



Table of content

- What is Oauth2
- Differences Oauth1 and Oauth2
- Oauth2 Roles
- Protocol types
- Protocol flows
- Token
- Demo Registration
- Workshop Client
- Workshop Server

What is Oauth2

- Authorization framework
- HTTP service
- All applications



Differences Oauth1 and Oauth2

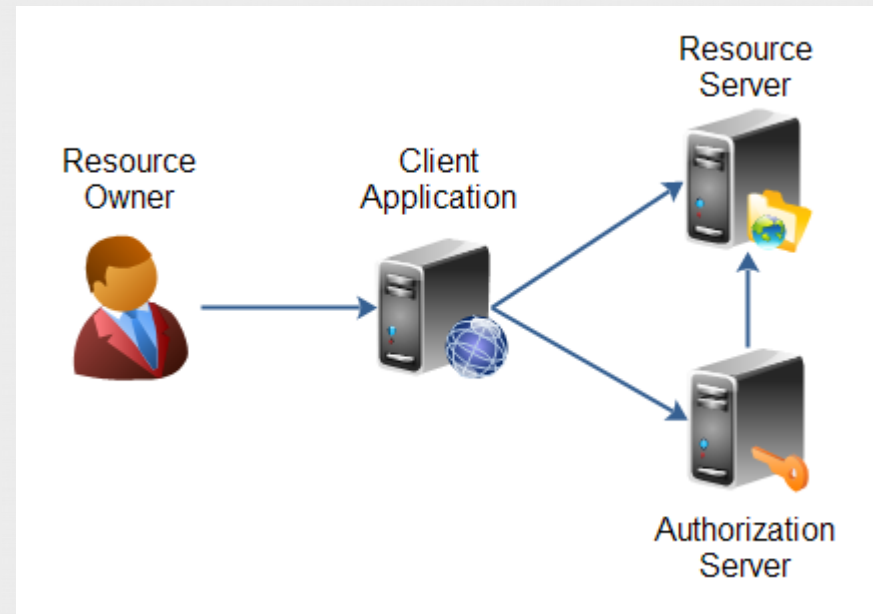
- Security
- Token life span
- Flows
- Roles

Oauth2 roles

- Resource owner
- Resource server
- Client Application
- Authorization Server
- User Agent

OAuth2 roles/structure

- Resource owner(User)
- Resource/Authorization Server: API
- Client: Application



Authorization grant

- Authorization code
- Implicit
- Resource Owner Password Credentials
- Client credentials
- Refresh token



Grant type: authorization code

- Most common
- Used for server side applications
- 5 steps
 - Authorization code link
 - User Authorizes application
 - Application receives authorization code
 - Application requests access token
 - Application receives access token

Grant type: authorization code

- Link components

code - Indicates that your server expects to receive an authorization code
client_id - The client ID you received when you first created the application
redirect_uri - Indicates the URI to return the user to after authorization is complete
scope - One or more scope values indicating which parts of the user's account you wish to access
state - A random string generated by your application, which you'll verify later

```
@Bean
@ConfigurationProperties("facebook.client")
public AuthorizationCodeResourceDetails
```

```
@Bean
@ConfigurationProperties("facebook.resource")
public ResourceServerProperties
```

