Apple Inc.

Apple Secure Key Store Cryptographic Module v10.0
FIPS 140-2 Non-Proprietary Security Policy

Hardware versions
1.2, 2.0

Firmware version
SEPOS

Prepared for:
Apple Inc.
One Apple Park Way
Cupertino, CA 95014
www.apple.com

Prepared by:
atsec information security Corp.
9130 Jollyville Road, Suite 260
Austin, TX 78759
www.atsec.com

© 2021 Apple Inc.
This document may be reproduced and distributed only in its original entirety without revision

Table of Contents

© 2021 Apple Inc.

List of Tables

List of Figures

© 2021 Apple Inc.

1 Introduction

1.1 Purpose

This document is a non-proprietary Security Policy for the Apple Secure Key Store Cryptographic Module. It describes the module and the FIPS 140-2 cryptographic services it provides. This document also defines the FIPS 140-2 security rules for operating the module.

This document was prepared in fulfillment of the FIPS 140-2 requirements for cryptographic modules and is intended for security officers, developers, system administrators, and end-users.

FIPS 140-2 details the requirements of the Governments of the U.S. and Canada for cryptographic modules, aimed at the objective of protecting sensitive but unclassified information.

For more information on the FIPS 140-2 standard and validation program please refer to the NIST CMVP website [CMVP].

Throughout the document “Apple Secure Key Store Cryptographic Module v10.0” “cryptographic module”, “SKS” or “the module” are used interchangeably to refer to the Apple Secure Key Store Cryptographic Module. “Device OS” refers to the Operating System implemented by the hardware platforms tested under this revalidation: iOS 13, iPadOS 13, tvOS 13, watchOS 6, and TxFW10.15.

1.2 Document Organization / Copyright

This non-proprietary Security Policy document may be reproduced and distributed only in its original entirety without any revision, © 2021 Apple Inc.

1.3 External Resources / References

The Apple website (https://www.apple.com) contains information on the full line of products from Apple Inc. For a detailed overview of the operating systems Apple Secure Key Store (SKS) and the Secure Enclave Processor (SEP) and the associated security properties refer to [DevOS] and [SEC]. For details on the OS releases with their corresponding validated modules and Crypto Officer Role Guides “Product security certifications, validations, and guidance for iOS” refer to the Apple OS Security Guide [Guides].

Additional References are listed below.

CMVP	Cryptographic Module Validation Program https://csrc.nist.gov/projects/cryptographic-module-validation-program
CAVP	Cryptographic Algorithm Validation Program https://csrc.nist.gov/projects/cryptographic-algorithm-validation-program
FIPS 140-2	Federal Information Processing Standards Publication, “FIPS PUB 140-2 Security Requirements for Cryptographic Modules,” Issued May-25-2001, Effective 15-Nov-2001, https://csrc.nist.gov/projects/cryptographic-module-validation-program/standards
FIPS 140-2 IG	NIST, “Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program,” https://csrc.nist.gov/projects/cryptographic-module-validation-program/standards
FIPS 180-4	Federal Information Processing Standards Publication 180-4, March 2012, Secure Hash Standard (SHS)
FIPS 186-4	Federal Information Processing Standards Publication 186-4, July 2013, Digital Signature Standard (DSS)

FIPS 197	Federal Information Processing Standards Publication 197, November 26, 2001 Announcing the ADVANCED ENCRYPTION STANDARD (AES)
FIPS 198	Federal Information Processing Standards Publication 198, July, 2008 The Keyed-Hash Message Authentication Code (HMAC)
SP800-38 A	NIST Special Publication 800-38A, “Recommendation for Block Cipher Modes of Operation”, December 2001
SP800-38 D	NIST Special Publication 800-38D, “Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC”, November 2007
SP800-38 E	NIST Special Publication 800-38E, “ Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices”, January 2010
SP800-38 F	NIST Special Publication 800-38F, “Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping”, December 2012
SP800-56Ar3	NIST Special Publication 800-56Ar3 Revision 3, “Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography” April 2018
SP800-56B	NIST Special Publication 800-56B Revision 1, “Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography” September 2014
SP800-57P1	NIST Special Publication 800-57, “Recommendation for Key Management – Part 1: General (Revised),” July 2012
SP 800-90A	NIST Special Publication 800-90A, “Recommendation for Random Number Generation Using Deterministic Random Bit Generators,” January 2012
SP800-108	NIST Special Publication 800-108, “Recommendation for Key Derivation Using Pseudorandom Functions”, October 2009
SP800-132	NIST Special Publication 800-132, “Recommendation for Password-Based Key Derivation”, December 2010
SEC	Security Overview https://developer.apple.com/security/
DevOS	OS Technical Overview https://developer.apple.com/
Guides	User Guidance and CO Guidance: iOS: https://support.apple.com/HT202739 macOS: https://support.apple.com/HT201159 T2: https://support.apple.com/HT208675 watchOS: https://support.apple.com/HT208390 tvOS: https://support.apple.com/HT208389 SEP:SKS: https://support.apple.com/HT209632 iPadOS: https://support.apple.com/HT211006

© 2021 Apple Inc.

