F5, Inc.

Cryptographic Module for BIG-IP®

Module Version 15.1.2.1 EHF

FIPS 140-2 Non-Proprietary Security Policy

Document Version 1.2

Last update:September 2022

Prepared by:

atsec information security corporation

9130 Jollyville Road, Suite 260

Austin, TX 78759

www.atsec.com

Table of Contents

Copyrights and Trademarks

F5® and BIG-IP® are registered trademarks of F5, Inc.

VMware ESXi™ is a registered trademark of VMware®, Inc.

Intel® Xeon® is a registered trademark of Intel® Corporation.

Dell is a registered trademark of Dell, Inc.

Azure and Hyper-V are registered trademarks of Microsoft

AWS is a trademark of Amazon.com, Inc.

1. Introduction

This document is the non-proprietary FIPS 140-2 Security Policy of Cryptographic Module for BIG-IP with software version 15.1.2.1 EHF. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-2 (Federal Information Processing Standards Publication 140-2 [FIPS140-2]) for a Security Level 1 module.

2. Cryptographic Module Specification

The following section describes the cryptographic module and how it conforms to the FIPS 140-2 specification in each of the required areas.

2.1. Module Overview

The Cryptographic Module for BIG-IP (hereafter referred to as “the module”) is a software library implementing general purpose cryptographic algorithms.

The software module provides cryptographic services to applications through an Application Program Interface (API). The module also interacts with the underlying operating system via system calls.

The software block diagram below shows the module, its interfaces with the operational environment and the delimitation of its logical boundary:

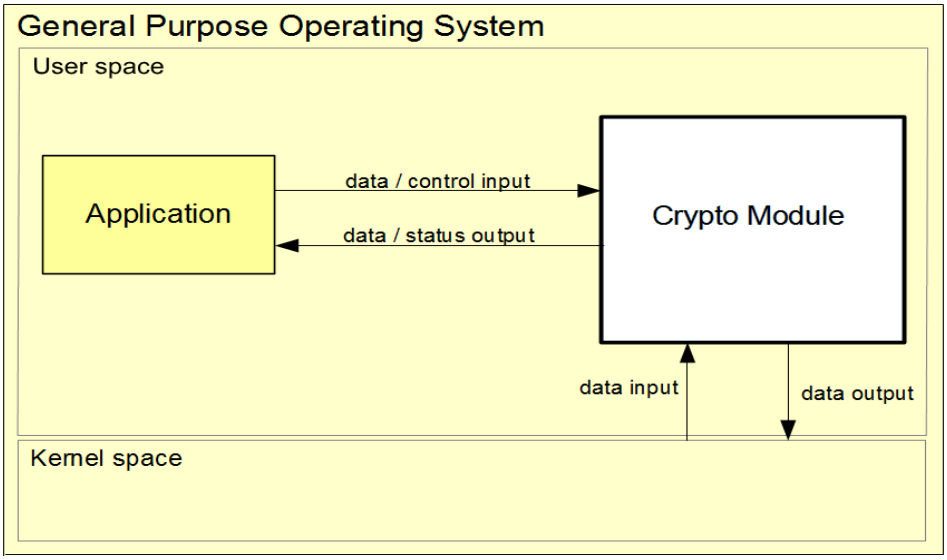

Figure 1 - Software Block Diagram

The module is implemented as a shared library. The cryptographic logical boundary consists of a shared library and the integrity check file used for integrity tests.

Filename	Purpose
libcrypto.so.1.0.2s	The binary for cryptographic implementations.
.libcrypto.so.1.0.2s.hmac	The integrity check file for libcrypto.so binary.

