Amazon Web Services Inc.

AWS-LC Cryptographic Module (static)

FIPS 140-3 Non-Proprietary Security Policy

Prepared by:

atsec information security corporation

4516 Seton Center Pkwy, Suite 250

Austin, TX 78759

www.atsec.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for version AWS-LC FIPS 2.0.0 of the AWS-LC Cryptographic Module (static). It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for an overall Security Level 1 module.

1.2 Security Levels

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	N/A
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	1
	Overall Level	1

Table 1: Security Levels

1.3 Additional Information

This Security Policy describes the features and design of the module named AWS-LC Cryptographic Module (static) using the terminology contained in the FIPS 140-3 specification. The FIPS 140-3 Security Requirements for Cryptographic Module specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information. The NIST/CCCS Cryptographic Module Validation Program (CMVP) validates cryptographic module to FIPS 140-3. Validated products are accepted by the Federal agencies of both the USA and Canada for the protection of sensitive or designated information.

This Non-Proprietary Security Policy may be reproduced and distributed, but only whole and intact and including this notice. Other documentation is proprietary to their authors.

In preparing the Security Policy document, the laboratory formatted the vendor-supplied documentation for consolidation without altering the technical statements therein contained. The further refining of the Security Policy document was conducted iteratively throughout the conformance testing, wherein the Security Policy was submitted to the vendor, who would then edit, modify, and add technical contents. The vendor would also supply additional documentation, which the laboratory formatted into the existing Security Policy, and resubmitted to the vendor for their final editing.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The AWS-LC Cryptographic Module (static) (hereafter referred to as “the module”) provides cryptographic services to applications running in the user space of the underlying operating system through a C language Application Program Interface (API).

Module Type: Software

Module Embodiment: MultiChipStand

Module Characteristics:

Cryptographic Boundary:

The block diagram in Figure 1 shows the cryptographic boundary of the module, its interfaces with the operational environment and the flow of information between the module and operator (depicted through the arrows).

The cryptographic boundary is defined as the AWS-LC Cryptographic Module (static) which is a cryptographic library consisting of the bcm.o file (version AWS-LC FIPS 2.0.0). This file is statically linked to the userspace application during the compilation process.

Tested Operational Environment’s Physical Perimeter (TOEPP):

The TOEPP is the general-purpose computer on which the module is installed.

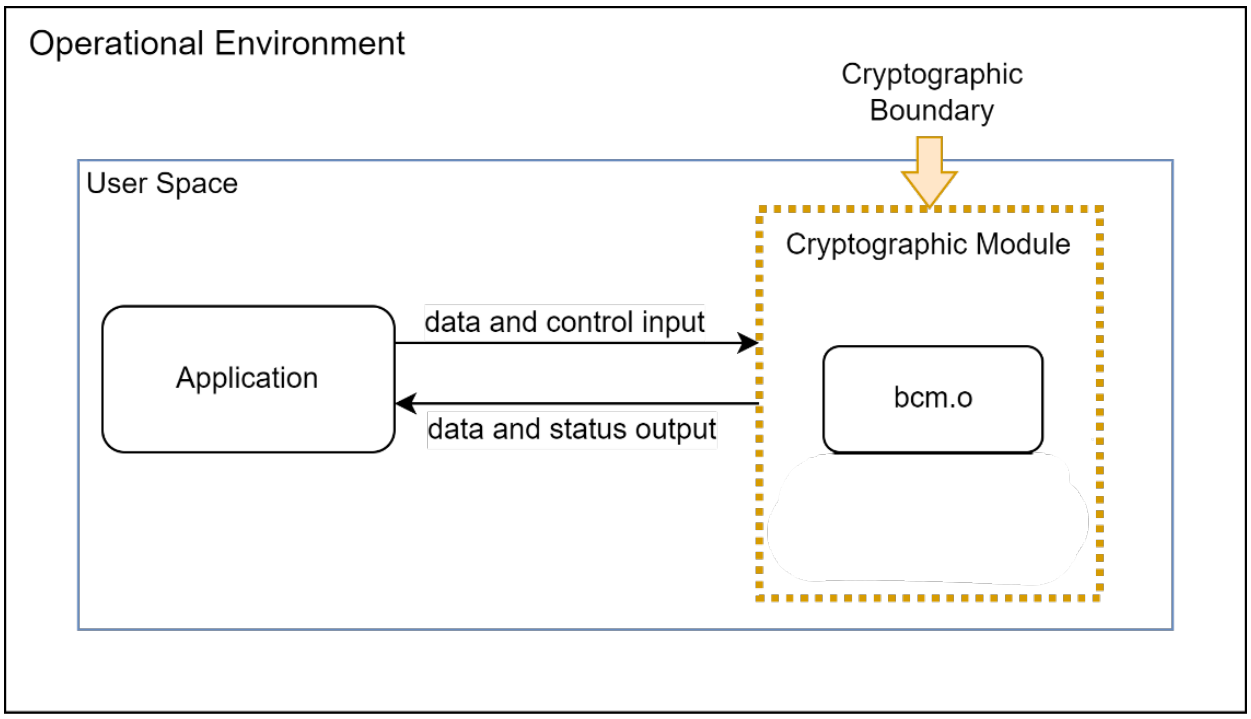

Figure 1: Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):

Package or File Name	Software/ Firmware Version	Features	Integrity Test
bcm.o on Amazon Linux 2 with Intel ®Xeon ® Platinum 8275CL	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256
bcm.o on Amazon Linux 2023 with Intel ®Xeon ® Platinum 8275CL	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256
bcm.o on Ubuntu 22.04 with Intel ®Xeon ® Platinum 8275CL	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256
bcm.o on Amazon Linux 2 with Gravition3	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256
bcm.o on Amazon Linux 2023 with Gravition3	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256
bcm.o on Ubuntu 22.04 with Gravition3	AWS-LC FIPS 2.0.0	N/A	HMAC-SHA2-256

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

Tested Module Identification – Hybrid Disjoint Hardware:

N/A for this module.

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Amazon Linux 2	Amazon EC2 c5.metal with 192 GiB system memory and Elastic Block Store (EBS) 200 GiB	Intel® Xeon® Platinum 8275CL	Yes	N/A	AWS-LC FIPS 2.0.0
Amazon Linux 2023	Amazon EC2 c5.metal with 192 GiB system memory and Elastic Block Store (EBS) 200 GiB	Intel® Xeon® Platinum 8275CL	Yes	N/A	AWS-LC FIPS 2.0.0
Ubuntu 22.04	Amazon EC2 c5.metal with 192 GiB system memory and Elastic Block Store (EBS) 200 GiB	Intel® Xeon® Platinum 8275CL	Yes	N/A	AWS-LC FIPS 2.0.0
Amazon Linux 2	Amazon EC2 c7g.metal with 128 GiB system memory and Elastic Block Store (EBS) 200 GiB	Graviton3	Yes	N/A	AWS-LC FIPS 2.0.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Amazon Linux 2023	Amazon EC2 c7g.metal with 128 GiB system memory and Elastic Block Store (EBS) 200 GiB	Graviton3	Yes	N/A	AWS-LC FIPS 2.0.0
Ubuntu 22.04	Amazon EC2 c7g.metal with 128 GiB system memory and Elastic Block Store (EBS) 200 GiB	Graviton3	Yes	N/A	AWS-LC FIPS 2.0.0

Table 3: Tested Operational Environments - Software, Firmware, Hybrid

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

N/A for this module.

