Mozilla Root Store Policy

Version 2.9

Effective September 1, 2023 (https://wiki.mozilla.org/CA/Root_Store_Policy_Archive)

1. Introduction

When distributing binary and source code versions of Firefox, Thunderbird, and other Mozilla-related software products, Mozilla includes with such software a set of X.509v3 root certificates from various Certification Authority (CA) operators. The included certificates have their "trust bits" set for various purposes, so that the software in question can use the CA certificates to anchor a chain of trust for certificates used by TLS servers and S/MIME email users without having to ask users for further permission or information.

This policy covers how the default set of certificates and associated trust bits is maintained for software products distributed by Mozilla. Other entities distributing software based on ours are free to adopt their own policies. In particular, under the terms of the relevant Mozilla license(s), distributors of such software are permitted to add or delete CA certificates and modify the values of the trust bits in the versions that they distribute. However, as with other software modifications, by making such changes a distributor may well affect its ability to use Mozilla trademarks in connection with its versions of the software. See the Mozilla trademark policy (https://www.mozilla.org/foundation/trademarks/policy/) for more information.

1.1 Scope

This policy applies, as appropriate, to certificates matching any of the following (and to the CA operators* that control or issue them):

1. CA certificates included in, or under consideration for inclusion in, the Mozilla root store;
2. intermediate certificates that have at least one valid, unrevoked chain up to such a CA certificate and that are technically capable of issuing working server or email certificates. Intermediate certificates that are not considered to be technically capable will contain either:
    - an Extended Key Usage (EKU) extension that does not contain any of these KeyPurposeIds: anyExtendedKeyUsage, id-kp-serverAuth, id-kp-emailProtection; or
    - name constraints that do not allow Subject Alternative Names (SANs) of any of the following types: dNSName, iPAddress, SRVName, or rfc822Name; *and*
3. end entity certificates that have at least one valid, unrevoked chain up to such a CA certificate through intermediate certificates that are all in scope and
    - an EKU extension that contains the anyExtendedKeyUsage KeyPurposeId, or no EKU extension;
    - an EKU extension that contains the id-kp-serverAuth KeyPurposeId; or

- an EKU extension that contains the id-kp-emailProtection KeyPurposeId and an rfc822Name or an otherName of type id-on-SmtpUTF8Mailbox in the subjectAltName.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

* A CA operator is an organization or legal entity that is in possession or control of a CA certificate and associated keys, which are capable of being used to issue new certificates.

1.2 Policy Ownership

Mozilla has appointed a CA Certificate module owner (https://wiki.mozilla.org/Modules/Activities#CA_Certificates) and peers to evaluate new CA requests on our behalf and to make decisions regarding all matters relating to CA certificates included in our root store.

Further, Mozilla has appointed a Mozilla CA Certificate Policy module owner (https://wiki.mozilla.org/Modules/Activities#Mozilla_CA_Certificate_Policy) and peers to maintain this policy. The policy will only be changed after public consultation with the Mozilla community, in order to ensure that all views are taken into account. This policy MAY be updated periodically in accordance with the Process for Updating the Root Store Policy (https://wiki.mozilla.org/CA/Updating_Root_Store_Policy). CA operators MUST adhere to the current version of this policy. You can contact the Mozilla CA Certificate Policy module team at certificates@mozilla.org (mailto:certificates@mozilla.org) if you have questions about this policy.

CA operators or others objecting to a particular decision by either team MAY appeal to the Firefox Technical Leadership Module Committee (https://wiki.mozilla.org/Modules/Firefox_Technical_Leadership) who will make a final decision.

2. Certificate Authorities

2.1 CA Operations

CA operators whose certificates are included in Mozilla's root store MUST:

1. provide some service relevant to users of our software products;
2. follow industry best practice for securing their networks, for example by conforming to the CA/Browser Forum's Network and Certificate System Security Requirements (https://cabforum.org/network-security-requirements/) or a successor document;
3. enforce multi-factor authentication for all accounts capable of causing certificate issuance or performing Registration Authority or Delegated Third Party functions, or implement technical controls operated by the CA to restrict certificate issuance through the account to a limited set of pre-approved domains or email addresses;
4. prior to issuing certificates, verify certificate requests in a manner that we deem acceptable for the stated purpose(s) of the certificates;
5. verify each dNSName or IPAddress in a SAN or commonName in server certificates in accordance with sections 3.2.2.4 and 3.2.2.5 of the CA/Browser Forum's Baseline Requirements at intervals of 398 days or less, and verify that all other information that is included in server certificates remains current and correct at intervals of 825 days or less;

6. otherwise operate in accordance with published criteria that we deem acceptable; *and*
7. ensure that all certificates within the scope of this policy, as described in Section 1.1, adhere to this policy.

CA operators MUST follow and be aware of discussions in both the Mozilla dev-security-policy (https://groups.google.com/a/mozilla.org/g/dev-security-policy) forum and the CCADB Public List (https://groups.google.com/a/ccadb.org/g/public), where root store policies and program updates are announced and public discussions of root inclusion requests occur. They are encouraged, but not required, to contribute to those discussions.

2.2 Validation Practices

We consider verification of certificate signing requests to be acceptable if it meets or exceeds the following requirements:

1. all information that is supplied by the certificate subscriber MUST be verified by using an independent source of information or an alternative communication channel before it is included in the certificate;
2. for a certificate capable of being used for digitally signing or encrypting email messages, the CA operator MUST take reasonable measures to verify that the entity submitting the request controls the email account associated with the email address referenced in the certificate *or* has been authorized by the email account holder to act on the account holder's behalf. This MUST be done using one or more of the methods documented in section 3.2.2 of the CA/Browser Forum's S/MIME Baseline Requirements (https://cabforum.org/smime-br/). The CA operator's CPS (or, if applicable, the CP or CP/CPS) MUST clearly specify the procedure(s) that the CA employs to perform this verification;
3. for a certificate capable of being used for TLS-enabled servers, the CA MUST ensure that the applicant has registered all domain(s) referenced in the certificate or has been authorized by the domain registrant to act on their behalf. This MUST be done using one or more of the methods documented in section 3.2.2.4 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/). The CA operator's CPS (or, if applicable, the CP or CP/CPS) MUST clearly specify the procedure(s) that the CA employs, and each documented procedure MUST state which subsection of 3.2.2.4 it is complying with;
4. for a certificate capable of being used for TLS-enabled servers, the CA MUST ensure that the applicant has control over all IP Address(es) referenced in the certificate. This MUST be done using one or more of the methods documented in section 3.2.2.5 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/). The CA operator's CPS (or, if applicable, the CP or CP/CPS) MUST clearly specify the procedure(s) that the CA employs, and each documented procedure SHOULD state which subsection of 3.2.2.5 it is complying with; *and*
5. for certificates marked as Extended Validation, CA operators MUST comply with the latest version of the Guidelines for the Issuance and Management of Extended Validation Certificates (https://cabforum.org/extended-validation/).

Validation methods are occasionally found to contain security flaws. When this happens, Mozilla expects CA operators to evaluate their practices and respond appropriately to mitigate the risk. Mozilla MAY require CAs to make disclosures or modifications, up to and including immediately discontinuing use of a method.

