

The Group Object Security for Constrained RESTful Environments (Group OSCORE) Profile of the Authentication and Authorization for Constrained Environments (ACE) Framework

draft-ietf-ace-group-oscore-profile-07

Marco Tiloca, RISE
Rikard Höglund, RISE
Francesca Palombini, Ericsson

IETF 126 meeting – Vienna – July 20th, 2026

Recap

› Group OSCORE profile of ACE

- Enable access control for accessing resources at group members
- Group OSCORE [1] is the security protocol used between C and the RSs
- C and the RSs have to first separately join the group as defined in [2]
- An access token is bound to the already existing Group OSCORE Security Context and to the authentication credential AUTH_CRED_C that C uses in the group

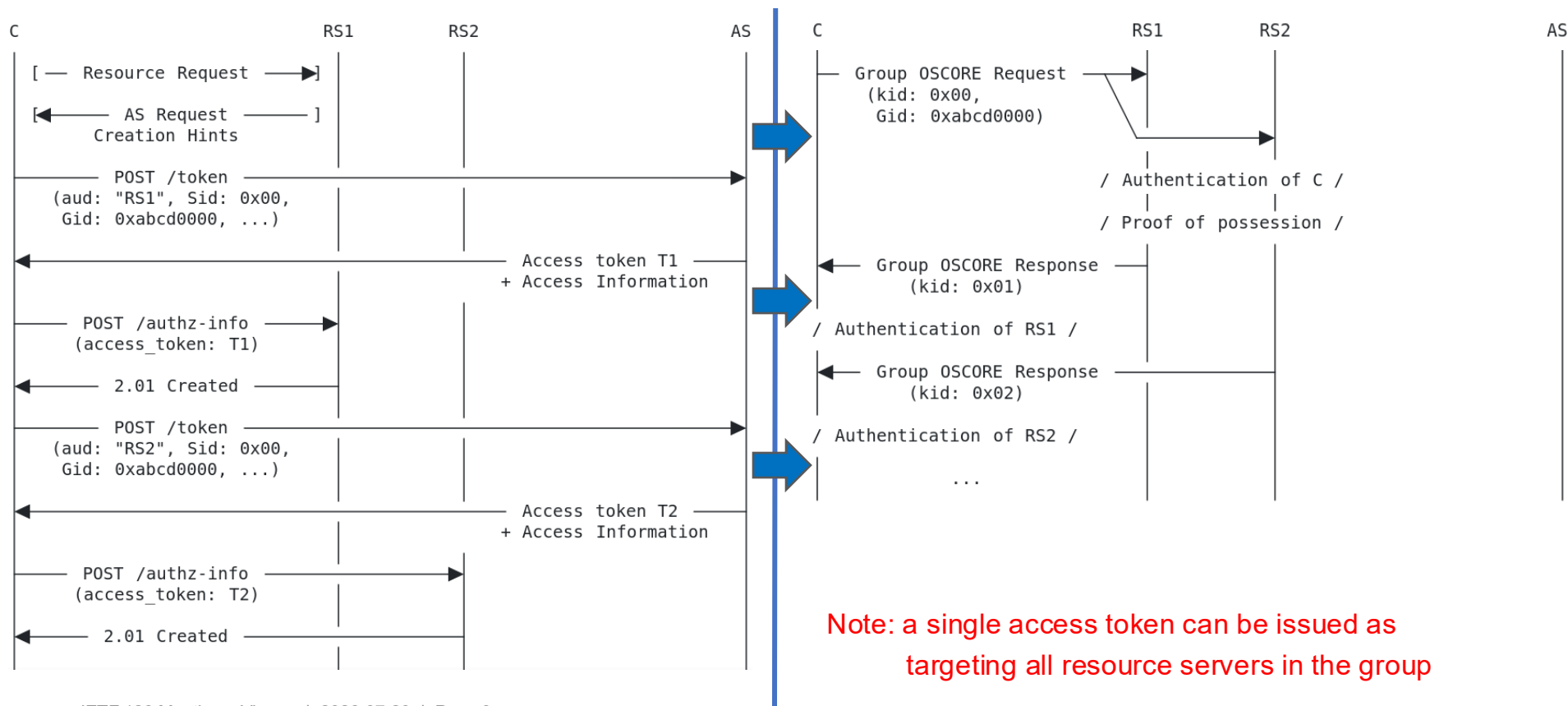
› Properties

- Proof of possession of C's private key
 - › Achieved by the RS when verifying a first Group OSCORE request from C
 - › Both the group mode and pairwise mode of Group OSCORE are covered
- The RS achieves proof of group membership for the exact client C
- Mutual authentication, when completing a first Group OSCORE exchange

