TEEP Architecture
draft-ietf-teep-architecture-06

Dave Thaler (presenting)

Ming Pei, David Wheeler, Hannes Tschofenig

Timeline since IETF 106

NOV 20, 2019 – IETF 106

DEC 12, 2019 – Draft -05 posted

DEC 23, 2019 – WGLC started

JAN 20, 2020 – WGLC completed

FEB 8, 2020 – Draft -06 posted

FEB 10, 2020 ← TODAY

MAR 09, 2020 – I-D cutoff

MAR 21, 2020 – IETF 107 begins

WGLC Results

- 19 github issues filed
  - Ming filed issues on behalf of reviews that were posted only to the list
- 17 (plus some minor editorial nits) are addressed in draft -06
- Will walk through some of these in order from “not addressed yet” (needs discussion/consensus) to “hopefully done”

Issues not done in -06

#132: Requirements on Personalization Data

- Does *all* personalization data require confidentiality, or can there be device instance-specific data that only needs integrity?
- Current text (unchanged so far):
  - The personalization data **must** be encrypted to preserve the confidentiality of potentially sensitive data contained within it. **Other than this requirement to support confidentiality, TEEP place no limitations or requirements on the personalization data.**
- Hannes proposed (PR #101) replacing with:
  - The personalization data **may need to** be encrypted to preserve the confidentiality of potentially sensitive data contained within it.

#128: Path to TEEP for managment of connectivity, device managment, data managment subscription

- Garrett LoVerde filed:

  “euicc should prove useful to provide embedded designs with IoT device services management as long as the MCU can mutually authenticate with the euicc.

  Where does TEEP fit into this picture?”

- No change in draft so far.
- No response yet either.
- Would it be sufficient to just have someone answer on the list?

Issues addressed(?) in -06

#139: Contradiction about whether a device must have an REE to use TEEP

- Issue:
  - Text about TEEP Agent implied must have a TEEP Broker
  - Text about TEEP Broker says it runs in a REE
  - Intro said talked about "REE **(if present)**"
  - Can you have a TEEP Agent (and TEEP-over-HTTP transport) in a device that is entirely a TEE?
- Proposed answer: Yes
- Actions taken in PR 138 (done in -06):
  - Updated TEEP Agent to not imply there **must** be a TEEP Broker, just "typically"
- PR 145 includes clearer wording including:
  - "In devices with no REE, the TEEP Broker would be absent and instead the TEEP protocol transport would be implemented inside the TEE itself."

#113: Section 6.2.3 contradicts section 4.2 (1/2)

- When multiple TEEs in a device, there might be 1 common TEEP broker, or 1 per TEE, or anywhere in between
  - Previously different sections implied different answers
- Review of PR 131 also touches on how TAM selection is done
- Draft -06 is consistent with TEEP-over-HTTP transport spec:

  The selection of which TAM to communicate with might be made with or without input from an Untrusted Application, but is ultimately the decision of a TEEP Agent.

#113: Section 6.2.3 contradicts section 4.2 (2/2)

- Ming wrote:
  - Considering the case where a TEEP Broker will initiate a call to TAM for some cases, how about the following changes?

    "The selection of which TAM to communicate with might be made with or without input from an Untrusted Application. **When a TEEP Broker initiates a request to TAM, it may get such information from an Untrusted Application, the application installer in REE, or other input. When a TEEP Broker initiates a request to TEEP Agent first and then a TAM, the TAM choice may be chosen by the TEEP Agent**."
- Per current transport spec:
  - TEEP Broker only initiates a call to a TAM at the request of the TEEP Agent
  - TEEP Broker never simply uses the information from the Untrusted Application
  - TEEP Broker passes it to the TEEP Agent, which then optionally overrides it (or not), before telling the TEEP Broker to connect to the TAM

Security Considerations

- Added discussion about threats per Tiru's review:
  - #118: **compromised REE that terminates TEEP Broker**
  - #119: **a malicious agent acts as TEEP broker that launches attacks**
    - Does not prevent DoS attack by a compromised REE or Broker
  - #120: **TEEP Broker is instructed to repeatedly install/uninstall TA**
    - TEEP Agent might limit repeated requests to avoid bothering TAM
  - #123: **Add discussion about malicious TA removal**
    - TAM responsible for not installing malicious TAs
    - Includes discussion of "good TA goes bad" case, TAM can uninstall TAs gone bad
    - Attestation can be used to detect malicious TAs

#122: Clarify code signing certificate type and where it is validated

- Self-signed vs CA-issued certs:
  - Clarified that both are legal for any certs
  - Validation just checks to see whether a cert is or chains up to a cert in the relevant Trust Anchor Store
- Is Code signature checked at TAM?
  - TA might not be distributed via TAM
  - TA might be encrypted so that TAM can’t do anything with it
- TAM can just authenticate the entity uploading the TA to the TAM
  - Currently mechanisms to do so are out of scope
    - This is currently not even discussed, **should it be**?
  - “It's the responsibility of the TAM to not install malicious trusted apps in the first place.”