Implicit grant

- Used for mobile/web applications
- Running resource in browser
- Simplicity

Grant type: Resource owner password credentials

- Username and password
- Only on authorization server
- Depending on trust

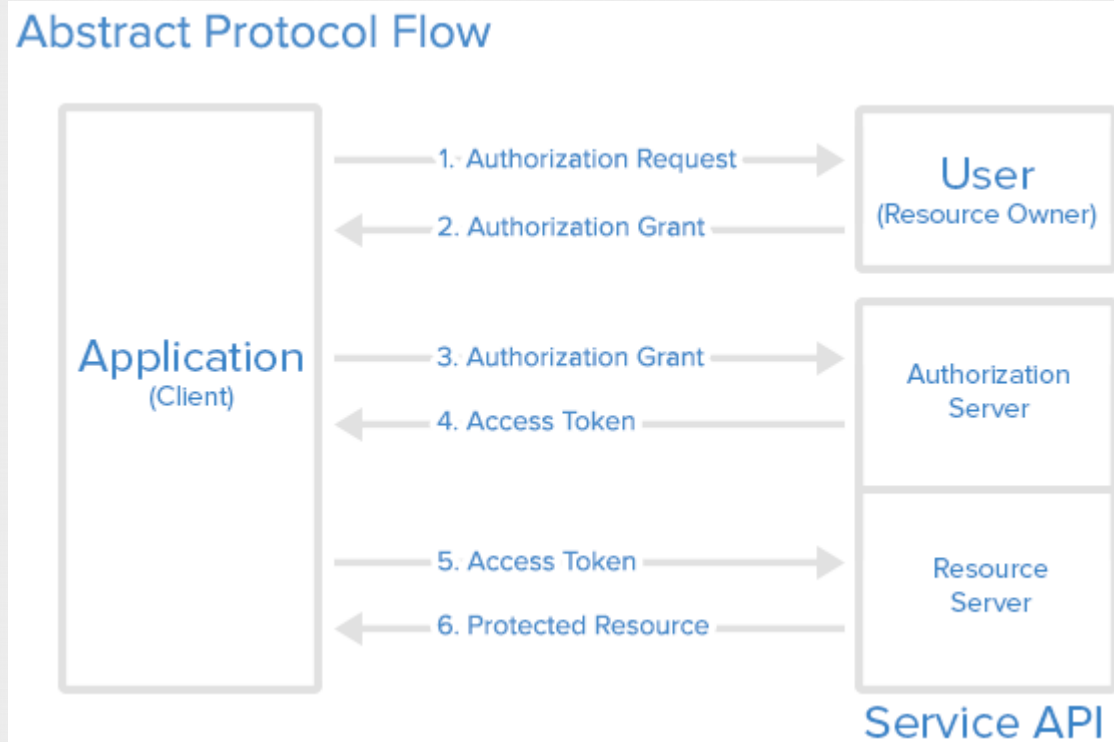
Grant type: Client credentials

- Own service accounts
- Updating registered description
- Access other data

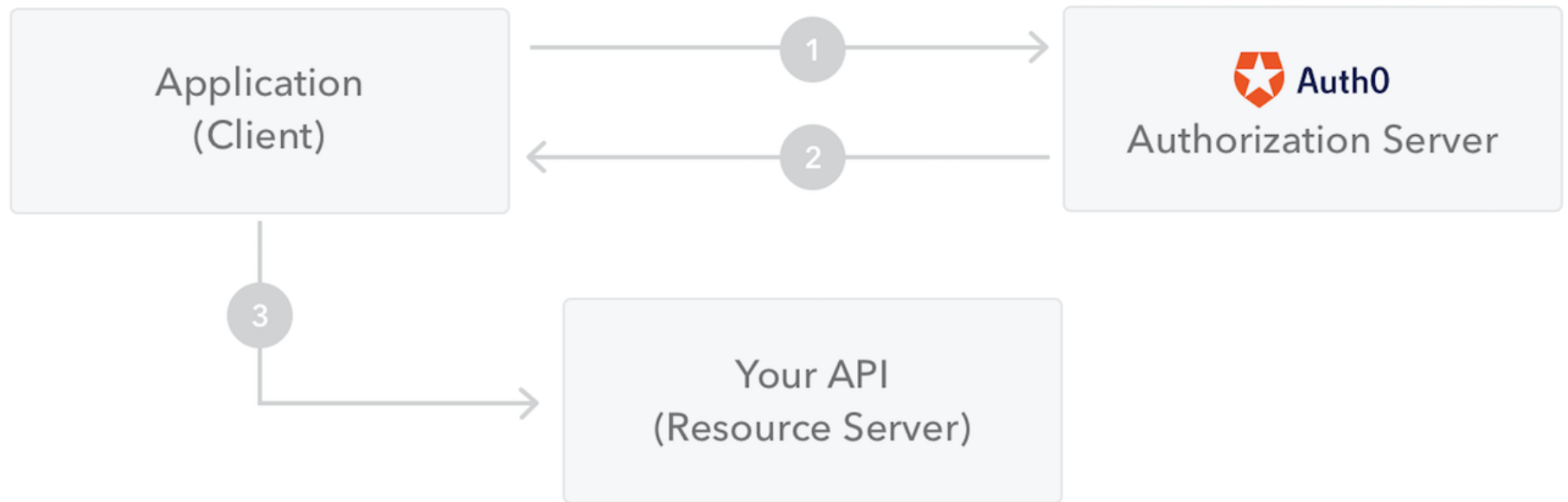
