A TLS+CWT (v2) implementation in mbedTLS

A project proposal to the Attestation SIG

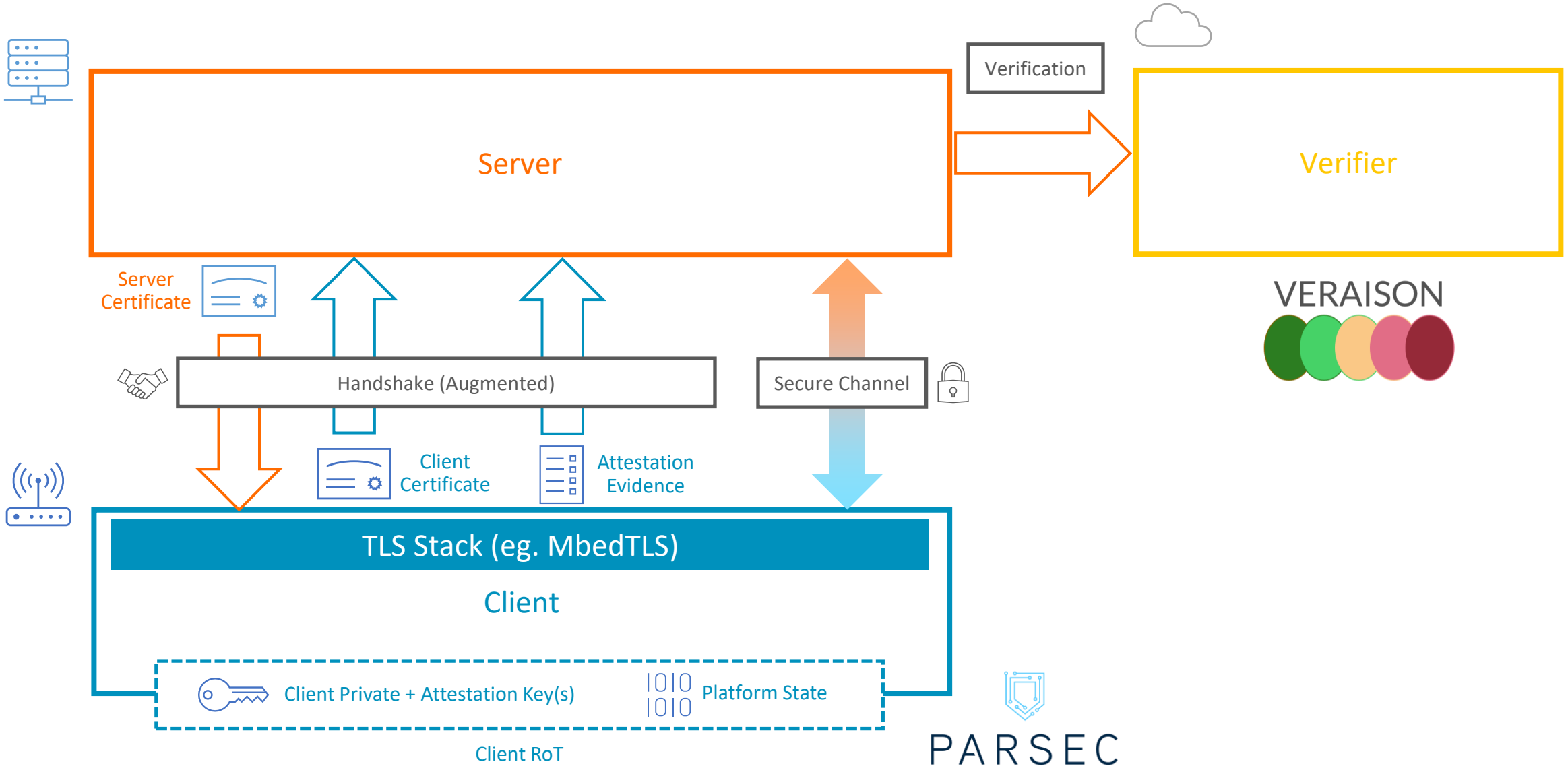
Context

- Hannes presented TLS+CWT to the SIG (slides)
- Brainstorming and prototyping ensued, which highlighted limitations in both the protocol and the existing APIs
- We sat down to sketch an improved version of TLS+CWT that would address the identified problems (explicit freshness indicators, attestation format agnosticism, support for different topologies)
- The (partial) result is in a new Internet Draft
  - The content is pretty rough because we wanted to hit the IETF submission cut-off date and be able to share the idea there

Plan

- Evolve the spec in parallel with the mbedTLS prototype
- Hook the TLS+CWT functionality into an end-to-end demo that includes
  - PARSEC as the "driver" to the attestation key holder and Veraison as the attestation verifier

Prototype Architecture



Project goals (1)

- Establish the protocol groundwork for using attestation-based credentials natively in TLS
  - Attestation evidence or results will be carried across instead of certificates
  - The resulting standard extension is envisioned as platform- and TEE-agnostic
  - The credentials will be treated as opaque blobs by the TLS implementation
  - *Fits under the CCC Attestation charter topic pursuing ease of exchange – utilising standard protocols – of confidentiality claims between parties, for which it also aims to deliver a design specification*

