VMware, Inc.
3401 Hillview Ave
Palo Alto, CA 94304, USA
Tel: 877-486-9273
Email: info@vmware.com
https://www.vmware.com

VMware's BoringCrypto Module

Software Version: 3.0

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level: 1
Document Version: 0.1

vmware

TABLE OF CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

1.1 Purpose

This is a non-proprietary Cryptographic Module Security Policy for the VMware's BoringCrypto Module from VMware, Inc. This Security Policy describes how the VMware's BoringCrypto 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 Canadian Centre of Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) website at https://csrc.nist.gov/projects/cryptographic-module-validation-program.

This document also describes how to run the module in a secure FIPS-Approved mode of operation. The VMware's BoringCrypto Module is also referred to in this document as “the module”.

1.2 Reference

This document deals only with operations and capabilities of the composite 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 VMware website (http://www.vmware.com) contains information on the full line of products from VMware.
- The CMVP website (https://csrc.nist.gov/Projects/Cryptographic-Module-Validation-Program/Validated-Modules/Search) contains options to get contact information for individuals to answer technical or sales-related questions for the module.

2 VMWARE'S BORINGCRYPTO MODULE

2.1 Introduction

VMware, Inc., a global leader in virtualization, cloud infrastructure, and business mobility, delivers customer-proven solutions that accelerate Information Technology (IT) by reducing complexity and enabling more flexible, agile service delivery. With VMware solutions, organizations are creating exceptional experiences by mobilizing everything, responding faster to opportunities with modern data and apps hosted across hybrid clouds, and safeguarding customer trust with a defense-in-depth approach to cybersecurity. VMware enables enterprises to adopt an IT model that addresses their unique business challenges. VMware's approach accelerates the transition to solutional-computing while preserving existing investments and improving security and control.

2.1.1 VMware's BoringCrypto Module

The VMware's BoringCrypto Module is a software cryptographic module that is built from the BoringCrypto source code according to the instructions prescribed in Section 3. The module is a software library that provides cryptographic functions to BoringSSL and various VMware applications via a well-defined C-language application program interface (API). The module only performs communications with the calling application that invokes the module services.

The VMware's BoringCrypto 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	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A[1]
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC[2]	1
9	Self-tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The VMware's BoringCrypto Module is a software cryptographic module with a multiple-chip standalone embodiment. The overall security level of the module is 1. The software version of the module is 3.0 and is built from ae223d6138807a13006342edfeef32e813246b39 version of the BoringCrypto source code.

[1] N/A – Not Applicable

[2] EMI/EMC – Electromagnetic Interference/Electromagnetic Compatibility

2.2.1 Physical Cryptographic Boundary

As a software module, there are no physical protection mechanisms implemented. Therefore, the module must rely on the physical characteristics of the host system. The module runs on a General-Purpose Computer (GPC) and the physical boundary of the cryptographic module is defined by the hard enclosure around the host system on which it runs. The module supports the physical interfaces of the GPC. See Figure 1 below for a block diagram of the typical GPC and its physical cryptographic boundary marked with red dotted line.

