**Version 1.1 (September 22, 2016)**

# Code Signing Working Group*

# Minimum Requirements for the Issuance and Management of Publicly-Trusted Code Signing Certificates

* This document was developed by the following members of the CA/Browser Forum Code Signing Working Group: Comodo, DigiCert, Entrust, GlobalSign, Izenpe, Microsoft, Symantec, SSC, and WoSign. It was not adopted by the Forum, but is presented here for publication.

# Contents

# 1. Scope

The Minimum Requirements for the Issuance and Management of Publicly-Trusted Code Signing Certificates describe a subset of the requirements that a Certification Authority must meet to issue publicly-trusted Code Signing Certificates. This document incorporates by reference both the Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates ("Baseline Requirements") and the Network and Certificate System Security Requirements as established by the CA/Browser Forum, copies of which are available on the CA/Browser Forum's website at www.cabforum.org.

The scope of these Requirements includes all "Code Signing Certificates", as defined below, and associated Timestamp Authorities, and all Certification Authorities technically capable of issuing Code Signing Certificates, including any Root CA that is publicly trusted for code signing and all other CAs that might serve to complete the validation path to such Root CA. These Requirements do not address the issuance, use, maintenance, or revocation of Certificates by enterprises that operate their own Public Key Infrastructure for internal purposes only, where the Root CA Certificate is not distributed by any Application Software Supplier (as defined in the Baseline Requirements).

# 2. Purpose

The primary goal of these Requirements is to enable trusted signing of code intended for public distribution, while addressing user concerns about the trustworthiness of signed objects and accurately identifying the software publisher. The Requirements also serve to inform users about the purpose of signed code, help users make informed decisions when relying on Certificates, help establish the legitimacy of signed code, help maintain the trustworthiness of software Platforms, help users make informed software choices, and limit the spread of malware. Code signing certificates do not identify a particular software object, identifying only the distributor of software.

# 3. References

As specified in the Baseline Requirements. Cross-references to Sections of the Baseline Requirements are notated with the letters "BR", as in "BR Section 1.2."

This document may also mention or refer to the CA/Browser Forum's Extended Validation Guidelines for the Issuance and Management of Extended Validation Certificates ("EV SSL Guidelines") and the Guidelines for the Issuance and Management of Extended Validation Code Signing Certificates ("EV Code Signing Guidelines"), also available on the CA/Browser Forum's website at www.cabforum.org.

# 4. Definitions

Capitalized Terms are as defined in the Baseline Requirements except where defined below:

**Anti-Malware Organization:** An entity that maintains information about Suspect Code and/or develops software used to prevent, detect, or remove malware.

**Application Software Supplier**: A supplier of software or other relying-party application software that displays or uses Code Signing Certificates, incorporates Root Certificates, and adopts these Requirements as all or part of its requirements for participation in a root store program.

**Certification Authority:** An organization subject to these Requirements that is responsible for a Code Signing Certificate and, under these Requirements, oversees the creation, issuance, revocation, and management of Code Signing Certificates. Where the CA is also the Root CA, references to the CA are synonymous with Root CA.

**Certificate Beneficiaries**: As defined in section 7.1.1.

**Certificate Requester:** A natural person who is the Applicant, employed by the Applicant, an authorized agent who has express authority to represent the Applicant, or the employee or agent of a third party (such as software publisher) who completes and submits a Certificate Request on behalf of the Applicant.

**Code Signature:** A Signature logically associated with a signed Object.

**Code Signing Certificate:** A digital certificate issued by a CA that contains a code Signing EKU, contains the anyExtendedKeyUsage EKU, or omits the EKU extension and is trusted in an Application Software Provider's root store to sign software objects. [NOTE: Appendix B, subsection (3) of Appendix B requires the presence of the codeSigning EKU and prohibits use of the anyExtendedKeyUsage EKU.]

**Declaration of Identity**: A written document that consists of the following:

1. the identity of the person performing the verification,
2. a signature of the Applicant,
3. a unique identifying number from an identification document of the Applicant,
4. the date of the verification, and
5. a signature of the Verifying Person.

**Effective Date**: The date this document is adopted as a root store requirement by an Application Software Supplier.

**High Risk Region of Concern (HRRC):** As set forth in Appendix D, a geographic location where the detected number of Code Signing Certificates associated with signed Suspect Code exceeds 5% of the total number of detected Code Signing Certificates originating or associated with the same geographic area.

**Issuer:** The CA providing a Code Signing Certificate to the Subscriber.

**Individual Applicant**: An Applicant who is a natural person and requests a Certificate that will list the Applicant's legal name as the Certificate's Subject.

**Lifetime Signing OID:** An optional extended key usage OID (1.3.6.1.4.1.311.10.3.13) used by Microsoft Authenticode to limit the lifetime of the code signature to the expiration of the code signing certificate.

**Object**: A contiguous set of bits that has been or can be digitally signed with a Private Key that corresponds to a Code Signing Certificate; also referred to herein as "Code".

**Organizational Applicant:** An Applicant that requests a Certificate with a name in the Subject field that is for an organization and not the name of an individual. Organizational Applicants include private and public corporations, LLCs, partnerships, government entities, non-profit organizations, trade associations, and other legal entities.

**Platform:** The computing environment in which an Application Software Supplier uses Code Signing Certificates, incorporates Root Certificates, and adopts these Requirements.

**QGIS:** As defined in the EV SSL Guidelines.

**QIIS:** As defined in the EV SSL Guidelines.

**Registration Identifier:** The unique code assigned to an Applicant by the Incorporating or Registration Agency in such entity's Jurisdiction of Incorporation or Registration.

**Requirements**: This document, the Baseline Requirements, and the Network and Certificate System Security Requirements.

**Signature**: An encrypted electronic data file which is attached to or logically associated with other electronic data and which (i) identifies and is uniquely linked to the signatory of the electronic data, (ii) is created using means that the signatory can maintain under its sole control, and (iii) is linked in a way so as to make any subsequent changes that have been made to the electronic data detectable.

**Signing Service**: An organization that signs an Object on behalf of a Subscriber using a Private Key associated with a Code Signing Certificate.

**Subscriber**: The Subject of a Code Signing Certificate. A Subscriber is the entity responsible for distributing the software but does not necessarily hold the copyright to any software.

**Suspect Code**: Code that contains malicious functionality or serious vulnerabilities, including spyware, malware and other code that installs without the user's consent and/or resists its own removal, and code that can be exploited in ways not intended by its designers to compromise the trustworthiness of the Platforms on which it executes.

**Takeover Attack**: An attack where a Signing Service or Private Key associated with a Code Signing Certificate has been compromised by means of fraud, theft, intentional malicious act of the Subject's agent, or other illegal conduct.

**Timestamp Authority**: A service operated by the CA or a delegated third party for its own code signing certificate users that timestamps data using a certificate chained to a public root, thereby asserting that the data (or the data from which the data were derived via a secure hashing algorithm) existed at the specified time. If the Timestamp Authority is delegated to a third party, the CA is responsible that the delegated third party complies with these guidelines.

**Timestamp Certificate**: A certificate issued to a Timestamp Authority to use to timestamp data.

**Trusted Platform Module**: A microcontroller that stores keys, passwords and digital certificates, usually affixed to the motherboard of a computer, which due to its physical nature makes the information stored there more secure against external software attack or physical theft.

**Verifying Person**: A notary, attorney, Latin notary, accountant, individual designated by a government agency as authorized to verify identities, or agent of the CA, who attests to the identity of an individual.

# 5. Abbreviations and Acronyms

As specified in the Baseline Requirements.

# 6. Conventions

Terms not otherwise defined in these Requirements are as defined in the CA's applicable agreements, user manuals, Certificate Policies, and Certification Practice Statements.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in these Requirements are used in accordance with RFC 2119.

# 7. Certificate Warranties and Representations

## *7.1 Certificate Beneficiaries*

Certificate Beneficiaries means any one of the following:

1. All Application Software Suppliers with whom the Issuer or its Root CA has entered into a contract for distribution of its Root Certificate in software distributed by such Application Software Suppliers, or

2. All Relying Parties who reasonably rely on such a Certificate while a Signature associated with the Certificate is valid.