2.3 Baseline Requirements Conformance

CA operations relating to issuance of certificates capable of being used for TLS-enabled servers MUST conform to the latest version of the CA/Browser Forum Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates (https://cabforum.org/baseline-requirements-documents/) ("TLS Baseline Requirements"). Certificates issued on or after September 1, 2023, that are capable of being used to digitally sign or encrypt email messages, and CA operations relating to the issuance of such certificates, MUST conform to the latest version of the CA/Browser Forum Baseline Requirements for the Issuance and Management of Publicly-Trusted S/MIME Certificates (https://cabforum.org/smime-br/) ("S/MIME Baseline Requirements"). In the event of inconsistency between this policy's requirements and either the S/MIME or TLS Baseline Requirements, this policy's requirements take precedence. The following is a list of known places where this policy takes precedence over the S/MIME and TLS Baseline Requirements. If you find an inconsistency that is not listed here, notify Mozilla so the item can be considered for addition or clarification.

- Insofar as the S/MIME or TLS Baseline Requirements attempt to define their own scope, the scope of this policy (section 1.1) overrides that. CA operations relating to issuance of **all** S/MIME or TLS server certificates in the scope of this policy SHALL conform to the S/MIME or TLS Baseline Requirements, as applicable.

- Mozilla MAY accept audits by auditors who do not meet the qualifications given in section 8.2 of the S/MIME or TLS Baseline Requirements, or refuse audits from auditors who do.

- Mozilla MAY restrict permitted algorithms to a subset of those allowed by the S/MIME or TLS Baseline Requirements.

2.4 Incidents

When a CA operator fails to comply with any requirement of this policy - whether it be a misissuance, a procedural or operational issue, or any other variety of non-compliance - the event is classified as an incident and MUST be reported to Mozilla as soon as the CA operator is made aware. At a minimum, CA operators MUST promptly report all incidents (https://wiki.mozilla.org/CA/Responding_To_An_Incident) to Mozilla in the form of an Incident Report that follows guidance provided on the CCADB website (https://www.ccadb.org/cas/incident-report).

Any matter documented in an audit as a qualification, a modified opinion, or a major non-conformity is also considered an incident and MUST have a corresponding Audit Incident Report (https://www.ccadb.org/cas/incident-report#audit-incident-reports). CA operators MUST regularly update the Incident Report until the corresponding bug is marked as resolved in Bugzilla (https://bugzilla.mozilla.org) by a root store representative. CA operators SHOULD cease issuance until the problem has been prevented from reoccurring.

Mozilla expects the timely remediation of the problems that caused or gave rise to the incident. In response to incidents, Mozilla MAY require the CA operator to submit a plan of action with milestones or to submit one or more additional audits to provide sufficient assurance that the incident has been remediated. Such audits MAY be expected sooner than the CA operator’s next scheduled audit, and thus MAY be expected to be for a period less than a year.

2.4.1 Vulnerability and Security Incident Reporting

Additionally, and not in lieu of the requirement to publicly report incidents as outlined above, a CA Operator MUST disclose a serious vulnerability or security incident in Bugzilla (https://bugzilla.mozilla.org) as a secure bug (https://bugzilla.mozilla.org/enter_bug.cgi?bug_type=task&component=CA%20Security%20Vulnerability&groups=ca-

program-security&product=CA%20Program) in accordance with guidance found on the Vulnerability Disclosure wiki page (https://wiki.mozilla.org/CA/Vulnerability_Disclosure).

3. Documentation

3.1 Audits

Before being included and at least annually thereafter, CA operators MUST obtain certain audits for their root certificates and all intermediate certificates that are technically capable of issuing working server or email certificates. This section describes the requirements for those audits.

3.1.1 Audit Criteria

We consider the criteria for CA operations published in the following documents to be acceptable:

WebTrust Program for Certification Authorities (WebTrust (https://www.cpacanada.ca/en/business-and-accounting-resources/audit-and-assurance/overview-of-webtrust-services/principles-and-criteria))

- WebTrust "Principles and Criteria for Certification Authorities - Version 2.2.2 (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216webtrustca-222final-(15).pdf)", or later;
- WebTrust "Principles and Criteria for Certification Authorities – SSL Baseline with Network Security - Version 2.6 (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216wtbr-26-rev-august-2022final.pdf)", or later;
- WebTrust "Principles and Criteria for Certification Authorities - Extended Validation SSL 1.7.8 (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216wtev-178final.pdf)", or later;
- WebTrust "Principles and Criteria for Certification Authorities - S/MIME Certificates (https://www.cpacanada.ca/-/media/site/operational/ms-member-services/docs/webtrust/01618_ms_smime-certificates_final_aoda-compliant.pdf);

European Telecommunications Standards Institute (ETSI)

- "Trust Service Providers practice" in ETSI EN 319 411-1 v1.3.1 or later version Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements (https://www.etsi.org/deliver/etsi_en/319400_319499/31941101/01.03.01_60/en_31941101v010301p.pdf), specifying a policy or policies appropriate to the trust bit(s) being applied for;
- "Trust Service Providers practice" in ETSI EN 319 411-2 v2.4.1 or later version Policy and security requirements for Trust Service Providers issuing certificates; Part 2: Requirements for trust service providers issuing EU qualified certificates (https://www.etsi.org/deliver/etsi_en/319400_319499/31941102/02.04.01_60/en_31941102v020401p.pdf), specifying a policy or policies appropriate to the trust bit(s) being applied for; *and*
- ETSI "Requirements for Trust Service Providers issuing publicly trusted S/MIME certificates (https://www.etsi.org/deliver/etsi_ts/119400_119499/11941106/01.01.01_60/ts_11941106v010101p.pdf)", ETSI TS 119 411-6 v1.1.1 or later version.

3.1.2 Required Audits

3.1.2.1 WebTrust

If being audited to the WebTrust criteria, the following audit requirements apply (see section 3.1.1 for specific version numbers):

- For the websites trust bit, a CA and all intermediate CAs technically capable of issuing server certificates MUST have all of the following audits:
    - WebTrust for CAs (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216webtrustca-222final-(15).pdf);
    - WebTrust for CAs - SSL Baseline with Network Security (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216wtbr-26-rev-august-2022final.pdf); *and*
    - WebTrust for CAs - EV SSL (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216wtev-178final.pdf) if capable of issuing EV certificates (https://wiki.mozilla.org/CA/EV_Processing_for_CAs#EV_TLS_Capable).
- For the email trust bit, a CA and all intermediate CAs technically capable of issuing email certificates MUST have all of the following audits:
    - WebTrust for CAs (https://www.cpacanada.ca/-/media/site/operational/ep-education-pld/docs/mds21216webtrustca-222final-(15).pdf); and,
    - for audit periods ending after October 30, 2023, a period-of-time audit performed in accordance with WebTrust for CAs - S/MIME (https://www.cpacanada.ca/-/media/site/operational/ms-member-services/docs/webtrust/01618_ms_smime-certificates_final_aoda-compliant.pdf).

