EAP Method Update
Internet-Draft
Intended status: Standards Track
Expires: 15 November 2026

T. Reddy
Nokia

14 May 2026

Post-Quantum Enhancements to TLS-Based EAP Methods
draft-ietf-emu-pqc-eap-tls-00

Abstract

This document proposes enhancements to TLS-based EAP methods, including the Extensible Authentication Protocol with Transport Layer Security (EAP-TLS), EAP Tunneled TLS (EAP-TTLS), Protected EAP (PEAP), and EAP Tunnel Method (TEAP), to incorporate post-quantum cryptographic mechanisms. It also addresses challenges related to large certificate sizes and long certificate chains, as identified in [RFC9191], and provides recommendations for integrating PQC algorithms into TLS-based EAP deployments.

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-emu-pqc-eap-tls/.

Discussion of this document takes place on the EAP Method Update Working Group mailing list (mailto:emu@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/emu. Subscribe at https://www.ietf.org/mailman/listinfo/emu/.

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 15 November 2026.

Table of Contents

1. Introduction

The emergence of a Cryptographically Relevant Quantum Computer (CRQC) would break the mathematical assumptions that underpin widely deployed public-key algorithms, rendering them insecure and obsolete. As a result, there is an urgent need to update protocols and infrastructure with post-quantum cryptographic (PQC) algorithms designed to resist attacks from both quantum and classical adversaries. The cryptographic primitives requiring replacement are discussed in [I-D.ietf-pquip-pqc-engineers], and the NIST PQC Standardization process has initially selected algorithms such as ML-KEM [FIPS203], ML-DSA [FIPS204], and SLH-DSA [FIPS205] for usage in security protocols.

To mitigate the risks posed by a CRQC, such as the potential compromise of encrypted data and the forging of digital signatures, existing security protocols must be upgraded to support PQC. These risks include "Harvest Now, Decrypt Later" (HNDL) attacks, where adversaries capture encrypted traffic today with the intent to

decrypt it once CRQCs become available. TLS-based EAP methods are widely used for network access authentication in enterprise and wireless environments. This document applies to all EAP methods that use TLS as their underlying transport, including EAP-TLS [RFC9190], EAP-TTLS [RFC5281], PEAP, and TEAP [RFC7170]. To continue providing long-term confidentiality and authentication guarantees, these methods must evolve to incorporate post-quantum algorithms.

However, transitioning these protocols to support PQC introduces practical challenges. [RFC9191] highlights issues related to large certificates and certificate chains in EAP-TLS, which can lead to session failures due to round-trip limitations. PQC certificates and certificate chains tend to be significantly larger than their traditional counterparts, further exacerbating these issues by increasing TLS handshake sizes and the likelihood of session failures. To address these challenges, this draft proposes mitigation strategies that enable the use of PQC within TLS-based EAP methods, ensuring secure and efficient authentication even in constrained network environments.

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.

This document adopts terminology defined in [I-D.ietf-pquip-pqt-hybrid-terminology]. For the purposes of this document, it is useful to categorize cryptographic algorithms into three distinct classes:

* Traditional Algorithm: An asymmetric cryptographic algorithm based on integer factorization, finite field discrete logarithms, or elliptic curve discrete logarithms. In the context of TLS, an example of a traditional key exchange algorithm is Elliptic Curve Diffie-Hellman (ECDH), which is almost exclusively used in its ephemeral mode, referred to as Elliptic Curve Diffie-Hellman Ephemeral (ECDHE).

* Post-Quantum Algorithm: An asymmetric cryptographic algorithm designed to be secure against attacks from both quantum and classical computers. An example of a post-quantum key exchange algorithm is the Module-Lattice Key Encapsulation Mechanism (ML-KEM).

* Hybrid Algorithm: We distinguish between key exchanges and signature algorithms:

  - Hybrid Key Exchange: A key exchange mechanism that combines two component algorithms

    o one traditional algorithm and one post-quantum algorithm. The resulting shared secret remains secure as long as at least one of the component key exchange algorithms remains unbroken.