## *7.2 Certificate Warranties*

1. **Compliance**. The Issuer and any Signing Service each represents that it has complied with these Requirements and the applicable Certificate Policy and Certification Practice Statement in issuing each Code Signing Certificate and operating its PKI or Signing Service.

2. **Identity of Subscriber**: At the time of issuance, the Issuer or Signing Service represents that it (i) operated a procedure for verifying the identity of the Subscriber that at least meets the requirements in Section 11 of this document, (ii) followed the procedure when issuing or managing the Certificate, and (iii) accurately described the same procedure in the Issuer's Certificate Policy or Certification Practice Statement.

3. **Authorization for Certificate:** At the time of issuance, the Issuer represents that it (i) operated a procedure for verifying that the Applicant authorized the issuance of the Certificate, (ii) followed the procedure, and (iii) accurately described the same procedure in the Issuer's Certificate Policy or Certification Practice Statement.

4. **Accuracy of Information:** At the time of issuance, the Issuer represents that it (i) operated a procedure for verifying that all of the information contained in the Certificate (with the exception of the subject:organizationalUnitName attribute) was true and accurate, (ii) followed the procedure, and (iii) accurately described the same procedure in the CA's Certificate Policy or Certification Practice Statement.

5. **Key Protection:** The Issuer represents that it provided the Subscriber at the time of issuance with documentation on how to securely store and prevent the misuse of Private Keys associated with Code Signing Certificates, or in the case of a Signing Service, securely stored and prevented the misuse of Private Keys associated with Code Signing Certificates;

6. **Subscriber Agreement:** The Issuer and Signing Service represent that the Issuer or Signing Service entered into a legally valid and enforceable Subscriber Agreement with the Applicant that satisfies these Requirements.

7. **Status:** The CA represents that it will maintain a 24 x 7 online-accessible Repository with current information regarding the status of Certificates as valid or revoked for the period required by these Requirements.

8. **Revocation:** The CA represents that it will revoke a Certificate upon the occurrence of a revocation event specified in these Requirements.

### *7.3 Applicant Warranty*

The Issuer or Signing Service MUST require, as part of the Subscriber Agreement, that the Applicant make the commitments and warranties set forth in Section 10.3.2 and/or Section 10.3.3 of this document, as applicable, for the benefit of the Issuer and the Certificate Beneficiaries.

# 8. Community and Applicability

### *8.1 Compliance*

The CA and/or all Signing Services MUST, at all times:

1. Comply with all laws applicable to its business and the Certificates it issues in each jurisdiction where it operates,

2. Comply with these Requirements,

3. Comply with the audit requirements set forth in Section 17 of this document, and

4. If a CA, be licensed as a CA in each jurisdiction where it operates, if licensing is required by the law of such jurisdiction for the issuance of Certificates.

If a court or government body with jurisdiction over the activities covered by these Requirements determines that the performance of any mandatory requirement is illegal, then such requirement is considered reformed to the minimum extent necessary to make the requirement valid and legal. This applies only to operations or certificate issuances that are subject to the laws of that jurisdiction. The parties involved MUST notify the Application Software Suppliers of the facts, circumstances, and law(s) involved.

## *8.2 Certificate Policies*

### 8.2.1 Implementation

The CA and its Root CA MUST develop, implement, enforce, display prominently on its Web site, and periodically update its policies and practices, including its Certificate Policy and/or Certification Practice Statement that implement the most current version of these Requirements.

With the exception of revocation checking for time-stamped and expired Certificates, Platforms are expected to validate Code Signatures in accordance with RFC 5280 when first encountered. Subsequent signature validation MAY ignore revocation, especially if rejecting the Code will cause the device to fail to boot. When a Platform encounters a Certificate that fails to validate due to revocation, the Platform should not permit the Code to execute. When a Platform encounters a Certificate that fails to validate for reasons other than revocation, the Platform should treat the Code as unsigned.

Ordinarily, a Code Signature created by a Subscriber is only considered valid until expiration of the Certificate. However, the "Timestamp" method and the "Signing Service" methods permit Code to remain valid for longer periods of time.

1. Timestamp Method: In this method, the Subscriber signs the Code, appends its Code Signing Certificate and submits it to a Timestamp Authority to be time-stamped. The resulting package can be considered valid after expiration of the Code Signing Certificate.

2. Signing Service Method: In this method, the Subscriber uses the service to sign compiled code, binary, file, app, or similar object. Alternatively, the service MAY sign a digest of the preceding objects. The resulting Code Signature is valid up to the expiration time of the Signing Service's Code Signing Certificate and any applicable revocation date, whichever comes first. Signing Services MAY also timestamp signed objects.

### 8.2.2 Disclosure

Each CA, including Root CAs, MUST publicly disclose their policies and practices through an appropriate and readily accessible online means that is available on a 24x7 basis. The CA MUST publicly disclose its Certificate Practice Statement and/or Certificate Policies and structure the disclosures in accordance with either RFC 2527 or RFC 3647.

## *8.3 Commitment to Comply*

Each CA MUST give public effect to these Requirements and represent that they will adhere to the latest published version by either (i) incorporating the Requirements directly into their respective

Certification Practice Statements or (ii) by referencing the Requirements using a clause such as the following:

[Name of CA] conforms to the current version of the Minimum Requirements for the Issuance and Management of Publicly-Trusted Code Signing Certificates published at https://aka.ms/csbr. If there is any inconsistency between this document and those Requirements, those Requirements take precedence over this document.

In either case, each CA MUST include a link to the official version of these Requirements. In addition, each CA MUST include (directly or by reference) applicable parts of these Requirements in all contracts with Subordinate CAs, RAs, Signing Services and subcontractors, that involve or relate to the issuance or management of Certificates. CAs MUST enforce compliance with such terms.

### *8.4 Trust model*

Each CA MUST represent that it has disclosed all Cross Certificates in its Certificate Policy/Certificate Practice Statement that identify the CA as the Subject, provided that the CA arranged for or accepted the establishment of the trust relationship (i.e. the Cross Certificate at issue).

# 9. Certificate Content and Profile

### *9.1 Issuer Information*

As specified in BR Section 7.1.4.1.

### *9.2 Subject Information*

Code Signing Certificates issued to Subscribers MUST include the following information in the fields listed:

#### 9.2.1 Subject Alternative Name Extension

No Stipulation

#### 9.2.2 Subject Common Name Field

**Certificate Field**: subject:commonName (OID 2.5.4.3)

**Required/Optional**: Required

**Contents**: This field MUST contain the Subject's legal name as verified under BR Section 3.2.

#### 9.2.3 Subject Domain Component Field

This field MUST not be present in a Code Signing Certificate.

#### 9.2.4 Subject Distinguished Name Fields

a. **Certificate Field**: subject:organizationName (OID 2.5.4.10)

**Required/Optional**: Required.

**Contents**: The subject:organizationName field MUST contain either the Subject's name or DBA as verified under BR Section 3.2. The CA MAY include information in this field that differs slightly from the verified name, such as common variations or abbreviations, provided that the CA documents the difference and any abbreviations used are locally accepted abbreviations; e.g., if the official record shows "Company Name Incorporated", the CA MAY use "Company Name Inc." or "Company Name". Because subject name attributes for individuals (e.g. givenName (2.5.4.42) and surname (2.5.4.4)) are not broadly supported by application software, the CA MAY use the subject:organizationName field to convey a natural person Subject's name or DBA. The CA MUST have a documented process for verifying that the information included in the subject:organizationName field is not misleading to a Relying Party.

b. **Certificate Field**: Number and street: subject:streetAddress (OID: 2.5.4.9)

**Required/Optional**: Optional.

**Contents**: If present, the subject:streetAddress field MUST contain the Subject's street address information as verified under BR Section 3.2.2.1 or 3.2.3.

c. **Certificate Field**: subject:localityName (OID: 2.5.4.7)

**Required/Optional**: Required if the subject:stateOrProvinceName field is absent. Optional if the subject:stateOrProvinceName field is present.