3.1.2.2 ETSI

If being audited to the ETSI criteria, the following audit requirements apply (see section 3.1.1 for version numbers):

- For the websites trust bit, a CA and all intermediate CAs technically capable of issuing server certificates MUST have one of the following audits, with at least one of the noted policies or sets of policies:
    - ETSI EN 319 411-1 (https://www.etsi.org/deliver/etsi_en/319400_319499/31941101/01.03.01_60/en_31941101v010301p.pdf) (LCP and (DVCP or OVCP)) and/or (NCP and EVCP); *or*
    - ETSI EN 319 411-2 (https://www.etsi.org/deliver/etsi_en/319400_319499/31941102/02.04.01_60/en_31941102v020401p.pdf) (QCP-w).

  An audit showing conformance with the EVCP policy is REQUIRED if a CA is capable of issuing EV certificates (https://wiki.mozilla.org/CA/EV_Processing_for_CAs#EV_TLS_Capable).

- For the email trust bit, a CA and all intermediate CAs technically capable of issuing email certificates MUST have the following audits, with at least one of the noted policies:
    - ETSI EN 319 411-1 (https://www.etsi.org/deliver/etsi_en/319400_319499/31941101/01.03.01_60/en_31941101v010301p.pdf) (LCP, NCP, or NCP+); *or*
    - ETSI EN 319 411-2 (https://www.etsi.org/deliver/etsi_en/319400_319499/31941102/02.04.01_60/en_31941102v020401p.pdf) (QCP-l, QCP-l-qscd, QCP-n, or QCP-n-qscd); and,
    - for audit periods ending after October 30, 2023, a period-of-time audit performed in accordance with ETSI TS 119 411-6 (https://www.etsi.org/deliver/etsi_ts/119400_119499/11941106/01.01.01_60/ts_11941106v010101p.pdf).

3.1.3 Audit Parameters

Full-surveillance period-of-time audits MUST be conducted and updated audit information provided no less frequently than **annually** from the time of CA key pair generation until the CA public key is no longer trusted by Mozilla's root store. This cradle-to-grave audit requirement applies equally to intermediate CAs as it does to root CAs. Successive period-of-time audits MUST be contiguous (no gaps).

Point-in-time audit statements MAY be used to confirm that all of the problems that an auditor previously identified in a qualified audit statement have been corrected. However, a point-in-time audit does not replace the period-of-time audit.

Audit reports that are being supplied to maintain a certificate within the Mozilla root store MUST be provided to Mozilla via the CCADB within three months of the point-in-time date or the end date of the period.

3.1.4 Public Audit Information

The publicly-available documentation relating to each audit MUST contain the information required by section 5.1 of the CCADB Policy (https://www.ccadb.org/policy) (v.1.2.3) and the CA locations that were or were not audited (https://wiki.mozilla.org/CA/Audit_Statements#Audited_Locations). Audit reports MUST also contain or be accompanied by the name of the lead auditor and qualifications of the team (https://wiki.mozilla.org/CA/Audit_Statements#Auditor_Qualifications) performing the audit, as required by section 3.2.

If Mozilla determines that an audit provided does not meet the requirements of this policy, then Mozilla MAY require that the CA operator obtain a new audit, at the CA operator's expense, for the period of time in question. Additionally, depending on the nature of concerns with the audit, Mozilla MAY require that the CA operator obtain such an audit from a new auditor.

3.2 Auditors

In normal circumstances, Mozilla requires that audits MUST be performed by a Qualified Auditor, as defined in section 8.2 of the S/MIME or TLS Baseline Requirements.

A Qualified Auditor MUST have relevant IT Security experience, or have audited a number of CAs, and be independent. ETSI Audit Attestation Letters MUST follow the Audit Attestation Letter template on the ACAB'c website (https://www.acab-c.com/downloads), and ETSI auditors MUST be members of the Accredited Conformity Assessment

Bodies' Council (https://www.acab-c.com/members/) and follow the ACAB'c Charter and Code of Conduct. WebTrust audit statements MUST follow the practitioner guidance, principles, and illustrative assurance reports on the CPA Canada website (https://www.cpacanada.ca/en/business-and-accounting-resources/audit-and-assurance/overview-of-webtrust-services/principles-and-criteria), and WebTrust auditors MUST be listed as enrolled WebTrust practitioners (https://www.cpacanada.ca/en/business-and-accounting-resources/audit-and-assurance/overview-of-webtrust-services/licensed-webtrust-practitioners-international) on the CPA Canada website. Mozilla MAY, at its sole discretion, decide to temporarily waive membership or enrollment requirements.

Each Audit Report MUST be accompanied by documentation provided to Mozilla of the audit team qualifications (https://wiki.mozilla.org/CA/Audit_Statements#Auditor_Qualifications) sufficient for Mozilla to determine the competence, experience, and independence of the auditor.

If a CA operator wishes to use auditors who do not fit the definition of Qualified Auditor, then it MUST receive written permission from Mozilla to do so in advance of the start of the audit engagement. Mozilla will make its own determination as to the suitability of the suggested party or parties, at its sole discretion.

3.3 CPs and CPSes

We rely on publicly disclosed documentation (e.g., in a Certificate Policy and Certification Practice Statement) to ascertain that our requirements are met. Therefore:

1. the publicly disclosed documentation MUST provide sufficient information for Mozilla to determine whether and how the CA operator complies with this policy, including a description of the steps taken by the CA to verify certificate requests;

2. the publicly disclosed documentation MUST be available from the CA operator’s official website;

3. the documentation MUST be made available to Mozilla under one of the following Creative Commons licenses (or later versions):

    - Attribution (CC-BY (https://creativecommons.org/licenses/by/4.0/)) 4.0;
    - Attribution-ShareAlike (CC-BY-SA (https://creativecommons.org/licenses/by-sa/4.0/)) 4.0;
    - Attribution-NoDerivs (CC-BY-ND (https://creativecommons.org/licenses/by-nd/4.0/)) 4.0; *or*
    - Public Domain Dedication (CC-0 (https://creativecommons.org/publicdomain/zero/1.0/)) 1.0;

    or a set of equally permissive licensing terms accepted by Mozilla in writing. If no such license is indicated, the fact of application is considered as permission from the CA operator to allow Mozilla and the public to deal with these documents, and any later versions for root certificates that are included in Mozilla's root store, under CC-BY-ND 4.0;

4. all CPs, CPSes, and combined CP/CPSes MUST be reviewed and updated as necessary at least once every 365 days, as required by the S/MIME or TLS Baseline Requirements. CA operators MUST indicate that this has happened by incrementing the version number and adding a dated changelog entry, even if no other changes are made to the document;

5. all CPs, CPSes, and combined CP/CPSes MUST be structured according to RFC 3647 and MUST:

- include at least every section and subsection defined in RFC 3647;
- only use the words "No Stipulation" to mean that the particular document imposes no requirements related to that section; and
- contain no sections that are blank and have no subsections;

6. CA operators MUST provide a way to clearly determine which CP, CPS, or combined CP/CPS applies to each of its root and intermediate certificates; *and*

7. CA operators SHALL maintain links to all historic versions of each CP and CPS (or CP/CPS) from the creation of included CA certificates, regardless of changes in ownership or control of such CA certificates, until the entire CA certificate hierarchies (i.e. end entity certificates, intermediate CA certificates, and cross-certificates) operated in accordance with such documents are no longer trusted by the Mozilla root store. For CA certificates that were included in Mozilla's root store before December 31, 2022, the CA Operator shall maintain links in their online repositories to all reasonably available historic versions of CPs and CPSes (or CP/CPSes) from creation of the included CA certificates.

