Apple Inc.

Apple corecrypto Module v14.1 [Apple silicon, User, 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

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 Apple corecrypto Module v14.1 [Apple silicon, User, 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.

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 v14.1 [Apple silicon, User, Software, SL1] cryptographic module (hereafter referred to as "the module") provides implementations of low-level cryptographic primitives to the Device OS's (visionOS) 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 user space and does not contain any terminating assertions or exceptions. It is implemented as a Device OS dynamically loadable library. After the library is loaded, its cryptographic functions are made available to the Device OS application.

Any internal error detected by the module is returned to the caller with an appropriate return code. The calling Device OS application 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
Module Characteristics:
Cryptographic Boundary:
The module cryptographic boundary is delineated by the dotted green rectangle in the Figure 1. The module executes within the user space of the computing platforms and operating systems listed in the Tested Operational Environments Table section 2.2.

Tested Operational Environment's Physical Perimeter (TOEPP):
The physical perimeter is represented by the most exterior black line in the block diagram Figure 1.

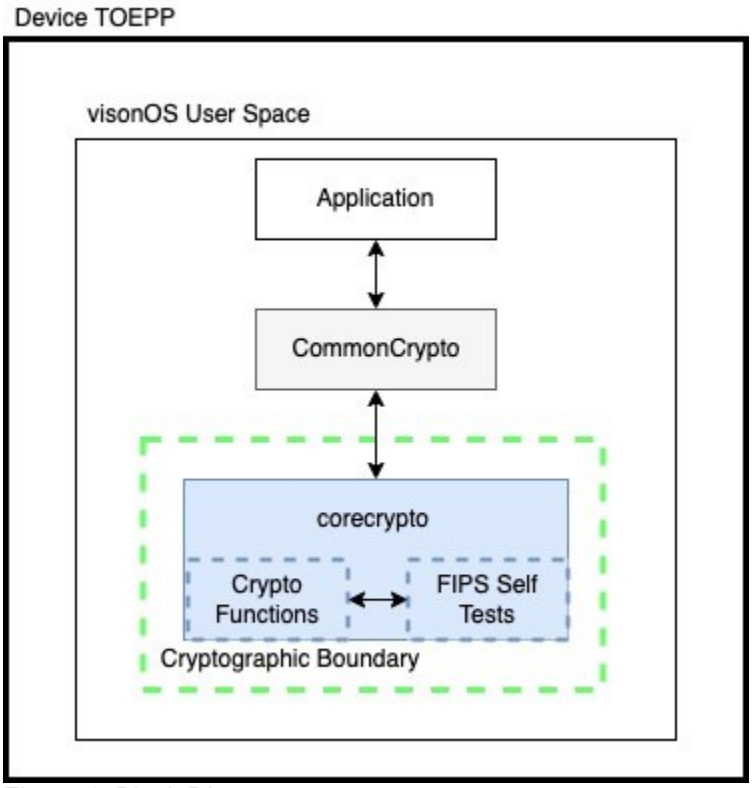

Figure 1: Block Diagram

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
corecrypto-1638.100.62	14.1	N/A	HMAC-SHA256

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)
visionOS 1	Apple Vision Pro	Apple M Series (ARMv8.6-A) M2	Yes	NA	14.1
visionOS 1	Apple Vision Pro	Apple M Series (ARMv8.6-A) M2	No	NA	14.1

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

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 '0' 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 any non-zero value 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 4: Modes List and Description

2.4.1 Mode Change Instructions and Status

The Module has an Approved and non-Approved mode of operation. The Approved mode of Operation is assumed automatically without any specific configuration. If the device starts up successfully then the module has passed all self-tests and is operating in the Approved mode. Any calls to the non-Approved security functions listed in the Non-Approved Services Table will cause the module to assume the non-Approved mode of operation.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A5404	-	SP 800-38A
AES-CBC	A5405	-	SP 800-38A
AES-CBC	A5406	-	SP 800-38A
AES-CBC	A5407	-	SP 800-38A
AES-CCM	A5405	-	SP 800-38C
AES-CCM	A5407	-	SP 800-38C
AES-CCM	A5408	-	SP 800-38C
AES-CFB128	A5404	-	SP 800-38A
AES-CFB128	A5405	-	SP 800-38A