Table 1 - Cryptographic Module Components

The module is aimed to run on a general-purpose computer; the physical boundary is the surface of the case of the target platform, as shown with dotted lines in the diagram below:

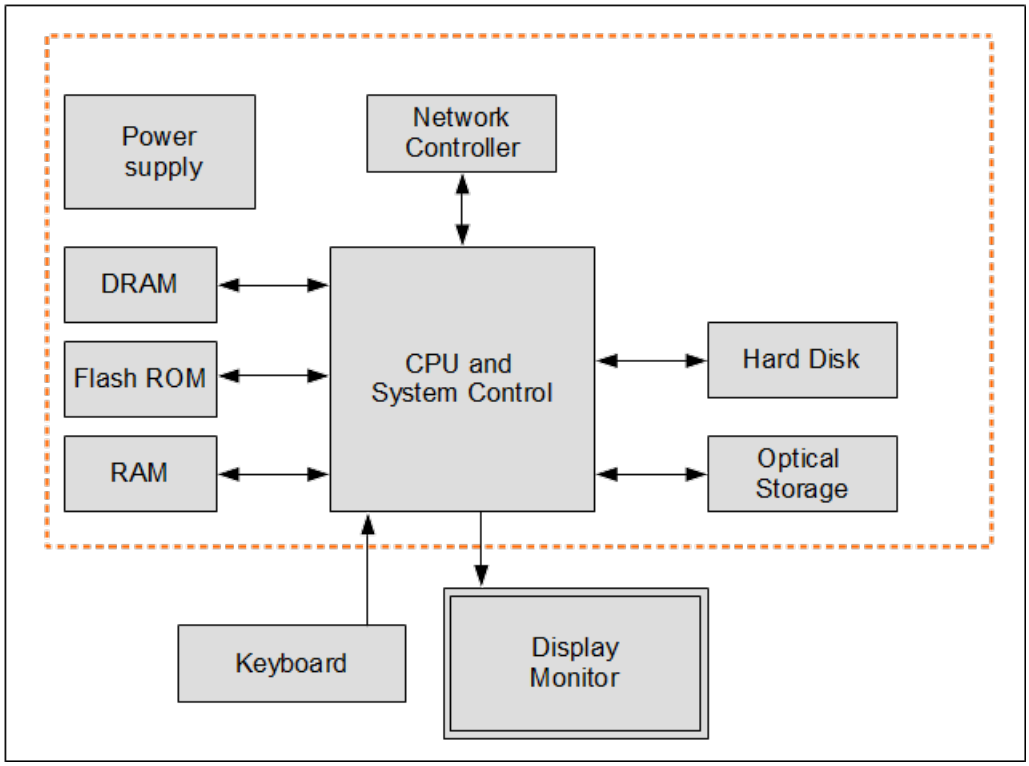

Figure 2 - Cryptographic Module Physical Boundary

2.2. FIPS 140-2 Validation

The module is a software-only, cryptographic module, runnning on multi-chip standalone device and validated at overall security level 1. The table below shows the security level claimed for each of the eleven sections that comprise the FIPS 140-2 standard:

FIPS 140-2 Section		Security 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
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC	1
9	Self-Tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A
Overall Level		1

Table 2 - Security Levels

The module has been tested on the following multichip standalone platform with the corresponding module variant and configuration options:

Module Version	Hardware	Processor	PAA function	Operating System
15.1.2.1 EHF	VMware ESXi™ 6.5 hypervisor running on Dell PowerEdge M630	Intel® Xeon® E5-2690 v4	with and without AES-NI	BIG-IP 15.1.2.1 EHF
15.1.2.1 EHF	Hyper-V 10.0 on Windows Server 2019 running on Dell PowerEdge R630	Intel® Xeon® E5-2660 v3	with and without AES-NI	BIG-IP 15.1.2.1 EHF
15.1.2.1 EHF	KVM Centos 7.0 running on Dell PowerEdge M630	Intel® Xeon® E5-2690 v4	with and without AES-NI	BIG-IP 15.1.2.1 EHF

Table 3 - Tested Platforms

In addition to the configurations tested by the laboratory, vendor-affirmed testing was performed on the following platforms for 15.1.2.1 EHF:

- Azure with Intel(R) Xeon(R) CPU E5-2686 v4 & BIG-IP 15.1.2.1 EHF running on Microsoft Corporation Hyper-V Virtual Machine
- AWS with Intel(R) Xeon(R) CPU E5-2673 v4 & BIG-IP 15.1.2.1 EHF running on Xen 4.2.amazon

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate (IG G.5).

