Related Certificate Descriptor

https://datatracker.ietf.org/doc/draft-lamps-okubo-certdiscovery

Tomofumi Okubo Corey Bonnell John Gray Mike Ounsworth Joe Mandel

Discovery -- RelatedCertificateDescriptor

- `draft-lamps-okubo-certdiscovery` defines an X.509 extension:

  “This document specifies the new certDiscovery access method for X.509 Subject Information Access (SIA) extension defined in [RFC5280].”

```
RelatedCertificateDescriptor :: SEQUENCE {
          uniformResourceIdentifier    IA5String,
          signatureAlgorithm   [0] IMPLICIT AlgorithmIdentifier OPTIONAL,
          publicKeyAlgorithm   [1] IMPLICIT AlgorithmIdentifier OPTIONAL,
  }
```

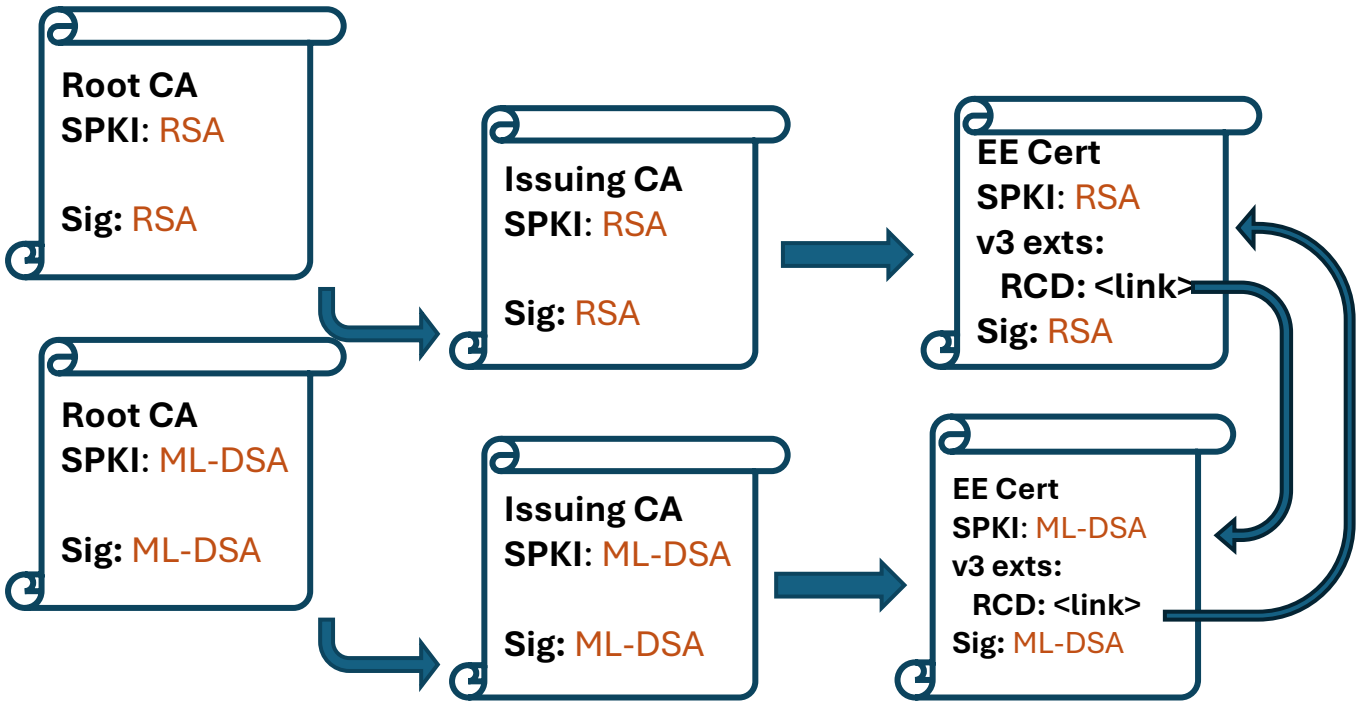
- A more flexible way of cross-referencing related certificates.

https://datatracker.ietf.org/doc/draft-lamps-okubo-certdiscovery/

Discovery -- RelatedCertificateDescriptor

Backwards Compat	Hybrid Security	Discovery
?	?	✓

- Allows for cross-linking End Entity (and potentially CA) certs in one or both directions.
- Can contain a URL so that the related cert is easy to fetch, especially in the case where the client has never seen the related cert before.
- Backwards compatibility and Hybrid security are not automatic, but clever verifiers could use the RCD to achieve them.



Comparison with RFC 5697

```
This extension MUST NOT be marked critical.

   id-pe-otherCerts OBJECT IDENTIFIER ::= { id-pe 19 }

   OtherCertificates ::= SEQUENCE OF SCVPCertID

   (RFC 5055)
    SCVPCertID ::= SEQUENCE {
     certHash         OCTET STRING,
     issuerSerial     SCVPIssuerSerial,
     hashAlgorithm    AlgorithmIdentifier DEFAULT { algorithm sha-1 } }

     SCVPIssuerSerial ::= SEQUENCE {
        issuer          GeneralNames,
        serialNumber    CertificateSerialNumber
     }
```

- This is close to what we want, but:
  1. A fetchable URL would be better.
  2. Given that the certHash is a required field, it is not possible to simultaneously issue certificates that reference each other with this mechanism.

Comparison with draft-ietf-lamps-cert-binding-for-multi-auth

```
--  Object Identifiers for certificate extension
  id-relatedCert OBJECT IDENTIFIER ::= { TBD2 }

--  X.509 Certificate extension
  RelatedCertificate ::= OCTET STRING
                -- hash of entire related certificate }
```

- In general, a cert hash does not help a client to learn about a new certificate, so a fetchable URL would be better.
- Given that the certHash is a required field, it is not possible to simultaneously issue certificates that reference each other with this mechanism.

Certificate Discovery – Add Hash?

Authors group discussing adding an optional hash of the certificate (borrowing from RFC 5697). This would add stronger binding in 1 direction but would constrain issuance order.

```
RelatedCertificateDescriptor ::= SEQUENCE {
    uniformResourceIdentifier IA5String,
    signatureAlgorithm [0] IMPLICIT AlgorithmIdentifier OPTIONAL,
    publicKeyAlgorithm [1] IMPLICIT AlgorithmIdentifier OPTIONAL,
    certHash [2] IMPLICIT CertHash OPTIONAL }

CertHash ::= SEQUENCE {
    value OCTET STRING,
    hashAlgorithm AlgorithmIdentifier DEFAULT {algorithm sha-256} }
```

Certificate Discovery – Next Steps

- Addressed Feedback from Russ and Carl
- Adoption?