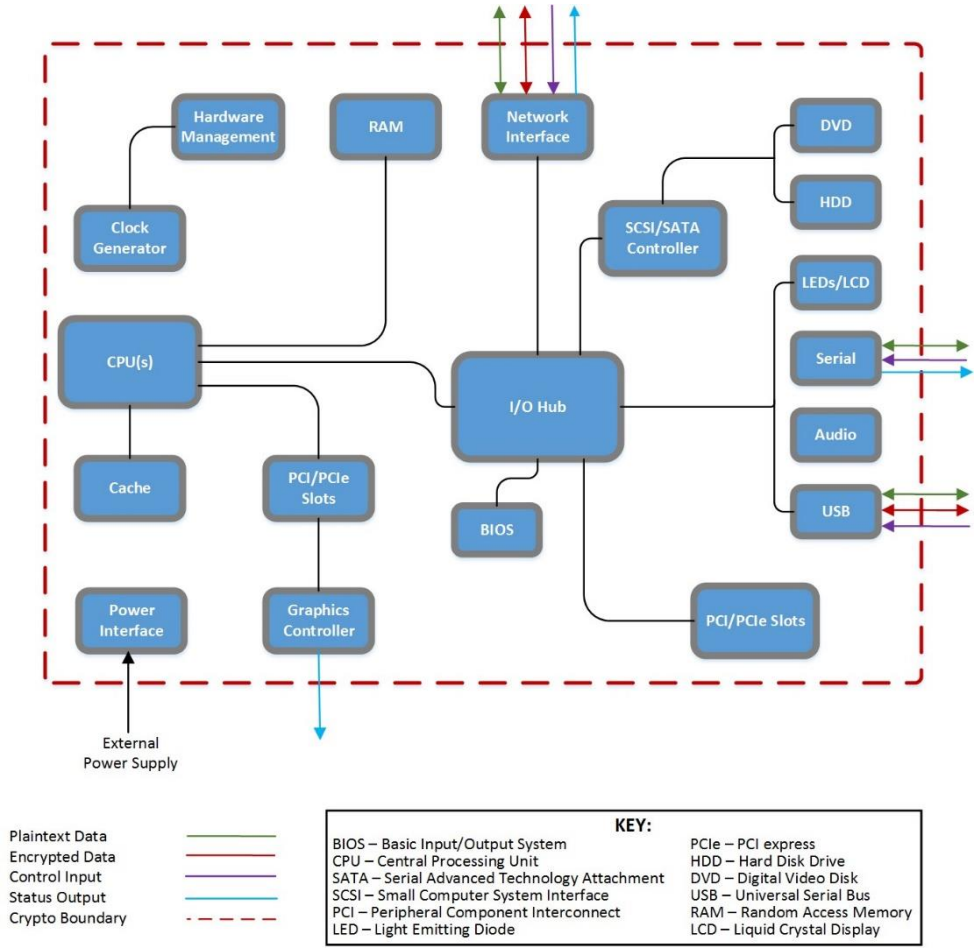

Figure 1 – Hardware Block Diagram

2.2.2 Logical Cryptographic Boundary

The logical cryptographic boundary of the module is defined by the binary file bcm.o, an object file that is statically linked to BoringSSL. The module only performs communications with the calling application that invokes the module services and the host OS.

Figure 2 and Figure 3 depict the logical cryptographic boundary of the module. The module's logical boundary is a contiguous perimeter that surrounds all memory-mapped functionality provided by the module when loaded and stored in the host platform's memory.