**Contents**: If present, the subject:localityName field MUST contain the Subject's locality information as verified under BR Section 3.2. If the subject:countryName field specifies the ISO 3166-1 user-assigned code of XX in accordance with BR Section 7.1.4.2.2.g., the localityName field MAY contain the Subject's locality and/or state or province information as verified under BR Section 3.2.2.1 or 3.2.3.

d. **Certificate Field**: subject:stateOrProvinceName (OID: 2.5.4.8)

**Required/Optional**: Required if the subject:localityName field is absent. Optional if thesubject:localityName field is present.

**Contents**: If present, the subject:stateOrProvinceName field MUST contain the Subject's state or province information as verified under BR Section 3.2.2.1 or 3.2.3. If the subject:countryName field specifies the ISO 3166-1 user-assigned code of XX in accordance with BR Section 7.1.4.2.2.g., the subject:stateOrProvinceName field MAY contain the full name of the Subject's country information as verified under BR Section 3.2.2.3.

e. **Certificate Field**: subject:postalCode (OID: 2.5.4.17)

**Required/Optional**: Optional

**Contents**: If present, the subject:postalCode field MUST contain the Subject's zip or postal information as verified under BR Section 3.2.2.1 or 3.2.3.

f. **Certificate Field**: subject:countryName (OID: 2.5.4.6)

**Required/Optional**: Required

**Contents**: The subject:countryName MUST contain the two-letter ISO 3166-1 country code associated with the location of the Subject verified under BR Section 3.2.2.3. If a Country is not represented by an official ISO 3166-1 country code, the CA MAY specify the ISO 3166-1 user-assigned code of XX indicating that an official ISO 3166-1 alpha-2 code has not been assigned.

#### 9.2.5 Reserved

#### 9.2.6 Subject Organizational Unit Field

**Certificate Field**: subject:organizationalUnitName

**Required/Optional**: Optional.

**Contents**: The CA MUST implement a process that prevents an OU attribute from including a name, DBA, tradename, trademark, address, location, or other text that refers to a specific natural person or Legal Entity unless the CA has verified this information in accordance with BR Section 3.2.

#### 9.2.7 Reserved

#### 9.2.8 Other Subject Attributes

As specified in BR Section 7.1.4.2.2.i.

### *9.3 Certificate Policy Identification*

This section sets forth minimum requirements for the content of the Subscriber, Subordinate CA, and Root CA Certificates, as they relate to the identification of Certificate Policy.

#### 9.3.1 Certificate Policy Identifiers

The following Certificate Policy Identifier is reserved for use by CAs as a required means of asserting compliance with these Requirements as follows:

{joint-iso-itu-t(2) international-organizations(23) ca-browser-forum(140) certificate-policies(1) code-signing-requirements(4) code signing(1)} (2.23.140.1.4.1)

#### 9.3.2 Root CA Requirements

A Root CA Certificate SHOULD NOT contain the certificatePolicies extension.

#### 9.3.3 Subordinate CA Certificates

A Certificate issued after the Effective Date to a Subordinate CA that is not an Affiliate of the Issuing CA:

1. MUST include the policy identifier specified in Section 9.3.1 that indicates the Subordinate CA's adherence to and compliance with these Requirements (i.e. either the CA/Browser Forum reserved identifiers or identifiers defined by the CA in its Certificate Policy and/or Certification Practice Statement), and

2. MUST NOT contain the “anyPolicy” identifier (2.5.29.32.0).

A Certificate issued after the Effective Date to a Subordinate CA that is an affiliate of the Issuing CA:

1. MUST include the CA/Browser Forum reserved identifier specified in Section 9.3.1 to indicate the Subordinate CA’s compliance with these Requirements, and

2. MAY contain the “anyPolicy” identifier (2.5.29.32.0) in place of an explicit policy identifier.

A Subordinate CA MUST represent, in its Certificate Policy and/or Certification Practice Statement, that all Certificates containing a policy identifier indicating compliance with these Requirements are issued and managed in accordance with these Requirements.

#### 9.3.4 Subscriber Certificates

A Certificate issued to a Subscriber MUST contain one or more policy identifier(s), defined by the CA, in the Certificate’s certificatePolicies extension that indicates adherence to and compliance with these Requirements. CAs complying with these Requirements MAY also assert the reserved policy OIDs in such Certificates.

The CA MUST document in its Certificate Policy or Certification Practice Statement that the Certificates it issues containing the specified policy identifier(s) are managed in accordance with these Requirements.

### *9.4 Maximum Validity Period*

Subscribers and Signing Authorities MAY sign Code at any point in the development or distribution process. Code Signatures may be verified at any time, including during download, unpacking, installation, reinstallation, or execution, or during a forensic investigation.

The validity period for a Code Signing Certificate issued to a Subscriber or Signing Service MUST NOT exceed 39 months.

The Timestamp Authority MUST use a new Timestamp Certificate with a new private key no later than every 15 months to minimize the impact to users in the event that a Timestamp Certificate's private key is compromised. The validity for a Time Stamp Certificate must not exceed 135 months. The Timestamp Certificate MUST meet the "Minimum Cryptographic Algorithm and Key Size Requirements" in Appendix A for the communicated time period.

### *9.5 Subscriber Public Key*

The CA SHALL reject a certificate request if the requested Public Key does not meet the requirements set forth in Appendix A or if it has a known weak Private Key (such as a Debian weak key, see http://wiki.debian.org/SSLkeys).

### *9.6 Certificate Serial Number*

As specified in BR Section 7.1.

### *9.7 Reserved*

### *9.8 Reserved*

# 10. Certificate Request

## *10.1 Documentation Requirements*

As specified in BR Section 5.4.1.

## *10.2 Certificate Request*

### 10.2.1 General

Prior to the issuance of a Certificate, the CA MUST obtain from the Applicant a request for a certificate in a form prescribed by the CA and that complies with these Requirements. One request MAY suffice for multiple Certificates to be issued to the same Applicant, subject to the aging and updating requirement in Section 11.3, provided that each Certificate is supported by a valid, current request signed by the appropriate Applicant Representative on behalf of the Applicant. The request MAY be made, submitted and/or signed electronically.

Prior to signing an Object, the Signing Authority MUST obtain from the Applicant a signing request in a form prescribed by the Signing Authority and that complies with these Requirements. One signing request MAY suffice for multiple signatures for the same Applicant, subject to the requirements specified herein. The signing request MAY be made, submitted and/or signed electronically.

### 10.2.2 Request and Certification

The certificate requestor signing request MUST contain a request from, or on behalf of, the Applicant and a certification by, or on behalf of, the Applicant that all of the information contained therein is correct.

### 10.2.3 Information Requirements

The certificate request or signing request MAY include all factual information about the Applicant necessary to issue the Certificate or sign the Object, and such additional information as is necessary for the CA or Signing Authority to obtain from the Applicant in order to comply with these Requirements and the CA's Certificate Policy and/or Certification Practice Statement. In cases where the certificate request or signing request does not contain all the necessary information about the Applicant, the CA or Signing Service MUST obtain the remaining information from the Applicant or, having obtained it from a reliable, independent, third-party data source, confirm it with the Applicant. The CA or Signing Service MUST establish and follow a documented procedure for verifying all data requested for inclusion in the Certificate by the Applicant.

### 10.2.4 Subscriber Private Key

If the CA or any Delegated Third Party is generating the Private Key on behalf of the Subscriber where the Private Keys will be transported to the Subscriber outside of the Signing Service's secure infrastructure, then the entity generating the Private Key MUST either transport the Private Key in

hardware with an activation method that is equivalent to 128 bits of encryption or encrypt the Private Key with at least 128 bits of encryption strength. Allowed methods include using a 128-bit AES key to wrap the private key or storing the key in a PKCS 12 file encrypted with a randomly generated password of more than 16 characters containing uppercase letters, lowercase letters, numbers, and symbols for transport.

