CyberCogs, Inc.

CyberCogs Hardware Security Module

FIPS 140-2 Non-Proprietary Security Policy

Hardware versions: CC50-3, CC100-3, CC200-3, CC300-3, CC400-3, CC50-4, CC100-4, CC200-4, CC300-4, CC400-4

Firmware version: 3.0

Date: September 5, 2022

Prepared by:

Acumen Security

2400 Research Blvd, Suite 395

Rockville, MD 20850

www.acumensecurity.net

Introduction

Federal Information Processing Standards Publication 140-2 — Security Requirements for Cryptographic Modules (FIPS 140-2) specifies requirements for cryptographic modules to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Canadian Centre for Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) run the FIPS 140 program. The NVLAP accredits independent testing labs to perform FIPS 140 testing; the CMVP validates modules meeting FIPS 140 validation. Validated is the term given to a module that is documented and tested against the FIPS 140 criteria.

More information is available on the CMVP website at: https://csrc.nist.gov/projects/cryptographic-module-validation-program

About this Document

This non-proprietary Cryptographic Module Security Policy for the CyberCogs Hardware Security Module (HSM) from CyberCogs, Inc. provides an overview of the product and a high-level description of how it meets the overall Level 3 security requirements of FIPS 140-2.

The CyberCogs Hardware Security Module may also be referred to as “the HSM”, or simply “the module” in this document.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. CyberCogs, Inc. shall have no liability for any error or damages of any kind resulting from the use of this document.

Notices

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

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Scope

This document describes the cryptographic module security policy for the CyberCogs, Inc. CyberCogs Hardware Security Module (HSM) (also referred to as the "module" or "HSM" hereafter). It contains specification of the security rules under which the cryptographic module operates, including the security rules derived from the requirements of the FIPS 140-2 standard.

1.2 Overview

The CyberCogs HSM is a multi-chip embedded hardware cryptographic module in the form of a PCI-Express card. The module is intended for use within a general-purpose or custom computing platform. The module is contained in its own secure tamper envelope providing active anti-tamper functionality.

The module is designed to offer blind secure PKCS11 key signing, encryption, and token storage with physical security, over a PCIe connection to a Host server. There is no exposure of unprotected data from the HSM to the host. The module provides secure key generation and storage for symmetric keys and asymmetric key pairs along with symmetric and asymmetric cryptographic services. Access to key material and cryptographic services for operators and application software is provided through the PKCS #11 programming interface.

The critical hardware components of the HSM are:

- HSM processor, which is a Zynq UltraScale+ MPSoC device that provides an ARMv8 quad core CPU
- MSP430 TSM (Token Security Module) that prevents cloning of the HSM, provides tamper protection, and also includes a real-time clock,
    - Battery backed internal SRAM (8KB),
    - 256KB FRAM which is used internally by the MSP430 TSM, which is used to store master keys (zeroized on tamper event),
- ATECC608B ring-oscillator based noise source,
- 2GB RAM, which is used to execute the ephemeral root POSIX filesystem (Petalinux 2018.3) utilized by the Zynq processor,
- 256 Mbit (32 Mbyte) non-volatile persistent NOR flash storage, used to store the Linux kernel and root filesystem image,
- Do we 512KB MRAM (4Mb (512K x 8)), non-volatile memory used to store non cryptographic usage information,
- 32GB NAND flash for storage of at-rest files, encrypted application data, signed update image (update)
- Connectors for USB (smartcards) and Serial (diagnostics).

A module may be explicitly configured to operate in either FIPS 140-2 Approved mode, or in a non-Approved mode of operation. Section 11 provides additional information for configuring the module in FIPS 140-2 Approved mode of operation.

2. Security Level

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	3
Cryptographic Module Ports and Interfaces	3
Roles, Services, and Authentication	3
Finite State Model	3
Physical Security	4
Operational Environment	N/A
Cryptographic Key Management	3
Electromagnetic Interference / Electromagnetic Compatibility	3
Self-Tests	3
Design Assurance	3
Mitigation of Other Attacks	N/A
Overall Level	3

Table 1 – FIPS 140-2 Target Level

3. Cryptographic Module Specification

3.1 Cryptographic Boundary

The CyberCogs HSM is a multi-chip embedded hardware module. The cryptographic boundary of the module is represented as a hard-metallic cover as shown in Figure 1 and Figure 2 (highlighted in red). Figure 1 depicts the top view of the CyberCogs HSM. Figure 2 depicts the bottom view. Note: There are 10 models tested (CC50-3, CC100-3, CC200-3, CC300-3, CC400-3, CC50-4, CC100-4, CC200-4, CC300-4, CC400-4). Each version uses an identical PCB, module components, firmware version, and enclosure. The models only differ by the number of configured CPU execution cores and how they are marketed to end-consumers.

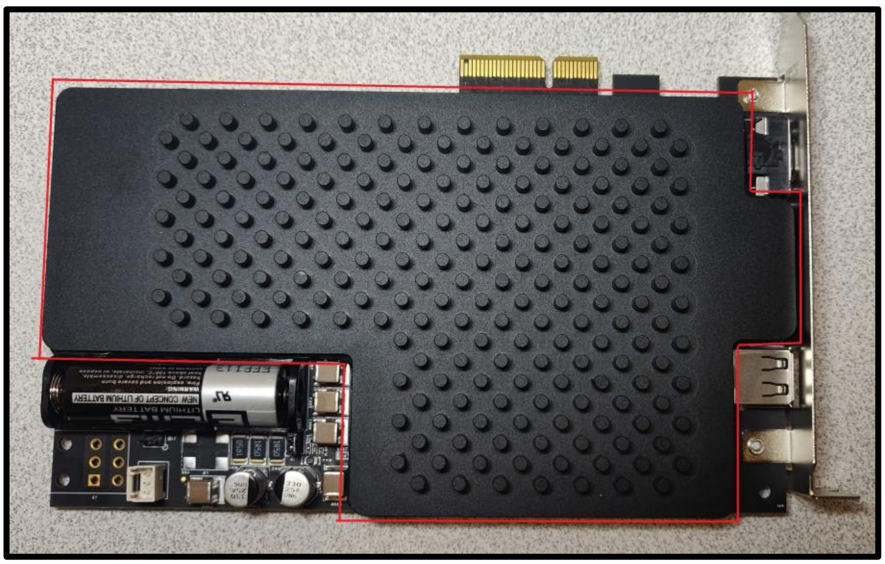

Figure 1 – CyberCogs HSM Top View

Figure 2 – CyberCogs HSM Bottom View

4. Cryptographic Module Ports and Interfaces

4.1 Module Interface Description

The module supports the following physical ports and interfaces:

- PCIe interface
- Serial port
- USB 2.0 port
- 2x Jumper pin

The physical ports of the module are shown in Figure 3. Port 1 is Type A USB port that is used to store split keys using a smart card reader directly connected to the USB port. Port 2 is Micro-DB9 for the Serial cable. The PCIe interface is depicted in Figures 1 and 2 above.