1.4 Acronyms

AES	Advanced Encryption Standard
API	Application Programming Interface
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining mode of operation
CKG	Cryptographic Key Generation
CMVP	Cryptographic Module Validation Program
CPE	Common Platform Enumeration
CPU	Central Processing Unit
CSP	Critical Security Parameter
DEK	Data Encryption Key
DRBG	Deterministic Random Bit Generator
ECB	Electronic Codebook mode of operation
ECC	Elliptic Curve Cryptography
EC Diffie-Hellman	DH based on Elliptic Curve Cryptography
ECDSA	DSA based on Elliptic Curve Cryptography
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FIFO	First-In First-Out
FIPS	Federal Information Processing Standard
GCM	Galois/Counter Mode
HMAC	Hash-based Message Authentication Code
HKDF	HMAC-based Extract-and-Expand Key Derivation Function (RFC5869)
IPC	Inter-Process Communication
KAT	Known Answer Test
KDF	Key Derivation Function
KEK	Key Encryption Key
KW	Key Wrapping
MDM	Mobile device management
MK	Master Key
NDRNG	Non-Deterministic Random Number Generator
NIST	National Institute of Standards and Technology
NVM	Non-Volatile Memory
OS	Operating System
PBKDF	Password-based Key Derivation Function (RFC 2898 PBKDF2)
PCT	Pair-wise Consistency Test
RCT	Repetition Count Test
REK	Root Encryption Key
RNG	Random Number Generator
SHS	Secure Hash Standard
SEP	Secure Enclave [Co-]Processor
SiP	System in Package
SKS	Secure Key Store
SoC	System on Chip
SSC	Shared Secret Computation
UID	Unique Identifier
XTS	XEX Tweakable Block Cipher with Ciphertext Stealing

© 2021 Apple Inc.

2 Cryptographic Module Specification

2.1 Module Description

The Apple Secure Key Store Cryptographic Module v10.0 is a hardware cryptographic module implemented as a sub-chip running on a single-chip processor.

The cryptographic services provided by the module are:

- data encryption / decryption
- generation of hash values
- key wrapping
- random number generation
- key generation
- key derivation

2.1.1 Module Validation Level

The module is intended to meet requirements of FIPS 140-2 security level 2 overall. The following table shows the security level for each of the eleven requirement areas of the validation.

FIPS 140-2 Security Requirement Area	Security Level
Cryptographic Module Specification	2
Cryptographic Module Ports and Interfaces	2
Roles, Services and Authentication	2
Finite State Model	2
Physical Security	2
Operational Environment	N/A
Cryptographic Key Management	2
EMI/EMC	2
Self-Tests	2
Design Assurance	2
Mitigation of Other Attacks	N/A

Table 1: Module Validation Level

2.1.2 Module Components

The module is a sub-chip hardware module which consists of both firmware and hardware components. The module's firmware operates within the SEPOS execution environment which is separate from the Device OS execution environment - the SEPOS execution environment is driven by its own CPU and uses isolated memory-. Both execution environments are physically separated on the SoC and thus execute independently of each other. The execution environment of the cryptographic module is SEPOS.

The two modules covered by this security policy are:

- Hardware DRBG version 1.2 with the operating system outlined in Table 2a
- Hardware DRBG version 2.0 with the operating system outlined in Table 2b

2.1.2.1 Firmware components

The firmware is bundled together with the underlying Device OS. The CPEs (Common Platform Enumeration) that identified the SEPOS version with associated hardware devices and Device OS are given in the Table 2a, 2b. Those CPEs are also found in the CAVP website for each of the CAVP certificates listed in Table 3. The SKS application linking with CoreCrypto is the

© 2021 Apple Inc.

cryptographic module. The firmware boundary is defined as the API offered by the module's mailbox interface to callers from the Device OS execution environment. SKS has an API layer that provides consistent interfaces to the supported services and therefore the supported cryptographic algorithms. In addition, the module provides Inter-process Communication (IPC) interfaces to other applications executing within the SEPOS execution environment.

2.1.2.2 Hardware components

The cryptographic module boundary includes a DRBG and a AES hardware component as part of the module which is integrated into the SoC and is reachable by the SEP execution environment. The module hardware version is 1.2 for hardware DRBG in Apple A8 and Apple 8X SoCs and hardware version 2.0 is found in all other SoCs.

2.1.3 Tested Platforms

The module has been tested on the following platforms which implement Apple A, S and T series processors.

Tested Platform	Operating System	SEPOS CPE
iPad mini 4 on Apple A8	SEPOS distributed with iPadOS 13	cpe:2.3:o:apple:sepOS:13:*:*:*:*:iPadOS:Apple_A_Series:iPadOS_13
iPad Air 2 on Apple A8X	SEPOS distributed with iPadOS 13	

Table 2a: Tested Platforms with hardware DRBG 1.2

Tested Platform	Operating System	SEPOS CPE
iPhone 6S Plus on Apple A9	SEPOS distributed with iOS 13	cpe:2.3:o:apple:sepOS:13:*:*:*:*:iOS:Apple_A_Series:iOS_13
iPhone 7 Plus on Apple A10 Fusion	SEPOS distributed with iOS 13	
iPhone 8 Plus on Apple A11 Bionic	SEPOS distributed with iOS 13	
iPhone Xs Max on Apple A12 Bionic	SEPOS distributed with iOS 13	
iPhone 11 Pro Max on Apple A13 Bionic	SEPOS distributed with iOS 13	
iPad (5th generation) on Apple A9	SEPOS distributed with iPadOS 13	cpe:2.3:o:apple:sepOS:13:*:*:*:*:iPadOS:Apple_A_Series:iPadOS_13
iPad Pro (9.7 inch) on Apple A9X	SEPOS distributed with iPadOS 13	
iPad (6th generation) on Apple A10 Fusion	SEPOS distributed with iPadOS 13	
iPad Pro (12.9-inch, 2nd generation) on Apple A10X Fusion	SEPOS distributed with iPadOS 13	
iPad mini (5th generation) on Apple A12 Bionic	SEPOS distributed with iPadOS 13	
iPad Pro (12.9-inch, 3rd generation) on Apple A12X Bionic	SEPOS distributed with iPadOS 13	
Apple TV 4K on Apple A10X Fusion	SEPOS distributed with tvOS 13	cpe:2.3:o:apple:sepOS:13:*:*:*:*:tvOS:Apple_A_Series:tvOS_13
Apple Watch Series 1 on Apple S1P	SEPOS distributed with watchOS 6	cpe:2.3:o:apple:sepOS:6:*:*:*:*:watchOS:Apple_S_Series:watchOS_6
Apple Watch Series 3 on Apple S3	SEPOS distributed with watchOS 6	
Apple Watch Series 4 on Apple S4	SEPOS distributed with watchOS 6	
Apple Watch Series 5 on Apple S5	SEPOS distributed with watchOS 6	

© 2021 Apple Inc.

