Date : 23-Jun-21
Spring Boot 9AM
Mr. RAGHU

OAuth 2.x [Open Authorization]

*) OAuth [Open Authorization] it is API Standard used for client applications
and EndUsers Authorized by 'Authorization and Resource Server'.

*) OAuth contains 3 components
a. EndUser
b. Client Application
c. Auth and Resource Server.

*) EndUser: persons who need services from client application.

*) Client Application: Apps that gives services, ex:
bookmyshow, carewale, zomato, redbus, quora, ...etc
These are called business applications.

*) Auth and Resource Server : A Server that validates User + Client App
and gives user data to Client App.

Ex: Facebook, Google, Github, LinkedIn ,...etc

*) This concept is used at day to day business applications.
Not recomanded in Finance based application.

*) It is also called as One time authentication [SSO - Single Sign On]
*) At a time for one EndUser/Browser only one OAuth service is provided.
ie AJAY--FACEBOOK, (AJAY CAN NOT USE GOOGLE,
until delete account in Client App created using facebook)

----------Workflow----------

1. Register ClientApp with Auth and Resource Server
2. Register EndUser with Auth and Resource Server
3. Validate Client/EndUser with Auth and Resource Server

---

1. Register ClientApp with Auth and Resource Server
Facebook as Authorization and Resource Server

*) Goto : https://developers.facebook.com/
*) Click on MyApps (Top right corner)
*) Click on Create App > Choose Consumer (Facebook Login) > Continue
*) Enter name : MY-TEST-BMS > Create App
*) Goto Settings > Basics
*) Copy
appId : 326340535741023
secret: a3f55737465cd00e58c496d5db6dff16

---
2. Register EndUser with Auth and Resource Server

Ex: ajay@gmail.com, create new FB account, Login, Post details, update profile
..etc

---
3. Validate Client/EndUser with Auth and Resource Server (Login Process)

S#1 User went to Client Application(BMS) and clicked on Login using Facebook
S#2 Client App is asking for user permission (Grant)
S#3 First, User has to Login with Auth and Resource Server (ex: Facebook Login)
S#4 then, click on button 'CONTINUE AS RAGHU".
S#5 User Grant(FB Login + Continue) is given to ClientApp(BMS)
S#6 Client makes request to Auth Server with 'ClientId, secret and User Grant'
S#7 Auth server validates Client Info with its database
S#8 If Valid, it provide AccessToken (Client + User) to client App
which is valid for Client And Specific user only.
S#9 Client App, uses Access Token and try to fetch user data from Resource Server.
S#10 Resource server validate token and fetch User data from DB
S#11 Resource server returns User data to client app as response
S#12 Client App may store user data in its DB.
S#13 Client says User Login success, access other services!!
S#14 User can see Home page, execute operations(book, view, cancel..etc)

======Backend application coding part==========
S#1 Create Spring Boot Starter Project
Name : SpringBoot2OAuthSsoEx
Dep : Web, Security, oauth2 Client

S#2 application.yml

```
server:
  port: 8081
spring:
  application:
    name: oauth-fb-login
  security:
    oauth2:
      client:
        registration:
          facebook:
            client-id: 326340535741023
            client-secret: a3f55737465cd00e58c496d5db6dff16
```

S#3 Security Config

```
package in.nareshit.raghu.config;
```

```
import org.springframework.context.annotation.Configuration;
import
org.springframework.security.config.annotation.authentication.builders
.AuthenticationManagerBuilder;
import
org.springframework.security.config.annotation.web.builders.HttpSecuri
ty;
import
org.springframework.security.config.annotation.web.configuration.WebSe
curityConfigurerAdapter;

@Configuration
public class SecurityConfig extends WebSecurityConfigurerAdapter {

        protected void configure(AuthenticationManagerBuilder auth)
throws Exception {

auth.inMemoryAuthentication().withUser("SAM").password("
{noop}SAM").authorities("ADMIN");
        }

        protected void configure(HttpSecurity http) throws Exception {
                http.authorizeRequests()
                .antMatchers("/","/login","/home").permitAll()
                .anyRequest().authenticated()
                .and().formLogin()
                .and().oauth2Login()
                .and().csrf().disable() ;
        }
}

S#4 RestController
package in.nareshit.raghu.rest;

import java.security.Principal;

import org.springframework.security.core.Authentication;
import
org.springframework.security.core.context.SecurityContextHolder;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class UserRestController {

        @GetMapping("/home")
        public String showHome() {
                return "WELCOME TO ALL!!";
        }

        @GetMapping("/data")
        public String showSecure() {
                return "WELCOME SECURED PAGE";
        }

        @GetMapping("/user")
```

```
        public Authentication showUser(Principal p) {
                System.out.println(p.getClass().getName());
                System.out.println("Current user => " + p.getName());
                return
SecurityContextHolder.getContext().getAuthentication();
                //return p;
        }
}
```

For FB:
https://developers.facebook.com/
https://www.facebook.com/settings?tab=applications&ref=settings

==========
=
For Google
#1 Goto https://console.cloud.google.com/
#2 Goto Home > Dashboard
#3 Click on create Project
#4. Enter project name and create button
  ex: test-oauth-new-app
#5 Goto : OAuth consent screen
#6 Choose External And create
#7 Create app with details (in 4 steps)
  Enter : test-app > save and continue
#8 Click on Credentials> create Credentials > Choose Oauth Client Id
  enter Details
    Type : webapplication
    name : sample
    URI : http://localhost:8081
    redirect URI : http://localhost:8081/oauth2/callback/google
> Finish

Copy Id, secret

ClientId: 998952254332-
p3a51bmik5dbq288tdo47j6raaufi1kp.apps.googleusercontent.com
Secret: EUqdnKVtpGdXHvtDttPtbOVo

For Github
S#1 goto https://github.com/settings/developers

191ff43e15642c6c0e7f
3207e6ee75375c38490d1229947252c1a30e73d9
http://localhost:8081
http://localhost:8081