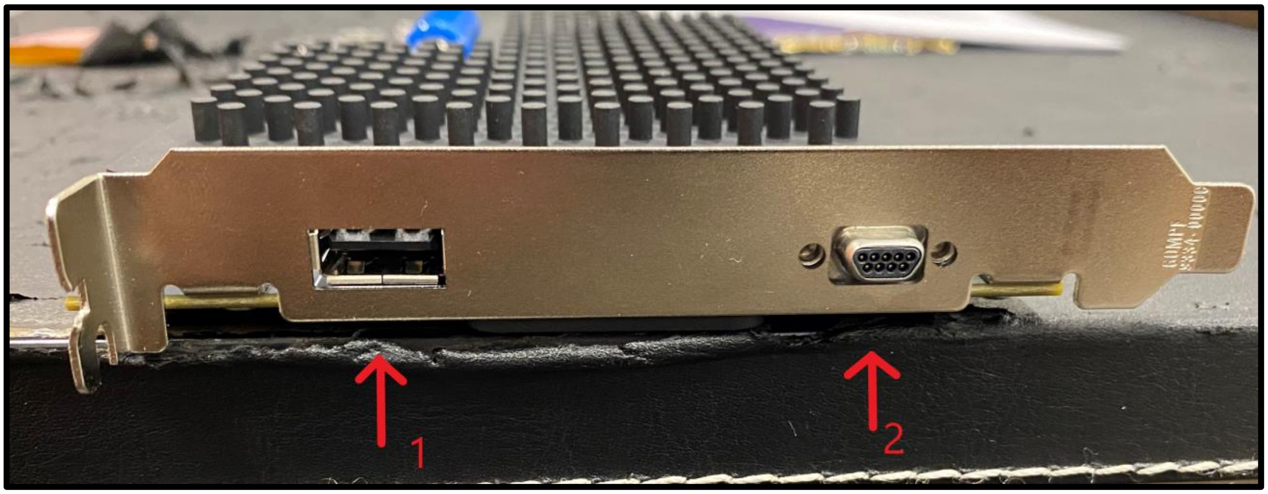

Figure 3 – CyberCogs HSM PCI Bracket View

FIPS 140-2 Interface	Physical Interface	Logical Interface
Data Input interface	PCIe interface	PKCS11 API
	USB port	Split Key Backup
Data Output interface	PCIe interface	PKCS11 API
	USB port	Split Key Backup
Control Input interface	PCIe interface	PKCS11 API
	2x Jumper Pin	User-specified tamper
Status Output interface	PCIe interface	PKCS11 API
	Serial port	Console status output
Power	PCIe interface	N/A

Table 2 – Mapping of FIPS 140-2 Interfaces to Physical and Logical Interfaces

5. Roles, Services and Authentication

5.1 Roles

The HSM supports the defined "User Role" and "Crypto Officer Role" using Identity based authentication.

These roles are mapped to four operator types; three of which are mapped to the Crypto Officer role and one is associated with the User role.

- Crypto-Officer role
  - Admin Slot/Token Security Officer (ATS):
  - Admin Slot/Token User (ATU)
  - Non-Admin Slot/Token Security Officer (SO)
- User role
  - Non-Admin Slot/Token User (USER)

These roles represent a default grouping of permissions that enable the ability to execute the module's services.

There is no Maintenance Role defined for the module.

5.1.1 User Role

In this role, the Slot/Token User is allowed to access CSPs, generate and import new CSPs, export CSPs in encrypted form, and perform cryptographic operations within a non-Admin slot/token. They may change their own password as well.

The list of services supported by the user role is listed in Section 5.3 of the document.

5.1.2 Crypto Officer (CO) Role

There are three types that fill the role of the Crypto Officer "which are" the Admin Slot/Token Security Officer (ATS), the Admin Slot/Token User (ATU), and the Non-Admin Slot/Token Security Officer (SO). The ATS is responsible for performing device administrative functions such as setting the clock and performing initial configuration of the device into FIPS mode. The ATU performs initialization services, token management, firmware updates and can issue a factory reset. The SO performs user password management functions, split key backup and restore operations.

The list of services supported by each CO role is listed in Section 5.3 of the document.

5.1.3 Unauthenticated Role

An unauthenticated user is not allowed to perform any security related functionality. The list of services supported by the unauthenticated operator is listed in Section 5.3 of the document and is compliant with IG 3.1.

5.2 Authentication

An operators of the HSM must authenticate themselves as one of these users in order to perform any cryptographic service that utilizes keys or to perform any actions that modify the state of the HSM.

Role	Type of Authentication	Authentication Data
Non-Admin Slot/Token User	Identity Based	Password
Non-Admin Slot/Token Security Officer	Identity Based	Password
Admin Slot/Token User	Identity Based	Password
Admin Slot/Token Security Officer	Identity Based	Password

Table 3 – Roles and Authentication Data

HSM operators are authenticated using identity-based authentication. The module supports multiple concurrent operators. Each operator provides their respective identity (unique token identifier), when using the PKCS11 API and also authentication data. That authentication data is supplied as a password.

5.2.1 Password Authentication

A password must be at least four (4) characters long. The acceptable character set is as follows:

- 26 Alphabetics **A** through **Z**
- 26 Alphabetics **a** through **z**
- 10 Numerics **0** through **9**
- 10 Symbolics found on shifted numerics **!** through **)**
- 22 Symbolics (unshifted and shifted)
- 1 Space key

The module supports a character set consisting of at least 95 possible ASCII characters. At a minimum length of 4 characters, the probability of randomly guessing the correct sequence is one (1) in 81,450,625 (the calculation should be 95*95*95*95 = 81,450,625). Therefore, for each attempt to use the authentication mechanism, the associated probability of a successful random attempt is approximately 1 in 81,450,625, which is less than the 1 in 1,000,000 required by FIPS 140-2.

