Short Distribution Chain (SDC) Workflow and New OAuth Parameters for the Authentication and Authorization for Constrained Environments (ACE) Framework

draft-ietf-ace-workflow-and-params-04

Marco Tiloca, RISE
Göran Selander, Ericsson

IETF 122 Meeting – Bangkok – March 19th, 2025

Recap

Set of updates to RFC 9200 (and more)

- **Define a new workflow for uploading the access token**
  - The AS uploads the access token to the RS, on behalf of C
  - Preferable if the C-RS communication leg is constrained, while the AS-RS leg is not
- **Define additional OAuth parameters to use in ACE**
  - Enabling the new workflow, also with dynamic update of access rights
  - Effectively enabling the issue of an access token for a group-audience
- **Extended semantics of the “ace_profile” parameter** ***NEW***
- **C and the AS can coordinate on authentication credentials by value or reference** ***NEW***
- **Amend two requirements on transport profiles of ACE**
- **Deprecate the original payload format of error responses**
  - Instead, use the problem-details format from RFC 9290

Updates in v -04

- **Suggestion from Dave Robin: give a name to the new workflow**
  - "Short Distribution Chain (SDC) Workflow"
  - The access token can go directly from the producer (AS) to the consumer (RS)
  - Hence the new draft title: "Short Distribution Chain (SDC) Workflow and ..."

- **Comment from Dave Robin, about the new workflow:**
  - C might not be an actual ACE client ➔ C might be "clueless" of details like access tokens
  - New text in Section 2: "*The SDC workflow is also suitable for deployments where clients are not aware of details such as the need for access tokens to be issued by the AS and uploaded at the RS.*"

- **Suggestion from Dave Robin: remove the content on bidirectional access rights**
  - Better to have it in a separate document, to be separately discussed
  - Moved out to https://datatracker.ietf.org/doc/draft-tiloca-ace-bidi-access-control/

Updates in v -04

- **Section 1.1 – More precise definition of "token series"**
  - *"A set of access tokens … **bound to the same proof-of-possession (PoP) key** and … sequentially issued by the same AS for the same pair (client, audience) **per the same profile of ACE**. …"*

- **Section 3.6 – Value of a token series identifier, for the "token_series_id" parameter**
  - The AS assigns a new identifier in a simpler and more efficient way
  - **(TS_ID, C, AUD)** MUST uniquely identify a token series and its corresponding access tokens
    - TS_ID: identifier of the token series
    - C: client to which the access token is issued
    - AUD: audience for which the access token is issued
  - AS must consider both ongoing and ended token series for selecting a new TS_ID
  - The triple used to be (TS_ID, ***AUTH_CRED_C***, AUD)
    - More complex to correctly enforce at the AS
    - To avoid ambiguities, AUTH_CRED_C had to be in access token requests for updating access rights

Updates in v -04

- **Section 4 – Extended semantics of the “ace_profile” parameter**
  - In the C-to-AS access token request
  - Suggested by Christian Amsüss – Thanks!
    - https://mailarchive.ietf.org/arch/msg/ace/eC0mQN8Jy1DoHH9lyib5zxC5MJM/

- **Original use defined in RFC 9200**
  - If “ace_profile” = null is included in the access token request …
  - Then “ace_profile” is included in the access token response, indicating the profile used

- **New additional use**
  - If “ace_profile” = int is included in the access token request …
  - Then the issued access token is issued per the profile specified in the request
  - The AS returns an error if the requested profile is not supported by the AS or by the targeted RS

Updates in v -04

Section 5 – Coordination between C and AS on authentication credentials

- **Some profiles of ACE use public authentication credentials**
  - C-to-AS request: C provides its authentication credential AUTH_CRED_C to AS
  - AS-to-C response: AS provides the authentication credential AUTH_CRED_RS of RS to C
- **Those credentials can be transported by value or identified by reference, e.g., in:**
  - https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/
  - https://datatracker.ietf.org/doc/draft-ietf-ace-authcred-dtls-profile/
- **Possible issues**
  1. The AS forgets AUTH_CRED_C, but C specifies it by reference in the C-to-AS request
  2. C forgets AUTH_CRED_RS, but the AS specifies it by reference in the AS-to-C response

Updates in v -04

Section 5 – Coordination between C and AS on authentication credentials

- **Solution to (1) – Defined new ACE error code "unknown_credential_referenced"**
  - The AS replies to C with an error response, including the new error code
  - C sends a new access token request, specifying AUTH_CRED_C by value

