Apple Inc.

Apple corecrypto Module v12

[Intel, User, Software]

FIPS 140-3 Non-Proprietary Security Policy

Document version: 1.0

January 15, 2025

Prepared by:

intertek acumen security

www.acumensecurity.net

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Table of Contents

List of Tables

List of Figures

© 2024 Apple Inc., All rights reserved.

This document may be reproduced and distributed only in its original entirely without revision

1. General

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.

This document is the non-proprietary FIPS 140-3 Security Policy for the Apple, Inc. corecrypto Module v12 [Intel, User, Software] 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 an Overall Security Level 1 module.

This document provides all tables and diagrams (when applicable) required by NIST SP 800-140B. The column names of the tables follow the template tables provided in NIST SP 800-140B.

Table 1 describes the individual security areas of FIPS 140-3, as well as the Security Levels of those individual areas.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

ISO/IEC 24759 Section 6. [Number Below]	FIPS 140-3 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	Not Applicable
8	Non-invasive Security	Not Applicable
9	Sensitive Security Parameter Management	1
10	Self-tests	1
11	Life-cycle Assurance	1
12	Mitigation of Other Attacks	Not Applicable

Table 1 – Security Levels

The module claims an overall Security Level 1.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

2. Cryptographic Module Specification

The Apple corecrypto Module v12 [Intel, User, Software] cryptographic module (hereafter referred to as "the module") is a software module running on a multi-chip standalone general-purpose computing platform. The version of module is 12, written as v12. The module provides implementations of low-level cryptographic primitives to the Host OS's (macOS Monterey 12) Security Framework and Common Crypto. The module has been tested by Acumen Security, LLC. CST lab on the following platforms with and without AES-NI:

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	macOS Monterey 12	MacBook Air	Intel i5 (Amber Lake)	AES-NI
2	macOS Monterey 12	MacBook Air	Intel i5 (Amber Lake)	N/A
3	macOS Monterey 12	iMac	Intel i5 (Comet Lake)	AES-NI
4	macOS Monterey 12	iMac	Intel i5 (Comet Lake)	N/A
5	macOS Monterey 12	MacBook Air	Intel i7 (Ice Lake)	AES-NI
6	macOS Monterey 12	MacBook Air	Intel i7 (Ice Lake)	N/A
7	macOS Monterey 12	MacBook Pro	Intel i7 (Coffee Lake)	AES-NI
8	macOS Monterey 12	MacBook Pro	Intel i7 (Coffee Lake)	N/A
9	macOS Monterey 12	iMac	Intel i7 (Comet Lake)	AES-NI
10	macOS Monterey 12	iMac	Intel i7 (Comet Lake)	N/A
11	macOS Monterey 12	MacBook Pro	Intel i9 (Coffee Lake)	AES-NI
12	macOS Monterey 12	MacBook Pro	Intel i9 (Coffee Lake)	N/A
13	macOS Monterey 12	iMac Pro	Xeon W Sky Lake	AES-NI
14	macOS Monterey 12	iMac Pro	Xeon W Sky Lake	N/A
15	macOS Monterey 12	Mac Pro	Xeon W Cascade Lake	AES-NI
16	macOS Monterey 12	Mac Pro	Xeon W Cascade Lake	N/A

Table 2 – Tested Operational Environments

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

In addition to the platforms listed in Table 2, Apple Inc. has also tested the module on the following platforms and claims vendor affirmation on them (the processor and year per platform have also been specified):

#	Operating System	Hardware Platform		
1	macOS Monterey 12	MacBook Pro	i5 (Ice Lake)	2020
2	macOS Monterey 12	MacBook Pro	i5 (Coffee Lake)	2020, 2019, 2018
3	macOS Monterey 12	MacBook Pro	i7 (Amber Lake)	2019, 2018
4	macOS Monterey 12	MacBook Pro	i7 (Coffee Lake)	2020, 2019, 2018
5	macOS Monterey 12	MacBook Pro	i7 (Ice Lake)	2020
6	macOS Monterey 12	MacBook Pro	i9 (Coffee Lake)	2019, 2018
7	macOS Monterey 12	MacBook Air	i5 (Ice Lake)	2020
8	macOS Monterey 12	MacBook Air	i7 (Ice Lake)	2020
9	macOS Monterey 12	MacBook Air	i5 (Amber Lake)	2019, 2018
10	macOS Monterey 12	MacBook Air	i7 (Amber Lake)	2018
11	macOS Monterey 12	Mac mini	i5 (Coffee Lake)	2018
12	macOS Monterey 12	Mac mini	i7 (Coffee Lake)	2018
13	macOS Monterey 12	iMac	i5 (Comet Lake)	2020
14	macOS Monterey 12	iMac	i7 (Comet Lake)	2020
15	macOS Monterey 12	iMac	i9 (Comet Lake)	2020
16	macOS Monterey 12	iMac	i5 (Coffee Lake)	2019
17	macOS Monterey 12	iMac	i7 (Coffee Lake)	2019
18	macOS Monterey 12	iMac	i9 (Coffee Lake)	2019

Table 3 – Vendor Affirmed Operational Environments

The 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

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

certificate.

The physical perimeter of the module which is also the Tested Operational Environment's Physical Perimeter (TOEPP), is the physical perimeter of the macOS device that contains the module. Consequently, the embodiment of the module is a multi-chip standalone cryptographic module.

(Figure 1) below depicts the following information:

- The location of the module with respect to the operating system (blue dotted outline), other supporting applications so that all the logical and physical layers between the cryptographic boundary and the physical perimeter are clearly defined; and
- The interactions of the logical object of the module with the operating system and other supporting applications resident within the physical perimeter boundary.