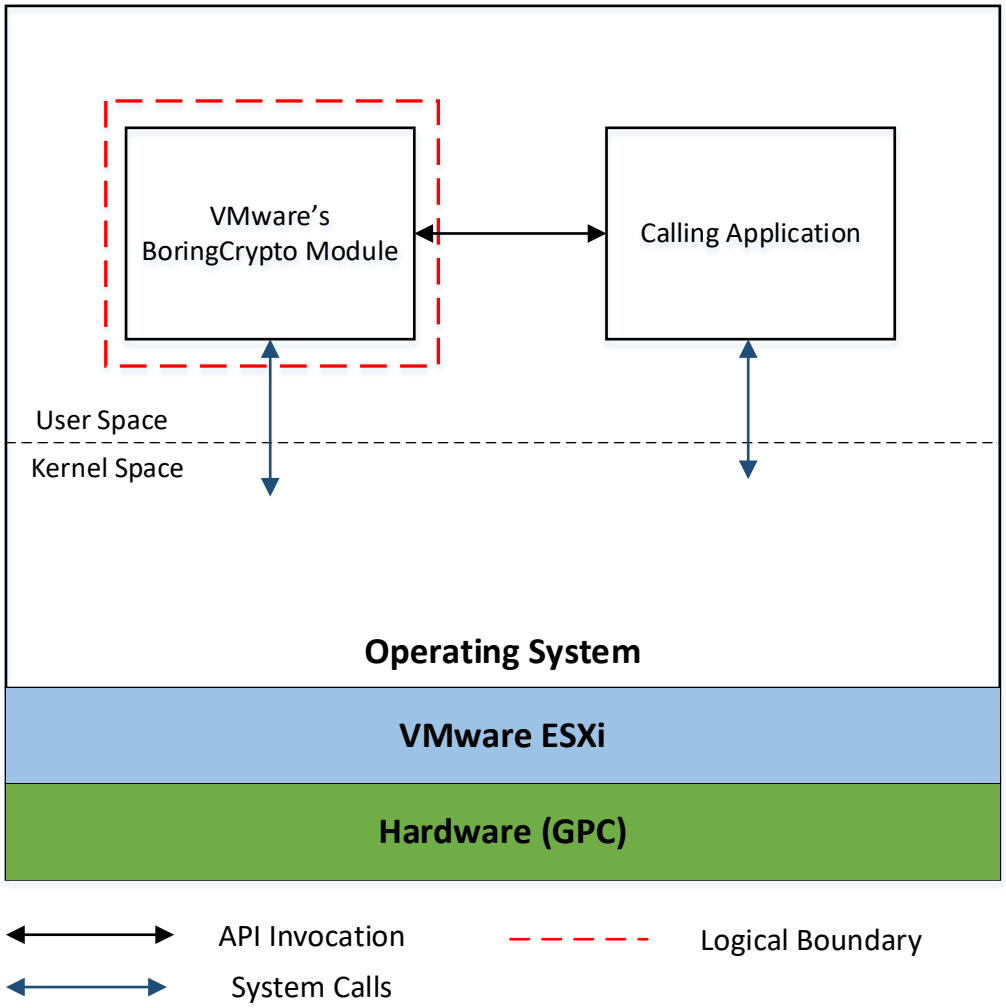

Figure 2 – Module's Logical Cryptographic Boundary in Guest OS

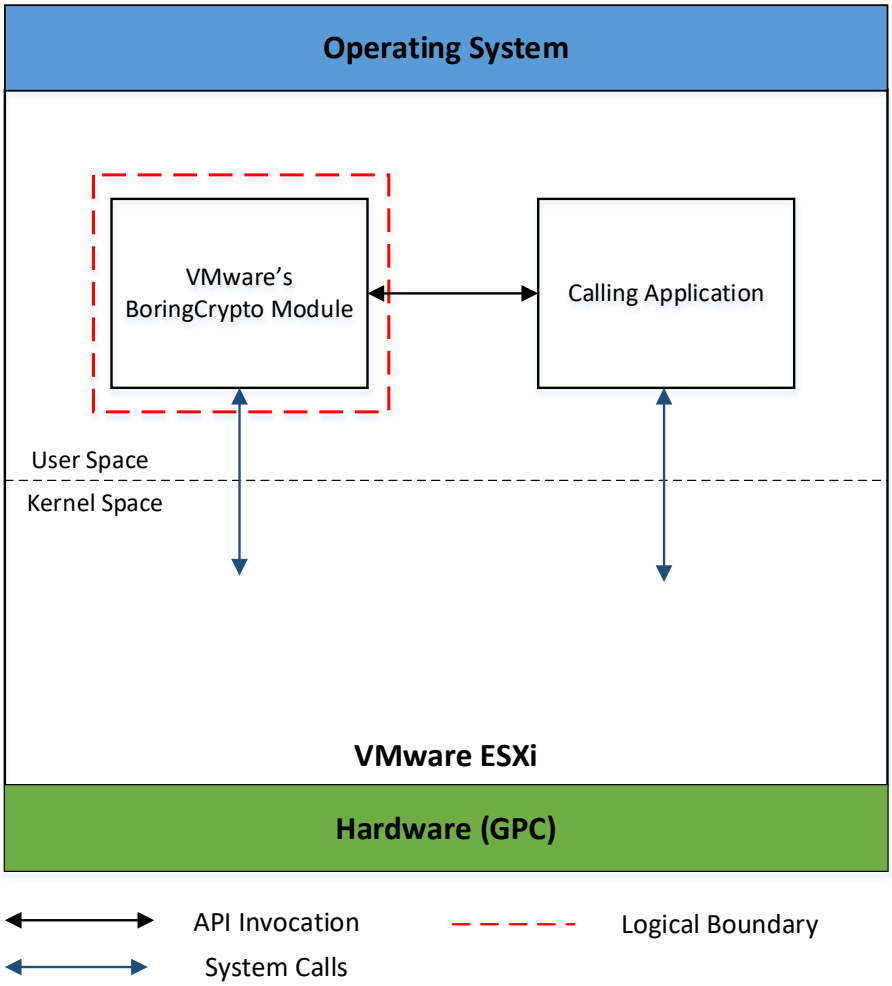

Figure 3 – Module's Logical Cryptographic Boundary in Hypervisor

2.2.3 Cryptographic Implementation and modes of operation

The module supports a FIPS approved mode of operation as well as a non-approved mode of operation. Successful completion of the self-tests and usage of approved security functions results in the module operating in the FIPS mode, whereas usage of any non-approved security function results in the module entering the non-FIPS mode of operation.

The module implements the FIPS-Approved algorithms listed in Table 2 below.

Table 2 – FIPS-Approved Algorithm Implementations

Function	Algorithm	Options	Cert #
Random Bit Generation	[SP 800-90Arev1] DRBG	CTR_DRBG (AES-256)	A1231

Key Generation	[SP 800-133] CKG	N/A	N/A (Vendor Affirmed)
Key Derivation	[SP 800-135rev1] CVL	TLS 1.0/1.1 and 1.2 KDF	A1231
Encryption, Decryption	[SP 800-67] Triple-DES	3-Key Triple-DES TECB, TCBC	A1231
Encryption, Decryption, Key Wrapping, Key Unwrapping, Authentication	[FIPS 197] AES [SP 800-38A] ECB, CBC, CTR [SP 800-38F] KW, KWP [SP 800-38D] GCM/GMAC	128/192/256 ECB, CBC, CTR; GCM/GMAC; KW, KWP	A1231
Signature Generation Component, Key Pair Generation, Signature Generation, Signature Verification, Public Key Validation	[FIPS 186-4] ECDSA	P-224/P-256/P-384/P-521	A1231
Digital Signature Generation, Digital Signature Verification, non-Digital Signature Applications	[FIPS 180-4] SHS	SHA-1, SHA-2 (224, 256, 384, 512)	A1231
Generation, Authentication	[FIPS 198-1] HMAC	HMAC with SHA-1, SHA-2 (224, 256, 384, 512)	A1231
Key Agreement Scheme – Shared Secret Computation per SP 800-56Arev3 and Key Derivation per SP 800-135 TLS KDF (CVL Cert. #A1231)	[SP 800-56Ar3] KAS-SSC	Ephemeral Unified EC Diffie-Hellman P-224/P-256/P-384/P-521, SHA-2 (224, 256, 384, 512)	N/A (Vendor Affirmed)