Algorithm	CAVP Cert	Properties	Reference
AES-CFB128	A5407	-	SP 800-38A
AES-CFB8	A5405	-	SP 800-38A
AES-CFB8	A5407	-	SP 800-38A
AES-CMAC	A5407	-	SP 800-38B
AES-CTR	A5405	-	SP 800-38A
AES-CTR	A5407	-	SP 800-38A
AES-CTR	A5408	-	SP 800-38A
AES-ECB	A5404	-	SP 800-38A
AES-ECB	A5405	-	SP 800-38A
AES-ECB	A5407	-	SP 800-38A
AES-ECB	A5408	-	SP 800-38A
AES-GCM	A5405	-	SP 800-38D
AES-GCM	A5407	-	SP 800-38D
AES-GCM	A5408	-	SP 800-38D
AES-KW	A5405	-	SP 800-38F
AES-KW	A5407	-	SP 800-38F
AES-OFB	A5404	-	SP 800-38A
AES-OFB	A5405	-	SP 800-38A
AES-OFB	A5407	-	SP 800-38A
AES-XTS Testing Revision 2.0	A5404	-	SP 800-38E
AES-XTS Testing Revision 2.0	A5405	-	SP 800-38E
AES-XTS Testing Revision 2.0	A5407	-	SP 800-38E
Counter DRBG	A5405	-	SP 800-90A Rev. 1
Counter DRBG	A5407	-	SP 800-90A Rev. 1
Counter DRBG	A5408	-	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A5407	-	FIPS 186-4
ECDSA KeyGen (FIPS186-4)	A5409	-	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A5407	-	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A5409	-	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A5407	-	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A5409	-	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A5407	-	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A5409	-	FIPS 186-4
HMAC-SHA-1	A5407	-	FIPS 198-1
HMAC-SHA-1	A5409	-	FIPS 198-1
HMAC-SHA2-224	A5407	-	FIPS 198-1
HMAC-SHA2-224	A5409	-	FIPS 198-1
HMAC-SHA2-256	A5407	-	FIPS 198-1
HMAC-SHA2-256	A5409	-	FIPS 198-1
HMAC-SHA2-256	A5410	-	FIPS 198-1
HMAC-SHA2-384	A5407	-	FIPS 198-1
HMAC-SHA2-384	A5409	-	FIPS 198-1
HMAC-SHA2-512	A5407	-	FIPS 198-1
HMAC-SHA2-512	A5409	-	FIPS 198-1
HMAC-SHA2-512/256	A5407	-	FIPS 198-1
HMAC-SHA2-512/256	A5409	-	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA3-224	A5407	-	FIPS 198-1
HMAC-SHA3-224	A5409	-	FIPS 198-1
HMAC-SHA3-256	A5407	-	FIPS 198-1
HMAC-SHA3-256	A5409	-	FIPS 198-1
HMAC-SHA3-384	A5407	-	FIPS 198-1
HMAC-SHA3-384	A5409	-	FIPS 198-1
HMAC-SHA3-512	A5407	-	FIPS 198-1
HMAC-SHA3-512	A5409	-	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A5407	-	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A5407	-	SP 800-56A Rev. 3
KDA HKDF SP800-56Cr2	A5409	-	SP 800-56C Rev. 2
KDF SP800-108	A5407	-	SP 800-108 Rev. 1
KDF SP800-108	A5409	-	SP 800-108 Rev. 1
PBKDF	A5407	-	SP 800-132
PBKDF	A5409	-	SP 800-132
RSA KeyGen (FIPS186-4)	A5407	-	FIPS 186-4
RSA KeyGen (FIPS186-4)	A5409	-	FIPS 186-4
RSA SigGen (FIPS186-4)	A5407	-	FIPS 186-4
RSA SigGen (FIPS186-4)	A5409	-	FIPS 186-4
RSA SigVer (FIPS186-4)	A5407	-	FIPS 186-4
RSA SigVer (FIPS186-4)	A5409	-	FIPS 186-4
Safe Primes Key Generation	A5407	-	SP 800-56A Rev. 3
SHA-1	A5407	-	FIPS 180-4
SHA-1	A5409	-	FIPS 180-4
SHA2-224	A5407	-	FIPS 180-4
SHA2-224	A5409	-	FIPS 180-4
SHA2-256	A5407	-	FIPS 180-4
SHA2-256	A5409	-	FIPS 180-4
SHA2-256	A5410	-	FIPS 180-4
SHA2-384	A5407	-	FIPS 180-4
SHA2-384	A5409	-	FIPS 180-4
SHA2-512	A5407	-	FIPS 180-4
SHA2-512	A5409	-	FIPS 180-4
SHA2-512/256	A5407	-	FIPS 180-4
SHA2-512/256	A5409	-	FIPS 180-4
SHA3-224	A5407	-	FIPS 202
SHA3-224	A5409	-	FIPS 202
SHA3-256	A5407	-	FIPS 202
SHA3-256	A5409	-	FIPS 202
SHA3-384	A5407	-	FIPS 202
SHA3-384	A5409	-	FIPS 202
SHA3-512	A5407	-	FIPS 202
SHA3-512	A5409	-	FIPS 202

Table 5: Approved Algorithms

Vendor-Affirmed Algorithms:

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

Table 6: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:
N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
MD5	Allowed in Approved mode with no security claimed per IG 2.4.A Digest Size: 128-bit	Message Digest (used as part of the TLS key establishment scheme v1.0, v1.1 only)

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

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
Diffie-Hellman	Shared Secret Computation using key size < 2048
ECDSA	PKG: Curve P-192; PKV: Curve P-192; compact point representation of points; Signature Generation: Curve P-192; Signature Verification: Curve P-192
EC Diffie-Hellman	Shared Secret Computation using curves < P-224
Ed25519	Key Generation, Signature Generation, Signature Verification, X25519 Key agreement
Integrated Encryption Scheme on elliptic curves	Encryption / Decryption
MD2	Message Digest size: 128-bit
MD4	Message Digest size: 128-bit
MD5	Message Digest (except in the TLS 1.0/1.1 context)
OMAC (One-Key CBC MAC)	MAC generation
RC2	Encryption / Decryption Key Sizes 8 to 1024-bits
RC4	Encryption / Decryption Key Sizes 8 to 4096-bits
RFC6637	Key Derivation Function
RIPEMD	Message Digest size: 160-bits
RSA Keygen	ANSI X9.31 Key Pair Generation; keys < 2048-bits
RSA Digital Signature	PKCS#1 v1.5 and PSS; Signature Generation Key Size < 2048; Signature Verification Key Size < 1024
RSA Key Wrapping	OAEP, PKCS#1 v1.5 and -PSS schemes
Triple-DES [SP 800-67]	Encrypt/Decrypt; CBC, CTR, CFB64, ECB, CFB8, OFB

Name	Use and Function
HPKE (Hybrid Public Key Encryption) [RFC9180]	Hybrid encryption scheme
Keccak	Message Digest