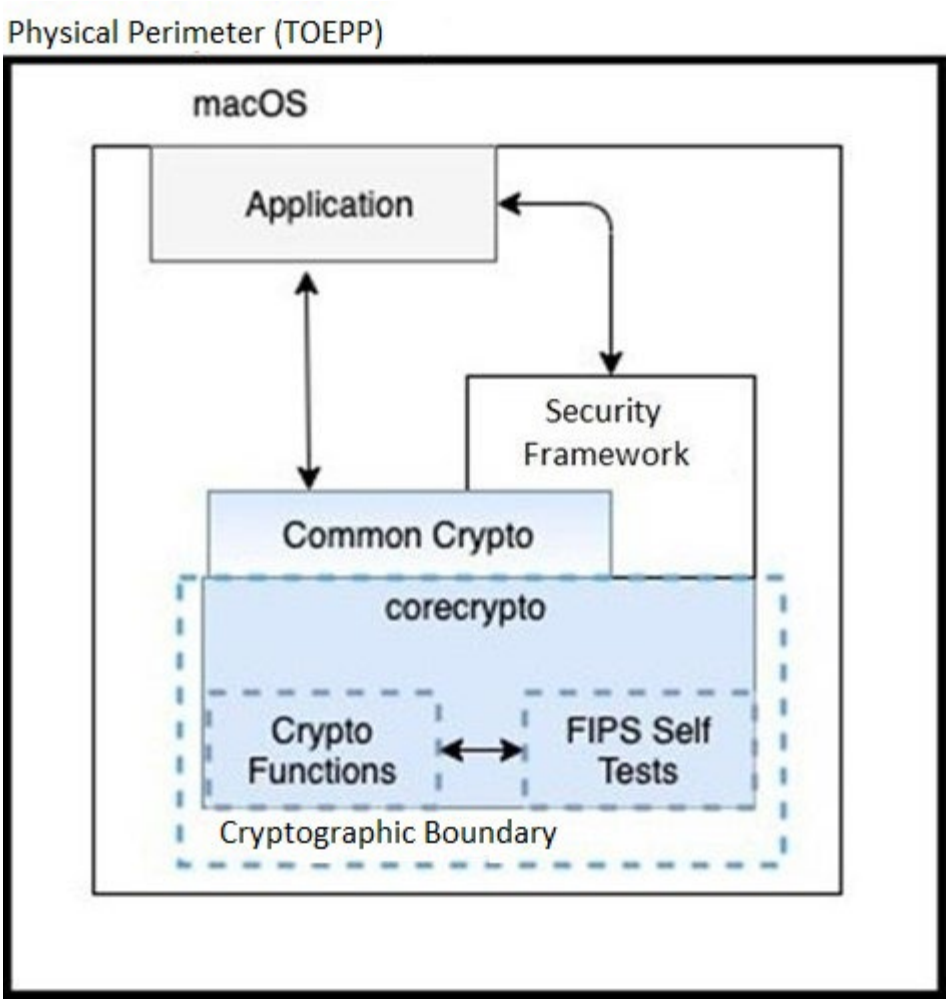

Figure 1 – Cryptographic boundary and physical perimeter

The table below lists all Approved or Vendor-affirmed security functions of the module, including specific key size(s) employed for approved services, and implemented modes of operation. The module

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

is in the Approved mode of operation when the module utilizes the services that use the security functions listed in the table below. The module supports an Approved mode and a non-Approved mode of operation. The module does not support a degraded operation.

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 10 - Non-Approved Services. If the device starts up successfully, then the module has passed all self-tests and is operating in the Approved mode.

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
#A2821 (vng_asm) #A2820 (c_ltc) #A2815 (c_asm) #A2814 (c-aesni) #A2822 (vng_aesni)	CTR_DRBG [SP800-90Ar1]	AES-128, AES-256 Derivation Function Enabled	128, 256 bits	Random Number Generation
#A2820 (c_ltc) #A2817 (c_avx2) #A2816 (c_avx) #A2818 (c_ssse3)	HMAC_DRBG [SP800-90Ar1]	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	112 bits or greater	Random Number Generation
#A2820 (c_ltc) #A2817 (c_avx2) #A2816 (c_avx) #A2818 (c_ssse3)	RSA [FIPS 186-4]	Key Generation (ANSI X9.31), Signature Generation (PKCS#1 v1.5) and (PKCS PSS) Signature Verification	Key Generation 2048, 3072, 4096 Signature Generation Modulus: 2048 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284,	Digital Signature and Asymmetric Key Generation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
		(PKCS#1 v1.5) and (PKCS PSS)	SHA2-512), 3072 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284, SHA2-512), 4096 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284, SHA2-512) Signature Verification Modulus: 1024 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284), 2048 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284, SHA2-512), 3072 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284, SHA2-512), 4096 (SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-284, SHA2-512)	
#A2820 (c_ltc) #A2817 (c_avx2) #A2816 (c_avx) #A2818 (c_ssse3)	ECDSA ANSI X9.62 [FIPS 186-4]	Key Pair Generation (PKG) Public Key Validation (PKV) Signature Generation Signature Verification	Key Pair Generation (PKG): P-224, P-256, P-384, P-521 Public Key Validation (PKV):	Digital Signature and Asymmetric Key Generation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
			P-224, P-256, P-384, P-521 Signature Generation: P-224, P-256, P-384, P-521 SHA2-224, SHA2-256, SHA2-384, SHA2-512 Signature Verification: P-224, P-256, P-384, P-521 SHA-1 (legacy), SHA2-224, SHA2-256, SHA2-384, SHA2-512	
#A2820 (c_ltc) #A2818 (c_ssse3) #A2823 (vng_Intel) #A2817 (c_avx2) #A2816 (c_avx)	SHS [FIPS 180-4]	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 (except A2823)	N/A	Message Digest
#A2820 (c_ltc)	AES [FIPS 197] [SP 800-38 A] [SP 800-38 C] [SP 800-38 D] [SP 800-38 E] [SP 800-38 F]	CBC, CCM, CFB128, CFB8, CMAC, CTR, ECB, GCM, KW, OFB, XTS	Key Length: 128, 192, 256 CMAC :128 XTS (128 and 256-bits key size only)	Symmetric Encryption and Decryption
#A2815 #A2814 (c_aesni)		CBC, CCM, CFB128, CFB8, CTR, ECB, GCM, KW, OFB, XTS	Key Length: 128, 192, 256 XTS (128 and 256-bits key size only)	