Key Generation, Signature Generation, Signature Verification	[FIPS 186-4] RSA	RSA (1024[3], 2048, 3072 bits)	A1231

The module also employs the following key establishment methodology/algorithms, which are non-approved but allowed to be used in FIPS-Approved mode of operation:

- RSA (key wrapping[4]; key establishment methodology provides between 112 and 256 bits of encryption strength)
- MD5 used only in the context of the TLS protocol (versions 1.0 and 1.1)
- NDRNG used only to seed the Approved DRBG

The module implements the following non-approved and non-compliant algorithms and services. The use of these are not allowed in FIPS-approved mode of operation. Please refer to

Table 3 below for the list of non-Approved algorithms and associated services.

Table 3 – Non FIPS-Approved Algorithm Implementations

Algorithm	Description/Service
MD5, MD4	Digest Generation
DES	Encryption and Decryption
AES-GCM/GMAC (non-compliant)	Encryption, Decryption and Authentication
AES (non-compliant)	Encryption and Decryption
RSA (non-compliant)	Digital Signature Generation and Asymmetric Key Generation
ECDSA (non-compliant)	Digital Signature Generation and Asymmetric Key Generation
POLYVAL	Digest Generation
Triple-DES (non-compliant)	Encryption and Decryption

2.3 Module Interfaces

The module's logical interfaces exist at a low level in the software as APIs. Both the APIs and physical interfaces can be categorized into the following interfaces defined by FIPS 140-2:

- Data input
- Data output
- Control input

[3] Only applicable to FIPS 186-4 RSA2 SigVer.

[4] No claim is made for SP 800-56B compliance, and no CSPs are established into or exported out of the module using this service.

- Status output
- Power input

As a software module, the module's manual controls, physical indicators, and physical and electrical characteristics are those of the host platform. A mapping of the FIPS 140-2 logical interfaces, the physical interfaces, and the module interfaces can be found in Table 4 below.

Table 4 – FIPS 140-2 Logical Interface Mapping

FIPS Interface	Physical Interface	Module Interface (API)
Data Input	Network port, Serial port, USB port, SCSI/SATA Controller	The function calls that accept input data for processing through their arguments.
Data Output	Network port, Serial port, USB port, SCSI/SATA Controller	The function calls that return by means of their return codes or argument generated or processed data back to the caller.
Control Input	Network port, Serial port, USB port, Power button	The function calls that are used to initialize and control the operation of the module.
Status Output	Network port, Serial port, USB port, Graphics controller	Return values for function calls; Module generated error messages.
Power Input	AC Power socket	Not applicable.

2.4 Roles, Services and Authentication

There are two roles supported by the module (as required by FIPS 140-2) that operators may assume: a Crypto Officer (CO) role and a User role. Authentication is not supported by the module. Roles are implicitly assumed by the entity accessing services provided by the module.

Only one role may be active at a time and the module does not allow concurrent operators. Each role and their corresponding services are detailed in the sections below. Please note that for the keys and Critical Security Parameters (CSPs) listed in Table 5 below, the types of access required have been indicated using the following notations:

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

2.4.1 Crypto Officer and User Roles

The CO and User roles share many services.

- Crypto Officer Role: Installation and initialization of the module on the host computer system and calling of any API functions.
- User Role: Loading of the module and calling any of the API functions.

Below, Table 5 describes the CO and User services and CSP access, while Table 3 in Section 2.2.3 above describes the Non-Approved algorithms and services.

Table 5 – Crypto Officer and Users Services

Role	Service	Description	CSP and Type of Access
CO	Initialization of the module	Initialization of the module following the Secure Operation section of the Security Policy	None
CO, User	Run self-test	Runs Self-tests on demand during module operation	None
CO, User	Show status	Show the status of the module	None
CO, User	Zeroize	Zeroizes all CSPs	All CSPs - WX
CO, User	Random bit generation	Generate random bits	CTR_DRBG CSPs (seed, internal state – key and V) – WX
CO, User	Key generation	Generate RSA and ECDSA private key. RSA, ECDSA, CTR_DRBG approved functions used.	RSA Private Key – WX ECDSA Private Key – WX
CO, User	Symmetric Encryption/Decryption	Encrypt or decrypt data using supplied key and algorithm specification (key passed in by the calling process)	AES Key – X Triple-DES Key – X
CO, User	Hash generation	Compute and return a message digest using SHA algorithm	None
CO, User	Message Authentication Code generation (HMAC)	Compute and return a hashed message authentication code	HMAC Key – X
CO, User	Key Transport	Wrap/unwrap a key on behalf of the calling application but does not establish keys into the module (key passed in by the calling process)	RSA Private Key – WX
CO, User	Key Agreement	Computation of shared secret on behalf of the calling application	EC DH Private Key – WX
CO, User	Signature Generation and Verification	Generate and verify RSA and ECDSA digital signatures (keys passed in by the calling process). RSA, ECDSA, CTR_DRBG approved functions used.	RSA Private Key – WX ECDSA Private Key – WX

Table 6 below list the non-Approved services provided by the module using the algorithms listed in Table 3.

