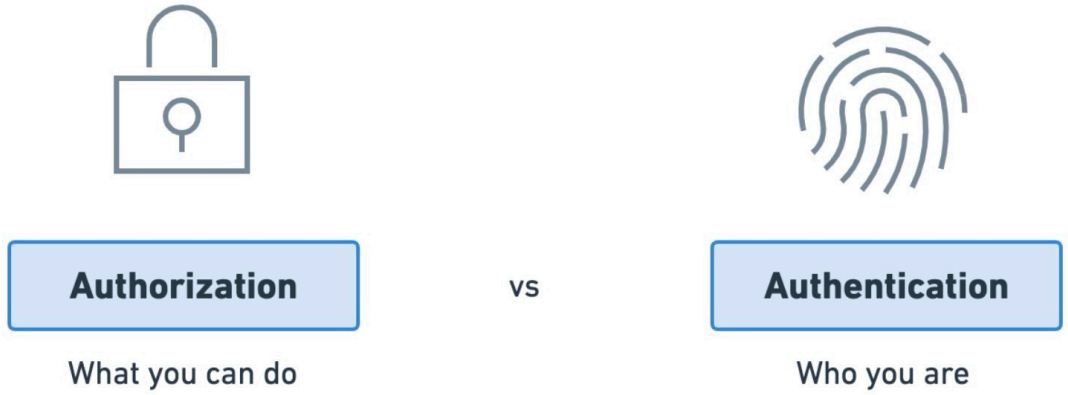
Wyższa Szkoła Ekonomii i Informatyki w Krakowie

Programowanie aplikacji webowych

Ryszard Brzegowy

Uwierzytelnianie, autoryzacja, kolejne konto…

- problem - aplikacje potrzebują identyfikować użytkownika
  - sklep internetowy - konto użytkownika
- problem - użytkownicy nie potrzebują setek kont do każdej z aplikacji
- problem - aplikacja potrzebuje uprawnienia do innego serwisu w którym użytkownik ma konto
  - sklep internetowy - polecenie produktu przez aplikację sklepu na ścianie fb użytkownika



źródło: https://blog.postman.com/pkce-oauth-how-to/

OAuth 2.0

- Otwarty standard autoryzacji (nie uwierzytelnienia!) użytkownika
- OAuth bazuje na tokenach i protokole https
- W procesie występuje trzech/czterech aktorów - serwer autoryzacyjny/zasobów, klient i użytkownik
- Serwer autoryzacyjny udziela dostępu klientowi do wybranych zasobów bez przekazywania do klienta danych prywatnych użytkownika (np. hasła)
- Klientami są nasze aplikacje
- Centrami autoryzacji są dostawcy poświadczeń - Google/Facebook/Microsoft/Apple itd.,
- Użytkownikami jesteśmy my.

- Inny standard: SAML (bardziej "enterprise" z uwagi na zasadę działania)

WSEI

Scenariusze

- Czysty OAuth 2.0 - uzyskaj dostęp do wybranych danych użytkownika (e-mail, avatar, lista znajomych itd)
- W połączeniu z OpenID Connect - logowanie (bez zakładania konta w aplikacji)

- Samo OAuth 2.0 NIE SŁUŻY DO UWIERZYTELNIANIA! Potencjalne problemy:
  - token na okaziciela
  - token jako dowód uwierzytelnienia
  - dostęp do danych jako dowód uwierzytelnienia

