ETSI TR 119 411-9 V1.1.1 (2026-03)

TECHNICAL REPORT

Electronic Signatures and Trust Infrastructures (ESI);
Policy and security requirements for
Trust Service Providers issuing certificates;
Part 9: Requirements on a Certificate Transparency (CT)
Ecosystem to make the issuing of
certificates transparent and verifiable

Reference
DTR/ESI-0019411-9

Keywords
authentication, EU qualified, SSL/TLS certificates, verifiable registry

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the ETSI Search & Browse Standards application.

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on ETSI deliver repository.

Users should be aware that the present document may be revised or have its status changed, this information is available in the Milestones listing.

If you find errors in the present document, please send your comments to the relevant service listed under Committee Support Staff.

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure (CVD) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2026.
All rights reserved.

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Electronic Signatures and Trust Infrastructures (ESI).

The present document is part 9 of a multi-part deliverable covering policy requirements for Trust Service Providers issuing certificates. Full details of the entire series can be found in part 1 [i.2].

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

The present document provides guidance for establishing a Certificate Transparency (CT) ecosystem to make the issuance of selected certificate types issued under Regulation (EU) No 910/2014 (eIDAS) [i.4] amended by Regulation (EU) No 2024/1183 [i.5] transparent and verifiable. IETF RFC 6962 [i.7] and IETF RFC 9162 [i.8] constitute a mandatory requirement for the issuance of wallet-relying party access certificates, as established by Commission Implementing Regulation (EU) 2025/848 [i.9].

Building on the demonstrated effectiveness of IETF RFC 6962 [i.7] and IETF RFC 9162 [i.8] in the WebPKI - improving accountability, enabling early misissuance detection, and supporting independent monitoring and auditing - the report maps core ecosystem roles (TSP-CA, CT log operator, monitor, auditor, relying party), reviews existing specifications (primarily IETF RFC 6962 [i.7], with forward compatibility to IETF RFC 9162 [i.8]), and describes operational artefacts (SCTs, inclusion and consistency proofs) and controls. It is proposed to adopt the fundamental CT structure while developing a European CT infrastructure with appropriate governance, qualification criteria and audit frameworks to address eIDAS-specific requirements, ensure operational resilience, and avoid undue external dependencies.

The present document further proposes building and operating the ecosystem in conformity with IETF RFC 6962 [i.7] as the interoperability baseline and considering the Chromium Static-CT-API (tiled logs) as an input to future standardization due to its promising operational properties (e.g. inclusion-before-SCT issuance and a static, cacheable read path). The resulting recommendations aim to preserve interoperability with widely deployed CT mechanisms while enabling effective regulatory oversight, lifecycle management, and long-term verifiability in the European context.

Introduction

Certificate Transparency (CT) [i.7] and [i.8] was originally developed to make the issuing of TLS certificates [i.6] for the Web Public Key Infrastructure (Web PKI) transparent and traceable. Even today, CT [i.7] and [i.8] is only used in this area and is supported by both browser vendor-accepted certificate issuers and many web browsers.

A CT ecosystem is necessary for the successful implementation of CT [i.7] and [i.8]. This CT ecosystem consists of various parties with different roles. These parties and their roles are:

- the certificate issuer accepted by the browser vendor, who issues the pre-certificate or TLS certificate (certificates) to be logged;
- the CT log, which stores the certificates sorted by order of receipt, cryptographically assured so that they cannot be deleted, changed or retroactively inserted without being noticed;
- the certificate consumer, e.g. a web browser that uses the certificate and retrieves information from the CT log;
- the CT monitor, which retrieves certificates from the CT log to help identifying misused or unauthorized certificates; and
- the CT auditor, which checks the integrity of the CT log.

Browser vendors define their own policies towards the CT logs and their operators as well as the accepted certificate issuers. Compliance with these policies helps them to trust the CT ecosystem and thus the Web PKI.

Regulation (EU) No 910/2014 [i.4] amended by Regulation (EU) No 2024/1183 [i.5] provides for the use of CT [i.7] and [i.8] for wallet-relying party access certificates in Commission Implementing Regulation No 2025/848 [i.9]. The requirement to use CT [i.7] and [i.8] for other certificates can be expanded.

