LAMPS
Internet-Draft
Intended status: Standards Track
Expires: 16 April 2026

T. Reddy
Nokia
J. Gray
Entrust
Y. Sheffer
Intuit

13 October 2025

X.509 Extensions for PQC or Composite Certificate Hosting Continuity
draft-reddy-lamps-x509-pq-commit-00

Abstract

This document specifies a new X.509 certificate extension, Post-Quantum or Composite Hosting Continuity (PQCHC), which enables a certificate subject to signal a intent to continue serving PQC or composite certificates for a defined continuity period after certificate expiration. This extension helps relying parties detect downgrade and man-in-the-middle (MitM) attacks during transition phases, where a cryptographically relevant quantum computer (CRQC) would make traditional certificates insecure.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 16 April 2026.

Copyright Notice

Table of Contents

1. Introduction

The migration to post-quantum cryptography (PQC) will not happen instantly. During the transition, servers are expected to host both traditional and PQC or composite certificates. This dual-hosting strategy ensures that services remain reachable by legacy clients that do not yet support PQC while also providing PQ or PQ/T authentication to updated clients. The decision to continue offering traditional certificates often depends on the size of the installed legacy base, the larger the base, the stronger the incentive to maintain traditional certificate support for accessibility. Relevant work on PQC certificates includes [I-D.ietf-lamps-dilithium-certificates] for ML-DSA and [I-D.ietf-lamps-x509-slhdsa] for SLH-DSA and [I-D.ietf-lamps-pq-composite-sigs] for composite certificates.

However, the continued use of traditional certificates becomes untenable once a cryptographically relevant quantum computer (CRQC) is known to exist. A publicly available CRQC would render classical public-key algorithms insecure, forcing server operators to revoke traditional certificates immediately, even if this disrupts access

for legacy clients. In practice, though, the availability of a CRQC may not be disclosed. Like a zero-day vulnerability, an adversary could secretly use a CRQC to compromise traditional certificates without alerting the wider ecosystem. In such a scenario, an attacker could suppress PQC or composite certificates and present only traditional ones to targeted clients, enabling an active MitM attack and giving the attacker full control over the encrypted session. Relying parties need a way to detect when a server that claims to support PQC or composite certificates suddenly offers only traditional credentials.

This document specifies a new X.509 certificate extension, Post-Quantum/Composite Hosting Continuity (PQCHC), which enables a certificate subject and its Certification Authority (CA) to assert an intent to continue serving PQC or composite credentials for a defined period beyond the nominal expiration of a certificate. This extension provides an operational signal that helps relying parties identify inconsistencies, for example, when a PQC-capable server unexpectedly stops advertising PQC or composite certificates and to take appropriate action to mitigate downgrade attacks.

The PQCHC extension is protocol-agnostic and applies to any PKI-based authentication context where a subject wishes to indicate continued availability of PQC or composite credentials to relying parties.

The PQCHC extension does not change certificate validation semantics and does not extend a certificate’s validity period. Instead, it provides an informational assurance of server behavior to help relying parties make security decisions during the transition to PQC.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. X.509 Certificate Extension

This section specifies the syntax and semantics of the PQC or composite Hosting Continuity (PQCHC) extension for X.509 public key certificates. This extension provides an informational assertion, made by the certificate subject, regarding the continued availability of pure PQC or composite credentials beyond the expiration date of the certificate that contains the extension.

3.1. Extension Overview

The PQCHC extension is an optional, non-critical X.509 certificate extension. When present, it indicates that the subject intends to continue deploying and presenting valid PQC or composite certificates both during the certificate's validity period and for the declared "continuityPeriod" after the certificate's "notAfter" date. This extension does not extend the certificate's validity period and does not modify path validation procedures as defined in [RFC5280].

The extension allows relying parties to detect inconsistencies during the PQC migration phase. For example, if a server previously declared an intent to keep its PQC certificate available for 12 months after expiry but suddenly presents only a traditional certificate, a relying party can infer potential misbehavior or active suppression of PQC credentials.

