draft-ietf-oauth-security-topics

Workgroup:	Web Authorization Protocol			
Internet-Draft:	draft-ietf-oauth-security-topics-16			
Published:	5 October 2020			
Intended Status:	Best Current Practice			
Expires:	8 April 2021			
Authors:	T. Lodderstedt *yes.com*	J. Bradley *Yubico*	A. Labunets	D. Fett *yes.com*

OAuth 2.0 Security Best Current Practice

Virtual Interim Meeting, Oct. 26, 2020

OAuth 2.0 Security Best Current Practice

- Describe the best current security practice for OAuth 2.0
- Update and extend the OAuth 2.0 Security Threat Model
- Incorporate experience from practice and research
- Cover new threats relevant to OAuth 2.0, in particular in high-risk environments like banking, eID

Status: First WGLC end of last year on version -13 (now -16).

What’s new since -13?

Prevent PKCE Downgrade Attacks

Attack: An attacker uses a stolen, non-PKCE bound code and injects it into a flow where PKCE is used.

New recommendation: AS MUST ensure that if there was no `code_challenge` in the authorization request, a request to the token endpoint containing a `code_verifier` is rejected.

Changes re PKCE & nonce

- **New:** PKCE is now a MUST for public clients
- **Unchanged:** for confidential clients, PKCE is RECOMMENDED.
  Nonce MAY be used with additional precautions.

Other Important Changes since WGLC

- Improved wording around implicit grant
- Allow variable port numbers in localhost redirect URIs (cf. RFC8252)
- Text on XSS (undermining token replay protection) and Clickjacking attacks
- mTLS is now a suggested method for token replay protection, no longer the only RECOMMENDED one
- Tightened discussions on potential solutions
- Improved examples
- Various editorial improvements

What’s left to discuss?

Not much!

Proposal: Recommend Use of Metadata

- For AS, publishing OAuth Metadata (RFC8414) is already RECOMMENDED (alternative: deployment-specific way to communicate PKCE support)
- For clients, using OAuth Metadata is not yet recommended.

Proposal: Make the use of OAuth Metadata for discovery RECOMMENDED.

Goal: Promote (security) automation, reduce chances for mistakes.

- Avoid misconfigured endpoints (variants of the mix-up attack)
- Easier support for new (security) mechanisms (PAR, JAR, PKCE, etc.)
- Easier key exchange
- Promote use of the OAuth issuer, also for mix-up mitigation

Proposal: iss for Mix-Up Mitigation

Current Recommendation: Use separate redirect URIs per issuer!

+ based on existing OAuth features
- not suitable for schemes with centralized client registration (open banking!)
- needs a lot of explanation for developers
- easy to get wrong
- hard to automate in libraries

Alternative: `iss` parameter in authorization response. So far not standardized.

Proposal: iss for Mix-Up Mitigation

New: draft-meyerzuselhausen-oauth-iss-auth-resp

Defines the `iss` parameter in the authorization response (+ metadata flag).

- + Simple mechanism
- + Formally proven security against mix-up attacks
- + Easy to automate in libraries when metadata flag is evaluated

Proposal:
Clients MUST prevent mix-up attacks, either by per-issuer redirect URIs or by using the iss parameter.

separate document

Go for WGLC_2?