- **Solution to (2) – Extended semantics of the "rs_cnf" parameter → Update to RFC 9201**
  - C specifies "rs_cnf" in the access token request to the AS, with possible values:
    a) true (0xf5): C wants that the access token response specifies AUTH_CRED_RS by value
    b) false (0xf4): C wants that the access token response specifies AUTH_CRED_RS by reference
    c) null (0xf6): C wants that the access token response does not specify AUTH_CRED_RS at all
  - The AS complies with the above, or returns an error if unable to
  - (a) addresses the actual issue; (b)(c) are optimizations for when C knows better than the AS

Next steps

- **Add more examples of message exchanges**
  - Using the new error code “unknown_credential_referenced”
  - Using the “rs_cnf” parameter in the access token request
- **Define a new ACE error code for the AS**
  - “failed_pop_verification”, for an error response in reply to an access token request
  - For the AS to use, when it does not achieve/confirm proof-of-possession of C's private key
- **Guidelines for using the “anchor_cnf” parameter with group-audiences**
- **On the new ACE workflow**
  - Detailed handling of an uploaded access token on the RS side
  - New parameter “updated_rights”, to prevent ambiguities when dynamically updating access rights (*)
- **Comments and reviews are welcome!**

(*) Appendix B provides a high-level plan for addressing this point

Thank you!

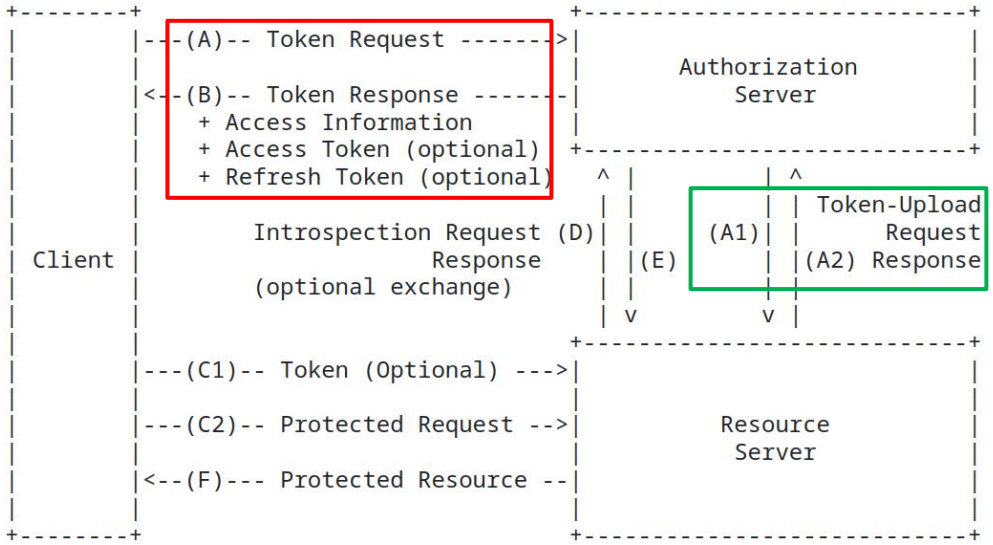
Comments/questions?

https://github.com/ace-wg/ace-workflow-and-params

Backup

New SDC workflow

- **(A) C-to-AS Token Request as usual**
  - C explicitly opts-in for the new workflow, by including the new parameter “token_upload”
  - The final choice about using it is on the AS
- **(A1) The AS uploads the access token to RS, on behalf of C**
  - No intention to replace the original workflow
  - The AS can dynamically choose the workflow to use, e.g., based on the specific RS



- **(A2) The AS receives a response from RS**
- **(B) AS-to-C Token Response**
  - New parameter “token_upload”, with value 0 (successful upload) or 1 (failed upload)
  - **0** → The Response includes: the access token; or a token hash; or neither. Then, C skips step C1.
  - **1** → The Response includes the access token. Then, C performs step C1.

Examples with new SDC workflow

```
Access Token Response

Header: Created (Code=2.01)
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  e'token_upload' : 0,
  / expires_in / 2 : 3600,
  / cnf /        8 : {
    / COSE_Key / 1 : {
      / kty / 1 : 4 / Symmetric /,
      / kid / 2 : h'3d027833fc6267ce',
      / k /  -1 : h'73657373696f6e6b6579'
    }
  }
}
```

Example 1: the AS successfully uploaded the access token


```
Access Token Response

Header: Created (Code=2.01)
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
    e'token_upload' : 1,
  / access_token / 1 : h'd08343a1...4819',
    / (full CWT elided for brevity;
       CWT contains the symmetric PoP key in the "cnf" claim) /
  / expires_in /   2 : 3600,
  / cnf /          8 : {
    / COSE_Key / 1 : {
      / kty / 1 : 4 / Symmetric /,
      / kid / 2 : h'3d027833fc6267ce',
      / k /  -1 : h'73657373696f6e6b6579'
    }
  }
}
```

Example 2: the AS attempted to upload the access token but failed