Apple Inc.

Apple corecrypto Module v13.0 [Apple silicon, Kernel, Software, SL1]

FIPS 140-3 Non-Proprietary Security Policy

Prepared for:
Apple Inc.
One Apple Park Way
Cupertino, CA 95014
Prepared by:
atsec information security corporation
4516 Seton Center Parkway, Suite 250
Austin, TX 78759
www.atsec.com

Table of Contents

List of Tables

List of Figures

Trademarks

Apple’s trademarks applicable to this document are listed in https://www.apple.com/legal/intellectual-property/trademark/appletmlist.html.
Other company, product, and service names may be trademarks or service marks of others.

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for Apple corecrypto Module v13.0 [Apple silicon, Kernel, Software, SL1] cryptographic module. 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 a Security Level 1 module.
This document provides all tables and diagrams (when applicable) required by NIST SP 800-140Br1.

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	N/A
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use: The Apple corecrypto Module v13.0 [Apple silicon, Kernel, Software, SL1] cryptographic module (hereafter referred to as "the module") provides implementations of low-level cryptographic primitives to the Device OS's kernels (iOS 16, iPadOS 16, watchOS 9, tvOS 16) Security Framework and Common Crypto. The module provides services intended to protect data in transit and at rest.

The module is optimized for library use within the Device OS kernel space and does not contain any terminating assertions or exceptions. It is implemented as a Device OS dynamically loadable library. The library is loaded into the Device OS kernel and its cryptographic functions are made available to Device OS kernel services only.

Any internal error detected by the module is returned to the caller with an appropriate return code. The calling Device OS kernel service must examine the return code and act accordingly. The module communicates any error status synchronously through the use of its documented return codes, thus indicating the module's status. Caller-induced or internal errors do not reveal any sensitive material to callers.

Module Type: Software

Module Embodiment: MultiChipStand

Cryptographic Boundary: The module cryptographic boundary is delineated by the dotted green rectangle in the Figure 1 where the Kernel Extension (KEXT) is a bundle that performs low-level tasks. KEXTs run in kernel space, which gives them elevated privileges and the ability to perform tasks that user-space apps can't.

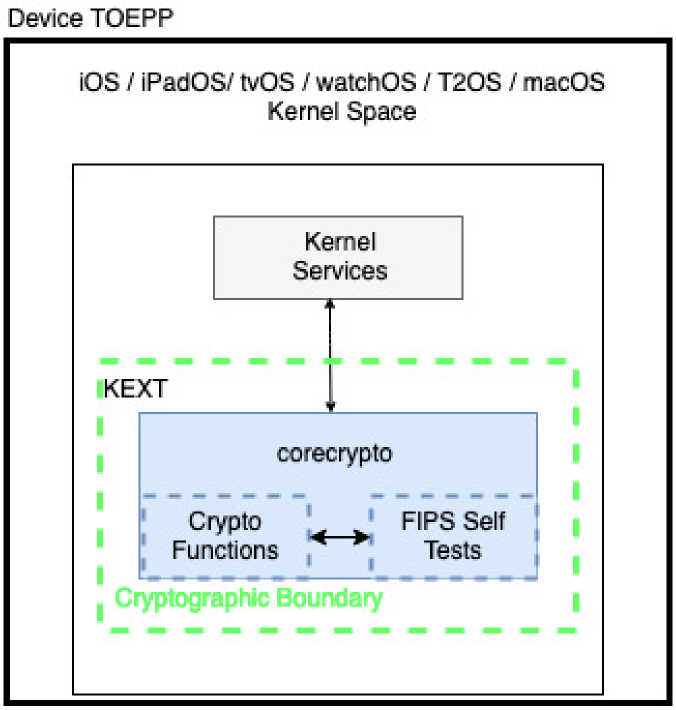

Figure 1: Block Diagram

Tested Operational Environment's Physical Perimeter (TOEPP): The physical perimeter is represented by the most exterior black line in the block diagram Figure 1. The module executes within the kernel space of the computing platforms and operating systems listed in the Tested Operational Environments Table section 2.2.

2.2 Tested and Vendor Affirmed Module Version and Identification

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
xnu-10002.60.75.0.3	v13.0	N/A	HMAC-SHA256

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

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
iPadOS 16	iPad (5th generation)	Apple A Series A9	Yes	NA	v13.0
iPadOS 16	iPad Pro 9.7-inch	Apple A Series A9X	Yes	NA	v13.0
iPadOS 16	iPad (7th generation)	Apple A Series A10 Fusion	Yes	NA	v13.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
iPadOS 16	iPad mini (5th generation)	Apple A Series A12 Bionic	Yes	NA	v13.0
iPadOS 16	iPad Pro 11-inch (1st generation)	Apple A Series A12X Bionic	Yes	NA	v13.0
iPadOS 16	iPad Pro 11-inch (2nd generation)	Apple A Series A12Z Bionic	Yes	NA	v13.0
iPadOS 16	iPad (9th generation)	Apple A Series A13 Bionic	Yes	NA	v13.0
iPadOS 16	iPad Air (4th generation)	Apple A Series A14 Bionic	Yes	NA	v13.0
iPadOS 16	iPad mini (6th generation)	Apple A Series A15 Bionic	Yes	NA	v13.0
iPadOS 16	iPad Pro 11-inch (3rd generation)	Apple M Series M1	Yes	NA	v13.0
iPadOS 16	iPad Pro 11-inch (4th generation)	Apple M Series M2	Yes	NA	v13.0
iOS 16	iPhone X	Apple A Series A11 Bionic	Yes	NA	v13.0
iOS 16	iPhone XS Max	Apple A Series A12 Bionic	Yes	NA	v13.0
iOS 16	iPhone 11 Pro	Apple A Series A13 Bionic	Yes	NA	v13.0
iOS 16	iPhone 12	Apple A Series A14 Bionic	Yes	NA	v13.0
iOS 16	iPhone 13 Pro Max	Apple A Series A15 Bionic	Yes	NA	v13.0
iOS 16	iPhone 14 Pro Max	Apple A Series A16 Bionic	Yes	NA	v13.0
watchOS 9	Apple Watch Series S4	Apple S Series S4	Yes	NA	v13.0
watchOS 9	Apple Watch Series S5	Apple S Series S5	Yes	NA	v13.0
watchOS 9	Apple Watch Series S6	Apple S Series S6	Yes	NA	v13.0
watchOS 9	Apple Watch Series S7	Apple S Series S7	Yes	NA	v13.0
watchOS 9	Apple Watch Series S8	Apple S Series S8	Yes	NA	v13.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
iPadOS 16	iPad Pro 10.5-inch	Apple A Series A10X Fusion	Yes	NA	v13.0
tvOS 16	Apple TV 4K (2nd generation)	Apple A Series A12 Bionic	Yes	NA	v13.0
tvOS 16	Apple TV 4K (3rd generation)	Apple A Series A15 Bionic	Yes	NA	v13.0