Incorrect passwords supplied to the HSM cause the module to step through a progressive delay before the next attempt is possible. For the first three consecutive incorrect passwords, the module does not impose any delay but stalls after three consecutive incorrect passwords. The module will no longer permit password authentication for the next 5 seconds. Each successive failed authentication attempt (once the timeout has expired) will increase the next timeout by a progressively larger multiple of 5 seconds. The Nth timeout is (N - 3)*5 seconds such that the cumulative delay incurred by that time is (5/2) (N - 3)*(N - 2) for N > 2. Note that any successful password authentication resets back to the initial condition.

Attempt Number	Minimum Cumulative Attack Time Offset
1	0
2	0
3	0
4	5
5	10
6	15
7	20

Table 4 – Delay enforced for successive incorrect password attempts

The maximum number of failed authentication attempts within one minute is 7. The associated probability of a successful random attempt for a minute is approximately 1 in (81,450,625/7), which is less than the 1 in 100,000 required by FIPS 140-2.

5.3 Services

The HSM supports a number of services that are available to authenticated users. The module implements the following access control policy on keys and CSPs in the module shown in the following table. The access policy is noted by R=Read, W=Write, X=Execute, and Z = Zeroize.

Service	Role Required	Key and CSP Access
HSM Initialization	ATU	Z: All Keys and CSPs X: DRBG-K RW: DRBG-V X: SPIN(default) W:SLP,SLC,SWP,SMK
Show HSM Clock	ATS or Unauthenticated	N/A
Set HSM Clock	ATS	N/A
Set FIPS Mode	ATS	Z: All Keys and CSPs X: DRBG-K RW: DRBG-V X: SPIN(default) W: SLP,SLC,SWP,SMK
Create Token	ATU	X: DRBG-K RW: DRBG-V X: SPIN(default, of the new token) W: SLP,SLC,SWP,SMK
(Re)Initialize Token	SO[1]	X: DRBG-K RW: DRBG-V X: SPIN W: SLP,SLC,SWP,SMK Z: PTO-AK, PTO-SK
Delete Token	ATU	Z: All Token Keys and CSPs
Set USER Password	SO	X: DRBG-K RW: DRBG-V X: UPIN W: ULP,ULC,UWP,UWK,UMK
Change USER Password	USER[2]	X: DRBG-K RW: DRBG-V X: UPIN W: ULP,ULC,UWP,UWK,UMK
Change CO Password	SO	X: DRBG-K RW: DRBG-V X: SPIN W: SLP,SLC,SWP,SWK,SMK
Establish channel to HSM	Unauthenticated	X: DRBG-K RW: DRBG-V WX: ECC Secure Channel Private Key/Public Key WXZ: ECDH Shared Secret WX: ECDH Secure Channel Session Key
Establish/Terminate PKCS11 Token Session	Unauthenticated	N/A
USER Login to Session (aka Token)	USER	X: UPIN R: ULP,ULC,UWP,UWP,UMK X: TMK,TEK

[1] The CO (Token-CO) role will be ATS whenever the token being addressed is the admin token in slot zero.
[2] The USER (Token-USER) role will be ATU whenever the token being addressed is the admin token in slot zero.

Service	Role Required	Key and CSP Access
CO Login to Session (aka Token)	SO	X: SPIN R: SLP,SLC,SWP,SWP,SMK X: TMK,TEK
Logout from Session (aka Token)	SO or USER	N/A
Generate SHA Digest	SO or USER or Unauthenticated	N/A
Generate Random Data	SO or USER or Unauthenticated	X: DRBG-K RW: DRBG-V
Encrypt Data	USER	X: PTO-SK or PTO-AK (public) X: DRBG-K RW: DRBG-V for IV when using AES-GCM
Decrypt Data	USER	X: PTO-SK or PTO-AK (private)
Generate HMAC and AES CMAC	USER	X: PTO-SK
Verify HMAC and AES CMAC	USER	X: PTO-SK
Generate Digital Signature	USER	X: PTO-AK (private)
Verify Digital Signature	USER	X: PTO-AK (public)
Split Key/Token Backup	SO	X: DRBG-K RW: DRBG-V WXZ: Split Password WX: Split key parameters R: PTO-SK or PTO-AK (public)
Split Key/Token Restore	SO	WXZ: Split Password RX: Split key parameters W: PTO-SK or PTO-AK (private)
Zeroization (Tamper)	SO or Unauthenticated	Z: All Keys and CSPs
Zeroization (Destroy Object)	USER	Z: PTO-SK or PTO-AK
Perform Firmware Update[3]	ATU	X: Firmware Update Key
Generate Symmetric Key	USER	X: DRBG-K RW:DRBG-V W: PTO-SK, TEK X: TMK,TEK
Promote AES Key To Wrapping Key	SO	RW: PTO-SK (attribute) X: TEK
Generate Key Pair	USER	X: DRBG-K RW: DRBG-V W: PTO-AK, TEK X: TMK,TEK

[3] Any firmware loaded into this module that is not shown on the module certificate, is out of the scope of this validation and requires a separate FIPS 140-2 validation.

Service	Role Required	Key and CSP Access
Key Agreement – SSC	USER	WXZ: PTO-AK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: DH/ECDH Shared Secret, TEK
Key Agreement – KDA	USER	WXZ: PTO-AK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: DH/ECDH Shared Secret, User KDF Derived Key, TEK
Key Derivation – KBKDF	USER	X: PTO-SK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: User KDF Derived Key, TEK
Wrap Key (with/without metadata)	USER	R: PTO-SK or PTO-AK (public) (with/without metadata) X: PTO-SK
Unwrap Key (with/without metadata)	USER	X: PTO-SK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: PTO-SK or PTO-AK (private) (with/without metadata)
PK Wrap Key (without metadata)	USER	R: PTO-SK or PTO-AK (public) (without metadata) X: PTO-AK
PK Unwrap Key (without metadata)	USER	X: PTO-AK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: PTO-SK or PTO-AK (private), TEK
Wrap Token	ATU of the wrapping HSM; SO and USER Passwords of the wrapped token	WX: ECIES Session Keys R: PTO-AK or PTO-SK X: PTO-AK (public)

