IETF 115
London
November 2022

Aaron Parecki
Dick Hardt
Torsten Lodderstedt

OAuth 2.1

https://datatracker.ietf.org/doc/draft-ietf-oauth-v2-1/

draft -07

Changes from IETF 114 Discussion

- Removed "third party" from abstract
- Added MFA and passwordless as additional motivations in introduction
- Mention PAR as one way redirect URI registration can happen
- Added a reference to requiring CORS headers on the token endpoint
- Updated reference to OMAP extension
- Fixed numbering in sequence diagram

More Changes Since IETF 113 Vienna

- Fixed some references (Thanks Falko)
- Updated HTTP references to RFC 9110, removes unused refs (Thanks Roberto)
- Updated reference for `application/x-www-form-urlencoded` to WHATWG URL
- #29 Clarified “authorization grant”
- #27 Clarified client credential grant
- #55 Cleaned up authorization code diagram

Diff: https://github.com/oauth-wg/oauth-v2-1/compare/draft-05...draft-ietf-oauth-v2-1-07

Issues: https://github.com/oauth-wg/oauth-v2-1/milestone/4?closed=1

Planned Changes for -08

- #70 Finish incorporating feedback from Justin and Vittorio (Security considerations, native apps)
- #64 Finish moving normative language from security considerations inline in the doc
- #97 Expand the differences from OAuth 2.0 to include for which roles each change is a breaking change

Still more open issues to discuss!

https://github.com/aaronpk/oauth-v2-1/issues

Discussion Topics

CORS Recommendations (#133)

We recently added a recommendation to support CORS on the token endpoint.

In practice, several other endpoints (mostly defined in extensions) also need to support CORS.

Additionally, the authorization endpoint does NOT need to support CORS, and Bad Things™ can happen if it does.

This suggests we need a broader CORS recommendation strategy in the spec.

(OpenID Connect also had this discussion recently)

CORS Recommendations (#133)

MUST NOT allow CORS requests at the authorization endpoint

To support JS apps, CORS must be supported at the token endpoint, and any other endpoints used by JS apps:

- AS metadata
- Revocation
- Dynamic client registration
- Other .well-known URLs?

Add suggestions here!

redirect_uri at token endpoint (#54)

Should redirect_uri required at the token endpoint?

- Serves no technical purpose with PKCE
- But we want backwards compatibility too
- Should we:
  - Not document it at all?
  - Make accepting (and verifying it if present) mandatory by the AS, but optional to send by the client?
  - Something else?
  - Does this actually simplify anything or does it just add new problems?

IETF 115
London
November 2022

Aaron Parecki
David Waite

OAuth 2.0 for Browser-Based Apps

(Best Current Practice)

https://datatracker.ietf.org/doc/draft-ietf-oauth-browser-based-apps/

draft -11

OAuth 2.0 for Browser Based Apps

- Includes recommendations for implementers building browser-based apps using OAuth 2.0
- "Browser-based apps" are defined as applications executing in a browser, aka "SPA" or "single-page apps", and may include a backend component

Changes from -09 to -11

Revised the names of the architectural patterns:

- Single-domain
- Backend-for-frontend proxy
- Token-mediating backend
- Token acquisition in the browser

Changes from -09 to -11

- Added more considerations when storing tokens in LocalStorage
- New patterns:
  - Token acquisition and management in a Service Worker (thanks Yannick!)
  - Token-mediating backend (adapted from Brian Campbell and Vittorio’s TMI-BFF draft)

Planned Changes

- #2 Add a section with recommendations and considerations for storing tokens
  - e.g. storing tokens in memory vs in LocalStorage
- #6 Review recent changes to the Security BCP to ensure this draft is consistent

Almost done…?

CORS recommendations?

Security BCP could recommend that the AS *not* add CORS headers.

Browser Apps BCP could recommend the specific endpoints that *do* need CORS headers in order to be able to be used by JS.

(OAuth 2.1 would consolidate these two into the draft.)