OAuth 2

Secure access delegation

Vladimír Tkáč

IBM

What is OAuth?

Open Authorization

for authentication use:

- SAML

- OpenID Connect (OIDC)

What is OAuth

RFC 6749: The OAuth 2.0 Authorization Framework
RFC 6750: The OAuth 2.0 Bearer Token Usage
RFC 7636: Proof Key for Code Exchange by OAuth Public Clients
RFC 8252: OAuth 2.0 for Native Apps
RFC 8628: OAuth 2.0 Device Authorization Grant
RFC 6819: OAuth 2.0 Threat Model and Security Considerations
RFC 7009: OAuth 2.0 Token Revocation
RFC 7662: OAuth 2.0 Token Introspection
RFC 8414: OAuth 2.0 Authorization Server Metadata
...

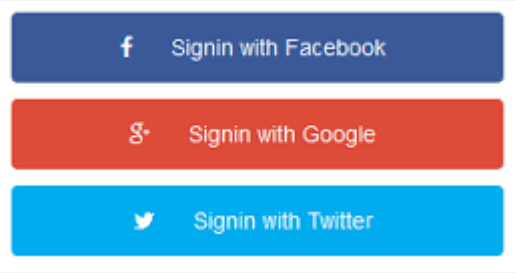
OAuth 2.0 BCP

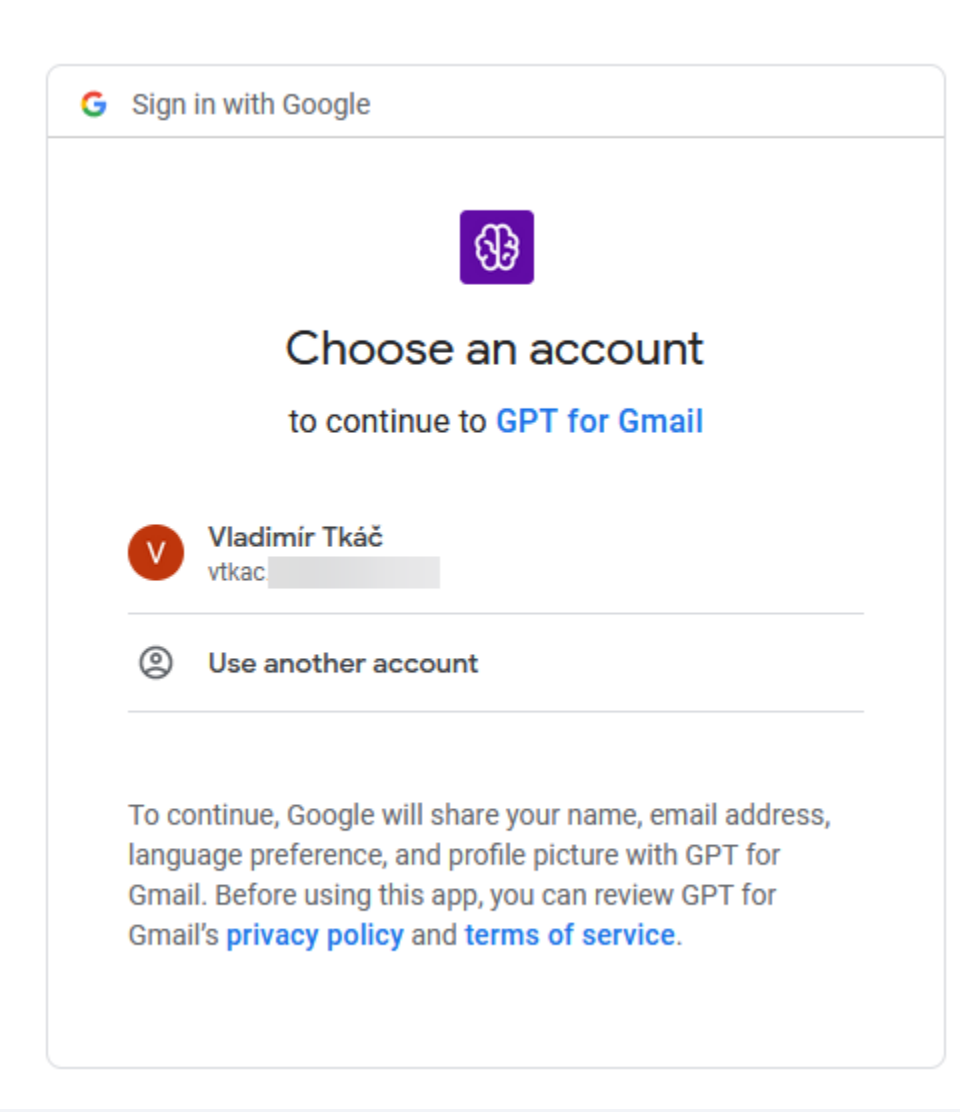