The present document is intended to provide recommendations for the standardization of a CT ecosystem with regard to formats, protocols, policies and security requirements for use by certificates issued in accordance with Regulation (EU) No 910/2014 [i.4] amended by Regulation (EU) No 2024/1183 [i.5].

1 Scope

The present document provides an overview of the technology known as 'Certificate Transparency (CT)' and explains its application to ensure transparency and traceability in the issuance of certificates. It enumerates the objectives associated with CT, provides a mapping of the underlying ecosystem and identifies the underlying technical standards and norms or initiatives of CT.

In addition, the present document uses the specific case of TLS certificates issued by certificate issuers accepted by browser vendors to highlight the functionality of CT and demonstrate the interaction between the different parties involved.

Finally, the present document provides guidance on how and which areas and processes should be considered in the context of European standardization in order to define a CT infrastructure that can be used to make certificates issued in accordance with Regulation (EU) No 910/2014 [i.4] amended by Regulation (EU) No 2024/1183 [i.5] transparent and verifiable.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] ETSI EN 319 401: "Electronic Signatures and Trust Infrastructures (ESI); General Policy Requirements for Trust Service Providers".

[i.2] ETSI EN 319 411-1: "Electronic Signatures and Trust Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements".

[i.3] ETSI EN 319 411-2: "Electronic Signatures and Trust Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 2: Requirements for trust service providers issuing EU qualified certificates".

[i.4] Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC.

[i.5] Regulation (EU) 2024/1183 of the European Parliament and of the Council of 11 April 2024 amending Regulation (EU) No 910/2014 as regards establishing the European Digital Identity Framework.

[i.6] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[i.7] IETF RFC 6962: "Certificate Transparency".

[i.8] IETF RFC 9162: "Certificate Transparency Version 2.0".

[i.9] Commission Implementing Regulation (EU) 2025/848 of 6 May 2025 laying down rules for the application of Regulation (EU) No 910/2014 of the European Parliament and of the Council as regards the registration of wallet-relying parties.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI EN 319 401 [i.1], ETSI EN 319 411-1 [i.2] and ETSI EN 319 411-2 [i.3] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI EN 319 401 [i.1], ETSI EN 319 411-1 [i.2], ETSI EN 319 411-2 [i.3] and the following apply:

CT	Certificate Transparency
SCT	Signed Certificate Timestamp

4 Certificate Transparency

4.1 Introduction

Certificate Transparency (CT) is an Internet security framework that enhances the integrity, accountability, and transparency of the Public Key Infrastructure (PKI). It does so by requiring that all issued digital certificates be publicly recorded in cryptographically verifiable, append-only logs. These publicly auditable logs enable the detection of misissued or unauthorized certificates, helping to address structural weaknesses in the PKI. CT is currently deployed exclusively within the WebPKI.

4.2 Objectives of Certificate Transparency

The following objectives are identified as central to the CT framework:

a) Detection of certificate misissuance

CT enables the timely identification of certificates that have been issued incorrectly or fraudulently by TSPs acting as CAs. Such misissuance may result from technical misconfigurations, non-compliance with certification policies, or compromise of TSP systems. Visibility into certificate issuance is a key mechanism for identifying and addressing such incidents.

b) Enhancement of transparency within the PKI

By requiring the publication of certificate issuance events in cryptographically verifiable logs, CT increases transparency within the PKI ecosystem. This transparency facilitates oversight and trust in the operational practices of TSPs.

c) Accountability of Trust Service Providers

CT supports increased accountability by requiring TSPs acting as CAs to submit all issued certificates to public logs. These logs enable independent review and long-term auditability of TSP behaviour by third parties.

d) Support for third-party certificate monitoring

CT allows credential owners, relying parties, and other interested stakeholders to monitor certificate issuance associated with their credentials. This capability supports early detection of unauthorized or unexpected certificates.

e) Facilitation of revocation and incident response

The real-time or near real-time availability of CT log data enables more efficient incident detection and response. This includes the timely revocation of misissued certificates and the implementation of corrective security measures.

f) Mitigation of undetected use of fraudulent certificates

CT reduces the risk of undetected certificate misuse by requiring that certificates be publicly logged before they are accepted by relying parties. This requirement limits the feasibility of Man-In-The-Middle (MITM) attacks and impersonation using undetected fraudulent certificates.