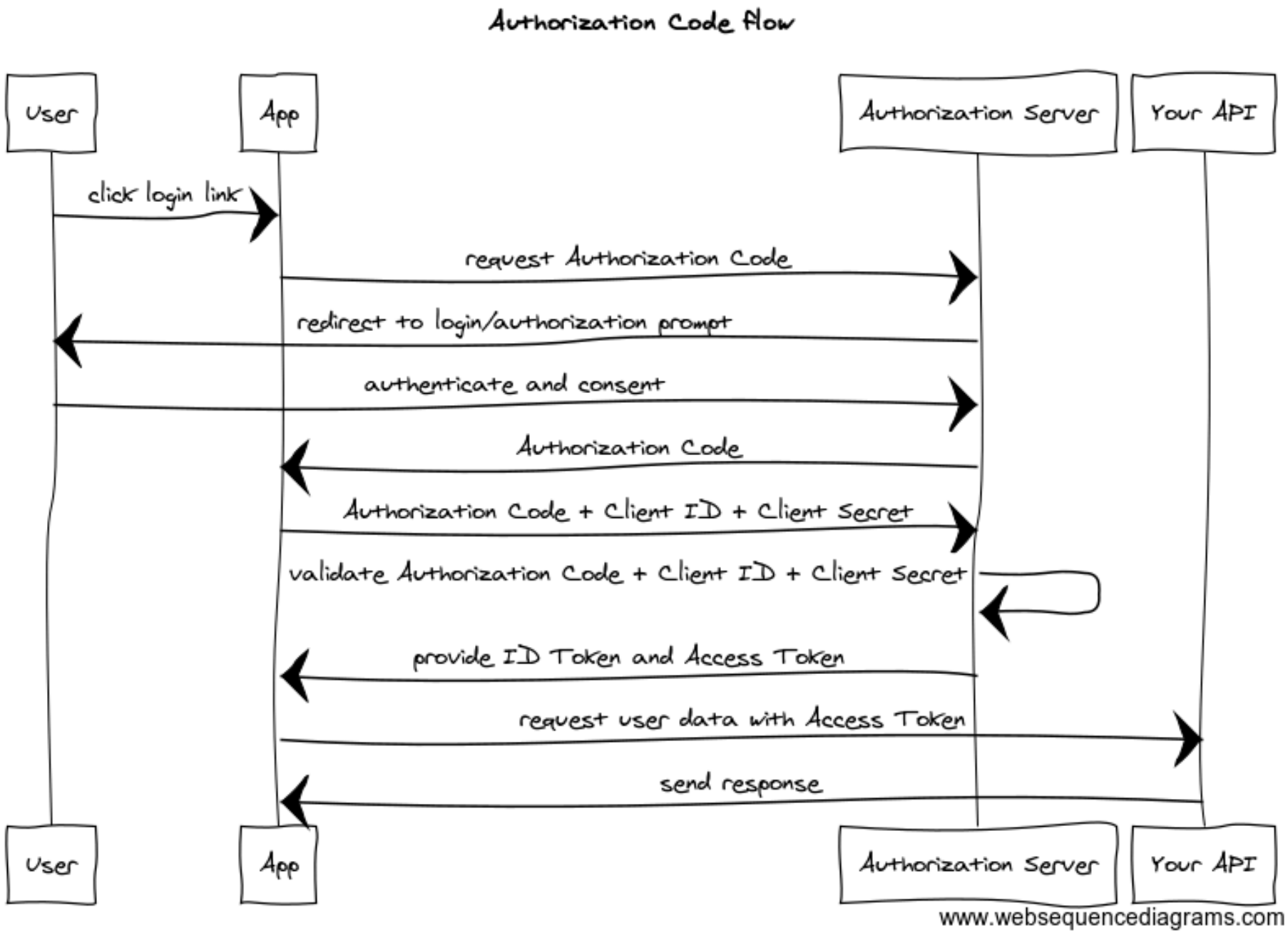
WSEI

OpenID Connect

- OpenID Connect (OIDC) to standard zbudowany na kanwie OAuth
- Bardzo często serwisy opisywane jako korzystające z OAuth 2.0, w rzeczywistości korzystają z opakowania w postaci OIDC
- OIDC łączy w sobie autoryzację świadczoną przez OAuth z uwierzytelnieniem użytkownika
- OIDC rozszerza informacje przekazywane z centrum autoryzacyjnego do klienta o wybrane dane użytkownika

