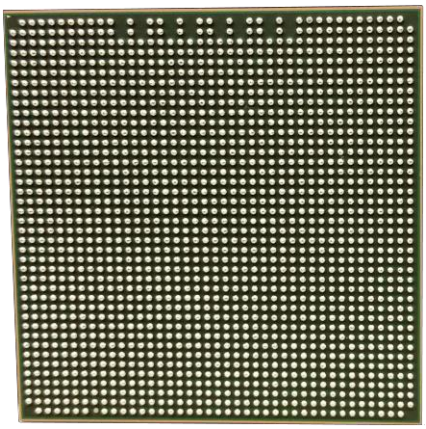

Figure 5 – ASIC Pin-Outs

Table 3 lists the physical ports and interfaces available in the WL3e Encryption Module, and provides the mapping from the physical ports and interfaces to logical interfaces as defined by FIPS 140-2. Interfaces are provided by both the KM and the ASIC. Note that the ASIC pins are categorized into the following groupings (with associated pin counts):

- Backplane Data In (40 pins)
- Backplane Data Out (40 pins)
- Line Data In (8 pins)
- Line Data Out (44 pins)
- Control In (52 pins)
- Status Out (59 pins)
- Power In (342 pins)

Table 3 – Logical Interface Mapping

FIPS 140-2 Logical Interface	Module Interface
Data Input Interface	KM mezzanine connector, ASIC Backplane Data In pins, ASIC Line Data In pins
Data Output Interface	KM mezzanine connector, ASIC Backplane Data Out pins, ASIC Line Data Out pins
Control Input Interface	KM mezzanine connector, ASIC Control In pins
Status Output Interface	KM mezzanine connector, ASIC Status Out pins
Power Interface	KM mezzanine connector, ASIC Power In pins

The ASIC also includes the following pin groupings that, based upon their purpose, are not mapped into the FIPS logical interface categories:

- General Purpose I/O[24] pins (provide interfaces for pre-installation scan testing; unused once installed)
- Internal Control/Status I/O pins (provide internal interfaces between module components)
- Ground pins

2.4 Roles, Services, and Authentication

The following sections described the authorized roles supported by the module, the services provided for those roles, and the authentication mechanisms employed.

2.4.1 Authorized Roles

The module supports two authorized roles: a CO role and a User role. The CO and the User roles are responsible for module initialization and module configuration, including security parameters, key management, status activities, and audit review.

The module offers two management interfaces:

- MyCryptoTool Interface – used for security-related configuration and management of the module.
- TCS Interface – used for non-security-related configuration and carrier provisioning of the module and also firmware loads.

While operators must assume an authorized role to access most module services, there are a limited number of services for which the operator is not required to assume an authorized role. Operators explicitly assume both the CO and User role by a mutually-authenticated HTTPS/TLS session over MyCryptoTool using digital certificates. Operators explicitly assume the CO role over the TCS interface using a username and password credential in the form of a preshared HMAC-SHA-256 authentication string.

2.4.2 Services

The services that require operators to assume an authorized role are listed in Table 4 below. Please note that the keys and Critical Security Parameters (CSPs) listed in Table 4 use the following indicators to show the type of access required:

[24] I/O – Input/Output

- R – Read: The CSP is read.
- W – Write: The CSP is established, generated, modified, or zeroized.
- X – Execute: The CSP is used within an Approved or Allowed security function or authentication mechanism.

Table 4 – Authorized Operator Services

Service	Operator		Description	Input	Output	CSP and Type of Access
	CO	User				
Initialize the module	✓	✓	Initialize the module	Command	Status output	None
Configure the module using MyCryptoTool	✓	✓	Configure enterprise settings and Import certificates	Command and parameters	Command response	CA ECDSA Public Key – R/X MKEK[25] – R/X KEK[26] – R/X
Monitor alarms	✓	✓	Monitor specific alarms for diagnostic purposes	Command	Status output	None
Manage data encryption certificate	✓	✓	Manage data encryption certificate enrollment, signing CA[27] certificate information, trusted CA certificates; Import CA certificate and CRL[28]; Clear CSPs	Command and parameters	Command response	BKEK[29] – R/X DEK[30] – R/W MKEK – R/X KEK – R/X CA ECDSA Public Key – R/X
Manage web access certificate and import CRL	✓	✓	Manage web access certificate and import CRL	Command and parameters	Command response	BKEK – R/X MKEK – R/X KEK – R/X CA ECDSA Public Key – R/X
Show FIPS status and statistics	✓	✓	Show the system status, FIPS-Approved mode, configuration settings, and active alarms.	Command	Status output	None
View system logs	✓	✓	View system status messages in historical alarm log and provisioning log.	Command	Status output	None
Zeroize using MyCryptoTool	✓	✓	Zeroize the keys and CSPs listed in the 'Zeroization' column in Table 7 below	Command	Command response	Please see the 'Zeroization' column in Table 7 below.
Employ encryption / decryption service	✓	✓	Encrypt or decrypt user data, keys, or management traffic	Command and parameters	Command response	MKEK – X DEK – X TLS Session Key – X

