Hybrid Certificate with Optional Post-Quantum Key/Signature

IETF 121 - LAMPS WG

Sun Shuzhou, He Yidi, **Lin Hsiao Ying**

Huawei

Datatracker: draft-sun-lamps-hybrid-scheme

Can be viewed as a combination of draft-truskovsky-lamps-pq-hybrid-x509 and draft-ounsworth-lamps-pq-external-pubkeys.

At a Glance

- The solved problem:
  A hybrid certificate supporting multiple public keys while (1) saving the potential transmission overhead for legacy devices and (2) keeping backward compatibility
- Core idea: Sign(Hash), Transmit(Hash/Value), Verify(Hash)
- Action(s): Propose new CSR attributes and extensions to implement

- Relations to existing solution(s):
  - draft-truskovsky-lamps-pq-hybrid-x509: cannot support transmit public key hash
  - ITU-T X.509 (10/2019): only support one pk/sig used at a time
  - draft-ounsworth-lamps-pq-external-pubkeys: cannot support transmit public key value

draft-truskovsky-lamps-pq-hybrid-x509

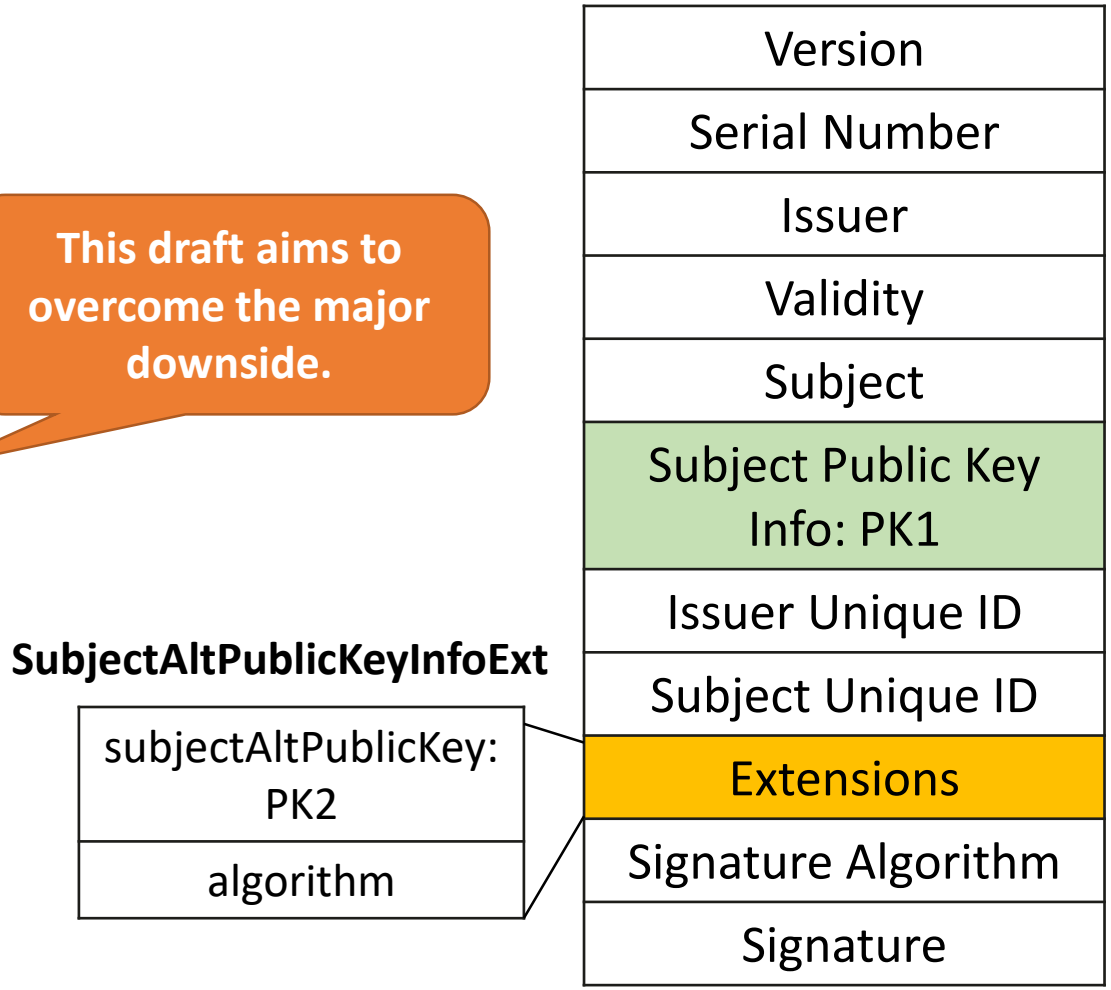
Multiple Public-Key Algorithm X.509 Certificates