Service	Role Required	Key and CSP Access
Unwrap Token	ATU of the unwrapping HSM; SO and USER Passwords of the token being unwrapped	WX: ECIES Session Keys: PTO-AK (private) X: DRBG-K RW: DRBG-V W: TEKs X: TMK, TEKs W: PTO-SK or PTO-AK, TEKs
CO Wrap Object	SO	R: PTO-SK or PTO-AK X: PTO-SK
CO Unwrap Object	SO	X: PTO-SK X: DRBG-K RW: DRBG-V W: TEK X: TMK, TEK W: PTO-SK or PTO-AK, TEK
Export Public Key	USER or Unauthenticated	R: PTO-AK (public)
Import Public Key	USER	W: PTO-AK (public)
Get Info	SO or USER or Unauthenticated	N/A
Get FIPS Status	SO or USER or Unauthenticated	N/A
Get Logs	ATU	N/A
Perform Self-Tests	ATU or Unauthenticated	N/A

Table 5 – Approved Roles, Services and CSP Access

6. Physical Security

The module is a multiple-chip embedded cryptographic module made of production-grade materials. the module includes only standard, production-quality ICs, designed to meet typical commercial-grade specifications for power, temperature, reliability, shock and vibration. The ICs used in the module are coated with commercial standard passivation.

The CyberCogs HSM Cryptographic module is contained within a production-grade tamper-evident metal enclosure; inside the metal enclosure consists of the flex circuit surrounding the entirety of the module's internal circuitry. In addition to the metal enclosure and flex circuit, the CyberCogs HSM module internal components are encapsulated in a hard, black, potting compound using production grade Epoxies 50-3150 FR. Please note that all the tests are performed on the metal enclosure and that meets the Level 4 requirements of FIPS 140-2 Physical Security. The hard potted epoxy acts as an additional layer of security for the module. The module does not have any ventilation holes, slits, or other openings and meets the opacity requirements. The module does not contain any doors or removable covers or a maintenance access interface. The flex circuit prevents physical access to any of the internal components such that attempts at accessing the internals will guarantee immediate zeroization of cryptographic keys and CSPs.

The CyberCogs HSM module provides Environmental Failure Protection (EFP). Normal operating temperature range is from 0°C to 60°C. When the module’s temperature is close to the appropriate extreme of the normal operating range (i.e., at 0°C and 60°C), the module continues to operate normally and extending the module's temperature below 0°C and above 60°C will zeroize all plaintext keys and CSPs of the module.

The ambient voltage range for the PCIe interface is 11.4V - 13.6V and the external battery has an operating voltage range of 2.7V - 3.5V. In both cases, whenever the module is close to the appropriate extreme of the normal operating range (i.e., at 11.4V and 13.6V for PCIe interface, 2.7V and 3.5V for external battery), the module continues to operate normal and extending the voltage below 11.4V (for PCIe interface) and 2.7V (for external battery) and above 13.6V (for PCIe interface) and 3.5V (for external battery) will result in zeroization of all plaintext keys and CSPs of the module.

7. Operational Environment

The operational environment of the module is limited and therefore the requirements of this section are not applicable. The module is classified as Limited OE as that is designed to accept only controlled firmware changes that successfully pass the firmware load test.

8. Cryptographic Algorithms and Key Management

8.1 Cryptographic Algorithms

The module implements the following Approved algorithms. There are algorithms, modes, and keys that have been CAVs tested but not used by the module. Only the algorithms, modes/methods, and key lengths/curves/moduli shown in this table are used by the module:

CAVP Cert #	Algorithm	Standard	Mode/ Method/ Key Lengths, Curves, or Moduli	Use
A1438	AES	FIPS PUB 197 NIST SP 800-38A	CBC (128, 192, 256) CTR (128, 192, 256) ECB (128, 192, 256)	Encryption/Decryption
		FIPS PUB 197 NIST SP 800-38D	GCM[4] (128, 192, 256)	Authenticated Encryption/Decryption
		NIST SP800-38C	CCM (128, 192, 256)	Authenticated Encryption/Decryption
		NIST SP800-38B	CMAC (128, 192, 256)	Authenticated Generation/Verification
	ECDSA	FIPS 186-4	Key Generation (P-224/256/384/521) Key Verification (P-224/256/384/521) Signature Generation (P-224/256/384/521) Signature Verification (P-224/256/384/521)	Key Gen/Key Ver/Sign/Verify
	HMAC	FIPS PUB 198-1	SHA-1, SHA2-224/SHA3-224 SHA2-256/SHA3-256 SHA2-384/SHA3-384 SHA2-512/SHA3-512	Keyed-Hash Message Authentication
	KAS	NIST SP800-56Arev3	KAS-ECC: Scheme: "OnePassDH" with One Step KDF and curves "P-224/256/384/521" Scheme: "Ephemeral Unified" with One Step KDF and curves "P-224/256/384/521" Scheme: "Static Unified" with One Step KDF and curves "P-224/256/384/521"	Key Establishment
	KAS-SSC	NIST SP800-56Arev3	KAS-ECC-SSC: Scheme: "OnePassDH" with curves "P-224/256/384/521" Scheme: "Ephemeral Unified" with curves "P-224/256/384/521" Scheme: "Static Unified" with "P-224/256/384/521"	Key Establishment

[4] The module meets scenario 2 of IG A.5. The IV is at least 96-bits in length as generated in its entirety internally by the module's Approved DRBG.

			KAS-FFC-SSC: Schemes: "dhEphem", "dhOneFlow", and "dhStatic"	
	Safe Primes	NIST SP800-56a r3	Key Generation and Key Verification: MODP-2048/3072/4096/6144/8192 Ffdhe2048/3072/4096/6144/8192	
	KTS	FIPS 140-2 IG D.9	AES Cert. #A1438: AES CCM 256-bit	Key Transport
	KTS	FIPS 140-2 IG D.9	AES Cert. #A1438 and AES Cert. #A1438): AES-256-CTR and AES-128-CMAC.	Key Transport
	KTS-RSA	NIST SP800-56Brev2	Key Generation Methods: rsakpg1-basic, rsakpg1-crt and rsakpg1-prime-factor Scheme: KTS-OAEP-basic Modulo(s): RSA 2048/3072/4096/6144/8192-bit	Key Wrap/Unwrap
	PBKDF[5]	NIST SP800-132	HMAC SHA2-224/256/384/512 and HMAC SHA3-224/256/384/512	Key Derivation Function
	RSA	FIPS PUB 186-4	Key Generation (186-4: 2048/3072/4096/6144/8192[6] bits), Signature Generation (186-4: PKCS1 v1.5/PSS – 2048/3072/4096/6144/8192 bits), Signature Verification (186-4: PKCS1 v1.5/PSS –2048/3072/4096/6144/8192 bits), Signature Verification (186-2: PKCS1 v1.5/PSS – 2048/3072/4096 bits).	Gen/Sign/Verify
	RSA (CVL)	FIPS PUB 186-4 SP800-56Brev2	Signature Primitive Decryption Primitive	
	SHS	FIPS PUB 180-4 (SHA-1 and SHA-2 functions)	SHA-1 SHA2-224/SHA3-224 SHA2-256/SHA3-256 SHA2-384/SHA3-384 SHA2-512/SHA3-512	Hashing

