OAuth 2.0 Security Best Current Practice

● Refines and enhances security guidance for OAuth 2.0 implementers

● Updates, but does not replace:
  ○ OAuth 2.0 Threat Model and Security Considerations (RFC 6819)
  ○ OAuth 2.0 Security Considerations (RFC 6749 & 6750)

● Updated, more comprehensive Threat Model

● Description of Attacks and Mitigations

● Simple and actionable recommendations
Changes Since IETF-103 (-09 … -12)

● WG Consensus to discourage use of implicit and the like

“… clients SHOULD NOT use the implicit grant (response type "token") or any other response type issuing access tokens in the authorization response, such as "token id_token" and "code token id_token", unless the issued access tokens are sender-constrained and access token injection in the authorization response is prevented.”

● Recommendation: use code + PKCE for SPAs

● Adoption has started
  ○ OpenID FAPI WG removed “code token id_token” from Part 2 spec
  ○ oidc-client-js library added support for code+PKCE
Changes since IETF-103 (contd.)

- Added text on refresh tokens including replay detection based on refresh token rotation
- Added refined attack model incorporating more dynamic setups
- Migrated draft to markdown
But we are not done yet...
Proposed Changes

- Discourage use of Resource Owner Password Credentials Grant Type
- Recommend client authn methods based on public key crypto
  - mTLS
  - private_key_jwt
- Use PKCE for CSRF prevention instead of state parameter
  - PKCE is mandatory now and can fulfill this additional task
  - Simplifies recommendations and makes state available again for original purpose (carry client transaction data)
- Recommend PKCE mode S256 over plain
  - Attacker cannot utilize leaked code_challenge to complete transaction
Further Challenges (FYI)

- **OAuth in SPAs**
  - Mechanisms for sender-constrained tokens using application level signing needed (TBD)

- **Secure Transaction Authorization/Rich Authorization Requests**
  - Lodging Intent Pattern (OIDF FAPI)
  - https://bitbucket.org/openid/fapi/src/master/Financial_API_Lodging_Intent.md

- **JWT protected authorization responses**
  - JWT Secured Authorization Response Mode for OAuth 2.0 (JARM) (OIDF FAPI)
  - https://bitbucket.org/openid/fapi/src/master/Financial_API_JWT_Secured_Authorization_Response_Mode.md
Q & A