2.3. Modes of operation

The module supports two modes of operation:

- in "FIPS mode" (the FIPS Approved mode of operation) only approved or allowed security functions with sufficient security strength can be used as specified in Table 5.
- in "non-FIPS mode" (the non-Approved mode of operation) only non-approved security functions can be used (Table 6).

The module enters FIPS mode after power-up tests succeed. Once the module is operational, the mode of operation is implicitly assumed depending on the security function invoked and the security strength of the cryptographic keys. Using any service in the Table 5 will implicitly put the module in FIPS mode and utilizing any non-approved service from Table 6 will put the module in non-FIPS mode implicitly. Critical Security Parameters (CSPs) used or stored in FIPS mode are not used in non-FIPS mode, and vice versa.

3. Cryptographic Module Ports and Interfaces

As a software-only module, the module does not have physical ports. For the purpose of the FIPS 140-2 validation, the physical ports are interpreted to be the physical ports of the hardware platform on which it runs.

The logical interfaces are the API through which the applications request services. The following table summarizes the four logical interfaces:

Logical Interface	Description
Data Input	API input parameters for data.
Data Output	API output parameters for data.
Control Input	API function calls for control.
Status Output	API return codes, error messages.

Table 4 - Ports and Interfaces

The Data Input interface consists of the input parameters of the API functions. The Data Output interface consists of the output parameters of the API functions. The Control Input interface consists of the API function calls used to control the behavior of the module. The Status Output interface includes the return values of the API functions and error messages.

4. Roles, Services and Authentication

4.1. Roles

The module supports the following roles:

- **User role**: performs all services (in both FIPS mode and non-FIPS mode of operation), except module initialization.
- **Crypto Officer role**: performs module initialization.

The User and Crypto Officer roles are implicitly assumed by the entity accessing the module services.

4.2. Services

The module provides services to users that assume one of the available roles. All services are described in detail in the user documentation.

The following Table 5 lists the Approved services and the non-Approved but allowed services in FIPS mode of operation, the roles that can request the service, the algorithms involved with their corresponding ACVT certificate numbers (if applicable), the CSPs involved and how they are accessed:

Service	Algorithms and Standards	CAVP Cert.	Role	CSP	Access
AES encryption and decryption	[FIPS197], [FIPS800-38A], [FIPS800-38D], AES-ECB, AES-CBC, AES-GCM, with AES-NI implementation	A1416	User	128/192/256-bit AES key	Read
	AES-ECB, AES-CBC, AES-GCM with assembler implementation	A1417	User		
AES key wrapping	[FIPS800-38F] AES-GCM, in AES-NI implementation	A1416	User	128 and 256-bit AES key	Read
	[FIPS800-38F] AES-GCM, in assembler implementation	A1417			
Random Number Generation	[SP800-90A] CTR_DRBG with AES-256 using AES-NI	A1416	User	Seed, V and Key values	Read, Write
	[SP800-90A] CTR_DRBG with AES-256 assembler	A1417			
	SP800-90B. Entropy source used to seed module’s DRBG.	ENT (NP)		Entropy input string	Read
RSA key pair generation	[FIPS186-4 Appendix B.3.3] RSA key generation	A1417	User	RSA key pair with 2048/3072-bit modulus size	Write