[5] Keys derived using PBKDF2 shall only be used in storage applications. The minimum password length allowed is 4, alpha-numeric characters. This puts the probability of the password being guessed at 1 in 81,450,625. A larger, more complex password is recommended to further decrease the probability of the password being guessed. The minimum salt length is 128 bits, which is randomly generated. For general purpose key derivation, the minimum iteration count is 1,000. For ULP/SLP and UWK/SWK derivation, the minimum iteration count is 98,304. For Split Key Backup keys, the minimum iteration count is 100,000.

[6] Per IG section A.14 and FIPS 186-4 section B.3.1, RSA keys with modulus sizes larger than 4096 are generated according to method A. Primes p and q are generated according to B.3.3 and C.3.1.

		FIPS PUB 202 (SHA-3 functions)		
	DRBG	SP 800-90A	AES-CTR-256	Random Bit Generation
N/A	CKG	SP 800-133rev2	Section 4 Option 1 (Symmetric keys and seed values for Asymmetric keys)	Cryptographic Key Generation

Table 6 – Approved Algorithms (HSM Crypto Library)

CAVP Cert #	Algorithm	Standard	Mode/ Method/ Key Lengths, Curves, or Moduli	Use
A1468	AES	NIST SP800-38F	KW (128, 192, 256) KWP (128, 192, 256)	Authenticated Encryption/Decryption
	KBKDF	SP800-108	KDF Mode: Counter MAC Modes: HMAC SHA2-224/256/384/512, HMAC SHA3-224/256/384/512 and AES CMAC (128/192/256 bit)	Key Based Key Derivation
	KDA	SP800-56C r1	Auxiliary Function: SHA2-224/256/384/512, SHA3-224/256/384/512	Key Derivation
	KTS	FIPS 140-2 IG D.9	<u>AES Cert. #A1468 (key establishment methodology provides between 128 and 256 bits of encryption strength):</u> AES KW and AES KWP (128, 192 and 256-bit) <u>AES Cert. #A1468 and HMAC Cert. #A1438:</u> AES 256 KWP	Key Wrapping / Unwrapping

Table 7 – Approved Algorithms (HSM P11 Crypto Extensions)

CAVP Cert #	Algorithm	Standard	Mode/ Method/ Key Lengths, Curves, or Moduli	Use
20	SHS	FIPS PUB 202 (SHA-3 functions)	SHA3-384	Hashing

Table 8 – Approved Algorithms (Xilinx SHA3/384 Library Implementation)

CAVP Cert #	Algorithm	Standard	Mode/ Method/ Key Lengths, Curves, or Moduli	Use

N/A	ENT (P)	SP 800-90B	N/A	Entropy Source[7]
C1777	DRBG	SP 800-90A	AES-CTR-128	Random Bit Generation

Table 9 – Approved Algorithms (ATECC608B Implementation)

The following non-Approved but Allowed algorithms are implemented when the module has been configured to operate in FIPS-approved mode.

Algorithm	Standard	Mode/ Method/ Key Lengths, Curves, or Moduli	Use
ECDSA	FIPS 186-4	• Brainpool r1 and t1 curves at sizes 224, 256, 320, 384, and 512 • Ed25519 • Ed448	Key Generation Signature Generation Signature Verification IG A.2, D.8 Scenario X2
EC Diffie-Hellman	SP 800-56Arev3, FIPS 140-2 IG A.2, D.8 Scenario X2	• Brainpool r1 and t1 curves at sizes 224, 256, 320, 384, and 512 • Curve25519 • Curve448	Key Agreement
AES (no security claimed)	SP 800-38A	• ECB (256 bit)	Obfuscation of stored CSP (Tamper Key) IG 1.23 Scenario 1

Table 10 – Allowed Algorithms

The Approved Key Transport schemes per IG D.9 are as follows:

- KTS (AES Cert. #A1438): AES CCM 256-bit
- KTS (AES Cert. #A1468; key establishment methodology provides between 128 and 256 bits of encryption strength): AES KW and AES KWP (128, 192 and 256-bit)
- KTS (AES Cert. #A1438 and AES Cert. #A1438): AES 256 CTR and AES 128 CMAC
- KTS (AES Cert. #A1468 and HMAC Cert. #A1438): AES 256 KWP
- KTS-RSA (Cert. #A1438; key establishment methodology provides between 112 and 192 bits of encryption strength)

[7] Per SP800-90B, the initial entropy estimate of a binary noise source is calculated as *HI =min(Horiginal, Hsubmitter).* As a result, *HI =min(0.874572, 0.5268) = 0.5268***.** The noise source is estimated to provide a full 128 bits of entropy for each 128-bit sample output from the noise source.

Non-FIPS Approved security functions/algorithms are not available for use when the module has been configured to operate in FIPS-approved mode. The following functions are only available in the non-Approved mode:

Non-Approved Security Functions
Triple DES ECB/CBC key gen, Encrypt/Decrypt
(PKCS11) AES CBC KW 128/192/256 bit Encrypt/Decrypt
HMAC SHA-1 Key derivation function
SHA-1 KDA Auxiliary function
RSA SHA1 (PKCS#1v1.5, PSS)
RSA X9.31
RSA 1024 bit
RSA Sign/Verify with message recovery
ECDSA SHA1
SM2 P-192, BP-160, BP-192
SM3
SM3 HMAC
SM3 Key derivation function
SM4 CTR/CBC/ECB 128 bit Encrypt/Decrypt
SM4 CMAC 128 bit Sign and Verify
SM4 CCM 128 bit Encrypt/Decrypt

Table 11 – Non-Approved Security functions

8.2 Cryptographic Keys and CSPs

The cryptographic keys and CSPs used by the module are described in
Table 12. Each CSP is stored in one or more of the following locations:

- Token NVM (NVM): the SD card filesystem.
- TSMRAM: MSP430 Token Security Module RAM.
- Token RAM: per-client-connection memory allocated/initialized after a client application establishes a secure channel with the HSM.
- Session RAM: per (PKCS11) session RAM allocated/initialized each time a client application instantiates a new PKCS11 session within a secure channel via API call C_OpenSession.
- Ephemeral RAM: stack memory.

