Crypto Officer Role Guide for FIPS 140-2 Compliance

iOS 9

Contents

Overview

In highly regulated industries, IT System Administrators and Crypto Officers are frequently required to ensure deployed systems are correctly using FIPS 140-2 Validated Cryptographic Modules. The two Apple Cryptographic Modules in iOS 9 achieved **FIPS 140-2 Level 1 Conformance Validation** under the Cryptographic Module Validation Program (CMVP) – a joint American and Canadian security accreditation program for cryptographic modules.

These two modules are identified under the CMVP with the module names of: a) "**Apple iOS CoreCrypto Module v6.0"** and b) "**Apple iOS CoreCrypto Kernel Module v6.0**." The **CoreCrypto Module** is available to developers for Applications and Services running in User Space. The **CoreCrypto Kernel Module** is used only by the iOS Kernel.

Within this and other Apple documents, those modules are also referred to with the name of **"Apple FIPS Cryptographic Module v6.0."**

Apple iOS CoreCrypto Module v6.0 Validation Certificate #2594

http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2016.htm#2594

Apple iOS CoreCrypto Kernel Module v6.0 Validation Certificate #2609

http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2016.htm#2609

All Apple Validated Crypto Modules can be found under CMVP's FIPS 140-2 Vendor List here - http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401vend.htm

This Crypto Officer Role Guide provides IT System Administrators with the necessary technical information to ensure FIPS 140-2 compliance of iOS 9 systems. This guide walks the reader through the system's assertion of cryptographic module integrity and the steps necessary if module integrity requires remediation.

Compliant Applications and Services

Compliancy Requirements on Crypto Officers are not limited to the use of products containing a validated cryptographic module, but extend to their attestation that applications and services in use are FIPS 140-2 Compliant. Compliance is defined by both the use of a FIPS 140-2 validated module and the proper use of FIPS-Approved Algorithms. A cryptographic module may contain additional algorithms that are not FIPS-Approved and if used, would indicate a Non-FIPS Compliant condition. A FIPS 140-2 Level 1 Conformance Validation does not require the cryptographic module ensures applications and services only use FIPS-Approved algorithms.

Apple

A high-level, non-exhaustive list of Apple applications and services that are FIPS 140-2 Compliant in iOS 9 would include the following:

Services

Data Protection, Hardware Encryption, HTTPS, Keychain Services, S/MIME, TLS/SSL, VPN, and 802.1X.

Applications

App Store, iTunes Store, Calendar, Contacts, FaceTime, Messages, Mail, Safari, and Software Update.

Developer and Crypto Officer Resources

There are resources available to developers providing guidance on cryptographic services and API documentation for iOS 9. Developers should refer to these resources to ensure their products and services are FIPS 140-2 Compliant on iOS 9.

Apple iOS CoreCrypto Module, v6.0 FIPS 140-2 Non-Proprietary Security Policy

http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140sp/140sp2594.pdf

Apple iOS CoreCrypto Kernel Module, v6.0 FIPS 140-2 Non-Proprietary Security Policy

http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140sp/140sp2609.pdf

iOS product security: Validations and Guidance

https://support.apple.com/en-us/HT202739

iOS Security Guide

The iOS Security Guide target audience is enterprise IT and provides both an overview and low-level details about the security services, processes and cryptographic algorithms in use throughout various parts of the platform.

https://www.apple.com/business/docs/iOS_Security_Guide.pdf

Security Overview

https://developer.apple.com/library/ios/documentation/Security/Conceptual/Security_Overview/Introduction/Introduction.html

Cryptographic Services Guide

https://developer.apple.com/library/ios/documentation/Security/Conceptual/Security_Overview/CryptographicServices/CryptographicServices.html

Certificate, Key, and Trust Services Programming Guide

https://developer.apple.com/library/ios/documentation/Security/Conceptual/CertKeyTrustProgGuide/

Compliant Platforms

Compliant platforms are all supported Apple systems running iOS 9. During the validation process for FIPS 140-2 Conformance, the cryptographic modules are put through operational testing environments on supported platforms and noted on the issued certificate. The **CoreCrypto** and **CoreCrypto Kernel** modules were validated under the following operational testing environments:

Module: **Apple iOS CoreCrypto Module v6.0**

Platforms: **A5** with iOS 9 (User Space)
A5X with iOS 9 (User Space
A6 with iOS 9 (User Space)
A6X with iOS 9 (User Space)
A7 with iOS 9 (User Space) 32-/64-bit
A8 with iOS 9 (User Space) 32-/64-bit
A8X with iOS 9 (User Space) 32-/64-bit
A9 with iOS 9 (User Space) 32-/64-bit
A9X with iOS 9 (User Space) 32-/64-bit

Module: **Apple iOS CoreCrypto Kernel Module v6.0**

Platforms: **A5** with iOS 9 (Kernel Space)
A5X with iOS 9 (Kernel Space)
A6 with iOS 9 (Kernel Space)
A6X with iOS 9 (Kernel Space)
A7 with iOS 9 (Kernel Space)
A8 with iOS 9 (Kernel Space
A8X with iOS 9 (Kernel Space)
A9 with iOS 9 (Kernel Space) 32-/64-bit
A9X with iOS 9 (Kernel Space) 32-/64-bit

Compliant hardware

For FIPS 140-2 Compliance, the platforms noted above articulate Apple systems which were used for operational testing of the cryptographic modules. The CoreCrypto and CoreCrypto Kernel modules on Apple systems with either the A5, A5X, A6, A6X, A7, A8, A8X, A9, A9X processors running iOS 9 also take advantage of the additional processor embedded cryptographic engine. Compliant hardware are all Apple systems meeting the technical specifications to run iOS 9. The platforms that are compatible with iOS 9 as of **March 2016** can be found here https://support.apple.com/kb/DL1842 which notes the following:

- iPhone

  iPhone 6S Plus, iPhone 6S, iPhone 6 Plus, iPhone 6, iPhone 5 SE, iPhone 5s, iPhone 5c, iPhone 5, iPhone 4s

- iPad

  iPad Pro (12.9”), iPad Pro (9.7”), iPad Air 2, iPad Air, iPad 4th generation), iPad (3rd generation), iPad 2, iPad mini 4, iPad mini 3, iPad mini 2, iPad mini

- iPod touch

  iPod touch (6th generation), iPod touch (5th generation)

The FIPS Power-On-Self-Test (POST) process flow

1. Apple iOS system is physically Powered on

2. Operating System (iOS 9) begins the bootstrap process

3. Operating System ensures integrity of the **CoreCrypto Kernel Module**

    3.1. Validation of the `corecrypto.kext`
        3.1.1. The kernel determines operating environment (i.e arm7)
        3.1.2. The kernel reads a validated HMAC_SHA256 from the `corecrypto.kext`
        3.1.3. The `corecrypto.kext` is launched and given the correct validated HMAC from 3.1.2
        3.1.4. The `corecrypto.kext` will generate an HMAC_SHA256 of the corecrypto.kext code and compare the result against the validated HMAC_SHA256 from 3.1.2
        3.1.5. If the calculated HMAC_SHA256 does not match the validated HMAC_SHA256, the system will panic and halt
    3.2. The cipher Power-On-Self-Test (POST) validates the algorithms and modes
        3.2.1. The `corecrypto.kext` performs POST on algorithms and modes
        3.2.2. If any part of the POST fails, the system will panic and halt

4. Operating System ensures Integrity of **CoreCrypto Module**

    4.1. Validation of the `corecrypto.dylib`
        4.1.1. Upon user space environment setup by the kernel, **launchCtl** will launch the integrity test application `/usr/libexec/cc_fips_test`
        4.1.2. An HMAC_SHA256 of the user space corecrypto.dylib will be generated and compared to the HMAC_SHA256 value stored at `/var/db/FIPS/fips_data`
        4.1.3. If the calculated HMAC_SHA256 does not match the stored HMAC_SHA256, the system will panic and halt
    4.2. The cipher Power-On-Self-Test (POST) validates the algorithms and modes
        4.2.1. The `cc_fips_test` performs POST on algorithms and modes
        4.2.2. If any part of the POST fails, the system will panic and halt

5. Halt upon failure of any tests

    5.1. If any phase or step of testing components fails, the system will log the failure and panic and halt the device immediately.
    5.2. The logging messages are sent to the `console` and can be viewed using tools such as Xcode's "Devices".

How to verify integrity of the modules

A boot-up of the iOS 9 device forces the FIPS POST which verifies the integrity of both the CoreCrypto Kernel and CoreCrypto modules. If the device boots-up successfully, both modules have passed integrity verification. If the device halts or shuts down during boot-up, an integrity issue has been found during the POST process.

Rebooting the iOS 9 device will always force integrity verification of both cryptographic modules.

How to mitigate integrity issues of the modules

If a crypto module integrity issue has been identified during the FIPS POST, the only recourse the Crypto Office has for mitigation is to re-install iOS 9 on the device.

If the Crypto Officer needs assistance in restoring the iOS 9 Software, Apple Knowledge Base Articles should prove to be quite helpful.

A few helpful support articles available from the Apple Support Knowledge Base:

Update the iOS software on your iPhone, iPad, or iPod touch

https://support.apple.com/en-us/HT204204

Resolve iOS update and restore errors in iTunes

https://support.apple.com/en-us/HT201210

If needing to perform an Apple Support-wide search for all articles pertaining to "Restoring iOS Software", use the following URL:

http://support.apple.com/kb/index?page=search&product=&q=Restoring%20iOS%20Software&src=support_site.kbase.search.searchresults

If choosing to perform an Apple Support-wide search for all articles pertaining to "FIPS iOS", use the following URL:

http://support.apple.com/kb/index?page=search&product=&q=FIPS%20iOS&src=support_site.kbase.search.searchresults

FIPS 140-2 Validated Algorithms

The CoreCrypto and CoreCrypto Kernel Modules are cryptographic libraries offering various cryptographic mechanisms to Apple frameworks. Algorithms from the two Apple cryptographic modules in iOS 9 achieved **Cryptographic Algorithm Validation** under the Cryptographic Algorithm Validation Program (CAVP).

Modes of Operation

The CoreCrypto and CoreCrypto Kernel Modules have an Approved and Non-Approved modes of operation. The Approved mode of operation is configured in the system by default and cannot be changed. If the device boots up successfully then CoreCrypto framework and CoreCrypto KEXT have passed all self-tests and are operating in the Approved mode.

The Approved security functions are listed in **Table 3: Approved Security Functions** of the Non-Proprietary Security Policy documents posted along with the module validation certificate under CMVP. The Security Policy document links can be found above in the *Developer Resources* section. Column four (Val. No.) lists the validation numbers obtained from NIST for successful validation testing of the implementation of the cryptographic algorithms on the platforms as shown in Table 2 under CAVP.

Any calls to the non-Approved security functions listed in **Table 4: Non-Approved Security Functions** of the Non-Proprietary Security Policy documents will cause the module to assume the non-Approved mode of operation. Operators of the modules are strongly advised to avoid calling the functions in Table 4. If the module is operating in the non-Approved mode, operators are strongly cautioned to not use any CSP's previously utilized in the Approved mode of operation.

Note in the Security Policy documents under Key / CSP Establishment that the module provides DH- and ECDH-based key establishment services in the Approved mode. The module provides key establishment services in the Approved mode through the PBKDFv2 algorithm. The PBKDFv2 function is provided as a service and returns the key derived from the provided password to the caller. 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, the quality of the salt as well as the number of iterations. The implementation of the PBKDFv2 function requires the user to provide this information.

Refer to http://csrc.nist.gov/groups/STM/cavp/index.html for the current standards, test requirements, and special abbreviations used.

To see the exhaustive list of all algorithms supported by the cryptographic modules, Crypto Officers are highly encouraged to obtain and read the Security Policy document for complete technical explanations on the CoreCrypto and CoreCrypto Kernel modules. Links are provided in the Developer and Crypto Officer Resources section above.

Suite B Cryptographic Algorithms

The CoreCrypto Module (User Space) does provide for the use of Suite B Cryptographic Algorithms as are called out on the NSA Suite B Cryptography web page. Those algorithms include AES (FIPS 197), ECDH (SP 800-56A), ECDSA (FIPS 186-4) and SHA-256/-384 (FIPS 180-4). For further information from NSA about Suite B Algorithms, refer to http://www.nsa.gov/ia/programs/suiteb_cryptography/.

Public Review of Cryptographic Libraries

The same libraries that secure iOS and OS X are available to third−party developers to help them build advanced security features.

Cryptographic Libraries https://developer.apple.com/cryptography/

— Security Framework

Security Framework provides interfaces for managing certificates, public and private keys, and trust policies. It supports the generation of cryptographically secure pseudorandom numbers. It also supports the storage of certificates and cryptographic keys in the keychain, which is a secure repository for sensitive user data.

— Common Crypto

The Common Crypto library provides additional support for operations like symmetric encryption, hash-based message authentication codes, and digests.

— corecrypto

Although the CoreCrypto Modules do not directly provide programming interfaces for developers and should not be used by iOS or OS X apps, the source code has been posted and is available to allow for verification of its security characteristics and correct functioning.