For Certificates transported outside of a Signing Service's secure infrastructure, the CA or Signing Service MUST require, by contract, each Subscriber to generate their own Private Key and protect the Private Key in accordance with Section 16.2 ("Private Key Protection").

## *10.3 Subscriber Agreement*

### 10.3.1 General

As specified in BR Section 9.6.3.

### 10.3.2 Agreement Requirements

The Applicant MUST make the following obligations and warranties through a Subscriber Agreement or Terms of Use:

1. **Accuracy of Information:** To provide accurate and complete information at all times in connection with the issuance of a Certificate, including in the Certificate Request and as otherwise requested by the CA.

2. **Protection of Private Key:** Where the key is available outside a Signing Service, to maintain sole control of, keep confidential, and properly protect, at all times in accordance with Section 16, the Private Key that corresponds to the Public Key to be included in the requested Certificate(s) (and any associated activation data or device, e.g. password or token). The CA MUST provide the Subscriber with documentation on how to protect a Private Key. The CA MAY provide this documentation as a white paper or as part of the Subscriber Agreement. The Subscriber MUST represent that it will generate and operate any device storing private keys in a secure manner, as described in a document of code signing best practices, which the CA MUST provide to the Subscriber during the ordering process. The CA MUST obligate the Subscriber to use passwords that are randomly generated with at least 16 characters containing uppercase letters, lowercase letters, numbers, and symbols to transport private keys.

3. **Private Key Reuse:** To not apply for a Code Signing Certificate if the Public Key in the Certificate is or will be used with a non-Code Signing Certificate.

4. **Use:** To use the Certificate and associated Private Key only for authorized and legal purposes, including not using the Certificate to sign Suspect Code and to use the Certificate and Private Key solely in compliance with all applicable laws and solely in accordance with the Subscriber Agreement or Terms of Use.

5. **Compliance with Industry Standards**: An acknowledgment and acceptance that the CA may modify the Subscriber Agreement or Terms of Use when necessary to comply with any changes in these Requirements or the Baseline Requirements.

6. **Prevention of Misuse:** To provide adequate network and other security controls to protect against misuse of the Private Key and that the CA will revoke the Certificate without requiring prior notification if there is unauthorized access to the Private Keys.

7. **Acceptance of Certificate:** Not to use the Certificate until after the Applicant, or an agent of Applicant, has reviewed and verified the Certificate contents for accuracy.

8. **Reporting and Revocation:** To promptly cease using a Certificate and its associated Private Key and promptly request that the CA revoke the Certificate if the Subscriber believes that (a) any information in the Certificate is, or becomes, incorrect or inaccurate, (b) the Private Key associated with the Public Key contained in the Certificate was misused or compromised, or (c) there is evidence that the Certificate was used to sign Suspect Code.

9. **Sharing of Information**: An acknowledgment and acceptance that, if: (a) the Certificate or the Applicant is identified as a source of Suspect Code, (b) the authority to request the Certificate cannot be verified, or (c) the Certificate is revoked for reasons other than Subscriber request (e.g. as a result of private key compromise, discovery of malware, etc.), then the CA is authorized to share information about the Applicant, signed application, Certificate, and surrounding circumstances with other CAs or industry groups, including the CA/Browser Forum.

10. **Termination of Use of Certificate:** To promptly cease using the Private Key corresponding to the Public Key listed in a Certificate upon expiration or revocation of the Certificate.

11. **Acknowledgment and Acceptance:** An acknowledgement and acceptance that the CA is entitled to revoke the certificate immediately if the Applicant were to violate the Terms of Use or the Subscriber Agreement.

#### 10.3.3 Service Agreement Requirements for Signing Authorities

The CA MUST contractually obligate each Signing Service to inform the CA if the Signing Service becomes aware (by whatever means) that the Signing Service has signed Suspect Code. The CA MUST require the Signing Service to request revocation of the affected Certificate and provide immediate notice to the CA if the Signing Service's private key, or private key activation data, is compromised or believed to be compromised. The CA MUST revoke the affected Certificate upon request by the Signing Service or if the CA determines the Signing Service failed to notify the CA within 24 hours after identifying a private key compromise.

Signing Authorities MUST obtain the Subscriber's commitment to:

1. Use such signing services solely for authorized purposes that comply with the Subscriber Agreement/Terms of Use, these Requirements, and all applicable laws,

2. Not knowingly submit software for signature that contains Suspect Code, and

3. Inform the Signing Service if it is discovered (by whatever means) that code submitted to the Signing Service for signature contained Suspect Code.

# 11. Verification Practices

## *11.1 Verification of Organizational Applicants*

Prior to issuing a Code Signing Certificate to an Organizational Applicant, the Issuer MUST:

1. Verify the Subject's legal identity, including any DBA proposed for inclusion in a Certificate, in accordance with Section 11.1.1 and 11.1.2 of this document,

2. Verify the Subject's address in accordance with Section 11.1.1 of this document,

3. Verify the Certificate Requester's authority to request a Code Signing Certificate and the authenticity of the Certificate Request using a Reliable Method of Communication in accordance with BR Section 3.2.5., and

4. If the Subject's or Subject's Affiliate's, Parent Company's, or Subsidiary Company's date of formation, as indicated by either a QIIS or QGIS, was less than three years prior to the date of the Certificate Request, verify the identity of the Certificate Requester.

### 11.1.1 Organization Identity and Address

As specified in BR Section 3.2.2.1. The CA MUST also obtain, whenever available, a specific Registration Identifier assigned to the Applicant by a government agency in the jurisdiction of the Applicant's legal creation, existence, or recognition.

### 11.1.2 DBA/Tradename

As specified in BR Section 3.2.2.2.

### 11.1.3 Requester Authority

As specified in BR Section 3.2.5.

## *11.2 Verification of Individual Applicants*

Prior to issuing a Code Signing Certificate to an Individual Applicant, the CA MUST:

1. Verify the Subject's identity under Section 11.2.1 of this document, and

2. Verify the authenticity of the identity under Section 11.2.2 of this document.

### 11.2.1 Individual Identity

The CA MUST verify the Applicant's identity using one of the following processes:

1. The CA MUST obtain a legible copy, which discernibly shows the Requester's face, of at least one currently valid government-issued photo ID (passport, driver's license, military ID, national ID, or equivalent document type). The CA MUST inspect the copy for any indication of alteration or falsification. The CA MUST also verify the address of the Requester using (i) a government-issued photo ID, (ii) a QIIS or QGIS, or (iii) an access code

to activate the Certificate where the access code was physically mailed to the Requester; OR

2. The CA MUST have the Requester digitally sign the Certificate Request using a valid personal Certificate that was issued under one of the following adopted standards: Qualified Certificates issued pursuant to ETSI TS 101 862, IGTF, Adobe Signing Certificate issued under the AATL or CDS program, the Kantara identity assurance framework at level 2, NIST SP 800-63 at level 2, or the FBCA CP at Basic or higher assurance.

#### 11.2.2 Authenticity of Identity

The CA MUST verify the authenticity of the Certificate Request using one of the following:

1. Having the Requester provide a photo of the Requester holding the submitted government-issued photo ID where the photo is of sufficient quality to read both the name listed on the photo ID and the issuing authority; OR

2. Having the CA perform an in-person or web camera-based verification of the Requester where an employee or contractor of the CA can see the Requester, review the Requester's photo ID, and confirm that the Requester is the individual identified in the submitted photo ID; OR

3. Having the CA obtain an executed Declaration of Identity of the Requester that includes at least one unique biometric identifier (such as a fingerprint or handwritten signature). The CA MUST confirm the document's authenticity directly with the Verifying Person using contact information confirmed with a QIIS or QGIS; OR

4. Verifying that the digital signature used to sign the Request under Section 11.2.1(2) is a valid signature and originated from a Certificate issued at the appropriate level of assurance as evidenced by the certificate chain. Acceptable verification under this section includes validation that the Certificate was issued by a CA qualified by the entity responsible for adopting, enforcing, or maintaining the adopted standard and chains to an intermediate certificate or root certificate designated as complying with such standard.