Abstract

Tombstone notice:

This draft is no longer being pursued at the IETF. A variant of this proposal was adopted in [itu-t-x509-2019], which allows two keys to be placed in a certificate but only one used at a time. The major downside of this proposal is that it requires the large PQC key to be sent even to legacy clients which will not use it. Additionally, this proposal does not present a generic encoding for the multiple signatures produced by the multiple keys contained in a hybrid certificate, leaving the responsibility to dependent protocols and applications for how to carry multiple signatures and how to signal that multiple signatures should have been present in order to detect stripping attacks. As such, this document represents only a partial solution to the dual-signature problem. How, and whether, to implement dual-signatures is an active and ongoing discussion topic at the IETF and work continues in this area across several working groups. The PQUIP WG serves as a central location for all PQC-related discussion.

This draft aims to overcome the major downside.

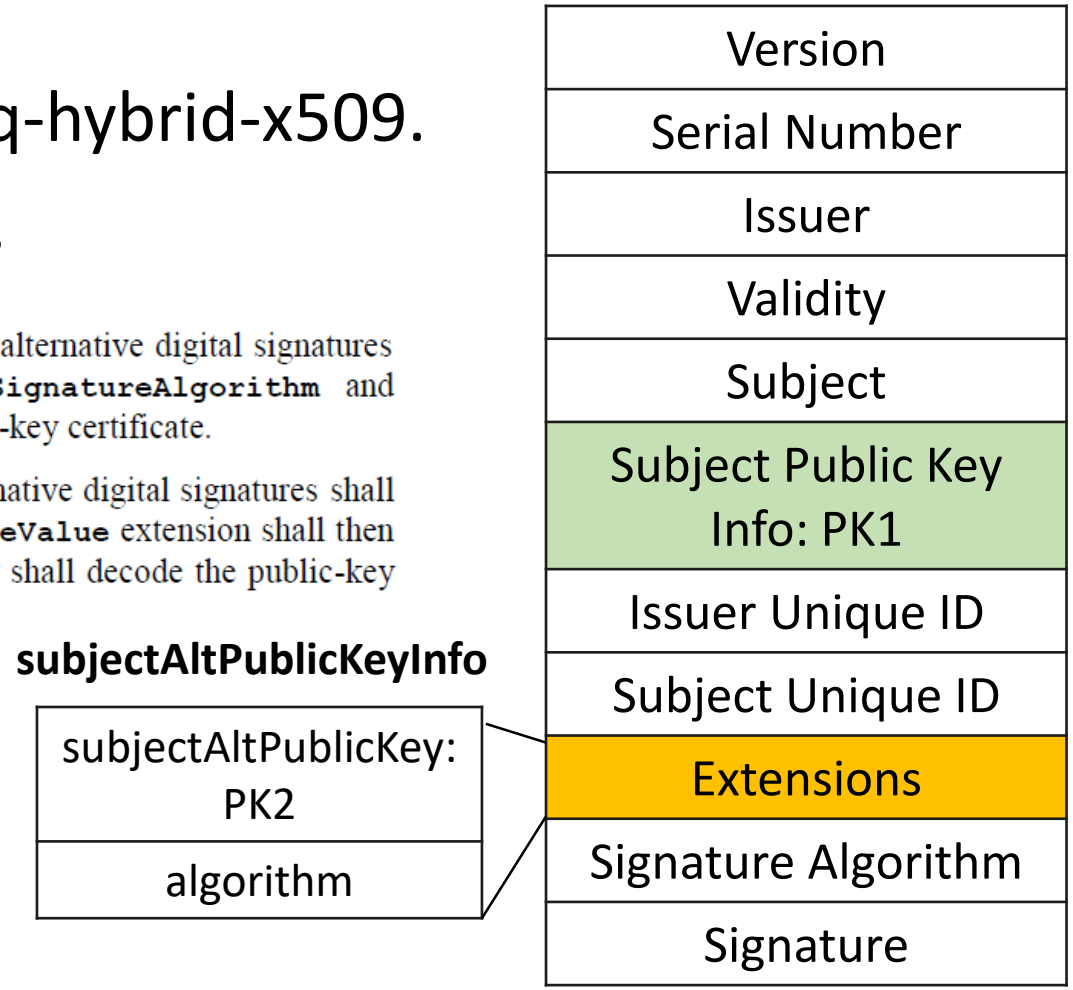


ITU-T X.509 (10/2019)

- A variant of draft-truskovsky-lamps-pq-hybrid-x509.
- Only allows one pk/sig used at a time.

A relying party that has not migrated to support alternative cryptographic algorithms and alternative digital signatures shall verify the native digital signature. The **`subjectAltPublicKeyInfo`**, **`altSignatureAlgorithm`** and **`altSignatureValue`** extensions shall then be included in the DER encoding of the public-key certificate.

A relying party that has migrated to support alternative cryptographic algorithms and alternative digital signatures shall verify the alternative digital signature. The **`signature`** component and the **`altSignatureValue`** extension shall then be excluded from the DER encoding of the public-key certificate. Thus, the relying party shall decode the public-key

