

OAuth 2

Secure access delegation

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

Step 1
Find Friends

Step 2
Profile Information

Step 3
Profile Picture


Are your friends already on Facebook?
Many of your friends may already be here. Searching your email account is the fastest way to find your friends on Facebook.


 **Gmail**


Your Email:


Email Password:

[Find Friends](#)

 Facebook will not store your password.

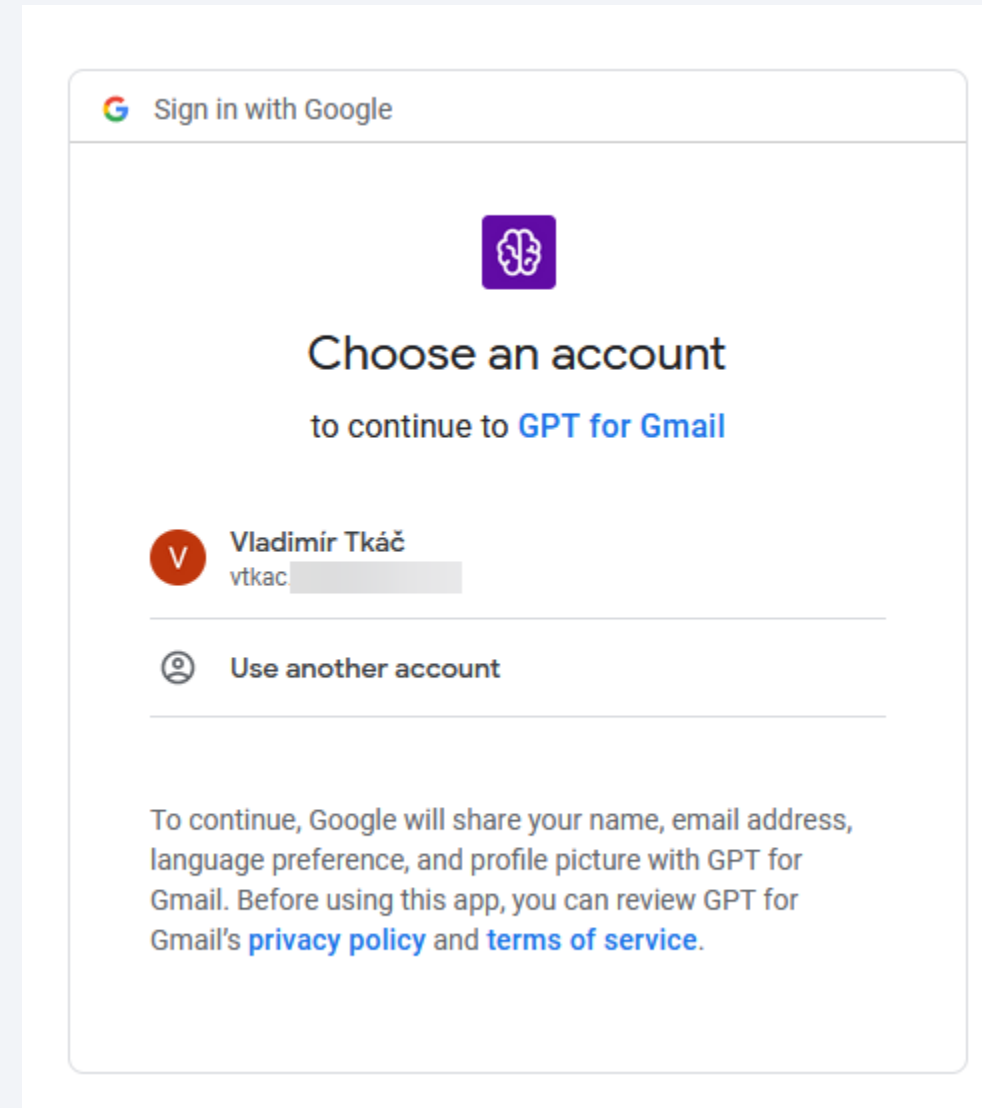
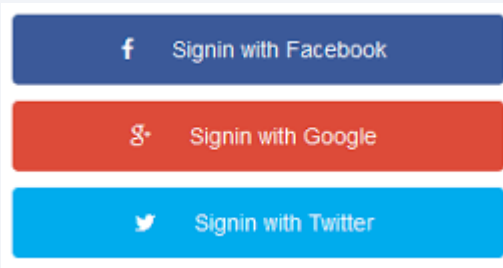
 **Yahoo!** [Find Friends](#)