Project goals (2)

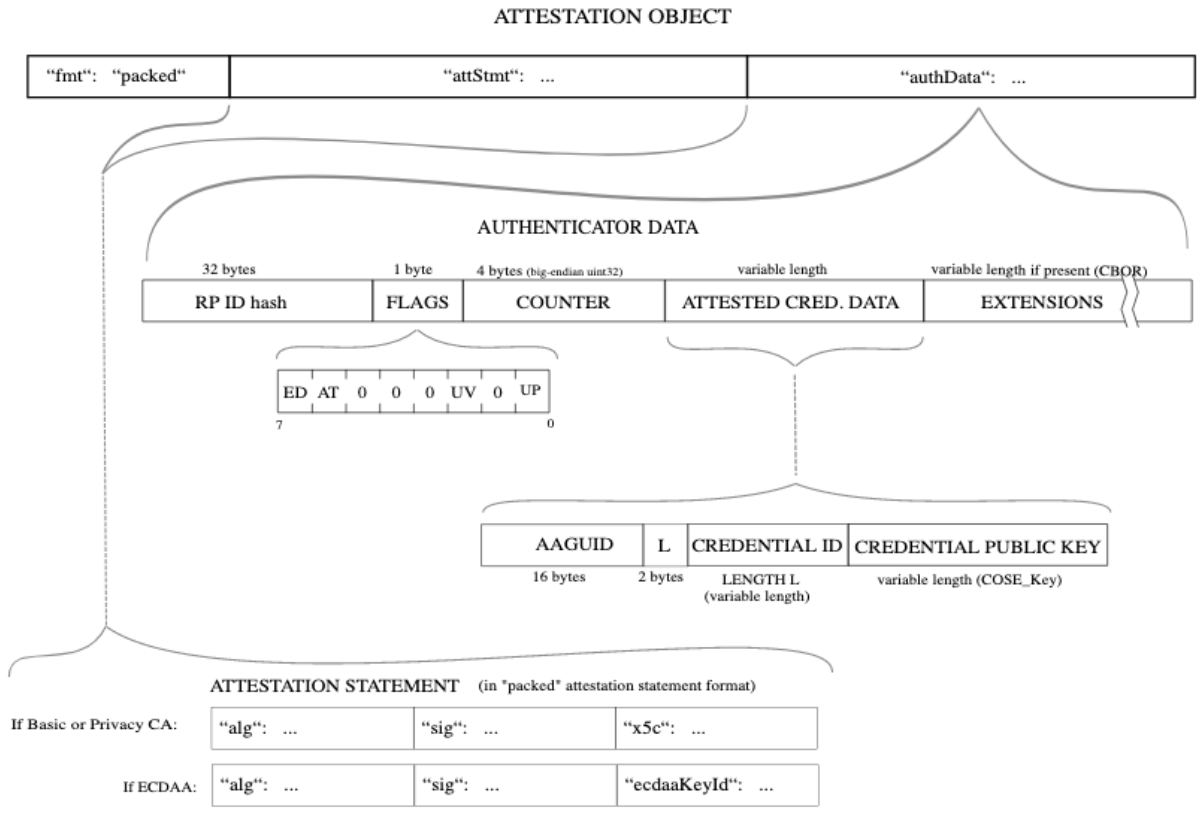
- Provide an end-to-end PoC using existing, widely-used/adopted open-source projects
  - MbedTLS as the base for the new TLS extension
  - Veraison for evidence verification
  - Parsec for interoperation with the RoT
  - *Advances CCC Attestation success criteria by providing a PoC based on existing tools, protocols, and formats*

Stretch goals (1)

- Sync'ing with similar efforts emerging around automatic certificate management
  - Recent drafts for extensions adding key attestation support in ACME and other certificate management protocols are dealing with similar issues
  - A concerted effort to standardize data formats would be beneficial
    - See next slide for more details about the format
  - Could ultimately help with homogenizing encapsulation of attestation-based credentials in x509 certificates
  - *Advances success criteria by attempting to plug gaps in (between) the existing standards and aiming for data-format interoperability*

Attestation formats

- Primarily based on a format defined under the WebAuthentication standard for conveying disjoint key attestation formats
- The attestation object and statement are designed for flexibility in terms of backend support, but aimed at usage within WebAuthn



Usage of WebAuthn statement format

- WebAuthn comes with several statement formats already defined (TPM, Android key attestation…)
- The ACME, LAMPS, and TLS+CWT++ drafts are now adopting and adapting these formats and the workflows around them for more generic use cases

Stretch goals (2)

- Aiding the design and implementation of similar extensions for other protocols (e.g., SSH)
  - Our technical documentation, design philosophy, and PoC could serve as a blue-print even in cases where the protocol is dissimilar
  - *It again ties back to the ease of exchange of confidentiality claims within standard protocols.*

Conclusion

- We’re hoping this direction of investigation and development is of interest to other stakeholders
  - More than happy to collaborate and align with other initiatives
- Also hoping to get this project adopted under the CCC Attestation SIG