4.3 Functionality and participants in the CT ecosystem

4.3.1 Functional Overview

The core functions of CT are described below:

a) Certificate Submission

Trust Service Providers (TSPs) acting as Certification Authorities (CAs) are responsible for submitting either the final certificate or a corresponding precertificate to one or more publicly accessible Certificate Transparency logs. This submission forms the basis for transparent disclosure of certificate issuance events.

b) Signed Certificate Timestamp (SCT) Issuance

Upon successful submission of the certificate or precertificate, the CT log issues a Signed Certificate Timestamp (SCT). The SCT is a cryptographic assertion that the submitted certificate will be included in the log within a predefined Maximum Merge Delay (MMD). The SCT serves as evidence of the log's commitment to incorporate the certificate in a timely and verifiable manner.

c) SCT Delivery Mechanisms

SCTs may be conveyed to relying parties through one or more of the following mechanisms:

- embedded in the issued X.509 certificate as an extension (static SCT);
- delivered during the TLS handshake using the TLS SCT extension;
- included in the Online Certificate Status Protocol (OCSP) response associated with the certificate (stapled SCT).

d) Log Structure and Verifiability

CT logs are structured as append-only Merkle Hash Trees, enabling the generation of cryptographic proofs for both inclusion and consistency. These proofs allow independent entities to verify the integrity and append-only nature of the log over time.

e) Client Enforcement

Relying parties (e.g. web browsers and other client software) validate the presence and correctness of one or more SCTs while processing the certificate (e.g. during the establishment of a secure connection via TLS). Certificates that do not meet client-side CT policy requirements may be treated as untrusted or invalid.

4.3.2 Ecosystem Roles

The CT ecosystem involves several distinct functional roles, each contributing to the overall integrity, auditability, and transparency of certificate issuance:

a) Trust Service Provider acting as Certification Authority (TSP-CA)

An entity that issues X.509 certificates in accordance with applicable certificate policies and ETSI standards. The TSP-CA is responsible for submitting issued certificates or precertificates to one or more CT logs and for obtaining the corresponding SCTs. The TSP-CA has also to ensure that SCTs are made available to relying parties through suitable delivery mechanisms.

b) CT Log Operator

An entity that maintains and operates one or more CT logs. The log operator is accountable for the performance, availability, and correctness of log behaviour in accordance with defined specifications. Qualification requirements and third-party audit obligations for logs may apply based on applicable policies and relying-party programs.

c) Relying Party (Client)

A software component (e.g. browser, TLS client) that performs validation of SCTs as part of its certificate verification process. The relying party enforces CT compliance and may reject connections involving certificates that do not satisfy CT policy requirements.

d) Monitor

An independent system or organization that continuously observes CT logs to detect certificate entries matching specific criteria (e.g. certificates for a particular credential). Monitors may notify affected credential owners or other stakeholders in the event of unexpected or unauthorized certificate issuance.

e) Auditor

An independent verifier that examines the cryptographic structure and behaviour of CT logs. Auditors use cryptographic proofs to confirm that the log remains append-only, consistent, and free of tampering or retroactive modification. The auditor plays a critical role in maintaining trust in the CT infrastructure.

4.4 Policies, norms, standards and initiatives

4.4.1 Policies

Trust and interoperability in the CT ecosystem will be ensured by common policies on all ecosystem roles.

4.4.2 Norms and standards

The following standards currently exist, although IETF RFC 9162 [i.8] has seen limited deployment to date.