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
#A2821 #A2822		CCM, CTR, ECB, GCM		
A2819 (c_glad)		CBC	Key Length: 128, 192, 256	
#A2812 (asm_aesni) #A2813 (asm_x86)		CBC, ECB, XTS	Key Length: 128, 192, 256 XTS (128 and 256-bits key size only)	
#A2823 (vng_Intel) #A2817 (c_avx2) #A2816 (c_avx) #A2820 (c_ltc) #A2818 (c_ssse3)	HMAC [FIPS 198]	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 SHA-512/256 (except A2823)	112 bits or greater	Keyed Hash
#A2820 (c_Itc)	KAS-FFC-SSC [SP800-56r3]	Scheme: dhEphem: KAS Role: initiator, responder	Domain Parameter Generation Methods: MODP-2048, MODP-4096, MODP-6144, MODP-8192	Key Agreement Scheme – Shared Secret Computation
#A2820(c_Itc)	KAS-ECC-SSC [SP800-56r3]	Scheme: ephemeralUnified: KAS Role: initiator, responder	Domain Parameter Generation Methods: P-224, P-256, P-384, P-	Key Agreement Scheme – Shared Secret Computation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
			521	
#A2820 (c_ltc) #A2817 (c_avx2) A2816 (c_avx) #A2818 (c_ssse3)	KBKDF [SP800-108]	KDF Mode: Counter and Feedback MAC Mode: HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512 CMAC-AES128, CMAC-AES192, CMAC-AES256 (only #A2820)	Supported Lengths: 8-4096 Increment 8 Fixed Data Order: Before Fixed Data Counter Length: 32	Key Derivation
#A2820 (c_ltc) #A2817 (c_avx2) #A2816 (c_avx) #A2818 (c_ssse3)	PBKDF [SP800-132]	HMAC with: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	Password length: 8- 128 bytes Increment 1 Salt Length: 128-4096 Increment 8 Iteration Count: 10-1000 Increment 1	Key Derivation
#A2820 (c_ltc)	Safe Primes Key generation	Key Generation for Diffie-Hellman (KAS-FFC-SSC)	Safe Prime Groups: MODP-2048, MODP-	Key Generation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
			3072, MODP-4096, MODP-6144, MODP-8192	
KAS-1 KAS-FFC-SSC Sp800-56Ar3/A2820	KAS	SP 800-56Arev3. KAS-FFC-SSC per IG D.F Scenario 2 path (1)	2048, 4096, 6144, 8192-bit key providing 112, 152, 176, 200 bits of encryption strength	Key Agreement – Shared Secret Computation
KAS-2 KAS-ECC-SSC Sp800-56Ar3/A2820	KAS	SP 800-56Arev3. KAS-ECC-SSC per IG D.F Scenario 2 path (1)	224 , 256, 384, 521-bit key providing 112, 128, 192, 256 bits of encryption strength	Key Agreement – Shared Secret Computation
KTS AES-KW/A2814 AES-KW/A2815 AES-KW/A2820	KTS	SP 800-38D and SP 800-38F; key wrapping per IG D.G	128, 192, and 256- bit keys providing 128, 192, or 256 bits of encryption strength	Key Wrapping
Vendor Affirmed	Cryptographic Key Generation (CKG) [SP800-133r2]	RSA Key Generation (ANSI X9.31), ECDSA Key Pair Generation (PKG), Sections 4, 5.1, 5.2, 6.2.2 and 6.2.3 per SP800-132r2	RSA: Modulus: 2048, 3072, 4096 ECDSA: P-224, P-256, P-384, P-521	Key Generation

Table 4 – Approved Algorithms

This module does not have non-Approved but Allowed Algorithms used in the Approved mode of operation.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

The table below list non-Approved but Allowed security functions with no security claimed:

Algorithm	Caveat	Use / Function
MD5	no security claimed	Message Digest (used as part of the TLS key establishment scheme only), Digest Size: 128-bit

Table 5 – Non-Approved Algorithms Not Allowed in the Approved Mode of Operation with No Security Claimed

The table below lists Non-Approved security functions that are not Allowed in the Approved Mode of Operation:

Algorithm/Function	Use/Function
RSA Signature Generation / Signature Verification / Asymmetric Key Generation	ANSI X9.31 Key Pair Generation Key Size < 2048 PKCS#1 v1.5 and PSS Signature Generation Key Size < 2048 PKCS#1 v1.5 and PSS Signature Verification Key Size< 1024
RSA Key Wrapping	OAEP, PKCS#1 v1.5 and -PSS schemes
Diffie-Hellman	Key agreement scheme
EC Diffie-Hellman	Key agreement scheme
Ed25519	Key Agreement Sig(gen) Sig(ver)
ANSI X9.63 KDF	Hash based Key Derivation Function
RFC6637	Key Derivation Function
HKDF [SP800-56Cr1]	Key Derivation Function
DES	Encryption / Decryption Key Size 56-bits
CAST	Encryption / Decryption Key Sizes 40 to 128-bits in 8-bit increments
RC4	Encryption / Decryption Key Sizes 8 to 4096-bits
RC2	Encryption / Decryption Key Sizes 8 to 1024-bits
MD2	Message Digest Digest size 128-bit
MD4	Message Digest Digest size 128-bit
RIPEMD	Message Digest Digest size 160-bits
ECDSA	PKG: Curve P-192 PKV: Curve P-192 Signature Generation: Curve P-192 Signature Verification: Curve P-192

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

	Key Pair Generation for compact point representation of points
	Key Pair Generation for compact point representation of points
Integrated Encryption Scheme on elliptic curves (ECIES)	Encryption / Decryption
Blowfish	Encryption / Decryption
OMAC (One-Key CBC MAC)	MAC generation
Triple-DES [SP 800-67]	CBC, CTR, CFB64, ECB, CFB8, OFB

Table 6 – Non-Approved Algorithms Not Allowed in the Approved Mode of Operation

Overall Security Rules of Operation

- AES-GCM IV is constructed in compliance with IG C.H scenario 1a (TLS 1.2) and scenario 1b (IPsec-v3). The TLS and IPSec/IKE protocols have not been reviewed or tested by the CAVP and CMVP. The GCM IV generation follows RFC 5288 and shall only be used for the TLS protocol version 1.2. 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 both protocols 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; however, it is met implicitly. The module does not retain any state when power is lost. As indicated in Table 10, column Storage, the module exclusively uses volatile storage. This means that AES-GCM key/IVs are not persistently stored during power off: therefore, there is no re-connection possible when the power is back on with re-generation of the key used for GCM. After restoration of the power, the user of the module (e.g., TLS, IKE) along with User application that implements the protocol, must perform a complete new key establishment operation using new random numbers (Entropy input string, DRBG seed, DRBG internal state V and Key, shared secret values that are not retained during power cycle, see table 10) with subsequent KDF operations to establish a new GCM key/IV pair on either side of the network communication channel.
- AES-XTS mode is only approved for hardware storage applications. The length of the AES-XTS data unit does not exceed 220 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.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

3. Cryptographic Module Interfaces

As a software-only module, the module does not have physical ports. For the purpose of the FIPS 140-3 validation, the physical ports are interpreted to be the physical ports of the hardware platform on which it runs.

The logical interfaces are the application program interface (API) through which applications request services and the Operating System calls that the module invokes.