  - PQ/T Hybrid Digital Signature: A multi-algorithm digital signature scheme composed of two or more component signature algorithms, where at least one is a post-quantum algorithm and at least one is a traditional algorithm.

Digital signature algorithms play a critical role in X.509 certificates, Certificate Transparency Signed Certificate Timestamps, Online Certificate Status Protocol (OCSP) statements, and any other mechanism that contributes signatures during a TLS handshake or in the context of a secure communication establishment.

3. Data Confidentiality in TLS-Based EAP Methods

One of the primary threats to TLS-based EAP methods is the HNDL attack. In this scenario, adversaries can passively capture EAP-TLS handshakes such as those transmitted over the air in Wi-Fi networks and store them for future decryption once CRQCs become available.

While EAP-TLS 1.3 [RFC9190] provides forward secrecy through ephemeral key exchange and improves privacy by encrypting client identity and reducing exposure of session metadata, these protections rely on the security of the underlying key exchange algorithm. In the presence of a CRQC, traditional key exchange mechanisms (e.g., ECDHE) would no longer provide long-term confidentiality. In such cases, an adversary could mount an HNDL attack by passively recording EAP-TLS handshakes and decrypting the captured traffic once quantum-capable cryptanalysis becomes feasible. This could retroactively expose information that TLS 1.3 is otherwise designed to protect, including:

* The identity of the authenticated client.

* Client credentials used in certificate-based authentication (e.g., usernames, device or organization identifiers).

* In the case of EAP-TTLS and TEAP, HNDL attacks present an additional threat. These methods typically carry legacy inner authentication protocols within the outer TLS tunnel, such as MS-CHAPv2. If a CRQC is used to break the outer TLS tunnel, the exposed inner authentication exchange could enable offline password attacks, potentially allowing an adversary to recover user credentials.

To preserve the intended privacy guarantees of TLS 1.3 and to protect against HNDL attacks, TLS-based EAP deployments that require long-term confidentiality will need to adopt post-quantum key exchange mechanisms, as outlined in Section 4 of [I-D.ietf-uta-pqc-app].

These mechanisms ensure that even if handshake data is recorded today, it cannot be decrypted in the future, maintaining the confidentiality and privacy of the TLS session.

Furthermore, to support hybrid or PQC-only key exchange in bandwidth or latency-constrained EAP deployments, EAP clients and servers should apply the optimizations described in Section 4.1 of [I-D.ietf-uta-pqc-app] to minimize performance overhead.

4. Post-Quantum Authentication in TLS-Based EAP Methods

Although a CRQC would primarily impact the confidentiality of recorded TLS sessions, it could also pose risks to authentication mechanisms that rely on traditional public-key algorithms with long-lived credentials. In particular, if quantum-capable cryptanalysis were to become practical within the validity period of a certificate, an adversary could recover the private key corresponding to a traditionally signed certificate and subsequently impersonate the certificate holder in real time. The feasibility and impact of such attacks depend on several factors, including certificate lifetimes and key management practices.

TLS-based EAP deployments rely on X.509 certificates issued by CAs, and the transition to PQ certificate authentication is constrained by the long lifecycle associated with distributing, deploying, and validating new trust anchors. If CRQCs arrive sooner than anticipated, deployed authentication systems may lack the agility to transition credentials and trust anchors in a timely manner.

As a result, deployments that rely on long-lived certificates or that require resistance to future quantum-capable adversaries face an increased risk of authentication compromise. In such scenarios, an on-path attacker that is able to recover a server's private key within the certificate validity period could impersonate access points (APs) in real time, potentially deceiving users into revealing credentials or connecting to rogue networks.

To mitigate these risks, TLS-based EAP deployments will need to adopt, over time, either PQ or PQ/T hybrid certificate-based authentication, as described in Section 5 of [I-D.ietf-uta-pqc-app].

The use of PQ or PQ/T hybrid certificates increases the size of individual certificates, certificate chains, and signatures, resulting in significantly larger handshake messages. These larger payloads can lead to packet fragmentation, retransmissions, and handshake delays, issues that are particularly disruptive in constrained or lossy network environments.