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

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

Operating System	Hardware Platform
iPadOS 16	iPad Pro 12.9-inch
iPadOS 16	iPad (6th generation)
iPadOS 16	iPad Pro 12.9-inch (2nd generation)
iPadOS 16	iPad Air (3rd generation)
iPadOS 16	iPad (8th generation)
iPadOS 16	iPad Pro 12.9-inch (3rd generation)
iPadOS 16	iPad Pro 12.9-inch (4th generation)
iPadOS 16	iPad Pro 12.9-inch (5th generation)
iPadOS 16	iPad Pro 12.9-inch (6th generation)
iOS 16	iPhone 8
iOS 16	iPhone 8 Plus
iOS 16	iPhone XS
iOS 16	iPhone XR
iOS 16	iPhone 11
iOS 16	iPhone 11 Pro Max
iOS 16	iPhone SE (2nd generation)
iOS 16	iPhone 12 mini
iOS 16	iPhone 12 Pro
iOS 16	iPhone 12 Pro Max
iOS 16	iPhone 13 mini
iOS 16	iPhone 13
iOS 16	iPhone 13 Pro
iOS 16	iPhone 14 Pro
watchOS 9	Apple Watch SE
macOS 13 Ventura	Mac mini
macOS 13 Ventura	iMac (24-inch)
macOS 13 Ventura	MacBook Pro (14-inch, 2021)
macOS 13 Ventura	MacBook Air

Table 4: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid

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

None for this module

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved mode	Approved mode of operation is entered when the module utilizes the services that use the security functions listed in the Approved Algorithms Table and the Vendor Affirmed Algorithms Table.	Approved	return a '1' from fips_allowed_mode() for block cipher functions and fips_allowed() for all other services to indicate the executed cryptographic algorithm was approved
Non-Approved mode	Non-Approved mode of operation is entered when the module utilizes non-approved security functions in the Table Non-Approved Algorithms Not Allowed in the Approved Mode of Operation.	Non-Approved	return a '0' from fips_allowed_mode() for block cipher functions and fips_allowed() for all other services to indicate the executed cryptographic algorithm was non- approved

Table 5: Modes List and Description

2.5 Algorithms

Approved Algorithms:

AES-CBC

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A3682	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CBC	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 6: Approved Algorithms - AES-CBC

AES-CCM

Algorithm	CAVP Cert	Properties	Reference
AES-CCM	A3685	Key Length - 128, 192, 256	SP 800-38C

Table 7: Approved Algorithms - AES-CCM

AES-CFB128

Algorithm	CAVP Cert	Properties	Reference
AES-CFB128	A3682	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB128	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 8: Approved Algorithms - AES-CFB128

AES-CFB8

Algorithm	CAVP Cert	Properties	Reference
AES-CFB8	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 9: Approved Algorithms - AES-CFB8

AES-CTR

Algorithm	CAVP Cert	Properties	Reference
AES-CTR	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A3685	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 10: Approved Algorithms - AES-CTR

AES-ECB

Algorithm	CAVP Cert	Properties	Reference
AES-ECB	A3682	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3685	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 11: Approved Algorithms - AES-ECB

AES-GCM

Algorithm	CAVP Cert	Properties	Reference
AES-GCM	A3685	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256	SP 800-38D

Table 12: Approved Algorithms - AES-GCM

AES-KW

Algorithm	CAVP Cert	Properties	Reference
AES-KW	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F

Table 13: Approved Algorithms - AES-KW

AES-OFB

Algorithm	CAVP Cert	Properties	Reference
AES-OFB	A3682	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-OFB	A3683	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Table 14: Approved Algorithms - AES-OFB

AES-XTS

Algorithm	CAVP Cert	Properties	Reference
AES-XTS Testing Revision 2.0	A3682	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E

Table 15: Approved Algorithms - AES-XTS

CTR_DRBG

Algorithm	CAVP Cert	Properties	Reference
Counter DRBG	A3683	Prediction Resistance - No Mode - AES-128, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
Counter DRBG	A3685	Prediction Resistance - No Mode - AES-128, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1

Table 16: Approved Algorithms - CTR_DRBG

ECDSA-KEYGEN

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyGen (FIPS186-4)	A3686	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - Testing Candidates	FIPS 186-4