3.2. PQCHC Certificate Extension Definition

The PQCHC certificate extension MAY be included in public key certificates [RFC5280]. The PQCHC certificate extension MUST be identified by the following object identifier (OID):

```
id-pe-pqchc OBJECT IDENTIFIER ::= {
    iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-pe(1) TBD2
}
```

The extension MUST have the following syntax:

```
PQCHC ::= SEQUENCE {
continuityPeriod INTEGER (0..MAX),
policyURI IA5String OPTIONAL
}
```

The extension value is an ASN.1 SEQUENCE containing:

* continuityPeriod : an INTEGER (measured in days) indicating the number of days after the certificate's "notAfter" date that the subject intends to continue presenting valid PQC/composite certificates. A value of zero explicitly indicates that no intent exists beyond the certificate's expiry.
* policyURI (OPTIONAL): an IA5String containing a URI where the operator publishes additional information related to PQC deployment, such as migration information.

The extension MUST be marked non-critical so that relying parties that do not understand it will ignore it, consistent with [RFC5280] guidance for informational extensions.

3.3. Certificate Extension Processing

When this extension is present:

* The certificate subject asserts that the subject's server(s) will continue deploying and presenting valid PQC or composite certificates for at least the indicated continuity period after this certificate's expiration.
* The extension is a CA-signed declaration of the subject's operational intent and does not create a contractual service-level agreement between the subject and any relying party. Relying parties MAY use the indicated "continuityPeriod" to trigger local policy decisions, for example, by monitoring the revocation status of the declared PQC or composite certificate. If, during the stated period, the PQC or composite certificate remains unrevoked but is not observed in use, this may indicate a downgrade and MitM attack.

The extension does not imply that the CA will automatically issue new certificates, nor does it extend the cryptographic validity of an expired certificate. Rather, it signals the server operator's operational intent to maintain PQC or composite credentials in parallel with traditional credentials for the indicated transition period.

After successful certificate path validation [RFC5280], a relying party that observes a certificate containing the PQCHC extension can cache the following information:

* the server identity (as indicated in the certificate's SubjectAltName),
* the PQC or composite algorithm identifier (e.g., an OID such as id-ml-dsa-44 [I-D.ietf-lamps-dilithium-certificates]) associated with the end-entity certificate,
* the remaining lifetime of the certificate at the time of observation (i.e., the time between observation and the certificate's "notAfter" date).

The effective continuity window is the sum of the remaining lifetime and the "continuityPeriod". During this window, the relying party can expect the server to present certificates for the same subjectAltName that use the cached PQC or composite algorithm. If, within the effective continuity window, a relying party observes only a traditional certificate while the cached PQC/composite certificate

remains unrevoked, the relying party SHOULD treat the behavior as suspicious and terminate the connection. The relying party MAY apply local downgrade-detection policy, which can range from logging or raising an alert, to attempting an alternate network path.

If the operator changes from one PQC algorithm to another, the cached algorithm identifier will not match. In this case, the relying party MUST start a new continuity period. This also helps detect algorithm upgrades or downgrades (e.g., from ML-DSA-44 to ML-DSA-87).

Note: PQCHC may also apply to PQC-to-PQC transitions. For example, an operator might switch from ML-DSA to SLH-DSA in response to deployment guidance. The security implications of treating PQC-to-PQC changes as continuity events remain an open question and require further analysis as NIST' s standardization process progresses.

4. Applicability and Deployment Considerations

PQCHC is an informational mechanism and not a cryptographic guarantee. It must be combined with traditional mechanisms such as revocation checking. Its effectiveness depends on both correct configuration by servers and correct interpretation by relying parties.

PQCHC can also appear in certificates used for CRL signing or OCSP response signing to assert that any successor certificate for these roles will use a PQC or composite algorithm. If, within the declared continuity period, a new CRL signing or OCSP responder certificate is issued using only a traditional algorithm, relying parties can interpret this as a potential downgrade.

PQCHC cannot detect all failure modes. In particular, if a server silently ceases to present PQC or composite certificates without revoking them, relying parties will continue to observe only traditional certificates and have no authoritative indication whether the change was intentional or the result of an attack. In such cases PQCHC can only highlight the inconsistency.

Operators should choose "continuityPeriod" values that balance the detection benefit against operational flexibility. A "continuityPeriod" that is too short increases the window in which an attacker can cause an undetected downgrade; a "continuityPeriod" that is overly long may unduly constrain the operator's ability to phase out an algorithm in response to new cryptanalytic evidence.

