

# OAuth and User Authentication

Jogesh K. Muppala



THE DEPARTMENT OF  
**COMPUTER SCIENCE & ENGINEERING**  
計算機科學及工程學系



香港科技大學  
THE HONG KONG UNIVERSITY OF  
SCIENCE AND TECHNOLOGY

# OAuth 1 and OAuth 2

- Authorization framework based on open standards for Internet users to log into third party websites/apps using their Social Network accounts
  - Facebook, Google, Twitter, Microsoft, Instagram, Github, DigitalOcean and many others
- OpenID is a related but complementary service

# OAuth 1 and OAuth 2

- OAuth 1 protocol:
  - First evolved from Twitter (Blaine Cook)
  - IETF RFC 5849
- OAuth 2 protocol:
  - Focuses on simplifying client development
  - IETF RFC 6749
  - Bearer token usage IETF RFC 6750

# OAuth 2 Roles

- Resource owner: You, the user that authorizes a client application to access their account
- Client Application: Application (website or app) that wants access to the resource server to obtain information about you
- Resource Server: Server hosting protected data (e.g., your personal information)
- Authorization Server: Server that issues an access token to the client application to request resource from the resource server

# OAuth 2 Tokens

- Access token: allows access to user data by the client application
  - Has limited lifetime
  - Need to be kept confidential
  - Scope: parameter used to limit the rights of the access token
- Refresh token: Used to refresh an expired access token

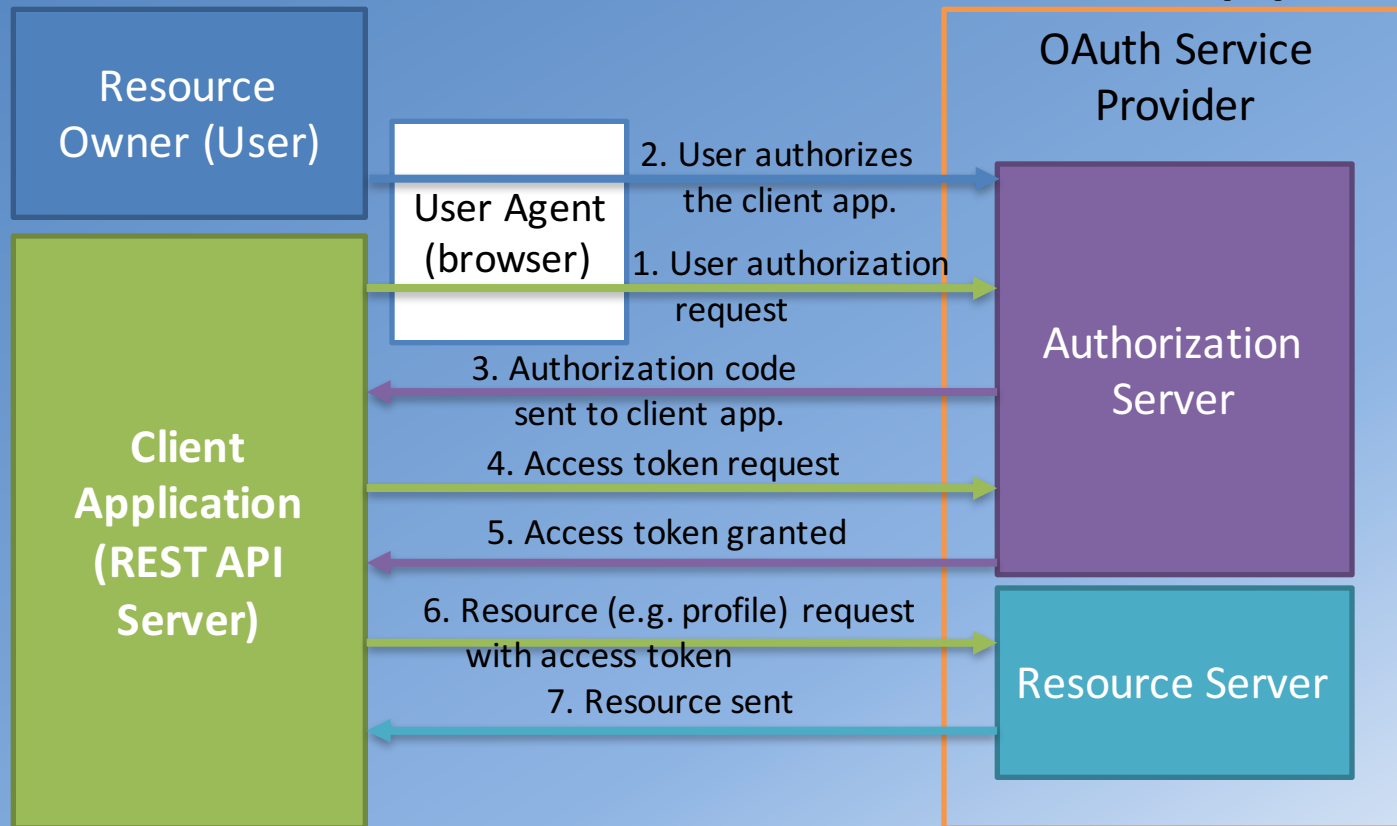
# Client Application Registration

- Register the client application on the OAuth service provider:
  - Client App Id
  - Client Secret
  - Redirect URL: URLs for the client for receiving the authorization code and access token

# Authorization Code Grant Approach

- Used when the client application is a web server.
- Obtain a long-lived access token

# Authorization Code Grant Approach





# Passport-Facebook Module

- Passport strategy for authenticating with Facebook using OAuth 2.0 API
- Installing:  
`npm install passport-facebook --save`
- Using  
`var FacebookStrategy = require('passport-facebook').Strategy;`

# Passport-Facebook Module

- Setting up Facebook Strategy:

```
passport.use(new FacebookStrategy({
  clientID: config.facebook.clientID,
  clientSecret: config.facebook.clientSecret,
  callbackURL: config.facebook.callbackURL
},
function(accessToken, refreshToken, profile, done) {
  ...
}
));
```

- Authenticating Users:

```
router.get('/facebook', passport.authenticate('facebook'), function(req, res){});

router.get('/facebook/callback', function(req,res,next){
  passport.authenticate('facebook', function(err, user, info) {
    ...
  }) (req, res, next);
});
```

# Exercise: Using OAuth with Passport and Facebook

- Configure your server to support user authentication based on OAuth providers
- Use Passport OAuth support through the passport-facebook module to support OAuth based authentication with Facebook for your users