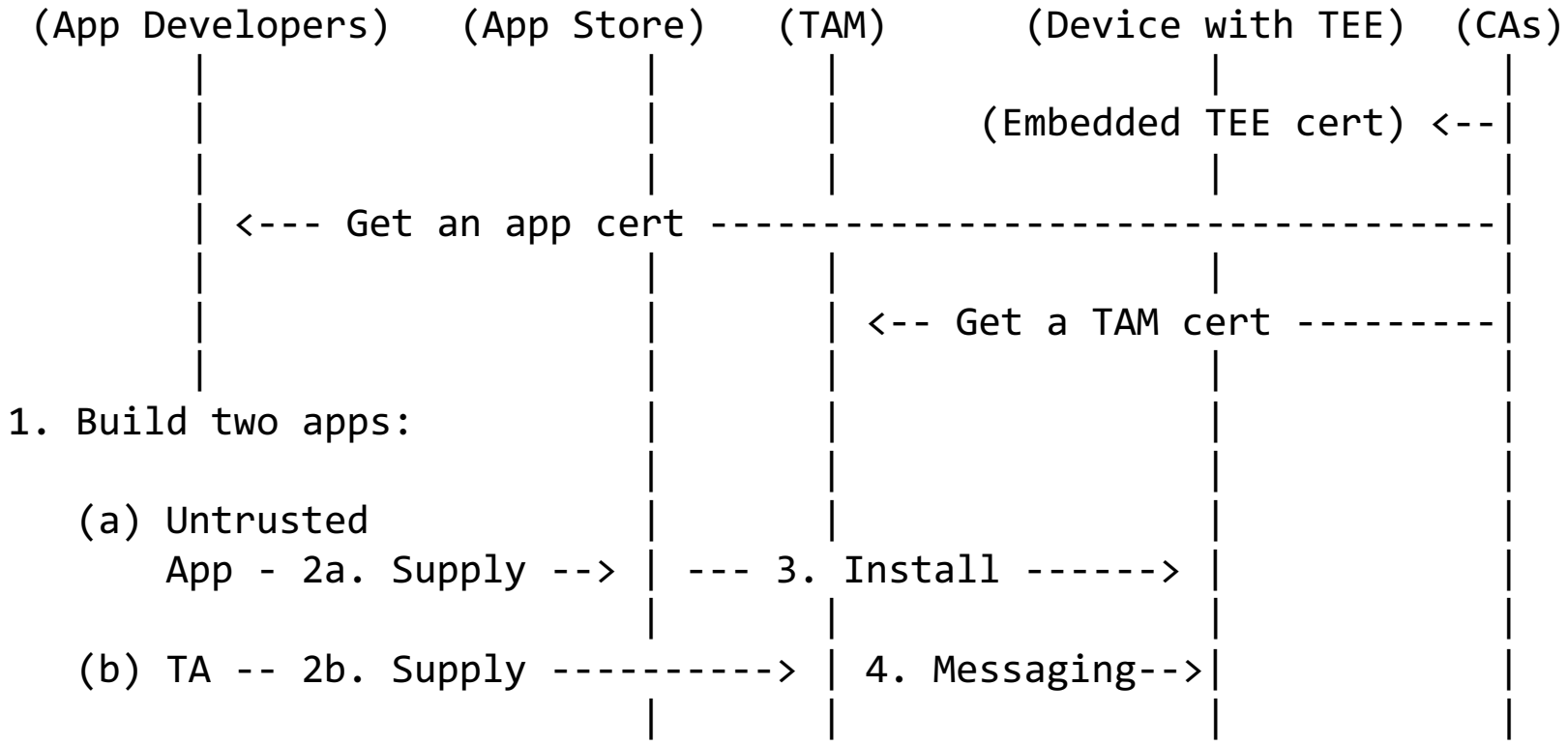
#121: Is this still valid - one TA by multiple SPs

- Tiru raised in his review, some list discussion ensued
- New text in -06:

  Each TA is digitally signed, protecting its integrity, and linking the TA back to the signer. The signer is usually the TA software author, but in some cases might be another party that the TA software author trusts, **or a party to whom the code has been licensed (in which case the same code might be signed by multiple licensees and distributed as if it were different TAs)**.

#114: Figure 3 is too hard to understand

- Rewrote diagram (which is just an ***example***) and discussion



#116: Incorrect/obsolete statements about TEEP Broker

- Issue:
  - Text was inconsistent with TEEP-over-HTTP WG consensus
  - Untrusted Application might not be involved in TEEP protocol exchange (e.g., might be done by installer)
  - TEEP Broker does not need access to TEE routing information in the content of TEEP messages
- Resolution:
  - Corrected text to be consistent with TEEP-over-HTTP transport discussion

Arguably just editorial

- #110: **Service Provider discussion is confusing**
  - Changed “service provider” to “TA developer” throughout doc
- #136: **List of problems solved is confusing**
  - Updated “service provider” scenarios to have corrected roles (per old TEEP WG slides on use cases)
  - Added the typical use case of Untrusted App install needs TA installed
- #112: **Text about carriers is confusing**
  - Carriers are now only used when explicitly mentioned as an *example*

Editorial-only

- Issues:
  - #127: **Definition of "Untrusted Application" under Terminology is confusing**
    - Nico suggested wordsmithing (accepted)
    - Separately suggested moving text down (not accepted since definition needed earlier)
  - #111 **Terminology section is not consistently ordered**
    - Reordered entries
  - #115: **Reference to IANA number**
    - Removed mention of IANA numbering space that was part of OTrP (and might appear in the TEEP protocol spec) not architectural
  - #129: **TEEP Agent introduced prematurely and insufficiently introduced in the definition of TAM under System Components**

Proposed Next steps

- Incorporate feedback from this meeting
- Post -07 before March 9 (ideally before end of Feb)
- Do a 2nd WGLC to finish before/at IETF 107?
- Goal is existing milestone:
  - “Apr 2020 - Progress Architecture document to the IESG for publication”