[1] <https://datatracker.ietf.org/doc/draft-ietf-core-oscore-groupcomm/>

[2] <https://datatracker.ietf.org/doc/draft-ietf-ace-key-groupcomm-oscore/>

Overview - protocol flow



Latest developments

› Previous presentation for version -03

- At IETF 121, November 2024

› Since then, both dependency drafts [1][2] have been approved for publication

› Updates from -03 to -04

- ‘cnf’ claim in the access token: authentication credential of C perfectly aligned with what C provided to the Group Manager (e.g., certificate chain)
- Proof-of-possession (PoP) evidence from C in the C-to-AS access token request
 - › Made optional to include: the AS might have achieved proof of possession of C’s private key before (e.g., in previous interactions)
- Suggested values for registered codepoints; alignments with other drafts; editorial fixes

[1] <https://datatracker.ietf.org/doc/draft-ietf-core-oscore-groupcomm/>

[2] <https://datatracker.ietf.org/doc/draft-ietf-ace-key-groupcomm-oscore/>

Latest developments

› Updates from -04 to -05

- Clarifications about:
 - › Optional inclusion of the PoP evidence in the C-to-AS access token request
 - › Keeping data associations locally up-to-date at the RS

› Updates from -05 to -06

- Section 8 “Guidelines on Using Multiple Profiles”
 - › Point to the revised semantics of the 'ace_profile' parameter in the C-to-AS access token request, as defined in [3]
- Added registrations to the "JSON Web Token Claims" IANA registry
- Minor clarifications

Updates from -06 to -07

› **Minor**

- Editorial improvement and fixes
- Added references to IANA registries

› **Section 2 – Extended and clarified early protocol overview**

› **The Group Manager (GM) at [2] is RECOMMENDED to be used**

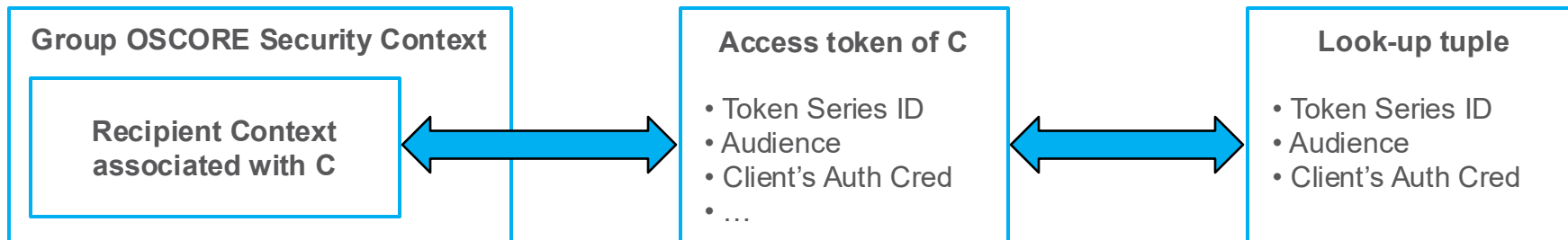
- Still, the wording about operations at the GM is agnostic of the specific realization of GM

› **Detailed clarifications about the RS:**

- On processing the received access token
- It can join the OSCORE group latest when receiving the access token
- On achieving proof of group membership for C through the GM
- On achieving proof of possession for C, after receiving the first Group OSCORE request from C

Updates from -06 to -07

- › The RS binds the access token with the Recipient Context associated with C
 - Not to the whole Group OSCORE Security Context
- › At the RS: simplification in binding the access token to other information
 - OLD: access token $\Leftrightarrow$ (Group ID, Client's Sender ID, Client's AuthCred, GM's AuthCred)
 - › No need to consider and keep up-to-date (Group ID, Client's Sender ID, GM's AuthCred)
 - › It would be redundant, due to the bound Recipient Context and its Security Context
 - NEW: access token $\Leftrightarrow$ (Token Series ID, Audience, Client's AuthCred)
 - › Supporting the dynamic update of access rights (see next slide)



Updates from -06 to -07

› Defined dynamic updates of access rights through token series

- Like in other profiles, a new access token can extend an ongoing token series
 - › Same audience, C's auth. credential, and (to-be-)bound Group OSCORE Security Context
- In this profile, access tokens are unrelated to establishing the Group OSCORE Security Context

› New Section 3.3 “Token Series”

- Specialized definition of token series introduced in [3]
- Allocation criteria for token series ID at the AS → Unique for each pair (Client, Audience)
- All access tokens in the same token series include the same token series ID

› Workflow: Sections 3.2, 3.4, 3.4.1, 3.4.2, and 4.2

- In the AS-to-C response issuing the first access token T1 : T1, “token_series_id” = ID_X, ...
- In the C-to-AS request asking for a follow-up access token T2 : “token_series_id” = ID_X, ...
- In the AS-to-C response issuing a follow-up access token T2 : T2, ...
- At the RS, T2 supersedes T1 paired with the same (Token Series ID, Audience, Client's AuthCred)

Next Steps

- › **Error handling at the AS**

- › Add the use of the ACE error code "failed_pop_verification" defined in [3]

- › **Elaborate about resource servers that are “monitors” (“silent servers”)**

- › By definition, they cannot reply with responses protected with Group OSCORE

- › **Consider setups with multiple application groups and security groups**

- A client might need an access token spanning multiple application/security groups
 - Currently, the access token can only target one security group

- › **Comments and reviews are welcome!**

Thank you!