Service	Algorithms and Standards	CAVP Cert.	Role	CSP	Access
RSA signature generation	PKCS#1 v1.5 RSA signature generation with SHA-256 and SHA-384			RSA private key with 2048/3072-bit modulus size	Read
RSA signature verification	PKCS#1 v1.5 RSA signature verification with SHA-1, SHA-256 and SHA-384			RSA public key with 2048/3072-bitmodulus size	Read
ECDSA key pair generation / EC Diffie-Hellman key pair generation	[FIPS186-4 Appendix B.4.2] ECC key pair generation	A1417	User	ECDSA/ECDH key pair for P-256 and P-384 curves	Write
ECDSA key verification	[FIPS186-4] Public Key Verification (PKV)			ECDSA public key for P-256 and P-384 curves	Read
ECDSA signature generation	ECDSA signature generation with SHA-256 and SHA-384			ECDSA private key according to P-256 and P-384 curves	Read
ECDSA signature verification	ECDSA signature verification with SHA-1, SHA-256 and SHA-384			ECDSA public key according to P-256 and P-384.	Read
EC Diffie-Hellman shared secret computation IG D.8 scenario X1 (path 1)	[SP800-56Ar3] KAS ECC SSC except KDF, Schemes: Ephemeral Unified, Section 5.7.1.2 ECC CDH Primitive	A1417	User	EC Diffie-Hellman public and private Key with P-256 and P-384 curves	Read, Write
KTS (IG D.9)	AES-GCM	A1416, A1417	User	128 and 256 bits	Read
Message digest	SHA-1 with SSSE3 implementation	A1416	User	n/a	n/a
	[FIPS180-4] SHA-1, SHA-256, SHA-384 with assembler implementation	A1417			
Message authentication	HMAC-SHA-1 with SSSE3 implementation	A1416	User	At least 112-bit HMAC key	Read
	[FIPS198-1] HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 with assembler implementation	A1417			
Show Status	n/a	n/a	User	n/a	n/a
Self-Tests	n/a	n/a	User	HMAC-SHA-256 key for module integrity test	Read
Zeroization	n/a	n/a	User	All Keys/CSPs	Zeroize

Service	Algorithms and Standards	CAVP Cert.	Role	CSP	Access
Module initialization	n/a	n/a	CO	n/a	n/a

Table 5 - Services in FIPS mode of operation

The following Table 6 lists the services only available in non-FIPS mode of operation.

Service	Role	Usage/Notes
Symmetric Encryption and decryption	User	Using AES with OFB, CFB, CTR, XTS, CCM, KW modes
		Using Blowfish, Camellia, CAST, DES, IDEA, RC2, RC4, SEED, SM2, SM4, Triple-DES algorithms
Message digest	User	SHA-224, SHA-512, SM3, MD4, MD5, MDC2, RIPEMD, Whirlpool
Message authentication	User	HMAC-SHA224, HMAC-SHA512, CMAC with AES, CMAC with Triple-DES
Key generation	User	RSA with key sizes other than 2048 and 3072 bits.
		ECDSA/ EC Diffie-Hellman with public/private key pair for curves other than P-256 and P-384
RSA signature generation and verification	User	Using PKCS #1 v1.5 scheme with key sizes other than 2048 and 3072 bits, for all SHA sizes
	User	Using PSS, X9.31 schemes
	User	Using PKCS #1 v1.5 scheme with modulus size 2048 and 3072 bits with SHA-1 (for Sig Gen only), SHA-224 and SHA-512 (for Sig Gen and Sig Ver)
ECDSA signature generation & verification	User	Using curves other than P-256 and P-384
		Using curves P-256 and P-384 with SHA-1 (for Sig Gen only), SHA-224 and SHA-512 (for Sig Gen and Sig Ver)
		Using SM2 algorithm
RSA encrypt/decrypt	User	With modulus sizes up to 16384 bits

Service	Role	Usage/Notes
DSA domain parameter generation, domain parameter verification, key pair generation, signature generation and verification	User	With all key and SHA sizes
Random Number Generation	User	Using HMAC_DRBG and Hash_DRBG for all SHA sizes
	User	CTR_DRBG with AES-128 or AES-192
	User	ANSI X9.31 RNG
Key Agreement	User	Diffie-Hellman Key agreement without KDF, J-PAKE, SRP
		EC Diffie-Hellman with curves other than P-256 and P-384 without KDF

Table 6 - Services in non-FIPS mode of operation

4.3. Operator Authentication

The module does not implement authentication. The role is implicitly assumed based on the service requested.