The underlying logical interfaces of the module are the C language User Interfaces (APIs). In detail these interfaces are described in (Table 7):

Logical interface	Data that passes over port/interface
Data input interface	Data inputs are provided in the variables passed in the API and callable service invocations, generally through caller-supplied buffers
Data output interface	Data outputs are provided in the variables passed in the API and callable service invocations, generally through caller-supplied buffers
Control input interface	Control inputs which control the mode of the module are provided through dedicated parameters.
Status output interface	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 7 – Ports and Interfaces

The module does not support a control output interface.

The module is optimized for library use within the macOS User space and does not contain any terminating assertions or exceptions. It is implemented as a macOS dynamically loadable library. The dynamically loadable library is loaded into the macOS User and its cryptographic functions are made available to the macOS application. Any internal error detected by the module is reflected back to the caller with an appropriate return code. The calling macOS 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. It is the responsibility of the caller to handle exceptional conditions in a FIPS 140-3 appropriate manner.

Caller-induced or internal errors do not reveal any sensitive material to callers. Cryptographic bypass

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

capability is not supported by the module.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

4. Roles, Services, and Authentication

The module supports a single instance of one authorized role: The Crypto Officer. No support is provided for multiple concurrent operators or a Maintenance Operator.

Role	Service	Input	Output
Crypto Officer (CO)	AES Encryption / Decryption (Perform approved security functions)	Input for Encryption: key and plain text Input for Decryption: key and cipher text	Output for Encryption: cipher text Output for Decryption: plain text
	AES Key Wrapping (Perform approved security functions)	key encryption key and key to be wrapped	wrapped key
	Secure Hash Generation (Perform approved security functions)	Message	Hash value
	HMAC generation (Perform approved security functions)	HMAC key and message	keyed Hash value
	RSA signature generation and verification (Perform approved security functions)	Input for SigGen: RSA private key and message Input for SigVer: RSA public key and signature	Output SigGen: signature Output for Sigver: True or False
	ECDSA signature generation and verification (Perform approved security functions)	Input for SigGen: ECDSA private key and message Input for SigVer: ECDSA public key and signature	Output for SigGen: signature Output for SigVer: True or False
	Random number generation (Perform approved security functions)	Entropy input string, nonce	Random numbers
	PBKDF (Perform approved security functions)	password	derived key
	KBKDF (Perform approved security functions)	key derivation key	Derived key
	ECDSA (key pair generation) (Perform approved security functions)	random numbers	generated private and public key pair
	RSA (key pair generation)	random prime numbers	generated private and public key pair

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Role	Service	Input	Output
	(Perform approved security functions)		
	Safe primes key generation (Perform approved security functions)	key size	generated private and public key pair
	Diffie-Hellman Key Shared Secret Computation (Perform approved security functions)	domain parameter, received public key and possessed private key	shared secret
	EC Diffie-Hellman Shared Secret Computation (Perform approved security functions)	domain parameter, received public key and possessed private key	shared secret
	Release all resources of symmetric crypto function context (Perform zeroisation)	handler of symmetric crypto function context	zeroised and released memory space
	Release all resources of hash context (Perform zeroisation)	handler of hash context	released memory space
	Release of all resources of Diffie-Hellman context for Diffie-Hellman and EC Diffie-Hellman (Perform zeroisation)	handler of (EC) DiffieHellman context	zeroised and released memory space
	Release of all resources of key derivation function context (Perform zeroisation)	handler of key derivation function context	zeroised and released memory space
	Release of all resources of asymmetric crypto function context (Perform zeroisation)	handler of asymmetric crypto function context	zeroised and released memory space
	Self-test (Perform self-tests)	power	Pass/Fail status
	Show Status	API invocation	Operational/Error status
	Show Module Info (Show module's versioning information)	API invocation	Module Base Name + Module Version Number

Table 8 – Roles, Service Commands, Input and Output

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

FIPS 140-3 does not require an authentication mechanism for level 1 modules. Therefore, the module does not implement an authentication mechanism for Crypto Officer. The Crypto Officer role is authorized to access all services provided by the module (see Table 9 - Approved Services and Table 10 - Non-Approved Services below).

The module implements a dedicated API function to indicate if a requested service utilizes an approved security function. For services listed in Table 9 - Approved Services, the indicator function returns 1. For services listed in Table 10 - Non-Approved Services, the indicator function returns 0.

The table below lists all approved services that can be used in the approved mode of operation. The abbreviations of the access rights to keys and SSPs have the following interpretation:

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
AES Encryption / Decryption (Perform approved security functions)	Input for Encryption: key and plain text Output for Encryption: cipher text Input for Decryption: key and cipher text Output for Decryption: plain text	Symmetric Encryption and Decryption AES-CBC (#A2820, #A2815, #A2814, A2819, #A2812, #A2813) AES-ECB (#A2820, #A2815, #A2814, #A2812, #A2813, #A2821, #A2822) AES-CCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-GCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-CFB128(#A2820, #A2815, #A2814) AES-CFB8(#A2820, #A2815, #A2814) AES-OFB (#A2820, #A2815, #A2814) AES-CTR (#A2820, #A2815, #A2814, #A2821, #A2822)	AES key	CO	W, E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
		AES-XTS (#A2820, #A2815, #A2814, #A2812, #A2813) CMAC (#A2820)				
AES Key Wrapping (Perform approved security functions)	Input: key encryption key and key to be wrapped Output: wrapped key	Key Wrapping KW (#A2820, #A2815, #A2814)	AES key, key to be wrapped, wrapped key	CO	W, R, E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Secure Hash Generation (Perform approved security functions)	Input: message Output: Hash value	Message Digest SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 (except for A2823) (#A2820, #A2818 #A2823, #A2817, #A2816)	none	CO	N/A	1
HMAC generation (Perform approved security functions)	Input: HMAC key and message Output: keyed Hash value	Keyed Hash SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/256 (except for A2823) (#A2820, #A2823, #A2818 #A2817, #A2816)	HMAC key	CO	W, E	1
RSA signature generation and verification (Perform approved security functions)	Input for SigGen: RSA private key and message Output: signature Input for SigVer: RSA public key and signature Output: True or False	Digital Signature Generation (PKCS#1 v1.5) and (PKCS PSS) Signature Verification (PKCS#1 v1.5) and (PKCS PSS) (#A2820, #A2817, #A2816, #A2818)	RSA Key Pair (including intermediate keygen values)	CO	W, E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
ECDSA signature generation and verification (Perform approved security functions)	Input for SigGen: ECDSA private key and message Output: signature Input for SigVer: ECDSA public key and signature Output: True or False	Digital Signature Generation: P-224, P-256, P-384, P-521 Signature Verification: P-224, P-256, P-384, P-521 (#A2820, #A2817, #A2816, #A2818)	ECDSA Key Pair (including intermediate keygen values)	CO	W, E	1
Random number generation (Perform approved security functions)	Input: Entropy input string, nonce Output: Random numbers (DRBG Output)	Random number generation CTR_DRBG (AES-128, AES-256) #A2820, #A2821, #A2815, #A2814, #A2822 HMAC_DRBG (SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512)	Entropy Input String, Seed, DRBG V and DRBG Key, random numbers (DRBG Output)	CO	G, R, W, E, Z	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
		#A2820, #A2817, #A2816, #A2818 #A2820, #A2817, #A2816, #A2818 CKG				
PBKDF (Perform approved security functions)	Input: password Output: derived key	Key Derivation PBKDF HMAC with: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 #A2820, #A2817, #A2816, #A2818 CKG	PBKDF Derived Keys (including password hash), PBKDF Password	CO	G, R, W, E	1
KBKDF (Perform approved security functions)	Input: key derivation key Output: derived key	Key Derivation KDF Mode: Counter and Feedback MAC Mode: HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512 Supported Lengths: 8-4096 Increment 8 Fixed Data Order: Before Fixed Data Counter Length: 32 KDF Mode:	KBKDF Key Derivation Key, KBKDF Derived Key	CO	G, R, W, E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
		Counter AES-CMAC based (for A2820 only) #A2820, #A2817, #A2816, #A2818 CKG				
ECDSA (key pair generation) (Perform approved security functions)	Input: random numbers Output: generated private and public key pair	Key Pair Generation (PKG): P-224, P-256, P-384, P-521 Public Key Validation (PKV): P-224, P-256, P-384, P-521 #A2820, #A2817, #A2816, #A2818 CKG	ECDSA Key Pair (including intermediate keygen values)	CO	G, R, E	1
RSA (key pair generation) (Perform approved security functions)	Input: random prime numbers Output: generated private and public key pair	Asymmetric Key Generation Modulus 2048, 3072, 4096 #A2820, #A2817, #A2816, #A2818 CKG	RSA Key Pair (including intermediate keygen values)	CO	G, R, E	1
Diffie-Hellman Key Shared Secret Computation (Perform approved	Input: domain parameter, received public key and possessed private key Output: shared secret	KAS-FFC-SSC #A2820	DH Key Pair (including intermediate keygen values)	CO	G, R, W, E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
security functions)						
EC Diffie-Hellman Shared Secret Computation (Perform approved security functions)	Input: domain parameter, received public key and possessed private key Output: shared secret	KAS-ECC-SSC #A2820	ECC CDH Key Pair (including intermediate keygen values)	CO	G, R, W, E	1
Safe primes key generation	Input: key size Output: generated private and public key pair	Safe primes key pair generation #A2820 CKG	DH Key Pair (including intermediate keygen values)	CO	G, R, E	1
Release all resources of symmetric crypto function context (Perform zeroisation)	Input: handler of symmetric crypto function context Output: zeroised and released memory space	N/A	AES key	CO	Z	1
Release all resources of hash context (Perform zeroisation)	Input: handler of hash context Output: released memory space	N/A	HMAC key	CO	Z	1
Release of all resources of Diffie-Hellman context for Diffie-Hellman and	Input: handler of (EC) Diffie-Hellman context Output: zeroised and released memory space	N/A	DH Key Pair (including intermediate keygen values), ECC CDH Key Pair (including intermediate	CO	Z	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
EC Diffie-Hellman (Perform zeroisation)			keygen values), DH Shared Secret, ECC CDH Shared Secret			
Release of all resources of key derivation function context (Perform zeroisation)	Input: handler of key derivation function context Output: zeroised and released memory space	N/A	KBKDF Key Derivation Key, PBKDF Password, KBKDF Derived Key and PBKDF Derived Key	CO	Z	1
Release of all resources of asymmetric crypto function context (Perform zeroisation)	Input: handler of asymmetric crypto function context Output: zeroised and released memory space	N/A	RSA/EC/DH keys	CO	Z	1
Self-test (Perform Self-tests)	Input: power Output: Pass/Fail status	AES-CCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-GCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-XTS (#A2820, #A2815, #A2814, #A2812, #A2813) AES-CBC (#A2820, #A2815, #A2814, A9819, #A2812, #A2813)	All SSPs	CO	E	1

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
		AES-ECB (#A2820, #A2815, #A2814, #A2812, #A2813, #A2821, #A2822) HMAC_DRBG (#A2820, #A2817, #A2816, #A2818) CTR_DRBG, #A2820, #A2821, #A2815, #A2814, #A2822 HMAC (#A2820, #A2818 #A2823, #A2817, #A2816) RSA Signature Generation (#A2820, #A2817, #A2816, #A2818) RSA Signature Verification (#A2820, #A2817, #A2816, #A2818) ECDSA Signature Generation (#A2820, #A2817, #A2816, #A2818)				

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Services	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
		ECDSA Signature Verification (#A2820, #A2817, #A2816, #A2818) DH and ECDH Z computation (#A2820) PBKDF (#A2820, #A2817, #A2816, #A2818) KBKDF (#A2820, #A2817, #A2816, #A2818)				
Show Status	Input: API invocation Output: Operational/Error status	N/A	None	CO	N/A	Status returned
Show Module Info (Show module's versioning information)	Input: API invocation Output: Module Base Name	N/A	None	CO	N/A	Versioning information returned

Table 9 – Approved Services

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

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Service	Description	Algorithms Accessed	Role	Indicator
Triple-DES encryption / decryption	Module does not meet FIPS 140-3 IG C.G because it does not have a control over the number of blocks to be encrypted under the same Triple-DES key. Input for Encryption: key and plain text Output for Encryption: cipher text Input for Decryption: key and cipher text Output for Decryption: plain text	Triple-DES	Crypto Officer (CO)	0
(other) symmetric encryption / decryption	They are non-approved encryption algorithms. Input for Encryption: key and plain text Output for Encryption: cipher text Input for Decryption: key and cipher text Output for Decryption: plain text	Blowfish, CAST5, DES, ECIES, RC2, RC4	Crypto Officer (CO)	0
RSA Key Wrapping	The CAST does not perform the full KTS, only the raw RSA encrypt/decrypt Input: RSA public key and key to be wrapped Output: wrapped key	RSA encrypt/decrypt	Crypto Officer (CO)	0