Table 17: Approved Algorithms - ECDSA-KEYGEN

ECDSA-KEYVER

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyVer (FIPS186-4)	A3686	Curve - P-224, P-256, P-384, P-521	FIPS 186-4

Table 18: Approved Algorithms - ECDSA-KEYVER

ECDSA-SIGGEN

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigGen (FIPS186-4)	A3686	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4

Table 19: Approved Algorithms - ECDSA-SIGGEN

ECDSA-SIGVER

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigVer (FIPS186-4)	A3686	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4

Table 20: Approved Algorithms - ECDSA-SIGVER

HMAC-SHA1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 21: Approved Algorithms - HMAC-SHA1

HMAC-SHA224

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-224	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 22: Approved Algorithms - HMAC-SHA224

HMAC-SHA256

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-256	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3687	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 23: Approved Algorithms - HMAC-SHA256

HMAC-SHA384

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-384	A3684	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 24: Approved Algorithms - HMAC-SHA384

HMAC-SHA512

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-512	A3684	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 25: Approved Algorithms - HMAC-SHA512

HMAC-SHA512/256

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-512/256	A3684	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A3686	Key Length - Key Length: 8-262144 Increment 8	FIPS 198-1

Table 26: Approved Algorithms - HMAC-SHA512/256

RSA-SIGGEN

Algorithm	CAVP Cert	Properties	Reference
RSA SigGen (FIPS186-4)	A3686	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4

Table 27: Approved Algorithms - RSA-SIGGEN

RSA-SIGVER

Algorithm	CAVP Cert	Properties	Reference
RSA SigVer (FIPS186-4)	A3686	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4

Table 28: Approved Algorithms - RSA-SIGVER

SHA1

Algorithm	CAVP Cert	Properties	Reference
SHA-1	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 29: Approved Algorithms - SHA1

SHA224

Algorithm	CAVP Cert	Properties	Reference
SHA2-224	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 30: Approved Algorithms - SHA224

SHA256

Algorithm	CAVP Cert	Properties	Reference
SHA2-256	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4
SHA2-256	A3687	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 31: Approved Algorithms - SHA256

SHA384

Algorithm	CAVP Cert	Properties	Reference
SHA2-384	A3684	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4
SHA2-384	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 32: Approved Algorithms - SHA384

SHA512

Algorithm	CAVP Cert	Properties	Reference
SHA2-512	A3684	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4
SHA2-512	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 33: Approved Algorithms - SHA512

SHA512/256

Algorithm	CAVP Cert	Properties	Reference
SHA2-512/256	A3684	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4
SHA2-512/256	A3686	Message Length - Message Length: 0-32768 Increment 8	FIPS 180-4

Table 34: Approved Algorithms - SHA512/256

The FIPS 186-4 CAVP tests in the listed ACVP certificates above are mathematically identical to the FIPS 186-5 CAVP tests. Per FIPS 140-3 C.K Additional Comments 2, the module claims compliance with FIPS 186-5 tests.

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	SP800-133rev2 section 4 example 1

Table 35: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
ANSI X9.63 KDF	Hash based Key Derivation Function
Blowfish	Encryption / Decryption
CAST5	Encryption / Decryption Key Sizes: 40 to 128 bits in 8-bit increments
DES	Encryption / Decryption Key Size: 56-bits
ECDSA	Generation / Verification / SigGen / SigVer with curve P-192
ECDSA KeyGen	Key Pair Generation for compact point representation of points
EdDSA	Key Generation, Signature Generation, Signature Verification with Ed25519
HKDF [SP800-56Crev2]	Key Derivation Function
Integrated Encryption Scheme on elliptic curves (ECIES)	Encryption / Decryption
MD2	Message Digest size: 128-bit
MD4	Message Digest size: 128-bit
OMAC (One-Key CBC MAC)	MAC generation /verification
RC2	Encryption / Decryption Key Sizes 8 to 1024-bits
RC4	Encryption / Decryption Key Sizes 8 to 4096-bits
RIPEMD	Message Digest size: 160-bits
RSA SigGen	PKCS#1 v1.5 and PSS; Signature Generation Key Size < 2048
RSA SigVer	Signature Verification Key Size < 1024
RSA Key Wrapping	OAEP, PKCS#1 v1.5 and -PSS schemes
Triple-DES [SP 800-67r2]	CBC, CTR, CFB64, ECB, CFB8, OFB
MD5	Message Digest size: 128-bit
RFC 6637 Key Derivation	SHA-256, SHA-512, AES-128, AES-256

Table 36: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Unauthenticated Symmetric Encryption and Decryption	BC-UnAuth	Key Size / Key Strength: 128, 192, 256-bits (for all but XTS, which supports 128 and 256 bit keys)	AES [FIPS 197; SP 800-38A]:ECB, CBC, CFB8, CFB128, OFB, CTR AES [FIPS 197; SP 800-38E]:XTS	AES-CBC: (A3682, A3683) AES-CFB128: (A3682, A3683) AES-XTS Testing Revision 2.0: (A3682) AES-ECB: (A3682, A3683, A3685) AES-OFB: (A3682, A3683) AES-CFB8: (A3683) AES-CTR: (A3683, A3685)
Authenticated Symmetric Encryption and Decryption	BC-Auth	Key Size/ Key Strength: 128, 192, 256-bits	AES [FIPS 197; SP 800-38C]:CCM AES [FIPS 197; SP 800-38D]:GCM	AES-CCM: (A3685) AES-GCM: (A3685)
Random Number Generation	DRBG	Key Length/ Key Strength: 128, 256	CTR_DRBG [SP800-90ARev1]:AES-128, AES-256 Derivation Function Enabled No Prediction Resistance	Counter DRBG: (A3683, A3685)
ECDSA Asymmetric Key Generation	AsymKeyPair-KeyGen CKG	Curve: P-224, P-256, P-384, P-521. Key Strength: from 112 to 256	key generation method:Testing Candidates Supported Curves:P-224, P-256, P-384, P-521	ECDSA KeyGen (FIPS186-4): (A3686) CKG: () Key Type: Asymmetric
ECDSA Public-Key Validation	AsymKeyPair-PubKeyVal	Curve: P-224, P-256, P-384, P-521. Key	ECDSA [FIPS 186-5]:Public-Key Validation (PKV)	ECDSA KeyVer (FIPS186-4): (A3686)