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.

2.3 Excluded Components

The module does not claim any excluded components.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	Automatically entered whenever an approved service is requested.	Approved	Equivalent to the indicator of the requested service.
Non-approved Mode	Automatically entered whenever a non-approved service is requested.	Non-Approved	Equivalent to the indicator of the requested service.

Table 4: Modes List and Description

Mode Change Instructions and Status:

When the module starts up successfully, after passing the pre-operational self-test and the cryptographic algorithms self-tests (CASTs), the module is operating in the approved mode of operation by default and can only be transitioned into the non-approved mode by calling one of the non-approved services listed in the Non-Approved Services table. The module will transition back to approved mode when approved service is called. Section 4 provides details on the service indicator implemented by the module. The service indicator identifies when an approved service is called.

Degraded Mode Description:

The module does not implement a degraded mode of operation.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyGen (FIPS186-5)	A4509	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4509	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4509	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4509	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4509	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4509	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4509	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4509	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A4509	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4509	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4509	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4509	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
RSA SigGen (FIPS186-5)	A4509	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4509	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4509	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4509	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
AES-CBC	A4510	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A4510	Key Length - 128	SP 800-38C
AES-CMAC	A4510	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4510	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4510	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4510	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4510	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-XTS Testing Revision 2.0	A4510	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4510	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4511	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4511	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4511	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D

Algorithm	CAVP Cert	Properties	Reference
AES-ECB	A4512	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4512	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4512	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-CBC	A4513	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A4513	Key Length - 128	SP 800-38C
AES-CMAC	A4513	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4513	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4513	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4513	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4513	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-XTS Testing Revision 2.0	A4513	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4513	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4514	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4514	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4514	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-CBC	A4515	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A4515	Key Length - 128	SP 800-38C
AES-CMAC	A4515	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4515	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4515	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4515	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4515	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F

Algorithm	CAVP Cert	Properties	Reference
AES-XTS Testing Revision 2.0	A4515	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4515	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4516	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4516	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4516	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
ECDSA KeyGen (FIPS186-5)	A4517	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4517	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4517	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4517	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4517	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4517	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4517	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4517	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2

Algorithm	CAVP Cert	Properties	Reference
KDF SSH (CVL)	A4517	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4517	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4517	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4517	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4517	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4517	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4517	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4517	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
ECDSA KeyGen (FIPS186-5)	A4518	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4518	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4518	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4518	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4518	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-224	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4518	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4518	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4518	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A4518	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4518	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4518	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4518	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4518	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4518	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4518	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-512	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4518	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
AES-CBC	A4519	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A4519	Key Length - 128	SP 800-38C
AES-CMAC	A4519	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4519	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4519	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4519	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4519	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-XTS Testing Revision 2.0	A4519	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4519	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4520	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4520	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4520	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4521	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4521	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4521	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4522	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4522	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4522	Direction - Decrypt, Encrypt IV Generation - External, Internal	SP 800-38D

Algorithm	CAVP Cert	Properties	Reference
		IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	
AES-CBC	A4523	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A4523	Key Length - 128	SP 800-38C
AES-CMAC	A4523	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4523	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4523	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4523	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4523	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-XTS Testing Revision 2.0	A4523	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4523	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4524	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4524	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4524	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4525	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4525	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4525	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4526	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4526	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4526	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-CBC	A4527	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Algorithm	CAVP Cert	Properties	Reference
AES-CCM	A4527	Key Length - 128	SP 800-38C
AES-CMAC	A4527	Direction - Generation, Verification Key Length - 128, 256	SP 800-38B
AES-CTR	A4527	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A4527	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A4527	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-KWP	A4527	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-XTS Testing Revision 2.0	A4527	Direction - Decrypt, Encrypt Key Length - 256	SP 800-38E
Counter DRBG	A4527	Prediction Resistance - No Mode - AES-256 Derivation Function Enabled - No	SP 800-90A Rev. 1
AES-ECB	A4528	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4528	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4528	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4529	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4529	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4529	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-ECB	A4530	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A4530	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
AES-GMAC	A4530	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 256	SP 800-38D
ECDSA KeyGen (FIPS186-5)	A4531	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4531	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4531	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
		384, SHA2-512 Component - No	
ECDSA SigVer (FIPS186-4)	A4531	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4531	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4531	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4531	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4531	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A4531	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4531	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4531	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4531	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4531	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4531	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4531	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-224	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4531	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
ECDSA KeyGen (FIPS186-5)	A4532	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4532	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4532	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4532	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4532	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4532	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4532	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4532	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2

Algorithm	CAVP Cert	Properties	Reference
KDF SSH (CVL)	A4532	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4532	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4532	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4532	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4532	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4532	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4532	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4532	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
ECDSA KeyGen (FIPS186-5)	A4533	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4533	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4533	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4533	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4533	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-224	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4533	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4533	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4533	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A4533	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4533	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4533	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4533	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A4533	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4533	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4533	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-512	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4533	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
ECDSA KeyGen (FIPS186-5)	A4534	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A4534	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4534	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A4534	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A4534	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A4534	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4534	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A4534	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-2048 Increment 8 HMAC Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A4534	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KDF TLS (CVL)	A4534	TLS Version - v1.0/1.1, v1.2 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A4534	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 14-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A4534	Key Generation Mode - probable Modulo - 2048, 3072, 4096	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
		Primality Tests - 2powSecStr Private Key Format - standard	
RSA SigGen (FIPS186-5)	A4534	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-4)	A4534	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-5)	A4534	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
SHA-1	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A4534	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Table 5: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength. EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength	AWS-LC Cryptographic Module (static build) (SHA_ASM)	SP 800-133Rev2 section 5.1 and 5.2
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength. EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.	AWS-LC Cryptographic Module (static build) (SHA_CE)	SP 800-133Rev2 section 5.1 and 5.2
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.	AWS-LC Cryptographic Module (static build) (NEON)	SP 800-133Rev2 section 5.1 and 5.2
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength.	AWS-LC Cryptographic Module (static build) (SHA_SHANI)	SP 800-133Rev2 section 5.1 and 5.2

Name	Properties	Implementation	Reference
	EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.		
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength. EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.	AWS-LC Cryptographic Module (static build) (SHA_AVX2)	SP 800-133Rev2 section 5.1 and 5.2
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength. EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.	AWS-LC Cryptographic Module (static build) (SHA_AVX)	SP 800-133Rev2 section 5.1 and 5.2
Cryptographic Key Generation (CKG)	RSA (FIPS 186-5):2048, 3072, 4096 bits with 112, 128, 149 bits of key strength. EC (FIPS 186-5):P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.	AWS-LC Cryptographic Module (static build) (SHA_SSSE3)	SP 800-133Rev2 section 5.1 and 5.2

Table 6: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

The module does not implement non-approved algorithms that are allowed in the approved mode of operation.