Table 6 – Non-Approved Services

Role	Service	Non-Approved Functions
CO, User	Symmetric Encryption and Decryption	AES (non-compliant), DES, TDES (non-compliant)
CO, User	Hash	MD4, MD5, POLYVAL
CO, User	Signature Generation and Verification	RSA (non-compliant), ECDSA (non-compliant)
CO, User	Transport key	RSA (non-compliant)
CO, User	Key Generation	RSA (non-compliant), ECDSA (non-compliant),

Table 7 below list the non-Approved or non-security relevant services that are also provided by the module over a non-public interface.

Table 7 – Non-Approved and Non-Security Relevant Services

Role	Service	Non-Approved Functions
CO, User	Larger Integer Operations	None
CO, User	Disable automatic generation of CTR_DRBG "additional_input" parameter"	CTR_DRBG
CO, User	Wegman-Carter hashing with POLYVAL	None

2.5 Physical Security

The VMware's BoringCrypto Module is a software module, which FIPS defines as a multi-chip standalone cryptographic module. The module being a software module, does not include physical security mechanisms. Thus, the FIPS 140-2 requirements for physical security are not applicable.

2.6 Operational Environment

The module was tested and found to be compliant with FIPS 140-2 requirements on the following platforms:

Table 8 – Tested Operational Environments

#	Operating System (OS) on ESXi 7.0	Processor Family	Optimizations (Target)	GPC

1	Amazon Linux 2	Intel® Xeon Gold 6230R	AES-NI	Dell PowerEdge R740
2	Amazon Linux 2	Intel® Xeon Gold 6230R	None	
3	Photon OS 2.0	Intel® Xeon Gold 6230R	AES-NI	
4	Photon OS 2.0	Intel® Xeon Gold 6230R	None	
5	Photon OS 3.0	Intel® Xeon Gold 6126R	AES-NI	
6	Photon OS 3.0	Intel® Xeon Gold 6126R	None	
7	Photon OS 3.0	Intel® Xeon Gold 6230R	AES-NI	
8	Photon OS 3.0	Intel® Xeon Gold 6230R	None	
9	Photon OS 4.0	Intel® Xeon Gold 6230R	AES-NI	
10	Photon OS 4.0	Intel® Xeon Gold 6230R	None	
11	RHEL 7.9	Intel® Xeon Gold 6230R	AES-NI	
12	RHEL 7.9	Intel® Xeon Gold 6230R	None	
13	RHEL 8.2	Intel® Xeon Gold 6230R	AES-NI	
14	RHEL 8.2	Intel® Xeon Gold 6230R	None	
15	Ubuntu 18.04	Intel® Xeon Gold 6230R	AES-NI	
16	Ubuntu 18.04	Intel® Xeon Gold 6230R	None	
17	Ubuntu 20.04	Intel® Xeon Gold 6230R	AES-NI	
18	Ubuntu 20.04	Intel® Xeon Gold 6230R	None	
19	Within ESXi 7.0 (as a host)	Intel® Xeon Gold 6126R	AES-NI	
20	Within ESXi 7.0 (as a host)	Intel® Xeon Gold 6126R	None	
21	Within ESXi 7.0 (as a host)	Intel® Xeon Gold 6230R	AES-NI	
22	Within ESXi 7.0 (as a host)	Intel® Xeon Gold 6230R	None	
#	**Operating System (OS)**	**Processor Family**	**Optimizations (Target)**	**GPC**
23	Ubuntu 20.04	Intel® Core i5	AES-NI	Dell PowerEdge R740
24	Ubuntu 20.04	Intel® Core i5	None	

Per IG G.5, VMware affirms that the module remains compliant with the FIPS 140-2 validation when operating on any general-purpose computer (GPC) provided that the GPC uses the specified single user operating system/mode specified on the validation certificate, or another compatible single user operating system. The CMVP allows vendor and user porting and re-compilation of a validated cryptographic module from the operational environment specified on the validation certificate to an operational environment which was not included as part of the validation testing as long as the porting rules are followed.