Name	Type	Description	Properties	Algorithms
		Strength: from 112 to 256		
ECDSA Digital Signature Generation	DigSig-SigGen	Curve: P-224, P-256, P-384, P-521. Key Strength: from 112 to 256	ECDSA [FIPS 186-5]:Signature Generation	ECDSA SigGen (FIPS186-4): (A3686)
ECDSA Digital Signature Verification	DigSig-SigVer	Curve: P-224, P-256, P-384, P-521. Key Strength: from 112 to 256	ECDSA [FIPS 186-5]:Signature Verification	ECDSA SigVer (FIPS186-4): (A3686)
HMAC Message Authentication	MAC	Key Length 8 - 262144 bits/ Key Strength: 112 to 256 bits	HMAC [FIPS 198] (vng_ltc):SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 HMAC [FIPS 198] (c_ltc):SHA-384, SHA-512, SHA-512/256 HMAC [FIPS 198] (vng_neon):SHA-256	HMAC-SHA2-384: (A3684, A3686) HMAC-SHA2-512: (A3684, A3686) HMAC-SHA2-512/256: (A3684, A3686) HMAC-SHA2-256: (A3686, A3687) HMAC-SHA-1: (A3686) HMAC-SHA2-224: (A3686)
key wrapping / key unwrapping	KTS-Wrap	Key Size/ Key Strength: 128, 192, 256-bits	KTS (AES) [SP 800-38F]:AES-KW	AES-KW: (A3683)
RSA Digital Signature Generation	DigSig-SigGen	Modulus: 2048, 3072, 4096. Key Strength: from 112 to 150	RSA [FIPS 186-5]:Signature Generation (PKCS#1 v1.5) and (PKCS PSS)	RSA SigGen (FIPS186-4): (A3686)
RSA Digital Signature Verification	DigSig-SigVer	Modulus: 1024 (legacy use per FIPS 140-3 IG C.K), 2048, 3072, 4096. Key	RSA [FIPS 186-5]:Signature Verification PKCS#1 v1.5) and (PKCS PSS)	RSA SigVer (FIPS186-4): (A3686)

Name	Type	Description	Properties	Algorithms
		Strength: from 80 to 150		
Message Digest	SHA	N/A	SHS [FIPS 180-4] (vng_ltc):SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 SHS [FIPS 180-4] (c_ltc):SHA-384, SHA-512, SHA-512/256 SHS [FIPS 180-4] (vng_neon):SHA-256	SHA2-384: (A3684, A3686) SHA2-512: (A3684, A3686) SHA2-512/256: (A3684, A3686) SHA2-224: (A3686) SHA2-256: (A3686, A3687) SHA-1: (A3686)

Table 37: Security Function Implementations

2.7 Algorithm Specific Information

AES-GCM

AES-GCM IV is constructed in compliance with IG C.H scenario 1.

The GCM IV generation follows RFC 4106 and shall only be used for the IPsec protocol version 3. When the IV in RFC 4106 exhausts the maximum number of possible values for a given security association, either party to the security association that encounters this condition triggers a rekeying with IKEv2 to establish a new encryption key for the security association. The module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED from which the AES-GCM encryption keys are derived.

In case the module's power is lost and then restored, the key used for the AES GCM encryption/decryption shall be re-distributed. This condition is not enforced by the module.

AES-XTS

AES-XTS mode is only approved for hardware storage applications. The length of the AES-XTS data unit does not exceed 2^{20} blocks. The module checks explicitly that Key_1 ≠ Key_2 before using the keys in the XTS-Algorithm to process data with them compliant with IG C.I.

2.8 RBG and Entropy

Cert Number	Vendor Name
E14	apple
E15	apple

Table 38: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Apple corecrypto physical entropy source	Physical	See Tested Operational Environment Table	256 bit	256 bit	SHA-256 [ACVP cert. # C1223]
Apple corecrypto non- physical entropy source	Non-Physical	See Tested Operational Environment Table	256 bit	256 bit	SHA-256 [ACVP Certs. # A3687]

Table 39: Entropy Sources

Entropy sources: Two entropy sources (one non-physical entropy source and one physical entropy source) residing within the TOEPP provide the random bits. The entropy sources are located within the physical perimeter of the module (TOEPP) but outside the cryptographic boundary of the module.

RBGs: The NIST [SP 800-90ARev1] approved deterministic random bit generators (DRBG) used for random number generation is a CTR_DRBG using AES-256 with derivation function and without prediction resistance.

The module performs DRBG health tests according to [SP800-90ARev1 section 11.3].

The deterministic random bit generators are seeded by "*read_random*". The *read_random* is the Kernel Space interface.

RBG Output: The output of entropy sources provides 256-bits of entropy to seed and reseed SP800-90ARev1 DRBG during initialization (seed) and reseeding (reseed).