3.4 Compliance Self-Assessments

CA operators with CA certificates capable of issuing working TLS server certificates MUST perform a Compliance Self-Assessment (https://www.ccadb.org/cas/self-assessment) annually. The annual self-assessment MUST be completed and submitted to the CCADB within 92 calendar days from the CA operator's earliest appearing root record “BR Audit Period End Date” that is after December 31, 2023. CA operators SHOULD submit the self-assessment at the same time as uploading audit reports in a CCADB Case (https://www.ccadb.org/cas/updates). CA operators SHOULD use the latest available version of the Compliance Self-Assessment (https://www.ccadb.org/cas/self-assessment) template, and MUST NOT use a version of the self-assessment template that has been superseded by more than 90 calendar days before submission.

4. Common CA Database

Mozilla manages its root store using the Common CA Database (CCADB). CA operators with certificates in Mozilla’s root store MUST use the CCADB, and are bound by the latest published version of the CCADB Policy (https://www.ccadb.org/policy), which is incorporated here by reference.

Mozilla has requirements for the use of the CCADB above and beyond those in the CCADB Policy, as indicated below in this section 4.

4.1 Additional Requirements

- CA operators with intermediate CA certificates that are capable of issuing TLS certificates chaining up to root certificates in Mozilla's root store SHALL populate the "Pertaining to Certificates Issued by This CA" section of the CCADB records corresponding to those intermediate CA certificates with either the CRL Distribution Point for the "Full CRL Issued By This CA" or a "JSON Array of Partitioned CRLs" within 7 days of such intermediate CA issuing its first certificate;
- Each CRL referenced by the JSON Array of Partitioned CRLs MUST contain a critical Issuing Distribution Point extension as described in section 6.1.2; *and*

- if the revocation of an intermediate certificate chaining up to a root in Mozilla's root store is due to a security concern, as well as performing the actions defined in the CCADB Policy, a Vulnerability Disclosure (https://wiki.mozilla.org/CA/Vulnerability_Disclosure) MUST be filed as a secure bug in Bugzilla (https://bugzilla.mozilla.org/enter_bug.cgi?bug_type=task&component=CA%20Security%20Vulnerability&groups=ca-program-security&product=CA%20Program).

4.2 Surveys

Mozilla MAY conduct a survey of CA operators from time to time. CA operators are REQUIRED to respond to the surveys with accurate information, within the timescale defined in the survey.

5. Certificates

5.1 Algorithms

Root certificates in our root store, and any certificate that chains up to them, MUST use only algorithms and key sizes from the following set:

- RSA keys whose modulus size in bits is divisible by 8, and is at least 2048 bits; *or*
- ECDSA keys using one of the following curves:
  - P-256; *or*
  - P-384.

The following curves are not prohibited, but are not currently supported: P-521, Curve25519, and Curve448.

EdDSA keys MAY be included in certificates that chain to a root certificate in our root store if the certificate contains 'id-kp-emailProtection` in the EKU extension. Otherwise, EdDSA keys MUST NOT be included.

The following sections detail encoding and signature algorithm requirements for each of these keys. The encoding requirements on signature algorithms apply to any contexts where the algorithm is encoded as an AlgorithmIdentifier, including:

- The signatureAlgorithm field of a Certificate;
- The signature field of a TBSCertificate;
- The signatureAlgorithm field of a CertificateList;
- The signature field of a TBSCertList; *and*
- The signatureAlgorithm field of a BasicOCSPResponse.

5.1.1 RSA

When RSA keys are encoded in a SubjectPublicKeyInfo structure, the algorithm field MUST consist of an rsaEncryption OID (1.2.840.113549.1.1.1) with a NULL parameter, as specified by RFC 8017, Appendix A.1 (https://datatracker.ietf.org/doc/html/rfc8017#appendix-A.1) and RFC 3279, Section 2.3.1

(https://datatracker.ietf.org/doc/html/rfc3279#section-2.3.1). The encoded AlgorithmIdentifier for an RSA key MUST match the following hex-encoded bytes: `300d06092a864886f70d0101010500`.

CAs MUST NOT use the id-RSASSA-PSS OID (1.2.840.113549.1.1.10) within a SubjectPublicKeyInfo to represent an RSA key.

When a root or intermediate certificate's RSA key is used to produce a signature, only the following algorithms MAY be used, and with the following encoding requirements:

- RSASSA-PKCS1-v1_5 with SHA-1.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300d06092a864886f70d0101050500`.

  See section 5.1.3 for further restrictions on the use of SHA-1.

- RSASSA-PKCS1-v1_5 with SHA-256.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300d06092a864886f70d01010b0500`.

- RSASSA-PKCS1-v1_5 with SHA-384.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300d06092a864886f70d01010c0500`.

- RSASSA-PKCS1-v1_5 with SHA-512.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300d06092a864886f70d01010d0500`.

- RSASSA-PSS with SHA-256, MGF-1 with SHA-256, and a salt length of 32 bytes.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes:

  ```
  304106092a864886f70d01010a3034a00f300d060960864801650304020105
  00a11c301a06092a864886f70d010108300d060960864801650304020105
  00a203020120
  ```

- RSASSA-PSS with SHA-384, MGF-1 with SHA-384, and a salt length of 48 bytes.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes:

  ```
  304106092a864886f70d01010a3034a00f300d060960864801650304020205
  00a11c301a06092a864886f70d010108300d060960864801650304020205
  00a203020130
  ```

- RSASSA-PSS with SHA-512, MGF-1 with SHA-512, and a salt length of 64 bytes.

  The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes:

  ```
  304106092a864886f70d01010a3034a00f300d060960864801650304020305
  00a11c301a06092a864886f70d010108300d060960864801650304020305
  00a203020140
  ```

The above RSASSA-PKCS1-v1_5 encodings consist of the corresponding OID, e.g. sha256WithRSAEncryption (1.2.840.113549.1.1.11), with an explicit NULL parameter, as specified in RFC 3279, Section 2.2.1 (https://datatracker.ietf.org/doc/html/rfc3279#section-2.2.1). Certificates MUST NOT omit this NULL parameter. Note this differs from ECDSA, which omits the parameter.

The above RSASSA-PSS encodings consist of the RSASSA-PSS OID (1.2.840.11.3549.1.1.10) with a corresponding RSASSA-PSS-params structure as parameter. The trailerField MUST be omitted, as it is unchanged from the default value. The AlgorithmIdentifier structures describing the hash functions in the hashAlgorithm field and in the maskGenAlgorithm's parameter MUST themselves include an explicit NULL in the parameter field, as specified by RFC 4055, Section 6 (https://datatracker.ietf.org/doc/html/rfc4055#section-6).

Note: as of Firefox version 100, RSASSA-PSS encodings are supported (https://bugzilla.mozilla.org/show_bug.cgi?id=1088140).

5.1.2 ECDSA

When ECDSA keys are encoded in a SubjectPublicKeyInfo structure, the algorithm field MUST be one of the following, as specified by RFC 5480, Section 2.1.1 (https://datatracker.ietf.org/doc/html/rfc5480#section-2.1.1):

- the encoded AlgorithmIdentifier for a P-256 key MUST match the following hex-encoded bytes: `301306072a8648ce3d020106082a8648ce3d030107`; *or*
- the encoded AlgorithmIdentifier for a P-384 key MUST match the following hex-encoded bytes: `301006072a8648ce3d020106052b81040022`.