### *11.3 Age of Certificate Data*

As specified in BR Section 3.3.1.

### *11.4 Denied List*

As specified in BR Section 4.1.1.

### *11.5 High Risk Certificate Requests*

In addition to the procedures required by BR Section 4.2.1, prior to issuing a Code Signing Certificate, each CA SHOULD check at least one database containing information about known or suspected producers, publishers, or distributors of Suspect Code, as identified or indicated by an Anti-Malware Organization and any database of deceptive names maintained by an Application Software Provider. The CA MUST determine whether the entity is identified as requesting a Code Signing Certificate from a High Risk Region of Concern. The CA MUST also maintain and check an

internal database listing Certificates revoked due to Signatures on Suspect Code and previous certificate requests rejected by the CA.

A CA identifying a high risk application under this section MUST follow the additional procedures defined in Section 11.7 of this document to ensure that the applicant will protect its Private Keys and not sign Suspect Code.

[These requirements do not specify a particular database and leave the decision of qualifying databases to the implementers.]

### *11.6 Data Source Accuracy*

As specified in BR Section 3.2.2.7.

### *11.7 Processing High Risk Applications*

CAs MUST not issue new or replacement Code Signing Certificates to an entity that the CA determined intentionally signed Suspect Code. The CA MUST keep meta-data about the reason for revoking a Code Signing Certificate as proof that the Code Signing Certificate was not revoked because the Applicant was intentionally signing Suspect Code.

CAs MAY issue new or replacement Code Signing Certificates to an entity who is the victim of a documented Takeover Attack, resulting in either a loss of control of their code-signing service or loss of the Private Key associated with their Code Signing Certificate.

If the CA is aware that the Applicant was the victim of a Takeover Attack, the CA MUST verify that the Applicant is protecting its Code Signing Private Keys under Section 16.3(1) or Section 16.3(2). The CA MUST verify the Applicant's compliance with Section 16.3(1) or Section 16.3(2) (i) through technical means that confirm the Private Keys are protected using the method described in 16.3(1) or 16.3.2(2) or (ii) by relying on a report provided by the Applicant that is signed by an auditor who is approved by the CA and who has IT and security training or is a CISA.

Documentation of a Takeover Attack MAY include a police report (validated by the CA) or public news report that admits that the attack took place. The Subscriber MUST provide a report from an auditor with IT and security training or a CISA that provides information on how the Subscriber was storing and using Private keys and how the intended solution for better security meets the guidelines for improved security.

Except where issuance is expressly authorized by the Application Software Supplier, CAs MUST not issue new Code Signing Certificates to an entity where the CA is aware that the entity has been the victim of two Takeover Attacks or where the CA is aware that entity breached a requirement under this Section to protect Private Keys under either Section 16.3(1) or 16.3(2).

### *11.8 Due Diligence*

1. The results of the verification processes and procedures outlined in these Requirements are intended to be viewed both individually and as a group. Thus, after all of the verification processes and procedures are completed, the CA MUST have a person who is not responsible for the collection of information review all of the information and

documentation assembled in support of the Code Signing Certificate application and look for discrepancies or other details requiring further explanation.

2. The CA MUST obtain and document further explanation or clarification from Applicant and other sources of information, as necessary, to resolve those discrepancies or details that require further explanation.

3. The CA MUST refrain from issuing a Code Signing Certificate until all of the information and documentation assembled in support of the Certificate is such that issuance of the Certificate will not communicate factual information that the CA knows, or with the exercise of due diligence should discover from the assembled information and documentation, to be inaccurate. If satisfactory explanation and/or additional documentation are not received within a reasonable time, the CA MUST decline the Certificate request and SHOULD notify the Applicant accordingly.

# 12. Certificate Issuance by a Root CA

Certificate issuance by the Root CA MUST require an individual authorized by the CA (i.e. the CA system operator, system officer, or PKI administrator) to deliberately issue a direct command in order for the Root CA to perform a certificate signing operation.

Root CA Private Keys MUST NOT be used to directly sign Certificates.

# 13. Certificate Revocation and Status Checking

## *13.1 Revocation*

### 13.1.1 Revocation Request

As specified in BR Section 4.9.3.

### 13.1.2 Certificate Problem Reporting

The CA MUST provide Anti-Malware Organizations, Subscribers, Relying Parties, Application Software Suppliers, and other third parties with clear instructions on how they can report suspected Private Key Compromise, Certificate misuse, Certificates used to sign Suspect Code, Takeover Attacks, or other types of possible fraud, compromise, misuse, inappropriate conduct, or any other matter related to Certificates. The CA MUST publicly disclose the instructions on its website.

### 13.1.3 Investigation

The CA MUST begin investigating Certificate Problem Reports within twenty-four hours of receipt, and decide whether revocation or other appropriate action is warranted based on at least the following criteria:

1. The nature of the alleged problem (adware, spyware, malware, software bug, etc.),
2. The number of Certificate Problem Reports received about a particular Certificate or Subscriber,

3. The entity making the report (for example, a notification from an Anti-Malware Organization or law enforcement agency carries more weight than an anonymous complaint), and
4. Relevant legislation.

### 13.1.4 Response

The CA MUST maintain a continuous 24x7 ability to communicate with Anti-Malware Organizations, Application Software Suppliers, and law enforcement agencies and respond to high-priority Certificate Problem Reports, such as reports requesting revocation of Certificates used to sign malicious code, fraud, or other illegal conduct.

The CA MUST acknowledge receipt of plausible notices about Suspect Code signed with a certificate issued by the CA or a Subordinate CA.

### 13.1.5 Reasons for Revoking a Subscriber Certificate

A CA MUST revoke a Code Signing Certificate in any of the four circumstances: (1) the Application Software Supplier requests revocation, (2) the subscriber requests revocation, , (3) a third party provides information that leads the CA to believe that the certificate is compromised or is being used for Suspect Code, or (4) the CA otherwise decides that the certificate should be revoked. This section describes the CA's obligations for each scenario.

13.1.5.1 Revocation Based on an Application Software Supplier's Request

If the Application Software Supplier requests the CA revoke because the Application Software Supplier believes that a Certificate attribute is deceptive, or that the Certificate is being used for malware, bundle ware, unwanted software, or some other illicit purpose, then the Application Software Supplier may request that the CA revoke the certificate.

Within two (2) business days of receipt of the request, the CA MUST either revoke the certificate or inform the Application Software Supplier that it is conducting an investigation.

If the CA decides to conduct an investigation, it MUST inform the Application Software Supplier whether or not it will revoke the Certificate, within two (2) business days.

If the CA decides that the revocation will have an unreasonable impact on its customer, then the CA MUST propose an alternative course of action to the Application Software Supplier based on its investigation.

13.1.5.2 Revocation Based on the Subscriber's Request

The CA MUST revoke a Code Signing Certificate within one (1) business day if the Subscriber requests in writing that the CA revoke the Certificate or notifies the CA that the original certificate request was not authorized and does not retroactively grant authorization.

13.1.5.3 Revocation Based on Reported or Detected Compromise or Use in Malware

For all incidents involving malware, CAs SHALL revoke the Code Signing Certificate in accordance with and within the following maximum timeframes. Nothing herein prohibits a CA from revoking a Code Signing Certificate prior to these timeframes.

1) The CA MUST contact the software publisher within one (1) business day after the CA is made aware of the incident.

2) The CA MUST determine the volume of relying parties that are impacted (e.g., based on OCSP logs) within 72 hours after being made aware of the incident.

3) The CA MUST request the software publisher send an acknowledgement to the CA within 72 hours of receipt of the request.

a. If the publisher responds within 72 hours, the CA and publisher MUST determine a "reasonable date" to revoke the certificate based on discussions with the CA.

b. If CA does not receive a response, the CA must notify the publisher that the CA will revoke in 7 days if no further response is received.

i. If the publisher responds within 7 days, the CA and the publisher will determine a "reasonable date" to revoke the certificate based on discussion with the CA.

ii. If no response is received after 7 days, the CA must revoke the certificate except if the CA has documented proof (e.g., OCSP logs) that this will cause significant impact to the general public.

