Handling Large Certificates and Long Certificate Chains in EAP-TLS

draft-ms-emu-eaptlscert-00

EMU IETF 102, Montreal, July 2018, John Mattsson

emu by Jon Bunting https://www.flickr.com/photos/84744710@N06/14766013011

DRAFT-MS-EMU-EAPTLSCERT-00

- **EMU WG Charter:** *”Gather experience regarding the use of large certificate and certificate chain sizes in the context of EAP-TLS (all versions), as some implementations and access networks may limit the number of EAP packet exchanges that can be handled. Document operational recommendations or other mitigation strategies to avoid issues.”*
- Agreed at IETF 101 that operational recommendations or other mitigation strategies should be documented in a new draft.
- New draft ”Handling Large Certificates and Long Certificate Chains in EAP-TLS” draft-ms-emu-eaptlscert-00 tries to accomplish this (submitted this week).
- **Is draft-ms-emu-eaptlscert-00 a good start? Something missing?**

PROBLEM STATEMENT

- Certificate chains can be quite large in size.
- This implies that EAP-TLS needs to be fragmented for transportation over lower-layers.
- Negatively affects legacy. Many authenticator (access point) implementations drops the EAP session if it hasn't finished after 40–50 packets.
- Results in failed authentication even when the two communicating parties have the correct credentials for mutual authentication.
- No mechanisms available to easily recover from such situations.

SOLUTIONS ALTERNATIVES

- **Use ECC cryptography**
  - Public keys: 384 bytes → 32 bytes
  - Signatures: 384 bytes → 64 bytes
  - Can be used with all TLS versions. TLS 1.3 requires implementations to support ECC.
- **Omit certificates the other endpoint is known to possess**
  - When using TLS 1.3, all certificates that specifies a trust anchor may be omitted.
  - When using TLS 1.2 or earlier, only the self-signed certificate that specifies the root certificate authority may be omitted.
- **Compress the certificate chain** [I-D.ietf-tls-certificate-compression]
  - Only possible with TLS 1.3

SOLUTIONS ALTERNATIVES

- **Cache the Server certificate** [RFC7924]
  - Omit transmission of server certificate chain obtained in earlier TLS handshake.
  - Can be used with all TLS versions.
  - The extension specifies that the server certificate chain can only be cached after a successful full handshake.
  - If the authenticator (access point) drops the EAP session after 40–50 packets, a successful full handshake might not be possible.
  - Could EAP-TLS allow temporary caching of a validated certificate chain even if the EAP-TLS exchange fails (this is currently not allowed according to RFC 7924)?
- **Anything else that should be described in the draft?**

WANTED

FEEDBACK