 **Windows Live Hotmail** [Find Friends](#)

 **Other Email Service** [Find Friends](#)

After OAuth

OAuth 2



How it works?

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



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

Client Name: *

Client ID: * NjJjMDhYzltMjkwZi00

Client Secret: *

Redirect URIs: *

https://myapp.com/redirect.html
https://localhost/redirect.html

Grant Types

Authorization Code

Refresh Token?

Implicit

Token Configuration

Access Token Expiration (seconds):

Refresh Token Expiration (seconds):

Multi-Factor-Authentication (MFA)

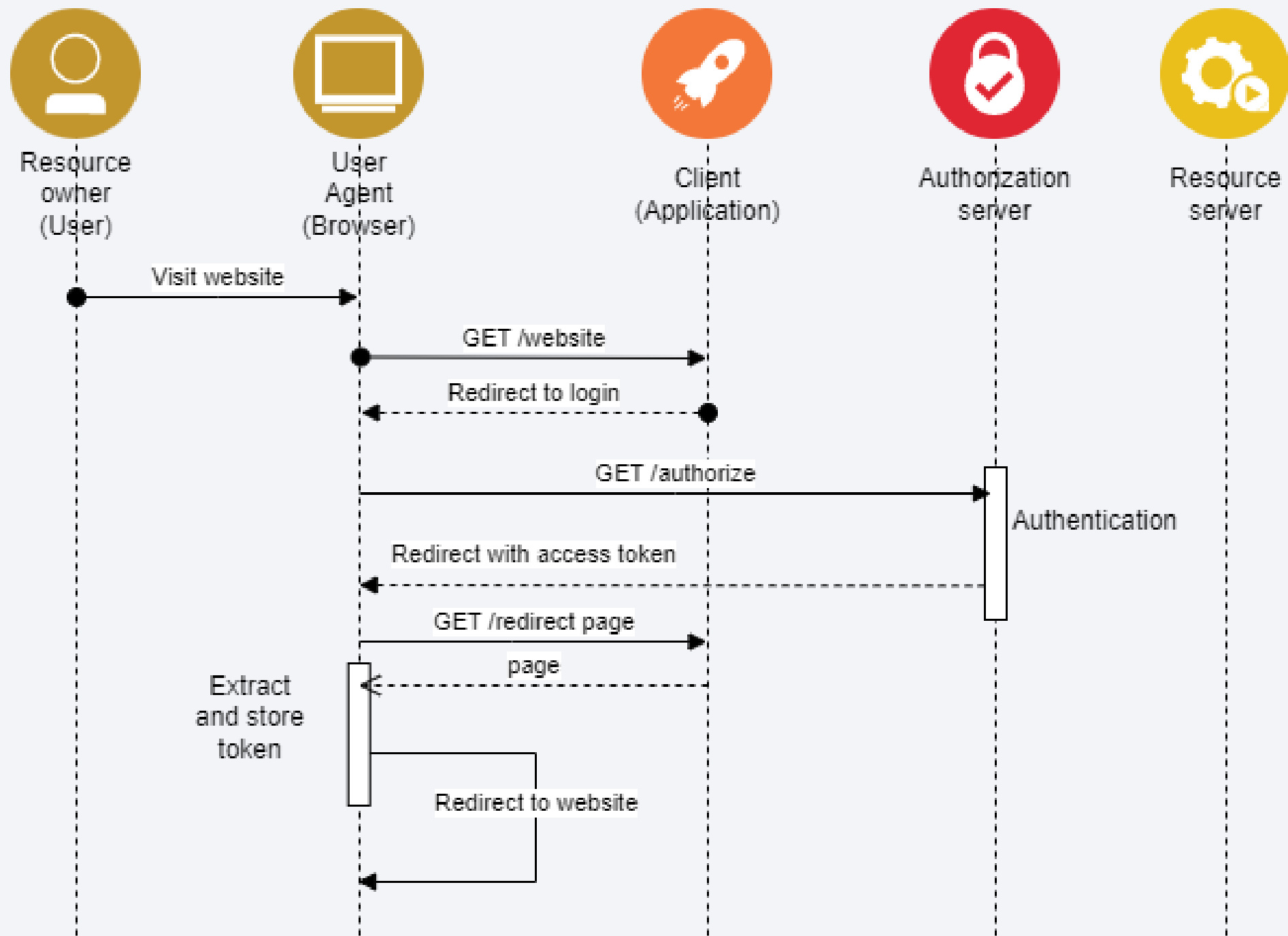
- Confidential clients
- Public clients

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

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=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklb  
bVKbMER7I  
&scope=profile  
&token_type=Bearer  
&expires_in=7200
```

Implicit
Flow:

Request:

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

Open
Redirect

Response:

302

Location:

```
https://evil.myapp.com  
/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7I  
&scope=profile  
&token_type=Bearer  
&expires_in=7200
```

Implicit
Flow:

Go Here:

Token
Injection

[https://myapp.com
/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7I
&scope=profile
&token_type=Bearer
&expires_in=7200](https://myapp.com/redirect.html#access_token=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklbbVKbMER7I&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=OpdyU8c1Zdzyb0Xg2gJmkCsiy5TCklb  
bVKbMER7I  
&scope=profile  
&token_type=Bearer  
