OAuth 2.0 based Authentication Module

Written in Typescript for AngularJS

Anand Singh Kunwar
(13110)

under the guidance of
Dr. Prabhakar T.V.

Outline

Outline

What is OAuth 2.0?

- OAuth 2.0 is the industry-standard protocol for authorization
- The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service:
  - either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service
  - or by allowing the third-party application to obtain access on its own behalf

Outline

Definitions

- Resource Owner: Entity that has the user's information. Can be:
  - User itself
  - Some app like Facebook

Definitions

- Resource Owner: Entity that has the user's information. Can be:
    - User itself
    - Some app like Facebook
- Client: App making protected resource requests on behalf of the resource owner

Definitions

- Resource Owner: Entity that has the user's information. Can be:
  - User itself
  - Some app like Facebook
- Client: App making protected resource requests on behalf of the resource owner
- Authorization Grant: A means to authorization

Definitions

- Resource Owner: Entity that has the user's information. Can be:
  - User itself
  - Some app like Facebook
- Client: App making protected resource requests on behalf of the resource owner
- Authorization Grant: A means to authorization
- Authorization Server: Entity that verifies the authorization grant and issues access token

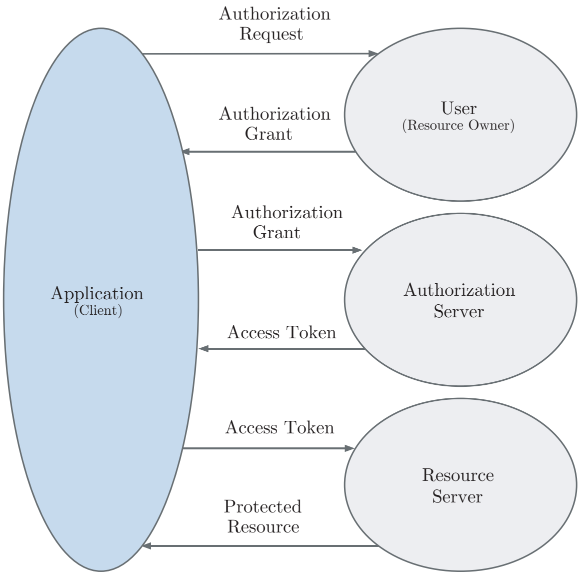
Definitions

- Resource Owner: Entity that has the user's information. Can be:
  - User itself
  - Some app like Facebook
- Client: App making protected resource requests on behalf of the resource owner
- Authorization Grant: A means to authorization
- Authorization Server: Entity that verifies the authorization grant and issues access token
- Access Token: Token used for getting hold of a protected resource

Definitions

- Resource Owner: Entity that has the user's information. Can be:
    - User itself
    - Some app like Facebook
- Client: App making protected resource requests on behalf of the resource owner
- Authorization Grant: A means to authorization
- Authorization Server: Entity that verifies the authorization grant and issues access token
- Access Token: Token used for getting hold of a protected resource
- Resource Server: Entity that holds the protected resource

OAuth 2.0 Flow



What is AngularJS?

- HTML wasn't built for dynamic views
- AngularJS tries to separate DOM manipulation(UI Logic) from business logic by using data binding
- MVC/MVVM Pattern Based
- Frontend web application framework based on Javascript actively Maintained by Google

Outline

Problem with Existing Libraries

- Satelizer: The most popular library for oAuth 2.0 based Token Authentication
  - Does not have an option for saving Refresh Tokens
  - Due to support for JSON Web Tokens(relatively new) a lot of backend libraries aren't compatible with it
- angular-oauth2: A recent one
  - Does not have the option for Social Authentication
  - Not actively maintained

Outline

Possible Solutions

- Mix and match libraries and find which is compatible with which.
    - Tedious as most of the AngularJS authentication libraries aren't built to be used with other authentication libraries.
- ngOAuth2: A library developed this semester that solves most of the problems mentioned previously
    - Implemented in Typescript for easy upgrades to newer versions of Angular
    - Very Recent