The above encodings consist of an ecPublicKey OID (1.2.840.10045.2.1) with a named curve parameter of the corresponding curve OID. Certificates MUST NOT use the implicit or specified curve forms.

When a root or intermediate certificate's ECDSA key is used to produce a signature, only the following algorithms MAY be used, and with the following encoding requirements:

- If the signing key is P-256, the signature MUST use ECDSA with SHA-256. The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300a06082a8648ce3d040302`.
- If the signing key is P-384, the signature MUST use ECDSA with SHA-384. The encoded AlgorithmIdentifier MUST match the following hex-encoded bytes: `300a06082a8648ce3d040303`.

The above encodings consist of the corresponding OID with the parameters field omitted, as specified by RFC 5758, Section 3.2 (https://datatracker.ietf.org/doc/html/rfc5758#section-3.2). Certificates MUST NOT include a NULL parameter. Note this differs from RSASSA-PKCS1-v1_5, which includes an explicit NULL.

5.1.3 SHA-1

Effective July 1, 2022, CAs SHALL NOT sign SHA-1 hashes over end entity certificates with an EKU extension containing the id-kp-emailProtection key purpose.

Effective July 1, 2023, CAs SHALL NOT sign SHA-1 hashes over:

- certificates with an EKU extension containing the id-kp-ocspSigning key purpose;
- intermediate certificates that chain up to roots in Mozilla's program;
- OCSP responses; *or*
- CRLs.

CAs MAY sign SHA-1 hashes over end entity certificates that chain up to roots in Mozilla's program only if all the following are true:

1. the end entity certificate:
    - is not within the scope of the S/MIME or TLS Baseline Requirements;
    - contains an EKU extension that does not contain the id-kp-serverAuth, id-kp-emailProtection, or anyExtendedKeyUsage key purposes; *and*
    - has at least 64 bits of entropy from a CSPRNG in the serial number; *and*
2. the issuing certificate:
    - contains an EKU extension that does not contain the id-kp-serverAuth, id-kp-emailProtection, or anyExtendedKeyUsage key purposes; *and*
    - has a pathlen:0 constraint.

CAs MAY sign SHA-1 hashes over intermediate certificates that chain up to roots in Mozilla's root store only if the certificate to be signed is a duplicate of an existing SHA-1 intermediate certificate with the only changes being all of:

- a new key (of the same size);
- a new serial number (of the same length); *and/or*
- the addition of an EKU and/or a pathlen constraint to meet the requirements outlined above.

CAs MUST NOT sign SHA-1 hashes over other data, including CT pre-certificates.

5.2 Forbidden and Required Practices

CA operations MUST at all times be in accordance with the applicable CP and CPS (or combined CP/CPS).

CA operators MUST maintain a certificate hierarchy such that an included root certificate does not directly issue end entity certificates to customers (i.e. a root certificate signs intermediate issuing certificates), as described in section 6.1.7 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/) and the S/MIME Baseline

Requirements (https://cabforum.org/smime-br/).

CA operators MUST maintain current best practices to prevent algorithm attacks against certificates. As such, all new certificates MUST have a serial number greater than zero, containing at least 64 bits of output from a CSPRNG.

CA operators MUST NOT issue certificates, CRLs, or OCSP responses, that have:

- ASN.1 DER encoding errors;
- invalid public keys (e.g., RSA certificates with public exponent equal to 1); *or*
- missing or incorrect extensions (e.g., TLS certificates with no subjectAltName extension, delegated OCSP responders without the id-pkix-ocsp-nocheck extension, partial/scoped CRLs that lack a distributionPoint in a critical issuingDistributionPoint extension).

CA operators MUST NOT issue certificates that have:

- duplicate issuer names and serial numbers (except that a Certificate Transparency pre-certificate is allowed to match the corresponding certificate); *or*
- cRLDistributionPoints or OCSP authorityInfoAccess extensions for which no operational CRL or OCSP service exists.

CA operators MUST NOT generate the key pairs for end entity certificates that have an EKU extension containing the KeyPurposeIds id-kp-serverAuth or anyExtendedKeyUsage, unless the certificate is being issued to the CA itself.

All end entity certificates MUST include an EKU extension containing KeyPurposeId(s) describing the intended usage(s) of the certificate, and the EKU extension MUST NOT contain the KeyPurposeId anyExtendedKeyUsage.

5.3 Intermediate Certificates

All certificates that are capable of being used to issue new certificates and that directly or transitively chain to a CA certificate included in Mozilla's root store MUST be operated in accordance with this policy.

A certificate is deemed capable of being used to issue new certificates if it contains an X.509v3 basicConstraints extension (https://datatracker.ietf.org/doc/html/rfc5280#section-6.1.4) with the cA boolean set to true.

A certificate is deemed to directly or transitively chain to a CA certificate included in Mozilla's root store if: (1) the certificate's Issuer Distinguished Name matches (according to the name-matching algorithm specified in RFC 5280, section 7.1) the Subject Distinguished Name in a CA certificate or intermediate certificate that is in scope according to section 1.1 of this Policy, *and* (2) the certificate is signed with a Private Key whose corresponding Public Key is encoded in the SubjectPublicKeyInfo of that CA certificate or intermediate certificate.

Intermediate certificates created after January 1, 2019, with the exception of cross-certificates that share a private key with a corresponding root certificate:

- MUST contain an EKU extension;
- MUST NOT include the anyExtendedKeyUsage KeyPurposeId; *and*
- MUST NOT include both the id-kp-serverAuth and id-kp-emailProtection KeyPurposeIds in the same certificate.

5.3.1 Technically Constrained

We encourage CA operators to technically constrain all intermediate certificates. For an intermediate certificate to be considered technically constrained, the certificate MUST include an Extended Key Usage (EKU) (https://datatracker.ietf.org/doc/html/rfc5280#section-4.2.1.12) extension specifying the extended key usage(s) allowed for the type of end entity certificates that the intermediate CA is authorized to issue. We also encourage CA operators to include only a single KeyPurposeID in the EKU extension of intermediate certificates. The anyExtendedKeyUsage KeyPurposeId MUST NOT appear within this extension.

The conformance requirements defined in section 2.3 of this policy also apply to technically constrained intermediate certificates.

If the intermediate CA certificate includes the id-kp-serverAuth extended key usage, then to be considered technically constrained, the certificate MUST be name-constrained as described in section 7.1.2.5 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/), each entry in permittedSubtrees having been validated according to section 3.2.2 of the TLS Baseline Requirements. The id-kp-clientAuth EKU MAY also be present.

If the intermediate CA certificate includes the id-kp-emailProtection extended key usage, then to be considered technically constrained, it MUST comply with section 7.1.5 of the S/MIME Baseline Requirements (https://cabforum.org/smime-br/) and include the Name Constraints X.509v3 extension with constraints on rfc822Name, with at least one name in permittedSubtrees, each name having been validated according to section 3.2.2 of the S/MIME Baseline Requirements (https://cabforum.org/smime-br/). The values id-kp-serverAuth and anyExtendedKeyUsage MUST NOT be present. The id-kp-clientAuth EKU MAY be present. Other values that the CA is allowed to use and are documented in the CA's CP, CPS, or combined CP/CPS MAY be present.