5. Physical Security

The module is comprised of software only and therefore this security policy does not make any claims on physical security.

6. Operational Environment

6.1. Applicability

The module operates in a modifiable operational environment per FIPS 140-2 level 1 specifications. The module runs on a BIG-IP 15.1.2.1 EHF operating systems executing on the hardware and hypervisor specified in Table 3 - Tested Platforms. BIG-IP consists of a Linux based operating system customized for performance that runs directly on the hardware or in virtual environment.

6.2. Policy

The operating system is restricted to a single operator; concurrent operators are explicitly excluded.

The application that requests cryptographic services is the single user of the module.

7. Cryptographic Key Management

The following Table 7 summarizes the CSPs that are used by the cryptographic services implemented in the module:

Name	Strength	Generation	Storage	Zeroization
AES Key	Bits:128,192 and 256	N/A. Input as API parameter	RAM	Zeroized by FIPS_cipher_ctx_cleanup()
AES Key wrapping Key	Bits: 128 and 256 bits	N/A. Input as API parameter	RAM	Zeroized by FIPS_cipher_ctx_cleanup()
HMAC Key	Min: 112	N/A. Input as API parameter	RAM	Zeroized by HMAC_CTX_cleanup()
RSA Key Pair	Modulus: 2048 and 3072	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] Key generation method, and the random value used in the key generation is obtained using [SP800-90A] DRBG.	RAM	Zeroized by FIPS_rsa_free()
ECDSA Key Pair	Curves P256 and P384	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] Key generation method, and the random value used in the key generation is obtained using [SP800-90A] DRBG.	RAM	Zeroized by EC_KEY_free()
EC Diffie-Hellman Key pair	Curves P256 and P384	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] Key generation method and the random value used in the key generation is obtained using [SP800-90A] DRBG	RAM	Zeroized by EC_KEY_free()
ECDH shared secret	Curves P256 and P384	Internally generated via SP800-56A ECC CDH shared secret computation	RAM	Zeroized by EC_KEY_free()
entropy input string	Bits: 256	Obtained from ENT (NP).	RAM	Zeroized by FIPS_drbg_free()
DRBG seed, V and Key values	-	Derived from entropy string as defined by [SP800-90A]	RAM	Zeroized by FIPS_drbg_free ()

Table 7 - Life cycle of CSPs

The following sections describe how CSPs (cryptographic keys in particular), are managed during its life cycle.

7.1. Key Generation

For generating RSA and ECDSA and EC Diffie-Hellman keys, the module implements asymmetric key generation services compliant with [FIPS186-4] and using a DRBG compliant with [SP800-90A]. A seed (i.e., the random value) used in asymmetric key generation is obtained from [SP800-90A] DRBG. In accordance with [FIPS 140-2 IG D.12], the cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per SP800-133r2 (vendor affirmed).

The module does not implement symmetric key generation.

7.2. Key Establishment

The module implements EC Diffie-Hellman shared secret computation, compliant with SP800-56Ar3 and scenario X1 (1) primitive only of IG D.8. The module provides EC Diffie-Hellman shared secret computation with curves P-256 or P-384, providing 128- or 192-bit equivalent security strength, respectively.

The module also provides key wrapping in the context of using the TLS protocol to send and receive key material in the payload. The key wrapping methods are provided by the TLS record layer using an approved authenticated encryption mode (i.e. AES GCM). The key wrapping method using AES GCM is an approved key transport method according to IG D.9. AES GCM provides 128 or 256 bits of encryption strength. The TLS protocol has not been reviewed or tested by the CAVP or CMVP.

7.3. Key Entry / Output

The module does not support manual key entry or intermediate key generation key output. In addition, the module does not produce key output outside its physical boundary. The keys can be entered or output from the module in plaintext form via API parameters, to and from the calling application only. This is allowed by [FIPS 140-2_IG] IG 7.7 Table 1, according to the “CM Software to/from App Software via GPC INT Path” entry which refers to keys communicated within the physical boundary of the GPC.