CSPs in the above locations can be zeroized or otherwise rendered invalid by the following events:

- Power cycle: removal of PCI bus power.
- Tamper: via either breach of the tamper enclosure or API command A_Tamper.
- Logout: Exiting an authenticated role via the API command C_Logout
- Loss of client connection: closure of the socket connection from the client, such as via API command C_Finalize.
- Token deletion: removal of the token and its CSPs via API command A_DelSlot
- Token initialization: CO reset of the state of a token via API command C_InitToken
- Password change: a change of a role password via API commands C_InitPIN, C_SetPIN, C_InitToken

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
Token (Object) Encryption Key (TEK)	Each unique TEK is used to encrypt one PTO-SK or PTO-AK (one-to-one mapping)	AES GCM 256 bit	Generated using the SP800-90A AES CTR DRBG	Token NVM (encrypted with TMK) Ephemeral RAM (plaintext)	NVM: invalidated upon TMK zeroization (token deletion or tamper).

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
			Neither input nor output		ER: zeroized upon the completion of an operation (e.g. load/store of the PTO from/to NVM).
Token Master Key (TMK)	Used to encrypt TEK (one master key per token)	AES KW 256 bit	Generated using the SP800-90A AES CTR DRBG Neither input nor output	Token NVM (this key is encrypted and stored twice: once with SWK and separately using UWK) Session RAM (plaintext)	NVM: invalidated upon SWK and UWK zeroization (on token deletion or during tamper event). SR: zeroized upon power cycle, or at session logout.
User/SO PIN (UPIN/SPIN)	Used for authentication of User/SO and derivation of UWK/SWK	Password must be at least four (4) valid characters as defined in Section 5.2.1.	Input via KTS during login (ECDH secure channel (host-to-card)	Ephemeral RAM (plaintext)	Zeroized upon the completion of an operation (e.g. authentication).
User/SO Login Code (ULC/SLC)	Used to authenticate User/SO by comparing the previously stored value	Output of SP800-132 PBKDF	Derived using SP800-132 PBKDF and written to Token NVM upon successful PIN initialization or change and verified during login.	Token NVM (Protected in accordance with IG 7.16) Token RAM (plaintext)	NVM: zeroized on token deletion, initialization, password change; invalidated on tamper. TR: zeroized upon power cycle. Also, as per NVM above. ER: zeroized upon the completion of an operation (e.g. comparison, password change). Also, as per NVM above.

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
			Neither input nor output.	Ephemeral RAM (plaintext, computed)	
User Wrap Key/SO Wrap Key (UWK/SWK)	Used to wrap/unwrap TMK (one per token)	AES KW 256 bit	Derived using SP800-132 PBKDF during login. Neither input nor output	Session RAM (plaintext) Ephemeral RAM (plaintext)	SR: zeroized upon logout, change of authorization state, loss of client connection. ER: zeroized upon the completion of an operation (e.g. authentication, password change)
Tamper Key	Key used to obfuscate SP 800-132 salt value	AES-256-ECB	Generated using the SP800-90A AES CTR DRBG Neither input nor output	TSMRAM (Plaintext)	TSMRAM: zeroized on tamper event.
Split Password	Used for derivation of the split key encryption key used to encrypt the PTO-SK and PTO-AK key splits for Slot Backup (One password per split)	Password must be at least four (4) valid characters as defined in Section 5.2.1.	Input in via KTS during operator input (ECDH secure channel (host-to-card).	Ephemeral RAM (plaintext)	Zeroized upon the completion of an operation (e.g. key derivation).

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
Split Key Parameters	Used as input to SP800-132 PBKDF for generating PTO-SK and/or PTO-AK splits (one set of parameters per split)	Salt value	Generated using the SP800-90A AES CTR DRBG. Input from and output to smart cards via USB (IG 2.1).	Ephemeral RAM (plaintext)	Zeroized upon the completion of an operation (e.g. key derivation).
Split Key Encryption/Decryption Key	Used to AEAD encrypt and decrypt split PTO-SK and/or PTO-AK backups (one per split)	AES CCM 256 bit	Key derived using SP800-132 PBKDF during Slot Backup and Slot Restore operations	Ephemeral RAM (plaintext)	Zeroized upon the completion of an operation (key wrapping).
Plaintext Token Object, Asymmetric private key (PTO-AK)	User token private key used to sign, verify, wrap, unwrap, or perform key agreement.	ECC: P-224, P-256, P-384, P-521, Brainpool r1 and t1 curves at sizes 224, 256, 320, 384, and 512, Ed25519, Ed448	Generated using the SP800-90A AES CTR DRBG and FIPS PUB 186-4 Encrypted key splits are input and output to smart cards	Session RAM (plaintext) Ephemeral RAM (plaintext) Token NVM (encrypted with TEK)	SR: deletion of the object, session closure, logout, loss of client connection, or tamper event. ER: completion of the operation (e.g. sign). NVM: invalidated upon TEK zeroization (deletion or during tamper event)

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
		FFC: 2048-bit 3072-bit 4096-bit 6144-bit 8192-bit RSA: 2048 bit 3072 bit 4096 bit 6144 bit 8192 bit	via USB (IG 2.1) Input and output via KTS (IG D.9)		
Plaintext Token Object, Symmetric key (PTO-SK)	User token symmetric key used to encrypt, decrypt, wrap, unwrap, generate MAC, verify MAC, or perform key derivation.	AES: 128 bit 192 bit 256 bit HMAC: SHA-1 SHA-224 SHA-256 SHA-384 SHA-512	Generated using the SP800-90A AES CTR DRBG Encrypted key splits are input and output to smart cards via USB (IG 2.1)	Session RAM (plaintext) Ephemeral RAM (plaintext) Token NVM (encrypted with TEK)	SR: deletion of the object, session closure, logout, loss of client connection, or tamper event. ER: completion of the operation (e.g. encrypt). NVM: invalidated upon TEK zeroization (deletion or during tamper event)