2.9 Key Generation

See vendor affirmed algorithms (CKG) in section 2.5.

The module does not implement symmetric key generation.

2.10 Key Establishment

2.11 Industry Protocols

No parts of the IPSec, 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 Data Output	Data inputs/outputs are provided in the variables passed in the C language Kernel Interfaces (KPIs) and callable service invocations, generally through caller-supplied buffers
N/A	Control Input	Control inputs which control the mode of the module are provided through dedicated parameters.
N/A	Status Output	Status output is provided in return codes and through messages. Documentation for each KPI lists possible return codes. A complete list of all return codes returned by the C language KPIs within the module is provided in the header files and the KPI documentation. Messages are also documented in the KPI documentation.

Table 40: Ports and Interfaces

The module does not implement a Control Output Logical Interface

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

FIPS 140-3 does not require an authentication mechanism for level 1 modules. Therefore, the module does not support an authentication mechanism for Crypto Officer. The Crypto Officer role is authorized to access all services provided by the module (see Table - Approved Services and Table - Non-Approved Services).

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 41: Roles

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
AES Encryption/Decryption	Execute AES-mode encrypt or decrypt operation	1	plaintext data and key / ciphertext data and key	ciphertext data / plaintext data	Unauthenticated Symmetric Encryption and Decryption Authenticated Symmetric Encryption and Decryption	Crypto Officer - AES key: W,E
AES Key Wrapping / Key Unwrapping	Execute AES-key wrapping or unwrapping operation	1	key wrapping key, unwrapped key / Wrapped key, AES key wrapping key	wrapped key / unwrapped key	key wrapping / key unwrapping	Crypto Officer - AES key-wrapping key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Secure Hash Generation	Generate a digest for the requested algorithm	1	message	digest	Message Digest	Crypto Officer
Message Authentication Generation	Generate a MAC digest using the requested SHA algorithm	1	message, MAC key, MAC algorithm	MAC	HMAC Message Authentication	Crypto Officer - HMAC key: W,E
Message Authentication Code Verification	Verify a MAC digest	1	MAC, message, MAC key, MAC algorithm	pass/fail	HMAC Message Authentication	Crypto Officer - HMAC key: W,E
RSA signature generation and verification	Sign a message with a specified RSA private key. Verify the signature of a message with a specified RSA public key.	1	SigGen: private key, message, hash function; SigVer: public key, digital signature, message, hash function	SigGen: computed signature; SigVer: pass/fail result of digital signature verification	RSA Digital Signature Generation RSA Digital Signature Verification	Crypto Officer - RSA key pair: W,E
ECDSA signature generation and verification	Sign a message with a specified ECDSA private key Verify the	1	SigGen: private key, message, hash function; SigVer: public	SigGen: computed signature; SigVer: pass/fail result of digital	ECDSA Digital Signature Generation ECDSA Digital Signature Verification	Crypto Officer - ECDSA key pair: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	signature of a message with a specified ECDSA public key		key, digital signature, message, hash function	signature verification		
Random Number Generation	Generate random number	1	length of generated number	random bit-string	Random Number Generation	Crypto Officer - Entropy input string: E - DRBG seed, internal state V value, and key: G,R,E
ECDSA key pair generation and validation	Generate a keypair for a requested elliptic curve and validity	1	domain parameters	key pair	ECDSA Asymmetric Key Generation ECDSA Public-Key Validation	Crypto Officer - ECDSA key pair: G,R,E
Self-test	execute CASTs	1	power	pass/fail results	Unauthenticated Symmetric Encryption and Decryption Authenticated Symmetric Encryption and Decryption Random Number Generation	Crypto Officer - HMAC key: E - AES key: E - AES key-wrapping key: E - ECDSA key

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					ECDSA Asymmetric Key Generation ECDSA Public-Key Validation ECDSA Digital Signature Generation ECDSA Digital Signature Verification HMAC Message Authentication key wrapping / key unwrapping RSA Digital Signature Generation RSA Digital Signature Verification Message Digest	pair: E - RSA key pair: E - DRBG seed, internal state V value, and key: E
Show Status	Return the module status	N/A	N/A	Status output	None	Crypto Officer
Show module version info	Return Module Base Name and Module Version Number	N/A	N/A	Module information	None	Crypto Officer
Zeroization	SSPs are zeroised when the	1	N/A	N/A	None	Crypto Officer - AES

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	system is powered down, when all resources of symmetric crypto function context, all resources of hash context, all resources of asymmetric crypto function context are released.					key: Z - AES key-wrapping key: Z - HMAC key: Z - ECDSA key pair: Z - RSA key pair: Z - Entropy input string: Z - DRBG seed, internal state V value, and key: Z

Table 42: Approved Services

The abbreviations of the access rights to SSPs have the following interpretation:
G = **Generate**: The module generates or derives the 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 = **Zeroise**: The module zeroises the SSP.
N/A = The service does not access any SSP during its operation

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Triple-DES encryption / decryption	Execute Triple-DES mode encrypt or decrypt operation.	Triple-DES [SP 800-67r2]	CO

