Attestation within TLS

draft-fossati-tls-attestation-00

Thomas Fossati, Hannes Tschofenig, Paul Howard, Ionut Mihalcea, Yogesh Deshpande
Status

• Early experiment with prototyping to see how to integrate attestation into TLS generically.

• Design requirements:
  1. Works with background check and passport model (RATS).
  2. Supports attestation of client and server-side.
  3. Usable with different attestation formats (e.g. TPM-based, EAT-based),
  4. TEE agnostic (e.g. Arm v9, TrustZone with v8-A).
Technicality

• New extension is conceptually like certificate_types but with the ability to carry a nonce.
• The attestation information is carried in the Certificate message.
• Platform attestation are typically “bearer tokens”. Example: Arm PSA initial attestation token
• For TLS we need a key bound to a token (PoP token).
• To support existing hardware we cannot modify the platform attestation tokens hardware. → Introduce separate Key Attestation Token
Sample End-to-End Flow

Verifier

Normal World / NSPE

- Linux
- App with TLS Stack

Secure World / SPE

- Trusted Firmware A with OP-TEE
- Attestation Service

Device

Security Engine (e.g. TPM)

- Client Private + Attestation Key(s)
- Platform State
- Platform Attestation Token
- Key Attestation Token
Resources

• Initial draft available at https://datatracker.ietf.org/doc/html/draft-fossati-tls-attestation-00

• Proof-of-Concept in development. Uses Mbed TLS (TLS 1.3), Parsec and Veraison:
  • Mbed TLS: https://github.com/Mbed-TLS/mbedtls
  • Parsec: https://parsec.community/
  • Veraison: https://github.com/veraison