7.4. Key / CSP Storage

Public and private keys are provided to the module by the calling process, and are destroyed when released by the appropriate API function calls.

The module does not perform persistent storage of keys. The only exception is the HMAC-SHA-256 key used for integrity test, which is stored in the module and relies on the operating system for protection.

7.5. Key / CSP Zeroization

The memory occupied by keys is allocated by regular memory allocation operating system calls. The application is responsible for calling the appropriate destruction functions provided in the module's API. The destruction functions overwrite the memory occupied by keys with “zeros” and deallocate the memory with the regular memory deallocation operating system call.

7.6. Random Number Generation

The module employs a Deterministic Random Bit Generator (DRBG) based on [SP800-90A] for the generation of random value used in asymmetric keys, and for providing a RNG service to calling applications.

The Approved DRBG provided by the module is the CTR_DRBG with AES-256. The DRBG is initialized during module initialization.

The module uses a Non-Physical entropy source (ENT (NP)) to seed the DRBG. The ENT (NP) provides at least 256 bits of entropy to the DRBG during initialization (seed) and reseeding (reseed). The entropy source is outside of the module’s logical boundary but within its physical boundary.

8. Self-Tests

8.1. Power-Up Tests

The module performs power-up tests automatically when the module is loaded into memory; power-up tests ensure that the module is not corrupted and that the cryptographic algorithms work as expected.

While the module is executing the power-up tests, services are not available, and input and output are inhibited. The module does not return control to the calling application until the power-up tests are completed. On successful completion of the power-up tests, the module enters operational mode and cryptographic services are available. If the module fails any of the power-up tests, it will return an error code and enter into the Error state to prohibit any further cryptographic operations. The module must be re-loaded in order to clear the error condition.

8.1.1. Integrity Tests

The integrity of the module is verified by comparing an HMAC-SHA-256 value calculated at run time with the HMAC value stored in the module that was computed at build time.

8.1.2. Cryptographic algorithm tests

The module performs self-tests on all FIPS-Approved cryptographic algorithms supported in the approved mode of operation, using the Known Answer Test (KAT) and Pair-wise Consistency Test (PCT) as shown in the following Table 8:

Algorithm	Test
CTR_DRBG	KAT with AES 256 bits with and without derivation function
AES	KAT of AES encryption with AES-GCM mode and 128 bit key KAT of AES decryption with ECB mode and 128 bit key
RSA	KAT of RSA PKCS#1 v1.5 signature generation with 2048 bit key and SHA-256 KAT of RSA PKCS#1 v1.5 signature verification with 2048 bit key and SHA-256
ECDSA	PCT of ECDSA signature generation and verification with P-256 curve
KAS SSC (EC Diffie-Hellman)	KAT of primitive “Z” computation with P-256 curve
HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384	KAT of HMAC-SHA-1 KAT of HMAC-SHA-256 KAT of HMAC-SHA-384
SHA-1, SHA-256, SHA-384	The SHA KATs are covered by the HMAC-SHA KATs (for all the SHA sizes) complying with IG 9.2

Table 8- Self-Tests

8.2. On-Demand self-tests

The module provides the Self-Test service to perform self-tests on demand. On demand self-tests can be invoked by powering-off and reloading the module. This service performs the same cryptographic algorithm tests executed during power-up. During the execution of the on-demand self-tests, crypto services are not available and no data output or input is possible.

8.3. Conditional Tests

The module performs conditional tests on the cryptographic algorithms shown in the following Table 9. If the module fails any of these tests, it will enter into the Error state to prohibit any further cryptographic operations. The module must be re-loaded in order to clear the error condition.