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
MD5	Allowed per IG 2.4.A	Message Digest used in TLS 1.0/1.1 KDF only

Table 7: Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
AES with OFB or CFB1, CFB8 modes	Encryption, Decryption
AES GCM, GCM, GMAC, XTS with keys not listed in Table 5	Encryption, Decryption
AES using aes_*_generic function	Encryption, Decryption
AES GMAC using aes_*_generic	Message Authentication Generation
Curve secp256k1	Signature Generation, Signature Verification, Shared Secret Computation
Diffie Hellman	Shared Secret Computation
HMAC-MD4, HMAC-MD5, HMAC-SHA1, HMAC-SHA-3, HMAC-RIPEMD-160	Message Authentication Generation
MD4	Message Digest

Name	Use and Function
MD5 (outside of TLS)	Message Digest
RSA using RSA_generate_key_ex	Key Generation
ECDSA using EC_KEY_generate_key	Key Generation
RSA using keys less than 2048 bits	Signature Generation
RSA using keys less than 1024 bits	Signature Verification
RSA without hashing	Sign/Verify primitive operations
RSA encryption primitive with PKCS#1 v1.5 and OAEP padding	Encryption
SHA-1, SHA-3	Signature Generation
SHAKE, RIPEMD-160, SHA-3	Message Digest
TLS KDF using any SHA algorithms other than SHA2-256, SHA2-384, SHA2-512; or TLS KDF using non-extended master secret	Key Derivation
RSA	Key Encapsulation/Un-encapsulation

Table 8: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Shared Secret Computation with EC Diffie-Hellman	KAS-SSC	Shared secret computation per SP 800-56ARev3	Curves:P-224, P-256, P-384, P-521 elliptic curves with 112-256 bits of key strength Compliance:Compliant with IG D.F scenario 2(1)	KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3 KAS-ECC-SSC Sp800-56Ar3
Key Wrapping/Unwrapping with AES KW, AES-KWP	KTS-Wrap	Key wrapping, key unwrapping using AES KW/KWP	Keys:128, 192, 256 bits with 128-256 bits of key strength Compliance:Compliant with IG D.G	AES-KW AES-KWP AES-KW AES-KWP AES-KW AES-KWP AES-KW AES-KWP AES-KW AES-KWP
Key Wrapping/Unwrapping with AES GCM	KTS-Wrap	Key wrapping, key unwrapping using AES GCM	Keys:128 and 256 bits with 128 and 256 bits of key strength Compliance: Compliant with IG D.G	AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM

Name	Type	Description	Properties	Algorithms
				AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM
Key Wrapping/Unwrapping with AES CCM	KTS-Wrap	Key wrapping, key unwrapping using AES CCM	Keys:128 bits with 128 bits of key strength Compliance:Compliant with IG D.G	AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Encryption/Decryption with AES	BC-UnAuth	Encryption, decryption using AES	Keys:128, 192, 256 bits keys with 128-256 of key strength	AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-ECB AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-ECB AES-ECB AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-ECB AES-ECB AES-CBC AES-CTR AES-ECB AES-XTS Testing Revision 2.0 AES-ECB AES-ECB AES-ECB
Signature Generation with RSA	DigSig-SigGen	Digital signature generation using RSA	Keys:2048, 3072, 4096 bits with 112-150 bits of strength	RSA SigGen (FIPS186-5) RSA SigGen

Name	Type	Description	Properties	Algorithms
				(FIPS186-5) RSA SigGen (FIPS186-5) RSA SigGen (FIPS186-5) RSA SigGen (FIPS186-5) RSA SigGen (FIPS186-5) RSA SigGen (FIPS186-5)
Signature Generation with ECDSA	DigSig-SigGen	Digital signature generation using ECDSA	Curves:P-224, P-256, P-384, P-521 with 112-256 bits of key strength	ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5) ECDSA SigGen (FIPS186-5)
Key Generation with RSA	AsymKeyPair-KeyGen	Key generation using RSA	Keys:2048, 3072, 4096 bits key with 112-150 bits of strength	RSA KeyGen (FIPS186-5) RSA KeyGen (FIPS186-5) RSA KeyGen (FIPS186-5) RSA KeyGen (FIPS186-5) RSA KeyGen (FIPS186-5) RSA KeyGen (FIPS186-5)
Key Generation with ECDSA	AsymKeyPair-KeyGen	Key generation using ECDSA	Curves:P-224, P-256, P-384, P-521 with 112-256 bits of strength	ECDSA KeyGen (FIPS186-5) ECDSA KeyGen (FIPS186-5) ECDSA KeyGen (FIPS186-5) ECDSA KeyGen (FIPS186-5) ECDSA KeyGen (FIPS186-5) ECDSA KeyGen (FIPS186-5)
Signature Verification with ECDSA	DigSig-SigVer	Signature verification using ECDSA	Curves:P-224, P-256, P-384, P-521 with 112-256 bits of strength	ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4)

Name	Type	Description	Properties	Algorithms
				ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-5) ECDSA SigVer (FIPS186-5) ECDSA SigVer (FIPS186-5) ECDSA SigVer (FIPS186-5) ECDSA SigVer (FIPS186-5) ECDSA SigVer (FIPS186-5)
Signature Verification with RSA	DigSig-SigVer	Signature verification using RSA	Keys:1024, 2048, 3072, 4096 bits with 80-150 bits of strength	RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-5)
Key Verification with ECDSA	AsymKeyPair-KeyVer	Key verification using ECDSA	Curves:P-224, P-256, P-384, P-521 with 112-256 bits of strength	ECDSA KeyVer (FIPS186-5) ECDSA KeyVer (FIPS186-5) ECDSA KeyVer (FIPS186-5)

Name	Type	Description	Properties	Algorithms
				ECDSA KeyVer (FIPS186-5) ECDSA KeyVer (FIPS186-5) ECDSA KeyVer (FIPS186-5) ECDSA KeyVer (FIPS186-5)
Key Derivation with TLS KDF	KAS-135KDF	Key derivation using TLS KDF	Derived keys:112 to 256 bits	KDF TLS KDF TLS KDF TLS KDF TLS KDF TLS KDF TLS KDF TLS
Key Derivation with SSH KDF	KAS-135KDF	Key derivation using SSH KDF	SSH Derived keys:112 to 256 bits	KDF SSH KDF SSH KDF SSH KDF SSH KDF SSH KDF SSH KDF SSH
Key Derivation with KDA HKDF	KAS-56CKDF	Key derivation using KDA HKDF	Derived keys:112 to 256 bits	KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1 KDA HKDF Sp800-56Cr1
Key Derivation with PBKDF	PBKDF	Key derivation using PBKDF	Derived keys:112 to 256 bits	PBKDF PBKDF PBKDF PBKDF PBKDF PBKDF PBKDF
Message Digest with SHA	SHA	Message digest using SHA		SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224

Name	Type	Description	Properties	Algorithms
				SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256 SHA-1 SHA2-224 SHA2-256 SHA2-384 SHA2-512 SHA2-512/256
Random Number Generation with DRBG	DRBG	Random number generation using DRBG	Compliance:Compliant with SP800-90ARev1	Counter DRBG Counter DRBG Counter DRBG Counter DRBG Counter DRBG Counter DRBG
Message Authentication Generation with HMAC	MAC	Message authentication generation using HMAC	SHA algorithm:SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/256	HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256

Name	Type	Description	Properties	Algorithms
				HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512/256
Message Authentication Generation with AES	MAC	Message authentication generation using AES CMAC/GMAC	Keys:128 or 256 bits with 128 or 256 bits of strength	AES-CMAC AES-GMAC AES-GMAC AES-CMAC AES-GMAC AES-CMAC AES-GMAC AES-CMAC AES-GMAC AES-GMAC AES-GMAC AES-CMAC AES-GMAC AES-GMAC AES-GMAC AES-CMAC AES-GMAC AES-GMAC AES-GMAC
Authenticated Encryption/Decryption with AES CCM	BC-Auth	Authenticated encryption and decryption using AES CCM	Keys:128 bits with 128 bits of strength	AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Authenticated Encryption/Decryption with AES GCM	BC-Auth	Authenticated encryption and decryption using AES GCM	Keys:128 or 256 bits with 128 or 256 bits of strength Authenticated Encryption:Internal IV Mode 8.2.2 Authenticated Decryption:External IV	AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM AES-GCM

Name	Type	Description	Properties	Algorithms
				AES-GCM AES-GCM AES-GCM AES-GCM

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

GCM IV

The module offers three AES GCM implementations. The GCM IV generation for these implementations complies respectively with IG C.H under Scenario 1, Scenario 2, and Scenario 5. The GCM shall only be used in the context of the AES-GCM encryption executing under each scenario, and using the referenced APIs explained next.