IETF RFC 6962 [i.7] - Certificate Transparency (June 2013)

IETF RFC 6962 [i.7] introduces an experimental protocol for publicly logging the existence of Transport Layer Security (TLS) certificates as they are issued or observed. It allows anyone to audit Certificate Authority (CA) activity and notice the issuance of suspect certificates, as well as to audit the certificate logs themselves.

IETF RFC 9162 [i.8] - Certificate Transparency Version 2.0 (December 2021)

IETF RFC 9162 [i.8] describes version 2.0 of the Certificate Transparency (CT) protocol for publicly logging the existence of TLS server certificates. It obsoletes IETF RFC 6962 [i.7] and specifies a new TLS extension used to send various CT log artifacts.

4.4.3 Initiatives

The Static Certificate Transparency API (Static-CT-API)

Static-CT-API is an experimental extension of Certificate Transparency introduced within the Chromium project and discussed in the CT Policy mailing list. Unlike the embedded SCT mechanism defined in IETF RFC 6962 [i.7] / IETF RFC 9162 [i.8], the Static-CT-API specifies a new log architecture based on "tiled logs" that aims to improve scalability, monitoring efficiency, and performance. It is currently documented in Chromium policy specifications and associated GitHub repositories but has not been standardized by the IETF or ETSI; instead, the interface is being maintained by the Community Cryptography Specification Project and referenced directly by browser policy documents. As of 2025, Chrome® accepts SCTs from Static-CT-API logs in combination with IETF RFC 6962 [i.7] logs, while other browser vendors are evaluating adoption. The Static-CT-API should therefore be considered a vendor-driven initiative in early deployment rather than a formal international standard.

5 Certificate Transparency

5.1 Use of Certificate Transparency with TLS

5.1.1 Introduction

CT is a mandatory policy component for publicly trusted Certification Authorities (CAs) issuing TLS server authentication certificates, as required by major browser vendors.

5.1.2 Precertificate and SCT Workflow

The issuance of a CT-compliant TLS certificate involves the following steps:

a) **Precertificate Generation**

Prior to the issuance of the final certificate, the CA generates a precertificate, which is structurally identical to the final certificate but includes a special X.509 extension (CT Poison extension) to indicate it is not valid for use in TLS.

b) **Submission to CT Logs**

The precertificate is submitted to one or more public CT logs that conform to IETF RFC 6962 [i.7] or IETF RFC 9162 [i.8]. Each log validates the submission and returns a Signed Certificate Timestamp (SCT), which includes:

- a hash of the submitted certificate or precertificate;
- a timestamp indicating receipt;
- a digital signature from the log.

c) **Final Certificate Issuance**

Upon receiving valid SCTs, the CA issues the final certificate. The SCTs may be embedded in the certificate, delivered through OCSP stapling, or transmitted during the TLS handshake via the TLS extension.

5.1.3 SCT Delivery Methods

The CT framework supports three delivery mechanisms for SCTs to the relying party (e.g. browser or TLS client):

- **Embedded SCT (Static CT):**
  SCTs are embedded directly in the X.509 certificate in a dedicated extension (`1.3.6.1.4.1.11129.2.4.2`). This is commonly used in environments where server support for dynamic SCT delivery is limited.

- **OCSP Stapling:**
  The CA includes SCTs in the OCSP response for the certificate. The server has then to staple this response during the TLS handshake.

- **TLS Extension (Dynamic SCT):**
  The server retrieves SCTs from the CA and delivers them during the TLS handshake using the `signed_certificate_timestamp` TLS extension.

Clients have to successfully validate at least one SCT for the certificate to be considered compliant with CT policies, subject to specific browser enforcement rules.

NOTE: In the present document, "Static CT" means embedded SCTs (IETF RFC 6962 [i.7] / IETF RFC 9162 [i.8]). The Static-CT-API is a different concept and is discussed separately in clause 4.4.3.

5.1.4 Log Requirements and Client Validation

CT logs are required to:

- Operate as publicly accessible, append-only, cryptographically verifiable Merkle Trees.
- Provide inclusion and consistency proofs as defined in IETF RFC 6962 [i.7] or IETF RFC 9162 [i.8].
- Meet operational parameters such as Maximum Merge Delay (MMD) and availability thresholds.