5.3.2 Publicly Disclosed and Audited

The operator of a CA certificate included in Mozilla's root store MUST publicly disclose in the CCADB all CA certificates it issues that chain up to that CA certificate trusted in Mozilla's root store that are technically capable of issuing working server or email certificates, including such CA certificates that are revoked but not yet expired and those CA certificates that share the same key pair whether they are self-signed, doppelgänger, reissued, cross-signed, or other roots. The CA operator with a certificate included in Mozilla's root store MUST disclose such CA certificate in the CCADB within one week of certificate creation, and before any such CA is allowed to issue certificates. Name-constrained CA certificates that are technically capable of issuing working server or email certificates that were exempt from disclosure in previous versions of this policy MUST also be disclosed in the CCADB, but the submission of an audit report under section 3.1 of this policy is not required.

All disclosure MUST be made freely available and without additional requirements, including, but not limited to, registration, legal agreements, or restrictions on redistribution of the certificates in whole or in part.

We recognize that technically constraining intermediate certificates as described above may not be practical in some cases. All certificates that are capable of being used to issue new certificates, that are not technically constrained, and that directly or transitively chain to a certificate included in Mozilla's root store MUST be audited in accordance with this policy, and the corresponding audit statements MUST be disclosed in the CCADB according to section 5 of the CCADB Policy (https://www.ccadb.org/policy#5-policies-audits-and-practices). If the CA operator has a currently valid audit report at the time of creation of the intermediate certificate, then the new intermediate certificate MUST appear on the CA operator's next periodic audit reports.

5.4 Precertificates

The logging of a precertificate in a Certificate Transparency log is considered by Mozilla to be a binding intent to issue a final certificate, as described in section 3.1 of RFC 6962 (https://datatracker.ietf.org/doc/html/rfc6962#section-3.1). "Final certificate" means a certificate that is not a precertificate. Precertificates are in-scope for enforcing compliance with these requirements. A final certificate is "based on" a precertificate if they have the same serial and issuer, or they have the same serial and the final certificate's issuer matches the precertificate's issuer's issuer. Thus,

- it is mississuance to issue a final certificate based on a precertificate if they do not exactly match each other according to RFC 6962, section 3.1;
- if a precertificate implies the existence of a final certificate that does not comply with this policy, it is considered misissuance of the final certificate, even if the certificate does not actually exist;
- a CA MUST be able to revoke a certificate presumed to exist, if revocation of the certificate is required under this policy, even if the final certificate does not actually exist; *and*
- a CA MUST provide CRL and OCSP services and responses in accordance with this policy for all certificates presumed to exist based on the presence of a precertificate, even if the certificate does not actually exist.

6. Revocation

CA operators MUST maintain an online 24x7 repository mechanism whereby application software can automatically check online the current status of all unexpired certificates issued by the CA.

For end entity certificates, CRLs MUST be updated and reissued at least every seven days, and the value of the nextUpdate field MUST NOT be more than ten days beyond the value of the thisUpdate field.

For end entity certificates, if the CA provides revocation information via an Online Certificate Status Protocol (OCSP) service:

- it MUST update that service at least every four days;
- responses MUST have a defined value in the nextUpdate field, and it MUST be no more than ten days after the thisUpdate field; *and*
- the value in the nextUpdate field MUST be before or equal to the notAfter date of all certificates included within the BasicOCSPResponse.certs field or, if the certs field is omitted, before or equal to the notAfter date of the CA certificate which issued the certificate that the BasicOCSPResponse is for.

Section 4.9.12 of a CA operator's CPS (or, if applicable, the CP or CP/CPS) MUST clearly specify the methods that parties may use to demonstrate private key compromise.

6.1 TLS

For any certificate in a hierarchy capable of being used for TLS-enabled servers, CAs MUST revoke certificates that they have issued upon the occurrence of any event listed in the appropriate subsection of section 4.9.1 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/), according to the timeline defined therein. CAs MUST also revoke any certificates issued in violation of the then-current version of this policy according to the timeline defined in section 4.9.1 of the TLS Baseline Requirements.

6.1.1 End Entity TLS Certificate CRLRevocation Reasons

When an end entity TLS certificate (i.e. a certificate capable of being used for TLS-enabled servers) is revoked for one of the reasons below, the specified CRLReason MUST be included in the reasonCode extension of the CRL entry corresponding to the end entity TLS certificate, as described in sections 4.9.1 and 7.2.2 of the TLS Baseline Requirements (https://cabforum.org/baseline-requirements-documents/).

- keyCompromise (RFC 5280 CRLReason #1);
- privilegeWithdrawn (RFC 5280 CRLReason #9);
- cessationOfOperation (RFC 5280 CRLReason #5);
- affiliationChanged (RFC 5280 CRLReason #3); *or*
- superseded (RFC 5280 CRLReason #4).

The keyCompromise, superseded, and privilegeWithdrawn CRLReasons MUST only be used for the situations listed in the CA/Browser Forum Baseline Requirements as corresponding to these revocation reasons. Otherwise, the keyCompromise, superseded, and privilegeWithdrawn CRLReasons MUST NOT be used.

Mozilla's wiki page, "Revocation Reasons" (https://wiki.mozilla.org/CA/Revocation_Reasons), provides further details about when the CRLReasons listed above must and must not be used.

6.1.2 TLS Certificate CRL Issuing Distribution Points

A CRL whose scope does not include all unexpired certificates that are issued by the CA SHALL contain a critical Issuing Distribution Point extension (OID 2.5.29.28). The distributionPoint field of the extension SHALL include a UniformResourceIdentifier whose value is derived from one of the two following sources:

1. The UniformResourceIdentifier as encoded in the distributionPoint field of an issued certificate's CRL Distribution Points extension (see RFC 5280 section 5.2.5); or
2. The URL as included in the "JSON Array of Partitioned CRLs" field in the CCADB entry corresponding to the certificate for the issuing CA.

6.2 S/MIME

For any certificate in a hierarchy capable of being used for S/MIME, CAs MUST revoke certificates that they have issued upon the occurrence of any event listed in the appropriate subsection of section 4.9.1 of the S/MIME Baseline Requirements (https://cabforum.org/smime-br/), according to the timeline defined therein. CAs MUST also revoke any certificates issued in violation of the then-current version of this policy according to the timeline defined in section 4.9.1 of the S/MIME Baseline Requirements.

7. Root Store Changes

Changes that are motivated by a security concern, such as a root or intermediate CA compromise, MUST be treated as security-sensitive, and a Vulnerability Disclosure (https://wiki.mozilla.org/CA/Vulnerability_Disclosure) MUST be filed as a secure bug in Bugzilla (https://bugzilla.mozilla.org/enter_bug.cgi?bug_type=task&component=CA%20Security%20Vulnerability&groups=ca-program-security&product=CA%20Program).

7.1 Inclusions

We will determine which CA certificates are included in Mozilla's root store based on the risks of such inclusion to typical users of our products (https://wiki.mozilla.org/CA/Root_Inclusion_Considerations). We will consider adding additional CA certificates to the default certificate set upon request only by an authorized representative of the subject CA. We will make such decisions through a public process (https://wiki.mozilla.org/CA/Application_Process).