Table 8: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Symmetric Encryption and Decryption	BC-UnAuth BC-Auth	Symmetric Encryption and Decryption	AES-CBC:Key Size / Key Strength: 128, 192, 256 bits AES-CFB128:Key Size / Key Strength: 128, 192, 256 bits AES-ECB:Key Size / Key Strength: 128, 192, 256 bits AES-OFB:Key Size / Key Strength: 128, 192, 256 bits AES-XTS Testing Revision 2.0:Key Size/ Key Strength: 128, 256 bits AES-CCM:Key Size / Key Strength: 128, 192, 256 bits AES-CFB8:Key Size / Key Strength: 128, 192, 256 bits AES-CTR:Key Size / Key Strength: 128, 192, 256 bits AES-GCM:Key Size / Key Strength: 128, 192, 256 bits	AES-CBC: (A5406, A5407, A5404, A5405) AES-CFB128: (A5407, A5404, A5405) AES-ECB: (A5407, A5408, A5404, A5405) AES-OFB: (A5407, A5404, A5405) AES-XTS Testing Revision 2.0: (A5407, A5404, A5405) AES-CCM: (A5407, A5408, A5405) AES-CFB8: (A5407, A5405) AES-CTR: (A5407, A5408, A5405) AES-GCM: (A5407, A5408, A5405)
Key Wrapping	KTS-Wrap	Key Wrapping	AES-KW:Key Size / Key	AES-KW: (A5407, A5405)

Name	Type	Description	Properties	Algorithms
			Strength: 128, 192, 256 bits	
Random Number Generation	DRBG	Random Number Generation	Counter DRBG:Key Size/ Key Strength: 128, 256 bits	Counter DRBG: (A5407, A5408, A5405)
Message authentication (MAC)	MAC	Message authentication (MAC)	AES-CMAC:Key Size / Key Strength: 128, 192, 256 bits HMAC-SHA-1:Key Size: 128 - 262144 bits; Key Strength: 128 bits HMAC-SHA2-224:Key Size: 224 - 262144 bits; Key Strength: 224 bits HMAC-SHA2-256:Key Size: 256 - 262144 bits; Key Strength: 256 bits HMAC-SHA2-384:Key Size: 384 - 262144 bits; Key Strength: 384 bits HMAC-SHA2-512:Key Size: 512 - 262144 bits; Key Strength: 512 bits HMAC-SHA2-512/256:Key Size: 512 - 262144 bits; Key Strength: 256 bits	AES-CMAC: (A5407) HMAC-SHA-1: (A5407, A5409) HMAC-SHA2-224: (A5407, A5409) HMAC-SHA2-256: (A5410, A5407, A5409) HMAC-SHA2-384: (A5407, A5409) HMAC-SHA2-512: (A5407, A5409) HMAC-SHA2-512/256: (A5407, A5409) HMAC-SHA3-224: (A5407, A5409) HMAC-SHA3-256: (A5407, A5409) HMAC-SHA3-384: (A5407, A5409) HMAC-SHA3-512: (A5407, A5409)
Asymmetric Key Generation	AsymKeyPair-KeyGen	Asymmetric Key Generation	ECDSA KeyGen (FIPS186-4):Key Size(Curve): P-	ECDSA KeyGen (FIPS186-4): (A5407, A5409)

Name	Type	Description	Properties	Algorithms
	CKG KAS-KeyGen		224, P-256, P-384, P-521; Key Strength: from 112 to 256 bits RSA KeyGen (FIPS186-4):Key Size: 2048, 3072, 4096 bits; Key Strength: from 112 to 150 bits Safe Primes Key Generation:Key Size: 2048, 3072, 4096, 6144, 8192 bits; Key Strength: from 112 to 200 bits	RSA KeyGen (FIPS186-4): (A5407, A5409) Counter DRBG: (A5407, A5408, A5405) Safe Primes Key Generation: (A5407) Asymmetric CKG: ()
Asymmetric Key Validation	AsymKeyPair-KeyVer	Asymmetric Key Validation	ECDSA KeyVer (FIPS186-4):Key Size(Curve): P-224, P-256, P-384, P-521; Key Strength: from 112 to 256 bits	ECDSA KeyVer (FIPS186-4): (A5407, A5409)
Digital Signature Generation	DigSig-SigGen	Digital Signature Generation	ECDSA SigGen (FIPS186-4):Key Size(Curve): P-224, P-256, P-384, P-521; Key Strength: from 112 to 256 bits RSA SigGen (FIPS186-4):Key Size: 2048, 3072, 4096 bits; Key Strength: from 112 to 150 bits	ECDSA SigGen (FIPS186-4): (A5407, A5409) RSA SigGen (FIPS186-4): (A5407, A5409) Counter DRBG: (A5407, A5408, A5405) SHA2-224: (A5407, A5409) SHA2-256: (A5410, A5407, A5409) SHA2-384: (A5407, A5409) SHA2-512: (A5407, A5409) SHA3-224: (A5407, A5409) SHA3-256: (A5407, A5409)