Scenario 1, TLS 1.2

For TLS 1.2, the module offers the GCM implementation via the functions EVP_aead_aes_128_gcm_tls12() and EVP_aead_aes_256_gcm_tls12(), and uses the context of Scenario 1 of IG C.H. The module is compliant with SP800-52rev2 and the mechanism for IV generation is compliant with RFC5288. The module supports acceptable AES-GCM ciphersuites from Section 3.3.1 of SP800-52rev2.

The module explicitly ensures that the counter (the nonce_explicit part of the IV) does not exhaust the maximum number of possible values of 2^{64-1} for a given session key. If this exhaustion condition is observed, the module returns an error indication to the calling application, which will then need to either abort the connection, or trigger a handshake to establish a new encryption 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.

Scenario 2, Random IV

In this implementation, the module offers the interfaces EVP_aead_aes_128_gcm_randnonce() and EVP_aead_aes_256_gcm_randnonce() for compliance with Scenario 2 of IG C.H and SP800-38D Section 8.2.2. The AES-GCM IV is generated randomly internal to the module using module's approved DRBG. The DRBG seeds itself from the entropy source. The GCM IV is 96 bits in length. Per Section 9, this 96-bit IV contains 96 bits of entropy.

Scenario 5, TLS 1.3

August 2018, using the ciphersuites that explicitly select AES-GCM as the encryption/decryption cipher (Appendix B.4 of RFC8446). The module supports acceptable AES-GCM ciphersuites from Section 3.3.1 of SP800-52rev2.

The module implements, within its boundary, an IV generation unit for TLS 1.3 that keeps control of the 64-bit counter value within the AES-GCM IV. If the exhaustion condition is observed, the module will return an error indication to the calling application, who will then need to either trigger a re-key of the session (i.e., a new key for AES-GCM), or terminate the connection.

In the event the module's power is lost and restored, the consuming application must ensure that new AES-GCM keys encryption or decryption under this scenario are established. TLS 1.3 provides session resumption, but the resumption procedure derives new AES-GCM encryption keys.

AES XTS

The length of a single data unit encrypted or decrypted with AES XTS shall not exceed 2^{20} AES blocks, that is 16MB, of data per XTS instance. An XTS instance is defined in Section 4 of SP 800-38E. The XTS mode shall only be used for the cryptographic protection of data on storage devices. It shall not be used for other purposes, such as the encryption of data in transit.

Key Derivation using SP 800-132 PBKDF2

The module provides password-based key derivation (PBKDF2), compliant with SP 800-132. The module supports option 1a from Section 5.4 of SP 800-132, in which the Master Key (MK) or a segment of it is used directly as the Data Protection Key (DPK). In accordance with SP 800-132 and FIPS 140-3 IG D.N, the following requirements shall be met:

- Derived keys shall only be used in storage applications. The MK shall not be used for other purposes. The module accepts a minimum length of 112 bits for the MK or DPK.
- Passwords or passphrases, used as an input for the PBKDF2, shall not be used as cryptographic Keys.
- The minimum length of the password or passphrase accepted by the module is 14 characters. This results in the estimated probability of guessing the password to be at most 10^{-14}. Combined with the minimum iteration count as described below, this provides an acceptable trade-off between user experience and security against brute-force attacks.
- A portion of the salt, with a length of at least 128 bits (this is verified by the module to determine the service is approved), shall be generated randomly using the SP 800-90Ar1 DRBG provided by the module.
- The iteration count shall be selected as large as possible, if the time required to generate the key using the entered password is acceptable for the users. The module restricts the minimum iteration count to be 1000.

Compliance to SP 800-56ARev3 assurances

The module offers ECDH shared secret computation services compliant to the SP 800-56ARev3 and meeting IG D.F scenario 2 path (1). To meet the required assurances listed in section 5.6 of SP 800-56ARev3, the module shall be used together with an application that implements the "TLS protocol" and the following steps shall be performed.

- The entity using the module, must use the module's "Key Pair Generation" service for generating ECDH ephemeral keys. This meets the assurances required by key pair owner defined in the section 5.6.2.1 of SP 800-56ARev3.
- As part of the module's shared secret computation (SSC) service, the module internally performs the public key validation on the peer's public key passed in as input to the SSC function. This meets the public key validity assurance required by the sections 5.6.2.2.1/5.6.2.2.2 of SP 800-56Arev3.
- The module does not support static keys therefore the "assurance of peer's possession of private key" is not applicable.

2.8 RBG and Entropy

N/A for this module.

N/A for this module.

The module provides an SP800-90Arev1-compliant Deterministic Random Bit Generator (DRBG) using CTR_DRBG mechanism with AES-256 for generation of key components of asymmetric keys, and random number generation. The DRBG is seeded with 256-bit of entropy input provided from an external entity to the module. This corresponds to scenario 2 (b) of IG 9.3.A i.e., the DRBG that receives a LOAD command with entropy obtained from inside the physical perimeter of the operational environment but outside of module's cryptographic boundary. The calling application shall use an entropy source that meets the security strength required for the CTR_DRBG as shown in NIST SP 800-90Arev1, Table 3 and should return an error if minimum strength cannot be met.

Per the IG 9.3.A requirement, the module includes the caveat "No assurance of the minimum strength of generated keys".

2.9 Key Generation

The module implements Cryptographic Key Generation (CKG, vendor affirmed), compliant with SP 800-133Rev2. When random values are required, they are obtained from the SP 800-90ARev1 approved DRBG, compliant with Section 4 of SP 800-133Rev2. The following methods are implemented:

ECDSA (FIPS 186-5, A.2.2 Rejection Sampling): P-224, P-256, P 384, P-521 elliptic curves with 112-256 bits of key strength.

RSA (FIPS 186-5, A.1.3 Random Probable Primes): 2048, 3072, 4096 bits with 112, 128, 149 bits of key strength.

Additionally, the module implements the following key derivation methods per SP800-133Rev2 section 6.2:

KDA HKDF (SP 800-56CRev1): 112-256 bits of key strength, using (HMAC) SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512.

PBKDF (SP 800-133Rev2, option 1a): 112-256 bits of key strength, using (HMAC) SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512.

SSH KDF (SP 800-135Rev1): 112-256 bits of key strength, using AES-128, AES-192, AES-256 with SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512.

KDF TLS (SP 800-135Rev1): 112-256 bits of key strength, using SHA2-256, SHA2-384, SHA2-512.

2.10 Key Establishment

The module implements SSP agreement and SSP transport methods as listed in the Security Function Implementations table.