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Service	Description	Algorithms Accessed	Role	Indicator
RSA Signature Generation/Signature Verification/Key-pair Generation	ANSI X9.31 Key Pair Generation Key Size < 2048 PKCS#1 v1.5 and PSS Signature Generation Key Size < 2048 PKCS#1 v1.5 and PSS Signature Verification Key Size < 1024	RSA KeyGen RSA SigGen RSA SigVer	Crypto Officer (CO)	0
ECDSA PKG, PKV, Signature Generation/Signature Verification	ECDSA keys with curve P-192	ECDSA PKG, PKV, SigGen/SigVer	Crypto Officer (CO)	0
Ed25519 Key Generation, Signature Generation/Signature Verification	256-bit key	Ed25519 KeyGen Ed25519 SigGen Ed25519 SigVer	Crypto Officer (CO)	0
ANSI X9.63 Key Derivation	SHA-1 hash-based	SHA-1	Crypto Officer (CO)	0
SP800-56Cr1 Key Derivation (HKDF)	SHA-256 hash-based	SHA-256	Crypto Officer (CO)	0
RFC6637 Key Derivation	SHA hash based	SHA-256, SHA-512, AES-128, AES-256	Crypto Officer (CO)	0
OMAC Message Authentication Code Generation and Verification	One-Key CBC MAC using 128-bit key	OMAC	Crypto Officer (CO)	0
Message digest verification	Input: message Output: message digest	MD2, MD4, RIPEMD	Crypto Officer (CO)	0
Diffie-Hellman	KAS-FFC-SSC using key sizes < 2048	KAS-FFC-SSC	Crypto Officer (CO)	0
EC Diffie-Hellman	KAS-ECC-SSC using curves < P-224	KAS-ECC-SSC	Crypto Officer (CO)	0
Ed25519 Key Agreement	Input: peer public key, own private key Output: shared secret	Ed25519	Crypto Officer (CO)	0
Integrated Encryption Scheme on elliptic	Encrypt: Input: peer public key, plaintext Output: public key,	Integrated Encryption Scheme on	Crypto Officer (CO)	0

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Service	Description	Algorithms Accessed	Role	Indicator
curves (ECIES) Encryption/Decryption	ciphertext (with authentication tag) Decrypt: Input: authentication tag, ciphertext, own private key Output: plaintext message or error	elliptic curves (ECIES)		

Table 10 – Non-Approved Services

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

5. Software/Firmware Security

Integrity Techniques

The Apple corecrypto Module v12 [Intel, User, Software] is in the form of binary executable code. A software integrity test is performed on the runtime image of the module. The HMAC-SHA2-256 implemented in the module is used as an 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. The Software Integrity Key (HMAC-SHA2-256 with 256 bits of security strength), a non-SSP, is stored in the module binary computed during build.

On-Demand Integrity Test

The integrity test is also performed as part of the Pre-Operational Self-Tests. It is automatically executed at power-on. It can also be invoked by powering-off and reloading the module to meet the on-demand request for integrity test.

In addition, the module provides the Self-Test service to perform self-tests, including integrity test and algorithm tests, on demand.

Software Loading

The module does not support loading of any additional software.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

6. Operational Environment

Applicability

The Apple corecrypto Module v12 [Intel, User, Software] operates in a modifiable operational environment per FIPS 140-3 level 1 specifications. The module is supplied as part of macOS Monterey 12, a commercially available general-purpose operating system executing on the hardware specified in section 2.

Policy

The operating system is restricted to a single operator (single-user mode; i.e. concurrent operators are explicitly excluded).

When the operating system loads the module into memory, it invokes the Self-Test functionality, which in turn runs the mandatory self-tests.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

7. Physical Security

The FIPS 140-3 physical security requirements do not apply to the Apple corecrypto Module v12 [Intel, User, Software] since it is a software module.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

8. Non-invasive Security

Currently, the non-invasive security is not required by FIPS 140-3 (see NIST SP 800-140F). The requirements of this area are not applicable to the module.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

9. Sensitive Security Parameter Management

The following table summarizes the keys and Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module:

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
AES Key (CSP)	Size: 128, 192, 256 Strength: 128, 192, 256	Symmetric Encryption and Decryption AES-CBC (#A2820, #A2815, #A2814, A9819, #A2812, #A2813) AES-ECB (#A2820, #A2815, #A2814, #A2812, #A2813, #A2821, #A2822) AES-CCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-GCM (#A2820, #A2815, #A2814, #A2821, #A2822) AES-CFB128(#A2820, #A2815, #A2814) AES-CFB8(#A282	N/A	Import and Export to calling application	N/A	N/A: The module does not provide persistent keys/SSPs storage	Automatic zeroisation when structure is deallocated or when the system is powered down	Symmetric Encryption and Decryption

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
		0, #A2815, #A2814) AES-OFB (#A2820, #A2815, #A2814) AES-CTR (#A2820, #A2815, #A2814, #A2821, #A2822) AES-XTS (#A2820, #A2815, #A2814, #A2812, #A2813) CMAC (#A2820)						
HMAC Key (CSP)	Min: 112 bits	HMAC generation SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512 SHA2-512/256 #A2820, #A2818 #A2823, #A2817, #A2816			N/A			Keyed Hash
ECDSA Key Pair (CSP)	Curves: P-224,	ECDSA Keygen #A2820, #A2817,	The key pairs are generated conforma		N/A			Digital Signature

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
	P-256, P-384, P-521 Strength: 112, 128, 192, 256	#A2816, #A2818 CKG	nt to SP800-133r2 (CKG) using FIPS186-4 Key Generation method, and the random value used in the key generation is generated using SP800-90Ar1 DRBG					
RSA Key Pair (CSP)	Modulus: 2048, 3072, 4096 Strength: 112, 128, 152	RSA Keygen #A2820, #A2817, #A2816, #A2818 CKG			N/A			Digital Signature
Entropy Input String (CSP)	256 bits	CTR_DRBG (AES-128, AES-256) #A2820, #A2821, #A2815, #A2814, #A2822 HMAC_DRBG (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512) #A2820, #A2817,	Obtained from the ENT	Import from OS; No Export	N/A			Random Number Generation
Seed (CSP)	256 bits		Derived from entropy input string as defined by SP 800-90Ar1		N/A			

