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-04

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

IETF 122 meeting – Bangkok – March 19th, 2025

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 RSs
  - The group joining must separately happen first, 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 of the client C
- **Properties**
  - Proof-of-Possession of the client's private key
    - Achieved when verifying a first Group OSCORE request from the client
    - 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/

Updates in v -04

- **Require that 'cnf' in the access token includes exactly what C uploaded to the Group Manager**
  - C might have uploaded AUTH_CRED_C to the Group Manager as provided within a chain or a bag
  - The inner value of the 'req_cnf' parameter in the access token request must specify AUTH_CRED_C as provided within the same chain or bag.
  - … and this same value is used in the 'cnf' claim in the access token.
  - When the RS retrieves the client's credential CRED from the Group Manager, the RS must verify that the credential is contained in the same chain or bag as was the case in 'cnf'

```
{
  / aud /           3 : "tempSensorInLivingRoom",
  / iat /           6 : 1719820800,
  / exp /           4 : 2035353600,
  / scope /         9 : "temperature_g firmware_p",
  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'
        }
      }
    }
  }
}
```

Updates in v -04

- **Improved example for guidelines on using multiple profiles**
  - An RS can be registered at an AS with multiple audiences
  - … the specific audience can be associated with a specific ACE profile such as the OSCORE profile or the Group OSCORE profile
  - This example has now been improved to use more realistic values for the audiences

- **Placeholder content for the client using the 'ace_profile' parameter to signal the desired profile**
  - The client could include the 'ace_profile' parameter in the access token Request to explicitly signal the desired profile to the AS (extended semantics defined in [1])
  - This complements or can replace the point above about usage of multiple profiles
  - Only placeholder content for now, to be extended with further details

[1] draft-ietf-ace-workflow-and-params

Updates in v -04

- **PoP evidence in the access token request is now optional**
  - Extended discussion on this point in IETF Dublin
  - Conclusion: Inclusion of the PoP evidence in every access token request is too restrictive
  - Why? The AS may have other means of performing PoP, or may have done so already
    - i.e., via a trusted third party or via the secure channel established with the client
  - **Solution**
    - When the AS successfully verifies a PoP evidence it marks the corresponding credential for that client as 'confirmed', and as 'non confirmed' otherwise
    - If the AS performs PoP through other means, the credential is also marked as 'confirmed'
    - If the client does not include a PoP evidence in the access token request and its credential is already confirmed the request is valid, if the credential is not confirmed the request is invalid

Updates in v -04

- **IANA considerations**
  - Now indicating value ranges for codepoints to register for new parameters defined in the draft
- **Editorial improvements and fixes**
  - Aligning terminology better with the ACE framework document
- **Aligned CBOR abbreviations to those used in other documents**
  - Now aligned with what is specified in *draft-ietf-ace-edhoc-oscore-profile*
  - i.e., specifying value 11 for 'kccs'

Next Steps

- **Add considerations about group rekeying**
  - Group rekeying can occur between the Access Token Request and the access token upload, …
  - … with consequent change of Group Identifier in that group

- **Enable dynamic update of access rights**
  - Follow the roadmap defined in Section 3.3.1
  - Using new parameters defined in *draft-ietf-ace-workflow-and-params*

- **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
  - This would mean that some parameters would become multi-value

- **Comments and reviews are welcome!**

Thank you!

Comments/questions?

