Protecting EST Payloads with OSCORE draft-ietf-ace-coap-est-oscore-06

Göran Selander, Ericsson
Shahid Raza, RISE
Martin Furuhed, Nexus
Mališa Vučinić, Inria
Timothy Claeys

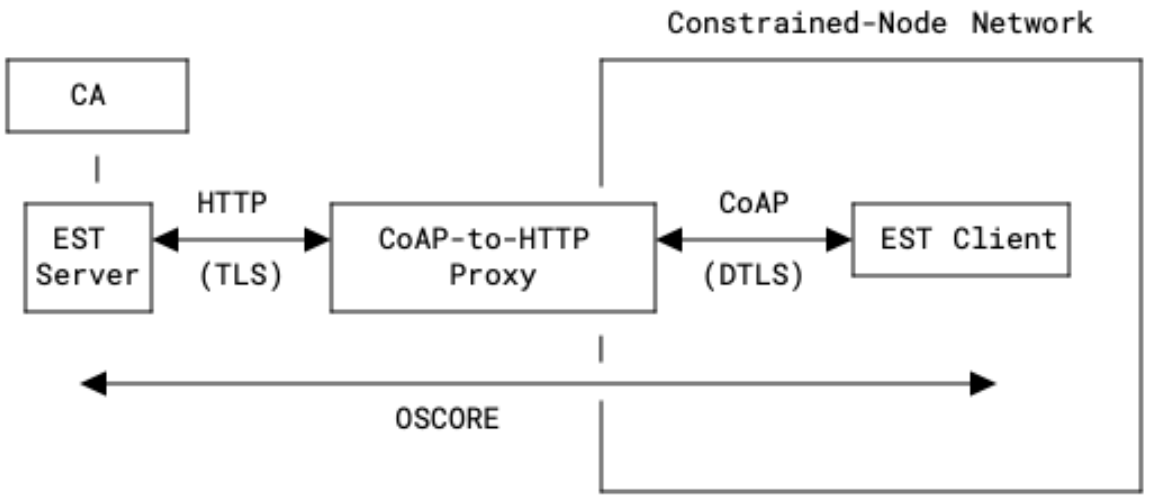
ACE @ IETF 121 – 8/11/2024

Status

- Published -06 on 21 October 2024
  - Received review on -05 by Esko Dijk -> Many thanks for the excellent review!
  - Fixes throughout the document based on Esko’s review
  - Review resulted in a total of 26 GitHub issues
  - 9 issues closed, the remaining to be discussed
- Goal of the presentation
  - Present the open issues and proposals to go forward

Open Issues

#76: Protocol support in HTTP-CoAP proxy section



- Many options for the EST-client to support
- We would need profiles of the spec to achieve interop
- Complicates the spec
- This is a spec about EST-oscore
- Proposal to simplify by removing support for DTLS at EST-client in the constrained part

#72: Consider future Content-Format identifier when responding to /skg

Section 4.3.2.

> In the case of CBOR-encoded request to /skg, the two parts of the response are also CBOR encoded. The certificate part is encoded as the application/cose-c509-cert object (Content-Format identifier TBD6), while the corresponding private key is encoded as application/cose-c509-privkey (Content-Format identifier TBD10). The function /skc is not available when using CBOR-encoded objects, and for server-side generated keys, clients MUST use the /skg function.

Esko's comment:

> There's also the client's Accept option that comes into play. There could be a future content-format that the client would specify in the Accept option; in which case the response could be another format than 62.
> But agree that when no Accept option is specified, then 62 is returned with CBOR-encoded contents.

Proposed action:

> Rewrite Section 4.3.2 in order to have analog paragraphs to Section 4.3.1 (ASN.1 objects) and particularly make sure that the text is compatible with the negotiation through the Accept option.

#65: Clarify how Explicit TA database should be populated

Section 3.2.

> EST-oscore, like EST-coaps, supports certificate-based authentication between the EST client and server. The client MUST be configured with an Implicit or Explicit Trust Anchor (TA) [RFC7030] database, enabling the client to authenticate the server. During the initial enrollment the client SHOULD populate its Explicit TA database and use it for subsequent authentications.

Esko's comment:

> There's a requirement on the client here, but what should the client do concretely? Are there some details needed on how to do this? And with whom are the "subsequent authentications" made, only with EST servers or with any servers/peers in the domain? (If it concerns EST server then it's in scope of this spec.)
> And what are the exception cases of the SHOULD ? It's also not so clear here how this requirement differs from RFC 9148 requirements on Explicit and Implicit TA database.

Proposed action:

> Reference the text from RFC 9148 on Implicit/Explicit TA database requirements and how they are populated (Section 9.1 of RFC 9148 and Section 9.2 of RFC 9148).

#59: Clarify the use of HTTP

Section 1 (Introduction)

> This document describes a method for protecting EST payloads over CoAP or HTTP with OSCORE [RFC8613]. OSCORE specifies an extension to CoAP which protects messages at the application layer and can be applied independently of how CoAP messages are transported. OSCORE can also be applied to CoAP-mappable HTTP which enables end-to-end security for mixed CoAP and HTTP transfer of application layer data. Hence EST payloads can be protected end-to-end independent of the underlying transport and through proxies translating between CoAP and HTTP.

Esko's comment:

> Paragraph 3 explains that EST can be carried over HTTP protected with OSCORE. This part could still be clarified: I initiallythought the intention was to enable HTTP transport without needing CoAP at all i.e. in none of the communication legs. This interpretation then conflicts with many following parts of the document that only mention CoAP and not HTTP used by the client. For example, paragraph 5 mentions "…context, the CoAP exchange carrying … " which then triggers the question whether this is really a CoAP exchange, or whether it can also bea HTTP exchange?
> Maybe the intention was to support only CoAP for the client side; which might be carried as HTTP for a particular network segment? (e.g. from proxy to EST server).
> Not sure how to resolve this; it depends on the intention of how HTTP is supposed to be used and this is not yet fully clearto me.

Proposed action:

> Remove mention of HTTP in the intro sentence. Add reference to Section 11 of RFC8613 (HTTP operation of OSCORE).

Next Steps

- Remaining issues are tracked and discussed in GitHub
- Next step is to resolve the open issues according to the current proposals

Thank you!