TEEP Architecture Draft

draft-ietf-teep-architecture-01

IETF#103, Bangkok

Agenda

- Interim work and meeting update
- Document Status
  - Changes from the last version
- Issues Update

Interim Meetings

- TEEP Architecture authors and Chairs met three times since IETF 102
  - Two work session + 1 interim WG meetings
- Work drafts in GitHub
- Issues filed and tracked in GitHub
  - 18 issues
- Selected several issues to work on and consensus reflected in document update

Document Status

- David Wheeler joined author group as a co-editor
- v01 includes editorial changes and also resolution of a few issues from interim meetings
  - Overview / Terminology updates (Root of Trust, Trust Anchor etc.)
- Issues resolved

Trusted Firmware

- Agreement in the IETF 102 to make trusted firmware functionality optional since it is TrustZone-specific
- Document change has been made
  - Further clarified TFW key usage and relation to a boot loader, not mentioning controversial “secure boot”

#5: option to not use secure boot

- TFW and Secure Boot clarification
  - Agreement in the IETF 102 to make trusted firmware functionality optional since it is TrustZone-specific
- Document change has been made
  - Further clarified TFW key usage and relation to a boot loader, not mentioning controversial “secure boot”

#7: Clarify meaning of Security Domain

- Ming to complete
- Agreed to keep Security Domain concept
  - It is used to main a trust boundary for trusted application.
  - Multiple Trust Applications can belong to the same SD, which can have some access sharing per TEE’s implementation choice

#16: Terminology for “Agent”

- Ming
- Changed to “Broker” in v02

Root of Trust vs. Trust Anchor

- Attempt to differentiate the certificates usage with different terminology.
- David Wheeler proposed terminology for the two terms.
- Andrew proposed to remove trust anchor term and to use terms like “TAM root CA certificate store”.

Administrator Use Case

- To update

#29: Device Admin vs. Device Owner

- Ming / Dave
- See also Admin case

Others – To Add

OPEN ISSUES

#3: Trusted App Distribution

- Two modes:
  - TA binary bundled with the Client Application
  - TA distributed by TAM
- Challenges with first approach is
  - Passing device or TA instance specific data requires real-time interaction with a TAM. This functionality is in use today.
  - Client Application is not authorized to query TEE device state. Who is authorized to update a TA in the future? What would be the Security Domain?

#4: Algorithm Agility

- Ming

#6: Attestation Agility

- Agreed in principle. More to work on.

#8: Multiple TEEs vs. Single TEE

- Ming check notes (?)
- Impact on message routing
  - Multiple Broker use (To be added)

#9: Install TA in Single Pass?

- Flow update per Hackathon feedback (Ming / Dave)
  - TAM initial zero by GET call is necessary
  - Optimize to do this Single Pass for a device that has had cached TAM information
  -

#10: Local TEE signing first

- See #9

#11: Role of Client Application

- Hannes

#12: Every Rich App talks to TAM?

- Ming
- Metadata file and installer are good to use
- See also #9 and #10

#13: Is it in scope: TA depends on another TA?

- Dave / Ming
- Discussed in interim work sessions
- Yes, in scope and can be supported
  - Complex: very deep dependency concern
  - Circular dependency

#14: Multiple TAMs for single Client App?

- Hannes

Others?