Key Provisioning for Group Communication using ACE

draft-ietf-ace-key-groupcomm-07

Francesca Palombini, Ericsson
Marco Tiloca, RISE

ACE WG, Interim, June 22, 2020

Quick Recap

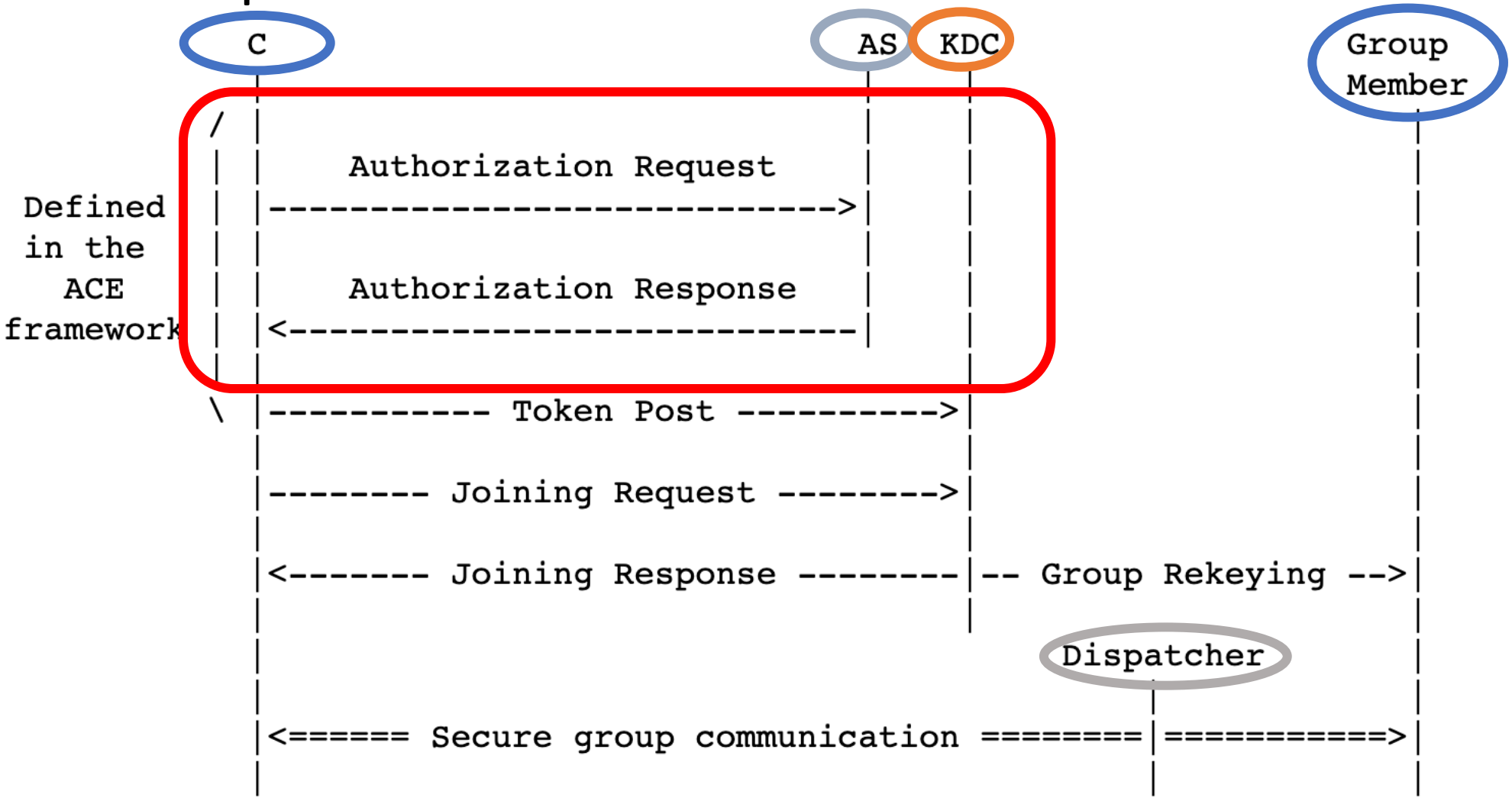

Figure 2: Message Flow Upon New Node's Joining

What happened since Interim 18-05-2020 – status update

Version -07 was submitted June 17th 2020 based on Daniel's review and leftover points discussed during last interim:

- Jim's question 6 - Keeping the same key identifier for groups → agreed during last interim that it was resolved with recent updates.
- Jim's question 5 - Congestion control needs to be included → Added some considerations about that: f0d159
- Jim review -04 - 4 open points:
  - Normalize scope → Added informative reference to aif: 8d116c *
  - MUST NOT not testable → removed in 7025c9
  - Add "nodes" to Uri-path of resource: /ace-group/GROUPNAME/nodes/NODE → was done in previous updates
  - Request previous keying material → agreed during last interim that we did not want to add this functionality
- Jim's question 1 - When does a KDC need to roll the keys over → Added security considerations: 7e1deb
- Peter review -03 → editorial except for one point specific to ace-key-groupcomm-oscore. Almost* everything implemented: 7fda36

* Follow up points discussed today

Daniel's review -06

- Daniel review -06 → answered and implemented in 7450ee and 9a3b2d:

Main change: registered the new ace parameters to the "OAuth Parameters Registry" + "OAuth Parameters CBOR Mappings Registry" instead of "AS Request Creation Hints"

o Parameter name: sign_info
o Parameter usage location: token request, token response
o Change Controller: IESG
o Specification Document(s): [[This specification]]

o Parameter name: pub_key_enc
o Parameter usage location: token request, token response
o Change Controller: IESG
o Specification Document(s): [[This specification]]

o Parameter name: kdcchallenge
o Parameter usage location: token response
o Change Controller: IESG
o Specification Document(s): [[This specification]]

This is consistent with the new Ace parameters in OSCORE profile: nonce1 and nonce2

Normalized scope

Example of scope:

```
gid = bstr
role = tstr
scope_entry = [ gid , ? ( role / [ 2*role ] ) ]
scope = << [ + scope_entry ] >>
```

How to use aif for Group OSCORE communication?

- Gid = h’01’
- Group resource = /topic-01
- Nodes can be requester, responder, monitor, requester+responder

Normalized scope

Roles:

- Requester: sends group requests and accept group responses
- Responder: accept group requests and sends group responses
- Requester+Responder: sends and accepts both
- Monitor: accepts group requests and does not reply

Let's say "role" of "scope-entry" is formatted with aif:

Problems:

- **aif only specifies policies for request methods, while roles also refers to responses.**
- **AS does not necessarily know the group resource.**

gid = h'01'
group resource uri-path = /topic-01

[h'01' , "requester"]
=
[h'01', ["/topic-01", 63]] (63 = all methods)

[h'01' , "responder"]
=
[h'01', ["/topic-01", 0]] (0 = no method)

[h'01' , ["responder" , "requester"]]
=
[h'01', [["/topic-01", 0] , ["/topic-01", 63]]

[h'01' , "monitor"]
=
[h'01', ["/topic-01", ???]]

Terminology

- Peter wanted us to define new terminology:
  - ”management channel” = Client – KDC
  - “group channel” = Client – other group members

- We don’t think this would help a lot in the document, and would prefer not to define new terminology.
- If there are sections where it is not clear between what parties the exchange is, we will rephrase.

Plan forward

- WGLC?