Refreshing your token



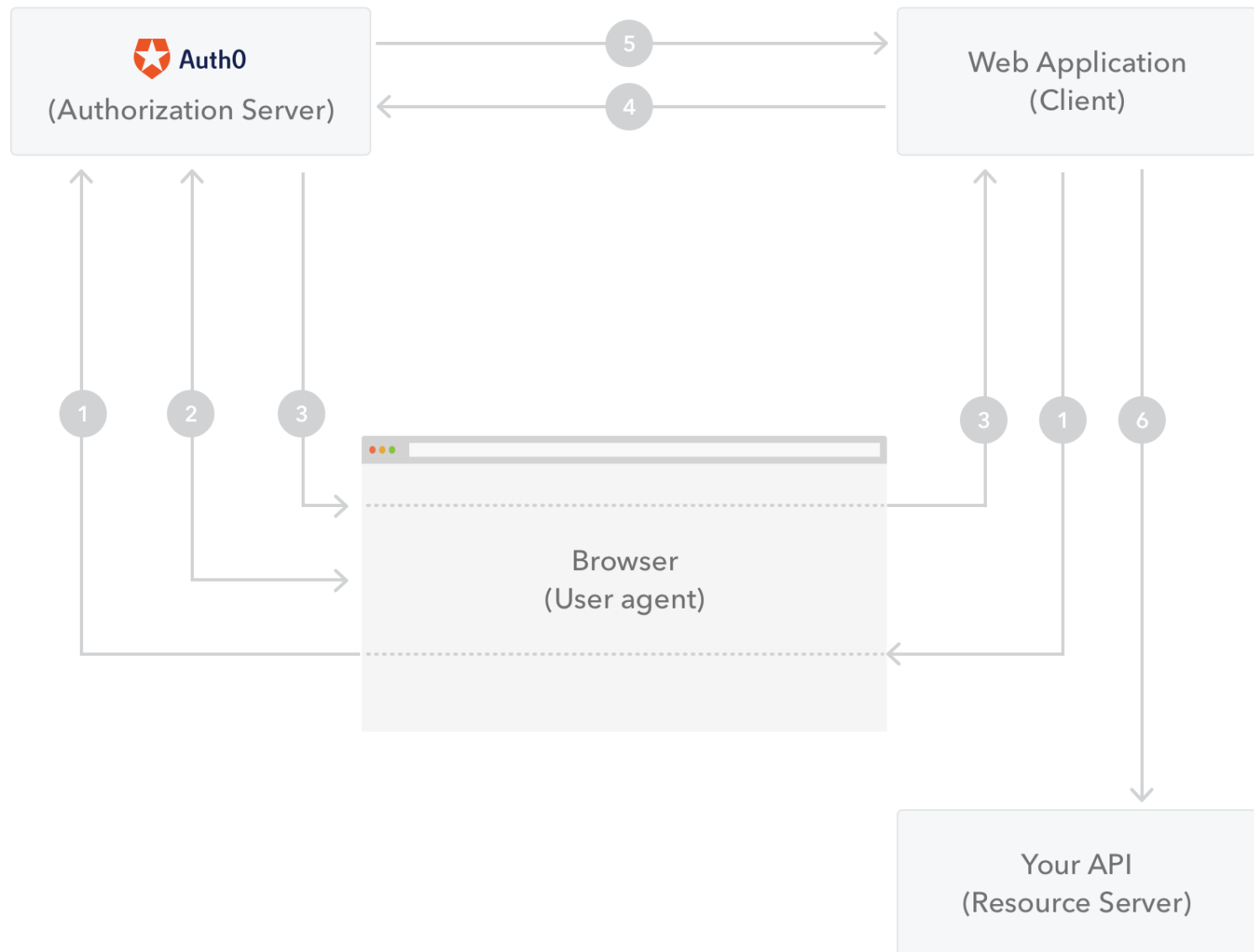
Abstract protocol flow



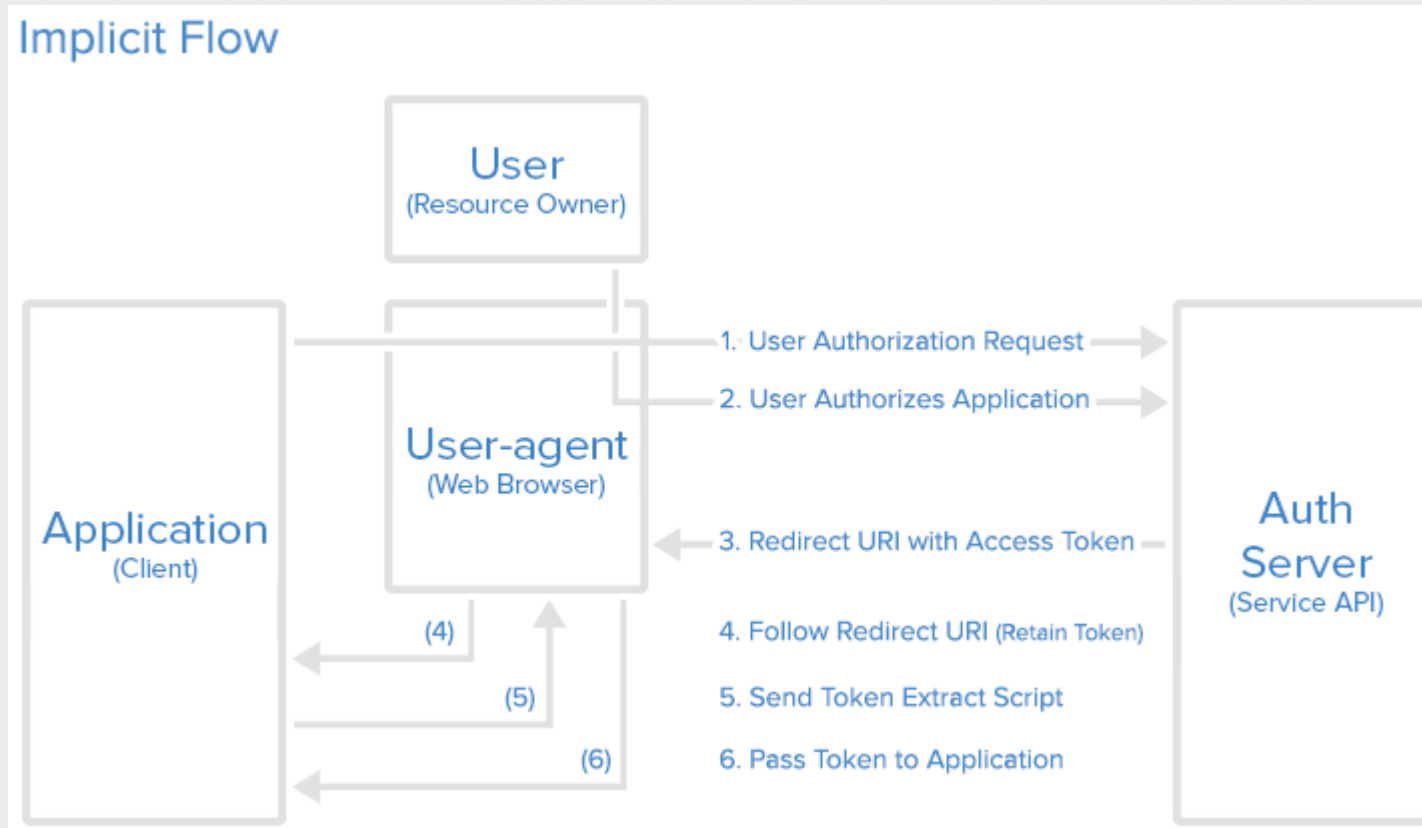
Client Credentials Grant



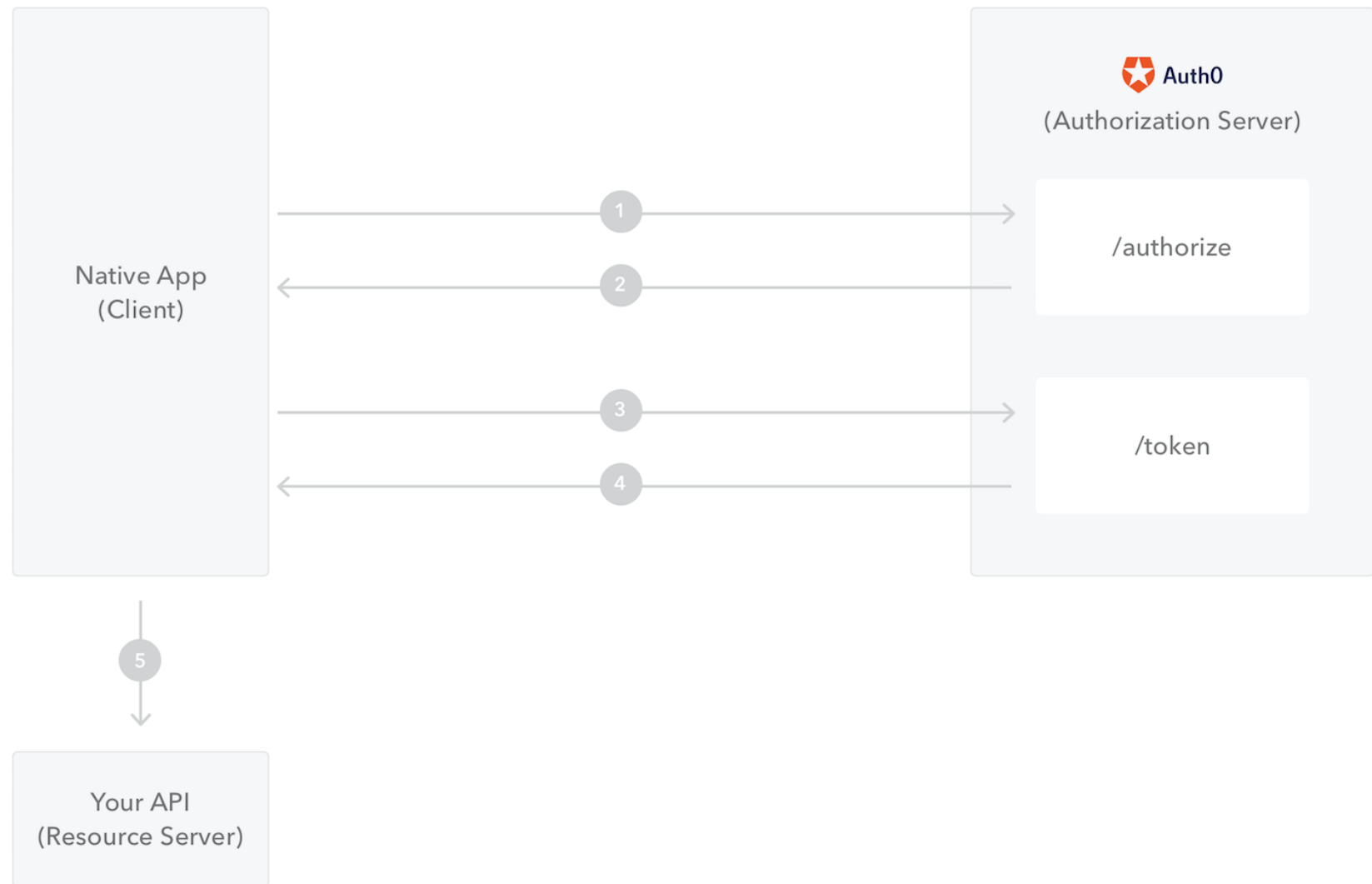
Authorization Code Grant

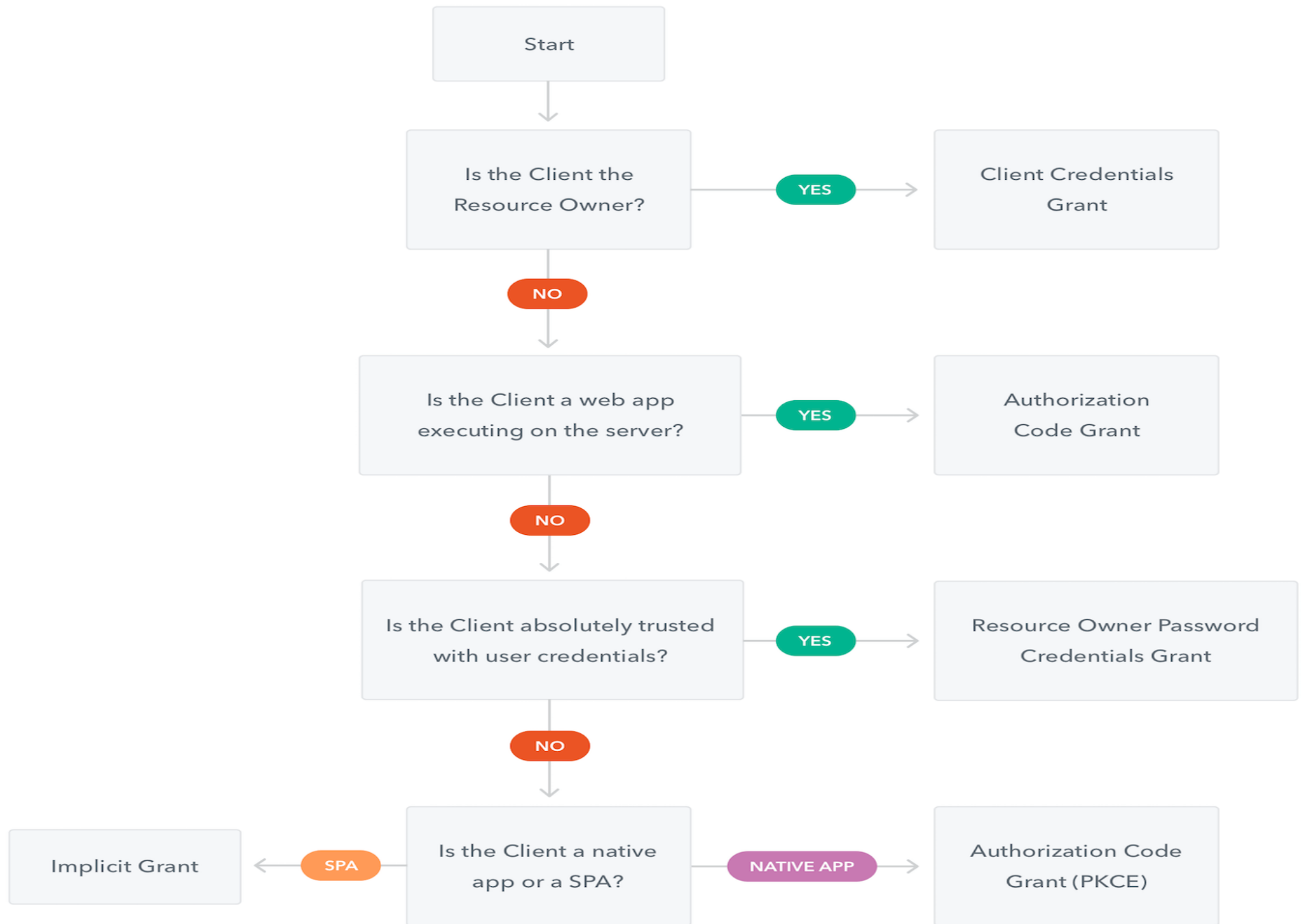


Implicit grant



Authorization Code Grant (PKCE)





Token

- Request:

`client_id` =CLIENT_ID
`client_secret` =CLIENT_SECRET
