key-attestation-exts

IETF 114 – Philadelphia – July 2022 – LAMPS Working Group

Sean Turner

Carl Wallace

Draft

- https://datatracker.ietf.org/doc/html/draft-wallace-lamps-key-attestation-ext-00
  - https://github.com/carl-wallace/draft-wallace-lamps-key-attestation-ext

Desired path forward

- Accept as working group draft and proceed on standards track

Why is a key attestation extension necessary?

- Keys may be generated using software or hardware mechanisms
- Many hardware cryptographic modules are capable of providing a verifiable key attestation that provide assurance that a key was generated by and is secured by hardware and is not exportable
  - May also provide some characteristics of the device, properties of the key, constraints on the key usage, etc. depending on the attestation type
- A CA may want to tailor certificate contents based on provenance of a key, i.e., assert certificate policy OID, key usage, etc. based on knowledge that key is secured by hardware
  - Or may elect to reject certificate requests lacking required characteristics

Why use WebAuthn attestation statement format?

- There are many different types of key attestations, so mechanism needs to be format agile
  - Prior proprietary work used different OIDs to identify formats
  - https://datatracker.ietf.org/doc/html/draft-bweeks-acme-device-attest-00 uses WebAuthn attestation format, with the $$attStmtType socket used to identify formats
- Elected the latter to simplify CA implementation given existence of ACME draft
- Question: is defining additional attestation statement formats that do not naturally occur in WebAuthn context OK or should we define a new registry?

Why PKCS #10, SCEP, CMC, CMP, CRMF, EST?

- All can be augmented using extensions or attributes (the draft uses same syntax and OID for both)
- ACME is already covered (at least for device certificates)

Things left to other specifications

- Nonces are discussed briefly in this draft, but details are necessarily left to key attestation format spec
- Likewise, any format-specific details or constraints (i.e., vendor name, device model, etc.) would be left to key attestation format spec and/or constraints spec