2.11 Industry Protocols

The module implements the SSH key derivation function for use in the SSH protocol (RFC 4253 and RFC 6668).

GCM with internal IV generation in the approved mode is compliant with versions 1.2 and 1.3 of the TLS protocol (RFC 5288 and 8446) and shall only be used in conjunction with the TLS protocol. Additionally, the module implements the TLS 1.2 and TLS 1.3 key derivation functions for use in the TLS protocol.

No parts of the SSH, TLS, other than those mentioned above, have been tested by the CAVP and CMVP.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	API input parameters for data.
N/A	Data Output	API output parameters for data.
N/A	Control Input	API function calls.
N/A	Status Output	API return codes, error message.

Table 10: Ports and Interfaces

As a Software module, the module interfaces are defined as Software or Firmware Module Interfaces (SMFI), and there are no physical ports. The module does not implement a control output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

The module does not support authentication.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 11: Roles

The module does not support concurrent operators.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Encryption	Encryption	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key, plaintext	Ciphertext	Encryption/Decryption with AES	Crypto Officer - AES Key: W,E
Decryption	Decryption	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key, ciphertext	Plaintext	Encryption/Decryption with AES	Crypto Officer - AES Key: W,E
Authenticated Encryption	Authenticated Encryption	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key, plaintext	Ciphertext	Authenticated Encryption/Decryption with AES CCM Authenticated Encryption/Decryption with AES GCM	Crypto Officer - AES Key: W,E
Authenticated Decryption	Authenticated Decryption	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key, ciphertext	Plaintext	Authenticated Encryption/Decryption with AES CCM Authenticated Encryption/Decryption with AES GCM	Crypto Officer - AES Key: W,E
Key Wrapping	Encrypting a key	Return value 1 from the function: FIPS_service_	AES key wrapping key, Key	Wrapped key	Key Wrapping/Unwrapping with AES KW, AES-KWP	Crypto Officer - AES Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		indicator_check_approved()	to be wrapped		Key Wrapping/Unwrapping with AES GCM Key Wrapping/Unwrapping with AES CCM	
Key unwrapping	Decrypting a key	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key unwrapping key, Key to be unwrapped	Unwrapped key	Key Wrapping/Unwrapping with AES KW, AES-KWP Key Wrapping/Unwrapping with AES GCM Key Wrapping/Unwrapping with AES CCM	Crypto Officer - AES Key: W,E
Message Authentication Generation	MAC computation	Return value 1 from the function: FIPS_service_indicator_check_approved()	AES key or HMAC key, message	MAC tag	Message Authentication Generation with HMAC Message Authentication Generation with AES	Crypto Officer - HMAC Key: W,E
Message Digest	Generating message digest	Return value 1 from the function: FIPS_service_indicator_check_approved()	Message	Message digest	Message Digest with SHA	Crypto Officer
Random Number Generation	Generating random numbers	Return value 1 from the function: FIPS_service_indicator_check_approved()	Output length	Random bytes	Random Number Generation with DRBG	Crypto Officer - Entropy Input: W,E - DRBG Seed: G,E - DRBG Internal State (V, Key): G,W,E
Key Generation	Generating a key pair	Return value 1 from the function: FIPS_service_indicator_check_approved()	Modulus size / Curve	RSA public key, RSA private key / EC public key, EC private key	Key Generation with RSA Key Generation with ECDSA	Crypto Officer - RSA Public Key: G,R - RSA Private Key: G,R - EC Public Key: G,R - EC Private Key: G,R

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Key Verification	Verifying the public key	Return value 1 from the function: FIPS_service_indicator_check_approved()	Public key	Success/ error	Key Verification with ECDSA	Crypto Officer - EC Public Key: W,E
Signature Generation	Generating signature	Return value 1 from the function: FIPS_service_indicator_check_approved()	Message, EC private key or RSA private key	Digital signature	Signature Generation with RSA Signature Generation with ECDSA	Crypto Officer - RSA Private Key: W,E - EC Private Key: W,E
Signature Verification	Verifying signature	Return value 1 from the function: FIPS_service_indicator_check_approved(	Signature, EC public key or RSA public key	Digital signature verification result	Signature Verification with ECDSA Signature Verification with RSA	Crypto Officer - RSA Public Key : W,E - EC Public Key: W,E
Shared Secret Computation	Calculating the Shared Secret	Return value 1 from the function: FIPS_service_indicator_check_approved()	EC public key, EC private key	Shared Secret	Shared Secret Computation with EC Diffie-Hellman	Crypto Officer - EC Public Key: W,E - EC Private Key: W,E - Shared Secret: G,R
Key Derivation with TLS KDF	Deriving Keys	Return value 1 from the function: FIPS_service_indicator_check_approved()	TLS Pre-Master Secret / TLS Master Secret	TLS Master secret / TLS Derived Key (AES/HMAC)	Key Derivation with TLS KDF	Crypto Officer - TLS Pre-Master Secret: W,E - TLS Master Secret : G,W,E - TLS Derived Key (AES/HMAC): G
Key Derivation with PBKDF	Deriving Keys	Return value 1 from the function: FIPS_service_indicator_check_approved()	Password, salt, iteration count	PBKDF Derived Key	Key Derivation with PBKDF	Crypto Officer - PBKDF Derived Key: G,R - Password: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Key Derivation with KDA HKDF	Deriving Keys	Return value 1 from the function: FIPS_service_indicator_check_approved()	Shared Secret, Key Length, Digest	KDA Derived Key	Key Derivation with KDA HKDF	Crypto Officer - KDA Derived Key: G,R - Shared Secret: W,E
Key Derivation with SSH KDF	Deriving Keys	Return value 1 from the function: FIPS_service_indicator_check_approved()	Shared Secret, Key Length	SSH Derived Key	Key Derivation with SSH KDF	Crypto Officer - SSH Derived Key: G,R - Shared Secret: W,E
Zeroization	Zeroize SSP in volatile memory	N/A	SSP	N/A	None	Crypto Officer - AES Key: Z - HMAC Key: Z - Entropy Input: Z - DRBG Seed: Z - DRBG Internal State (V, Key): Z - RSA Public Key : Z - RSA Private Key: Z - RSA Private Key: Z - EC Public Key: Z - EC Private Key: Z - Shared Secret: Z - TLS Pre-Master Secret: Z - TLS Master Secret : Z - TLS Derived Key (AES/HMAC): Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS Derived Key (AES/HMAC): Z - Password: Z - Intermediate Key Generation Value: Z
On-Demand Self-test	Initiate power-on self-tests by reset	N/A	N/A	Pass or fail	Shared Secret Computation with EC Diffie-Hellman Key Wrapping/Unwrapping with AES KW, AES-KWP Key Wrapping/Unwrapping with AES GCM Key Wrapping/Unwrapping with AES CCM Encryption/Decryption with AES Signature Generation with RSA Signature Generation with ECDSA Key Generation with RSA Key Generation with ECDSA Key Generation with RSA Signature Verification with ECDSA Signature Verification with RSA Key Verification with ECDSA Key Derivation with TLS KDF Key Derivation with SSH KDF Key Derivation with KDA HKDF Key Derivation with PBKDF Message Digest	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					with SHA Random Number Generation with DRBG Message Authentication Generation with HMAC Message Authentication Generation with AES Authenticated Encryption/Decryption with AES CCM Authenticated Encryption/Decryption with AES GCM	
On-Demand Integrity Test	Initiate integrity test on-demand	N/A	N/A	Pass or fail	Message Authentication Generation with HMAC	Crypto Officer
Show Status	Show status of the module state	N/A	N/A	Module status	None	Crypto Officer
Show Version	Show the version of the module using awslc_version_string	N/A	N/A	Module name and version	None	Crypto Officer

Table 12: Approved Services

For the above table, the convention below applies when specifying the access permissions (types) that the service has for each SSP.

