RATS PKIX Evidence

Hannes Tschofenig, Siemens
Jean-Pierre Fiset, Crypto4A Technologies Inc.
Mike Ounsworth, Entrust Inc.

RATS 120

Introduction

- Design team effort so far has been on a LAMPS about defining a CSR extension to carry evidence to help a Certification Authority (CA) to approve the issuance of a certificate.
  - Motivation: policies from CA/B forum on certificate issuance
  - Good progress: draft-ietf-lamps-csr-attestation in WGLC.
- To continue the effort, **we are essentially proposing an ASN.1 port of RATS-EAT**
  - can be re-used in X.509 certificate extensions; and,
  - generic and extensible – essentially EAT + some HSM specific claims, in ASN.1.

History

- Crypto4A (HSM vendor) has already proposed an “attestation format” that is currently in use (presented at ICMC 2023).
- This effort modifies the original C4A work to effectively be a profile of RATS EAT.
- The design team meetings occurring every other Monday [1] are used to advance this effort as the membership between the two groups (LAMPS and RATS) has a large intersection. A good portion of the membership comes from Trusted Computing Group (TCG).
- RATS WG has already proposed a number of evidence claims in a different format (CBOR), but the HSM and CA community want an ASN.1 version.

[1]: email us if you want the meeting invite.

Design Considerations

- Format notation is ASN.1 with DER encoding
  - HSM and other cryptographic devices are expected to produce this information.
  - EAT has already defined many claims in CBOR, but many cryptographic end devices do not have the facilities to produce CBOR.
  - This approach favours re-use within other format (CSR, X.509), and lower-impact code changes within audited HSM kernels.
  - Can easily be re-used as an X.509v3 extension to carry evidence (with some heavy footnotes about privacy concerns).
- Extend EAT claims relevant for attesting HSM environments
  (ex.: FIPS mode, PKCS#11 private key protection properties, etc.)
  - Maybe we eventually register these to EAT as well
- Cryptographic requirements:
  - Mirror PKIX CMS crypto structure as much as possible.
  - Allow for multiple (dual) signatures.

References

- Git repo:
  - https://github.com/EntrustCorporation/draft-rats-pkix-evidence
- Explored design space:
  - 2023 LAMPS: draft-ounsworth-pkix-key-attestation
  - 2023 RATS: draft-ounsworth-rats-x509-evidence
- Current Draft:
  - https://datatracker.ietf.org/doc/draft-ietf-rats-eat
- Original work from Crypto4A:
  - https://support.crypto4a.com/public/documentation/C4A-302-0043-AttestationInQasm.html

Format - Top Level

```
PkixEvidenceStatement ::= SEQUENCE {
 tbsEvidence TBSEvidenceStatement
 signatureValues SEQUENCE SIZE (1..MAX) OF BIT STRING,
 relatedCertificates [0] IMPLICIT SEQUENCE of Certificate OPTIONAL
 -- As defined in RFC 5280
}
```

- Evidence statement is composed of a “to be signed” section
- Integrity and authenticity of the statement is ensured with one or multiple signatures
  - Multiple signatures help transition during PQC adoption
- Structure contains an optional supporting certificate chain, to help with validation.

Format - To Be Signed

```
TBSEvidenceStatement ::= SEQUENCE {
 version INTEGER,
 claims SEQUENCE SIZE (1..MAX) OF EVIDENCE-CLAIM,
 signatureInfos SEQUENCE SIZE (1..MAX) OF SignatureInfo
}
```

- The protected section where a sequence of “claims” is provided
- Version is provided as future proofing
- Signature information structures are provided to make outer signature “attached”
  - Departure from original C4A format; in debate

Format - Evidence Claim

```
EVIDENCE-CLAIM ::= TYPE-IDENTIFIER
-- TYPE-IDENTIFIER definition from X.681
TYPE-IDENTIFIER ::= CLASS
{
   &id OBJECT IDENTIFIER UNIQUE,
   &Type
}
WITH SYNTAX {&Type IDENTIFIED BY &id}
```

- Each claim is generic in nature
- Defined by an OID
- The nature and semantics of the Type are defined by the OID
- This provides extensibility to use cases not identified by the original design

Format - Proposed Initial Claims

- Mostly translated from EAT effort
- Proposed as starting point; format is extensible
- Claims related to platform generating the claims
  - Serial number, model, version
  - Unique identifiers, boot information and count, endorsements, state information (FIPS)
  - Current time, uptime
  - Measurements and nested evidences
  - Challenge
- Claims related to keys or objects
  - Key identifier
  - Capabilities, purposes and expiry
  - Exportable, locally generated

Proposed Claims

5.1. Platform Claims

Claim	OID	Value
Oemid	TBD	UTF8String
Hwmodel	TBD	UTF8String
Hwversion	TBD	UTF8String
Hwserial	TBD	UTF8String
Ueid	TBD	UTF8String
Sueid	TBD	UTF8String
EnvID	TBD	UTF8String
Swname	TBD	UTF8String
Swversion	TBD	UTF8String
Oemboot	TBD	BOOLEAN

(currently 23 Platform Claims)

5.2. Key Claims

Claim	OID	Value
KeyId	TBD	IA5String
PubKey	TBD	OCTET STRING
Purpose	TBD	CHOICE
NonExportable	TBD	BOOLEAN
Imported	TBD	BOOLEAN
KeyExpiry	TBD	Time

Some claims are direct ports from EAT; some are new.

Ex.: EAT Ueid is explicitly not human-readable, but HSMs have Serial Numbers on stickers on the case, so we need a “Hwserial” claim to tie the evidence to a physical device.

Claim definition

Example of porting a CBOR EAT claim to ASN.1:

```
The oemid claim is defined as follows:

   id-ce-evidence-oemid OBJECT IDENTIFIER ::=
         { id-ce TBD_evidence TBD_oemid }

   claim_oemid ::= SEQUENCE {
       type    INTEGER ( PEN(1), IEEE(2), RANDOM(3),...),
       value   OCTET STRING
   }
```

Current discussions

- Initial set of proposed claim types
- Concept of “multiple subjects”
  - Relates to whether an attestation should refer to a single instance or if multiple instances should be considered
  - Completeness vs complexity
- Concept of “signature infos” inside “to-be-signed” section
  - Should signatures be detachable?
  - Risk vs flexibility

- But generally, we think a signed “SEQUENCE OF CLAIM” is the right design.
- **Adoption from IETF RATS?**