[25] MKEK – Master Key Encryption Key
[26] KEK – Key Encryption Key
[27] CA – Certificate Authority
[28] CRL – Certificate Revocation List
[29] BKEK – Base Key Encryption Key
[30] DEK – Data Encryption Key

Service	Operator		Description	Input	Output	CSP and Type of Access
	CO	User				
Authenticate management traffic	✓	✓	Authenticate management traffic	Command and parameters	Command response	TLS Authentication Key – X
Generate asymmetric key pair (data path)	✓	✓	Generate the asymmetric key pair (ECDSA) for data path encryption	Command and parameters	Key pair	Module Data Path ECDSA Private Key – W Module Data Path ECDSA Public Key – W
Generate asymmetric key pair (web access)	✓	✓	Generate the asymmetric key pair (ECDSA)	Command and parameters	Key pair	Module Web Access ECDSA Private Key – W Module Web Access ECDSA Public Key – W
Generate signature (Certificate Signing Request)	✓	✓	Generate a signature for the supplied message using specified key and ECDSA algorithm	Command and parameters	Status, signature	Module ECDSA Private Key – R/X
Verify signature	✓	✓	Verify the signature on the supplied message using the specified key and ECDSA algorithm	Command and parameters	Status	Module ECDSA Public Key – R/X
Perform device diagnostics	✓	✓	Test the module during operation; Monitor the module	Command and parameters	Command response and status via log and LEDs	None
Upgrade KM application firmware	✓		Upgrade the KM application firmware using ECDSA signature verification	Command and parameters	Command response and status output	ECDSA Public Key – R/X
Upgrade KM FPGA	✓		Upgrade the KM FPGA using ECDSA signature verification	Command and parameters	Command response and status output	ECDSA Public Key – R/X

In FIPS-Approved mode, the module provides a limited number of services for which the operator is not required to assume an authorized role (see Table 5). None of the services listed in the table disclose cryptographic keys and CSPs or otherwise affect the security of the module.

Table 5 – Additional Services

Service	Description	Input	Output	CSP and Type of Access
Perform operator authentication	Authenticate operators to the module	Command	Status output	CO ECDSA Public key – R/X User ECDSA Public key – R/X CA ECDSA Public Key – R/X Preshared Authentication String – R/X
Perform peer authentication	Authenticate peer devices to the module	Command	Status output	Peer ECDSA Public key – R/X
Zeroize using TCS	Zeroize the keys and CSPs listed in the 'Zeroization' column in Table 7 below	Command	Command response	Please see the 'Zeroization' column in Table 7 below.
Perform on-demand self-tests	Perform Power-up Self-Tests on demand via module restart	Power button on the host system or command	Status output	All plaintext keys and CSPs – W
Show system status and statistics using TCS	Show the system status, system identification, and configuration settings of the module	Command	Status output	None
Configure the module using TCS	Configure and manage the carrier provisioning	Command	Response and status output	None

2.4.3 Authentication

The module supports identity-based authentication. Module operators must authenticate to the module before being allowed access to services that require the assumption of an authorized role. The module authenticates an operator using digital certificates containing the public key of the operator. The authentication is achieved by initiating a TLS session and using digital certificates for mutual authentication. The process of mutual authentication provides assurance to the module that it is communicating with an authenticated operator. The strength calculation below provides minimum strength based on the public key size in the digital certificates.

The module employs the authentication methods described in Table 6 to authenticate COs and Users.