Name	Description	Algorithms	Role
RSA Key Encapsulation	The CAST does not perform the full KTS, only the raw RSA encrypt/decrypt.	RSA Key Wrapping	CO
RSA Signature Generation	Sign a message with a non-approved RSA private key size	RSA SigGen	CO
RSA Signature Verification	Verify the signature of a message with a non-approved RSA public key size	RSA SigVer	CO
ECDSA key-pair generation, ECDSA PKV, ECDSA signature generation, ECDSA signature verification	For curve P-192	ECDSA	CO
ECDSA Key Pair Generation for compact point representation of points	For compact point representation of points	ECDSA KeyGen	CO
EdDSA Key Generation, Signature Generation, Signature Verification	Ed25519	EdDSA	CO
ECIES	Elliptic Curve encrypt/ decrypt	Integrated Encryption Scheme on elliptic curves (ECIES)	CO
ANSI X9.63 Key Derivation	SHA-1 hash-based	ANSI X9.63 KDF	CO
SP800-56Crev2 Key Derivation (HKDF)	SHA-256 hash-based	HKDF [SP800-56Crev2]	CO
OMAC Message Authentication Code Generation	One-Key CBC-MAC using 128-bit key	OMAC (One-Key CBC MAC)	CO
OMAC Message Authentication Code Verification	One-Key CBC-MAC using 128-bit key	OMAC (One-Key CBC MAC)	CO
Message digest generation	Message digest generation using non-approved algorithms	MD2 MD4 RIPEMD MD5	CO
Symmetric encryption / decryption	Symmetric encryption / decryption using non-approved algorithms	Blowfish CAST5 DES RC2 RC4	CO
RFC 6637 KDF	SHA-256, SHA-512, AES-128, AES-256	RFC 6637 Key Derivation	CO

Table 43: Non-Approved Services

4.5 External Software/Firmware Loaded

N/A

5 Software/Firmware Security

5.1 Integrity Techniques

A software integrity test is performed on the runtime image of the module. The HMAC-SHA256 implemented in the module is used as the approved algorithm for the integrity test. If the test fails, the module enters an error state where no cryptographic services are provided, and data output is prohibited i.e. the module is not operational.

5.2 Initiate on Demand

The module's integrity test can be performed on demand by power-cycling the computing platform. Integrity test on demand is performed as part of the Pre-Operational Self-Tests, automatically executed at power-on.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

6.2 Configuration Settings and Restrictions

The module is supplied as part of Device OS, a commercially available general-purpose operating system executing on the computing platforms specified in section 2.2.

7 Physical Security

The FIPS 140-3 physical security requirements do not apply to the Apple corecrypto Module v13.0 [Apple silicon, Kernel, Software, SL1] since it is a software module.

8 Non-Invasive Security

8.1 Mitigation Techniques

Per IG 12.A, until the requirements of NIST SP 800-140F are defined, non-invasive mechanisms fall under ISO/IEC 19790:2012 Section 7.12 Mitigation of other attacks.
The requirements of this area are not applicable to the module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	The module stores ephemeral SSPs in RAM provided by the operational environment. They are received for use or generated by the module only at the command of the calling application. The operating system protects all SSPs through the memory separation and protection mechanisms. No process other than the module itself can access the SSPs in its process' memory.	Dynamic

Table 44: Storage Areas

9.2 SSP Input-Output Methods

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

Table 45: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Context object destruction	SSPs are zeroised when the appropriate context object is destroyed	Zeroization when structure is deallocated	By calling the zeroization function cc_clear
Power down	SSPs are zeroised when the system is powered down	SSPs are zeroised when the system is powered down	Operator can initiate power down

Zeroization Method	Description	Rationale	Operator Initiation
Intermediate value zeroization	Intermediate keygen values are zeroized before the module returns from the key generation function.	Intermediate keygen values are zeroized before the module returns from the key generation function.	N/A

Table 46: SSP Zeroization Methods

Data output interfaces are inhibited while zeroisation is performed.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES key	AES key	128 to 256 bits - 128 to 256 bits	Symmetric - CSP			Unauthenticated Symmetric Encryption and Decryption Authenticated Symmetric Encryption and Decryption
AES key-wrapping key	AES KW	128 to 256 bits - 128 to 256 bits	symmetric - CSP			key wrapping / key unwrapping
HMAC key	HMAC key	128 to 256 - 128 to 256	MAC - CSP			HMAC Message Authentication
ECDSA key pair	ECDSA key pair (including intermediate keygen values)	P-224, P-256, P-384, P-521 - 112 to 256 bits	Asymmetric - CSP	ECDSA Asymmetric Key Generation		ECDSA Public-Key Validation ECDSA Digital Signature Generation ECDSA Digital Signature Verification
RSA key pair	RSA key pair	2048 - 4096 - 112 to 150 bits	Asymmetric - CSP			RSA Digital Signature Generation RSA Digital

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						Signature Verification
Entropy input string	Entropy input string	256 bits - 256 bits	Entropy input string - CSP			Random Number Generation
DRBG seed, internal state V value, and key	DRBG input parameters	256 bits - 256 bits	DRBG - CSP	Random Number Generation		Random Number Generation

Table 47: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES key	KPI input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
AES key-wrapping key	KPI input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
HMAC key	KPI input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
ECDSA key pair	KPI input parameters KPI output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down Intermediate value zeroization	DRBG seed, internal state V value, and key:Used With
RSA key pair	KPI input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down Intermediate	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				value zeroization	
Entropy input string		RAM:Plaintext	Storage duration during the usage of the CSP	Power down	DRBG seed, internal state V value, and key:Used With
DRBG seed, internal state V value, and key			Storage duration during the usage of the CSP	Power down	Entropy input string:Used With

Table 48: SSP Table 2

10 Self-Tests

While the module is executing the self-tests, services are not available, and input and output are inhibited.