To address these impacts, TLS-based EAP deployments can apply certificate chain optimization techniques outlined in Section 6.1 of [I-D.ietf-uta-pqc-app] to reduce transmission overhead and improve handshake reliability.

5. EST Integration

The EAP client is expected to validate the certificate presented by the EAP server using a trust anchor that is provisioned out-of-band prior to authentication (e.g., using EST). The intermediate certificates are provided by the EAP server during the TLS handshake. The EAP client relies solely on the pre-provisioned trust anchor to build and validate the certificate chain. This model assumes a managed deployment environment with explicitly configured trust relationships between the EAP client and EAP server.

To further reduce handshake overhead, particularly in deployments using large certificate chains due to post-quantum (PQ) or composite certificates, this draft proposes an optimization that leverages the Enrollment over Secure Transport (EST) protocol [RFC7030], extended by [RFC8295]. Specifically, it allows intermediate certificates to be retrieved in advance by using EST, thereby avoiding the need to transmit them during each TLS handshake.

For EAP methods that use TLS as an outer tunnel (e.g., PEAP and TEAP), the EST optimization described in this section applies to the certificates used in the outer TLS tunnel. The EST pre-fetching of client intermediate certificates is relevant only when mutual TLS authentication is used. This is always the case for EAP-TLS, and optionally the case for EAP-TTLS and TEAP when client certificate authentication is used in the outer tunnel.

This section defines extensions to EST to support retrieval of the certificate chain used by an EAP server and EAP clients. The first extension enables clients to obtain access to the complete set of published intermediate certificates of the EAP server.

A new path component is defined under the EST well-known URI:

GET /.well-known/est/eapservercertchain

The ’/eapservercertchain’ is intended for informational retrieval only and does not require client authentication. It allows clients to retrieve the intermediate certificate chain that the EAP server presents during TLS handshakes. This request is performed using the HTTPS protocol. The EST server MUST support requests without requiring client authentication. The certificate chain provided by the EST server MUST be the same certificate chain the EAP server uses in a TLS-based EAP session.

The second extension enables EAP servers to obtain access to the complete set of published intermediate certificates of the EAP clients. Rather than relying on static configuration, the EAP server can dynamically fetch the client’s intermediate certificate chain from a trusted EST server within the same administrative domain.

A new path component is defined under the EST well-known URI:

GET /.well-known/est/eapclientcertchain

The ’/eapclientcertchain’ is intended for informational retrieval only and does not require client authentication. It allows the EAP server to retrieve the intermediate certificate chain that the EAP clients present during TLS handshakes. This request is performed using the HTTPS protocol. The EST server MUST support requests without requiring client authentication. The certificate chain provided by the EST server MUST be the same certificate chain EAP clients use in the TLS-based EAP session.

EAP clients and servers MUST authenticate the EST server using a trust anchor obtained via a suitable bootstrapping mechanism before retrieving intermediate certificate chains via HTTPS. Various

bootstrapping mechanisms exist for establishing this trust, such as BRSKI [RFC8995], EST [RFC7030], or out-of-band provisioning. The choice of bootstrapping mechanism is a deployment decision and is out of scope for this document. Certificate chains retrieved from an unauthenticated or untrusted EST server MUST NOT be used for TLS chain validation.

EAP servers and clients are RECOMMENDED to cache retrieved certificate chains to reduce latency and network overhead. However, they SHOULD implement mechanisms to detect changes or expiration. These include periodic re-fetching, honoring HTTP cache control headers (e.g., Cache-Control, ETag), and verifying the validity period of intermediate certificates.

EAP clients MAY omit intermediate certificates from the TLS handshake only if they have been explicitly configured by the administrator to do so. Such configuration is recommended only in deployments where both the EAP client and EAP server support this specification and have completed EST pre-fetching as part of provisioning. If no such configuration is present, the EAP client MUST include the full certificate chain in the TLS handshake. Similarly, an EAP server MAY omit intermediate certificates from the TLS handshake only if it has been explicitly configured by the administrator to do so. Administrators are advised to ensure that clients in the deployment have retrieved the server's intermediate certificates via EST as part of their provisioning process before enabling this configuration.