A CA revoking a Certificate because the Certificate was associated with signed Suspect Code or other fraudulent or illegal conduct SHOULD provide all relevant information and risk indicators to other CAs or industry groups. The CA SHOULD indicate whether its investigation found that the Suspect Code was a false positive or an inadvertent signing.

#### 13.1.6 Reasons for Revoking a Subordinate CA Certificate

As specified in BR Section 4.9.1.2.

#### 13.1.7 Certificate Revocation Date

When revoking a Certificate, the CA SHOULD work with the Subscriber to estimate a date of when the revocation should occur in order to mitigate the impact of revocation on validly signed Code. For key compromise events, this date SHOULD be the earliest date of suspected compromise.

### *13.2 Certificate Status Checking*

#### 13.2.1 Mechanisms

In addition to the requirements specified in BR Section 4.9.7 through 4.9.10, CAs MUST provide up-to-date revocation status information. CAs MUST provide OCSP responses for Code Signing Certificates and Timestamp Certificates for the time period specified in their CPS, which MUST be at least 10 years after the expiration of the certificate. If a CA issues CRLs, the serial number of a revoked certificate MUST remain on the CRL for at least 10 years after the expiration of the certificate. Application Software Suppliers MAY require the CA to support a longer life-time in its contract with the CA. If the CA wishes to stop supporting validation of Code Signing Certificates or

Timestamp Certificates prior to the date specified in its Certificate Policy/Certificate Practice Statement, the CA MUST give 90 days' prior notice to all Application Software Suppliers relying on the root certificate and permit the Application Software Suppliers sufficient time to take appropriate action as determined by the Application Software Supplier.

If a Code Signing Certificate contains the Lifetime Signing OID, the Signature becomes invalid when the Code Signing Certificate expires, even if the Signature is timestamped. Because the Lifetime Signing OID is intended to be used with test purposes only, a CA MAY cease maintaining revocation information for a Code Signing Certificate with the Lifetime Signing OID after the Code Signing Certificate expires.

Whenever practical, Platforms should check the revocation status of the Certificates that they rely upon. However, this is not always practical, such as when signed Code is loaded earlier in the boot sequence than the network communication stack.

In the timestamp model, the Platform should check the revocation status at the time the time-stamp was applied. In addition to checking revocation status, where practical, Platforms should consult blacklists for Suspect Code.

A Certificate MAY have a one-to-one relationship or one-to-many relationship with the signed Code. Regardless, revocation of a Certificate may invalidate the signatures on all those signed Objects, some of which could be perfectly sound. Because of this, the CA MAY specify a revocation date in a CRL or OCSP response to time-bind the set of software affected by the revocation, and software should continue to treat objects containing a time-stamp dated before the revocation date as valid.

Because some Application Software Suppliers utilize non-standard revocation mechanisms, CAs MUST, if requested by the Application Software Supplier and using a method of communication specified by the Application Software Vendor, notify the Application Software Supplier whenever the CA revokes a Code Signing Certificate because (i) the CA mis-issued the Certificate, (ii) the Certificate was used to sign Suspect Code, or (iii) there is a suspected or actual compromise of the Applicant's or CA's Private Key.

### 13.2.2 Repository

The CA SHALL maintain an online 24x7 Repository that application software can use to automatically check the current status of Code Signing and Timestamp Certificates issued by the CA.

For the status of Code Signing Certificates:

1. If the CA publishes a CRL, then the CA SHALL update and reissue CRLs at least once every seven days, and the value of the nextUpdate field MUST NOT be more than ten days beyond the value of the thisUpdate field; and

2. The CA SHALL update information provided via an Online Certificate Status Protocol at least every four days. OCSP responses from this service MUST have a maximum expiration time of ten days.

For the status of Timestamp Certificates:

1. The CA SHALL update and reissue CRLs at least (i) once every twelve months and (ii) within 24 hours after revoking a Timestamp Certificate, and the value of the nextUpdate field MUST NOT be more than twelve months beyond the value of the thisUpdate field; and

2. The CA SHALL update information provided via an Online Certificate Status Protocol at least (i) every twelve months and (ii) within 24 hours after revoking a Subordinate CA Certificate.

The CA SHALL support an OCSP capability using the GET method for Certificates issued in accordance with these Requirements.

# 14. Employees and Third Parties

## *14.1 Trustworthiness and Competence*

As specified in BR Section 5.3.

## *14.2 Delegation of Functions to Registration Authorities and Subcontractors*

### 14.2.1 General

Except as stated in Section 14.2.2 of this document, the CA MAY delegate the performance of all, or any part, of these Requirements to a Delegated Third Party, provided that the process as a whole fulfills all of the requirements of this document.

Before the CA authorizes a Delegated Third Party to perform a delegated function, the CA MUST contractually require the Delegated Third Party to:

1. Meet the qualification requirements of BR Section 5.3 when applicable to the delegated function,
2. Retain documentation in accordance with BR Section 5.4.1,
3. Abide by the other provisions of these Requirements that are applicable to the delegated function, and
4. Comply with (a) the CA's Certificate Policy/Certification Practice Statement or (b) the Delegated Third Party's practice statement that the CA has verified complies with these Requirements.

The CA MUST verify that the Signing Service and any other Delegated Third Party's personnel involved in the issuance of a Certificate meet the training and skills requirements of Section 14 of this document and the document retention and event logging requirements of Section 15 of this document.

If a Delegated Third Party fulfills any of the CA's obligations under Section 11.5 (High Risk Requests) of this document, the CA MUST verify that the process used by the Delegated Third Party to identify and further verify High Risk Certificate Requests provides at least the same level of assurance as the CA's own processes.

### 14.2.2 Compliance Obligation

In all cases, the CA MUST contractually obligate each Delegated Third Party to comply with all applicable requirements in these Requirements and to perform them as required of the CA itself. The CA MUST enforce these obligations and internally audit each Delegated Third Party's compliance with these Requirements on an annual basis.

### 14.2.3 Allocation of Liability

As specified in Section BR Sections 9.8 and 9.9.

# 15. Data Records

The Timestamp Authority MUST log the following information:

1. All data related to the creation of a timestamp, including all requests for a time-stamp, the connecting IP, and results of the timestamp,
2. Physical or remote access to a timestamp server, including the time of the access and the identity of the individual accessing the server,
3. History of the timestamp server configuration,
4. Any attempt to delete or modify timestamp logs,
5. Security events, including:

   a. Successful and unsuccessful PKI system access attempts;

   b. PKI and security system actions performed;

   c. Security profile changes;

   d. System crashes, hardware failures, and other anomalies;

   e. Firewall and router activities; and

   f. Entries to and exits from the CA facility

1. Revocation of a timestamp certificate,
2. Major changes to the timestamp server's time,
3. System startup and shutdown, and
4. Equipment failures or malfunctions.

   Data MUST be retained as specified in BR Section 5.4.3. except for item number 1 above which MUST be retained for a minimum of 5 days.

# 16. Data Security and Private Key Protection

The requirements in BR Sections 6.1 and 6.2 apply equally to Code Signing Certificates. In addition:

## *16.1 Timestamp Authority Key Protection*

1. Each CA MUST operate an RFC-3161-compliant Timestamp Authority that is available for use by customers of its Code Signing Certificates. CAs MUST recommend to Subscribers that they use the CA's Timestamping Authority to time-stamp signed code.

2. A Timestamp Authority MUST protect its signing key using a process that is at least to FIPS 140-2 Level 3, Common Criteria EAL 4+ (ALC_FLR.2), or higher. The CA MUST protect its signing operations in accordance with the CA/Browser Forum's Network Security Guidelines. Any changes to its signing process MUST be an auditable event.

3. The Timestamp Authority MUST ensure that clock synchronization is maintained when a leap second occurs. A Timestamp Authority MUST synchronize its timestamp server at least every 24 hours with a UTC(k) time source. The timestamp server MUST automatically detect and report on clock drifts or jumps out of synchronization with UTC. Clock adjustments of one second or greater MUST be auditable events.