We will not charge any fees to have a CA operator's certificate(s) included in Mozilla's root store.

We reserve the right to not include certificates from a particular CA operator in our root store. This includes (but is not limited to) cases where we believe that a CA operator has caused undue risks to users' security, e.g. by knowingly issuing certificates without the knowledge of the entities whose information is referenced in those certificates ('MITM certificates'). Mozilla is under no obligation to explain the reasoning behind any inclusion decision.

Before being included, CA operators MUST provide evidence that their CA certificates fully comply with the current Mozilla Root Store Requirements and the S/MIME or TLS Baseline Requirements, and have continually, from the time of CA private key creation, complied with the then-current Mozilla Root Store Policy and the S/MIME or TLS Baseline Requirements, as applicable.

To request that its certificate(s) be added to Mozilla's root store, a CA operator SHOULD submit a formal request by submitting a bug report (https://bugzilla.mozilla.org/enter_bug.cgi?bug_type=task&product=CA%20Program&component=CA%20Certificate%20Root%20Program) into Bugzilla (https://bugzilla.mozilla.org), filed against the "CA Certificate Root Program" component of the "CA Program" product. Mozilla's wiki page, "Applying for root inclusion in Mozilla products (https://wiki.mozilla.org/CA/Application_Process)", provides further details about how to submit a formal request. The request MUST be made by an authorized representative of the subject CA operator, and MUST include the following:

1. the certificate data (or links to the data) for the CA certificate(s) requested for inclusion;
2. for each CA certificate requested for inclusion, whether or not the CA issues certificates for each of the following purposes within the certificate hierarchy associated with the CA certificate:
    - TLS-enabled servers
    - digitally-signed and/or encrypted email;
3. for each CA certificate requested for inclusion, whether the CA issues Extended Validation certificates within the certificate hierarchy associated with the CA certificate and, if so, the CA/Browser Forum EV policy OID of 2.23.140.1.1 associated with the CA certificate;
4. a Certificate Policy and Certification Practice Statement (or links to a CP and CPS) or equivalent disclosure document(s) for the CA or CAs in question;
5. an auditor-witnessed root key generation ceremony report and contiguous period-of-time audit reports performed thereafter no less frequently than annually; *and*
6. information as to how the CA operator has fulfilled the requirements stated above regarding its verification of certificate signing requests and its conformance to a set of acceptable operational criteria.

We will reject requests where the CA operator does not provide such information within a reasonable period of time after submitting its request.

7.2 Updates

Changes MAY be made to CA certificates that are included in Mozilla's root store as follows:

1. enabling a trust bit in a CA certificate that is currently included, MAY only be done after careful consideration of the CA operator's current policies, practices, and audits, and MAY be requested by a representative of the CA or a representative of Mozilla by submitting a bug report into Bugzilla (https://bugzilla.mozilla.org), as described in Mozilla's wiki page, "Applying for root inclusion in Mozilla products (https://wiki.mozilla.org/CA/Application_Process)";
2. enabling EV in a CA certificate that is currently included, MAY only be done after careful consideration of the CA operator's current policies, practices, and audits, and MAY be requested by a representative of the CA operator or a representative of Mozilla by submitting a bug report into Bugzilla (https://bugzilla.mozilla.org), as described in Mozilla's wiki page, "Applying for root inclusion in Mozilla products (https://wiki.mozilla.org/CA/Application_Process)";
3. disabling a CA certificate is the act of turning off one or more of the trust bits (websites or email), and MAY be requested by a representative of the CA operator or a representative of Mozilla by submitting a bug report into Bugzilla (https://bugzilla.mozilla.org), as described in the Root Change Process (https://wiki.mozilla.org/CA/Certificate_Change_Process); *and*
4. a representative of the CA operator or a representative of Mozilla MAY request that a CA certificate be removed by submitting a bug report into Bugzilla (https://bugzilla.mozilla.org), as described in the Root Change Process (https://wiki.mozilla.org/CA/Certificate_Change_Process).

7.3 Removals

Mozilla MAY, at its sole discretion, decide to disable (partially or fully), or remove a certificate, at any time and for any reason. This MAY happen immediately or on a planned future date. Mozilla will disable or remove a certificate if the CA operator demonstrates ongoing or egregious practices that do not maintain the expected level of service or that do not comply with the requirements of this policy.

Mozilla will take any steps we deem appropriate to protect our users if we learn that a CA operator has knowingly or intentionally mis-issued one or more certificates. This MAY include, but is not limited to, disablement (partially or fully) or removal of all the CA operator's certificates from Mozilla's root store.

The category of mis-issued certificates includes (but is not limited to) those issued to someone who should not have received them, those containing information which was not properly validated, those having incorrect technical constraints, and those using algorithms other than those permitted.

A failure to provide notifications or updates in the CCADB or as otherwise required in a timely manner SHALL also be grounds for disabling a CA operator's root certificates or removing them from Mozilla's root store. For this policy and the CCADB policies, "a timely manner" means within 30 days of when the appropriate data or documentation becomes available to the CA operator, unless a Mozilla policy document specifies a different rule.

If Mozilla disables or removes a CA operator's certificate(s) from Mozilla's root store based on a CA operator's actions (or failure to act) that are contrary to this policy, Mozilla will publicize that fact (for example, on the Mozilla dev-security-policy list (https://groups.google.com/a/mozilla.org/g/dev-security-policy), and on our websites) and MAY also alert relevant

news, government, or industry organizations.

7.4 Root CA Lifecycles

For a root CA certificate trusted for server authentication, Mozilla will remove the websites trust bit when the CA key material is more than 15 years from the CA key material generation date. For a root CA certificate trusted for secure email, Mozilla will set the "Distrust for S/MIME After Date" for the CA certificate to 18 years from the CA key material generation date. The CA key material generation date SHALL be determined by reference to the auditor-witnessed key generation ceremony report. If the CA operator cannot provide the key generation ceremony report for a root CA certificate created before July 1, 2012, then Mozilla will use the “Valid From” date in the root CA certificate to establish the key material generation date. For transition purposes, root CA certificates in the Mozilla root store will be distrusted according to the schedule located at https://wiki.mozilla.org/CA/Root_CA_Lifecycles (https://wiki.mozilla.org/CA/Root_CA_Lifecycles), which is subject to change if underlying algorithms become more susceptible to cryptanalytic attack or if other circumstances arise that make this schedule obsolete.

CA operators are strongly urged to apply to Mozilla for inclusion of their next generation root certificate at least 2 years before the distrust date of the CA certificate they wish to replace.

8. CA Operational Changes

CA operators SHALL NOT assume that trust is transferable. All CA operators whose certificates are included in Mozilla's root store MUST notify Mozilla (mailto:certificates@mozilla.org) before:

- ownership or control of the CA’s certificate(s) changes;
- an organization other than the CA operator obtains control of an unconstrained intermediate certificate (as defined in section 5.3 of this policy) that directly or transitively chains to a certificate included in Mozilla's root store - see Process for non-Technically-Constrained Subordinate CAs (https://wiki.mozilla.org/CA/External_Sub_CAs);
- ownership or control of the CA’s operations changes; *or*
- there is a change in the CA's operations that could affect the CA's ability to comply with the requirements of this Policy.