Name	Type	Description	Properties	Algorithms
				SHA3-384: (A5407, A5409) SHA3-512: (A5407, A5409)
Digital Signature Verification	DigSig-SigVer	Digital Signature Verification (usage of SHA1 is considered Legacy Use)	ECDSA SigVer (FIPS186-4):Key Size(Curve): P-224, P-256, P-384, P-521; Key Strength: from 112 to 256 bits RSA SigVer (FIPS186-4):Key Size: 1024, 2048, 3072, 4096 bits; Key Strength: from 80 to 150 bits	ECDSA SigVer (FIPS186-4): (A5407, A5409) RSA SigVer (FIPS186-4): (A5407, A5409) SHA-1: (A5407, A5409) SHA2-224: (A5407, A5409) SHA2-256: (A5410, A5407, A5409) SHA2-384: (A5407, A5409) SHA2-512: (A5407, A5409) SHA3-224: (A5407, A5409) SHA3-256: (A5407, A5409) SHA3-384: (A5407, A5409) SHA3-512: (A5407, A5409)
Shared Secret Computation	KAS-SSC	Shared Secret Computation	KAS-ECC-SSC Sp800-56Ar3:Key Size(Curve): P-224, P-256, P-384, P-521; Key Strength: from 112 to 256 bits KAS-FFC-SSC Sp800-56Ar3:Key Size: 2048, 3072, 4096, 6144, 8192 bits; Key Strength: from 112 to 200 bits	KAS-ECC-SSC Sp800-56Ar3: (A5407) KAS-FFC-SSC Sp800-56Ar3: (A5407)
Key Derivation HKDF	KAS-56CKDF	Key Derivation	KDA HKDF SP800-56Cr2:Shared	KDA HKDF SP800-56Cr2: (A5409)

Name	Type	Description	Properties	Algorithms
			Secret Length: 224-2048 Increment 8; Derived Key Length: 2048	HMAC-SHA-1: (A5409) HMAC-SHA2-224: (A5409) HMAC-SHA2-256: (A5409) HMAC-SHA2-384: (A5409) HMAC-SHA2-512: (A5409) HMAC-SHA3-224: (A5409) HMAC-SHA3-256: (A5409) HMAC-SHA3-384: (A5409) HMAC-SHA3-512: (A5409)
Key Derivation CMAC KBKDF	KBKDF	Key Derivation	KDF SP800-108rev1 with AES-CMAC:Key Size / Key Strength: 128, 192, 256 bits; Supported Lengths: 8-4096 Increment 8; Fixed Data Order: Before Fixed Data; Counter Length: 8, 16, 24, 32	KDF SP800-108: (A5407) AES-CMAC: (A5407)
Key Derivation HMAC KBKDF	KBKDF	Key Derivation	KDF SP800-108rev1 with HMAC:Key Size: 8-262144 Increment 8; Supported Lengths: 8-4096 Increment 8; Fixed Data Order: Before Fixed Data; Counter Length: 32	KDF SP800-108: (A5407, A5409) HMAC-SHA-1: (A5407, A5409) HMAC-SHA2-224: (A5407, A5409) HMAC-SHA2-256: (A5407, A5409) HMAC-SHA2-384: (A5407, A5409) HMAC-SHA2-512: (A5407,

Name	Type	Description	Properties	Algorithms
				A5409) HMAC-SHA3-224: (A5407, A5409) HMAC-SHA3-256: (A5407, A5409) HMAC-SHA3-384: (A5407, A5409) HMAC-SHA3-512: (A5407, A5409)
Key Derivation PBKDF	PBKDF	Key Derivation	PBKDF:Key Size: 128 - 262144; Key Strength: 128 - 256; Password length: 8- 128 bytes Increment 1; Salt Length: 128-4096 Increment 8; Iteration Count: 10-1000 Increment 1	PBKDF: (A5407, A5409) HMAC-SHA-1: (A5407, A5409) HMAC-SHA2-224: (A5407, A5409) HMAC-SHA2-256: (A5407, A5409) HMAC-SHA2-384: (A5407, A5409) HMAC-SHA2-512: (A5407, A5409) HMAC-SHA3-224: (A5407, A5409) HMAC-SHA3-256: (A5407, A5409) HMAC-SHA3-384: (A5407, A5409) HMAC-SHA3-512: (A5407, A5409)
Message Digest	SHA	Message Digest	SHA-1:N/A SHA2-224:N/A SHA2-256:N/A SHA2-384:N/A SHA2-512:N/A SHA2-512/256:N/A	SHA-1: (A5407, A5409) SHA2-224: (A5407, A5409) SHA2-256: (A5410, A5407, A5409)

Name	Type	Description	Properties	Algorithms
				SHA2-384: (A5407, A5409) SHA2-512: (A5407, A5409) SHA2-512/256: (A5407, A5409)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

GCM IV

AES-GCM IV is constructed in compliance with IG C.H scenario 1 (TLS 1.2) and scenario 2 (IPsec-v3).

The GCM IV generation follows RFC 5288 shall only be used for the TLS protocol version 1.2. This implementation is compatible with acceptable AES-GCM ciphersuites from SP800-52r2 Section 3.3.1. The counter portion of the IV is set by the module within its cryptographic boundary. The module does not implement the TLS protocol. The module's implementation of AES-GCM is used together with an application that runs outside the module's cryptographic boundary. The design of the TLS protocol implicitly ensures that the nonce_explicit, or counter portion of the IV will not exhaust all of its possible values.

The GCM IV generation follows RFC 4106 and shall only be used for the IPsec-v3 protocol version 3. The counter portion of the IV is set by the module within its cryptographic boundary. The module does not implement the IPsec protocol. The module's implementation of AES-GCM is used together with an application that runs outside the module's cryptographic boundary. The design of the IPsec protocol implicitly ensures that the nonce_explicit, or counter portion of the IV will not exhaust all of its possible values.

In compliance with IG C.H section 3, if the module's power is lost and then restored, the key used for the AES GCM encryption/ decryption shall be re-distributed.

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.

Key Derivation using SP 800-132 PBKDF2

The module implements a CAVP tested key derivation function compliant to SP800-132 and IG D.N. The service returns the key derived from the provided password to the caller. The length of the password used as input to PBKDFv2 shall be at least 8 characters and the worst-case probability of guessing the value is 10^8 assuming all characters are digits only. The user shall choose the password length and the iteration count in such a way that the combination will

make the key derivation computationally intensive. PBKDFv2 is implemented to support the option 1a specified in section 5.4 of SP800-132. The derived keys may only be used in storage applications.

