interim-2020-oauth-06 : oauth
Monday Apr-27-2020 1200

1. The OAuth 2.1 Authorization Framework
https://datatracker.ietf.org/doc/draft-parecki-oauth-v2-1/

2. JWT Response for OAuth Token Introspection
https://datatracker.ietf.org/doc/draft-ietf-oauth-jwt-introspection-response/

Slides and content can be found at
https://datatracker.ietf.org/meeting/interim-2020-oauth-06/session/oauth

Webex Link
https://ietf.webex.com/ietf/j.php?MTID=mec2af7ee3fbb8c161501b6294c762114

Participants (33)

- Aaron Parecki
- Anthony Nadalin
- Artem Snisarenko
- BHupinder
- Brian Campbell
- Brock Allen
- Cristofer Gonzales
- Dick Hardt
- Daniel Fett
- Dmunson
- Dominick Baier
- Filip Skokan
- Francesca Palombini
- Francis Pouatcha (adorsys)
- George Fietcher
- Janak Amarasena
- Jim Schaad
- Justin Richer
- Mark Russell
- Micah Silverman
- Michael Peck

- Peter Yee
- Mike Jones
- Tim Cappalli
- Torsten Lodderstedt
- Sebastian Ebling
- Matt de Haast

OAuth 2.1 Authorization Framework

Aaron Parecki

OAuth 2.1 Summary

- Consolidation of:
  - Native Apps BCP, PKCE, Browser-Based Apps BCP, Security BCP, Bearer Tokens
- Grant types defined: Authorization Code with PKCE, Client Credentials
- Exact redirect URI matching
- No Bearer tokens in query strings
- Refresh tokens must be sender-constrained or one-time use
- Implicit and password grants are omitted

Changes Since - 01

- Added HTTP 307 redirect section from Security BCP
- Editorial and typo fixes
- Updated references to other specs such as HTTP

Open Questions

1. **Should we reference Dynamic Client Registration and AS metadata as option methods for registration and discovery respectively?**
   a. Justin R: Suggest that the document at least reference the other related documents
      i. Brian, Torsten +1
2. **Should TLS be required for redirect URIs?**

    a. Except for localhost and non-HTTP redirect URIs
    b. Torsten: OpenID Connect already requires TLS. (RFC6749)
3. **Confidential Clients**: “What is the intended definition of confidential clients”?
    a. **Dick H**: The intent “Could the client keep a secret.” and a Dynamic client could keep a secret. This could be confusing.
    b. **George**: Agreed with Dick; The original intent, can you or cannot keep a secret. For example: if using PKI, then the client can keep a secret.
    c. **Justin**: Notes that there is a difference between “At Configuration” and “At Runtime”. That this speaks to a public client that isn’t able to keep a secret at runtime.
    d. **Torsten**: Agreed with Justin: We need to distinguish the type of client. Can distinguish the client and what policy we can assign to the client.
    e. **Mike J**: Recommended that distinctions would be good if they could be added to this draft to help developers.
4. **What’s Next**
    a. **Dick H**: There has been confusion about the contents of the document and we might use open discussions to clean the contents and resolve the confusion.
    b. **Mike J**: OAuth has been extended by several other groups and other extensions, OpenID Connect is an example.
        i. **Mike** would like to see what is defined in the registry as OAuth response types to be recognized in this document
        ii. **Dick**: Would like to see IANNA references and the various registries.
        iii. **Aaron P**: There is wording in the document noting that other extensions do extend the framework.
        iv. **Torsten**:Would ask the group to ask “what do they want to achieve with 2.1”. For example, BCP provided new guidelines. In contrast, 2.1 just combines existing documents. We must be careful to not create confusion or weaken the language of other documents.
        v. **Justin R**: The goal of this document should create a new baseline.
5. **Mike J**: Would like the draft to state that many of the extensions are valid and to provide an example
6. **Vittorio B**: Sender constraint language for specific endpoints wasn’t very clear. Hopefully we can clear that up in this draft.
    a. Francis P: refresh tokens are always sender constrained.

b. The risk is diminished from a native, desktop app than a browser application.

JWT Response for OAuth Token Introspection

Overview

1. Introduces an additional response mode for “OAuth 2.0 Token Introspection” (RFC 7662)
2. The response is carried as signed (and optionally encrypted) JWT
3. Allows for non-repudiation
4. Allows application-level security

Recent Changes

1. Moved the data of the introspection token into the top-level JWT claim “token_introspection”
2. Allows separation of the carrier JWT claims from the actual token introspection response claims, for the example “iat”.

What’s Next

1. Submit to WG review before submitting to publication
    a. **Roman**: +1
    b. **Justin R**: This change addresses several reservations that he expressed.
    c. **George F**: Is the expectation that additional behavior be included in the JWT? Also, is it possible to get double ‘jti’?
        i. Does this allow to JWT’s to be cached for a specified time and a particular use?
    d. **Filip S**: +1 the idea. Additional questions and comments.
    e. **Justin R**: (RFC 7762) does speak about caching and how long the claim should be cached. We maybe should at least mention or reference (RFC 7762)
    f. **Aaron P**: Question, with addition of this wrapper or container object, is possible to use this same JWT or Token with a completely different vocabulary.

i. **Torsten**: Yes, the AS could include other tokens or claims.
ii. **Filip S**: These claims are included in the introspection spec and how this should be handled.