Attested TLS

draft-bft-rats-kat
draft-fossati-tls-attestation
draft-ftbs-rats-msg-wrap
Initial Attestation

• Bootloader creates a hash over the software (bootloaders, firmware, etc).
• Passes this software measurement up to the attestation service.
• This measurement will, if requested, be exposed in a PAT – Platform Attestation Token.
• PAT can be used to determine whether the software on the device has been modified.
Attestation in IoT Device Onboarding

• IoT device wants to onboard to a device management infrastructure.

• Setup:
  • IoT device (TLS Client) is the attester
  • Onboarding server (TLS Server) is the relying party.

• Assume: Background Check Model (i.e. relying party conveys evidence from the attester to the verifier to obtain attestation results).
• TLS client indicates what formats of attestation technologies it supports. ①

• TLS server selects what it wants to use (if anything). ②

It has to include a nonce (which it obtains from the verifier)
• TLS client creates Identity Key (IK) via the attestation service. Private key cannot be exported.

• TLS client requests KAT Bundle with
  - Nonce, and
  - IK-pub
  as input.
• TLS client conveys KAT Bundle in Certificate message to TLS server.
• TLS client uses IK priv to demonstrate possession of private key.

• TLS client transmits CertificateVerify to server.
• TLS server passes the received evidence to the verifier.

• TLS server receives IK pub and attestation result from verifier.
Status

- Confidential computing use case also described in the draft (but not presented today).
  - TLS server is attested rather than the client.

- Prototyping effort ongoing and supported by

- Passport model not (yet) described in the draft.
More Info

• Drafts:
  • https://datatracker.ietf.org/doc/draft-fossati-tls-attestation/
  • https://datatracker.ietf.org/doc/draft-ftbs-rats-msg-wrap/
  • https://datatracker.ietf.org/doc/draft-bft-rats-kat/

• Prototyping code:
  • Veraison: https://github.com/veraison/services/tree/ietf-115-hackathon
  • Parsec: https://github.com/ionut-arm/parsec-se-driver/tree/attested-tls/
  • TLS extension: https://github.com/hannestschofenig/mbedtls/tree/tls-attestation