KAS
The module does not establish SSPs using an approved key agreement scheme (KAS). However, it does offer some or all of the underlying KAS cryptographic functionality to be used by an external operator/application as part of an approved KAS.

SHA-1:
Digital signature generation using SHA-1 is non-approved and not allowed in approved services. Digital signature verification using SHA-1 is considered approved ("Legacy"). HMAC using SHA-1 are approved.

The SHA-1 algorithm, as implemented by the module, will be non-approved for all purposes except signature verification, starting January 1, 2031.

Note: Algorithms designated as "Legacy" can only be used on data that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M.

2.8 RBG and Entropy

Cert Number	Vendor Name
E113	apple
E181	apple

Table 10: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Apple corecrypto physical entropy source	Physical	See Tested Operational Environment Table in section 2.2	256 bit	256 bit	SHA-256 [ACVP cert. #C1223]
Apple corecrypto non-physical entropy source	Non-Physical	See Tested Operational Environment Table in section 2.2	18432 bits (576 32-bit samples)	512	SHA2-512 [ACVP cert #A5369]

Table 11: Entropy Sources

Entropy sources: The random bits used to seed and reseed the module's approved DRBG comes from a physical entropy source residing within the TOEPP. The entropy source includes a vetted conditioning component in the form of a SHA-256. The min-entropy rate at the output of the entropy source (h_out for the output of the conditioning component per Section 3.1.5 of SP 800-90B) is 256 bits per 256-bit output.

The entropy source follows IG 9.3.A scenario 1.(b) i.e., the module is a software module and the entropy sources reside outside of the cryptographic boundary but inside module's TOEPP.

DRBG(s): The module implements an SP 800-90ARev1 approved deterministic random bit generator (DRBG) in the form of 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.

DRBG Output: The output of CTR_DRBG provides up to 256-bits of security strength.

2.9 Key Generation

The module implements asymmetric key generation compliant to SP800-133r2 Section 4 examples 1 and is listed as a vendor affirmed algorithm per FIPS 140_3 IG D.H. The seed material used to generate the asymmetric key pairs is provided directly output from the module's CTR_DRBG.

2.10 Key Establishment

The module implements KAS-FFC-SSC and KAS-ECC-SSC compliant to [SP800-56Ar3] and is listed as an approved algorithm per FIPS 140_3 IG D.F scenario 2 path (1). The module only implements shared secret computation. All required assurances from Section 5.6.2 of SP 800-56Arev3 are met by the module.

2.11 Industry Protocols

No parts of the TLS or IPsec protocols, 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 inputs are provided in the variables passed in the API and callable service invocations, generally through caller-supplied buffers
N/A	Data Output	Data outputs are provided in the variables passed in the API 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 API lists possible return codes. A complete list of all return codes returned by the C language APIs within the module is provided in the header files and the API documentation. Messages are also documented in the API documentation.

Table 12: 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	Crypto Officer	None

Table 13: Roles

4.3 Approved Services

The module implements a dedicated API function to indicate if a requested service utilizes an approved security function. The approved service indicator utilizes one of two functions (fips_allowed and fips_allowed_mode) depending on the service in question. Calling fips_allowed_mode with any approved AES mode will return a zero to indicate it is an approved algorithm. Similarly, calling fips_allowed with any other approved algorithm will return zero. Calling either of these with an algorithm not listed in the Approved Algorithms Table will return a non-zero value, and as such indicates a non-approved service.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
AES Encryption/Decryption	Execute AES-mode encrypt or decrypt operation	0	plaintext data and key / ciphertext data and key	ciphertext data / plaintext data	Symmetric Encryption and Decryption	Crypto Officer - AES key: W,E
AES Key Wrapping / Key unwrapping	Execute AES-key wrapping or unwrapping operation	0	AES key wrapping key, key to be wrapped / wrapped key, AES key wrapping key	wrapped key / unwrapped key	Key Wrapping	Crypto Officer - AES key-wrapping key: W,E
Secure Hash Generation	Generate a digest for the requested algorithm	0	message	digest	Message Digest	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Message Authentication Generation	Generate a MAC digest using the requested SHA algorithm or AES algorithm	0	message, MAC key, MAC algorithm	MAC	Message authentication (MAC)	Crypto Officer - AES key: W,E - HMAC key: W,E
Message Authentication Verification	Verify a MAC digest	0	MAC, message, MAC key, MAC algorithm	pass/fail	Message authentication (MAC)	Crypto Officer - AES key: W,E - 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.	0	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	Digital Signature Generation 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 signature of a message with a specified ECDSA public key	0	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	Digital Signature Generation Digital Signature Verification	Crypto Officer - ECDSA key pair: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Random Number Generation	Generate random number	0	requested number of bits	random bit-string	Random Number Generation	Crypto Officer - Entropy input string: E - DRBG seed, internal state V value, and key (IG D.L compliant): G,W,E
PBKDF	Derive key from password	0	Password	PBKDF derived key	Key Derivation PBKDF	Crypto Officer - PBKDF derived key: G,R - PBKDF password: W,E
KBKDF	Derive key from key derivation key	0	KBKDF key derivation key	KBKDF derived key	Key Derivation CMAC KBKDF Key Derivation HMAC KBKDF	Crypto Officer - KBKDF key derivation key: W,E - KBKDF derived key: G,R
HKDF	Derive key from key derivation input keying material	0	HKDF input keying material	HKDF derived key	Key Derivation HKDF	Crypto Officer - HKDF input keying material: W,E - HKDF derived key: G,R,E
RSA key pair generation	Generate a keypair for a	0	key size	key pair	Asymmetric Key Generation	Crypto Officer - DRBG seed,