&expires_in=7200  
&state=kdafoe24r2of24rn4or
```

• Access token exposure



The screenshot shows a GitHub thread with three comments. Each comment includes a URL with an access token and an expiration time. The first comment shows a token starting with 'Y4t07FkNV92rMJEjI7dPikcTd1HPbCccmJRAZ[...]99e56b0f0839708d'. The second comment shows a token starting with 'fKOzFV1YTggXnU4MPvL8P7v0dQspN0IRJcU9d[...]8d2bf3926aa091f9'. The third comment shows a token starting with 'Ex5KOC83c3JsHSLjcFS6XK3iZr6KCNiD8HUYZ[...]bd10658f0bbdeb64'. All tokens are highlighted in yellow in the original image.

likewise if trying to go to [\[redacted\]](#) I get redirected here:
[\[redacted\]/redirect.html#access_token=Y4t07FkNV92rMJEjI7dPikcTd1HPbCccmJRAZ\[...\]99e56b0f0839708d&token_type=Bearer&expires_in=7200](#) ... Show more
1 reaction 1 reply

hi support team. I'm trying to open my publisher site, but when it is going to the URL, it suddenly ... -odm-r
When I click on it, I end up on this URL:
[\[redacted\]/redirect.html#access_token=fKOzFV1YTggXnU4MPvL8P7v0dQspN0IRJcU9d\[...\]8d2bf3926aa091f9&token_type=Bearer&expires_in=7200](#) ... Show more
1 reaction 3 replies

hi - my page is loading briefly and then shows a redirect page with white screen:
Link: [\[redacted\]](#)
Redirect link with white screen:
[\[redacted\]/redirect.html#access_token=Ex5KOC83c3JsHSLjcFS6XK3iZr6KCNiD8HUYZ\[...\]bd10658f0bbdeb64&token_type=Bearer&expires_in=7200](#)
5 replies

Authorization code flow

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

Proof Key for Code Exchange (PKCE)