Best Current Practice

Why OAuth?

Before OAuth

After OAuth


Signin with Facebook
Signin with Google
Signin with Twitter


Sign in with Google
Choose an account
to continue to GPT for Gmail
V
Vladimír Tkáč
vtkac
Use another account
To continue, Google will share your name, email address, language preference, and profile picture with GPT for Gmail. Before using this app, you can review GPT for Gmail's privacy policy and terms of service.

How it works?

OAuth Token Types

- Access token
- Refresh token
- ID Token (from OIDC)

OAuth Roles

Resource Owner (User)

User Agent (Browser, mobile app, desktop app, …)

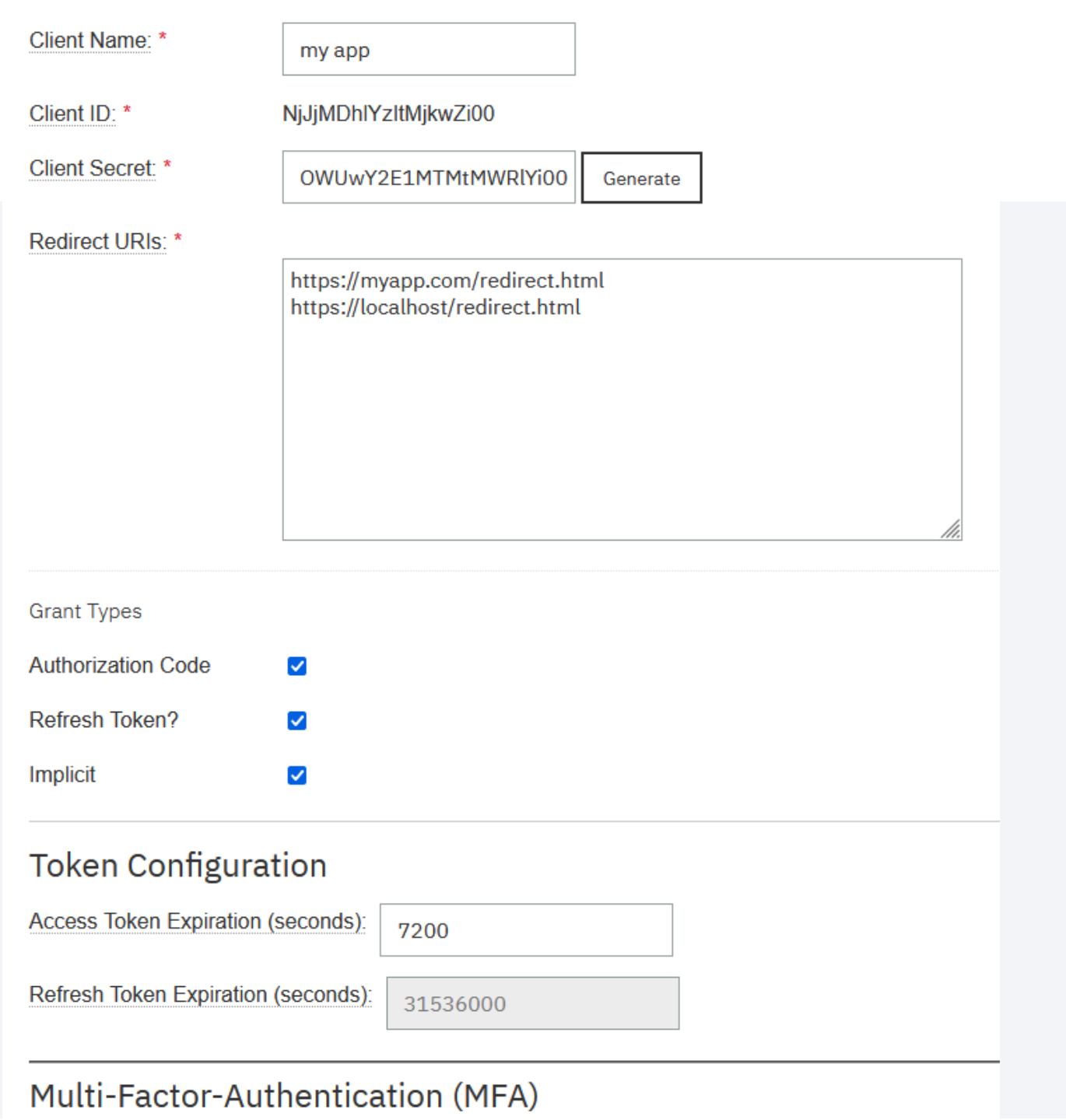
Resource server (App with data)