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

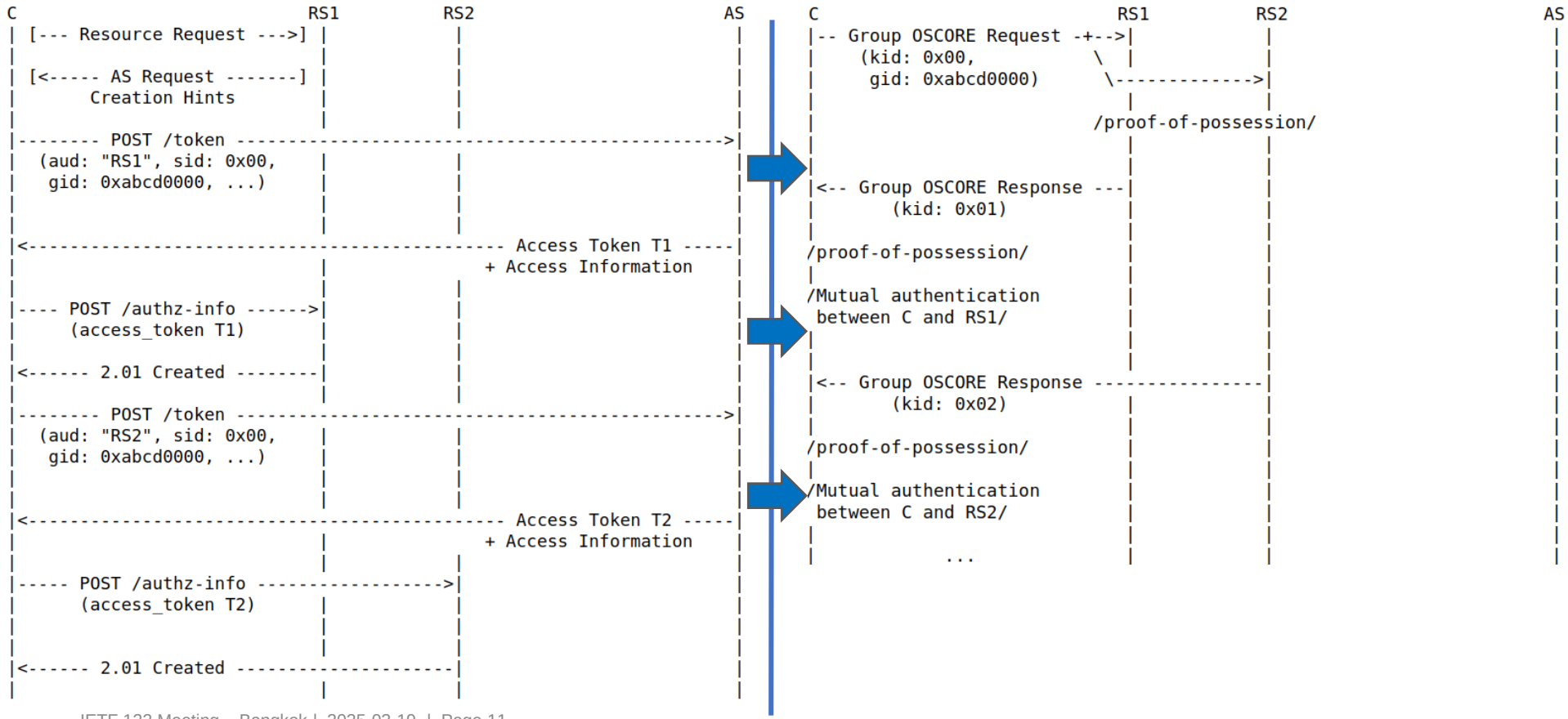
Backup

Applicability and Features

- **For application scenarios relying on group communication**
  - A client wants to access resources at multiple resource servers
  - 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 membership)
    - access control to the resource space provided by servers in the group - This draft
- **Follows 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

[3] NIST-800-207: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf

Overview - Protocol flow



Detailed message exchange (1/2)

- **The C-to-AS Access Token Request includes also:**
  - 'context_id': Group ID ('kid_context') of the OSCORE group
  - 'salt_input': Client Sender ID ('kid') in the OSCORE group
  - 'client_cred_verify': Client's proof-of-possession evidence
  - 'req_cnf': Client's auth. 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 : "tempSensor4711",
  / 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'
        }
      }
    }
  }
}
```

Access Token Request

- **What is the input to compute the proof-of-possession (PoP) evidence?**
  - If **(D)TLS** is used between C and AS ==> an exporter value (Section 7.5 of RFC 8446)
  - If **OSCORE** is used between C and AS ==> PRK = HMAC-Hash(x1 | x2, IKM)
    - x1 = Context ID 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)

- **Nothing special in the AS-to-C Access Token Response**

```
Header: Created (Code=2.01)
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / access_token / 1 : h'8343a1010aa2044c...00',
  / ace_profile / 38 : e'coap_group_oscore',
  / expires_in /   2 : 3600
}
```

Access Token Response

- **The Access Token includes also:**
  - ‘contextId_input’ : Group ID of the OSCORE group
  - ‘salt_input’: Client Sender ID in the OSCORE group
  - ‘cnf’: Client’s auth. credential in the OSCORE Group

- **Token POST to the RS and response from the RS**
  - RS checks C’s auth. credential with the Group Manager (GM)
  - RS stores the access token and associates it with the quartet (Group ID; Sender ID; C’s auth. Credential; GM’s auth. Credential)
  - Another group member cannot impersonate C

```
{
  / aud /           3 : "tempSensorInLivingRoom",
  / iat /           6 : 1719820800,
  / exp /           4 : 2035353600,
  / scope /         9 : "temperature_g firmware_p",
  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