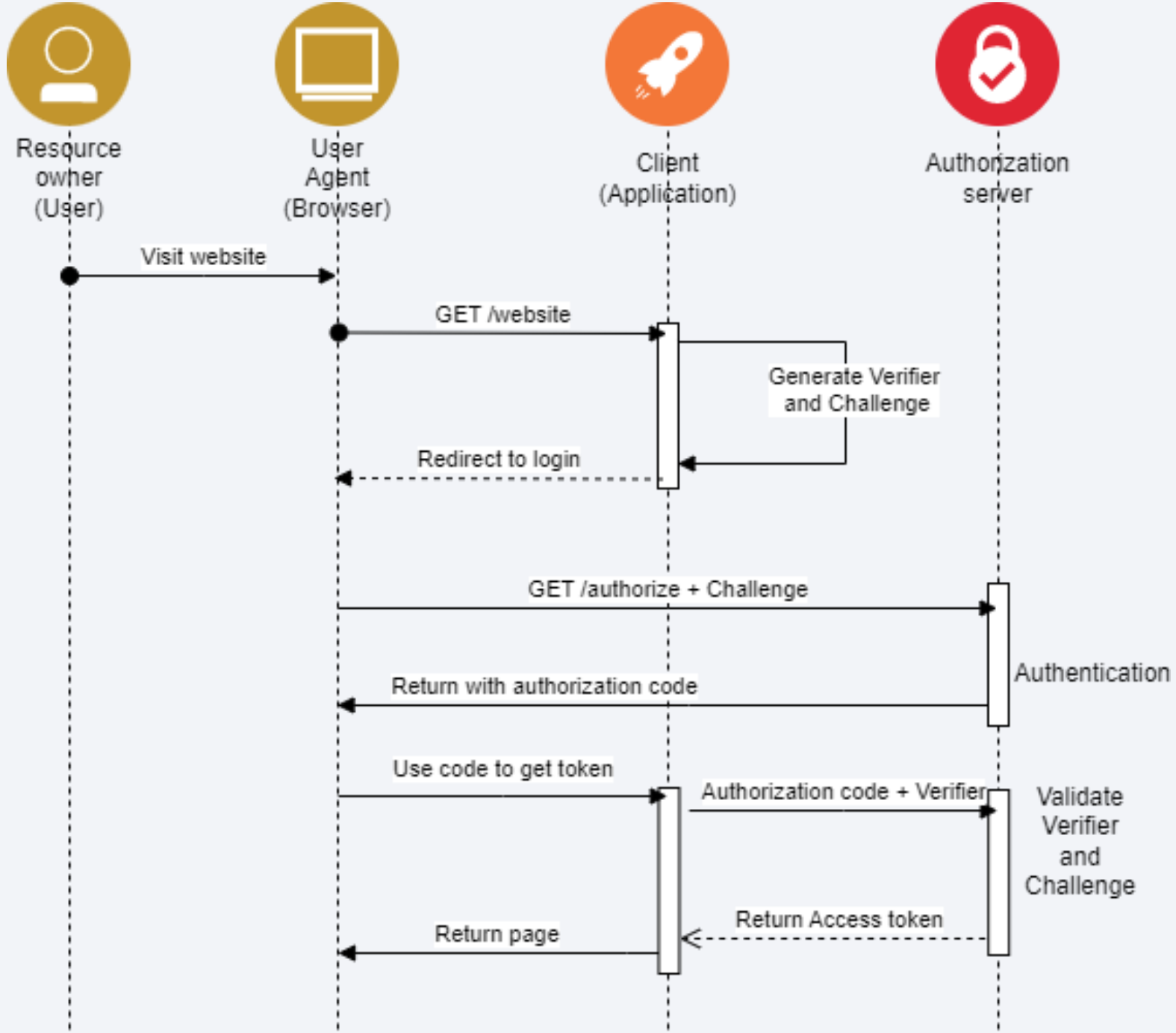
Code verifier

- random value generated by the client application

Code challenge

- `base64(s256(code_verifier))`

Authorization Code Flow (with PKCE)



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

OAuth 2.1

- 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