`grant_type`=authorization_code
`Code`=AUTHORIZATION_CODE
`redirect_uri` =CALLBACK_URL

- Receive:

`access_token`:"ACCESS_TOKEN"
`token_type`:"bearer"
`expires_in`:"2592000"
`refresh_token`:"REFRESH_TOKEN"
`scope`:"read"
`uid`:"100101"
`info`:{`"name"`:"Mark E. Mark",`"email"`:"mark@thefunkybunch.com"}}

Workshop registration

Workshop client

Workshop Client

```
facebook:
  client:
    clientId: 233668646673605
    clientSecret: 33b17e044ee6a4fa383f46ec6e28eald
    accessTokenUri: https://graph.facebook.com/oauth/access token
    userAuthorizationUri: https://www.facebook.com/dialog/oauth
    tokenName: oauth_token
    authenticationScheme: query
    clientAuthenticationScheme: form
  resource:
    userInfoUri: https://graph.facebook.com/me
```

Workshop client

```
facebook:
  client:
    clientId: 233668646673605
    clientSecret: 33b17e044ee6a4fa383f46ec6e28eald
    accessTokenUri: https://graph.facebook.com/oauth/access\_token
    userAuthorizationUri: https://www.facebook.com/dialog/oauth
    tokenName: oauth_token
    authenticationScheme: query
    clientAuthenticationScheme: form
    scope: email
  resource:
    userInfoUri: https://graph.facebook.com/me?fields=id,name,email
```