Algorithm	Test
CTR_DRBG	Continuous random number generator test for DRBG
RSA key generation	PCT using SHA-256
ECDSA and EC Diffie-Hellman key generation	PCT using SHA-256

Table 9 - Conditional Tests

9. Guidance

9.1. Delivery

The module is distributed as a part of BIG-IP product in the form of the 15.1.2.1 EHF ISO. The module i.e. libcrypto.so binary gets installed together with the product. The FIPS validated module activation requires installation of the 'FIPS 140-2 Compliant Mode' add-on license.

9.2. Crypto Officer Guidance

On the BIG-IP product the Crypto Officer should run the command '**tmsh show sys version**[1]' to ensure that Sys::version shows the information below.

module version 15.1.2.1 EHF
Sys::Version
Main Package
Product BIG-IP
Version 15.1.2.1
Build: 0.375.10
Edition Engineering Hotfix

The Crypto Officer should also verify the FIPS validated module license activation by running the command: '***tmsh show sys license***' which should list 'FIPS 140-2 Level 1, BIG-IP VE-1G to 10G,' under the 'Active Modules' list. After the FIPS validated module license is installed, the command prompt will change to 'REBOOT REQUIRED'. The Crypto Officer must reboot the BIG-IP for all FIPS-compliant changes to take effect.

9.3. User Guidance

The module supports two modes of operation. Table 5 lists the FIPS approved services. Using the services in Table 6 will put the module in non-FIPS mode implicitly.

The user shall consider the following requirements and restrictions when using the module.

For TLS 1.2, the module offers the AES-GCM implementation and uses the context of Scenario 1 of IG A.5. The module is compliant with SP800-52Rev2 section 3.3.1 and the mechanism for IV generation is compliant with RFC5288.

The module does not implement the TLS protocol. The module's implementation of AES-GCM is used together with an application that runs outside the module's cryptographic boundary The design of the TLS protocol implicitly ensures that the counter (the nonce_explicit part of the IV) does not exhaust the maximum number of possible values for a given session key.

In the event the module's power is lost and restored, the consuming application must ensure that a new key for use with the AES-GCM key encryption or decryption under this scenario shall be established.

[1] The Sys::License information shown with command line *'tmsh show sys license'* shows a License Version of 15.1.2 that is the first released number and not the current Sys:: Version number of 15.1.2.1

10. Mitigation of Other Attacks

The module does not implement security mechanisms to mitigate other attacks.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
CBC	Cipher Block Chaining
CFB	Cipher Feedback
CSP	Critical Security Parameter
CTR	Counter Mode
CVL	Component Validation List
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ENT (NP)	Non-Physical entropy source
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
J-PAKE	Password Authentication Key exchange by Juggling
KAS	Key Agreement Scheme
KAT	Known Answer Test
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Accelerators
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SSSE3	Supplemental Streaming SIMD Extensions 3
XTS	XEX-based Tweaked-codebook mode with cipher text stealing

Appendix B. References Selection

FIPS140-2 **FIPS PUB 140-2 - Security Requirements For Cryptographic Modules**
May 2001
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-2.pdf

FIPS140-2_IG **Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program**
https://csrc.nist.gov/csrc/media/projects/cryptographic-module-validation-program/documents/fips140-2/fips1402ig.pdf

FIPS180-4 **Secure Hash Standard (SHS)**
Aug 2015
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

FIPS186-4 **Digital Signature Standard (DSS)**
July 2013
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

FIPS197 **Advanced Encryption Standard**
November 2001
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

FIPS198-1 **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.198-1.pdf

PKCS#1 **Public Key Cryptography Standards (PKCS) #1: RSA Cryptography**
https://tools.ietf.org/html/rfc8017

SP800-38A **NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques**
December 2001
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf

SP800-38D **NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC**
November 2007
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf

SP800-56Ar3 **NIST Special Publication 800-56A - Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography**
Apr 2018, rev3
https://doi.org/10.6028/NIST.SP.800-56Ar3

SP800-90A **NIST Special Publication 800-90A - Recommendation for Random Number Generation Using Deterministic Random Bit Generators**
Jun 2015
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf