Apple Inc.
Certification Authority
Certification Practice Statement
Worldwide Developer Relations

Version 1.11
Effective Date: June 3, 2014

Table of Contents

1. Introduction 5
   1.1. Trademarks 5
   1.2. Table of acronyms 5
   1.3. Definitions 5
2. General business practices 6
   2.1. Identification 6
   2.2. Community and applicability 6
      2.2.1. iOS Development Certificates 7
      2.2.2. iOS Submission Certificates 7
      2.2.3. Development Client SSL Certificates 7
      2.2.4. Production Client SSL Certificates 7
      2.2.5. Push CSR Signing Certificates 7
      2.2.6. Safari Extension Signing Certificates 7
      2.2.7. Mac Application Development Certificates 7
      2.2.8. Mac Application Submission Certificates 8
      2.2.9. Mac Installer Package Submission Certificates 8
      2.2.10. Mac App Store Application Certificates 8
      2.2.11. Mac App Store Installer Package Certificates 8
      2.2.12. Mac App Store Receipt Signing Certificates 8
      2.2.13. Mac Provisioning Profile Signing Certificates 8
      2.2.14. Pass Certificates 8
      2.2.15. Website Push Notification Certificates 8
      2.2.16. OS X Server Authentication Certificates 8
      2.2.17. VoIP Services Push Certificates 8
   2.3. Contact details 9
   2.4. Apportionment of liability 9
      2.4.1. Warranties to Subscribers and Relying Parties 9
      2.4.2. CA disclaimers of warranties 9
      2.4.3. CA limitations of liability 9
      2.4.4. Subscriber warranties 9
      2.4.5. Private key compromise 10
      2.4.6. Subscriber and Relying Party liability 10
   2.5. Financial responsibility 10
      2.5.1. Indemnification by Subscribers and Relying Parties 10
      2.5.2. Fiduciary relationships 10
   2.6. Interpretation and enforcement 10
      2.6.1. Governing law 10
      2.6.2. Severability, survival, merger, notice 10
      2.6.3. Dispute resolution procedures 10
   2.7. Fees 10
      2.7.1. Certificate issuance or renewal fees 10
      2.7.2. Certificate access fees 11
      2.7.3. Revocation or status information access fees 11
      2.7.4. Fees for other services 11
      2.7.5. Refund policy 11
   2.8. Publication and Repository 11
      2.8.1. Publication of CA information 11
      2.8.2. Frequency of publication 11
      2.8.3. Access controls 11
   2.9. Compliance audit requirements 11
   2.10. Conditions for applicability 11
      2.10.1. Permitted uses 11
      2.10.2. Limitations on use 13

2.11. Obligations 13
2.11.1. General Sub-CA obligations 13
2.11.2. Notification of issuance to Subscriber 14
2.11.3. Notification of issuance to others 14
2.11.4. Notification of revocation to Subscriber 14
2.11.5. Notification of revocation to others 14
2.11.6. Registration Authority obligations 14
2.11.7. Subscriber obligations 14
2.11.8. Relying Party obligations 14
3. Key life cycle management 16
3.1. Sub-CA key generation 16
3.2. Sub-CA private key protection 16
3.2.1. Sub-CA private key storage 16
3.2.2. Sub-CA private key control 16
3.2.3. Sub-CA key escrow 16
3.2.4. Sub-CA key backup 16
3.2.5. Sub-CA key archival 16
3.3. Sub-CA provided Subscriber key management 16
3.4. Sub-CA public key distribution 17
3.5. Sub-CA key changeover 17
4. Certificate life cycle management 18
4.1. External RA requirements 18
4.2. Certificate registration 18
4.3. Certificate renewal 18
4.4. Certificate rekey 18
4.5. Certificate issuance 18
4.6. Certificate acceptance 18
4.7. Certificate distribution 19
4.8. Certificate revocation 19
4.9. Certificate suspension 19
4.10. Certificate status 19
4.10.1. CRL usage 19
4.10.2. OCSP usage 19
4.11. Certificate profile 20
4.11.1. iOS Development and Submission Certificates 20
4.11.2. APNs SSL Certificates 21
4.11.3. Push CSR Signing Certificates 21
4.11.4. Safari Extension Signing Certificates 21
4.11.5. Mac Application Development Certificates 22
4.11.6. Mac Application Submission Certificates 22
4.11.7. Mac Installer Package Submission Certificates 22
4.11.8. Mac App Store Application Certificates 23
4.11.9. Mac App Store Installer Package Certificates 23
4.11.10. Mac App Store Receipt Signing Certificates 23
4.11.11. Mac Provisioning Profile Signing Certificates 24
4.11.12. Pass Certificates 24
4.11.13. Website Push Notification Certificates 24
4.11.14. VoIP Services Push Certificates 25
4.11.15. OS X Server Authentication Certificates 26
4.12. CRL Profile 26
4.13. Integrated circuit cards 26
5. Environmental controls 27
5.1. CPS administration 27
5.2. CA termination 27
5.3. Confidentiality 27

5.4. Intellectual property rights 28
5.5. Physical security 28
5.6. Business continuity management 28
5.7. Event logging 28
5.7.1. Archiving 28
5.7.2. Event journal reviews 28
6. Revision history 29

1. Introduction

This Certification Practice Statement ("CPS") describes the practices employed by the Apple Worldwide Developer Relations Subordinate Certification Authority ("WWDR Sub-CA") and the Developer Authentication Certification Authority ("Developer Authentication Sub-CA") in issuing and managing digital certificates and related services. These practices, and the structure of this document, are designed to align to the requirements defined in the Apple Certificate Policy ("CP"). Where the CP defines policies that all applicable Apple Sub-CA's are required to follow, this CPS provides more detailed information about the practices employed by the WWDR Sub-CA and the Developer Authentication Sub-CA relating to certificate lifecycle services, such as issuance, management, revocation, renewal, and rekeying, as well as details relating to other business, legal, and technical matters specific to the WWDR Sub-CA and Developer Authentication Sub-CA, collectively referred to as the WWDR Public Key Infrastructure ("WWDR PKI").

Apple Inc. ("Apple") established the Apple Root Certification Authority ("Apple Root CA") and the Apple PKI in support of the generation, issuance, distribution, revocation, administration and management of public/private cryptographic keys that are contained in CA-signed X.509 Certificates. The Apple PKI is intended to support internal and external Apple cryptographic requirements, where authentication of an organization or individual presenting a digitally signed or encrypted object to a Relying Party is of benefit to participants in the Apple PKI.

1.1. Trademarks

Apple, Mac, iOS, iPhone, iPad, iPod and iPod touch are trademarks of Apple Inc., in the United States and other countries.

1.2. Table of acronyms

Please refer to the CP for a table of acronyms used within this document.

1.3. Definitions

For the purposes of this CPS:

- "Developer" means an individual or organization, or an individual authorized to act on behalf of an individual or company (principal), that has registered as a member of the Developer Program with WWDR and has received a Certificate from the WWDR Sub-CA or the Developer Authentication Sub-CA.
- "Apple Developer website" refers to the web environment located at developer.apple.com/.
- "APNs" refers to the Apple Push Notification service, the Apple service that allows for the propagation of information to Mac or iOS devices.

Please refer to the CP for all other definitions used within this document.

2. General business practices

This section establishes and sets forth the general business practices of the WWDR Sub-CA and the Developer Authentication Sub-CA.

2.1. Identification

The practices set forth in this CPS apply exclusively to the WWDR Sub-CA and the Developer Authentication Sub-CA. This CPS is structured similarly to the CP, disclosing details of the practices employed by the WWDR Sub-CA and the Developer Authentication Sub-CA that address the more general requirements defined in the CP. This document assumes the reader is familiar with the general concepts of digital signatures, certificates, and public-key infrastructure. If the reader is new to Public Key Infrastructure concepts, the reader may choose to consult the introduction and overview of the WebTrust Program for Certification Authorities, a guide published by the American Institute of Certified Public Accountants (AICPA) and freely available for download from their web site, www.aicpa.org. The guide contains an overview of PKI, including an orientation on key concepts such as digital signatures, asymmetric key pairs, certification authorities, registration authorities, policy and practice statements, and business issues and considerations.

For the purposes of this CPS, the term Apple PKI refers collectively to Apple PKI Service Providers and End Entities. Apple PKI Service Providers consist of (1) Apple Certification Authorities ("CAs"), including the Apple Root CA, the WWDR Sub-CA, the Developer Authentication Sub-CA and their related management teams that generate, issue, distribute, revoke and manage cryptographic keys and Certificates, (2) Apple Registration Authorities ("Apple RA"), and (3) the Apple CA Policy Authority ("Apple PA," or "PA"). End Entities are Subscribers of Certificates.

The WWDR Sub-CA and the Developer Authentication Sub-CA issues and administers Certificates in accordance with policies in the Apple CP document.

2.2. Community and applicability

This CPS is applicable to the following certificates issued by the WWDR Sub-CA and the Developer Authentication Sub-CA:

- WWDR iOS Software Development Certificates ("iOS Development Certificates")
- WWDR iOS Software Submission Certificates ("iOS Submission Certificates")
- WWDR Apple Push Notification service Development SSL Certificates ("Development SSL Certificates")
- WWDR Apple Push Notification service Production SSL Certificates ("Production SSL Certificates")
- WWDR Push Certificate Signing Request Signing Certificates ("Push CSR Signing Certificates")
- WWDR Safari Extension Signing Certificates ("Safari Certificates")
- WWDR Mac App Development Certificates ("Mac App Development Certificates")
- WWDR Mac App Submission Certificates ("Mac App Submission Certificates")
- WWDR Mac Installer Package Submission Certificates ("Mac Installer Package Submission Certificates")
- Mac App Store Application Signing Certificates ("Mac App Store Application Certificates")

- Mac App Store Installer Package Signing Certificates ("Mac App Store Installer Package Certificates")
- Mac App Store Receipt Signing Certificates
- Mac Provisioning Profile Signing Certificates
- Pass Certificates
- Website Push Notification Certificates
- OS X Server Authentication Certificates
- VoIP Services Push Certificates

Certificates used exclusively for functions internal to Apple Products and/or Apple processes, such as device profile signing Certificates, event log signing Certificates, etc., are not included within the scope of this CPS.

2.2.1. iOS Development Certificates

The WWDR Sub-CA issues and administers Certificates that may be used by Developers to digitally sign a software application, enabling the application to be tested on an Apple iPhone, iPad and/or iPod touch.

2.2.2. iOS Submission Certificates

The WWDR Sub-CA issues and administers Certificates used by a Developer to digitally sign software applications for submission to Apple, or for distribution to their internal employees.

2.2.3. Development Client SSL Certificates

The WWDR Sub-CA issues and administers Client SSL Certificates that are used by Developers to provide SSL connectivity and client authentication for the Apple Push Notification service Development environment.

2.2.4. Production Client SSL Certificates

The WWDR Sub-CA issues and administers Client SSL Certificates that are used by Developers to provide SSL connectivity and client authentication for the Apple Push Notification service Production environment.

2.2.5. Push CSR Signing Certificates

The WWDR Sub-CA issues and administers Certificates that are used to digitally sign Certificate Signing Requests, enabling these Certificate Signing Requests to be submitted to the Apple Push Certificate Portal.

2.2.6. Safari Extension Signing Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign a Safari web browser extension, enabling it to be installed in the Safari application.

2.2.7. Mac Application Development Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign a Mac application bundle, enabling it to be tested on an Apple Mac.

2.2.8. Mac Application Submission Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign a Mac application bundle, enabling it to be submitted to Apple.

2.2.9. Mac Installer Package Submission Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign a Mac installer package, enabling it to be submitted to Apple.

2.2.10. Mac App Store Application Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Apple to sign application bundles distributed through the Mac App Store.

2.2.11. Mac App Store Installer Package Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Apple to sign installer packages distributed through the Mac App Store.

2.2.12. Mac App Store Receipt Signing Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Apple to sign receipts for applications delivered through the Mac App Store.

2.2.13. Mac Provisioning Profile Signing Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Apple to sign provisioning profiles for Mac application development and submission to the Mac App Store.

2.2.14. Pass Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign passes for Passbook, enabling them to be installed on an Apple device, and to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates.

2.2.15. Website Push Notification Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers to digitally sign notification packages for websites on OS X, enabling them to be authenticated on an Apple device, and to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates.

2.2.16. OS X Server Authentication Certificates

The Developer Authentication Sub-CA issues and administers Certificates that are used by Developers to access WWDR API's through Developer Tools.

2.2.17. VoIP Services Push Certificates

The WWDR Sub-CA issues and administers Certificates that are used by Developers of VoIP apps to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates for VoIP apps.

2.3. Contact details

The CA's Certificate Policies are administered by the Apple CA Policy Authority. The contact information for this CPS is:

Apple CA Policy Authority
C/O General Counsel
Apple Inc.
1 Infinite Loop
Cupertino, CA 95014

(408) 996-1010
policy_authority@apple.com

2.4. Apportionment of liability

For Certificates issued to Developers, a subscriber agreement is incorporated in the applicable License Agreement. For Mac App Store Receipt Signing Certificates, a relying party agreement will be incorporated into the applicable License Agreement. There is not an applicable Relying Party agreement for any other WWDR Sub-CA or Developer Authentication Sub-CA Certificates as the relying parties are internal to Apple. Except as provided herein, parties external to Apple are expressly prohibited from placing reliance on any aspects of the WWDR PKI.

2.4.1. Warranties to Subscribers and Relying Parties

The WWDR Sub-CA and the Developer Authentication Sub-CA does not warrant the use of any Certificate to any Subscriber or Relying Party.

2.4.2. CA disclaimers of warranties

To the extent permitted by applicable law, subscriber agreements disclaim warranties from Apple, including any warranty of merchantability or fitness for a particular purpose.

2.4.3. CA limitations of liability

To the extent permitted by applicable law, subscriber agreements shall limit liability on the part of Apple and shall exclude liability for indirect, special, incidental, and consequential damages.

2.4.4. Subscriber warranties

For Certificates issued to Developers, subscriber agreements shall require Subscribers to warrant that:

- They will take no action to interfere with the normal operation of a Certificate or products that rely on such certificates;
- They are solely responsible for preventing any unauthorized person from having access to the Subscriber's private key stored on any device for which the Subscriber is developing software for Apple platforms; and
- The Certificates are being used exclusively for authorized and legal purposes.

2.4.5. Private key compromise

Apple reserves the right to revoke any Certificates, without notice, if it believes the Subscriber's private key has been compromised, or upon request from the Subscriber.

2.4.6. Subscriber and Relying Party liability

Subscribers and Relying Parties will hold Apple harmless from any and all liabilities, losses, actions, damages or claims (including all reasonable expenses, costs, and attorneys fees) arising out of or relating to their use of any digital Certificate.

2.5. Financial responsibility

This section sets forth policies as requirements on the WWDR Sub-CA and the Developer Authentication Sub-CA related to indemnification by Relying Parties and disclosure of fiduciary relationships in relying party agreements.

2.5.1. Indemnification by Subscribers and Relying Parties

Any subscriber or relying party agreement may, at Apple's discretion, include an indemnification clause by Subscribers and/or Relying Parties.

2.5.2. Fiduciary relationships

There is no fiduciary relationship between Apple and Subscribers and/or Relying Parties.

2.6. Interpretation and enforcement

Interpretation and enforcement of any subscriber or relying party agreement is governed by the terms and conditions in the SDK License Agreement.

2.6.1. Governing law

Governing law is set forth in the SDK License Agreement.

2.6.2. Severability, survival, merger, notice

Severability, survival, merger and notice if applicable, is governed by the terms and conditions in the SDK License Agreement.

2.6.3. Dispute resolution procedures

Dispute resolution procedures are set forth in the SDK License Agreement.

2.7. Fees

This section sets forth policies associated with any fees charged to Subscribers for certification authority services for each type of Certificate.

2.7.1. Certificate issuance or renewal fees

No fees are charged for this service. Certificates issued to Developers are available at no additional cost to members of the iOS, Mac, or Safari Developer Program. Certificates are valid for the duration of the membership period unless otherwise revoked.

2.7.2. Certificate access fees

No fees are charged for this service.

2.7.3. Revocation or status information access fees

No fees are charged for this service.

2.7.4. Fees for other services

No other fees are charged for CA services.

2.7.5. Refund policy

Not applicable.

2.8. Publication and Repository

The WWDR Sub-CA and the Developer Authentication Sub-CA operate a private repository, which is not publicly accessible.

2.8.1. Publication of CA information

The latest version of this CPS for the WWDR Sub-CA can be found at http://www.apple.com/appleca.

2.8.2. Frequency of publication

Public key Certificates issued to Developers by the WWDR Sub-CA are made available to Subscribers via the Apple Developer website upon issuance. Certificate status is also made available via a Certificate Revocation List ("CRL") which is published upon issuance, and/or via the Online Certificate Status Protocol ("OCSP").

2.8.3. Access controls

There is no public repository of certificates. Developers shall have access to their own Certificates through the Apple Developer website. Certificate status information is publicly available via CRL or OCSP, which will be provided in the manner described by the CRL Distribution Points, or the Certificate Authority Information Access (AIA) extension present in the leaf Certificates issued by the WWDR Sub-CA.

2.9. Compliance audit requirements

The WWDR Sub-CA and the Developer Authentication Sub-CA adopts wholly all policies under this section in the CP.

2.10. Conditions for applicability

This section sets forth practices related to the use of the WWDR Sub-CA and the Developer Authentication Sub-CA.

2.10.1. Permitted uses

The WWDR Sub-CA and the Developer Authentication Sub-CA will create keys, manage keys, issue Certificates, manage key life cycles, manage certificate life cycles, operate a private

repository, and perform other functions to support distribution for the following types of Certificates:

- iOS Development Certificates: This type of Certificate may be used by Developers authorized by Apple to digitally sign a software application enabling the application to be tested on an Apple Mac, iPhone, iPad and/or iPod touch and/or distributed for internal company use.
- iOS Submission Certificates: This type of Certificate may be used by a Developer to digitally sign software applications for submission to Apple, or for distribution to their internal employees.
- Development SSL Certificates: This type of Certificate may be used by Developers authorized by Apple to create and maintain SSL connectivity to the Apple Push Notification service Development environment enabling remote notifications to be sent via the Apple Push Notification service Development server to an Apple Mac, iPhone, iPad and/or iPod touch.
- Production SSL Certificates: This type of Certificate may be used by Developers authorized by Apple to create and maintain SSL connectivity to the Apple Push Notification service Production environment enabling remote notifications to be sent via the Apple Push Notification service Production server to an Apple Mac, iPhone, iPad and/or iPod touch.
- Push CSR Signing Certificates: This type of Certificate, may be used by Apple authorized Developers, enabling CSRs to be digitally signed and submitted to the Apple Push Certificate Portal.
- Safari Certificates: This type of Certificate may be used by Developers authorized by Apple to digitally sign a Safari web browser extension, enabling it to be installed by Safari users.
- Mac Application Development Certificates: This type of Certificate may be used by Developers authorized to sign a Mac OS X application bundle enabling it to be tested on a Mac used for application development.
- Mac Application Submission Certificates: This type of Certificate may be used by Developers authorized to sign a Mac OS X application bundle enabling it to be submitted to the App Store.
- Mac Installer Package Certificates: This type of Certificate may be used by Developers authorized to sign a Mac OS X installer package enabling it to be submitted to the App Store.
- Mac App Store Application Certificates: This type of Certificate may be used by Apple to sign a Mac OS X application bundle enabling it to be distributed via the Mac App Store.
- Mac App Store Installer Package Certificates: This type of Certificate may be used by Apple to sign a Mac OS X installer package enabling it to be distributed via the Mac App Store.
- Mac App Store Receipt Signing Certificates: This type of Certificate may be used by Apple to sign receipts for applications delivered through the Mac App Store.
- Mac Provisioning Profile Signing Certificates: This type of Certificate may be used by Apple to sign provisioning profiles used for Mac application development and submission to the Mac App Store.

- Pass Certificates: This type of Certificate may be used by Developers authorized by Apple to digitally sign passes for Passbook, enabling them to be installed on an Apple device, and to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates.
- Website Push Notification Certificates: This type of Certificate may be used by Developers authorized to digitally sign notification packages for websites on OS X, enabling them to be authenticated to an Apple device and to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates.
- OS X Server Authentication Certificates: This type of Certificate may be used by Developers to access WWDR APIs' through Developer Tools.
- VoIP Services Push Certificates: This type of Certificate may be used by Developers of VoIP apps to create and maintain SSL connectivity to the Apple Push Notification Services Production environment for notification of updates for VoIP apps.

Certificates used exclusively for functions internal to Apple products and/or Apple processes, such as device profile signing Certificates, event log signing Certificates, etc. are not included within the scope of this CPS.

2.10.2. Limitations on use

The WWDR Sub-CA and the Developer Authentication Sub-CA will not allow its Certificates to be used to create a certification authority or to allow its private key to sign a Certificate issued by another certification authority.

Except for internal-use Certificates, the WWDR Sub-CA and Developer Authentication Sub-CA Certificates shall not be used for any purpose that is not identified in this CPS § 2.10.1 as a permitted use.

2.11. Obligations

This section sets forth policies related to the obligations of the WWDR Sub-CA and Developer Authentication Sub-CA.

2.11.1. General Sub-CA obligations

The WWDR Sub-CA and the Developer Authentication Sub-CA shall:

- Conform its operations to the Apple CP and to this CPS as the same may be amended from time to time.
- Issue and publish Certificates in accordance with the Apple CP and this CPS.
- Revoke Certificates issued by the WWDR Sub-CA and the Developer Authentication Sub-CA, upon receipt of a valid request to revoke the Certificate from a person authorized to request such a revocation. The validity of the request and the authorization of the person making the request will be determined by the WWDR Sub-CA and the Developer Authentication Sub-CA.
- Publish Certificate Revocation Lists (CRLs) on a regular basis in accordance with the Apple CP. As applicable, the CA shall notify the subscriber that the certificate has been revoked.

2.11.2. Notification of issuance to Subscriber

For certificates issued to Developers, notification to Subscribers is deemed to have taken place when newly issued Certificates are made available via the Apple Developer website, or downloaded to the Subscriber's machine.

2.11.3. Notification of issuance to others

For certificates issued to Developers, notification to a Developer's Agent and/or Administrator is deemed to have taken place when newly issued certificates are made available via the Apple Developer website.

2.11.4. Notification of revocation to Subscriber

Upon revocation of the Subscriber's certificate, the Subscriber is notified by email.

2.11.5. Notification of revocation to others

Notification of revocation to others is deemed to have taken place upon publication of the CRL or update of the certificate status information in the OCSP Response.

2.11.6. Registration Authority obligations

An external RA is not used. The WWDR Sub-CA and the Developer Authentication Sub-CA performs limited RA services to provide reasonable assurance that Certificates are only issued to members of the iOS Developer Program and/or Safari Developer Program and/or Mac Developer Program.

2.11.7. Subscriber obligations

Subscribers are obligated to:

- Safeguard their private key from compromise.
- Use their Certificates exclusively for legal purposes.
- Promptly request that a Certificate be revoked if the Subscriber has reason to believe there has been a compromise of the Certificate's associated private key. For Pass Certificates, and Website Push Notification Certificates, a request for revocation is initiated by sending an email to product-security@apple.com. For OS X Server Authentication Certificates, revocation is initiated by disassociating the OS X Server from the team via Xcode, or via the Apple Developer website. For all other WWDR Sub-CA certificates issued to Developers, revocation is performed via the Apple Developer website. After authenticating to the website, the Subscriber follows the link to the revocation page and identifies the Certificate to be revoked. Only Certificates issued to the authenticated Subscriber, can be revoked based upon a request from such entity.
- Promptly request that a Certificate be revoked if the Subscriber is not authorized to use the applicable Certificate on behalf of the Subscriber entity (e.g. no longer employed by the Subscriber entity).
- Take no action to transfer their Certificate to any third party.

2.11.8. Relying Party obligations

Relying Parties are obligated to:

- Acknowledge that they are solely responsible for deciding whether or not to rely on the information in a Certificate, and agree that they have sufficient information to make an informed decision. Apple shall not be responsible for assessing the appropriateness of the use of a Certificate.
- Acknowledge that, to the extent permitted by applicable law, Apple hereby disclaims all warranties regarding the use of any Certificates, including any warranty of merchantability or fitness for a particular purpose. In addition, Apple hereby limits its liability and excludes all liability for indirect, special, incidental, and consequential damages.
- Restrict reliance on Certificates issued by the CA to the purposes for which those Certificates were issued, in accordance with the CP and this CPS.

3. Key life cycle management

This section sets forth practices related to the key life cycle management controls of the WWDR Sub-CA and the Developer Authentication Sub-CA.

3.1. Sub-CA key generation

The WWDR Sub-CA and the Developer Authentication Sub-CA signing key generation occurs using a secure cryptographic device meeting the requirements as described in CP §3.2. The signing key pair is 2048-bit using the RSA algorithm.

The WWDR Sub-CA and the Developer Authentication Sub-CA shall sign Certificates and Sub-CA CRLs.

The WWDR Sub-CA and the Developer Authentication Sub-CA private keys will cease to be used, and be replaced at the end of a designated period, up to a maximum of fifteen (15) years, or when a compromise is known or suspected.

3.2. Sub-CA private key protection

3.2.1. Sub-CA private key storage

Each Sub-CA private key is stored in a Hardware Security Module (HSM) that is tamper resistant and certified at a minimum level of FIPS 140-1 Level 4.

3.2.2. Sub-CA private key control

There is a separation of physical and logical access to each Sub-CA private key, and a minimum of two individuals is required for physical access to the HSM where the Sub-CA private keys are stored. The private key is stored in encrypted key fragments with split knowledge and ownership and *m of n* key fragments are required for private key recovery.

3.2.3. Sub-CA key escrow

The WWDR Sub-CA and Developer Authentication Sub-CA private keys shall not be placed in escrow.

3.2.4. Sub-CA key backup

The WWDR Sub-CA and Developer Authentication Sub-CA private keys are backed up for recovery purposes. Backups are stored in a secured environment, and m of n key fragments are required for logical recovery.

3.2.5. Sub-CA key archival

The WWDR Sub-CA and the Developer Authentication Sub-CA shall archive any necessary keys for a period of time sufficient to support the responsibilities of the WWDR Sub-CA.

3.3. Sub-CA provided Subscriber key management

The WWDR Sub-CA and the Developer Authentication Sub-CA do not provide Subscriber key management services.

3.4. Sub-CA public key distribution

The WWDR Sub-CA and Developer Authentication Sub-CA public key will be contained in an X.509 Certificate that may be provided to Subscribers as necessary to support the WWDR PKI.

3.5. Sub-CA key changeover

When a new private key is required, a new signing key pair will be generated and all subsequently issued certificates and CRLs are signed with the new private signing key. The corresponding new WWDR Sub-CA and/or Developer Authentication public key Certificate may be provided to Subscribers as necessary to support the WWDR PKI.

4. Certificate life cycle management

This section sets forth practices related to the certificate life cycle management controls of the WWDR Sub-CA and the Developer Authentication Sub-CA.

4.1. External RA requirements

An external Registration Authority is not utilized by the WWDR Sub-CA or the Developer Authentication Sub-CA.

4.2. Certificate registration

Eligible Subscribers create a Certificate Signing Request ("CSR") using a corresponding private/public key pair generated on the client computer. For OS X Server Authentication certificates, the CSR is automatically generated and sent via Xcode when the Subscriber logs in with their Apple ID and password and requests to join the server to a team. For all other WWDR certificates, Subscribers upload the completed CSRs manually after logging into the Apple Developer website with their Apple ID and password.

Upon receipt, the CSR is processed by the Apple Developer website. The issuance of a Developer Certificate is contingent upon the requesting Subscriber being an eligible member of the iOS Developer Program and/or Safari Developer Program and/or Mac Developer Program. The Apple Developer website verifies that the account is one that is eligible for the issuance of Apple Worldwide Developer Relations certificates and that, if applicable, the iOS and/or Mac Developer subscription payments are current.

The name associated with a Certificate is either the individual Subscriber's name and organization, or the individual Subscriber's email address, or the organization name, or application identifier, or team identifier, only as applicable.

Certificates with Apple as the intended Subscriber are issued at the request of an authorized Apple employee in accordance with internal business practices.

4.3. Certificate renewal

For OS X Server Authentication certificates, Xcode will automatically initiate a new Certificate request when the existing certificate is close to expiring. A new certificate is generated if the Subscriber continues to be an eligible member of a Developer Program.

For all other WWDR certificates, a new certificate can be issued at the request of the Subscriber using the same process used at initial Certificate issuance.

4.4. Certificate rekey

The WWDR Sub-CA and the Developer ID Sub-CA does not rekey certificates. Compromised keys result in completely new key sets and certificates being issued.

4.5. Certificate issuance

Certificates are issued to the ISO 9594/X.509 standard and signed using the WWDR Sub-CA or the Developer Authentication Sub-CA signing key.

4.6. Certificate acceptance

For certificates issued to Developers, Certificates shall be deemed accepted and valid immediately upon issuance.

4.7. Certificate distribution

Certificates will be distributed to the Developer through the Apple Developer website, or automatically downloaded to the Developer's machine via Xcode.

4.8. Certificate revocation

Certificates with Apple as the intended Subscriber are revoked at the request of an authorized Apple employee in accordance with internal business practices.

For Pass Certificates, and Website Push Notification Certificates, the Subscriber may initiate a revocation request by sending an email to product-security@apple.com. The request for revocation will then be evaluated by Apple.

For all other certificates issued to Developers, the certificate revocation process will commence upon receipt of a valid request to revoke the Certificate from the Subscriber via the Apple Developer website or via Xcode. The Subscriber will be required to log in using their Apple ID and password. After authentication, the Subscriber will indicate that they wish to revoke their Certificate. Revocation of a Certificate cannot be undone. After revocation, a new Certificate must be requested.

Certificates may be revoked by Apple for any reason.

Revoked certificates are noted in the WWDR Sub-CA CRL, and/or via the Online Certificate Status Protocol.

4.9. Certificate suspension

Certificate suspension is not supported. Instead, Subscribers are required to revoke their current Certificates and request new ones.

4.10. Certificate status

The WWDR Sub-CA utilizes two methods for certificate validation, Certificate Revocation List (CRL) and Online Certificate Status Protocol (OCSP). Refer to the CRL Distribution Point ("CDP") or the Authority Information Access ("AIA") extension in the Certificates for the status information method used.

4.10.1. CRL usage

Subscribers may use a CRL, which is updated periodically at Apple's sole discretion, to determine the status of a particular Certificate. Revoked Certificates remain in the CRL until the Certificates have expired. More than one valid CRL Certificate may exist at one time.

Certificates and CRLs issued by the WWDR Sub-CA and the Developer Authentication Sub-CA shall be retained for a period of not less than two (2) years.

4.10.2. OCSP usage

Subscribers may use OCSP to determine the status of a particular Certificate. Revoked Certificates remain marked as "revoked" for the certificate lifetime. A delegate leaf Certificate is used to sign all OCSP responses. This leaf is signed by the WWDR Sub-CA or Developer Authentication Sub-CA private key.

OSCP status requests must contain at a minimum the certificate serial number to receive a valid response. Once an OCSP request has been validated there will be a signed response back to the requestor indicating the status of the Certificate and showing the request was successful. Failed OCSP requests will generate a failure status back to the requestor.

4.11. Certificate profile

Certificates issued by the WWDR Sub-CA shall conform to the X.509 version 3 Certificate format, and shall contain the following elements:

Field/Attribute	Value
Issuer DN	C = US, O = Apple Inc., OU = Apple Worldwide Developer Relations, CN = Apple Worldwide Developer Relations Certification Authority
CRL Distribution Points and/or Certificate Authority Information Access	URL of the location where a Relying Party can check the status of a certificate.

Individual WWDR Sub-CA certificate profiles also contain the following

4.11.1. iOS Development and Submission Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Code Signing (1.3.6.1.5.5.7.3.3)
Custom Extensions (One of the following)	Yes	iPhone Software Submission Signing (1.2.840.113635.100.6.1.4)
	Yes	iPhone Software Development Signing (1.2.840.113635.100.6.1.2)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.2. APNs SSL Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	No	Digital Signature
Extended Key Usage	No	Client Authentication (1.3.6.1.5.5.7.3.2)
Custom Extensions	No	Apple Push Notification services Development (1.2.840.113635.100.6.3.1)
	No	Apple Push Notification service Production (1.2.840.113635.100.6.3.2)
Basic Constraints	No	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.3. Push CSR Signing Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Custom Extensions	No	CSR Signing (1.2.840.113635.100.4.12)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.4. Safari Extension Signing Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Safari Extension Signing (1.2.840.113635.100.4.8)
Custom Extensions	Yes	Safari Extension Signing (1.2.840.113635.100.6.1.5)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.5. Mac Application Development Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Code Signing (1.3.6.1.5.5.7.3.3)
Custom Extensions	Yes	Mac Application Software Development Signing (1.2.840.113635.100.6.1.12)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.6. Mac Application Submission Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Code Signing (1.3.6.1.5.5.7.3.3)
Custom Extensions	Yes	Mac Application Software Submission Signing (1.2.840.113635.100.6.1.7)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.7. Mac Installer Package Submission Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Mac Installer Package Signing (1.2.840.113635.100.4.9)
Custom Extensions	Yes	Mac Installer Package Submission Signing (1.2.840.113635.100.6.1.8)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.8. Mac App Store Application Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Code Signing (1.3.6.1.5.5.7.3.3)
Custom Extensions	No	Mac App Store Application Software Signing (1.2.840.113635.100.6.1.9)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.9. Mac App Store Installer Package Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	Yes	Mac App Store Installer Package Signing (1.2.840.113635.100.4.10)
Custom Extensions	No	Mac Installer Package Submission Signing (1.2.840.113635.100.6.1.10)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.10. Mac App Store Receipt Signing Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Custom Extensions	No	Mac App Store Receipt Signing (1.2.840.113635.100.6.11.1)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Mac App Store Receipt Signing Policy (1.2.840.113635.100.5.6.1)

4.11.11. Mac Provisioning Profile Signing Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	Yes	Digital Signature
Custom Extensions	No	Mac OS X Provisioning Profile Signing (1.2.840.113635.4.11)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.12. Pass Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-1 with RSA Encryption
Key Usage	No	Digital Signature
Extended Key Usage	No	Client Authentication (1.3.6.1.5.5.7.3.2)
	No	Apple custom extension (1.2.840.113635.100.4.14)
Custom Extensions	No	Apple custom extension (1.2.840.113635.100.6.1.16)
	No	Apple Push Notification service Production (1.2.840.113635.100.6.3.2)
Basic Constraints	No	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.13. Website Push Notification Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-2 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	No	Client Authentication (1.3.6.1.5.5.7.3.2)
	No	Apple custom extension (1.2.840.113635.100.4.15)
Custom Extensions	No	Apple custom extension (1.2.840.113635.100.6.1.17)

	No	Apple Push Notification service Production (1.2.840.113635.100.6.3.2)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.11.14. VoIP Services Push Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-2 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	No	Client Authentication (1.3.6.1.5.5.7.3.2)
Custom Extensions	No	Apple Push Notification services Development (1.2.840.113635.100.6.3.1)
	No	Apple Push Notification service Production (1.2.840.113635.100.6.3.2)
	No	Apple custom extension (1.2.840.113635.100.6.3.3)
	No	Apple custom extension (1.2.840.113635.100.6.3.4)
	No	Apple custom extension (1.2.840.113635.100.6.3.5)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

Certificates issued by the Developer Authentication Sub-CA shall conform to the X.509 version 3 Certificate format, and shall contain the following elements:

Field/Attribute	Value
Issuer DN	C = US, O = Apple Inc., OU = Apple Worldwide Developer Relations, CN = Developer Authentication Certification Authority
CRL Distribution Points and/or Certificate Authority Information Access	URL of the location where a Relying Party can check the status of a certificate.

Individual Developer Authentication Sub-CA certificate profiles also contain the following

4.11.15. OS X Server Authentication Certificates

Field/Attribute	Critical	Value
Signature Algorithm	N/A	SHA-2 with RSA Encryption
Key Usage	Yes	Digital Signature
Extended Key Usage	No	Client Authentication (1.3.6.1.5.5.7.3.2)
Custom Extensions	No	Apple custom extension (1.2.840.113635.100.6.21)
Basic Constraints	Yes	Certification Authority = No
Certificate Policy	No	Apple Certificate Policy (1.2.840.113635.100.5.1)

4.12. CRL Profile

A CRL issued by the WWDR Sub-CA or the Developer Authentication Sub-CA shall conform to the X.509 version 2 CRL format. Each CRL shall contain the following fields:

- Signature Algorithm of SHA-1 with RSA Encryption
- Issuer matching the Sub-CA Certificate's Distinguished Name
- This Update with the time of CRL issuance
- Next Update defining the period of validity
- Authority Key Identifier extension
- List of Revoked Certificates

4.13. Integrated circuit cards

Not applicable.

5. Environmental controls

This section sets forth practices related to the environmental controls of the WWDR Sub-CA and the Developer Authentication Sub-CA.

5.1. CPS administration

Apple has designated a management group called the Policy Authority (PA) with final authority and responsibility for specifying and approving this CPS.

This authorized body has performed an assessment to evaluate business risks and determine the security requirements and operational procedures to be included in the CPS for the following:

- Key life cycle management controls
- Certificate life cycle management controls
- CA environmental controls

The CPS is made publicly available to all Subscribers and Relying Parties, including any revisions that occur from time to time.

Any changes to the CPS, along with the effective date of the changes, shall be reviewed by the PA, and posted in a timely manner.

5.2. CA termination

As set forth in this section, any decision to terminate the WWDR Sub-CA or the Developer Authentication Sub-CA shall be approved by a member of the Apple Executive Team prior to the effective date of termination.

At the time of termination, Apple will develop a termination plan addressing the following:

- Provision of notice to related parties affected by the termination,
- Revocation of certificates issued by the Sub-CA,
- Preservation of the Sub-CA's archives and records

5.3. Confidentiality

Apple shall keep the following information confidential at all times:

- All private signing and client authentication keys
- Security and annual audits and security parameters
- Personal or non-public information about Subscribers
- Security mechanisms

Except as required to support the WebTrust audit performed by an independent external audit firm, confidential information should not be released to third parties unless required by law or requested by a court with jurisdiction over the CA. The information will be kept confidential even after the termination of the CA.

The following information shall not be considered confidential:

- Information included in Certificates

- The WWDR Sub-CA public Certificate
- The Developer Authentication Sub-CA public Certificate
- Information contained in the CA's CPS and CP documents
- Any Certificate status or Certificate revocation reason code

5.4. Intellectual property rights

Certificates and CRLs issued by the WWDR Sub-CA or Developer Authentication Sub-CA, information provided via the OCSP, the CPS and the CP are the property of Apple.

5.5. Physical security

Physical protection of equipment supporting the WWDR PKI is achieved through the creation of clearly defined security perimeters with appropriate physical barriers to entry. Details of the physical security policies and procedures are in appropriate internal security documents.

Equipment is located or protected to reduce the risks from environmental threats and hazards, including but not limited to power and air conditioning disruption or failure, water exposure, fire, telecommunications disruption or failure and opportunities for unauthorized access.

At end of life, cryptographic devices are physically destroyed or zeroized in accordance to manufacturers' guidance prior to disposal.

5.6. Business continuity management

Business continuity plans have been developed to maintain or restore the Sub-CA business operations in a timely manner following interruption or failure of critical business processes.

5.7. Event logging

5.7.1. Archiving

Event journal data is archived on a periodic basis.

A risk assessment has been performed to determine the appropriate length of time for retention of archived event journals.

5.7.2. Event journal reviews

Current or archived event journals may only be retrieved by authorized individuals and only for valid business or security reasons.

Event journals are reviewed periodically.

The review of current and archived event journals includes the identification and follow-up of exceptional, unauthorized, or suspicious activity.

6. Revision history

Issue Number	Issue Date	Details
1.0	03/06/08	Initial release.
1.1	03/17/09	Updates to reflect the addition of Apple Push Notification Service Client SSL Certificates.
1.2	06/07/10	Updates to reflect the addition of iPad, the Authority Information Access extension to Developer Identification Certificates and Safari Extension Signing Certificates.
1.3	10/20/10	Updates to reflect the addition of Mac Application and Mac Installer Package Submission Certificates.
1.4	01/06/11	Updates to reflect the addition of Mac App Store Application, Mac App Store Installer Package, and Mac App Store Receipt Signing Certificates.
1.5	07/18/11	Updates to reflect the addition of the Mac Provisioning Profile Signing Certificate.
1.6	9/2/11	Updates to reflect the addition of the Mac Application Development Certificate.
1.7	10/4/11	Updates to reflect the addition of the Push CSR Signing Certificate.
1.8	6/11/12	Updates to reflect the addition of the Pass Certificate. Minor updates to clarify the roles of existing certificate types.
1.9	3/21/13	Updates to reflect changes to the revocation process for Pass Certificates. Updated format of certificate profile descriptions.

1.10	6/10/13	Updates to reflect the addition of the Website Push Notification Certificates and the addition of the Developer Authentication Sub-CA and OS X Server Authentication Certificate Profile. Clarifications to the CA termination process.
1.11	6/3/14	Updates to reflect the addition of the VoIP Services Push Certificates.