Table 6 – Authentication Mechanism

Authentication Type	Strength
Public Key Certificates	The module supports ECDSA digital certificate authentication of COs and Users during MyCryptoTool access. Using conservative estimates and equating the use of ECDSA with the P-384 elliptic curve to a 192-bit symmetric key, the probability for a random attempt to succeed is: *1:2^{192} or 1: 6.28 x 10^{51}* which is less than 1:1,000,000 as required by FIPS 140-2. The fastest network connection supported by the modules over Management interfaces is 5 Mb/s[31]. Hence, at most (5 ×10^6 × 60 = 3 × 10^8 =) 300,000,000 bits of data can be transmitted in one minute. Therefore, the probability that a random attempt will succeed or a false acceptance will occur in one minute is: *1: (2^{192} possible keys / ((3×10^8 bits per minute) / 192 bits per key))* *1: (2^{192} possible keys / 1,562,500 keys per minute)* *1: 4.02 × 10^{51}* which is less than 1:100,000 within one minute as required by FIPS 140-2.
Preshared Key	The module supports the use of a preshared authentication string for the TCS interface accessing the module on behalf of the CO. An HMAC-SHA-256 operation with a 512-bit key is performed on the preshared authentication string. The 256-bit output value of HMAC-SHA-256 will have an equivalent symmetric key strength of 128 bits, Using conservative estimates, the probability for a random attempt to succeed is: *1:2^{128} or 1: 3.40 × 10^{38}* which is less than 1:1,000,000 (as required by FIPS 140-2). The module implements a 200 ms[32] delay between authentication attempts yielding a rate of five attempts per second, or 300 attempts per minute. Given that an attacker will have, at most, 300 attempts in one minute, and there are 1: 3.40 × 10^{38} possibilities, the probability that a random attempt will succeed or a false acceptance will occur in one minute is: *1: 3.40 × 10^{38} / 300 attempts per minute* *1: 1.13 x 10^{36}* which is less than 1:100,000 within one minute (as required by FIPS 140-2).

The module also performs authentication of peers using public key certificates, but the module does not provide any authenticated services to peers.

2.5 Physical Security

All CSPs are stored and protected within the WL3e Encryption Module's components using the following physical security mechanisms, which provide opacity and tamper evidence:

- The wire connections that provide the communications path between the KM and the ASIC are embedded beneath multiple layers of the PCB (part number NTK539QS-220), preventing visual access. Any attempts to access or tamper with the embedded wires will damage the PCB layers, leaving visual evidence of the attempt.

- The KM is enclosed in a hard aluminum casing (part number NTK53926-501 for Hardware Ver 2.0; part number NTK53926-502 for Hardware Ver 2.1 and Ver 2.2) that is completely opaque within the visible spectrum. The enclosure is secured using two tamper-evident labels (part number 415-

[31] Mb/s – Megabits Per Second
[32] ms - millisecond

2424-001) applied at the factory; their locations can be seen in Figure 6. Any attempt to remove the tamper-evident labels will leave visual evidence of the attempt.

- The ASIC is mounted on the motherboard's PCB and covered by a heatsink (part number 410-7025-001), preventing any visibility of the component. The heatsink is affixed to the PCB using screws, and is protected using a steel security plate (part number 410-7023-001), and a tamper-evident label (part number 415-2424-001) as shown in Figure 6. Any attempt to defeat these mechanisms will result in physical damage to the module.

- There are two plastic opacity rings (part number 420-2160-002) that surround the KM and ASIC's PCB connection points that prevent visibility from a side angle. The opacity rings will break if a tamper attempt is made.

- On the bottom of the PCB, a heat spreader (part number 410-6598-001) is affixed to prevent tamper attacks from the underside of the KM. The heat spreader is secured by a tamper-evident label (part number 415-2424-001) over the screw that holds the heat spreader in place. The location of heat spreader and tamper-evident label can be seen below in Figure 6. Any attempt to remove the tamper-evident label will leave visual evidence of the attempt. Any attempt the defeat the heat spreader will result in visible damage to the heat spreader.

2.6 Operational Environment

The operational environment of the WL3e Encryption Module does not provide the module operator access to a general-purpose operating system (OS). The KM contains a Xilinx Zync 7020 (Xilinx XC7Z020) with Cortex A9 dual-core processor running an embedded Linux kernel in a non-modifiable operational environment. The Linux operating system on the KM is not modifiable by the operator, and only the KM firmware's signed image can be executed.

