TEEP over HTTP

draft-ietf-teep-otrp-over-http-06

Dave Thaler <dthaler@microsoft.com>
Timeline

• NOV 2019 (IETF 106): got consensus on one remaining issue (#5)
  • “deal with #5 and we can proceed with WGLC”

• FEB 2020: Draft updated and WGLC started, ended Feb. 26
  • Two reviews received during WGLC (thanks Russ and Tiru!)

• APR 2020: Subsequent re-check by Mark Nottingham for conformance with bcp56bis

• APR 2020: Virtual interim, discussed WGLC results, and solicited additional reviews from Hannes and Ming
Issues raised since 1st WGLC initiated, then discussed at April interim:
8. TEEP Server must support all message formats in Single API?
10. TLS considerations
11. Update examples to use teep+cbor media type
12. TAM certificate caching
19. Why allow HTTP?
20. Use of HTTP error codes
21. bcp67bis as informative reference

Issues raised since April interim:
22. Redirect handling
23. Move section 3 (broker architecture) to architecture doc
24. Use of HTTP headers
15. Hannes’s other comments on draft 06
16. Ming’s comments on draft -06
Actions taken for past issues

• #21: bcp67bis as informative reference

OLD: When not called out explicitly in this document, all implementation recommendations in [I-D.ietf-httpbis-bcp56bis] apply to use of HTTP by TEEP.

NEW: For the motivation behind the HTTP recommendations in this document, see the discussion of HTTP as a transport in [I-D.ietf-httpbis-bcp56bis].

UNCHANGED: See Section 6 of [I-D.ietf-httpbis-bcp56bis] for additional discussion of HTTP(S) security considerations.
Actions taken for past issues

• #10: TLS considerations (Refer to IoT device TLS considerations RFC 7925)
• #19: Why allow HTTP?

• New text for above issues:
  • It is strongly RECOMMENDED that implementations use HTTPS. Although TEEP is protected end-to-end inside of HTTP, there is still value in using HTTPS for transport, since HTTPS can provide additional protections as discussed in Sections 4.4.2 and 6 of [I-D.ietf-httpbis-bcp56bis].
  • However, there may be constrained nodes where code space is an issue. [RFC7925] provides TLS profiles that can be used in many constrained nodes, but in rare cases the most constrained nodes might need to use HTTP without a TLS stack, relying on the end-to-end security provided by the TEEP protocol.
  • When HTTPS is used, TLS certificates MUST be checked according to [RFC2818], as well as [RFC6125] if PKIX certificates are used. See [BCP195] for additional TLS recommendations and [RFC7925] for TLS recommendations related to IoT devices.
#15: Hannes’s other comments on draft 06

1. Reference RFC 6125 for cert checking (DONE)
2. Remove TEEP/HTTP layer in docs?
   - No change since point is to explain relationship between docs (HTTP, this doc, and TEEP protocol doc)
3. Are we using Cookies? (I would say that we don’t. Currently not discussed.)
   - DONE: added “Cookies are not used.”
4. The note that the TEEP Agent can start with a QueryResponse if it has the TAM public key is IMHO incorrect
   - Was an optimization in OTrP to reduce RTT’s, open issue in TEEP protocol
   - Removed sentence since was informative in an example anyway
5. Be explicit about protocol end indication and 204 No Content
   - 2xx is required, but added “SHOULD be status 204 (No Content)”
#24: Use of HTTP headers (1/2)

1. Proposed adding (DONE, except last sentence since not for 204)
   • If the TAM does not receive the appropriate Content-Type and Accept header fields, the TAM SHOULD fail the request, returning a 406 (not acceptable) response. TAM responses MUST include a Content-Length header.

2. The text says that the client uses the Accept header but I don’t see any normative language there.
   • “sends an HTTP(S) POST to the TAM URI with an Accept header” implies normative
#24: Use of HTTP headers (2/2)

- Hannes: “Overkill” in X-Content-Type-Options, Content-Security-Policy, Referrer-Policy header recommendations

- Current SHOULD:
  - Cache-Control: no-store
  - X-Content-Type-Options: nosniff
  - Content-Security-Policy: default-src 'none'
  - Referrer-Policy: no-referrer

- Current text motivated by bcp67bis and MNot review, propose keeping

- Hannes proposes (added, since not a technical change):
  - The "Cache-control" header SHOULD be set to disable caching of any TEEP protocol messages by HTTP intermediaries. Otherwise, there is the risk of stale TEEP messages.
#22 Redirect handling

Hannes: Text says “Redirects MAY be automatically followed.” How should a developer decide whether it wants to follow the redirect?

Bcp56bis says:

As noted in [I-D.ietf-httpbis-semantics], a user agent is allowed to automatically follow a 3xx redirect that has a Location response header field, even if they don't understand the semantics of the specific status code. However, they aren't required to do so; therefore, if an application using HTTP desires redirects to be automatically followed, it needs to explicitly specify the circumstances when this is required.

Cases to think about, mentioned in bcp56bis:

- Proxy requires redirection
- Permanent change of server URI
#23: Move section 3 (broker architecture) to architecture doc

- Hannes proposed moving this section to the arch draft
  - Seems reasonable
- Ok with WG?
#16: Ming’s comments on draft -06

• Various editorial fixes (done!)
  • Notably: Clarified that a TEE is a SHOULD (not a MUST) on the TAM side

• Scope of TEEP protocol is to update “code and data in a TEE”
  • Ming suggested narrowing to updating (only) “TAs and data”
  • Dave: dependency on RATS attestation & SUIT manifests mean not just TAs but also their dependencies, which might include other TAs, trusted OS, and/or trusted firmware
    • This seems natural in both RATS and SUIT, and unnatural to preclude their use in TEEP
Next steps

• Address any feedback from this meeting
• Anything else before we’re done?