Apple T2[1]	SEPOS distributed with TxFW 10.15	cpe:2.3:o:apple:sepOS:10.15:*:*:*:*:macOS:Apple_T_Series:TxFW_10.15

Table 2b: Tested Platforms with hardware DRBG 2.0

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

for iOS13:

- iPhone 6s and iPhone SE with an Apple A9
- iPhone 7 with an Apple A10 Fusion
- iPhone 8 and iPhone X with an Apple A11 Bionic
- iPhone Xr and iPhone Xs with an Apple A12 Bionic
- iPhone SE (2nd generation), iPhone 11 and iPhone 11 Pro with an Apple A13 Bionic

for iPadOS 13:

- iPad Pro (12.9) with an Apple A9X
- iPad (7th generation) with an Apple A10 Fusion
- iPad Pro (10.5-inch) with an Apple A10X Fusion
- iPad Air (3rd generation) with an Apple A12 Bionic
- iPad Pro (11-inch) with an Apple A12X Bionic
- iPad Pro 11” (2nd generation) and iPad Pro 12.9" (4th generation) with an Apple A12Z Bionic

Note: 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 certificate (IG G.5).

2.2 Mode of Operation

The Apple Secure Key Store Cryptographic 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 Table 4 will cause the module to assume the non-Approved mode of operation.

The module transitions back into FIPS mode immediately when invoking one of the approved ciphers as all keys and Critical Security Parameters (CSPs) handled by the module are exclusively assigned to different services which are bound to either approved or non-approved ciphers. There are no keys and CSPs shared between approved or non- approved functions as this is technically impossible due to the fact that the non-approved functions use key types that are cryptographically unusable by approved functions. A re-invocation of the self-tests or integrity tests is not required.

Even when using this FIPS 140-2 non-approved mode, the module ensures that the self-tests are always performed during initialization time of the module.

The module contains multiple implementations of the same cipher as listed below. If multiple implementations of the same cipher are present, the module selects the most appropriate cipher based on internal heuristics.

---

[1] The following Mac computers contain an embedded Apple T2 Security Chip:
iMac Pro
Mac Pro introduced in 2019
Mac mini introduced in 2018
MacBook Air introduced in 2018 or later
MacBook Pro introduced in 2018 or later

© 2021 Apple Inc.

2.2.1 Approved or Allowed Security Functions

The Approved security functions are listed in Table 3. The Algorithm Certificate Numbers are obtained from NIST CAVP for successful validation testing of the cryptographic algorithm implementations for the modules that run on the hardware platforms (Tables 2a and 2b).

For the current standards, test requirements, and special abbreviations used in the following table, please refer to [CAVP].

Cryptographic Function	Standards / Algorithm	Modes/Options	Algorithm Certificate Number
Random Number Generation	[SP 800-90A] DRBG	Hardware DRBG (CTR_DRBG) Modes: AES-256 No Derivation Function Prediction Resistance Enabled	2014, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2028, 2029, C323, C324, C331, A501 (trng)
Symmetric Encryption and Decryption	[FIPS 197] AES SP 800-38A SP 800-38C SP 800-38D SP 800-38E SP 800-38F	Key Lengths: 128, 192, 256 (bits) Modes: CBC, CFB8, CCM, CTR, ECB, KW, CFB128, OFB; XTS (*key length: 128 and 256-bits only*)	A498 (c_asm) A497 (c_ltc)
		Key Lengths: 128, 192, 256 (bits) Mode: CBC	A499 (c_glad)
		Key Lengths: 128, 192, 256 (bits) Modes: CBC, OFB CFB128, ECB; XTS(*key length: 128 and 256-bits only*)	A494 (asm_arm)
		Key Lengths: 128, 192, 256 (bits) Modes: CCM CTR ECB	A496 (vng_asm)
		SKG AES Hardware Implementation Key Lengths: 128, 256 (bits) Modes: ECB CBC	C312, C313, C314, C315, C317, C318, C319, C320, C322, C325, C326, C330, C358, A510 (skg)
		Hardware AES Implementation serving DRBG Key Lengths: 256 (bits) Mode: ECB	5261, 5270, 5271, 5272, 5273, 5274, 5275, 5276, 5278, 5279, C323, C324, C331, A501 (trng)
Digital Signature and Asymmetric Key Generation	[FIPS 186-4] ECDSA ANSI X9.62	Public Key Generation: P-224, P-256, P-384, P-521 Public Key Validation (PKV): P-224, P-256, P-384, P-521 Signature Generation: P-224, P-256, P-384, P-521 Signature Verification: P-224, P-256, P-384, P-521	A495 (vng_ltc)

© 2021 Apple Inc.

Cryptographic Function	Standards / Algorithm	Modes/Options	Algorithm Certificate Number
Message Digest	[FIPS 180-4] SHS	Modes SHA-1 | SHA-384 SHA-224 | SHA-512 SHA-256	A497 (c_ltc) A495 (vng_ltc)
		Mode SHA-256	A500[2] (vng_neon)
KAS-SSC using EC keys generated by the module	Section 5.7.1.2 [SP800-56Ar3]	EC Diffie-Hellman (ECC CDH) primitive Curves:P-224, P-256, P-384, P-521	Vendor affirmed
Keyed Hash	[FIPS 198] HMAC	Key size: 112 bits or greater Modes HMAC-SHA-1 | HMAC-SHA-384 HMAC-SHA-224 | HMAC-SHA-512 HMAC-SHA-256	A497 (c_ltc) A495 (vng_ltc)
		Key size: 112 bits or greater Mode: HMAC-SHA-256	A500[2] (vng_neon)
KTS	[SP800-38F]	Key Lengths: 128, 192, 256 (bits) Mode: AES-KW	A498 (c_asm) A497 (c_ltc)
Key Derivation	[SP 800-132] PBKDF	Password Based Key Derivation using HMAC with SHA-1, SHA2-224, SHA2-256, SHA2-384 or SHA2-512	Vendor Affirmed[3]
CKG	[SP800-133]	CTR-DRBG (with AES-256 underlined cipher) for symmetric key generation and FIPS 186-4 for EC key generation	Vendor Affirmed
NDRNG	Random number generation	N/A	Non-Approved, but Allowed; provided by the underlying operational environment

Table 3: Approved, Allowed or Vendor Affirmed Security Functions

2.2.2 Non-Approved Security Functions

Cryptographic Function	Usage / Description	Caveat
ECDH shared secret computation using EC keys generated outside the module	Shared secret computation using P curves with keys generated outside of the module	Non-Approved
ECDH Key Agreement	Key Agreement using curve 25519	Non-Approved

[2] The S1P and S3 from the armv7 processor family do not implement vng_neon and do not have the A500 ACVT certificate.

[3] PBKDF is vendor affirmed as it owns the ACVT certificate A497 (c_ltc) but does not implement a self-test.

© 2021 Apple Inc.

Cryptographic Function	Usage / Description	Caveat
EDDSA	Digital signature generation using Ed25519	Non-Approved
RFC5869 Key Derivation	HMAC based Key Derivation Function	Non-Approved
ANSI X9.63 Key Derivation	Hash based KDF	Non-Approved
AES-GCM	Encryption and Decryption	Non-Approved: the IV generation is not compliant to FIPS 140-2 IG A.5 (A496, A497, A498 certs.)

Table 4: Non-Approved or Non-Compliant Functions

2.3 Cryptographic Module Boundary

The module is defined as a sub-chip hardware module with a single-chip embodiment. The physical boundary of the module is the perimeter of the package (i.e. SoC or SiP).

The module's logical boundary is the "secure key store" firmware application (i.e. SKS; blue outline) together with the DRBG and AES hardware engines. The DRBG and AES hardware engines and the associated secure key store firmware represent the sub-chip cryptographic subsystem boundary of a single-chip hardware module. The module's logical and physical boundaries are depicted in the logical block diagram given in Figure 1.

2.3.1 Block Diagram

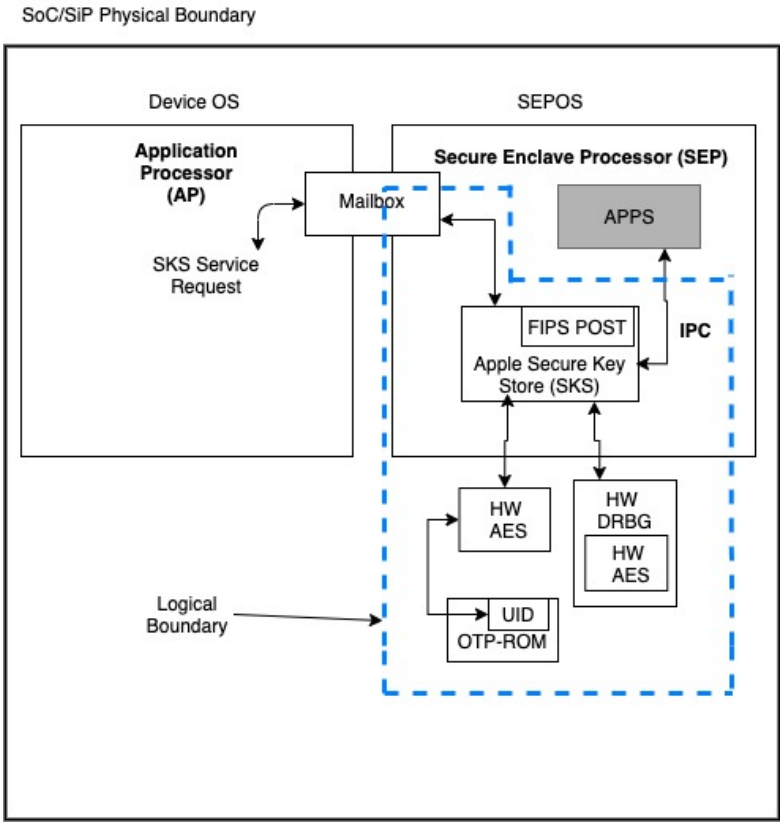

Figure 1: Cryptographic Module Block Diagram

© 2021 Apple Inc.

3 Cryptographic Module Ports and Interfaces

The underlying logical interfaces of the module are the mailbox interface used between the module and the Device OS kernel, and the IPC channel to other SEP applications:

- Data input and Data output are provided through the memory used for mailbox and IPC.
- Control inputs which control the mode of the module are provided through the mailbox by the Device OS' kernel and to applications located within the SEPOS execution environment through IPC. The HMAC control value is provided as part of the executable image implementing the module.
- Status output is provided in return codes and through messages returned via the mailbox or the IPC. Documentation for each service invocation lists possible return codes.

The module's logical interfaces used for input data and control information are logically disconnected from the logical paths used for the output of data and status information by virtue of the module's API. The module's API distinguishes all output data from key/CSP information. The module is optimized for use with the SEP coprocessor and does not contain any terminating assertions or exceptions. It is implemented as a hardware module with hardware engines DRBG and AES implemented as part of the SoC and accessible to the SEP execution environment. Any internal error detected by the module is reflected back to the caller with an appropriate return code. The calling Device OS application must examine the return code and act accordingly.

The function executing FIPS 140-2 module self-tests does not return an error code but causes the system to shut down if any self-test fails – see Section 9.

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 exception conditions in a FIPS 140-2 appropriate manner.

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

Cryptographic bypass capability is not supported by the module.

© 2021 Apple Inc.

4 Roles, Services and Authentication

This section defines the roles, services and authentication mechanisms and methods with respect to the applicable FIPS 140-2 requirements.

4.1 Roles

The module supports two authorized roles: A Crypto Officer Role and a User Role. No support is provided for a Maintenance operator. The module does not implement a bypass mode nor concurrent operators.

When a device is delivered, the Crypto Officer is responsible of initializing the module i.e. configure the device by properly setting up key registers for storage of keys/CSPs and the FIFOs that will be later used by the software applications. The Crypto Officer is implicitly assumed. The User can perform services from Table 6a and 6b only after the Crypto Officer takes possession by initializing it, thus creating data to be protected is generated.

The User of the module are software applications that assume the User Role when requesting any cryptographic services provided by the module.

A device can be completely factory reset which implies that all data are cryptographically destroyed by the module. Factory reset returns the module to an uninitialized state as an initially shipped device and thus the aforementioned statement about taking possession applies again.

Role	General Responsibilities and Services (details see below)
User	Utilization of cryptographic services of the module.
Crypto Officer (CO)	Initialization and configuration of the module (e.g. reboot, self-test).

Table 5: Roles

4.2 Operator Authentication

Within the constraints of FIPS 140-2 level 2, the module implements a role-based authentication mechanism for operator authentication.

The module implements passcode-based authentication in the following way: When the User requests a crypto service from the module, it must provide the passcode and a reference to the user keybag that is stored encrypted under SP800-38F AES Key Wrapping (AES-KW) within SKS. The module uses PBKDF to derive an AES key from the Operator provided passcode. The derived AES key is then used by the module's SP800-38F AES Key Unwrapping function (i.e. AES-KW-AD[4]) to decrypt the reference user keybag and to verify the authenticity of the decrypted key. As AES-KW is an authentication cipher, the decryption operation will only succeed without an authentication error. This implies that the user provided the correct passcode to derive the correct AES key for AES Key Unwrapping. Any other passcode will derive a different AES key which will result in a wrong decrypted user key that fails the authentication check. If the user keybag can be successfully unwrapped, the user is authenticated to the module and the requested crypto service will then be proceeded with the unwrapped user key. The failure of unwrapping user keybag is also a user authentication failure and the Operator will be denied access to the module.

The module does not maintain authenticated sessions upon power cycling. All authentication data is obscured during data entry.

[4] Section 6.2 SP800-38F, Algorithm 4: KW-AD(C)

© 2021 Apple Inc.

4.2.1 Strength of Authentication

The minimum length of the passcode to be configured is 7 digits, each with 10 different possibilities for usage. The chance of a random attempt falsely succeeding is $1/10^7$ which is less than 1/1,000,000 as required by FIPS 140-2.

Furthermore, the module implements delays between passcode attempts. After the fifth failed attempt the module requires 1 min delay. This means that an attacker has the probability of successfully guessing the password in one minute as $5/10^7$ which is far less than the requirement of 1/100,000

4.3 Services

The module provides services to authorized operators of either the User or Crypto Officer Roles according to the applicable FIPS 140-2 security requirements.

Table 6a contains the cryptographic services employed by the module in the Approved and Table 6b contains the cryptographic services employed by the module in the non-Approved mode. For each available service it lists, the associated role, the Critical Security Parameters (CSPs) and cryptographic keys involved, and the type(s) of access to the CSPs and cryptographic keys.

CSPs contain security-related information (for example, secret and private cryptographic keys) whose disclosure or modification can compromise the main security objective of the module, namely the protection of sensitive information.

The access types are denoted as follows:

- 'R': the item is read or referenced by the service
- 'W': the item is written or updated by the service
- 'Z': the persistent item is zeroized by the service

Service Number	Service	Roles		CSPs & crypto keys	Access Type
		User	CO		
service_1	Class D File System Services Encryption and Decryption of: - Class D key - File System keys	X		Key wrapping: UID, AES Key used to wrap Class D Key, Class D Key File system keys: DEK Storage controller key: KEK	R W
service_2	User Keybag Services Encryption and Decryption of: - Device OS file system object storage keys - User Keybag - User File system keys	X		Keybag wrapping: AES Key shared with NVM Storage Controller Keybag content: KEK	R W

© 2021 Apple Inc.

Service Number	Service	Roles		CSPs & crypto keys	Access Type
		User	CO		
service_3	Device Keybag Services ECDSA signature generation and signature verification Encryption and Decryption of: - Trusted device communication keys - Device keybag - Device-specific keys - Keychain keys	X		Keybag wrapping: UID, AES Keys as part of module-managed keybags Keybag content: KEK Keychain keys: DEK Sign/verify key: ECDSA Private Key	R W
service_4	Backup Keybag Services Encryption and Decryption of: - Backup keys - Backup keybag - Backup data	X		Keybag wrapping: AES Key used to wrap Backup Keybag, PBKDF password for Backup Keybag, PBKDF Salt for Backup Keybag Keybag content: KEK Backup keys: DEK	R W
service_5	Escrow Keybag Services Device system update Authentication Encryption and Decryption of: - MDM keys - Escrow Keybag - Escrow file system keys	X		Keybag wrapping: AES Key used to wrap Escrow Keybag Keybag content: KEK from User Keybag, DEK	R W
service_6	iCloud Keybag Services Encryption and Decryption of: - iCloud data keys - iCloud Keybag - iCloud user data	X		Keybag wrapping: REK derived from UID Keybag content: DEK	
service_7	Create REK	X		UID PBKDF Password PBKDF Salt for REK DRBG internal state Entropy input string REK KEK as AES key used to wrap REK	R W

© 2021 Apple Inc.

Service Number	Service	Roles		CSPs & crypto keys	Access Type
		User	CO		
service_8	Update REK	X		Old / new PBKDF Password PBKDF Salt for REK Old / new REK derived from PBKDF Password DRBG internal state Entropy input string UID	R W
service_9	Generate Ref-Keys	X		EC Key Pair Password PBKDF Salt for EC Private Key Encryption Key	R W
service_10	Generate Shared Secret using EC keys generated by the module	X		EC Key Pair Password PBKDF Salt for EC Private Key Encryption Key	R W
service_11	Erase all content (Factory Reset)		X	All Keys and CSPs except UID (see Section 7.5)	Z
service_12	Reboot that implies Self-test		X	None	N/A
service_13	Show Status		X	None	N/A

Table 6a: Approved Services in Approved Mode

Service	Roles	
	User	CO
Generate Shared Secret using EC keys generated outside of the module	X	
Ed 25519 Digital signature generation	X	
Hash based KDF based on ANSI X9.63	X	
EC Diffie-Hellman Key Agreement using curve 25519	X	
RFC 5869 based HKDF	X	
AES-GCM Encryption and Decryption	X	

Table 6b: Non-Approved Services in Non-Approved Mode

© 2021 Apple Inc.

5 Physical Security

The Apple Secure Key Store Cryptographic Module v10.0 is a hardware module implemented as a sub-chip and is identified as a single-chip embodiment. The physical boundary is considered to be each SoC listed in Tables 2a and 2b. The module conforms to the Level 2 requirements for physical security. The physical components that comprise the module are of production grade components with industry standard passivation applied. In addition, the module is covered with a tamper-evident coating that deters direct observation, probing, or manipulation of the single-chip.

© 2021 Apple Inc.

6 Operational Environment

The Apple Secure Key Store Cryptographic Module v10.0 operates in a limited operational environment per FIPS 140-2 level 2 specifications. The module operates within the SEPOS execution environment which is separate from the Device OS execution environment. The SEP operating system provides memory isolation between all applications executing on it. The Device OS is unable to access the module's memory or observe the module's operation.

© 2021 Apple Inc.

7 Cryptographic Key Management

Table 7 summarizes the CSPs that are used by the cryptographic services implemented in the module. The rightmost column maps to the service number listed in Table 6a.

CSPs		Generation	Entry and Output	Zeroization	Service #s in Table 6a
1	Device-specific hardware key (UID)	A8, A8X: N/A: Entered during manufacturing process Other SoCs: Generated using module's DRBG during manufacturing process	A8, A8X: Entry during manufacturing process Other SoCs: N/A – Generated using module's DRBG during manufacturing process and is never output.	N/A	1,3,7,8
2	Root Encryption Key (REK)	Derived from passcode using PBKDF2 and entanglement with UID	N/A – Generated inside the module and is never output.	Zeroized when freeing the secure memory.	6,7,8,11
3	PBKDF Password for the REK	N/A	Entered by calling application.	Zeroized when freeing the secure memory.	7,8,11
4	Backup Keybag Key	Derived from passcode using PBKDF2	N/A – Generated inside the module and is never output.	Zeroized when freeing the secure memory.	4,11
5	PBKDF Password for the Backup Keybag	N/A	Entered by calling application.	Zeroized when freeing the secure memory.	4,11
6	Escrow Keybag and the class D key encryption key	Derived from UID	N/A – Generated inside the module and is never output	Zeroized when freeing the secure memory.	1,5,11
7	Module-managed keybags key	Symmetric key generation services of the module	Entered encrypted using AES-256 KW. Output encrypted using AES-256 KW	Non-volatile store: cryptographically zeroized when overwriting the KEK Volatile store: zeroized when freeing the secure memory.	3,11
8	File system object DEK	Symmetric key generation services of the module	Entered encrypted using AES-256 KW. Output encrypted using AES-256 KW	Non-volatile store: cryptographically zeroized when overwriting the KEK Volatile store: zeroized when	1,3,4, 5,6,11

Passcode Key

用户登录输入 Passcode

用户备份时系统生成并提示的备份密码

Bag1

EMF Key

© 2021 Apple Inc.

CSPs		Generation	Entry and Output	Zeroization	Service #s in Table 6a
				freeing the secure memory.	
9	NVM storage controller shared key	Symmetric key generation service of the module	Output to the storage controller of the SoC	Zeroized when volatile memory loses power during power down	2,11
10	Keychain ECDSA Private Keys	asymmetric key generation services of the module following FIPS 186-4	N/A – Generated inside the module and is never output.	Zeroized when freeing the secure memory	3,11
11	Entropy input string	Obtained from NDRNG	N/A	Zeroized when freeing the secure memory	7,8
12	DRBG internal state: V value, key and seed material	Updated during DRBG initialization	N/A	Zeroized when freeing the secure memory	7,8,11
13	PBKDF Salt for REK	Symmetric key generation services of the module	Entered in wrapped form using AES-256 KW. Output in wrapped form using AES-256 KW	Zeroized when freeing the secure memory	7,8,11
14	PBKDF Salt for Backup Keybag	Symmetric key generation services of the module	Entered in wrapped form using AES-256 KW. Output in wrapped form using AES-256 KW	Zeroized when freeing the secure memory	4,11
15	Class D Key	Symmetric key generation services of the module	Entered encrypted using AES-256 KW. Output encrypted using AES-256 KW	Non-volatile store: cryptographically zeroized when caller requests new key Volatile store: zeroized when freeing the secure memory.	1,11
16	REK wrapping key	Symmetric key generation services of the module	Entered in plaintext by calling application within physical the boundary Output to in plaintext calling application within the physical boundary	Zeroized when freeing the secure memory.	7,11

AES引擎的临时封装密钥

钥匙包存储的随机盐值

Dkey

KEK？？？仅用于MacOS

© 2021 Apple Inc.

CSPs		Generation	Entry and Output	Zeroization	Service #s in Table 6a
17	Ref-Key	asymmetric key generation services of the module following FIPS 186-4	N/A – Generated inside the module and never output.	Zeroized when freeing the secure memory	9,10
18	Shared Secret	Generated using EC Diffie-Hellman SSC (800-56Ar3)	sent to calling application	Zeroized when freeing the secure memory	9,10
19	HMAC Key	Symmetric key generation services of the module	Entered by calling application	Zeroized when freeing the secure memory	11,12

密钥包存储的HMAC校验值

Table 7: Life Cycle of Critical Security Parameters

The following sections define the key management features available through the Apple Secure Key Store Cryptographic Module v10.0.

7.1 Random Number Generation

A FIPS 140-2 approved deterministic random bit generator based on AES as specified in NIST SP 800-90A is used. The Approved DRBG used for random number generation is a CTR_DRBG using AES-256 without derivation function and with prediction resistance. The deterministic random bit generator is seeded by an internal hardware noise source consisting of 8 ring oscillators (A8 and A8X) or 24 ring oscillators (A9 and later A series, S series and T2 SoCs). The hardware noise source provides 256-bits of security strength in seeding and reseeding the module approved DRBGs.

7.2 Key / CSP Generation

The following approved key generation methods are used by the module.

- The Approved DRBG specified in section 7.1 is used to generate secret symmetric keys for the AES algorithm.
- The Approved DRBG specified in section 7.1 is used to generate the random values used in the key generation of asymmetric keys. Asymmetric keys for the ECDSA / ECDH algorithm are generated using FIPS 186-4.
- The module provides a key generation service for symmetric ciphers and HMAC keys as well as for asymmetric keys. The key generation service is compliant with SP800-133 (CKG- vendor affirmed) that requires the symmetric key is a XOR of the DRBG output with a value V. In case of the module, the value V is a string of zeros which implies that the key is unmodified from the output of the DRBG. A seed (i.e. the random value) used in asymmetric key generation is obtained from DRBG.

It is not possible for the module to output information during the key generating process.

7.3 Key / CSP Establishment

The module provides the following key establishment services.

- Key wrapping compliant to [SP 800-38F] using AES key wrapping mode (AES-KW with Certs. #A497 and #A498). The key wrapping service provides between 128 and 256-bit of encryption strength.

© 2021 Apple Inc.

- Key derivation services in the Approved mode through the PBKDF2 algorithm. The module supports option 1a from Section 5.4 of SP 800-132, whereby the Master Key (MK) is used directly as the Data Protection Key (DPK). Keys derived from passwords may only be used for data at rest. The caller shall observe all requirements and should consider all recommendations specified in SP800-132 with respect to the strength of the generated key, including the quality of the password (passcode strength is detailed in section 4.2.1), the quality of the salt as well as the number of iterations.
- Key agreement scheme based on SP800-56Ar3 without KDF. The module implements ECC primitive calculation based on Section 5.7.1.2 ECC CDH Primitive of SP 800-56Ar3. The module itself does not include implementation of key derivation function. In the approved mode, the module provides EC Diffie-Hellman shared secret computation with curves P-224, P-256, P-384 or P-521 providing between 112 and 256-bit equivalent security strength.

7.4 Key / CSP Entry and Output

The module does not support entry or output of cryptographic keys beyond the physical boundary of the SoC. Within the physical boundary, all secret keys and CSPs are entered into, or output from the Apple Secure Key Store Cryptographic Module in wrapped form using AES-KW (SP800-38F). The exception is the User's password which is entered into the module in the clear. All keys and CSPs entered into the module are electronically entered Key / CSP Storage

The Apple Secure Key Store Cryptographic Module v10.0 considers all keys in memory to be ephemeral.

The keys managed by the module in keybags are stored in non-volatile memory by the Device OS. The keybag is wrapped with AES-256 KW followed by an export to the Device OS for permanent storage. After a power-up, the module imports the wrapped keybags from Device OS and unwraps them.

The module protects all keys at runtime, secret or private, and CSPs through the memory protection mechanisms provided by SEPOS. No process can read the memory of another process.

7.5 Key / CSP Zeroization

Cleartext keys and CSPs are zeroized immediately after their usage is completed or when the device is powered down. Additionally, the user can zeroize the entire device directly (locally) or remotely, returning it to the original factory settings.

The exception is the key called the device UID which is stored in a specially protected hardware component. The UID key is programmed during manufacturing process and cannot be directly read or written by any software/firmware. It can only be used for an AES encryption or decryption operation. The UID is used to wrap the file system Class D key or keys that are intended to be bound to the current device. For wrapping the remaining Class keys, a key is derived using the KDF from the UID and a key derived from the User's password. Therefore, the UID is required for the lifetime of the device. The UID stored in hardware exists as "blown fuses" and cannot be zeroized.

© 2021 Apple Inc.

8 Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)

The cryptographic module hardware component cannot be certified by the FCC as it is not a standalone device. It is a sub-chip embedded in the devices listed in Section 2.1.3. However, the tested platforms containing the cryptographic module are conformant to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Appliances, Class A (business use).