All KM firmware downloads are digitally signed, and a conditional self-test (ECDSA signature verification) is performed during each download. If the signature test fails, the new KM firmware is ignored and the current firmware remains loaded. Only FIPS-validated firmware may be loaded into KM to maintain the module's validation.

The ASIC contains an embedded ARM946E-S ARM[33] processor with 128 KB[34] of ITCM[35], 128 KB of DTCM[36], 8 KB of instruction cache, and 4 KB of data cache. Program and data storage is provided by 64 KB of ROM [37]and 2 MB[38] of RAM. The ASIC firmware is stored in ROM prior to being loaded into RAM. While the ASIC firmware is still in ROM, a 32-bit CRC check of the ROM bootloader is performed. If successful, the firmware is loaded into RAM. Immediately upon loading into RAM, an ECDSA signature verification test using NIST P-384 curve is performed on the firmware to ensure that the image has not been modified or corrupted in any way. Once loaded, the ASIC operating environment cannot be modified.

[33] ARM – Advanced Reduced Instruction Set Computing (RISC) Machines
[34] KB – Kilobytes
[35] ITCM – Instruction Tightly Coupled Memory
[36] DTCM – Data Tightly Coupled Memory
[37] ROM – Read-Only Memory
[38] MB - Megabytes

2.7 Cryptographic Key Management

The module generates keys as described in example #1 of FIPS 140-2 Implementation Guidance 7.8. It uses the FIPS-Approved CTR_DRBG (as specified in SP 800-90A) to generate cryptographic keys and ECDSA key pairs. The DRBG is seeded from seeding material provided by a hardware-based NDRNG, which provides an entropy source and whitening circuitry to supply a uniformly-distributed unbiased random sequence of bits to the DRBG.

The module supports the CSPs listed below in Table 7.

Table 7 – Cryptographic Keys, Cryptographic Key Components, and CSPs

CSP	CSP Type	Generation / Input	Output	Storage	Zeroization	Use
Base Key Encryption Key (BKEK)	AES 256-bit key	Preloaded at the factory	Never exits the module	Stored in plaintext in non-readable, write once, non-probe-able eFuse within the KM processor	N/A	Used for decrypting the MKEK stored in the module in non-volatile memory
Master Key Encryption Key (MKEK)	AES 256-bit key	Preloaded at the factory	Never exits the module	Encrypted with the BKEK and stored in non-volatile memory;	N/A	Used for encrypting or decrypting KEK.
Key Encryption Key (KEK)	AES 256-bit key	Generated internally	Never exits the module	Encrypted with MKEK and stored in non-volatile memory	By command via MyCryptoTool and TCS interface	Used for encrypting or decrypting private key of an entity key pair
Data Encryption Key (DEK)	AES 256-bit key	Generated internally	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for encrypting or decrypting payload data between an authorized external entity and the module
Initialization Vector (IV)	128-bit value	For encryption: generated internally (using an Approved DRBG with a cryptographically-strong entropy source) For decryption: generated externally and enters the module in encrypted form	For encryption: exits the module in encrypted form For decryption: never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used with AES-GCM for encrypting or decrypting payload data between an authorized external entity and the module

CSP	CSP Type	Generation / Input	Output	Storage	Zeroization	Use
Preshared Authentication String	256-bit value	Preloaded at the factory	Never exits the module	Stored plaintext in non-volatile memory (embedded in code)	N/A	Used for authenticating a CO for the Firmware Load service
IKEv2 ECDH[39] Private Component	384-bit value	Generated internally during IKEv2 negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for exchanging shared secret to derive session keys during IKEv2
IKEv2 ECDH Public Component	384-bit value	For the public component of the module: generated internally during IKEv2 negotiation For the public component of a peer: generated externally and enters the module in plaintext	For the public component of the module: exits the module in plaintext For the public component of a peer: never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for exchanging shared secret to derive session keys during IKEv2
IKEv2 Session Encryption Key	AES 256-bit key	Generated internally during EC DH key negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used with AES-GCM for encrypting/decrypting IKEv2 messages
IKEv2 Session Authentication Key	HMAC SHA-384	Generated internally during EC DH key negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for authenticating IKEv2 messages