CA operators SHOULD err on the side of notification if there is any doubt. Mozilla will normally keep commercially sensitive information confidential. Throughout any change, CA operations MUST continue to meet the requirements of this policy. If one of the above events occurs, Mozilla MAY require additional audit(s) as a condition of remaining in the root store. CA operators are encouraged to notify Mozilla in advance in order to avoid unfortunate surprises.

In addition, one or more of the following sections MAY apply.

8.1 Change in Legal Ownership

This section applies when one company buys or takes a controlling stake in a CA or CA operator, or when an organization obtains control of a CA key pair that is within the scope of Mozilla's root store, unless it is constrained in compliance with section 5.3.1 of this policy.

Mozilla MUST be notified of any resulting changes in the CA operator's CP, CPS, or combined CP/CPS.

If the receiving or acquiring company is new to the Mozilla root store, it MUST demonstrate compliance with the entirety of this policy. There MUST be a public discussion regarding its admittance to the root store. If Mozilla reaches a positive conclusion after public discussion, then the affected certificate(s) MAY remain in the root store. If the entire CA operation is not included in the scope of the transaction, issuance is not permitted until the discussion has been resolved with a positive conclusion.

8.2 Change in Operational Personnel

This section applies when operation of a CA certificate that is within the scope of Mozilla's root store and not constrained in compliance with section 5.3.1 of this policy is transferred to a different organization, whether by acquisition or contract.

The transferor MUST ensure that the transferee is able to fully comply with this policy. The transferor will continue to be responsible for the root certificate's private key until Mozilla has been provided with an audit statement (or opinion letter) confirming successful transfer of the root certificate and key. Issuance MUST NOT occur until the transferee has provided all the information required by the CCADB, and demonstrated to Mozilla that they have all the appropriate audits, CP/CPS documents, and other systems in place.

The transferor MUST notify Mozilla about any necessary changes to EV status or trust bits in Mozilla's root store. If the transferee will be technically capable of issuing EV certificates, the transferor MUST confirm that the transferee has or will get the relevant audits before issuing EV certificates.

8.3 Change in Secure Location

This section only applies when section 8.1 and/or section 8.2 applies, and when the cryptographic hardware related to a CA certificate that is within the scope of Mozilla's root store and not constrained in compliance with section 5.3.1 of this policy is consequently moved from one secure location to another.

This policy and the relevant WebTrust or ETSI requirements apply at all times, even during the physical relocation of a CA's online operations to a new data center and moving parts of an offline root certificate from one location to another. As such, a CA operator MUST always ensure that physical access to CA equipment is limited to authorized individuals, the equipment is operated under multi-person control, and unauthorized CA system usage is able to be detected at all times. The auditor MUST confirm that there are appropriate procedures in place to ensure that the requirements are met and that those procedures are followed.

The following steps MUST be taken by the organization(s) concerned:

- ensure that annual audit statements are current;
- notify Mozilla of the pending change;
- create a transfer plan (and legal agreement if more than one organization is involved) and have it reviewed by the auditors;
- stop new certificate issuance at the current site before the transfer begins;
- have an audit performed at the current site to confirm when the root certificate is ready for transfer, and ensure that key material is properly secured;

- have the transfer ceremony witnessed by auditors and video recorded, with a physical exchange of the HSM or ciphertext containing the associated key material and certificates, and the multi-party authorization keys;
- perform an audit at the new site to confirm that the transfer was successful, that the private key remained secure throughout the transfer, and that the root certificate is ready to resume issuance. This requirement MAY be met by including the transferred root certificate and key in the new owner's regular audits or by getting a point-in-time audit; *and*
- send links to the updated CP, CPS, and the updated audit statements, opinion letter, or point-in-time audit statement to Mozilla.

The regular annual audit statements MUST still happen in a timely manner.

If a security issue arises during key transfer, then the organization(s) concerned MUST immediately file a Vulnerability Disclosure (https://wiki.mozilla.org/CA/Vulnerability_Disclosure) in Bugzilla using a secure bug (https://bugzilla.mozilla.org/enter_bug.cgi?bug_type=task&component=CA%20Security%20Vulnerability&groups=ca-program-security&product=CA%20Program).

8.4 Externally-Operated Subordinate CAs

The operator of a root CA certificate that is included in Mozilla's root store is at all times completely and ultimately accountable for every certificate signed under that root CA certificate, whether directly or through subordinate CAs or cross-certified CAs. The operator of the root CA certificate SHALL ensure that the operator of each subordinate or cross-certified CA fully and continually adheres to this policy.

The root CA operator MUST complete Mozilla's Process for non-Technically-Constrained Subordinate CAs (https://wiki.mozilla.org/CA/External_Sub_CAs) (including successful review and approval by Mozilla) before a new externally-operated subordinate CA begins issuing certificates under any of the following conditions:

- the subordinate CA operator will obtain a unconstrained (per section 5.3.1 of this policy) CA certificate, and the subordinate CA operator is not approved by Mozilla to issue the type of certificates (email, TLS, or EV TLS), which they will be able to issue under the new CA certificate;
- the root CA operator is cross-signing a CA certificate of a CA operator who is not currently in Mozilla's root store; *or*
- the root CA operator is cross-signing a CA certificate of another CA operator who is currently in Mozilla's root store, but the other CA operator has not been approved for the same trust bits (email or websites) or EV, and those trust bits or EV will be recognized under the cross-signed certificate that it will be receiving.

We reserve the right to not approve subordinate CA certificates. This includes (but is not limited to) cases where we believe that approval of a subordinate CA operator would cause undue risks to users' security. Mozilla is under no obligation to explain the reasoning behind such decisions.

When any of the following conditions apply, the root CA operator is not required to perform Mozilla's Process for non-Technically-Constrained Subordinate CAs (https://wiki.mozilla.org/CA/External_Sub_CAs) before the subordinate CA certificate begins issuing certificates:

- the subordinate CA will be operated directly by the root CA operator under the exact same policies and practices of the root CA operator and within the same scope of audit reporting, and no new organizations will be involved in

the management or operation of the CA;

- the CA certificate is technically constrained as described in section 5.3.1 of this policy;
- the subordinate CA operator:
    - has previously undergone the Process for non-Technically-Constrained Subordinate CAs (https://wiki.mozilla.org/CA/External_Sub_CAs);
    - has been approved for the type of certificates to be issued (email, TLS, or EV TLS); *and*
    - will operate under the same policies and practices as the previous review, and under the same scope of audit reporting as the prior subordinate CA certificate. (Newer versions of policies and practices MAY be used, provided that the subordinate CA operator follows the same versions of the policies for both the existing and new CA certificates.)
- as of June 1, 2022, the subordinate CA operator was already trusted for issuing the same type of certificates under an existing subordinate CA certificate that directly or transitively chains to a certificate included in Mozilla's root store; *or*
- the root CA operator is cross-signing a CA certificate of another CA operator that is currently in Mozilla's root store, and that other CA operator:
    - will only be able to issue the same type of certificate (email, TLS, or EV TLS) that they are already approved for in Mozilla's root store; *and*
    - will operate both the cross-signed certificate and their CA certificate(s) under the same policies, practices, and scope of audit that their CA certificate was approved for. Newer versions of policies and practices MAY be used, provided that the cross-signed CA operator follows the same versions of the policies for both the cross-signed certificate and their CA certificate(s).