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
DRBG V (CSP)	256 bits	#A2816, #A2818	Generated Internally using the approved DRBG	N/A	N/A	N/A: The module does not provide persistent keys/SSPs storage	Automatic zeroisation when structure is deallocated or when the system is powered down	Generation
DRBG Key (CSP)			Generated Internally using the approved DRBG	N/A	N/A	N/A: The module does not provide persistent keys/SSPs storage	Automatic zeroisation when structure is deallocated or when the system is powered down	Generation
DRBG Output (CSP)		CTR_DRBG (AES-128, AES-256) #A2820, #A2821, #A2815, #A2814, #A2822 HMAC_DRBG (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512) #A2820,	Generated Internally using the approved DRBG	N/A	N/A	N/A: The module does not provide persistent keys/SSPs storage	Automatic zeroisation when structure is deallocated or when the system is powered down	Generation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
		#A2817, #A2816, #A2818 CKG						
PBKDF Derived Keys (including password hash) (CSP)	Min: 112 bits	PBKDF #A2820, #A2817, #A2816, #A2818 CKG	Internally generated via SP800-132 PBKDF key derivation algorithm	No Import; Export to calling application	N/A			Key Derivation
PBKDF Password (CSP)	N/A	PBKDF #A2820, #A2817, #A2816, #A2818	N/A	imported from calling application, No Export	N/A			Key Derivation
KBKDF Key Derivation Key (CSP)	Min: 112	KBKDF Key Derivation KDF Mode: Counter and Feedback MAC Mode: HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512 #A2820, #A2817,	N/A	imported	N/A			
KBKDF Derived Key (CSP)	Min: 112 bits		Internally generated via SP800-108 KBKDF key derivation algorithm	No Import; Export to calling application	N/A			Key Derivation

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
		#A2816, #A2818 KBKDF CMAC based (for A2820 only) CKG						
DH Key Pair (CSP)	2048	KAS-FFC-SSC #A2820 CKG	Generated using Safe-prime groups MODP groups belonging to (RFC 3526)	Import from calling application, No Export	N/A			KAS-SSC FFC
DH Shared Secret (CSP)			N/A	No Import; Export to calling application	Computed using SP800-56Ar3 DH shared secret computation			

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroisation	Use and related keys
ECC CDH Key Pair (CSP)	Curves: P-224, P-256, P-384, P-521 Strength: 112, 128, 192, 256	KAS-ECC-SSC #A2820 CKG	Generated using FIPS 186-4 Key Generation method, and the random value used in key generation is generated using SP800-90Ar1 DRBG	Import from calling application, No Export	N/A			KAS-SSC ECC
ECC CDH Shared Secret (CSP)	Curves: P-224, P-256, P-384, P-521 Strength: 112, 128, 192, 256	KAS-ECC-SSC #A2820	Internally generated via SP800-56Ar3 ECC CDH shared secret computation	No Import; Export to calling application	N/A			KAS-SSC ECC

Table 11 – SSPs

The Software Integrity Key (HMAC-SHA2-256 with 256 bits of security strength), a non-SSP, is stored in the module binary computed during build.

Random Number Generation

A NIST approved deterministic random bit generator based on a block cipher as specified in NIST [SP 800-90Ar1] is used. The default Approved DRBG used for random number generation is a CTR_DRBG

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

using AES-256 with derivation function and without prediction resistance. The random numbers used for key generation are all generated by CTR_DRBG in this module. Per section 10.2.1.1 of [SP 800-90Ar1], the internal state of CTR_DRBG is the value V, Key and a reseed counter.

The module also employs a HMAC_DRBG for random number generation. The HMAC_DRBG is only used at the early boot time of macOS User for memory randomization. The output of HMAC_DRBG is not used for key generation. Per section 10.1.2.1 of [SP 800-90Ar1], the internal state of HMAC_DRBG is the value V, Key and a reseed counter.

The deterministic random bit generators are seeded by read_random. The read_random is the User Space interface that extracts random bits from the entropy pool. The output of entropy pool provides 256-bits of entropy to seed and reseed SP800-90B DRBG during initialization (seed) and reseeding (reseed).

Entropy sources	Minimum number of bits of entropy	Details
NISP SP800-90B compliant ENT (P) ESV Cert. #E14	256-bits per 256-bit output sample	The seed is provided by an SP 800-90B compliant entropy source

Table 12 – Non-Deterministic Random Number Generation Specification

Key / SSP Generation

The module generates Keys and SSPs in accordance with FIPS 140-3 IG D.H. The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per sections 4, 5.1, 5.2, 6.2.2 and 6.2.3 [SP800-133r2] (vendor affirmed), compliant with [FIPS186-4], and using DRBG compliant with [SP800-90Ar1]. A seed (i.e., the random value) used in asymmetric key pair generation is a direct output from [SP800-90Ar1] CTR_DRBG. The key generation service for RSA, Diffie-Hellman, and EC key pairs as well as the [SP 800-90Ar1] DRBG have been ACVT tested with algorithm certificates found in Table 4.

Keys/SSPs Establishment

The module provides the following key/SSP establishment services in the Approved mode:

- AES-Key Wrapping
    - The module implements a Key Transport Scheme (KTS) using AES-KW compliant to [SP800-38F]. The SSP establishment methodology provides between 128 and 256 bits of encryption strength.
- Diffie-Hellman Shared Secret Computation
    - The module implements a shared secret compliant to [SP800-56Ar3]. The shared secret computation provides between 112 and 200 bits of encryption strength.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

- EC Diffie-Hellman Shared Secret Computation
    - The module implements a shared secret Z compliant to [SP800-56Ar3]. The shared secret computation provides between 112 and 256 bits of encryption strength.
- PBKDF Key Derivation
    - The module implements a CAVP compliance tested key derivation function compliant to [SP800-132]. The service returns the key derived from the provided password to the caller. The password consists of at least eight alphanumeric characters from the ninety-six (96) printable and human- readable characters[2]. PBKDFv2 is implemented to support all options specified in section 5.4 of [SP800-132]. The keys derived from [SP800-132] map to section 4.1 of [SP800-133r2] as indirect generation from DRBG. The derived keys may only be used in storage applications.
- KBKDF Key Derivation
    - The KBKDF is compliant to [SP800-108]. The module implements both Counter and Feedback modes with HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2-256, HMAC-SHA2-384, or HMAC-SHA2-512 as the PRF.

Keys/SSPs Import/Export

All keys and SSPs that are entered from, or output to module, are entered from or output to the invoking application running on the same device. Keys/SSPs entered into the module are electronically entered in plain text form. Keys/SSPs are output from the module in plain text form if required by the calling application.

The module allows the output of plaintext CSPs (i.e., EC/DH/RSA Key Pairs). To prevent inadvertent output of sensitive information, the module performs the following two independent internal actions:

1. The module will internally request the random number generation service to obtain the random numbers and verify that the service completed without errors.