[39] ECDH – Elliptic Curve Diffie-Hellman

CSP	CSP Type	Generation / Input	Output	Storage	Zeroization	Use
TLS Session Key	AES 128, 256-bit key	Generated internally during session negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used with AES-GCM for encrypting/decrypting TLS messages
TLS Authentication Key	HMAC SHA-256, HMAC SHA-384	Generated internally during session negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for authenticating TLS messages
TLS Pre-Master Secret	384-bit random value	Generated internally during session negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Establish the TLS Master Secret
TLS Master Secret	384-bit random value	Generated internally during session negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Establish the TLS Session Key
Peer ECDSA Public Key	384-bit key	Enters the module in encrypted form	Never exits the module	Stored in plaintext in RAM	By command via MyCryptoTool and TCS interface	Used for peer device authentication for IKE v2 communications
CA ECDSA Public Key	384-bit key	Enters the module in encrypted form	Never exits the module	Stored plaintext in non-volatile memory	By command via MyCryptoTool and TCS interface	Used for authenticating the operator
CO ECDSA Public Key	384-bit key	Enters the module in encrypted form	Never exits the module	Stored in plaintext in RAM	By command via MyCryptoTool and TCS interface	Used for authenticating the CO
User ECDSA Public Key	384-bit key	Enters the module in encrypted form	Never exits the module	Stored in plaintext in RAM	By command via MyCryptoTool and TCS interface	Used for authenticating the User

CSP	CSP Type	Generation / Input	Output	Storage	Zeroization	Use
Module Data Path ECDSA Private Key	384-bit key	Generated internally using Approved DRBG; imported in encrypted form	Never exits the module	Stored encrypted with KEK in non-volatile memory	By command via MyCryptoTool and TCS interface	Used for peer device authentication for IKE v2 communications
Module Data Path ECDSA Public Key	384-bit key	Generated internally using Approved DRBG; imported in encrypted form	Exits the module in encrypted form	Stored plaintext in non-volatile memory	By command via MyCryptoTool and TCS interface	Used for peer device authentication for IKE v2 communications
Module Web Access ECDSA Private Key	384-bit key	Generated internally using Approved DRBG; imported in encrypted form	Never exits the module	Stored encrypted with KEK in non-volatile memory	By command via MyCryptoTool and TCS interface	Used with certificates in mutual authentication
Module Web Access ECDSA Public Key	384-bit key	Generated internally using Approved DRBG; imported in encrypted form	Exits the module in encrypted form	Stored plaintext in non-volatile memory	By command via MyCryptoTool and TCS interface	Used with certificates in mutual authentication
ECDH Private Component	384-bit value	Generated internally during HTTPS negotiation	Never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for establishing HTTPS session for MyCryptoTool
ECDH Public Component	384-bit value	For the public component of the module: generated internally during HTTPS negotiation For the public component of a peer: enters the module in plaintext	For the public component of the module: exits the module in plaintext For the public component of a peer: never exits the module	Stored in plaintext in RAM	By session termination, reboot, power removal, or command via MyCryptoTool and TCS interface	Used for establishing HTTPS session for MyCryptoTool
DRBG Seed	384-bit value	Generated internally using entropy input	Never exits the module	Stored in plaintext in RAM	By reboot, power removal, or command via MyCryptoTool and TCS interface	Used for random number generation

CSP	CSP Type	Generation / Input	Output	Storage	Zeroization	Use
Entropy Input String	512-bit value	Generated internally using NDRNG	Never exits the module	Stored in plaintext in RAM	By power removal or command via MyCryptoTool and TCS interface	Used for random number generation

2.8 EMI / EMC

The module was tested and found to be conformant to the EMI/EMC requirements specified by Title 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class A (i.e., for business use).

2.9 Self-Tests

The module performs various self-tests (power-up self-tests, conditional self-tests, and critical function self-tests) on the cryptographic algorithm implementations to verify their functionality and correctness.