Client (App requesting access to data)

Authorization server (approves or denies access)

OAuth Client Registration

- Client ID
- Client Secret



OAuth Client types

- Confidential clients
- Public clients

OAuth Flows

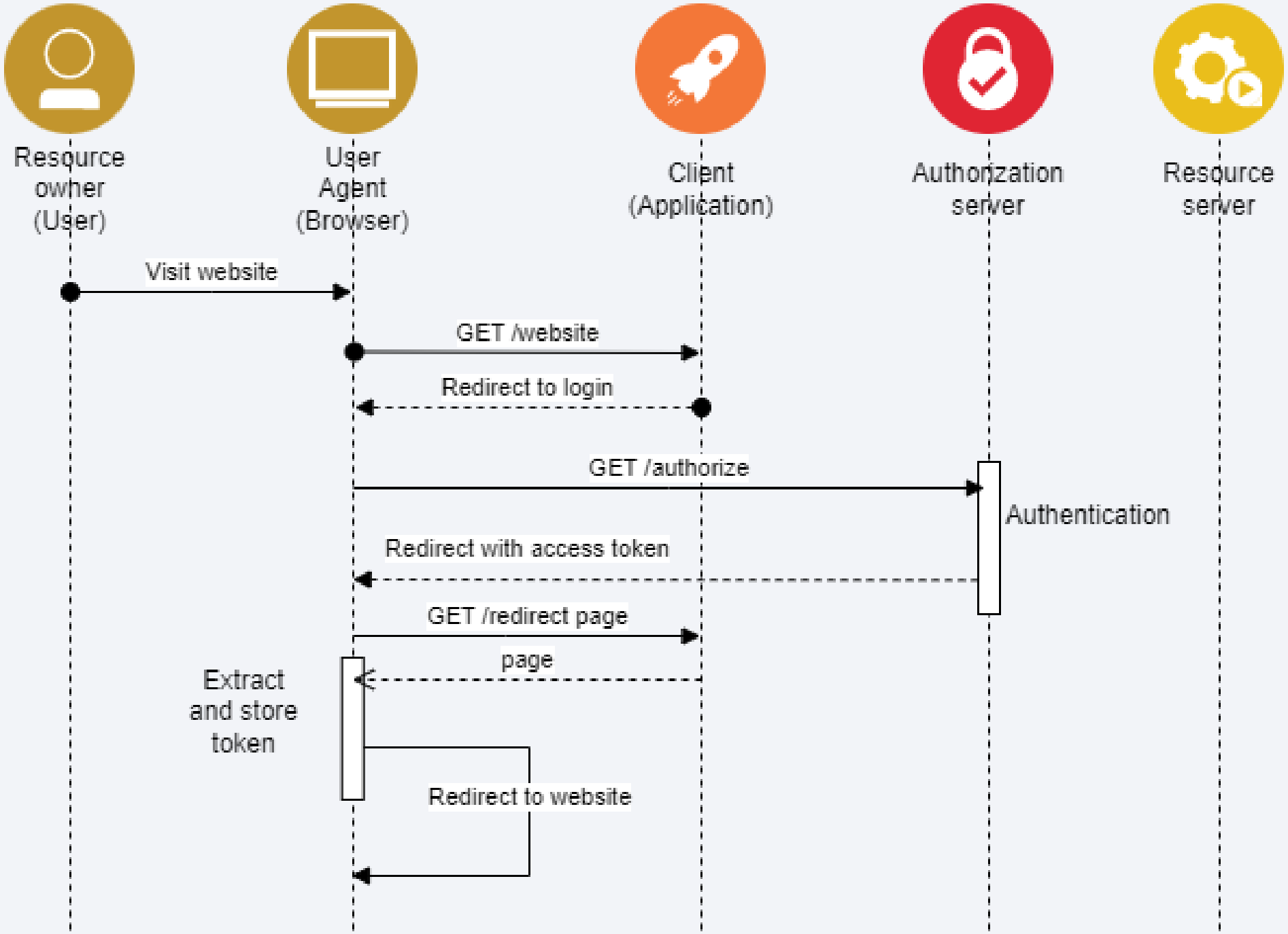
- Implicit Flow
- Authorization Code Flow (with **PKCE**)
- Resource Owner Password Credentials Flow
- Client Credentials Flow
- Device Authorization Flow