- R = Read: The SSP is read from the module (e.g., the SSP is output).
- W = Write: The SSP is updated, imported, or written to the module.
- E = Execute: The module uses the SSP in performing a cryptographic operation.
- Z = Zeroize: The module zeroizes the SSP.

For the role, CO indicates “Crypto Officer”.

The module implements a service indicator that indicates whether the invoked service is approved. The service indicator is a return value 1 from the FIPS_service_indicator_check_approved function. This function is used together with two other functions. The usage is as follows:

- STEP 1: Should be called before invoking the service.

  int before = FIPS_service_indicator_before_call();

- STEP 2: Make a service call i.e., API function for performing a service.

  Func();

- STEP 3: Should be called after invoking the service.

int after = FIPS_service_indicator_after_call();

- STEP 4: Return value 1 indicates approved service was invoked.

  int ret = FIPS_service_indicator_check_approved(before, after);

Alternatively, all the above steps can be done by using a single call using the function CALL_SERVICE_AND_CHECK_APPROVED(approved, func).

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Encryption	Encryption	AES with OFB or CFB1, CFB8 modes AES GCM, GCM, GMAC, XTS with keys not listed in Table 5 AES using aes_*_generic function AES GMAC using aes_*_generic RSA encryption primitive with PKCS#1 v1.5 and OAEP padding	CO
Decryption	Decryption	AES with OFB or CFB1, CFB8 modes AES GCM, GCM, GMAC, XTS with keys not listed in Table 5 AES using aes_*_generic function AES GMAC using aes_*_generic	CO
Message Authentication Generation	MAC computation	AES GMAC using aes_*_generic HMAC-MD4, HMAC-MD5, HMAC-SHA1, HMAC-SHA-3, HMAC-RIPEMD-160	CO
Message Digest	Generating message digest	MD4 MD5 (outside of TLS) SHAKE, RIPEMD-160, SHA-3	CO
Signature Generation	Generating signatures	RSA using keys less than 2048 bits RSA without hashing SHA-1, SHA-3	CO
Signature Verification	Verifying signatures	RSA using keys less than 1024 bits RSA without hashing	CO
Key Generation	Generating key pair	RSA using RSA_generate_key_ex ECDSA using EC_KEY_generate_key	CO
Shared Secret Computation	Calculating shared secret	Curve secp256k1 Diffie Hellman	CO
Key Derivation	Deriving TLS keys	TLS KDF using any SHA algorithms other than SHA2-256, SHA2-384, SHA2-512; or TLS KDF using non-extended master secret	CO
Key Encapsulation	Encrypting a key	RSA	CO
Key Un-encapsulation	Decrypting a key	RSA	CO

Table 13: Non-Approved Services

4.5 External Software/Firmware Loaded

The module does not support loading of external software or firmware.

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module is verified by comparing a HMAC value calculated at run time on the bcm.o file, with the HMAC-SHA2-256 value stored within the module that was computed at build time.

5.2 Initiate on Demand

The module provides on-demand integrity test. The integrity test can be performed on demand by reloading the module. Additionally, the integrity test can be performed using the On-Demand Integrity Test service, which calls the BORINGSSL_integrity_test function.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied:

The module should be compiled and installed as stated in section 11. The user should confirm that the module is installed correctly by following steps 4 and 5 listed in section 11.

6.2 Configuration Settings and Restrictions

Instrumentation tools like the ptrace system call, gdb and strace, userspace live patching, as well as other tracing mechanisms offered by the Linux environment such as ftrace or systemtap, shall not be used in the operational environment. The use of any of these tools implies that the cryptographic module is running in a non-validated operational environment.

7 Physical Security

7.1 Mechanisms and Actions Required

N/A for this module.

The module is comprised of software only and therefore this section is not applicable.

7.4 Fault Induction Mitigation

7.5 EFP/EFT Information

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature			
HighTemperature			
LowVoltage			
HighVoltage			

Table 14: EFP/EFT Information

7.6 Hardness Testing Temperature Ranges

Temperature Type	Temperature
LowTemperature	
HighTemperature	

Table 15: Hardness Testing Temperatures

8 Non-Invasive Security

8.1 Mitigation Techniques

The module claims no non-invasive security techniques.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	Temporary storage for SSPs used by the module as part of service execution. The module does not perform persistent storage of SSPs	Dynamic

Table 16: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API input parameters	Operator calling application (TOEPP)	Cryptographic module	Plaintext	Manual	Electronic	
API output parameters	Cryptographic module	Operator calling application (TOEPP)	Plaintext	Manual	Electronic	

Table 17: SSP Input-Output Methods

The module does not support entry and output of SSPs beyond the physical perimeter of the operational environment. The SSPs are provided to the module via API input parameters in the plaintext form and output via API output parameters in the plaintext form to and from the calling application running on the same operational environment.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Free Cipher Handle	Zeroizes the SSPs contained within the cipher handle.	Memory occupied by SSPs is overwritten with zeroes, which renders the SSP values irretrievable.	By calling the appropriate zeroization functions: OpenSSL_cleanse, EVP_CIPHER_CTX_cleanup, EVP_AEAD_CTX_zero, HMAC_CTX_cleanup, CTR_DRBG_clear, RSA_free, EC_KEY_free
Module Reset	De-allocates the volatile memory used to store SSPs	Volatile memory used by the module is overwritten within nanoseconds when power is removed.	By unloading and reloading the module.
Automatically	Automatically zeroized when no longer needed	Memory occupied by SSPs is overwritten with zeros, which renders the SSP values irretrievable.	N/A