2. Once the keys are generated the module will perform the pairwise consistency test and verify that the test is completed without errors.

Only after successful completion of both actions, are the generated CSPs output via the KPI output parameter in plaintext.

Keys/SSPs Storage

The Apple corecrypto Module v12 [Intel, User, Software] stores ephemeral keys/SSPs in memory only. They are received for use or generated by the module only at the command of the calling application. The module does not provide persistent keys/SSPs storage.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

The module protects all keys/SSPs through the memory separation and protection mechanisms provided by the operating system. No process other than the module itself can access the keys/SSPs in its process' memory.

Keys/SSPs Zeroization

Keys and SSPs are zeroised when the appropriate context object is destroyed or when the system is powered down. Input and output interfaces are inhibited while zeroisation is performed.

© 2024 Apple Inc., All rights reserved.
This document may be reproduced and distributed only in its original entirely without revision

10. Self-tests

The module performs pre-operational self-tests automatically when the module is loaded into memory; the pre- operational self-tests triggered at power-on ensure that the module is not corrupted and that the cryptographic algorithms work as expected.

FIPS 140-3 only requires that software/firmware integrity test(s) and the requisite cryptographic algorithm(s) be tested during power-up, but the Apple corecrypto Module v12 [Intel, User, Software] runs all Cryptographic Algorithm Tests (CASTs) during power-up as well.

The following tests are performed each time the Apple corecrypto Module v12 [Intel, User, Software] starts. If any of the following tests fails the device (tested platform) fails to startup. To invoke the self-tests (pre-operational and CASTs) on demand (and periodically), the user may reboot the system.

While the module is executing the self-tests, services are not available and input and output are inhibited. The self-tests are implemented for the following algorithms:

- **Pre-operational Self-Tests:**
    - HMAC-SHA2-256: Used for module integrity test

- **Conditional Self-Tests:**
    - Conditional Cryptographic Algorithm Self-tests (CAST):
        - AES CBC 128 bits Encrypt KAT
        - AES CCM 128 bits Encrypt KAT
        - AES GCM 128 bits Encrypt KAT
        - AES XTS 128 bits Encrypt KAT
        - AES ECB 128 bits Encrypt KAT
        - AES CMAC 128 bits Encrypt KAT
        - AES CBC 128 bits Decrypt KAT
        - AES CCM 128 bits Decrypt KAT
        - AES GCM 128 bits Decrypt KAT
        - AES XTS 128 bits Decrypt KAT
        - AES ECB 128 bits Decrypt KAT
        - AES CMAC 128 bits Decrypt KAT
        - CTR_DRBG KAT
            - Generate, Instantiate and Reseed
        - HMAC_DRBG KAT
            - Generate, Instantiate and Reseed
        - HMAC-SHA-1 KAT; covers SHA-1 KAT
        - HMAC-SHA2-256 KAT; covers SHA2-256 KAT
        - HMAC-SHA2-512 KAT; covers SHA2-512 KAT

Public Material – May be reproduced only in its original entirety (without revision).

        - RSA 2048 bits SHA2-256 Signature Generation KAT
        - RSA 2048 bits SHA2-256 Verify KAT
        - ECDSA P-224 SHA2-224 Sig Gen KAT
        - ECDSA P-224 SHA2-224 Sig Ver KAT
        - KAS-FFC-SSC KAT
        - KAS-ECC-SSC KAT
        - PBKDF KAT
        - KBKDF counter KAT
        - NIST SP 800-90B Repetitive Count Test (RCT)
        - NIST SP 800-90B Adaptive Proportion Test (APT)

    - Pairwise consistency test when generating ECDSA key pairs (for signature generation/verification/key agreement (KAS-ECC-SSC))
    - Pairwise consistency test when generating RSA key pairs (for signature generation/verification)
    - Pairwise consistency test when generating DH key pairs (for key agreement)

Integrity Test

A software integrity test is performed on the runtime image of the Apple corecrypto Module v12 [Intel, User, Software]. The module's HMAC-SHA2-256 is used as an approved algorithm for the integrity test. If the test fails, then the device powers itself off. The HMAC value is pre-computed at build time and stored in the module. The HMAC value is recalculated during runtime and compared with the stored value.

Conditional Tests

The following sub-sections describe the conditional tests supported by the Apple

corecrypto Module v12 [Intel, User, Software].

Cryptographic algorithm tests

The Apple corecrypto Module v12 [Intel, User, Software] runs all Cryptographic Algorithm Tests during power-up. These tests are detailed above in this section.

Pairwise Consistency Test

The Apple corecrypto Module v12 [Intel, User, Software] does generate asymmetric keys and performs all required pair-wise consistency tests on the newly generated key pairs.

Error Handling

If any of the above-mentioned self-tests fail, the module reports the cause of the error and enters an error state where no cryptographic services are provided and data output is

Public Material – May be reproduced only in its original entirety (without revision).

prohibited. The only method to clear the error state is to power cycle the device. The module will only enter into the operational state after successfully passing the preoperational software integrity test and the Conditional CASTs. The module returns the "FAILED: fipspost_post_integrity" error indicator in case of a software integrity test failure, "FAILED: <algorithm>" in case of a CAST failure.

Public Material – May be reproduced only in its original entirety (without revision).

11. Life-cycle Assurance

Delivery and Operation

The module is built into macOS Monterey 12 and delivered with macOS. There is no standalone delivery of the module as a software library.

The vendor's internal development process guarantees that the correct version of module goes with its intended macOS version. For additional assurance, the module is digitally signed by vendor and it is verified during the integration into macOS.

This digital signature-based integrity protection during the delivery/integration process is not to be confused with the HMAC-SHA2-256 based integrity check performed by the module itself as its pre-operational self-test. No additional maintenance requirements apply.

Crypto Officer 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 10 - Non-Approved Services. If the device starts up successfully, then the module has passed all self-tests and is operating in the Approved mode.

A Crypto Officer Role Guide is provided by Apple which offers IT System Administrators with the necessary technical information to ensure FIPS 140-3 Compliance of macOS Monterey 12 systems. This guide walks the reader through the system's assertion of cryptographic module integrity and the steps necessary if module integrity requires remediation. A link to the Guide can be found on the Product security, validations, and guidance page.

Public Material – May be reproduced only in its original entirety (without revision).

12. Mitigation of Other Attacks

The module does not claim mitigation of other attacks.

Public Material – May be reproduced only in its original entirety (without revision).