## *16.2 Signing Service Requirements*

The Signing Service MUST ensure that a Subscriber's private key is generated, stored, and used in a secure environment that has controls to prevent theft or misuse. A Signing Service MUST enforce multi-factor authentication to access and authorize Code Signing and obtain a representation from the Subscriber that they will securely store the tokens required for multi-factor access. A system used to host a Signing Service MUST NOT be used for web browsing. The Signing Service MUST run a regularly updated antivirus solution to scan the service for possible virus infection. The Signing Service MUST comply with the Network Security Guidelines as a "Delegated Third Party".

## *16.3 Subscriber Private Key Protection*

The CA MUST obtain a representation from the Subscriber that the Subscriber will use one of the following options to generate and protect their Code Signing Certificate private keys:

1. A Trusted Platform Module (TPM) that generates and secures a key pair and that can document the Subscriber's private key protection through a TPM key attestation.

2. A hardware crypto module with a unit design form factor certified as conforming to at least FIPS 140 Level 2, Common Criteria EAL 4+, or equivalent.

3. Another type of hardware storage token with a unit design form factor of SD Card or USB token (not necessarily certified as conformant with FIPS 140 Level 2 or Common Criteria EAL 4+). The Subscriber MUST also warrant that it will keep the token physically separate from the device that hosts the code signing function until a signing session is begun.

A CA MUST recommend that the Subscriber protect Private Keys using the method described in Section 16.3(1) or 16.3(2) over the method described in Section 16.3(3) and obligate the Subscriber to protect Private Keys in accordance with 10.3.2(2).

# 17. Audit

## *17.1 Eligible Audit Schemes*

The CA MUST undergo a conformity assessment audit for compliance with these Requirements performed in accordance with one of the following schemes:

1. WebTrust for Certification Authorities v2.0;

2. A national scheme that audits conformance to ETSI TS 102 042;

Whichever scheme is chosen, it MUST incorporate periodic monitoring and/or accountability procedures to ensure that its audits continue to be conducted in accordance with the requirements of the scheme.

The audit MUST be conducted by a Qualified Auditor, as specified in BR Section 8.2.

## *17.2 Audit Period*

As specified in BR Section 8.1.

## *17.3 Audit Report*

As specified in BR Section 8.6.

## *17.4 Pre-Issuance Readiness Audit*

If the CA has a currently valid Audit Report indicating compliance with an audit scheme listed in Section 17.1, then no pre-issuance readiness assessment is necessary.

If the CA does not have a currently valid Audit Report indicating compliance with one of the audit schemes listed in Section 17.1, then, before issuing Publicly-Trusted Certificates, the CA MUST successfully complete a point-in-time readiness assessment performed in accordance with applicable standards under one of the audit schemes listed in Section 17.1. The point-in-time readiness assessment MUST be completed no earlier than twelve (12) months prior to issuing Publicly-Trusted Certificates and MUST be followed by a complete audit under such scheme within ninety (90) days of issuing the first Publicly-Trusted Certificate.

## *17.5 Audit of Delegated Functions*

Audits MUST be conducted for all obligations under these Guidelines, including timestamping and signing services, regardless of whether they are performed directly by the CA or by a Delegated Third Party. Functions performed by a Delegated Third Party MUST be included in the CA's audit or the CA MUST obtain an audit report from the Delegated Third Party. If the opinion is that the Delegated Third Party does not comply, then the CA MUST not allow the Delegated Third Party to continue performing delegated functions.

The audit period for the Delegated Third Party MUST NOT exceed one year (ideally aligned with the CA's audit).

### *17.6 Auditor Qualifications*

As specified in BR Section 8.2.

### *17.7 Key Generation Ceremony*

As specified in BR Section 6.1.1.1.

## 18. Liability and Indemnification

As specified in BR Section 9.

# Appendix A

# Minimum Cryptographic Algorithm and Key Size Requirements

Certificates and Timestamp tokens issued after the effective date of these guidelines MUST meet the following requirements for algorithm type and key size.

**(1) Code Signing Root, Subordinate CA, and Code Signing Certificates**

The table below defines cryptographic requirements for Code Signing Certificates issued within the specified time and their corresponding Root Certificates and Subordinate CA Certificates.

**Note: The digest algorithm used to issue the Root Certificate does not have security relevance and need not conform to the requirements in the table below.**

| | Code Signing Certificates issued prior to January 1, 2021and their corresponding Root Certificates and Subordinate CA Certificates | Code Signing Certificates issued on or after January 1, 2021 and their corresponding Root Certificates and Subordinate CA Certificates |
|---|---|---|
| Digest algorithm | SHA-256, SHA-384 or SHA-512 (SHA-1 for legacy implementations only)* | SHA-256, SHA-384 or SHA-512 |
| Minimum RSA modulus size (bits) | 2048 | 3072 |
| ECC curve | NIST P-256, P-384, or P-521 | NIST P-256, P-384, or P-521 |
| Minimum DSA modulus and divisor size (bits) | L= 2048, N= 224 or L= 2048, N= 256 | L= 2048, N= 224 or L= 2048, N= 256 |

*****CAs can issue SHA-1 certificates to legacy platforms that do not support SHA-2 only for code signing and timestamping certificates.

**(2) Timestamp Root, Subordinate CA, and Timestamp Certificates**

The table below defines cryptographic requirements for Timestamp Certificates issued within the specified time and their corresponding Root Certificates and Subordinate CA Certificates.

**Note: The digest algorithm used to issue the Root Certificate does not have security relevance and need not conform to the requirements in the table below.**

| | Timestamp Certificates issued prior to January 1, 2021 and their corresponding Root Certificates and Subordinate CA Certificates | Timestamp Certificates issued on or after January 1, 2021 and their corresponding Root Certificates and Subordinate CA Certificates |
|---|---|---|

| | | |
|---|---|---|
| Digest algorithm | SHA-256, SHA-384 or SHA-512 (SHA-1 for legacy implementations only)* | SHA-256, SHA-384 or SHA-512 |
| Minimum RSA modulus size (bits) | 2048 | 3072 |
| ECC curve | NIST P-256, P-384, or P-521 | NIST P-256, P-384, or P-521 |
| Minimum DSA modulus and divisor size (bits) | L= 2048, N= 224 or L= 2048, N= 256 | L= 2048, N= 224 or L= 2048, N= 256 |

*****CAs can issue SHA-1 certificates to legacy platforms that do not support SHA-2 only for code signing and timestamping certificates.

### (3) Timestamp Tokens

The digest algorithms used to sign Timestamp tokens must match the digest algorithm used to sign the Timestamp Certificate.

| | Generated prior to January 1, 2021 | Generated on or after January 1, 2021 |
|---|---|---|
| Digest algorithm | SHA-256, SHA-384 or SHA-512 (SHA-1 for legacy implementations only)* | SHA-256, SHA-384 or SHA-512 |

*****CAs can issue SHA-1 certificates to legacy platforms that do not support SHA-2 only for code signing and timestamping certificates.

# Appendix B

# Certificate Extensions (Normative)

This appendix specifies the requirements for extensions in Certificates issued after the date of these guidelines (including Subordinate CA certificates)

**(1) Root CA Certificates**

As specified in Appendix A of the Baseline Requirements.

**(2) Certificates for Subordinate CAs issuing Code Signing Certificates**

A. certificatePolicies

This extension MUST be present and SHOULD NOT be marked critical.

certificatePolicies:policyIdentifier (Required)

If the certificate is issued to a Subordinate CA that is not an Affiliate of the entity that controls the Root CA, then the set of policy identifiers MUST include a Policy Identifier, defined by the Subordinate CA, which indicates a Certificate Policy asserting the Subordinate CA's adherence to and compliance with these Requirements.

The following fields MUST be present if the Subordinate CA is not an Affiliate of the entity that controls the Root CA.

certificatePolicies:policyQualifiers:policyQualifierId

- id-qt 1 [RFC 5280]

certificatePolicies:policyQualifiers:qualifier:cPSuri

- HTTP URL for the Root CA's Certification Practice Statement

B. cRLDistributionPoint

This extension MUST be present, MUST NOT be marked critical, and MUST contain the HTTP URL of the CA's CRL service.