2.9.1 Power-Up Self-Tests

The Ciena 6500 Flex3 WaveLogic 3e OCLD Encryption Module performs the following self-tests at power-up to verify the integrity of the firmware images and the correct operation of the FIPS-Approved algorithms implemented in the module:

- Integrity tests for the KM:
  - KM application firmware image (Zone A) using ECDSA signature verification
  - KM FPGA image (Zone A) using ECDSA signature verification
  - KM application firmware image (Zone B) using ECDSA signature verification
  - KM FPGA image (Zone B) using ECDSA signature verification
- Integrity test for the ASIC:
  - ASIC firmware image using ECDSA signature verification
- Cryptographic algorithm tests for all implementations of the following FIPS-Approved algorithms:
  - KM
    - AES Encryption Known Answer Test (KAT)
    - AES Decryption KAT
    - Triple-DES Encryption KAT
    - Triple-DES Decryption KAT
    - SHA-1 KAT
    - SHA-256, 384, 512 KAT
    - HMAC SHA-1 KAT
    - HMAC SHA-256, 384, 512 KAT
    - SP 800-90A CTR_DRBG KAT
    - ECDSA 186-4 Signature Generation Pairwise Consistency Test (PCT)
    - ECDSA 186-4 Signature Verification PCT
  - ASIC
    - AES Encryption KAT
    - AES Decryption KAT

The power-up self-tests can be performed at any time by power-cycling the module or via TCS command.

2.9.2 Conditional Self-Tests

The module implements the following conditional self-tests:

- Continuous Random Number Generator Test (CRNGT) for the SP 800-90A CTR_DRBG
- CRNGT for the NDRNG
- ECDSA key pair generation and verification

- Firmware Load Test for the KM Application using ECDSA signature verification
- Firmware Load Test for the KM FPGA using ECDSA signature verification

2.9.3 Critical Functions Tests

The module performs the following critical functions tests:

- SP 800-90 CTR_DRBG Instantiate Health Test
- SP 800-90 CTR_DRBG Generate Health Test
- SP 800-90 CTR_DRBG Reseed Health Test
- SP 800-90 CTR_DRBG Uninstantiate Health Test

2.9.4 Self-Test Failure Handling

Upon the failure of any power-up self-test (except the Zone A KM application firmware Integrity test, Zone B KM firmware integrity test, or the Zone B FPGA integrity test), conditional self-test (except the firmware load tests), or critical functions tests, the module goes into “Critical Error” state and disables all access to cryptographic functions and CSPs. All data outputs via data output interfaces are inhibited upon any self-test failure. A permanent error status will be relayed via the status output interface, which then is interpreted either in the illumination of an LED or as a recorded entry to the system log file or alarm history log file.

During the integrity tests at start up, the module first checks the Zone A firmware image. If this test fails, the module transitions to the Zone A Soft Error state where it will forgo the Zone A FPGA self-test and proceed with the Zone B application firmware integrity and FPGA tests. If the Zone A firmware image passes the integrity check, the Zone A FPGA is checked. If the Zone A FPGA integrity check fails, the module transitions to the Critical Error state. If the Zone A FPGA integrity passes, the module checks the firmware and FPGA within Zone B.

If the Zone B firmware integrity check fails, the module transitions to either the Critical Error state (if the Zone A firmware integrity check also failed) or the Zone B Soft Error state (if the Zone A application firmware integrity check passed).

If the Zone B firmware integrity check passes, but the Zone B FPGA integrity check fails, the module transitions to a Zone B Soft Error state where a new firmware image can be loaded from the TCS interface followed by a reboot.

Upon failure of the firmware load test, the module enters “Soft Error” state. The soft error state is a non-persistent state wherein the module resolves the error by rejecting the loading of the new firmware. Upon rejection, the error state is cleared, and the module resumes its services using the previously-loaded firmware.

While the error state persists, the module replies to all cryptographic service requests with a pre-defined error message to indicate the current error status. The management interface does not respond to any commands until the module is operational. The module requires rebooting or power-cycling to come out of the error state and resume normal operations. In the case of a KM firmware or KM FPGA load corruption in Zone B that cannot be corrected by the TCS interface, the module will not be able to resume normal operation and the Crypto Officer should contact Ciena.

2.10 Mitigation of Other Attacks

This section is not applicable. The module does not claim to mitigate any other attacks.

3. Secure Operation

The WL3e Encryption Module meets Level 2 requirements for FIPS 140-2. The following sections describe how to place and keep the module in FIPS-Approved mode of operation.

3.1 Initial Setup

The module does not require any installation activities as it is delivered to the customer pre-installed on the circuit pack from the factory. Either the CO or the User can perform the Secure Operation responsibilities and tasks listed here; however, this Security Policy places this responsibility solely on the CO.