Relying parties (e.g. browsers) perform the following during TLS connection establishment:

- Parse and validate SCT(s) received via one or more of the supported methods.
- Verify that the SCT signature corresponds to a trusted CT log.
- Optionally check inclusion of the certificate in the CT log post-connection.

5.1.5 Browser Enforcement Policies

Browser enforcement (informative; policies subject to change). As of August 2025:

- Google® Chrome (Chromium): Enforces CT for publicly trusted TLS certificates in accordance with the Chrome CT Policy; acceptance criteria (including recognized logs and SCT delivery methods) are defined by the program policy.
- Apple Safari®: Enforces CT for publicly trusted TLS certificates as documented in Apple's CT policy for its root program (in effect for newly issued certificates since 2021); SCTs may be delivered embedded or via OCSP/TLS from recognized logs.
- Mozilla Firefox®: Enforces CT for publicly trusted TLS certificates starting with Firefox 135 (February 2025), in line with Mozilla's Root Store Policy; SCTs may be delivered embedded or via OCSP/TLS from recognized logs.
- Microsoft Windows / Edge®: Provides optional (opt-in) CT validation on recent Windows releases; Windows maintains a Certificate Transparency Log Monitor (CTLM) list of recognized logs. Current implementations focus on event logging, with possible enforcement extensions documented by the Microsoft Trusted Root Program (June 2025).

NOTE: Vendor policies evolve frequently. Implementers should consult the latest program documentation rather than rely on static summaries.

5.2 Certificate Transparency Log Incidents - Observations, Causes

5.2.1 Background

CT logs are expected to:

i) incorporate accepted entries within the log's configured Maximum Merge Delay (MMD); and

ii) meet ≥ 99 % per-endpoint availability measured over a rolling 90-day window.

Where expectations are not met, operators commonly communicate incident status, adjust operations, or, where appropriate, transition logs through the defined lifecycle states (Qualified, Usable, ReadOnly, Retired, Rejected). In practice, incidents have arisen from infrastructure faults, configuration and storage issues, and rare hardware errors that can compromise the Merkle tree state. Where recovery would break append-only guarantees, retirement of the affected shard or log has been used.

5.2.2 Incident taxonomy (operator-neutral)

The following incidents classes were identified:

A. Submission errors / service unavailability. Periods of elevated HTTP errors or temporary rejection of submissions (sometimes source-network specific) caused issuance delays until submitters failed over to alternate endpoints.

B. MMD breaches and inclusion gaps. Logs accepted precertificates but did not incorporate them within the declared MMD; in severe cases, accepted entries remained unincorporated and the shard moved towards deactivation.

C. Signing-key exposure risk (precautionary deactivation / retirement). A management-plane orchestration vulnerability created a credible risk that SCT-signing keys could have been exposed; there was no evidence of misuse, and the log was first placed in read-only mode and subsequently retired as a precaution.

D. Hardware-induced data corruption. A single-bit flip corrupted a leaf hash, rendering a shard irrecoverable without violating append-only properties.

E. Datacentre-level outages. Complete power loss and control-plane disruption at a hosting site produced multi-shard unavailability windows; service restoration relied on disaster-recovery procedures with subsequent backlog catch-up.

F. Storage exhaustion and database corruption. Disk-space exhaustion triggered index corruption and persistent merge failures with sustained MMD violations; the impacted shard transitioned to read-only operation and was retired on a defined schedule.

G. Consistency violations (inconsistent STHs). Logs that present conflicting Merkle tree views cannot be relied upon without a fresh rebuild.

5.2.3 Chronology of representative incidents (anonymized)

The following entries are operator-neutral and time-ordered; they illustrate the incident classes above without naming specific operators or shards.