Comments/questions?

<https://github.com/ace-wg/ace-group-oscore-profile>

Backup

Applicability and features

› For application scenarios relying on group communication

- A client wants to access resources at one or more resource servers in a group
- Secure communication by using a shared set of keying material
- Aims to enforce access control within the group, for resources at servers

› Separation between group membership and access control

- Being a legitimate group member does not naturally imply access rights
- The following two concepts are separate:
 - › Access to the secure group communication channel (through group membership)
 - › Access control to the resource space provided by servers in the group - This document

› Follow the Zero-Trust paradigm [3]

- Focus on resource protection
- Trust is never granted implicitly, but must be continually evaluated
- Access control enforcement must be as granular as possible

Detailed message exchange (1/2)

› The C-to-AS access token request also includes:

- ‘context_id’: **Group ID** (‘kid_context’) of the OSCORE group
- ‘salt_input’: client’s **Sender ID** (‘kid’) in the OSCORE group
- ‘client_cred_verify’: client’s **proof-of-possession evidence**
- ‘req_cnf’: **client’s authentication credential** in the OSCORE group

› Proof-of-possession evidence in ‘client_cred_verify’

- Computed with the client’s private key used in the OSCORE group

```
Header: POST (Code=0.02)
Uri-Host: "as.example.com"
Uri-Path: "token"
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / audience /      5 : "tempSensorInLivingRoom",
  / scope /         9 : "read",
  e'context_id_param' : h'abcd0000',
  e'salt_input_param' : h'00',
  e'client_cred_verify' : h'c5a6...f100' / elided for brevity /,
  / req_cnf /       4 : {
    e'kccs' : {
      / sub / 2 : "42-50-31-FF-EF-37-32-39",
      / cnf / 8 : {
        / COSE_Key / 1 : {
          / kty / 1 : 2 / EC2 /,
          / crv / -1 : 1 / P-256 /,
          / x / -2 : h'd7cc072de2205bdc1537a543d53c60a6
                    acb62eccd890c7fa27c9e354089bbe13',
          / y / -3 : h'f95e1d4b851a2cc80fff87d8e23f22af
                    b725d535e515d020731e79a3b4e47120'
        }
      }
    }
  }
}
```

C-to-AS access token request

› Input to compute the proof-of-possession (PoP) evidence

- If **(D)TLS** is used between C and AS ==> an exporter value (Sect. 4 of RFC 5705 / Sect. 7.5 of RFC 8446)
- If **OSCORE** is used between C and AS ==> PRK = HMAC-Hash(x1 | x2, IKM)
 - › x1 = ID Context of the C-AS OSCORE Security Context;
 - › x2 = Sender ID of C in the C-AS OSCORE Security Context;
 - › IKM = OSCORE Master Secret of the C-AS OSCORE Security Context

Detailed message exchange (2/2)

› The AS-to-C access token response also includes:

- ‘token_series_id’ : **Series ID** identifier of the token series, only when issuing the first access token in the series

› The access token also includes:

- ‘token_series_id’ : **Series ID** identifier of the token series
- ‘context_id’ : **Group ID** of the OSCORE group (*)
- ‘salt_input’: client’s **Sender ID** in the OSCORE group (*)
- ‘cnf’: **client’s authentication credential** in the OSCORE Group (*) Only if the first access token in the series

› Token POST to the RS and response from the RS

- RS checks **C’s auth. credential** with the Group Manager
- RS stores the access token and associates it with the tuple (**Series ID**; Audience; **Client’s authentication credential**)
- Another group member cannot impersonate C

Header: Created (Code=2.01)
Content-Format: 19 (application/ace+cbor)
Payload:

```
{  
  / access_token /      1 : h'8343a1010aa2044c...00'  
                        / elided for brevity /,  
  / ace_profile /      38 : e'coap_group_oscore',  
  / expires_in /      2 : 3600,  
  e'token_series_id_param' : h'3a'  
}
```

AS-to-C access token response

```
{  
  / aud /      3 : "tempSensorInLivingRoom",  
  / iat /      6 : 1719820800,  
  / exp /      4 : 2035353600,  
  / scope /    9 : "temperature_g_firmware_p",  
  / ace_profile / 38 : e'coap_group_oscore',  
  e'token_series_id_claim' : h'3a'  
  e'context_id_claim' : h'abcd0000',  
  e'salt_input_claim' : h'00',  
  / cnf /      8 : {  
    e'kccs' : {  
      / sub / 2 : "42-50-31-FF-EF-37-32-39",  
      / cnf / 8 : {  
        / COSE_Key / 1 : {  
          / kty / 1 : 2 / EC2 /,  
          / crv / -1 : 1 / P-256 /,  
          / x / -2 : h'd7cc072de2205bdc1537a543d53c60a6  
                    acb62eccd890c7fa27c9e354089bbe13',  
          / y / -3 : h'f95e1d4b851a2cc80fff87d8e23f22af  
                    b725d535e515d020731e79a3b4e47120'  
        }  
      }  
    }  
  }  
}
```

Access token