The module is shipped from the factory with the required physical security mechanisms (tamper-evident labels, opacity rings, security plate, heatsink, PCB layers, and heat spreader) installed. After removing the circuit pack from the shipping package, but prior to use, the CO must perform a physical inspection of the unit for signs of damage. The CO must ensure that all physical security mechanisms are in place. Additionally, the CO should check the package for any irregular tears or openings. If damage is found or tampering is suspected, the CO should immediately contact Ciena.

The KM is contained in a strong, hard metal enclosure, and is protected by two tamper-evident labels. The wire connections between the KM and ASIC are protected from view and from tampering by multiple PCB layers. The bottom of the PCB where the KM connects is protected using the heat spreader and tamper-evident label. The ASIC component of the module is protected by the installed heatsink and security plate, with one tamper-evident label over one of the screws. In total, the module requires four tamper-evident labels (see Figure 6 for the locations of the tamper-evident labels).

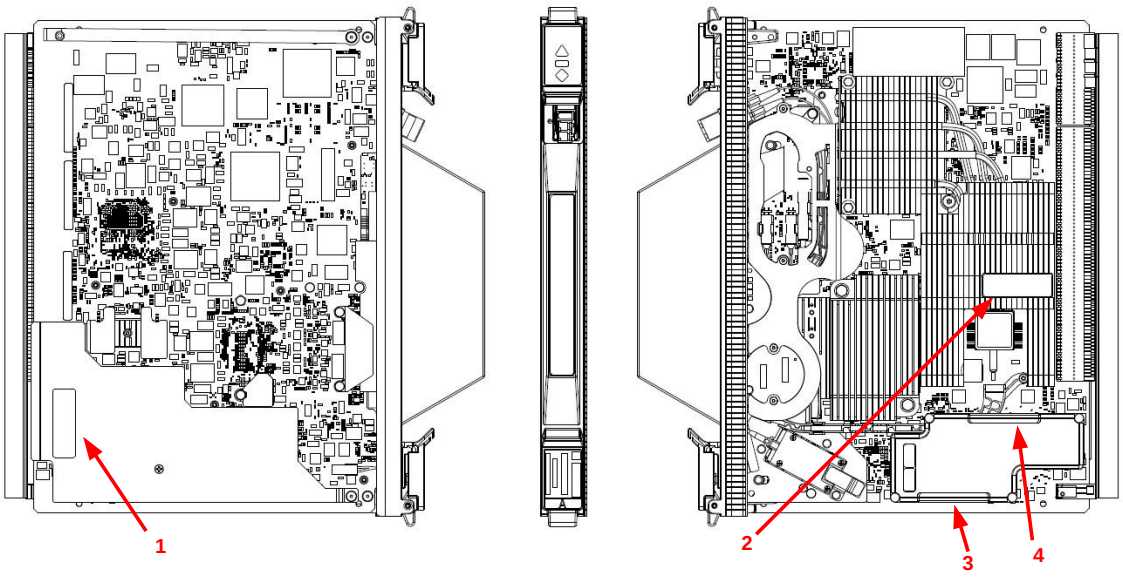

Figure 6 – Tamper-Evident Label Locations

The CO is responsible for the configuration the module, which includes configuring the data path parameters and certificates. The CO must install the web server certificate and at least one CA certificate in order for the module to be able to verify the submitted CO and User ECDSA Public Keys during TLS mutual authentication for the MyCryptoTool interface. Please refer to Chapter 4, "Provisioning Certificate Management using MyCryptoTool", in Ciena's *User's Guide and Technical Practices* document for more information. Once the module's web server certificate has been configured, the web server software will restart for the certificate change to take effect and begin enforcing TLS mutual authentication. When the web server has completed the restart process, the module operates only in FIPS-Approved mode of operation. At any point in time, the "FIPS mode" status of the module can be viewed using the MyCryptoTool interface.

Once properly provisioned, the module will operate in FIPS-Approved mode of operation until it is decommissioned by the CO or the physical security is breached.

3.2 Secure Management

The CO is responsible for maintaining and monitoring the status of the module to ensure that it is running in its FIPS-Approved mode. For additional details regarding the management of the module, please refer to Ciena's *User's Guide and Technical Practices* document.