Note: A TLS extension could be used to explicitly signal support for intermediate certificate omission between peers, avoiding the need for administrator configuration. Such a mechanism is considered a possible future solution but is out of scope for this document.

6. Security Considerations

The security considerations outlined in [I-D.ietf-uta-pqc-app] and [I-D.ietf-pquip-pqc-engineers] must be carefully evaluated and taken into account for all TLS-based EAP deployments.

7. IANA Considerations

This document defines two new path components under the EST well-known URI '/.well-known/est/', following the extension mechanism established by [RFC8295]: '/eapservercertchain' and '/eapclientcertchain'. As these are sub-paths under the already-registered '/.well-known/est/' prefix defined in [RFC7030], no new IANA registry entries are required.

Acknowledgements

Thanks to John Mattsson, Hannes Tschofenig, Alan Dekok and Michael Richardson for the discussion and comments.

References

Normative References

[I-D.ietf-uta-pqc-app]
Reddy.K, T. and H. Tschofenig, "Post-Quantum Cryptography Recommendations for TLS-based Applications", Work in Progress, Internet-Draft, draft-ietf-uta-pqc-app-01, 24 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-uta-pqc-app-01>.

[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>.

[RFC7030] Pritikin, M., Ed., Yee, P., Ed., and D. Harkins, Ed., "Enrollment over Secure Transport", RFC 7030, DOI 10.17487/RFC7030, October 2013, <https://www.rfc-editor.org/rfc/rfc7030>.

[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>.

[RFC8295] Turner, S., "EST (Enrollment over Secure Transport) Extensions", RFC 8295, DOI 10.17487/RFC8295, January 2018, <https://www.rfc-editor.org/rfc/rfc8295>.

[RFC9190] Preu Mattsson, J. and M. Sethi, "EAP-TLS 1.3: Using the Extensible Authentication Protocol with TLS 1.3", RFC 9190, DOI 10.17487/RFC9190, February 2022, <https://www.rfc-editor.org/rfc/rfc9190>.

Informative References

[FIPS203] National Institute of Standards and Technology (NIST), "Module-Lattice-Based Key-Encapsulation Mechanism Standard", FIPS 203, 2024.

[FIPS204] National Institute of Standards and Technology (NIST), "Module-Lattice-Based Digital Signature Standard", FIPS 204, 2024.

[FIPS205] National Institute of Standards and Technology (NIST), "Stateless Hash-Based Digital Signature Standard", FIPS 205, 2024.

[I-D.ietf-pquip-pqc-engineers] Banerjee, A., Reddy.K, T., Schoinianakis, D., Hollebeek, T., and M. Ounsworth, "Post-Quantum Cryptography for Engineers", Work in Progress, Internet-Draft, draft-ietf-pquip-pqc-engineers-14, 25 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-pquip-pqc-engineers-14>.

[I-D.ietf-pquip-pqt-hybrid-terminology] D, F., P, M., and B. Hale, "Terminology for Post-Quantum Traditional Hybrid Schemes", Work in Progress, Internet-Draft, draft-ietf-pquip-pqt-hybrid-terminology-06, 10 January 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-pquip-pqt-hybrid-terminology-06>.

[RFC5281] Funk, P. and S. Blake-Wilson, "Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)", RFC 5281, DOI 10.17487/RFC5281, August 2008, <https://www.rfc-editor.org/rfc/rfc5281>.

[RFC7170] Zhou, H., Cam-Winget, N., Salowey, J., and S. Hanna, "Tunnel Extensible Authentication Protocol (TEAP) Version 1", RFC 7170, DOI 10.17487/RFC7170, May 2014, <https://www.rfc-editor.org/rfc/rfc7170>.

[RFC8995] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", RFC 8995, DOI 10.17487/RFC8995, May 2021, <https://www.rfc-editor.org/rfc/rfc8995>.

[RFC9191] Sethi, M., Preu Mattsson, J., and S. Turner, "Handling Large Certificates and Long Certificate Chains in TLS-Based EAP Methods", RFC 9191, DOI 10.17487/RFC9191, February 2022, <https://www.rfc-editor.org/rfc/rfc9191>.

Author’s Address

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