Implicit flow

OAuth: Implicit flow

- Designed for native/JavaScript apps
- Not recommended
- Removed in OAuth 2.1

Implicit Flow



Implicit Flow

Request:

/authorize?client_id=A1B2C300Z
&redirect_uri=https://myapp.com/redirect.html
&response_type=token
&scope=profile

Response:

302

Location:

https://myapp.com/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7l
&scope=profile
&token_type=Bearer
&expires_in=7200

Implicit Flow:

Open Redirect

Request:

/authorize?client_id=A1B2C300Z
&redirect_uri=https://evil.myapp.com/redirect.html
&response_type=token
&scope=profile

Response:
302

Location:

https://evil.myapp.com
/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7l
&scope=profile
&token_type=Bearer
&expires_in=7200

Implicit Flow:

Token Injection

Go Here:

https://myapp.com
/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7l
&scope=profile
&token_type=Bearer
&expires_in=7200

Implicit Flow:

State

Request:

/authorize?client_id=A1B2C300Z
&redirect_uri=https://myapp.com/redirect.html
&response_type=token
&scope=profile
&state=kdafoe24r2of24rn4or

Response:

302

Location:

https://myapp.com/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7l
&scope=profile
&token_type=Bearer
&expires_in=7200
&state=kdafoe24r2of24rn4or

OAuth: Implicit flow

- Access token exposure

Authorization code flow

OAuth: Authorization code with PCKE flow

- Used by confidential and public clients
- Client's integrity is verified
  - *Proof Key for Code Exchange (PKCE)*

Proof Key for Code Exchange (PKCE)