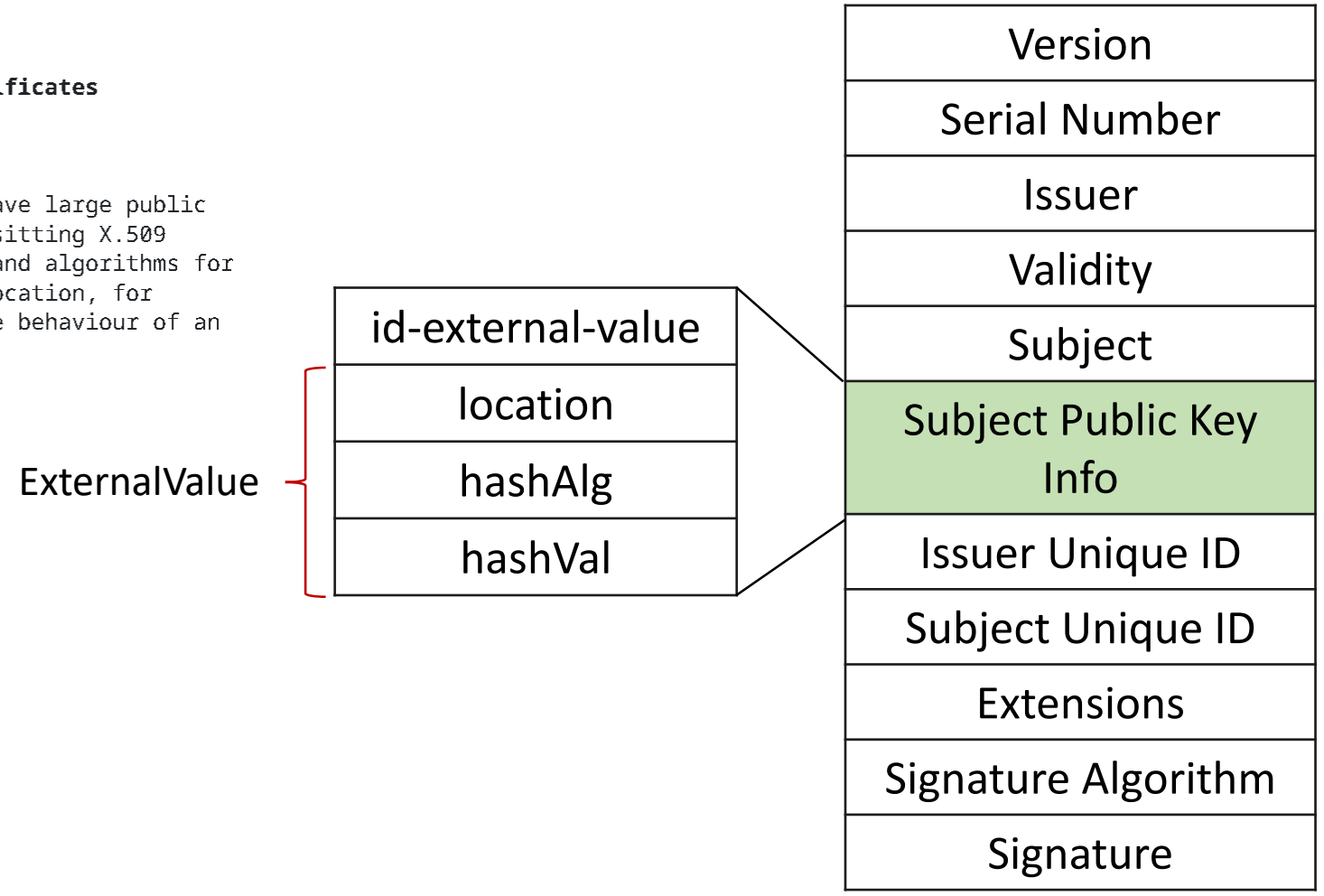


draft-ounsworth-lamps-pq-external-pubkeys

External Keys For Use In Internet X.509 Certificates

Abstract

Many of the post quantum cryptographic algorithms have large public keys. In the interest of reducing bandwidth of transitting X.509 certificates, this document defines new public key and algorithms for referencing external public key data by hash, and location, for example URL. This mechanism is designed to mimic the behaviour of an Authority Information Access extension.