- 2018-11-30 - Submission failures affecting CT endpoints. Elevated errors for submissions from a single CA network subsided after failover to an alternate site. Impact: transient issuance delays. Status: service restored after failover. (Class A)

- Q2 2019 - Availability drop with MMD alarms. Monitors observed availability below target with MMD breaches for accepted entries, consistent with capacity or merge-pipeline issues. Impact: delayed inclusion. Status:operator investigation and recovery; monitoring intensified. (Class B)
- May 2020 - Signing-key exposure risk (precautionary retirement). Following exploitation of widely publicized management-software vulnerabilities, one log's SCT-signing key may have been exposed. The log was deactivated and subsequently retired as a precaution. Impact: submitters required alternate SCT sources. Status:retired. (Class C)
- Mid-2021 - Hardware bit-flip corrupts Merkle state. A random bit flip corrupted a leaf hash, making proofs across that position unreliable; cryptographically safe repair was infeasible, so the shard was retired. Impact: proof errors for monitors spanning the affected range. Status: retired. (Class D)
- November 2023 - Datacentre power loss causing multi-shard outage. A hosting site experienced complete power loss, degrading control-plane functions and blocking acceptance / processing of new entries for documented windows; disaster-recovery facilities restored most services and logs caught up. Impact: temporary submission failure and delayed inclusion. Status: recovered. (Class E)
- January 2024 - Storage exhaustion → DB index corruption → retirement. Disk-space exhaustion caused database index corruption and sustained inability to merge accepted entries, with confirmed MMD violations; the shard entered read-only operation and was retired on a published schedule. Impact: SCTs from that shard no longer counted after retirement where runtime SCTs were required. Status: retired. (Class F)

Lifecycle note (not an incident): Ecosystem lifecycle clean-ups periodically remove older shards whose covered certificate sets have fully expired. This planned removal reduces operational load and does not indicate a failure.

5.3 Summary

Certificate Transparency is a mandatory mechanism in the issuance and validation of public TLS certificates. By mandating certificate disclosure to verifiable logs and enabling independent monitoring, CT significantly strengthens the accountability of CAs and the integrity of the PKI ecosystem. Proper implementation and enforcement of CT policies are essential to the ongoing trustworthiness of secure web communications.

6 Recommendations for the use of CT for certificates issued in accordance with Regulation (EU) No 910/2014

6.1 Transfer of the current use of CT to a CT ecosystem for certificates issued in accordance with Regulation (EU) No 910/2014

The use of Certificate Transparency (CT) in the WebPKI has proven effective in increasing accountability of certificate issuers, enabling early detection of misissuance, and supporting independent monitoring and auditing. It is proposed to make these benefits available for selected categories of certificates issued under Regulation (EU) No 910/2014 [i.4] and its implementing acts, so that issuance events become transparent and verifiable for relying parties and supervisory stakeholders in the European context.

To achieve this, the fundamental structure of the CT ecosystem should be adopted, namely the roles of certificate issuer (TSP-CA), CT log operator, monitor, auditor, and relying party, together with well-defined submission and verification artefacts (e.g. SCTs, inclusion and consistency proofs) and operational controls. Existing technical specifications (e.g. IETF RFC 6962 [i.7], with forward compatibility to IETF RFC 9162 [i.8] where feasible) provide a solid basis for formats and protocols; at the same time, it is proposed to establish a European CT infrastructure with appropriate governance, qualification criteria, and audit frameworks to address requirements arising from eIDAS, ensure operational resilience, and avoid undue dependencies on extra-European programs or policies. This approach preserves interoperability with widely deployed CT mechanisms while allowing Europe-specific policy objectives - such as regulatory oversight, lifecycle management, and long-term verifiability - to be met.

6.2 Recommendation for the use of existing standards, norms and initiatives

6.2.1 Recommendation for the use of existing standards

It is proposed that implementers build and operate the CT ecosystem in conformity with IETF RFC 6962 [i.7] as the primary basis for interoperability. This includes the generation and submission of precertificates or certificates to IETF RFC 6962 [i.7] conformant logs, the issuance and delivery of SCTs (embedded in the certificate or delivered via OCSP/TLS), and the provision and verification of inclusion and consistency proofs as defined therein. Where feasible, implementations should maintain forward compatibility with IETF RFC 9162 [i.8] without affecting current operations.