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
			Input and output via KTS (IG D.9)		
ECC Secure Channel Private Key/Public Key	Asymmetric keys used to derive Secure Channel ECDH Shared Secret	ECDSA P-521	Generated using the SP800-90A AES CTR DRBG and FIPS PUB 186-4	Ephemeral RAM (plaintext)	Zeroized upon power cycle or at the end of a session.
ECDH Secure Channel Session Key	Symmetric keys used to encrypt session data	Secure Channel: AES-256-CTR and AES-128-CMAC	Derived according to SP 800-56Crev1	Session RAM (plaintext)	Zeroized upon power cycle or at the end of a secure channel session.
ECDH Secure Channel Shared Secret	Shared Secret Computation (Secure Channel)	Shared Secret	Generated according to SP 800-56Arev3	Ephemeral RAM (plaintext)	Zeroized immediately after deriving ECDH Secure Channel Session.
ECDH Shared Secret	Shared Secret Computation (User-defined)	Shared Secret	Generated according to SP 800-56Arev3	Session RAM (plaintext) Ephemeral RAM (plaintext) Token NVM (encrypted with TEK)	SR: deletion of the object, session closure, logout, loss of client connection, or tamper event. ER: completion of the operation (e.g. encrypt). NVM: invalidated upon TEK zeroization (deletion or during tamper event)

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
DH Shared Secret	Shared Secret Computation (User-defined)	Shared Secret	Generated according to SP 800-56Arev3	Session RAM (plaintext) Ephemeral RAM (plaintext) Token NVM (encrypted with TEK)	SR: deletion of the object, session closure, logout, loss of client connection, or tamper event. ER: completion of the operation (e.g. encrypt). NVM: invalidated upon TEK zeroization (deletion or during tamper event)
User Defined KDF Derived Key	Output of SP 800-56Crev1 KDF	User Defined: AES, HMAC, CMAC	Derived according to SP 800-56Crev1	Session RAM (plaintext) Ephemeral RAM (plaintext) Token NVM (encrypted with TEK)	SR: deletion of the object, session closure, logout, loss of client connection, or tamper event. ER: completion of the operation (e.g. encrypt). NVM: invalidated upon TEK zeroization (deletion or during tamper event)
ECIES Session Keys	Symmetric keys used to encrypt and authenticate remote token export/import	AES-256-KWP and HMAC-SHA-512	Derived according to SP 800-56Crev1	Ephemeral RAM (plaintext)	Zeroized at the end of the transaction (wrap, etc.).
DRBG Entropy Input String	Random bit generation	DRBG input	Internally Generated from hardware	Ephemeral RAM (plaintext)	Zeroized upon DRBG consumption of input string.

Keys / CSP	Description	Key / CSP Type	Input/Output	Storage	Zeroization
			sources		
DRBG Seed	Random bit generation	DRBG input	Internally Generated from hardware sources	Ephemeral RAM (plaintext)	Zeroized upon DRBG consumption of seed.
DRBG V (DRBG-V)	Random bit generation	Internal state value	Internal value used as part of SP 800-90a CTR_DRBG	Ephemeral RAM (plaintext)	Zeroized upon power cycle or end of client connection.
DRBG Key (DRBG-K)	Random bit generation	Internal state value	Internal value used as part of SP 800-90a CTR_DRBG	Ephemeral RAM (plaintext)	Zeroized upon power cycle or end of client connection.
Manufacturer Installed Keys/CSPs					
Firmware Update Key	Used for the verification of the firmware update package	RSA 4096-bit with SHA3-384 hash	Input during manufacturing and not Output.	Boot ROM (plaintext)	N/A

Table 12 – Approved Keys and CSPs Table

8.3 Cryptographic Key Zeroization

All plaintext keys and CSPs are zeroized within the HSM when one of the following actions occur:

- By removing the external RTC Battery.
- By receiving a user-defined Tamper Signal.
- By executing the command "CCconf tamper".
- Any breach to the metal enclosure of the module occurs.
- If the module goes outside the normal operational (voltage/temperature) conditions.

In all the above cases, all plaintext cryptographic keys and CSPs are Zeroized.

The transition of the module from FIPS mode to non-FIPS mode or vice-versa causes an implied tamper event such that all keying material is lost during the transition thus zeroizing all plaintext keys and CSPs of the module before entering or exiting FIPS mode of operation.

9. Self-Tests

FIPS 140-2 requires the module to perform self-tests to ensure the module integrity and the correctness of the cryptographic functionality at start-up. Some functions also require conditional tests during normal operation of the module.

If any of the power-on self-tests fail, the module enters an error state where no cryptographic functions can be executed. The module will automatically tamper, resulting in zeroization of all stored CSPs and returning the module to a factory default uninitialized state.

If the module encounters an error in the conditional self-tests (failed upgrade, noise source error, etc.) the module will enter a soft error state where the requested command fails. The error condition may be cleared by re-executing the service which prompted the conditional self-test.

If the error condition is not cleared, then the module is considered to be malfunctioning and should be returned to the manufacturer.

9.1 Power-On Self-Tests

Power-on self-tests are run upon the initialization of the module and do not require operator intervention to run.

The module implements the following startup tests:

- SP800-90B RCT and APT Health tests - the startup test runs over 1024 samples which meets the SP800-90B requirement.

The module implements the following integrity test:

- Firmware Integrity test using a SHA3-384 EDC.

The module implements the following power-on self-tests for the HSM Crypto Library:

- AES-ECB KAT (Encrypt – 128, 192 and 256 bits)
- AES-ECB KAT (Decrypt – 128, 192 and 256 bits)
- AES-CBC KAT (Encrypt – 128 and 256 bits)
- AES-CBC KAT (Decrypt – 128 and 256 bits)
- AES-CCM KAT (Encrypt – 128, 192 and 256 bits)
- AES-CCM KAT (Decrypt – 128, 192 and 256 bits)
- AES-GCM KAT (Encrypt – 128 and 192 bits)
- AES-GCM KAT (Decrypt – 128 and 192 bits)
- SHA-1 KAT
- SHA2-224 KAT
- SHA2-256 KAT
- SHA2-384 KAT
- SHA2-512 KAT