5. Security Considerations

5.1. Certificate Caching Versus PQCHC

This section compares PQCHC to relying solely on client-side caching of previously observed PQC certificates and describes how PQCHC mitigates downgrade attacks.

A relying party could, in principle, remember that it previously observed a PQC or composite certificate and enforce that expectation in subsequent connections. However, such caching is limited by certificate expiration: once the cached certificate passes its "notAfter" date, the relying party can no longer rely on it for validation, and any enforced expectation is effectively lost. An attacker who can suppress PQC or composite certificates at that moment could cause a downgrade to traditional certificates without detection.

The PQCHC extension mitigates this limitation by providing a CA-signed assertion that the subject intends to continue deploying valid PQC or composite certificates for the configured continuityPeriod beyond the certificate's expiration. Because this assertion is signed by the issuing CA and carried in the certificate, it provides a durable, authenticated basis for relying parties to detect inconsistencies that simple client-side caching cannot address.

5.2. Downgrade attack detection

The PQCHC certificate extension provides a signal analogous to HTTP Strict Transport Security (HSTS) mechanism. Once a relying party has successfully observed a server presenting a PQC or composite certificate containing the PQCHC extension, it can "pin" the expectation that the server will continue to offer PQC or composite credentials until the date indicated by the extension.

This mechanism helps relying parties detect inconsistent server behavior:

* If a PQC or composite certificate carrying PQCHC is expected to remain valid until a stated date but is revoked early, the change can be interpreted as a legitimate operational decision by the server operator.
* If the certificate remains unrevoked yet is not observed by clients in subsequent connections, this anomaly may indicate an active downgrade attack in which a MitM suppresses the PQC or composite certificate and presents only traditional credentials.

5.3. Bootstrap limitation

Like HSTS mechanism, the PQCHC extension cannot provide protection on the very first connection. If an attacker successfully downgrades that initial handshake so the client never sees a PQC or composite certificate with PQCHC, the client will not learn the expected behavior and gains no additional assurance from this extension.

5.4. Revocation check

Because an active MitM can block PQC or composite certificates, relying parties cannot confirm server behavior solely by observing TLS handshakes. Certificate revocation remains the most reliable mechanism for determining when a server has intentionally stopped using a PQC or composite certificate.

Revocation can be signaled through CRLs published by CAs, OCSP, Browser curated revocation lists (e.g., CRLSet), or by relying on short-lived certificates that naturally expire quickly. During the transition to PQC, revocation information must remain verifiable by both legacy and upgraded clients. This requires that CRLs and OCSP responses be signed using both a traditional algorithm (e.g., RSA or ECDSA) for compatibility with legacy clients, and a PQC or composite algorithm for protection against CRQC adversaries. Legacy clients will continue to validate the traditional signature, while PQ-aware clients MUST verify the PQC or composite signature to ensure that revocation signaling itself cannot be forged.

Curated browser/OS revocation lists can provide emergency coverage but generally do not cover all domains, limiting their completeness. OCSP stapling is also insufficient in this setting, since although an OCSP response itself provides freshness, stapling relies on the server to deliver that response and cannot guarantee that PQC certificates are included when a CRQC-capable adversary acts as a MitM and suppresses them.

The PQCHC extension provides an informational signal that can support policy decisions, monitoring, and detection of possible downgrades. However, it is not a cryptographic proof of server behavior, and relying parties must continue to perform revocation checks to verify certificate validity. When inconsistencies are detected, relying parties may apply fallback mechanisms such as attempting connection through a different network interface or a VPN tunnel to help distinguish between genuine server changes and MitM downgrade attack.

A PQC certificate may be revoked for legitimate reasons such as key compromise, operational error, or migration to a new CA. An operator using the PQCHC facility SHOULD obtain and serve a new PQC

certificate for the remainder of the declared continuity period. This enables relying parties to continue to observe PQC credentials without requiring real-time revocation checking to distinguish legitimate operator actions from active downgrade attacks. In addition, this simplifies PQCHC by focusing on continuity of PQC credential availability rather than revocation mechanisms.

