Work in progress towards

Key Provisioning for Group Communication using ACE
draft-ietf-ace-key-groupcomm-14

Francesca Palombini, Ericsson
Marco Tiloca, RISE

ACE WG Interim Meeting, September 14th, 2021

Since IETF 111

- Received two WGLC reviews – Thanks a lot!
  - Göran [1a] – Response at [1b]
  - Cigdem [2a] – Response at [2b]

- Comments organized into three groups
  - Editorial/nits
  - Clarifications
  - Design changes

[1a] https://mailarchive.ietf.org/arch/msg/ace/pr2gBhvqy9j8AfUdQVTZLwamXac/

[1b] https://mailarchive.ietf.org/arch/msg/ace/dEU04pB3u-iYNBwSlfjJaqkEvgo/

[2a] https://mailarchive.ietf.org/arch/msg/ace/gv_uRo2Y45jqOLJghVSbAARWky0/

[2b] https://mailarchive.ietf.org/arch/msg/ace/IL72zPmsIgF2j0Bgm7zO2fUTEm8/

Selected clarification requests (1/3)

- Related to group rekeying
  - Examples of additional administrative key material (e.g., in key-graph schemes like LKH)
  - Who decides it’s time to rekey the group? → Only the KDC
  - What reasons can trigger a group rekeying?
    - Change of group membership; regular refreshing; …
  - New dedicated section covering group rekeying, still at a high-level

- What can follow a PUT to ace-group/GROUPNAME/nodes/NODENAME ?
  - Just return new indidividual keying material ; or rekey the whole group ; or both

- Have a single boilerplate about common consistency checks for the KDC handlers

Selected clarification requests (2/3)

- Section restructuring, as a pair sequence (handler, example). Proposal in [1b]:

4. Keying Material Provisioning and Group Membership Management
  4.1 Overview of the Interface at the KDC

  4.2 ace-group
    4.2.1 FETCH handler
      4.2.1.1 Example <Content from current Section 4.2>

  4.3 ace-group/GROUPNAME
    4.3.1 POST handler
      4.3.1.1 Example <Content from current Section 4.3>
    4.3.2 GET handler
      4.3.1.1 Example <currently missing>

- Ok with this?

Selected clarification requests (3/3)

- Categorize message parameters into mandatory/conditional/optional to support
  - Think of a “miniminalistic” group member
  - A profile has also to categorize possible new parameters it introduces
  - Proposed classification of parameters in [1b] :
    - Always to support ; Conditionally to support ; Optional to support
  - Ok with this?

- Minimal set of operations to support
  - The KDC generally supports all of them
    - A profile can rule out parts of the KDC interface as “not provided”, if unneeded
  - For a group member, proposed classification in [1b] :
    - Always to support ; Optional to support
  - Ok with this?

Design changes (1/3)

- Some error responses from the KDC are enhanced and include an Error ID
  - Content format is *application/ace-groupcomm+cbor* and the payload is a CBOR map
  - A group member may just not understand specific Error IDs in ‘error’, and that’s fine
  - The additional and textual ‘error_description’ is already optional
  - Thinking of making this “more optional” or limited. Options in [1b] :
    1. Remove the parameter 'error_description' altogether.
    2. Make it optional for the KDC to use these enhanced error responses.
  - Thoughts?

Design changes (2/3)

- Recommended approach for one-to-one group rekeying – Proposal in [2b] :
  - The KDC should make /ace-group/GROUPNAME observable
  - If not planning to observe /ace-group/GROUPNAME , the joining node must specify 'control_uri' in the joining request, where the KDC can send individual requests
  - The KDC must support at least one push-based approach, minimally a point-to-point one. More efficient alternatives, e.g. based on multicast, remain possible (see next slide)
  - For point-to-point rekeying, notifications and/or requests are used, based on the above
  - Ok with this?

- General improvements to group rekeying
  - When rekeying due a member's joining, rekeying messages can include the public key of the new group member. We can rely on the existing 'pub_keys' parameter. Objections?
  - Define <u>a new dedicated parameter</u> (better than a group policy value) for the Joining Response, indicating the group key management scheme. If absent, a default point-to-point scheme to be defined by the application profile is assumed. Objections?

Design changes (3/3)

- One-to-many group rekeying, e.g. through multicast, for better scalability
  - Possible and considered in the past; we need additions to fully enable it. Proposal in [1b] :
    - Define a new 'mgt_group_uri' parameter in the Joining Response, specifying a “base URI”, with the multicast IP address where the KDC sends multicast control message (e.g., due to rekey)
    - This assumes and requires that 'control_uri' is also provided by a joining group member.
    - Actual resources to target can have full uri IP_ADDR:PORT/ace-group/GROUPNAME/something , where *something* is pre-defined (e.g., "rekeying“) and reflects the exact management operation
  - Ok with this?

- The above requires source authentication of one-to-many rekeying messages
  - Need for the KDC’s public key; *key-groupcomm-oscore* already defines its provisioning
  - Move the general provisioning definition here?

- Provide **high-level** guidelines on the protection of these messages
  - Likely possible only at the application level, using the additional administrative key material
  - Details can be left to application profiles to specify. Ok with this?

Open points

- REQ16 deals with KDC policies related to former group members, see [2b]
  - A possible policy is about retaining public keys of former members, for a certain amount of time
  - Cigdem: *I think this is a wider policy e.g., how long does the KDC retain any information about the historical group members?*
  - Marco: … *you'd like a policy … to explicitly define also how the retention time is determined, possibly on a per-node basis.* *Correct?*

- Group rekeing through a pub-sub broker [2b] – Might become a separate thread
  - Cigdem: *This is not a good scenario for pub-sub, as the broker should not know the keys. …* [it] *becomes a recursive problem …*
  - Marco: *This … is not referring exactly to the pub-sub profile of ACE to do that … .*
    - *For … rekeying the main security group, the KDC is a publisher and all the group members are subscribers of a "rekeying topic".*
    - … [rekeying messages] *would be protected by the KDC at the application level, using additional administrative key material shared between the KDC and the members of the main security group*.
    - *Actually, I believe the pub-sub profile of ACE may assist for this case too.*

Next steps

- Address the WGLC reviews (ongoing)

- More to clarify
  - Scope: intermediate specification to build application profiles for group communication
  - Key assumption: trust relation between KDC and (candidate) group members
  - Further protocol-specific security considerations are for the application profiles

- Submit version -14 before the cut-off

Thank you!