- SHA3-224 KAT
- SHA3-256 KAT
- SHA3-384 KAT
- SHA3-512 KAT
- HMAC-SHA-1 KAT
- HMAC-SHA-224 KAT
- HMAC-SHA-256 KAT
- HMAC-SHA-384 KAT
- HMAC-SHA-512 KAT
- AES-256 CTR DRBG KAT
  - SP800-90A Section 11 health tests
- SP800-56A rev3 KAT for ECC and FFC
  - ECDH primitive "Z" KAT (Curves used: P-256, P-384 and P-521)
  - DH primitive "Z" KAT (Moduli: 2048 bit)
- PBKDF KAT
- ECDSA P-521 Sign/Verify PCT
- RSA 2048-bit modulus using PKCS1 v1.5 Sign/Verify KAT
- RSA 2048-bit modulus using OAEP Encrypt/Decrypt KAT

The module implements the following power-on self-tests for the HSM P11 Crypto Extensions:

- AES Key Wrap KAT (Encrypt – 128, 192 and 256 bits)
- AES Key Wrap KAT (Decrypt – 128, 192 and 256 bits)
- AES KWP KAT (Encrypt – 128, 192 and 256 bits)
- AES KWP KAT (Decrypt – 128, 192 and 256 bits)
- KBKDF in counter mode KAT
- One Step KDA (SP800-56C r1) KAT

The module implements the following power-on self-tests for the ATECC608B DRBG:

- AES-128 CTR DRBG KAT
  - SP800-90A Section 11 health tests

9.2 Conditional Self-Tests

Conditional self-tests are tests that run during operation of the module. The module performs the following conditional self-tests:

Type	Test Description
Pairwise-consistency Test	Whenever an RSA and ECDSA key pair of any valid size is generated on the HSM, before the operation is completed and the keys are made available for use to the operator, a pair-wise consistency test is executed on the key pair.
Repetition Count Test on Entropy	This test is intended to identify if the noise source is repeating a given value continuously. The test is implemented per the details of SP800-90B section

Source	4.4.1.
Adaptive Proportion Test on Entropy Source	The test continuously measures the local frequency of occurrence of a sample value in a sequence of noise source samples to determine if the sample value occurs too frequently. The test is implemented per the details of SP800-90B section 4.4.2.
Firmware Load Test	When firmware is updated on the HSM, the update image must be validated before the underlying firmware on the device is updated. This is accomplished through an "SHA256 RSA 4096" signature validation on the update image.

Table 13 – Conditional Self-tests

10. EMI/EMC

The module conforms to the EMI/EMC requirements specified by part 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class B.

11. Guidance and Secure Operation

11.1 Crypto Officer Guidance

To fully utilize the CyberCogs HSM, a user will need to use the provided Cryptoki Library that is PKCS11 compliant, an administration application (CCconf), and, optionally, smart cards.

To initialize the module into the FIPS approved mode, perform the following steps:

1. Enter FIPS mode by running "CCconf set-ci=policy:FIPS"
2. Enter the default administrative token PIN
3. Initialize the administrative token by running "CCconf init-token=0"
4. Set a new administrative token (ATS) PIN on the newly created administrative token by running "CCconf set-so-pin=0"
5. Set a new administrative token user (ATU) PIN on the newly created administrative token by running "CCconf init-user=0"
6. Create a new non-administrative token by running "CCconf create=1"
7. Initialize the new token by repeating steps 2-5 but replacing the '0' value with '1'

11.2 Operator Guidance

The PKCS11 interface may be managed by module operators using the "CCconf" command line interface. CCconf is a CyberCogs-developed CLI provided with the module for ease of use, which abstracts the module's PKCS11 API interface to the end-user or operator. A reference for the module's PKCS11 API may be found in the PKCS #11 Cryptographic Token Interface Base Specification Version 2.40.

11.3 Verifying Module Status

The module implements a persistent indicator that provides the operator with assurance that the module is running in a FIPS Approved mode of operation. To query the FIPS Approved mode indicator, run "CCconf query=ciPolicy" command. It will output the following policy flags:

- +PolicyLock +FipsAlgorithms +FwUpTamper -CrmTamperProof +RequireLogin +NoCrm +SecureChannel

If the status returned does not match the above, the module is not running in the FIPS Approved mode and should be reinitialized.

11.4 Module Self-Tests

To verify whether the module has successfully run its power-on self-tests, run the "CCconf query=ciPostFailures" command. The output returned should be "none". If any other status is returned, the module is not running in the FIPS Approved mode and should be reinitialized.

To perform the module self-tests on-demand, run the "CCconf reset" command, which will re-initialize the module and re-execute the power-on self-tests.

11.5 Non-Approved Mode of Operation

If the steps outlined in section 11.1 above have not been followed, the module will be in a non-approved mode of operation where the non-approved algorithms defined in Table 11 will be available. When FIPS mode is enabled or disabled, the module will automatically tamper, meaning the CSPs will be zeroized, thus cannot be shared between approved and non-approved modes.

The module will perform the self-tests as described in section 9 regardless of whether the module is running in an approved or non-approved mode.

11.6 Zeroization

The module may be zeroized by either physically removing the RTC battery or by receiving a user-specified tamper signal on the jumper pins or by running the following command:

- CCconf tamper

Glossary

Term	Description
AES	Advanced Encryption Standard
ARM	Advanced RISC Machine
ATS	Admin Slot/Token Security Officer
ATU	Admin Slot/Token User
CAVP	Cryptographic Algorithm Validation Program
CKG	Cryptographic Key Generation
CMVP	Cryptographic Module Validation Program
CRNGT	Continuous Random Number Generator Test
CSP	Critical Security Parameter
CTR	Counter
DRBG	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
EMI/EMC	Electromagnetic Interference / Electromagnetic Compatibility
FIPS	Federal Information Processing Standards
FPGA	Field Programmable Gate Array
GCM	Galois/Counter Mode
HMAC	Hashed Message Authentication Code
HSM	Hardware Security Module
IG	Implementation Guidance
IV	Initialization Vector
KAS-SSC	Key Agreement Scheme-Shared Secret Computation
KAT	Known Answer Test
KBKDF	Key-Based Key Derivation Function
KDA	Key Derivation Algorithm
KDF	Key-Derivation Function
KTS	Key-Transport Scheme
LED	Light Emitting Diode
NIST	National Institute of Standards and Technology
PBKDF	Password-Based Key Derivation Function
PCB	Printed Circuit Board
PCIe	Peripheral Component Interconnect Express
RSA	Rivest Shamir Adleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SO	Non-Admin Slot/Token Security Officer
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
USER	Non-Admin Slot/Token User

Table 14 – Glossary of Terms