6. Privacy Considerations

Relying parties need to perform certificate revocation checks to determine whether the PQC or composite certificate has been intentionally withdrawn by the server. Revocation mechanisms such as Online Certificate Status Protocol (OCSP) queries to the CA can expose which servers a client is contacting, leading to privacy leaks. To mitigate this, revocation checking can be performed using privacy-preserving techniques such as Oblivious HTTP (OHAI) [RFC9458], where queries are relayed through intermediaries in a way that hides the client identity from the CA.

7. IANA Considerations

For the certificate extension in Section 3.2, IANA is requested to assign an object identifier (OID) for the extension (TBD2) with a Description of "id-pe-pqchc". The OID for the extension should be allocated in the "SMI Security for PKIX Certificate Extension" registry (1.3.6.1.5.5.7.1).

For the ASN.1 module in Appendix A , IANA is requested to assign an object identifier (OID) for the module identifier (TBD1) with a Description of "id-mod-pqchc". The OID for the module should be allocated in the "SMI Security for PKIX Module Identifier" registry (1.3.6.1.5.5.7.0).

8. Acknowledgments

Thanks to Mike Ounsworth for the discussion, review and comments.

9. Normative References

[I-D.ietf-lamps-dilithium-certificates]
Massimo, J., Kampanakis, P., Turner, S., and B. Westerbaan, "Internet X.509 Public Key Infrastructure - Algorithm Identifiers for the Module-Lattice-Based Digital Signature Algorithm (ML-DSA)", Work in Progress, Internet-Draft, draft-ietf-lamps-dilithium-certificates-13, 30 September 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-dilithium-certificates-13>.

[I-D.ietf-lamps-pq-composite-sigs]
Ounsworth, M., Gray, J., Pala, M., Klauner, J., and S. Fluhrer, "Composite ML-DSA for use in X.509 Public Key Infrastructure", Work in Progress, Internet-Draft, draft-ietf-lamps-pq-composite-sigs-12, 10 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-pq-composite-sigs-12>.

[I-D.ietf-lamps-x509-slhdsa]
Bashiri, K., Fluhrer, S., Gazdag, S., Van Geest, D., and S. Kousidis, "Internet X.509 Public Key Infrastructure: Algorithm Identifiers for SLH-DSA", Work in Progress, Internet-Draft, draft-ietf-lamps-x509-slhdsa-09, 30 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-x509-slhdsa-09>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC9458] Thomson, M. and C. A. Wood, "Oblivious HTTP", RFC 9458, DOI 10.17487/RFC9458, January 2024, <https://www.rfc-editor.org/rfc/rfc9458>.

[X680] ITU-T, "Information Technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, ISO/IEC 8824-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.680>.

[X690] ITU-T, "Information Technology -- Abstract Syntax Notation One (ASN.1): ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, ISO/IEC 8825-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.690>.

Appendix A. ASN.1 Module

The following module adheres to ASN.1 specifications [X680] and [X690].

```
<CODE BEGINS>
PQCHC-Module
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-pqchc (TBD) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- PKIX OID base
id-pkix OBJECT IDENTIFIER ::=
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) }

-- Module / extension arcs
id-mod OBJECT IDENTIFIER ::= { id-pkix 0 }
id-pe  OBJECT IDENTIFIER ::= { id-pkix 1 }

-- IANA: assign values for the following OIDs
id-mod-pqchc OBJECT IDENTIFIER ::= { id-mod TBD1 }   -- module identifier
id-pe-pqchc  OBJECT IDENTIFIER ::= { id-pe  TBD2 }   -- certificate extension OID

-- PQCHC extension ASN.1 syntax
PQCHC ::= SEQUENCE {
    continuityPeriod INTEGER (0..MAX),  -- number of days after certificate notAfter
                                        -- a value of 0 indicates no post-expiry inten
t
    policyURI         IA5String OPTIONAL
}

END
<CODE ENDS>
```

Authors' Addresses

Tirumaleswar Reddy
Nokia
Bangalore
Karnataka
India
Email: k.tirumaleswar_reddy@nokia.com

John Gray
Entrust Limited
2500 Solandt Road Suite 100
Ottawa, Ontario K2K 3G5
Canada
Email: john.gray@entrust.com

Yaron Sheffer
Intuit
7 HaPsagot St.
Petah Tikva
Israel
Email: yaronf.ietf@gmail.com