10.1 Pre-Operational Self-Tests

The module performs a pre-operational software integrity automatically when the module is loaded into memory (i.e., at power on) before the module transitions to the operational state. A software integrity test is performed on the runtime image of the module with HMAC-SHA256 used to perform the approved integrity technique. Prior to using HMAC-SHA-256, a Conditional Cryptographic Algorithm Self-Tests (CAST) is performed.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A3687)	112-bit key	Message Authentication	SW/FW Integrity	Module successful execution	The HMAC-SHA2-256 value calculated at runtime is compared with the HMAC-SHA2-256 value stored in the module, computed at compilation time.

Table 49: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A3685)	128-bit key	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at Power-on before the integrity test
Counter DRBG (A3685)	AES 128-bit key	KAT	CAST	Module becomes operational	Health test per SP800- 90ARev1 section 11.3	Test runs at Power-on before the integrity test
HMAC-SHA2-256 (A3686)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at Power-on before the integrity test
HMAC-SHA-1 (A3686)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at Power-on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						before the integrity test
HMAC-SHA2-512 (A3684)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at Power-on before the integrity test
RSA SigGen (FIPS186-4) (A3686)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature Generation service request	Test runs at Power-on before the integrity test
RSA SigVer (FIPS186-4) (A3686)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature Verification service request	Test runs at Power-on before the integrity test
ECDSA KeyGen (FIPS186-4) (A3686)	SHA2-256 and respective keys	PCT	PCT	Successful key generation	Key generation	Key pair generation.
ECDSA SigGen (FIPS186-4) (A3686)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature Generation or Key Generation service request	Test runs at Power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A3686)	P-224 with SHA-224	KAT	CAST	Module becomes operational	Signature Verification or Key Generation service request	Test runs at Power-on before the integrity test
AES-CBC (A3682)	128-bit key	KAT	CAST	Module becomes operational	Encryption and decryption run separately	Test runs at Power-on before the integrity test
AES-ECB (A3682)	128-bit key	KAT	CAST	Module becomes operational	Encryption and decryption run separately	Test runs at Power-on before the integrity test
AES-XTS Testing Revision 2.0 (A3682)	128-bit key	KAT	CAST	Module becomes operational	Encryption	Test runs at Power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CCM (A3685)	128-bit key	KAT	CAST	Module becomes operational	Authenticated encryption / decryption operations are performed separately	Test runs at Power-on before the integrity test
HMAC-SHA2-512/256 (A3686)	SHA2-512/256	KAT	CAST	Module becomes operational	Message authentication	Test runs at Power-on before the integrity test

Table 50: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A3687)	Message Authentication	SW/FW Integrity	Whenever module is powered on	Upon every power on

Table 51: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A3685)	KAT	CAST	On Demand	Manually
Counter DRBG (A3685)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3686)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3686)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3684)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3686)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3686)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA KeyGen (FIPS186-4) (A3686)	PCT	PCT	Upon generation of an ECDSA key pair	Upon generation of an ECDSA key pair
ECDSA SigGen (FIPS186-4) (A3686)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3686)	KAT	CAST	On Demand	Manually
AES-CBC (A3682)	KAT	CAST	On Demand	Manually
AES-ECB (A3682)	KAT	CAST	On Demand	Manually
AES-XTS Testing Revision 2.0 (A3682)	KAT	CAST	On Demand	Manually
AES-CCM (A3685)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512/256 (A3686)	KAT	CAST	On Demand	Manually

Table 52: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error State	1) The HMAC-SHA-256 value computed over the module did not match the pre-computed value or 2) The computed value in the invoked Conditional CAST did not	1) Pre-operational Software Integrity Test failure or 2) Conditional CAST failure 3) Conditional PCT failure	The only method to recover from the error state is to power cycle the device which results in the module being reloaded into memory and reperforming the pre-	1) Print statement "FAILED: fipspost_post_integrity" to stdout or 2) Print statement "FAILED:<event>" to stdout (<event> refers to any of the CASTs listed in Conditional Self-Tests Table. 3) Error code "CCEC_GENERATE_KEY_CONSISTENCY" returned for ECDSA error code

Name	Description	Conditions	Recovery Method	Indicator
	match the known value or 3) The signature failed to verify successfully in the Conditional PCT. No cryptographic services are provided, and data output is prohibited		operational software integrity test and the Conditional CASTs.	

Table 53: Error States

10.5 Operator Initiation of Self-Tests

The module permits operators to initiate the pre-operational or conditional self-tests on demand for periodic testing of the module by rebooting the system (i.e., power-cycling).

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Startup Procedures: The module is built into Device OS defined in section 2 and delivered/ installed with the respective Device OS. There is no standalone delivery of the module as a software library.

Installation Process and Authentication Mechanisms: The vendor's internal development process guarantees that the correct version of module goes with its intended Device OS version. For additional assurance, the module is digitally signed by vendor, and it is verified during the integration into Host Device OS.

This digital signature-based integrity protection during the delivery/integration process is not to be confused with the HMAC-256 based integrity check performed by the module itself as part of its pre-operational self- tests.

11.2 Administrator Guidance

The Approved mode of operation is configured in the system by default and can only be transitioned into the non-Approved mode by calling one of the non-Approved services listed in Table - Non-Approved Services. If the device starts up successfully, then the module has passed all self-tests and is operating in the Approved mode.

The ESV Public Use Document (PUD) reference for physical entropy source is:
https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E14_PublicUse.pdf
The ESV Public Use Document (PUD) reference for non-physical entropy source is:
https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E15_PublicUse.pdf

Apple Platform Certifications guide [platform certifications] and Apple Platform Security guide [SEC] are provided by Apple which offers IT System Administrators with the necessary technical information to ensure FIPS 140-3 Compliance of the deployed systems. This guide walks the reader through the system's assertion of cryptographic module integrity and the steps necessary if module integrity requires remediation.