Outline

Introduction

- AngularJS library for OAuth 2.0 based Token Authentication written in Typescript
- Events support on a success/error authentication/authorization response
- Social Login support

Logical View

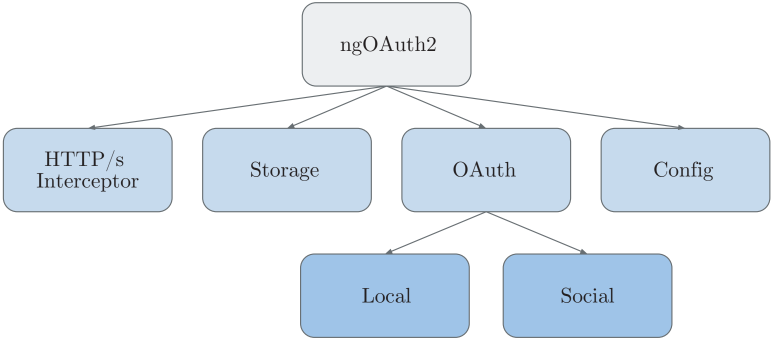
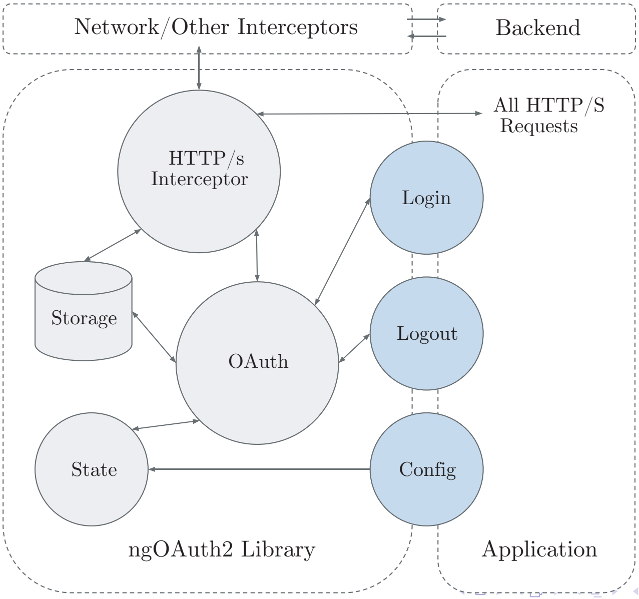


Figure: Logical view for ngOAuth2

Process View



Outline

Design Choices

- Typescript, Javascript(ES6 (2015), ES5(2012)), CoffeeScript? Typescript

Design Choices

- Typescript, Javascript(ES6 (2015), ES5(2012)), CoffeeScript? Typescript
- npm scripts vs gulp scripts? gulp scripts

Design Choices

- Typescript, Javascript(ES6 (2015), ES5(2012)), CoffeeScript? Typescript
- npm scripts vs gulp scripts? gulp scripts
- Regular Tokens vs JWT? Regular Tokens for now, JWT, a future support

Design Choices

- Typescript, Javascript(ES6 (2015), ES5(2012)), CoffeeScript? Typescript
- npm scripts vs gulp scripts? gulp scripts
- Regular Tokens vs JWT? Regular Tokens for now, JWT, a future support
- LocalStorage, SessionStorage, Cookies? Plan to support all, currently support LocalStorage and SessionStorage

Other Technologies Used

- UglifyJS: To uglify and minify JS for smaller size footprint
- Browserify: To convert require('modules') into browser compatible bundled dependency
- Yeoman: To generate the initial library template
- TSLint: For checking TypeScript code for readability, maintainability, and functionality errors.

Directory Structure

- src/
  - ngOAuth2/
    - constants/
    - interfaces/
    - providers/
    - services/
  - main.ts
- dist/
- gulpfile.js
- package.json
- tslint.json
- ...

Future Work

- Support for Cookies
- Support JWT
- Support for social platforms other than Facebook and Google
- Future Angular versions support

Demo

Thank You!