VMware, Inc. affirms that the VMware's BoringCrypto Module runs in its configured, Approved mode of operation on the following binary compatible platforms executing VMware ESXi 6.0, ESXi 6.5, ESXi 6.7, ESXi 7.0, or without ESXi with any of the above listed operating systems:

- Dell PowerEdge R740 with Intel Xeon Gold 6126 Processor running ESXi 6.7 or 7.0 with Photon OS 2.0, Photon OS 3.0, Photon OS 4.0, Ubuntu 16.04, Ubuntu 18.04, Ubuntu 20.04, RHEL

(including 7 and 8), CentOS, Amazon Linux 2, or any other Linux Distributions

- Dell PowerEdge R640 with Intel Xeon Gold 5218 Processor running VMware SD-WAN OS 4.x or 5.x on VMware ESXi 7.0
- Dell PowerEdge R640 with Intel Xeon Gold 5218 Processor running Ubuntu 18.04 or 20.04 on ESXi 7.0
- VMware SD-WAN Edge 610, 620, 680 with Intel Atom C3308, C3558, or C3958 running SD-WAN OS 5.x
- VMware SD-WAN Edge 3810 with Intel Xeon D-2187NT running SD-WAN OS 5.x
- Dell PowerEdge R530, R730, R740, R830, R840, R930, R940, FC640, T320, T430 with Intel Xeon Processor and R740 Gen 14 with Intel Xeon Gold, Silver, or Platinum series Processor
- HPE ProLiant Gen 10: DL 180, DL 360, DL 385, DL560 with Intel Xeon Processor and DL38P Gen8 with AMD Opteron Processor
- Cisco UCS Servers with Intel Xeon Processors, B200, B480, M5 B-Series Blade Serves; C125, C220, C480 M5 C-Series Blade Servers; B22 M-Series Blade Servers and, C24 M3-Series Rackmount Servers
- A general-purpose computer (GPC) that uses the specified single-user operating system/mode or another compatible single-user operating system.

CMVP makes no claims to the correct operation of the module or the minimum strength of generated keys when ported to an OE not listed on the validation certificate. No assurance of the minimum strength of generated keys.

In addition to its full AES software implementations, the VMware's BoringCrypto Module is capable of leveraging the AES-NI instruction set of supported Intel and AMD processors in order to accelerate AES calculations.

All cryptographic keys and CSPs are under the control of the OS, which protects its CSPs against unauthorized disclosure, modification, and substitution. The module only allows access to CSPs through its well-defined API.

The tested operating systems segregate user processes into separate process spaces. Each process space is logically separated from all other processes by the operating system software and hardware. The module functions entirely within the process space of the calling application, and implicitly satisfies the FIPS 140-2 requirement for a single-user mode of operation.

2.7 Cryptographic Key Management

The module supports the CSPs listed below in Table 9 and

Table 10.

Table 9 – List of Cryptographic Keys, Key Components, and CSPs

Key/CSP	Key Type	Generation/Input	Output	Storage	Zeroization	Use
AES key	128, 192, 256-bit keys	Via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Encryption, Decryption
AES-GCM/GMAC Key	128 and 256-bit keys	Via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Encryption, Decryption, Generation, Verification
AES Wrapping Key	128, 192, 256-bit keys	Via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Key Wrapping
Triple-DES key	Keying Option 1	Via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Encryption, Decryption
ECDSA Signing Key	P-224/P-256/P-384/P-521	Internally Generated or via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Signature Generation
EC DH Private Key	P-224/P-256/P-384/P-521	Internally Generated or via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Key Agreement
HMAC key	160, 224, 256, 384, 512 bit-keys	Via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Message Authentication
RSA Key (Key Transport)	2048 - 16384-bit keys	Internally Generated or via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Key Transport

RSA Private Key (Signature Generation)	2048 - 16384-bit keys	Internally Generated or via API in plaintext	Via API in plaintext	In RAM	Reboot OS; Cycle host power	Signature Generation
TLS Pre-Master Secret	384-bit value	Internally Generated	Via API in plaintext	In RAM	Reboot OS; Cycle host power	TLS protocol
TLS Master Secret	384-bit value	Internally Derived using the SP 800-135 TLS KDF	Via API in plaintext	In RAM	Reboot OS; Cycle host power	TLS protocol
CTR_DRBG Entropy Input	384-bit value	Via API in plaintext	Never output from the module	In RAM	Reboot OS; Cycle host power	Entropy input for CTR_DRBG
CTR_DRBG 'V' Value	128-bit value	Generated internally	Never output from the module	In RAM	Reboot OS; Cycle host power	Internal state value used with CTR_DRBG
CTR DRBG 'Key' Value	256-bit value	Generated internally	Never output from the module	In RAM	Reboot OS; Cycle host power	Internal state value used with CTR_DRBG

The RSA generation is consistent with Table B.1 of FIPS 186-4 per IG A.14.