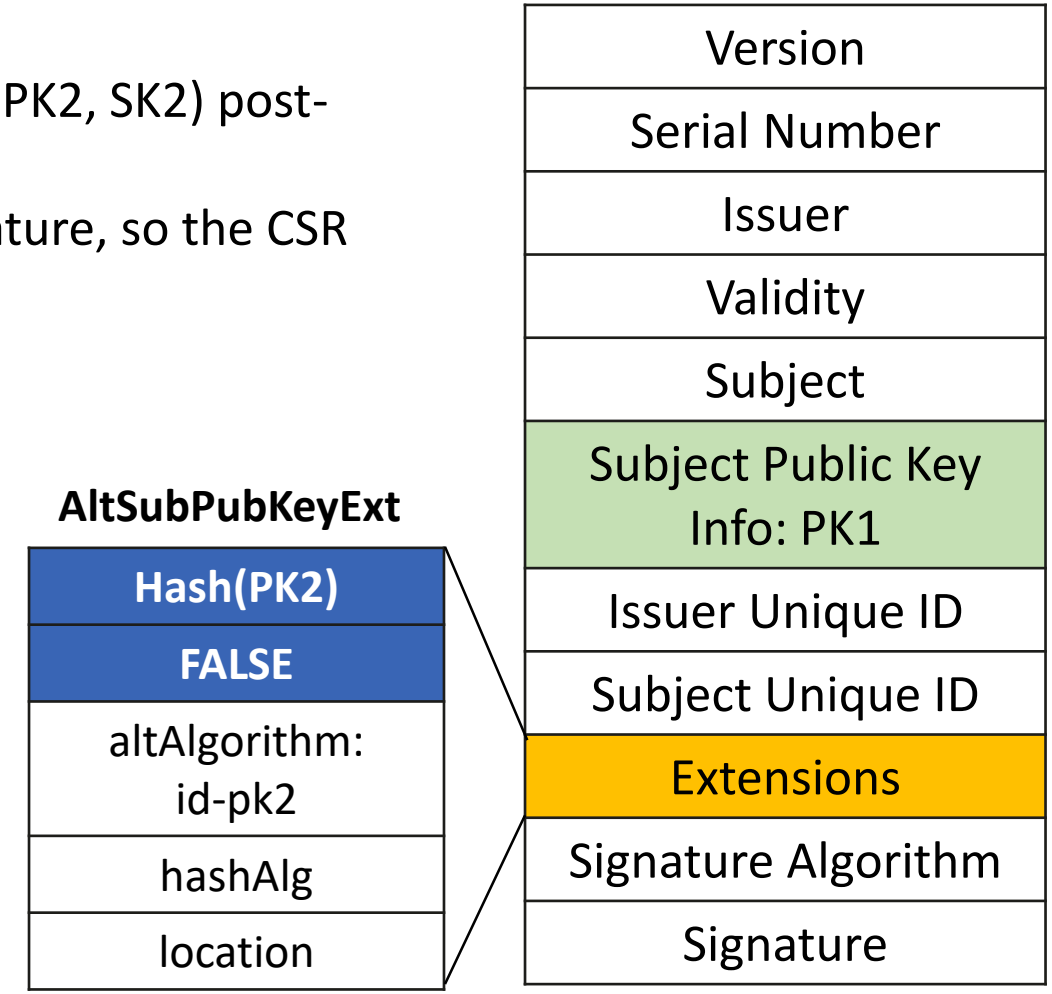


Certificate Issuance

1. Subscriber has two key pairs: (PK1, SK1) traditional, (PK2, SK2) post-quantum
2. Subscriber generates a CSR with the composite signature, so the CSR syntax does not need to be changed.

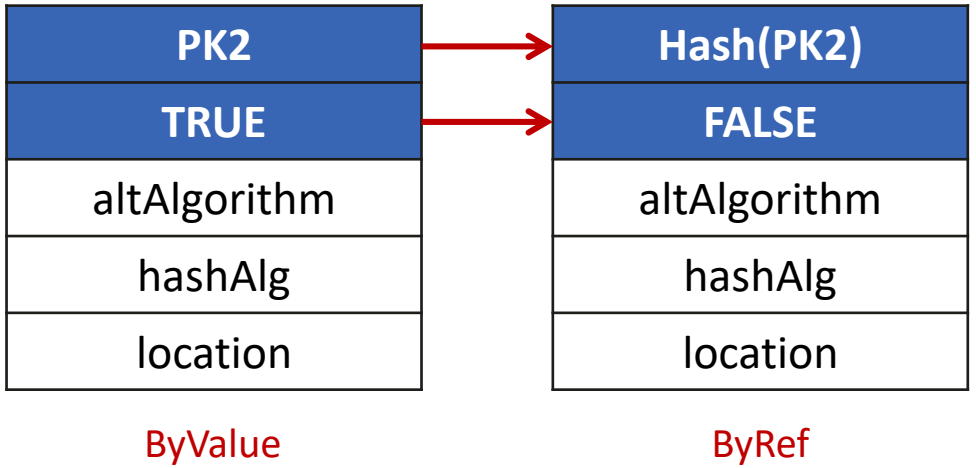
CSR: PK1 || PK2, …, Sig1 || Sig2

3. CA verifies the CSR and generates a certificate:
   - PK1 is in the SPKI.
   - Hash of PK2 is in the new proposed extension.



Certificate Transmission & Verification

1. Subscriber has the ability to construct two forms of the proposed hybrid certificate.
2. Subscriber and Relying Party negotiate the exact form to be transmitted, e.g., in TLS.
   - If RP supports PQC, send ByValue certificate.
   - If RP does not support PQC, send ByRef certificate.
3. Relying Party obtains one or two keys from the certificate:
   - Convert ByValue form to ByRef form
   - Verify the certificate as usual



Advantage: If RP don't support PQC, then: 1) No PQC keys/sigs will be sent, only their hashes. 2) Still can use traditional algorithms.
Disadvantage: Maybe it's not easy to accept that one can change the content of a certificate.

Content in the Draft

1. Define 4 CSR attributes for specifying hash algorithms and locations
2. Define 2 certificate extensions: AltSubPubKeyExt and AltSignatureExt
3. Describe scheme workflow:
    A. CSR creation and verification
    B. Extension creation
    C. Certificate creation, transmission, and verification

Thank You

Two Signatures

CA has two key pairs: (PK1, SK1) traditional, (PK2, SK2) post-quantum.

The sign procedure follows draft-truskovsky-lamps-pq-hybrid-x509.