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	requested modulus					internal state V value, and key (IG D.L compliant): W,E - RSA key pair: G,R
ECDSA key pair generation	Generate a keypair for a requested elliptic curve	0	curve size	key pair	Asymmetric Key Generation Asymmetric Key Validation	Crypto Officer - DRBG seed, internal state V value, and key (IG D.L compliant): W,E - ECDSA key pair: G,R
Safe primes key generation	Generate a keypair for a requested 'safe' domain parameter	0	key size	key pair	Asymmetric Key Generation	Crypto Officer - DRBG seed, internal state V value, and key (IG D.L compliant): W,E - Diffie-Hellman key pair: G,R
Diffie-Hellman shared secret computation	Generate a shared secret	0	domain parameter, received public key and possesse	shared secret	Shared Secret Computation	Crypto Officer - Diffie-Hellman key pair: W,E - Diffie-Hellman

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			d private key			shared secret: G,R
EC Diffie-Hellman shared secret computation	Generate a shared secret	0	domain parameter, received public key and possessed private key	shared secret	Shared Secret Computation	Crypto Officer - EC Diffie Hellman key pair: W,E - EC Diffie-Hellman shared secret: G,R
Self-test	execute pre operational self-tests and all conditional CASTs from section 10.2	N/A	power	pass/fail results	Symmetric Encryption and Decryption Key Wrapping Random Number Generation Message authentication (MAC) Asymmetric Key Generation Asymmetric Key Validation Digital Signature Generation Digital Signature Verification Shared Secret Computation Key Derivation PBKDF	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					Key Derivation CMAC KBKDF Key Derivation HMAC KBKDF Key Derivation HKDF Message Digest	
Show Status	Return the module status	N/A	N/A	Status output	None	Crypto Officer
Show module and 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 system is powered down, when all resources of symmetric crypto function context, all resources of hash context, all resources of Diffie-Hellman context for Diffie-Hellman and EC Diffie-	0	length of context to zeroize and address of context to be zeroized	N/A	None	Crypto Officer - AES 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,

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	Hellman, all resources of asymmetric crypto function context and all resources of key derivation function context are released					and key (IG D.L compliant): Z - PBKDF derived key: Z - PBKDF password: Z - KBKDF key derivation key: Z - KBKDF derived key: Z - Diffie-Hellman key pair: Z - EC Diffie Hellman key pair: Z - Diffie-Hellman shared secret: Z - EC Diffie-Hellman shared secret: Z

Table 14: Approved Services

4.4 Non-Approved Services

Name	Description	Algorithms	Role
ANSI X9.63 KDF	Hash based Key Derivation Function	ANSI X9.63 KDF	CO
Blowfish	Encryption / Decryption	Blowfish	CO
CAST5	Encryption / Decryption Key Sizes: 40 to 128 bits in 8-bit increments	CAST5	CO
DES	Encryption / Decryption Key Size: 56-bits	DES	CO
Diffie-Hellman	Shared Secret Computation using key size < 2048	Diffie-Hellman	CO

Name	Description	Algorithms	Role
ECDSA	PKG: Curve P-192; PKV: Curve P-192; compact point representation of points; Signature Generation: Curve P-192; Signature Verification: Curve P-192	ECDSA	CO
EC Diffie-Hellman	Shared Secret Computation using curves < P-224	EC Diffie-Hellman	CO
Ed25519	Key Generation, Signature Generation, Signature Verification, Key agreement	Ed25519	CO
Integrated Encryption Scheme on elliptic curves	Encryption / Decryption	Integrated Encryption Scheme on elliptic curves	CO
MD2	Message Digest size: 128-bit	MD2	CO
MD4	Message Digest size: 128-bit	MD4	CO
MD5	Message Digest (except in the TLS 1.0/1.1 context)	MD5	CO
OMAC (One-Key CBC MAC)	MAC generation	OMAC (One-Key CBC MAC)	CO
RC2	Encryption / Decryption Key Sizes 8 to 1024-bits	RC2	CO
RC4	Encryption / Decryption Key Sizes 8 to 4096-bits	RC4	CO
RFC6637	Key Derivation Function	RFC6637	CO
RIPEMD	Message Digest size: 160-bits	RIPEMD	CO
RSA Keygen	ANSI X9.31 Key Pair Generation; keys < 2048-bits	RSA Keygen	CO
RSA Digital Signature	PKCS#1 v1.5 and PSS; Signature Generation Key Size < 2048; Signature Verification Key Size < 1024	RSA Digital Signature	CO
RSA Key Wrapping	OAEP, PKCS#1 v1.5 and -PSS schemes	RSA Key Wrapping	CO
Triple-DES [SP 800-67]	Encrypt/Decrypt; CBC, CTR, CFB64, ECB, CFB8, OFB	Triple-DES [SP 800-67]	CO
HPKE (Hybrid Public Key Encryption)	Hybrid encryption scheme	HPKE (Hybrid Public Key Encryption) [RFC9180]	CO
Keccak	Message Digest	Keccak	CO

Table 15: Non-Approved Services

4.5 External Software/Firmware Loaded

The module does not load any external software.

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 tests on demand is performed as part of the Pre-Operational Self-Tests. It is 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 v14.1 [Apple silicon, User, Software, SL1] since it is a software module.

