draft-ietf-suit-manifest-16

ietf 113

Brendan Moran

2022-03-24

Open Issues

- MTI Algorithms
- Crypto-agility

MTI Algorithms

- PQC MTI seems sensible, but what does that mean?
- What is required to support PQC?
  - Authors?
  - Bootloaders?
  - Update clients?
  - All of the above?
- At minimum, it seems that authors should be required to support PQC.

MTI implications

- HSS-LMS (W4 / H5) vs ECDSA (secp256r1)
  - 2/5 verification time
  - 74x signature size
  - 1.5x stack size
  - 2x Code Size
  - Signature size would be smaller with W=8 (20x) but verification time would be substantially higher (approx. 16x)
- Can we realistically require this from bootloaders?
  - Stack size should be irrelevant. Signature size, code size most relevant

https://eprint.iacr.org/2021/781.pdf

Alternatives?

- Falcon-512 (vs. ECDSA secp256r1)
  - No private key management overhead
  - 10.4x signature size (666 bytes)
  - 1/20 verification time
  - 4x verification RAM (4kB)
    - Round 3 comments suggest ram could be reduced to 2kB
  - 8.9x code size (57kB)
  - Not yet accepted by NIST

Crypto-agility

- Plausible for updatable update clients
  - Requires that hardware is built with the extra requirements in mind (2x size for crypto code, 1.5x stack size)
  - Over-specifying hardware may not be plausible
- Not plausible for non-updatable code (e.g. stage-0 bootloaders)
- What is the crypto-agility story for bootloaders?
  - Should we advocate dual-signature?
    - ECDSA/EDDSA for boot
    - HSS-LMS for updater