Table 10 – List of Public Keys

Key Name	Description
ECDSA Public Key (Signature Verification)	ECDSA (P-224/P-256/P-384/P-521) Verification of digital signatures
EC DH Public Key (Key Agreement)	EC DH (P-224/P-256/P-384/P-521) Public Key
RSA Public Key (Key Transport)	RSA (2048 to 16384 bits) Transport of encrypted keys
RSA Public Key (Signature Verification)	RSA (1024 to 16384 bits) Verification of digital signatures

For all CSPs and Public Keys:

Storage: RAM, associated to entities by memory location. The module uses CSPs passed in by the calling application on the stack. Being a software module, it does not store any keys or CSP persistently (beyond the lifetime of an API call). The host OS is responsible for the protection of process and memory space from unauthorized access.

Generation: The module implements SP 800-90A compliant DRBG services for creation of symmetric keys, and for generation of ECDSA, EC Diffie-Hellman and RSA keys as shown in Table 2 (per Section 5 of SP 800-133). The calling application is responsible for storage of generated keys returned by the module. A minimum of 128 bits of entropy per each call is requested by the module from its Operational Environment. The module prevents output of intermediate or key information by logically disconnecting the output data path from the processes that perform key generation/zeroization.

Zeroization: The module relies on the calling applications for the exchange of parameters and zeroization of sensitive data for temporarily stored CSPs, is performed by the OS and the calling operation.

Private and secret keys as well as seeds and entropy input are provided to the module by the calling application and are destroyed when released by the appropriate API function calls. Keys residing in internally allocated data structures (during the lifetime of an API call) can only be accessed using the module defined API. The operating system protects memory and process space from unauthorized access. Only the calling application that creates or imports keys can use or export such keys. All API functions are executed by the invoking calling application in a non-overlapping sequence such that no two API functions will execute concurrently. An authorized application as user (Crypto Officer and User) has access to all key data generated during the operation of the module.

2.8 Self-Tests

Cryptographic self-tests are performed by the module upon initialization or invocation of Self-tests, as well as conditionally when the module is operating in the FIPS-Approved mode and a random number is generated, or asymmetric keys are generated. The following sections list the self-tests performed by the module, their expected error status, and any error resolutions.

2.8.1 Power-Up Self-Tests

The module performs power-on self-tests automatically upon initialization of the module and without any operator intervention. If any of the self-tests fail, the module will enter an error state and no services will be provided by the module. All power-on self-tests must pass before an operator can perform services.

The VMware's BoringCrypto Module performs the following Power-up Self-tests:

- Software integrity check (HMAC SHA-512 Integrity Test)
- Known Answer Tests (KATs)
  - AES Encryption and Decryption KATs with 128-bit key
  - AES GCM/GMAC Encryption and Decryption KATs with 128-bit key
  - Triple-DES Encryption and Decryption KATs with 168-bit Key
  - ECDSA Signature Generation and Verification KATs using P-256
  - HMAC SHA-1 and SHA-512 KATs
  - SP 800-90A CTR_DRBG KAT with AES 256-bit key
  - RSA Signature Generation/Verification and Encryption/Decryption KATs using 2048-bit key
  - SHA-1, SHA-256, and SHA-512 KATs

The power-up self-tests may also be performed on-demand by power-cycling the host platform.

2.8.2 Conditional Self-Tests

The module also implements the following conditional self-tests:

- ECDSA Pairwise Consistency Test (PCT) on each key pair generation
- RSA Pairwise Consistency Test (PCT) on each key pair generation
- NDRNG Continuous Random Number Generation Test (CRNGT) per IG 9.8
- DRBG Health Tests as per Section 11.3 of SP 800-90A

Every time a key pair is generated for use with signature generation/verification and key transport, the module performs a PCT on the generated keys.

2.9 Mitigation of Other Attacks

This section is not applicable. The module was not designed to mitigate any attacks beyond the FIPS 140-2 Level 1 requirements for this validation.

3 SECURE OPERATION

The VMware's BoringCrypto Module meets Level 1 requirements for FIPS 140-2. The sections below describe how to install, use, and keep the module in the FIPS-Approved mode of operation.

3.1 Installation Instructions

The module must be installed with the following instructions to comply with FIPS-validated mode of operation

Operational Environmental prerequisites for installation of module

1. Download the required tools

- Clang compiler version 7.0.1 (http://releases.llvm.org/download.html)
- Go programming language version 1.12.7 (https://golang.org/dl/)
- Ninja build system version 1.90 (https://github.com/ninja-build/ninja/releases)

2. Execute the following command to create a CMake toolchain file and to specify the use of Clang in your environment

    - printf "set(CMAKE_C_COMPILER \"clang\")\nset(CMAKE_CXX_COMPILER \"clang++\")\n" > ${HOME}/toolchain

Module installation and verification

1. Download the tarball containing the complete module. Additions, deletions, modifications to any content of the tarball will not be allowed for FIPS-validated mode of operation