8 Non-Invasive Security

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 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

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	Invocation of 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
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 18: 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			Symmetric Encryption and Decryption Message authentication (MAC)
AES key-wrapping key	AES KW	128 to 256 bits - 128 to 256 bits	symmetric - CSP			Key Wrapping
HMAC key	HMAC key	8 - 262144 bits - 112 to 256-bits	MAC - CSP			Message authentication (MAC)
ECDSA key pair	ECDSA key pair (including intermediate keygen values)	P-224, P-256, P-384, P-521 - 112 to 256 bits	Asymmetric - CSP	Asymmetric Key Generation		Digital Signature Generation Digital Signature Verification
RSA key pair	RSA key pair (including intermediate keygen values)	2048 - 4096 - 112 to 150 bits	Asymmetric - CSP	Asymmetric Key Generation		Digital Signature Generation Digital 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 (IG D.L compliant)	DRBG input parameters	256 bits - 256 bits	DRBG - CSP	Random Number Generation		Random Number Generation
PBKDF derived key	PBKDF derived key	128 to 256 bits - 128 to 256 bits	Storage key - CSP	Key Derivation PBKDF		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
PBKDF password	PBKDF password	64 to 1024 bits - N/A	Password - CSP			Key Derivation PBKDF
KBKDF key derivation key	KBKDF key derivation key	128 to 256 bits - 128 to 256 bits	Derivation key - CSP			Key Derivation CMAC KBKDF Key Derivation HMAC KBKDF
KBKDF derived key	KBKDF derived key	128 to 256 bits - 128 to 256 bits	Derived key - CSP	Key Derivation CMAC KBKDF Key Derivation HMAC KBKDF		
HKDF input keying material	HKDF key derivation keying material	128 to 256 bits - 128 to 256 bits	Derivation key - CSP			Key Derivation HKDF
HKDF derived key	HKDF derived key	128 to 256 bits - 128 to 256 bits	Derived key - CSP	Key Derivation HKDF		
Diffie-Hellman key pair	Diffie-Hellman key pair (including intermediate keygen values)	MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192 - 112 to 200 bits	Asymmetric - CSP	Asymmetric Key Generation		Shared Secret Computation
Diffie-Hellman shared secret	Diffie-Hellman shared secret	MODP-2048, MODP-3072, MODP-4096,	Asymmetric - CSP		Shared Secret Computation	

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		MODP-6144, MODP-8192 - 112 to 200 bits				
EC Diffie Hellman key pair	EC Diffie-Hellman key pair (including intermediate keygen values)	P-224, P-256, P-384, P-521 - 112-256 bits	Asymmetric - CSP	Asymmetric Key Generation		Shared Secret Computation
EC Diffie-Hellman shared secret	EC Diffie-Hellman shared secret	P-224, P-256, P-384, P-521 - 112-256 bits	Asymmetric - CSP		Shared Secret Computation	

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES key	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
AES key-wrapping key	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
HMAC key	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	
ECDSA key pair	API input parameters API 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 (IG D.L compliant):Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
RSA key pair	API input parameters API 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 (IG D.L compliant):Derived From
Entropy input string		RAM:Plaintext	Storage duration during the usage of the CSP	Power down	DRBG seed, internal state V value, and key (IG D.L compliant):Generates
DRBG seed, internal state V value, and key (IG D.L compliant)		RAM:Plaintext	Storage duration during the usage of the CSP	Power down	Entropy input string:Derived From
PBKDF derived key	API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	PBKDF password:Derived From
PBKDF password	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	PBKDF derived key:Derives
KBKDF key derivation key	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	KBKDF derived key:Derives
KBKDF derived key	API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	KBKDF key derivation key:Derived From
HKDF input keying material	API input parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	HKDF derived key:Derives

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
HKDF derived key	API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	HKDF input keying material:Derived From
Diffie-Hellman key pair	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down Intermediate value zeroization	Diffie-Hellman shared secret:Generates
Diffie-Hellman shared secret	API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	Diffie- Hellman key pair:Derived From
EC Diffie Hellman key pair	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down Intermediate value zeroization	EC Diffie-Hellman shared secret:Generates
EC Diffie-Hellman shared secret	API output parameters	RAM:Plaintext	From service invocation to service completion	Context object destruction Power down	EC Diffie Hellman key pair:Derived From

Table 20: 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 (A5407)	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 21: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A5407)	128-bit key, encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-GCM (A5408)	128-bit key, encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-GCM (A5405)	128-bit key, encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
Counter DRBG (A5407)	128-bit key	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Counter DRBG (A5408)	128-bit key	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Counter DRBG (A5405)	128-bit key	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						before the integrity test
HMAC-SHA2-256 (A5410)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A5407)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A5409)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A5407)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A5409)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A5407)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A5409)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512/256 (A5407)	SHA2-512/256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512/256 (A5409)	SHA2-512/256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A5407)	SHA3-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A5409)	SHA3-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA3-256 (A5407)	SHA3-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A5409)	SHA3-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A5407)	SHA3-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A5409)	SHA3-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A5407)	SHA3-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A5409)	SHA3-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
RSA KeyGen (FIPS186-4) (A5407)	PCT with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA KeyGen (FIPS186-4) (A5409)	PCT with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA SigGen (FIPS186-4) (A5407)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-4) (A5409)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-4) (A5407)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigVer (FIPS186-4) (A5409)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA KeyGen (FIPS186-4) (A5407)	PCT with SHA2-256	PCT	PCT	Successful key pair generation	ECDSA: Sign/Verify; ECDH: SP800-56Arev3 section 5.6.2.1.4	Key pair generation
ECDSA KeyGen (FIPS186-4) (A5409)	PCT with SHA2-256	PCT	PCT	Successful key pair generation	ECDSA: Sign/Verify; ECDH: SP800-56Arev3 section 5.6.2.1.4	Key pair generation
ECDSA SigGen (FIPS186-4) (A5407)	P-224 with SHA-224	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-4) (A5409)	P-224 with SHA-224	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A5407)	P-224 with SHA-224	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A5409)	P-224 with SHA-224	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A5407)	P-224 curve	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-FFC-SSC Sp800-56Ar3 (A5407)	MODP-2048	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KDA HKDF SP800-56Cr2 (A5409)	2048 bit key, SHA2-256	KAT	CAST	Module becomes operational	HMAC key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KDF SP800-108 (A5407)	Counter mode using SHA-1, SHA-256, SHA-512	KAT	CAST	Module becomes operational	Key-based key derivation	Test runs at power-on before the integrity test
KDF SP800-108 (A5409)	Counter mode using SHA-1, SHA-256, SHA-512	KAT	CAST	Module becomes operational	Key-based key derivation	Test runs at power-on before the integrity test
PBKDF (A5407)	SHA-1, SHA-256, SHA-512	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A5409)	SHA-1, SHA-256, SHA-512	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
Safe Primes Key Generation (A5407)	MODP-2048	PCT	PCT	Successful key pair generation	SP 800-56Arev3 Section 5.6.2.1.4 method 'b' 1	
AES-CBC (A5406)	128-bit key encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-CBC (A5407)	128-bit key encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-CBC (A5404)	128-bit key encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-CBC (A5405)	128-bit key encrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A5407)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A5408)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A5404)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A5405)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5407)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5404)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5405)	128-bit key decrypt	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test