© 2021 Apple Inc.

9 Self-Tests

FIPS 140-2 requires that the module perform self-tests to ensure the integrity of the module and the correctness of the cryptographic functionality at start up. In addition, the noise source feeding the random bit generator requires continuous verification. The module runs all the required self-tests which are invoked automatically when module is powered on.

The occurrence of any self-test error triggers an immediate shutdown of the module, preventing any operation.

All self-tests performed by the module are listed and described in this section.

9.1 Power-Up Tests

The following tests are performed each time the Apple Secure Key Store Cryptographic Module v10.0 starts and must be completed successfully for the module to operate. The invocation of each individual self-tests is a synchronous operation which must be finished before the boot can continue. While executing the Power-Up Tests, the data output interface is inhibited. If any of the following power-up tests fail, the module enters into an Error state whereby data output interface is inhibited, the status of the test is output to the status output interface and the device powers itself off. To rerun the self-tests on demand, the user must reboot the module.

9.1.1 Cryptographic Algorithm Tests

Algorithm	Modes	Test
AES Implementation selected by the module for the corresponding environment AES-128	ECB, CBC, CCM	KAT[5] Separate encryption / decryption operations are performed
AES SKG Hardware Implementation AES-128	ECB, CBC	KAT Separate encryption / decryption operations are performed
Hardware DRBG (CTR_DRBG)	N/A	KAT
HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512	N/A	KAT (which also covers KAT for SHA)
ECDSA	Signature Generation Signature Verification	PCT
EC Diffie-Hellman “Z” computation	N/A	KAT

Table 8: Cryptographic Algorithm Tests

9.1.2 Firmware Integrity Tests

A firmware integrity test is performed on the runtime image of the Apple Secure Key Store Cryptographic Module v10.0. The module’s HMAC-SHA-256 is used as an Approved algorithm for the integrity test. If the test fails, then the device powers itself off.

9.1.3 Critical Function Tests

No other critical function test is performed on power up.

[5] Self-test is subject to the "selector" approach for the different implementations of AES (cf. section 2.2).

© 2021 Apple Inc.

9.2 Conditional Tests

The following sections describe the conditional tests supported by the Apple Secure Key Store Cryptographic Module v10.0.

9.2.1 Repetition Count Test

The Apple Secure Key Store Cryptographic Module v10.0 performs a continuous random number generator test by way of a Repetition Count Test (RCT) on the NDRNG, whenever the DRBG is seeded or reseeded.

9.2.2 Pair-wise Consistency Test

The Apple Secure Key Store Cryptographic Module v10.0 performs a pair-wise consistency tests on asymmetric keys generated for ECDSA cipher.

9.2.3 SP 800-90A Health Tests

The Apple Secure Key Store Cryptographic Module v10.0 performs the health tests as specified in section 11.3 of SP 800-90A.

9.2.4 Critical Function Test

No other critical function test is performed conditionally.

© 2021 Apple Inc.

10 Design Assurance

10.1 Configuration Management

Apple manages and records source code and associated documentation files by using the revision control system called “Git”.

The Apple module hardware data, which includes descriptions, parts data, part types, bills of materials, manufacturers, changes, history, and documentation are managed and recorded. Additionally, configuration management is provided for the module’s FIPS documentation.

The following naming/numbering convention for documentation is applied.

<evaluation>_<module>_<os>_<mode>_<doc name>_<doc version (#.#)>

Example: FIPS_SEP_SECPOL_2.0

Document management utilities provide access control, versioning, and logging. Access to the Git repository (source tree) is granted or denied by the server administrator in accordance with company and team policy.

10.2 Delivery and Operation

The module’s firmware with the SEPOS is delivered as part of the Device OS image. The Approved mode is configured by default and can only be transitioned into the non-Approved mode by calling one of the non-Approved algorithms listed in Table 4. The module returns to the Approved mode once an Approved function (Table 3) is called (see also section 2.2).

10.3 Development

The Apple crypto module (like any other Apple software) undergoes frequent builds utilizing a “train” philosophy. Source code is submitted to the Build and Integration group (B & I). Integration can only take place after the corecrypto project successfully passes internal integration and system tests on all platforms. B & I group builds, integrates, system tests on all the platforms and checks on the operating systems and apps that they produce. Copies of older versions are archived offsite in underground granite vaults.

10.4 Guidance

The following guidance items are to be used for assistance in maintaining the module’s validated status while in use.

10.4.1 Cryptographic 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 algorithms listed in Table 4. If the device starts up successfully then the module has passed all self-tests and is operating in the Approved mode.

10.4.2 User Guidance

As above, 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 algorithms listed in Table 4. If the device starts up successfully then the module has passed all self-tests and is operating in the Approved mode.

10.4.2.1 Module Usage Considerations

A user of the module must consider the following requirements and restrictions when using the module:

© 2021 Apple Inc.

- As specified in SP800-38E, the AES algorithm in XTS mode is designed for the cryptographic protection of data on storage devices. It can only be used for encryption of data at rest.
- To meet the requirement stated in IG A.9, the module implements a check to ensure that the two AES keys used in AES XTS mode are not identical.
- Keys derived from derived passwords using PBKDF2 may only be used in storage applications.
- In order to use the ECDH shared secret computation [800-56Arev3] service in compliance with IG D.1, the caller shall adhere to the following requirements:
    - The shared secret computation service should only be use with keys generated by the module.
    - The keys shall be generated as closet to its time of use as possible.
    - The key should only be used for one transaction.
    - The key shall be destroyed when no longer needed.

© 2021 Apple Inc.

11 Mitigation of Other Attacks

The module has not been designed to mitigate any specific attacks outside the scope of FIPS 140-2 requirements.

© 2021 Apple Inc.