C. authorityInformationAccess

This extension MUST be present and MUST NOT be marked critical. The extension MUST contain the HTTP URL of the CA's OCSP responder (accessMethod = 1.3.6.1.5.5.7.48.1), and/or the HTTP URL for the Root CA's certificate (accessMethod = 1.3.6.1.5.5.7.48.2).

D. basicConstraints

This extension MUST appear as a critical extension in all CA certificates that contain Public Keys used to validate digital signatures on certificates. The cA field MUST be set true. The pathLenConstraint field MAY be present.

E. keyUsage

This extension MUST be present and MUST be marked critical. Bit positions for keyCertSign and cRLSign MUST be set. If the Subordinate CA Private Key is used for signing OCSP responses, then the digitalSignature bit MUST be set.

F. extkeyUsage (EKU)

The id-kp-codeSigning [RFC5280] value MUST be present.

The following EKUs MAY be present: documentSigning and emailProtection.

The value anyExtendedKeyUsage (2.5.29.37.0) or serverAuth (1.3.6.1.5.5.7.3.1) MUST NOT be present.

Other values SHOULD NOT be present. If any other value is present, the CA MUST have a business agreement with a Platform vendor requiring that EKU in order to issue a Platform-specific code signing certificate with that EKU.

This extension SHOULD be marked non-critical.

The CA MUST set all other fields and extensions in accordance to RFC 5280.

**(3) Code Signing Certificates**

A. certificatePolicies

This extension MUST be present and SHOULD NOT be marked critical.

certificatePolicies:policyIdentifier (Required)

- A Policy Identifier, defined by the Issuer, that indicates a Certificate Policy asserting the Issuer's adherence to and compliance with these Requirements.

certificatePolicies:policyQualifiers:policyQualifierId (Recommended)

- id-qt 1 [RFC 5280]

certificatePolicies:policyQualifiers:qualifier:cPSuri (Optional)

- HTTP URL for the Subordinate CA's Certification Practice Statement

B. cRLDistributionPoint

This extension MAY be present. If present, it MUST NOT be marked critical, and it MUST contain the HTTP URL of the CA's CRL service.

C. authorityInformationAccess

This extension MUST be present and MUST NOT be marked critical. The extension MUST contain the HTTP URL of the CA's OCSP responder (accessMethod = 1.3.6.1.5.5.7.48.1) and the HTTP URL for the Root CA's certificate (accessMethod = 1.3.6.1.5.5.7.48.2).

D. basicConstraints (optional)

   If present, the cA field MUST be set false.

E. keyUsage (required)

   This extension MUST be present and MUST be marked critical. The bit positions for digitalSignature MUST be set. Bit positions for keyCertSign and cRLSign MUST NOT be set. All other bit positions SHOULD NOT be set.

F. extKeyUsage (EKU) (required)

   The value id-kp-codeSigning [RFC5280] MUST be present.

   The following EKUs MAY be present: documentSigning, lifetimeSigning, and emailProtection.

   The value anyExtendedKeyUsage (2.5.29.37.0) or serverAuth (1.3.6.1.5.5.7.3.1) MUST NOT be present.

   Other values SHOULD NOT be present. If any other value is present, the CA MUST have a business agreement with a Platform vendor requiring that EKU in order to issue a Platform-specific code signing certificate with that EKU.

The CA MUST set all other fields and extensions in accordance to RFC 5280.

**(4) Certificates for Subordinate CAs issuing Timestamp Certificates**

A. certificatePolicies

   This extension MUST be present and SHOULD NOT be marked critical.

   certificatePolicies:policyIdentifier (Required)

   If the certificate is issued to a Subordinate CA that is not an Affiliate of the entity that controls the Root CA, then the set of policy identifiers MUST include a Policy Identifier, defined by the Subordinate CA, which indicates a Certificate Policy asserting the Subordinate CA's adherence to and compliance with these Requirements.

   The following fields MUST be present if the Subordinate CA is not an Affiliate of the entity that controls the Root CA.

   certificatePolicies:policyQualifiers:policyQualifierId

   - id-qt 1 [RFC 5280]

   certificatePolicies:policyQualifiers:qualifier:cPSuri

   - HTTP URL for the Root CA's Certification Practice Statement

B. cRLDistributionPoint

This extension MUST be present, MUST NOT be marked critical, and MUST contain the HTTP URL of the CA's CRL service.

C. authorityInformationAccess

This extension MUST be present and MUST NOT be marked critical. The extension MUST contain the HTTP URL of the CA's OCSP responder (accessMethod = 1.3.6.1.5.5.7.48.1), and/or the HTTP URL for the Root CA's certificate (accessMethod = 1.3.6.1.5.5.7.48.2).

D. basicConstraints

This extension MUST appear as a critical extension in all CA certificates that contain Public Keys used to validate digital signatures on certificates. The cA field MUST be set true. The pathLenConstraint field MAY be present.

E. keyUsage

This extension MUST be present and MUST be marked critical. Bit positions for keyCertSign and cRLSign MUST be set. If the Subordinate CA Private Key is used for signing OCSP responses, then the digitalSignature bit MUST be set.

F. extkeyUsage (EKU)

The id-kp-timeStamping [RFC5280] value MUST be present.

The value anyExtendedKeyUsage (2.5.29.37.0) MUST NOT be present.

Other values SHOULD NOT be present. If any other value is present, the CA MUST have a business agreement with a Platform vendor requiring that EKU in order to issue a Platform-specific code signing certificate with that EKU.

This extension SHOULD be marked non-critical.

The CA MUST set all other fields and extensions in accordance to RFC 5280.

**(5) Timestamp Certificates**

A. certificatePolicies

This extension MUST be present and SHOULD NOT be marked critical.

certificatePolicies:policyIdentifier (Required)

- A Policy Identifier, defined by the Issuer, that indicates a Certificate Policy asserting the Issuer's adherence to and compliance with these Requirements.

certificatePolicies:policyQualifiers:policyQualifierId (Recommended)

- id-qt 1 [RFC 5280]

certificatePolicies:policyQualifiers:qualifier:cPSuri (Optional)

- HTTP URL for the Subordinate CA's Certification Practice Statement

B. cRLDistributionPoint

   This extension MAY be present. If present, it MUST NOT be marked critical, and it MUST contain the HTTP URL of the CA's CRL service.

C. authorityInformationAccess

   This extension MUST be present and MUST NOT be marked critical. The extension MUST contain the HTTP URL of the CA's OCSP responder (accessMethod = 1.3.6.1.5.5.7.48.1) and the HTTP URL for the Root CA's certificate (accessMethod = 1.3.6.1.5.5.7.48.2).

D. basicConstraints (optional)

   If present, the cA field MUST be set false.

E. keyUsage (required)

   This extension MUST be present and MUST be marked critical. The bit positions for digitalSignature MUST be set. Bit positions for keyCertSign and cRLSign MUST NOT be set. All other bit positions SHOULD NOT be set.

F. extKeyUsage (EKU) (required)

   The value id-kp-timeStamping [RFC5280] MUST be present and MUST be marked critical.

   The value anyExtendedKeyUsage (2.5.29.37.0) MUST NOT be present.

   Other values SHOULD NOT be present. If any other value is present, the CA MUST have a business agreement with a Platform vendor requiring that EKU in order to issue a Platform-specific code signing certificate with that EKU.

The CA MUST set all other fields and extensions in accordance to RFC 5280.

# Appendix C

# User Agent Verification (Normative)

As specified in Appendix C of the Baseline Requirements.

# APPENDIX D

# HIGH RISK REGIONS OF CONCERN

The geographic locations listed below have more than 5% of the Code Signing Certificates for that location associated with signed Suspect Code when compared to the number of all Code Signing Certificates for that area. Applications originating or associated from one of these HRRCs are considered high risk and require additional verification as specified under Section 11.7 of this document:

NONE