Key Management for OSCORE Groups in ACE

draft-ietf-ace-key-groupcomm-oscore-04

Marco Tiloca, RISE
Jiye Park, Universitaet Duisburg-Essen
Francesca Palombini, Ericsson

Interim, ACE WG, Jan 31, 2020

Updates since IETF 106

- Overall revision of the RESTful API with the Group Manager
  - Aligned with the latest updates in *ace-key-groupcomm*

- Size of ’cnonce’ and ’rsnonce’, as signature challenges to compute ’client_cred_verify’
  - Both are 8 bytes in size, as agreed at IETF 106

- Derived replacement for ’rsnonce’, for Access Token not posted to /authz-info
  - With reference to the DTLS profile, use a TLS exporter, as agreed at IETF 106

- Derived replacement for ’rsnonce’, for a client re-joining the group and relying on a still valid Access Token
  - If the client and the Group Manager use DTLS, use a TLS exporter
  - If the client and the Group Manager use OSCORE, use the output of a HKDF-Extract with:
    - IKM ← the OSCORE Master Secret
    - salt ← (ID Context || Sender ID of the client)

Updates since IETF 106

- Extended and revised profile requirements (Appendix A)
  - REQ17, REQ18, OPT5 and OPT6, as just added to *ace-key-groupcomm*

- Editorials and clarifications
  - New abstract, reflecting that this document is a groupcomm application profile of ACE
  - Content as general and of high-level applicability moved to *ace-key-groupcomm*
  - Addressed Peter's review: https://mailarchive.ietf.org/arch/msg/ace/jN-IUwl_skLG4tEIzj6roYoWY_Y

TODO left

- Open points (following slide)

- Align with next updates of *ace-key-groupcomm*

Open Points

1. When required, we use a TLS exporter to derive a signature challenge
   - We are using the same exporter label EXPORTER-ACE-Sign-Challenge from *ace-mqtt-tls-profile*
   - Register a separate exporter label for this profile, for context separation through different label values.

   Issues with that?

2. Ben Kaduk @ IETF 106: the 8-byte size of N_S and N_C should be justified (e.g. security considerations)
   - Same usage and size as in the MQTT profile. We should converge on same reasons.
   - RFC 8613 has that kind of considerations in Appendix B.2, i.e. probability of collisions vs. #messages
   - Action: same considerations here, for N_S and N_C.

   Issues with that?

3. A GET to /group-oscore/GROUPNAME/NODENAME and a GET to /group-oscore/GROUPNAME retrieve the exact content: ‘gkty’, ‘key’, ‘num’, [‘ace-groupcomm-profile’, ‘exp’, ‘mgt-key-material’]
   - The former should to return the group+individual keying material
   - The latter should return the group keying material, but ‘key’ includes the node’s Sender ID in ‘clientId’
   - Action: remove ‘clientId’ from ‘key’, for GET to /group-oscore/GROUPNAME

   Issues with that?