draft-ietf-lamps-csr-attestation

Mike Ounsworth, Hannes Tschofenig,
Henk Birkholtz

LAMPS 118

Context / Goal

- Vendors already have attestation data (evidence or endorsements) in <any format>, and we need a way to put them into a CSR.

- The draft defines CSR Attributes / Extensions to carry arbitrary attestation statements.

ENTRUST

Diff from 117 (pre-adoption-00): Multiple attestations and multiple cert chains

Original

```
 AttestStatement ::= SEQUENCE {
     type   OBJECT IDENTIFIER,
     value  ANY
   }
AttestCertsAttribute ATTRIBUTE ::= {
    TYPE SET OF CertificateChoice
    COUNTS MAX 1
    IDENTIFIED BY id-aa-attestChainCerts
  }
```

The problem was that this will scale badly (yield a very large cert bag) if there are many evidence and endorsement statements in a single CSR.

In theory cert path builders should be able to handle that, but in practice …

In particular, if multiple evidence and endorsement statements each come with their own cert chains, then needing to merge them into a single cert bag, de-duplicate, etc, could be complicated.

ENTRUST

Diff from 117 (pre-adoption-00): Multiple attestations and multiple cert chains

Current

```
EvidenceBundle ::= SEQUENCE
{
  evidence  SEQUENCE OF EvidenceStatement,
  certs SEQUENCE OF CertificateAlternatives OPTIONAL
}
```

This was Carl's suggestion; "best of all worlds";

- Attestations (evidence / endorsements) with identical or overlapping cert chains can be grouped into one EvidenceBundle.
- Attestations with different cert chains can be separated to help the verifier and allow easier construction of "compound" attestations without needing to merge cert chains.

@MSJ: You raised objections on-list to this change, can you come to the mic to explain them?

ENTRUST

Diff from 117 (pre-adoption-00): CRMF support

- We now have ASN.1 for both PKCS#10 ATTRIBUTES and CRMF EXTENSIONS.
  - With the caveat that these are CRMF CSR EXTENSIONS and **not** certificate extensions.
  - Thanks Hendrik for pointing out that CRMF support was missing.

```
-- For PKCS#10
attr-evidence ATTRIBUTE ::= {
  TYPE SEQUENCE OF EvidenceBundle
  IDENTIFIED BY id-aa-evidenceStatement
}
```

```
-- For CRMF
ext-evidence EXTENSION ::= {
  SYNTAX SEQUENCE OF EvidenceBundle
  IDENTIFIED BY id-aa-evidenceStatement
}
```

"The Extension version is intended only for use within CRMF CSRs and MUST NOT be used within X.509 certificates due to the privacy implications of publishing Evidence about the end entity's hardware environment. See Section 6 for more discussion."

ENTRUST

Diff from 117 (pre-adoption-00): CertificateAlternatives

- Renamed from “CertificateChoice” to avoid conflict with RFC 5652’s
- Removal of opaqueCert since that usecase is well-enough covered by typedCert

```
CertificateAlternatives ::=
   CHOICE {
      cert Certificate,
      ~~opaqueCert     [0] IMPLICIT OCTET STRING,~~
      typedCert      [0] IMPLICIT TypedCert,
      typedFlatCert [1] IMPLICIT TypedFlatCert
   }
```


ENTRUST

Open design question: nonces and freshness

- Philosophical conflict:
  - RATS Architecture assumes that the RP has an online connection where it can ask the Attester for Evidence, and receive a provably fresh one (ex.: via nonce).
  - CSRs provide no guarantee of freshness. In fact, often this is undesirable;
    - CSR PEM is generated in an air-gapped network and carried out (ex.: ACME).
    - It is common to see reuse of the same CSR across multiple CAs, or across multiple cert renewals with the same CA (this is bad practice probably, but still very common).
- We have included some hand-wavy text in Security Consideration 6.1 saying:

  "… thus the issue of freshness is left up to the discretion of protocol designers and implementors."
- *draft-tschofenig-lamps-nonce-cmp-est* will be presented later by Hendrik which adds appropriate attestation nonces for CMP and EST.

ENTRUST

WGLC?

- Probably not quite yet, but getting close.

ENTRUST