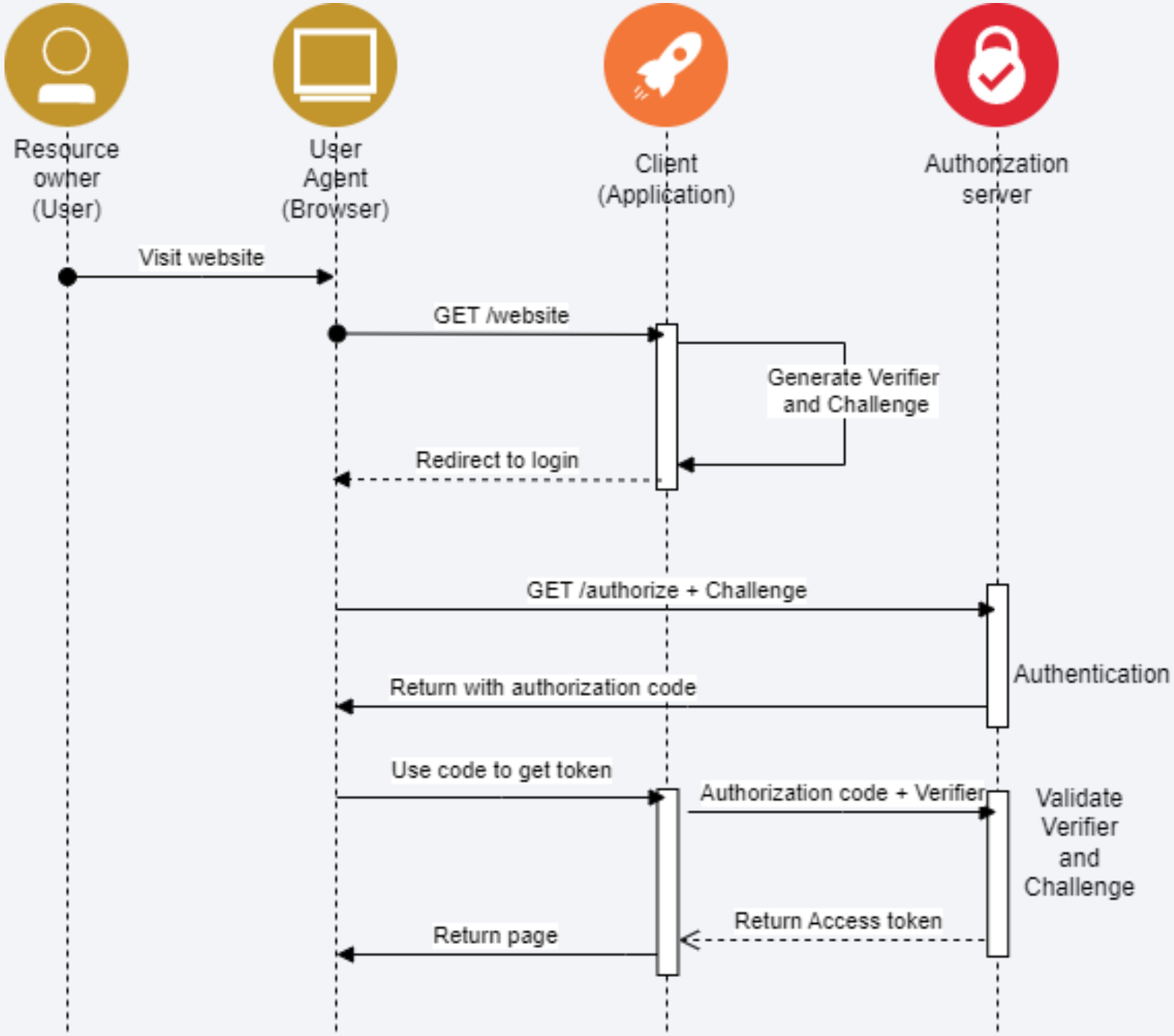
Code verifier

– random value generated by the client application

Code challenge

– base64(s256(code_verifier))

Authorization Code Flow (with **PKCE**)



OAuth: Authorization code with PCKE flow

- ~~Open Redirect~~
- ~~Token Injection~~
- ~~Access token exposure~~
- Authorization Code Interception

OAuth 2.1

OAuth 2.1 (in-progress)

- PKCE is required for all OAuth clients using the authorization code flow
- Redirect URIs must be compared using exact string matching
- The Implicit flow is omitted from this specification
- The Resource Owner Password Credentials grant is omitted from this specification
- Refresh tokens for public clients must either be sender-constrained or one-time use

Q&A

Thank you