6.2.2 Recommendation to consider Static-CT-API (tiled logs)

It is proposed that the Static-CT-API (tiled-log architecture), currently maintained as a community specification, be considered as input to future standardization of CT formats and operations, alongside IETF RFC 6962 [i.7] and IETF RFC 9162 [i.8]. The approach offers operational advantages, most notably inclusion-before-SCT issuance (eliminating merge-delay failure modes), a static, cacheable read path (tiles and signed checkpoints suitable for distribution via object storage / CDNs), and improved resilience and archival verifiability (inclusion evidence can remain verifiable even after a log's operational retirement, noting that relying-party acceptance remains subject to client policy).

6.3 Recommendation for the definition of formats, protocols, policies and security requirements

6.3.1 Introduction

In order to ensure the reliable, secure, and auditable operation of the Certificate Transparency (CT) ecosystem, it is essential that each participating entity operates under a clearly defined and standardized policy framework.

To this end, ETSI should develop role-specific policy frameworks that define minimum requirements and structural expectations for policies applicable to all functional roles within the CT ecosystem. These policy frameworks will serve as a foundation for harmonised implementation and trust establishment across diverse environments and service providers.

The following roles are considered essential for standardized policy development:

- Trust Service Provider acting as Certification Authority (TSP-CA);
- Relying Party;
- Supervisory Body;
- CT Log Operator;
- CT Monitor; and
- CT Log Auditor.

6.3.2 Policy Framework for Trust Service Providers acting as Certification Authorities (TSP-CAs)

A policy framework for TSP-CAs should define mandatory provisions regarding:

- CT log selection criteria, including trust anchors and qualification status.
- SCT acquisition requirements, such as the number and diversity of required SCTs per certificate.
- SCT delivery mechanisms and applicable use cases (e.g. embedded, TLS extension, stapled via OCSP).

- Handling of precertificates, including generation, submission, and association with final certificates.
- Internal control measures, such as monitoring and verification of SCT validity before certificate issuance.

6.3.3 Policy Framework for Relying Parties

A relying-party policy framework should define:

- Transparent trust criteria including its publication (e.g. a versioned CT policy disclosing conditions for trusting a certificate).
- SCT verification rules.
- Incident and privacy handling.

6.3.4 Policy Framework for Supervisory Bodies

A supervisory policy framework should define:

- Ecosystem governance and role definitions, including baseline obligations for TSP-CAs, CT log operators, monitors, auditors, and relying parties.
- Admission and lifecycle rules for CT logs (registration, qualification, state transitions, delisting) with machine-readable publication (e.g. trusted list).
- Minimum measurement and reporting requirements (e.g. availability, MMD, inclusion timeliness) and formats for advisories and retirement schedules.
- Coordination and escalation procedures for ecosystem incidents, including stakeholder notification, time-boxed remediation windows, and graduated enforcement actions.
- Change management and transparency, including versioned policy updates, stakeholder consultation, and public change logs.

6.3.5 Policy Framework for CT Log Operators

The framework for CT log operators should define requirements related to:

- Log service behaviour, including the Maximum Merge Delay (MMD), certificate inclusion guarantees, and log integrity.
- Cryptographic controls, such as signing algorithms, hash functions, and key lifecycle management.
- Audit procedures, frequency, scope, and publication of results.
- Operational controls, including submission filtering, rate limiting, and mitigation of malicious activity.

6.3.6 Policy Framework for CT Monitors

A monitoring policy framework should define:

- Scope and frequency of log observation.
- Alerting mechanisms for affected credential owners or stakeholders in case of anomalies.
- Criteria for identifying relevant certificate entries.
- Data handling and protection practices, particularly regarding the processing of credential-related metadata.

6.3.7 Policy Framework for CT Log Auditors

The auditing policy framework should address:

- Verification methodology for inclusion and consistency proofs.
- Audit intervals and coverage scope.
- Procedures for reporting and publishing results.
- Processes for dealing with inconsistencies, including stakeholder notification and incident escalation.

History

Version	Date	Status
V1.1.1	March 2026	Publication