Table 18: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES Key	AES key used for encryption, decryption, and computing MAC tags	128-256 bits - 128-256 bits	Symmetric key - CSP			Key Wrapping/Unwrapping with AES KW, AES-KWP Key Wrapping/Unwrapping with AES GCM Key Wrapping/Unwrapping with AES CCM Encryption/Decryption with AES Message Authentication Generation with AES Authenticated Encryption/Decryption with AES CCM Authenticated Encryption/Decryption with AES GCM
HMAC Key	HMAC key for Message Authentication Generation	112-524288 bits - 112-256 bits	Authentication key - CSP			Message Authentication Generation with HMAC
Entropy Input	Entropy input used to seed the DRBGs	256 bits - 256 bits	Entropy - CSP			Random Number Generation with DRBG
DRBG Seed	DRBG seed derived from entropy input as defined in SP 800-90Ar1	256 bits - 256 bits	DRBG seed - CSP	Random Number Generation with DRBG		Random Number Generation with DRBG
DRBG Internal State (V, Key)	Internal state of CTR_DRBG	256 bits - 256 bits	Internal state - CSP	Random Number Generation with DRBG		Random Number Generation with DRBG
RSA Public Key	RSA public key used for RSA key generation, signature verification	1024, 2048, 3072, 4096 bits - 80-150 bits	Public key - PSP	Key Generation with RSA		Key Generation with RSA Signature Verification with RSA
RSA Private Key	RSA private key used for RSA key generation, signature generation	2048, 3072, 4096 bits - 112-150 bits	Private key - CSP	Key Generation with RSA		Signature Generation with RSA Key Generation with RSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
EC Public Key	EC public key used for EC key generation, key verification, signature verification, shared secret computation	P-224, P-256, P-384, P-521 - 112-256 bits	Public key - PSP	Key Generation with ECDSA		Shared Secret Computation with EC Diffie-Hellman Key Generation with ECDSA Signature Verification with ECDSA
EC Private Key	EC private key used for EC key generation, key verification, signature generation, shared secret computation	P-224, P-256, P-384, P-521 - 112-256 bits	Private key - CSP	Key Generation with ECDSA		Shared Secret Computation with EC Diffie-Hellman Signature Generation with ECDSA Key Generation with ECDSA
Shared Secret	Shared Secret generated by KAS-ECC-SSC	P-224, P-256, P-384, P-521 - 112-256 bits	Shard secret - CSP		Shared Secret Computation with EC Diffie-Hellman	Key Derivation with TLS KDF Key Derivation with SSH KDF Key Derivation with KDA HKDF
TLS Pre-Master Secret	TLS Pre-Master secret used for deriving the TLS Master Secret	112-256 bits - N/A	TLS pre-master secret - CSP			Key Derivation with TLS KDF Key Derivation with KDA HKDF
TLS Master Secret	TLS Master secret used for deriving the TLS Derived Key	384 bits - N/A	TLS master secret - CSP	Key Derivation with TLS KDF Key Derivation with KDA HKDF		Key Derivation with TLS KDF Key Derivation with KDA HKDF
TLS Derived Key (AES/HMAC)	TLS Derived Key from TLS Master Secret	AES: 128-256 bits HMAC: 112 to 256 bits - AES: 128-256 bits HMAC: 112 to 256 bits	Symmetric key - CSP	Key Derivation with TLS KDF		Key Derivation with TLS KDF
KDA Derived Key	KDA HKDF derived key	112 to 256 bits - N/A	Symmetric key - CSP	Key Derivation with KDA HKDF		Key Derivation with KDA HKDF

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SSH Derived Key	SSH KDF derived key	112 to 256 bits - N/A	Symmetric key - CSP	Key Derivation with SSH KDF		Key Derivation with SSH KDF
PBKDF Derived Key	PBKDF derived key	112 to 256 bits - N/A	Symmetric key - CSP	Key Derivation with PBKDF		Key Derivation with PBKDF
Password	Password for PBKDF	112-524288 bits - N/A	Password - CSP			Key Derivation with PBKDF
Intermediate Key Generation Value	Intermediate key generation value	224-4096 bits - 112-256 bits	Intermediate value - CSP	Key Generation with RSA Key Generation with ECDSA		Key Generation with ECDSA Key Generation with RSA

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES Key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Free Cipher Handle Module Reset	
HMAC Key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Free Cipher Handle Module Reset	
Entropy Input	API input parameters	RAM:Plaintext	from service invocation to service completion	Automatically	DRBG Seed:Generation Of
DRBG Seed		RAM:Plaintext	from service invocation to service completion	Automatically	Entropy Input:Derived From
DRBG Internal State (V, Key)			from service invocation to service completion	Automatically	DRBG Seed:Derived From
RSA Public Key	API input parameters API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	RSA Private Key:Paired With
RSA Private Key	API input parameters API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	RSA Public Key :Paired With
EC Public Key	API input parameters	RAM:Plaintext	from service invocation to	Free Cipher Handle Module Reset	EC Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	API output parameters		service completion		Shared Secret:Generation Of
EC Private Key	API input parameters API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	EC Public Key:Paired With Shared Secret:Generation Of
Shared Secret	API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	EC Public Key:Derived From EC Private Key:Derived From
TLS Pre-Master Secret	API input parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	TLS Master Secret :Derivation Of
TLS Master Secret		RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	TLS Pre-Master Secret:Derived From
TLS Derived Key (AES/HMAC)	API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	TLS Master Secret :Derived From
KDA Derived Key	API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	Shared Secret:Derived From
SSH Derived Key	API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	Shared Secret:Derived From
PBKDF Derived Key	API output parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	Password:Derived From
Password	API input parameters	RAM:Plaintext	from service invocation to service completion	Free Cipher Handle Module Reset	Derived Key:Derivation Of
Intermediate Key Generation Value			from service invocation to service completion	Automatically	RSA Public Key :Generation Of RSA Private Key:Generation Of EC Public Key:Generation Of EC Private Key:Generation Of

Table 20: SSP Table 2

9.5 Transitions

The SHA-1 algorithm as implemented by the module will be non-approved for all purposes, starting January 1, 2030.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A4509)	SHA2-256	Message Authentication	SW/FW Integrity	Module becomes operational	Integrity test for bcm.o

Table 21: Pre-Operational Self-Tests

The module performs the pre-operational self-test automatically when the module is loaded into memory; the pre-operational self-test is the software integrity test that ensures that the module is not corrupted. While the module is executing the pre-operational self-test, services are not available, and input and output are inhibited.

The software integrity test is performed after a set of conditional cryptographic algorithm self-tests (CASTs). The set of CASTs executed before the software integrity test consists of HMAC-SHA2-256 KAT, which is used in the pre-operational self-test, and the SHA2-256 KAT.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CBC (A4513)	128-bit AES key	Encrypt KAT	CAST	Module is operational	Encrypt	Power up
AES-CBC (A4510)	128-bit AES key	Decrypt KAT	CAST	Module is operational	Decrypt	Power up
AES-GCM (A4511)	128-bit AES key	Encrypt KAT	CAST	Module is operational	Encrypt	Power up
AES-GCM (A4511)	128-bit AES key	Decrypt KAT	CAST	Module is operational	Decrypt	Power up
SHA-1 (A4509)	N/A	SHA-1 KAT	CAST	Module is operational	Message digest	Power up
SHA2-256 (A4509)	N/A	SHA2-256 KAT	CAST	Module is operational	Message digest	Power up
SHA2-512 (A4509)	N/A	SHA2-512 KAT	CAST	Module is operational	Message digest	Power up
HMAC-SHA2-256 (A4509)	SHA2-256	HMAC KAT	CAST	Module is operational	Message authentication	Power up
Counter DRBG (A4513)	AES 256	CTR_DRBG KAT	CAST	Module is operational	Seed Generation	Power up
Counter DRBG (A4513)	N/A	SP800-90Ar1 Section 11.3 Health Test	CAST	Module is operational	Seed Generation	Power up
ECDSA SigGen (FIPS186-5) (A4509)	P-256 Curve and SHA2-256	Sign KAT	CAST	Module is operational	Sign	Signature Generation or Key Generation service request

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA SigVer (FIPS186-4) (A4509)	P-256 Curve and SHA2-256	Verify KAT	CAST	Module is operational	Verify	Signature verification or Key Generation service request
KAS-ECC-SSC Sp800-56Ar3 (A4509)	P-256 Curve	Z computation	CAST	Module is operational	Shared secret computation	Shared secret computation request
ECDSA KeyGen (FIPS186-5) (A4509)	Respective Curve and SHA2-256	Signature generation and verification	PCT	Module is operational	Sign and Verify	Key generation
KDF TLS (A4509)	SHA2-256	TLS 1.2 KAT	CAST	Module is operational	Key derivation	Power up
KDA HKDF Sp800-56Cr1 (A4509)	HMAC-SHA2-256	KDA HKDF KAT	CAST	Module is operational	Key derivation	Power up
PBKDF (A4509)	HMAC-SHA2-256	PBKDF2 KAT	CAST	Module is operational	Key derivation	Power up
RSA SigGen (FIPS186-5) (A4509)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	Sign KAT	CAST	Module is operational	Sign	Signature Generation or Key Generation service request
RSA SigVer (FIPS186-4) (A4509)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	Verify KAT	CAST	Module is operational	Verify	Signature Verification or Key Generation service request
RSA KeyGen (FIPS186-5) (A4509)	SHA2-256 and respective keys	Signature generation and verification	PCT	Module is operational	Sign and Verify	Key generation