WSEI

OAuth 2.0

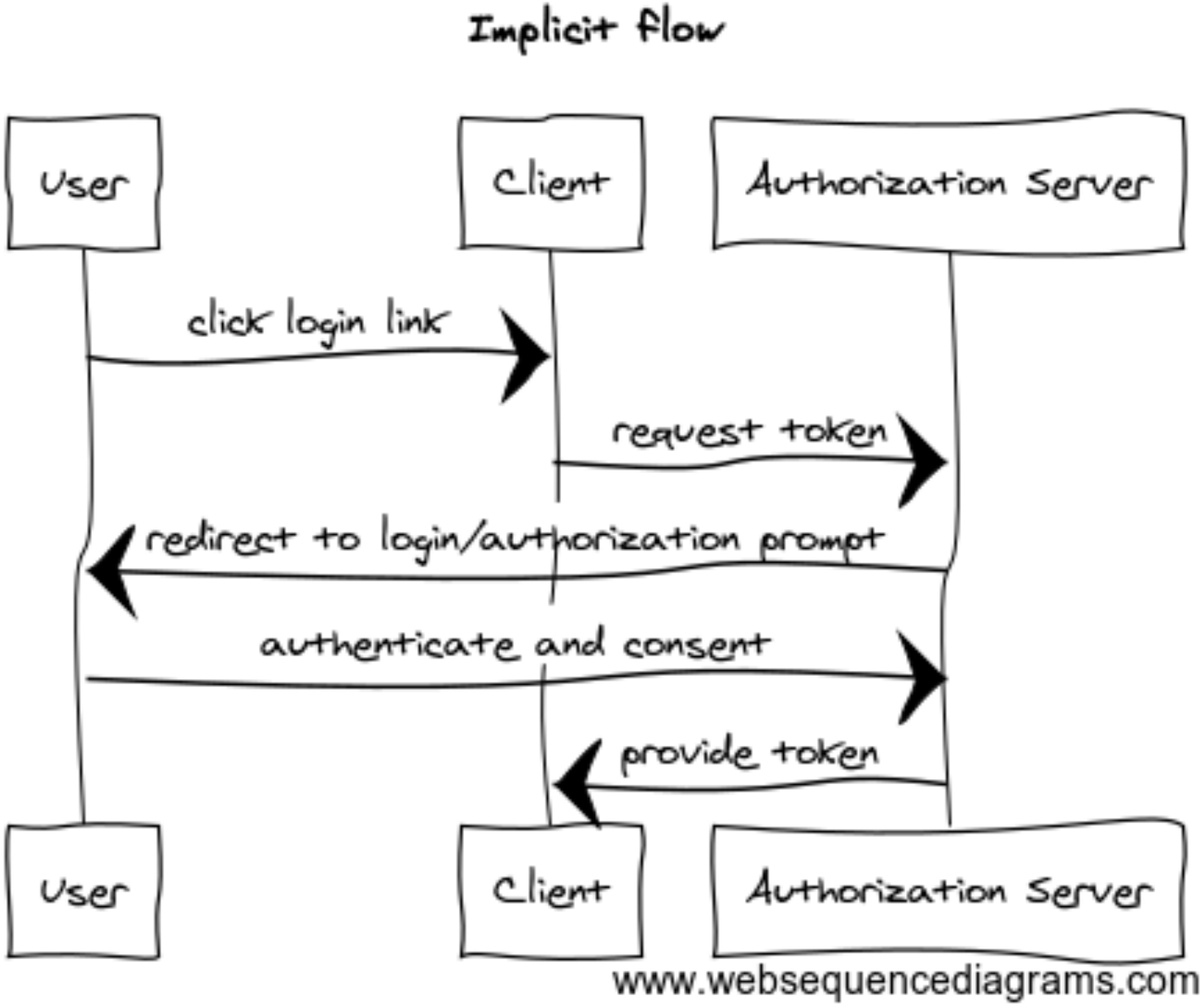


źródło: https://blog.postman.com/pkce-oauth-how-to/

OAuth 2.0

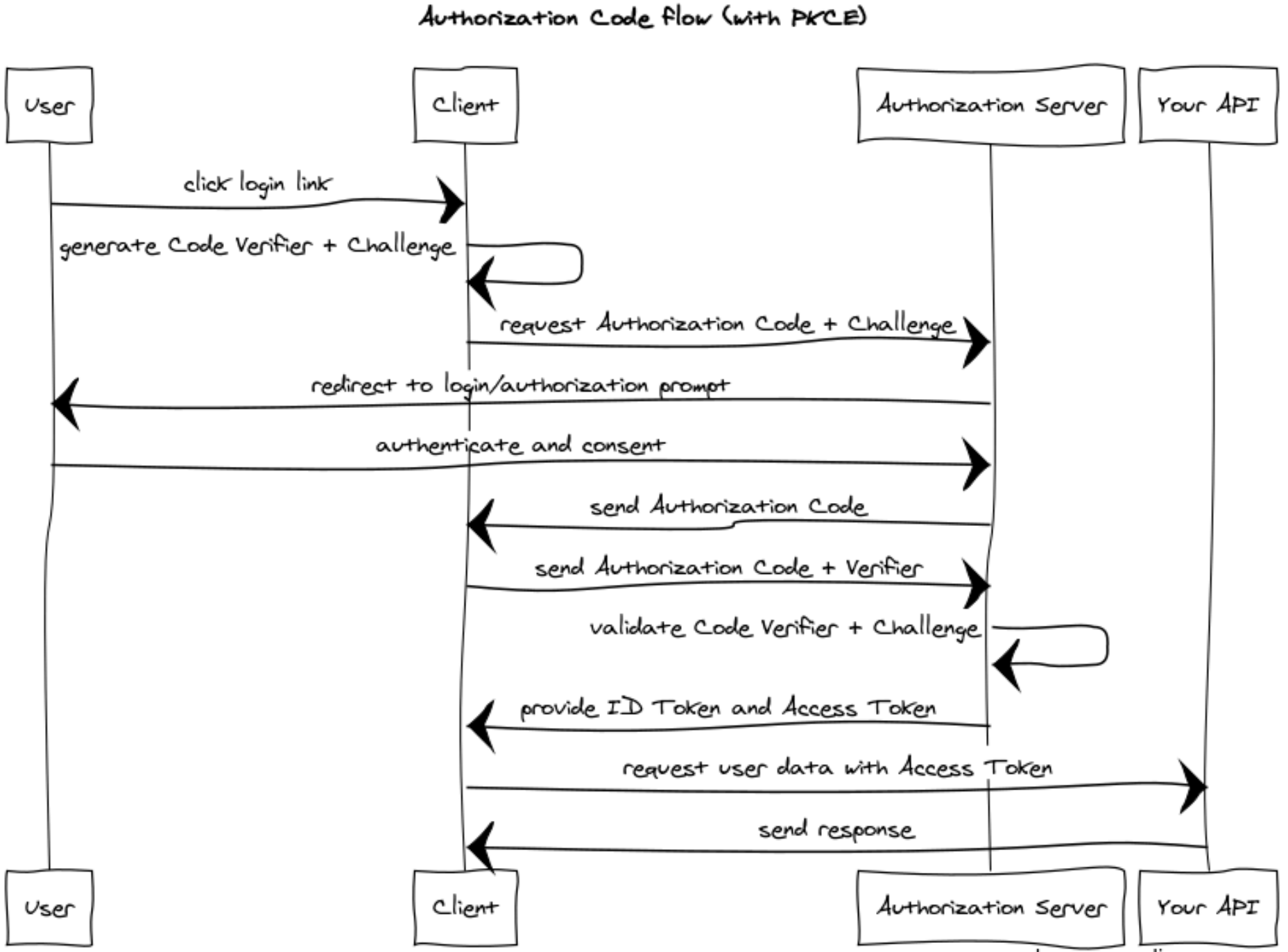
- Implicit flow - dla apek webowych (nie ma secret)

źródło: https://blog.postman.com/pkce-oauth-how-to/



OAuth 2.0

- PKCE



źródło: https://blog.postman.com/pkce-oauth-how-to/

Zasoby

- https://www.oauth.com/playground/index.html
- jwt.io
- https://auth0.com/
- https://sekurak.pl/oauth-2-0-jak-dziala-jak-testowac-problemy-bezpieczenstwa/
- https://blog.postman.com/pkce-oauth-how-to/

WSEI

Wyższa Szkoła Ekonomii
i Informatyki w Krakowie