    - https://commondatastorage.googleapis.com/chromium-boringssl-fips/boringssl-ae223d6138807a13006342edfeef32e813246b39.tar.xz

      or by doing the following:

    - wget https://commondatastorage.googleapis.com/chromium-boringssl-fips/boringsslae223d6138807a13006342edfeef32e813246b39.tar.xz

2. Verify the authenticity of the tarball by issuing the following command.

    - sha256sum boringssl-ae223d6138807a13006342edfeef32e813246b39.tar.xz

3. Compare the SHA-256 digest value of the tarball with the following value. The SHA-256 digest value must be identical

    - b5fdf23274d4179c2077b5e8fa625d9debd7a390aac1d165b7e47234f648bb8

4. Extract the tarball

    - tar -xvf boringssl-ae223d6138807a13006342edfeef32e813246b39.tar.xz

5. Compile the module

    - cd boringssl
    - mkdir build && cd build && cmake -GNinja -DCMAKE_TOOLCHAIN_FILE=${HOME}/toolchain DFIPS=1 DCMAKE_BUILD_TYPE=Release ..

- ninja
- ninja run_tests

6. Verify the module is compiled and installed correctly by executing the following command. The command should return a value of "1" to identify the module is operating in FIPS-validated mode of operation

   - ./tool/bssl isfips

3.2 Secure Operation

3.2.1 Module Initialization

The operating system bootloader initializes the module in the user space. The module must be loaded first before any cryptographic function is performed. Once the module is loaded, power on self-tests are automatically initiated, the module is designed with a default entry point (DEP) per FIPS 140-2 IG 9.10. Any failure to the power on self-tests will render the module in a failed state and no cryptographic operations will be performed.

3.2.2 Usage of AES-GCM, OFB, CFB, and CFB8

An externally generated IV used for GCM must also originate from a FIPS-approved source. Use of non-FIPS-approved sources for the external IV will render the module to operate in non-approved mode of operation.

Per IG A.5, the module complies with Provision 3 for the requirements of nonce field and the counter part of the IV. The AES-GCM implementation in the module is used in conjunction with an application executing outside of the module's cryptographic boundary. It is the responsibility of the application to negotiate the protocol session's keys and the 32-bit nonce value of the IV and to redistribute the AES-GCM key when the module power is lost and restored.

AES OFB, CFB, and CFB8 modes should not be used in the FIPS-Approved mode.

3.2.3 Usage of Triple-DES

It is the responsibility of the calling application to ensure that no more than 2^{32} or 2^{28} 64-bit data block encryptions using the same 3-key Triple-DES are performed by the module in order to comply with IG A.13.

3.2.4 Asymmetric Algorithm Keys

RSA Key sizes with a modulus of 1024 bits are allowed to be used for signature verification for legacy purposes only. Use of these keys for signature generation are not allowed in Approved mode of operation and will render the module in the non-approved mode of operation. Usage of keys generated by non-Approved methods in Approved algorithms is prohibited.

4 Acronyms

Table 11 provides definitions for the acronyms used in this document.

Table 11 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard – New Instructions
AMD	Advanced Micro Devices
API	Application Programming Interface
BIOS	Basic Input/Output System
CBC	Cipher Block Chaining
CCM	Counter with CBC-MAC
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CPU	Central Processing Unit
CCCS	Canadian Centre for Cyber Security
CSP	Critical Security Parameter
CTR	Counter-mode
DES	Data Encryption Standard
DRBG	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
DVD	Digital Video Disc
EC	Elliptical Curve
ECB	Electronic Code Book
EC DH	Elliptical Curve Diffie-Hellman
ECDSA	Elliptical Curve Digital Signature Algorithm
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FIPS	Federal Information Processing Standard
GCM	Galois/Counter Mode
GMAC	Galois Message Authentication Code
GPC	General Purpose Computer

HMAC	(Keyed) Hash Message Authenticating Code
IG	Implementation Guidance
IT	Information Technology
IV	Initialization Vector
KAT	Known Answer Test
MAC	Message Authentication Code
MD	Message-Digest Algorithm
LCD	Liquid Crystal Display
LED	Light Emitting Diode
N/A	Not Applicable
NIST	National Institute of Standards and Technology
OFB	Output Feedback
OS	Operating System
PCI	Peripheral Component Interconnect
PCT	Pair-wise Consistency Test
PCIe	Peripheral Component Interconnect Express
PRNG	Pseudo Random Number Generator
RAM	Random Access Memory
NDRNG	Non-Deterministic Random Number Generator
RNG	Random Number Generator
RSA	Rivest, Shamir and Adleman
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Interface
SHA	Secure Hash Algorithm
SLES	SUSE Linux Enterprise Server
SP	Special Publication
TCBC	Triple-DES Cipher Block Chaining
TDES	Triple-Data Encryption Standard
TECB	Triple-DES Electronic Code Book
USB	Universal Serial Bus

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com