11.3 Non-Administrator Guidance

None

11.4 Design and Rules

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

- AES-GCM see section 2.7.
- AES-XTS see section 2.7.

11.6 End of Life

The module secure sanitization is accomplished through the Lost Mode, remote wipe, and remote lock sections of the provided vendor document [platform certifications].

12 Mitigation of Other Attacks

The module does not claim mitigation of other attacks.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
CAVP	Cryptographic Algorithm Validation Program
CAST	Cryptographic Algorithm Self-Test
CAST5	A symmetric-key 64-bit block cipher with 128-bit key
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ESVP	Entropy Source Validation Program
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
KAT	Known Answer Test
KDF	Key Derivation Function
KEXT	Kernel Extension
KW	AES Key Wrap
MAC	Message Authentication Code
KPI	Kernel Programming Interface
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PKG	Key-Pair Generation
PKV	Public Key Validation
PSS	Probabilistic Signature Scheme
PUD	Public Use Document (ESVP)
RSA	Rivest, Shamir, Addleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
TOEPP	Tested Operational Environment Physical Perimeter
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

Appendix B. References

FIPS140-3 **FIPS PUB 140-3 - Security Requirements for Cryptographic Modules**
March 2019
https://doi.org/10.6028/NIST.FIPS.140-3

SP 800-140x **CMVP FIPS 140-3 Related Reference**
https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-standards

FIPS140-3_IG **Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program**
September 2020
https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-ig-announcements

FIPS140-3_MM **CMVP FIPS 140-3 Draft Management Manual**
https://csrc.nist.gov/csrc/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/FIPS-140-3-CMVP%20Management%20Manual%20v2.0.pdf

SP 800-140 **FIPS 140-3 Derived Test Requirements (DTR)**
https://csrc.nist.gov/publications/detail/sp/800-140/final

SP 800-140A **CMVP Documentation Requirements**
https://csrc.nist.gov/publications/detail/sp/800-140a/final

SP 800-140Br1 **CMVP Security Policy Requirements**
https://doi.org/10.6028/NIST.SP.800-140Br1

SP 800-140C **CMVP Approved Security Functions**
https://csrc.nist.gov/publications/detail/sp/800-140c/final

SP 800-140D **CMVP Approved Sensitive Security Parameter Generation and Establishment Methods**
https://csrc.nist.gov/publications/detail/sp/800-140d/final

SP 800-140E **CMVP Approved Authentication Mechanisms** https://csrc.nist.gov/publications/detail/sp/800-140e/final

SP 800-140F **CMVP Approved Non-Invasive Attack Mitigation Test Metrics**
https://csrc.nist.gov/publications/detail/sp/800-140f/final

FIPS180-4 **Secure Hash Standard (SHS)**
March 2012
http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

FIPS186-5 **Digital Signature Standard (DSS)**
F3b 2023
https://doi.org/10.6028/NIST.FIPS.186-5

FIPS197 **Advanced Encryption Standard**
November 2001
http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

FIPS198-1 **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
http://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

PKCS#1
Public Key Cryptography Standards (PKCS) #1: RSA Cryptography
Specifications Version 2.1
February 2003
http://www.ietf.org/rfc/rfc3447.txt

RFC3394
Advanced Encryption Standard (AES) Key Wrap Algorithm
September 2002
http://www.ietf.org/rfc/rfc3394.txt

RFC5649
Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm
September 2009
http://www.ietf.org/rfc/rfc5649.txt

SP800-38A
NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques
December 2001
http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

SP800-38C
NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality
May 2004
http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf

SP800-38D
NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC
November 2007
http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

SP800-38E
NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices
January 2010
http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf

SP800-38F
NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping
December 2012
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf

SP800-56Cr2
Recommendation for Key-Derivation Methods in Key-Establishment Schemes
August 2020
https://doi.org/10.6028/NIST.SP.800-56Cr2

SP800-57
NIST Special Publication 800-57 Part 1 Revision 5 - Recommendation for Key Management Part 1: General
May 2020
https://doi.org/10.6028/NIST.SP.800-57pt1r5

SP800-67r2
NIST Special Publication 800-67 Revision 1 - Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher
January 2012 (withdrawn January 2014)
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-67r2.pdf

SP800-90Ar1
NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators
June 2015
http://dx.doi.org/10.6028/NIST.SP.800-90Ar1

SP800-90B
NIST Special Publication 800-90B - Recommendation for the Entropy Sources Used for Random Bit Generation
January 2018
https://doi.org/10.6028/NIST.SP.800-90B

SP800-108r1
NIST Special Publication 800-108r1 - Recommendation for Key Derivation Using Pseudorandom Functions
Aug 2022
https://doi.org/10.6028/NIST.SP.800-108r1

SP800-131Ar2
Transitioning the Use of Cryptographic Algorithms and Key Lengths
March 2019
https://doi.org/10.6028/NIST.SP.800-131Ar2

SP800-133r2
Recommendation for Cryptographic Key Generation
June 2020
https://doi.org/10.6028/NIST.SP.800-133r2

SP800-135r1
NIST Special Publication 800-135 Revision 1 - Recommendation for Existing Application-Specific Key Derivation Functions
December 2011
http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-135r1.pdf

SEC
Apple Platform Security
https://support.apple.com/guide/security/welcome/web
https://manuals.info.apple.com/MANUALS/1000/MA1902/en_US/apple-platform-security-guide.pdf

platform certifications
Apple Platform Certifications
https://support.apple.com/guide/certifications/welcome/web