Table 22: Conditional Self-Tests

Conditional Cryptographic Algorithm Tests

The module performs self-tests on approved cryptographic algorithms, using the tests shown in Table 22. Data output through the data output interface is inhibited during the self-tests. The CASTs are performed in the form of Known Answer Tests (KATs), in which the calculated output is compared with the expected known answer (that are hard-coded in the module). A failed match causes a failure of the self-test. If any of these self-tests fails, the module transitions to error state.

Conditional Pair-Wise Consistency Tests

The module implements RSA and ECDSA key generation service and performs the respective pairwise consistency test (PCT) using sign and verify functions when the keys are generated (Table 22). If any of these self-tests fails, the module transitions to error state and is aborted.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A4509)	Message Authentication	SW/FW Integrity	On demand	Manually

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A4513)	Encrypt KAT	CAST	On demand	Manually
AES-CBC (A4510)	Decrypt KAT	CAST	On demand	Manually
AES-GCM (A4511)	Encrypt KAT	CAST	On demand	Manually
AES-GCM (A4511)	Decrypt KAT	CAST	On demand	Manually
SHA-1 (A4509)	SHA-1 KAT	CAST	On demand	Manually
SHA2-256 (A4509)	SHA2-256 KAT	CAST	On demand	Manually
SHA2-512 (A4509)	SHA2-512 KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A4509)	HMAC KAT	CAST	On demand	Manually
Counter DRBG (A4513)	CTR_DRBG KAT	CAST	On demand	Manually
Counter DRBG (A4513)	SP800-90Ar1 Section 11.3 Health Test	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A4509)	Sign KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-4) (A4509)	Verify KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A4509)	Z computation	CAST	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A4509)	Signature generation and verification	PCT	On demand	Manually
KDF TLS (A4509)	TLS 1.2 KAT	CAST	On demand	Manually
KDA HKDF Sp800-56Cr1 (A4509)	KDA HKDF KAT	CAST	On demand	Manually
PBKDF (A4509)	PBKDF2 KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A4509)	Sign KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-4) (A4509)	Verify KAT	CAST	On demand	Manually
RSA KeyGen (FIPS186-5) (A4509)	Signature generation and verification	PCT	On demand	Manually

Table 24: Conditional Periodic Information

The module does not support periodic self-tests.

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error	The library is aborted with SIGABRT signal. Module is no longer operational the data output interface is inhibited	Pre-operational test failure	Module reset	Error message is output on the stderr and then the module is aborted.
PCT Error	The library is aborted with SIGABRT signal. Module is no longer operational the data output interface is inhibited	Conditional test failure	Module reset	For CAST failure, an error message is output on the stderr and then the module is aborted. For PCT failure, an error message is output in the error queue and then the module generates new key, If the PCT still does not pass, eventually the module will be aborted after 5 tries.

Table 25: Error States

If the module fails any of the self-tests, the module enters an error state. To recover from any error state, the module must be rebooted.

10.5 Operator Initiation of Self-Tests

The software integrity tests and the CASTs for AES, SHS, DRBG, HMAC, KAS-ECC-SSC, TLS KDF, KDA HKDF, PBKDF2 can be invoked by unloading and subsequently re-initializing the module. The CASTs for ECDSA and RSA can be invoked by requesting the corresponding Key Generation or Digital Signature services. Additionally, all the CASTs can be invoked by calling the BORINGSSL_self_test function. The PCTs can be invoked on demand by requesting the Key Generation service.

10.6 Additional Information

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The module bcm.o is embedded into the usersapce application which can be obtained by building the source code at the following location [1]. The set of files specified in the archive constitutes the complete set of source files of the validated module. There shall be no additions, deletions, or alterations of this set as used during module build.

[1] https://github.com/aws/aws-lc/archive/refs/tags/AWS-LC-FIPS-2.0.0.zip

The downloaded zip file can be verified by issuing the “sha256sum AWS-LC-FIPS-2.0.0.zip” command. The expected SHA2-256 digest value is:

6241EC2F13A5F80224EE9CD8592ED66A97D426481066FEAA4EFC6F24E60BBC96

After the zip file is extracted, the instructions listed below will compile the module. The compilation instructions must be executed separately on platforms that have different processors and/or operating systems. Due to six possible combinations of OS/processor, the module count is six (i.e., there are six separate binaries generated, one for each entry listed in the Tested Operational Environments table).

Amazon Linux 2 and Amazon Linux 2023:

1. sudo yum groupinstall "Development Tools"
2. sudo yum install cmake3 golang
3. cd aws-lc-fips-2022-11-02/
4. mkdir build
5. cd build
6. cmake3 -DFIPS=1 ..
7. make

Ubuntu 22.04:

1. sudo apt-get install build-essential
2. sudo apt-get install cmake
3. Get latest Golang archive for your architecture
4. sudo tar -C /usr/local -xzf go*.tar.gz
5. cd aws-lc-fips-2022-11-02/
6. mkdir build
7. cd build
8. cmake -DFIPS=1 -DGO_EXECUTABLE=/usr/local/go/bin/go ..
9. make

Upon completion of the build process, the module’s status can be verified by the command below. If the value obtained is “1” then the module has been installed and configured to operate in FIPS compliant manner.

```
./tool/bssl isfips
```

Lastly, the user can call the “show version” service using awslc_version_string function and the expected output is “AWS-LC FIPS 2.0.0” which is the module version. This will confirm that the module is in the operational mode. Additionally, the “AWS-LC FIPS” also acts as the module identifier and the verification of the "static" part can be done using following command with an application that was used for static linking. The "T" in the output confirms that the module is statically linked.

Command: *nm <application_name> | grep awslc_version_string*

Example Output: *0000000000a5bdff **T** awslc_version_string*

11.2 Administrator Guidance

When the module is at end of life, for the GitHub repo, the README will be modified to mark the library as deprecated. After a 6-month window, more restrictive branch permissions will be added such that only administrators can read from the FIPS branch.

The module does not possess persistent storage of SSPs. The SSP value only exists in volatile memory and that value vanishes when the module is powered off. So as a first step for the secure sanitization, the module needs to be powered off. Then for actual deprecation, the module will be upgraded to newer version that is approved. This upgrade process will uninstall/remove the old/terminated module and provide a new replacement.

12 Mitigation of Other Attacks

12.1 Attack List

RSA timing attacks.

12.2 Mitigation Effectiveness

RSA is vulnerable to timing attacks. In a setup where attackers can measure the time of RSA decryption or signature operations, blinding must be used to protect the RSA operation from that attack.

The module provides the mechanism to use the blinding for RSA. When the blinding is on, the module generates a random value to form a blinding factor in the RSA key before the RSA key is used in the RSA cryptographic operations.