Table 22: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A5407)	Message Authentication	SW/FW Integrity	Whenever module is powered on	Upon every power on

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A5407)	KAT	CAST	On Demand	Manually
AES-GCM (A5408)	KAT	CAST	On Demand	Manually
AES-GCM (A5405)	KAT	CAST	On Demand	Manually
Counter DRBG (A5407)	KAT	CAST	On Demand	Manually
Counter DRBG (A5408)	KAT	CAST	On Demand	Manually
Counter DRBG (A5405)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A5410)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512/256 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512/256 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA3-224 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA3-224 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA3-256 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA3-256 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA3-384 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA3-384 (A5409)	KAT	CAST	On Demand	Manually
HMAC-SHA3-512 (A5407)	KAT	CAST	On Demand	Manually
HMAC-SHA3-512 (A5409)	KAT	CAST	On Demand	Manually
RSA KeyGen (FIPS186-4) (A5407)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-4) (A5409)	PCT	PCT	On Demand	Manually
RSA SigGen (FIPS186-4) (A5407)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A5409)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-4) (A5407)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A5409)	KAT	CAST	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A5407)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A5409)	PCT	PCT	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A5407)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A5409)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A5407)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A5409)	KAT	CAST	On Demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A5407)	KAT	CAST	On Demand	Manually
KAS-FFC-SSC Sp800-56Ar3 (A5407)	KAT	CAST	On Demand	Manually
KDA HKDF SP800-56Cr2 (A5409)	KAT	CAST	On Demand	Manually
KDF SP800-108 (A5407)	KAT	CAST	On Demand	Manually
KDF SP800-108 (A5409)	KAT	CAST	On Demand	Manually
PBKDF (A5407)	KAT	CAST	On Demand	Manually
PBKDF (A5409)	KAT	CAST	On Demand	Manually
Safe Primes Key Generation (A5407)	PCT	PCT	On Demand	Manually
AES-CBC (A5406)	KAT	CAST	On Demand	Manually
AES-CBC (A5407)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A5404)	KAT	CAST	On Demand	Manually
AES-CBC (A5405)	KAT	CAST	On Demand	Manually
AES-ECB (A5407)	KAT	CAST	On Demand	Manually
AES-ECB (A5408)	KAT	CAST	On Demand	Manually
AES-ECB (A5404)	KAT	CAST	On Demand	Manually
AES-ECB (A5405)	KAT	CAST	On Demand	Manually
AES-XTS Testing Revision 2.0 (A5407)	KAT	CAST	On Demand	Manually
AES-XTS Testing Revision 2.0 (A5404)	KAT	CAST	On Demand	Manually
AES-XTS Testing Revision 2.0 (A5405)	KAT	CAST	On Demand	Manually

Table 24: 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 match the known value or 3) The signature failed to	1) Pre-operational Software Integrity Test failure or 2) Conditional CAST failure 3) Conditional PCT failure	Power cycle the device which results in the module being reloaded into memory and reperforming the pre-operational software integrity test and the Conditional CASTs.	1) Error message "FAILED: fipspost_post_integrity" send to caller or 2) Error message "FAILED:<event>" sent to caller (<event> refers to any of the cryptographic functions listed Table - Conditional Self-Tests 3) Error code "CCEC_GENERATE_KEY_CONSISTENCY" returned for ECDSA and EC Diffie-Hellman Error code "CCRSA_GENERATE_KEY_CONSISTENCY" returned for RSA Error code "CCDH_GENERATE_KEY_CONSISTENCY" returned for Diffie-Hellman

Name	Description	Conditions	Recovery Method	Indicator
	generate/verify successfully in the Conditional PCT. No cryptographic services are provided, and data output is prohibited			

Table 25: 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.

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.
- PBKDF see section 2.7.

11.4.1 IG C.F Compliance

All of the RSA modulus sizes used by the cryptographic module have been CAVP tested and the certificates are listed in the Approved Algorithms Table of this security policy. There are no untested RSA modulus sizes used by the cryptographic module.

11.5 End of Life

The module secure sanitization is accomplished by first powering the module down, which will zeroize all SSPs within volatile memory. Following the power-down, an uninstall by way of system wipe or system update will zeroize the corecrypto-1638.100.62 binary file listed in Table 2.

12 Mitigation of Other Attacks

The module does not claim mitigation of other attacks.