3.2.1 Management

When configured according to the CO guidance in this Security Policy, the module only runs in an Approved mode of operation. The CO is able to monitor and configure the module via MyCryptoTool. Detailed instructions for monitoring and troubleshooting the module are provided in the Ciena's *User's Guide and Technical Practices* document.

3.2.2 Physical Inspection

As the labels are applied at the factory, the CO shall inspect the module to ensure that the labels are applied correctly. The CO shall inspect the module for evidence of tampering at six-month intervals. The CO shall visually inspect the tamper-evident labels for tears, rips, dissolved adhesive, and other signs of tampering. The CO shall also inspect the PCB, the KM component's enclosure, and the ASIC's heatsink, security plate, and tamper-evident label for any signs of damage. If evidence of tampering is found during periodic inspection, the Crypto Officer should send the module back to Ciena Corporation for repair or replacement.

3.2.3 Monitoring Status

The Crypto Officer should monitor the module's status regularly. The operational status of the module can be viewed using MyCryptoTool. At any point of time, the "FIPS mode" status of the module can be viewed by accessing the "Encryption Details", "Data Encryption Certificate Management", "Web Access Certificate Management", "Active Alarms", or "Historical Logs" web page of the MyCryptoTool interface. The line at the top of these pages indicates "FIPS mode" of the module.

3.2.4 Zeroization

All ephemeral keys used by the module are zeroized on reboot, loss of power, session termination, or MyCryptoTool erasure. The "Clear CSP (Critical Security Parameter)" button on MyCryptoTool and the Zeroize

command via TCS also allows an operator to clear certificates' public keys, private keys, and the KEK. The BKEK, MKEK and KEK CSPs reside in non-volatile memory.

The other CSPs are stored in the volatile and non-volatile memories of the module. The zeroization of the KEK, which encrypts all other CSPs, renders all the other CSPs stored in non-volatile memory useless, thereby effectively zeroizing them. The zeroization of KEK renders asymmetric private keys inaccessible, thereby rendering them unusable. The only public key that is stored in a file is embedded in code and is used for verifying the integrity of the firmware load image files cannot be zeroized.

3.3 User Guidance

The User shall follow all the instructions and guidelines provided for the Crypto Officer in Section 3 of this Security Policy document in order to ensure the secure operation of the module.

4. Acronyms

Table 8 provides definitions for the acronyms used in this document.

Table 8 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
ARM	Advanced RISC Machine
ASIC	Application-Specific Integrated Circuit
CA	Certificate Authority
CBC	Cipher Block Chaining
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CRL	Certificate Revocation List
CRNGT	Continuous Random Number Generator Test
CSE	Communications Security Establishment
CSP	Critical Security Parameter
CTR	Counter
DEK	Data Encryption Key
DES	Data Encryption Standard
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DTCM	Data Tightly Coupled Memory
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EEPROM	Electrically-Erasable Programmable Read-Only Memory
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FIPS	Federal Information Processing Standard
FPGA	Field Programmable Gate Array
Gb/s	Gigabit Per Second
GbE	Gigabit Ethernet
GCM	Galois/Counter Mode
HMAC	(Keyed-) Hash Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure

Acronym	Definition
I/O	Input/Output
ITCM	Instruction Tightly Coupled Memory
IKE	Internet Key Exchange
IV	Initialization Vector
KAT	Known Answer Test
KB	Kilobyte
KEK	Key Encrypting Key
KM	Krypto Module
LED	Light Emitting Diode
Mb/s	Megabits per second
MKEK	Master Key Encrypting Key
ms	millisecond
N/A	Not Applicable
NDRNG	Non-Deterministic Random Number Generator
NIST	National Institute of Standards and Technology
NOR	Not Or
OCLD	Optical Channel Laser and Detector
OS	Operating System
OTN	Optical Transport Network
OTR	Optical Transponder
PCB	Printed Circuit Board
PCT	Pairwise Consistency Test
PKCS	Public-Key Cryptography Standard
PKG	Public Key Generation
PKV	Public Key Validity
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RAM	Random Access Memory
RISC	Reduced Instruction Set Computing
ROM	Read Only Memory
SDRAM	Synchronous Dynamic Random Access Memory
SHA	Secure Hash Algorithm
SP	Special Publication
TLS	Transport Layer Security
WL3e	WaveLogic 3 Extreme