15 Spring Security OAuth2登录（三方登录）

Spring Security OAuth2登录（三方登录）

众所周知，OAuth2有三个角色：授权服务器(Authorization Server、AS)、资源服务器(Resource Server、RS)、客户端(Client)。

客户端可以通过获取授权服务器的授权，以此获取资源服务器的资源。

正是这样，OAuth2可以用来实现 OAuth2登录（三方登录）。在OAuth2登录的场景中，用户信息则是"资源"！

实现原理就是：Client获取用户授权，得到令牌，通过令牌获取用户信息（资源）。再在本地构建用户登录认证信息，维持用户会话状态，以此达到登录的目的。

示例

下文通过 SpringSecurity 和 Gitee提供的授权功能 实现一个OAuth2登录的示例。

我们项目就相当于是客户端，通过获取gitee用户授权，得到授权码，进而得到令牌，通过令牌获取gitee用户信息，在我们项目构建用户登录认证信息，维持用户会话状态，以此达到登录的目的。

Gitee OAuth2 接入文档

1. 创建 SpringBoot 项目

一个 pom，一个 Application 启动类，相信难不倒你，跳过。

本文使用 SpringBoot 2.3.1.RELEASE

2. 引入依赖

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-oauth2-client</artifactId>

</dependency>

</dependencies>

3. 编写配置

SecurityConfiguration

@Configuration

@EnableWebSecurity

public class SecurityConfiguration extends WebSecurityConfigurerAdapter {

@Override

public void configure(WebSecurity web) throws Exception {

web

//忽略权限验证

.ignoring()

.antMatchers("/error")

.antMatchers("/favicon.ico")

;

}

@Override

protected void configure(HttpSecurity http) throws Exception {

http

.authorizeRequests()

.anyRequest().authenticated()

.and()

// 支持 OAuth2 登录

.oauth2Login()

.and()

.csrf().disable()

;

}

}

application.yml

server:

port: 8080

spring:

security:

oauth2:

client:

registration: # 定义应用信息

gitee-app: # 应用的唯一标识

clientName: gitee的client

client-id: b4faeee3e90bc756d669e252e888154fbd629f62e8a22b4a3c6b14b5d57ef84f

client-secret: 00b159cb0b1651e1753c48a72e89dc7e58db740c3d8e0a88a9fd6c65153ea3f0

clientAuthenticationMethod: basic

authorizationGrantType: authorization\_code

redirectUri: '{baseUrl}/{action}/oauth2/code/{registrationId}'

provider: gitee # 指向 provider 中的唯一标识，默认值是 registrationId

provider: # 定义授权服务器信息

gitee: # 授权服务器的唯一标识

authorizationUri: https://gitee.com/oauth/authorize

tokenUri: https://gitee.com/oauth/token

userInfoUri: https://gitee.com/api/v5/user

userInfoAuthenticationMethod: GET

userNameAttribute: login

logging:

level:

root: info

static/index.html

写个页面放到 src/main/resources/static/index.html，便于观察结果

<html>

<head>

<meta charset="utf-8">

<title>首页</title>

</head>

<body>

<h5>登录成功</h5>

<div>

<form method="get" action="/logout">

<button type="submit">Logout</button>

</form>

</div>

<div>

<a href="/current">/current</a>

</div>

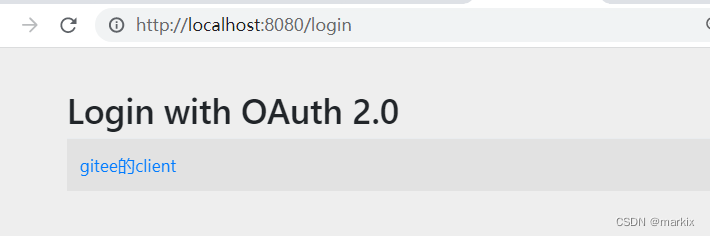
</body>

</html>

4. 启动

5. 测试

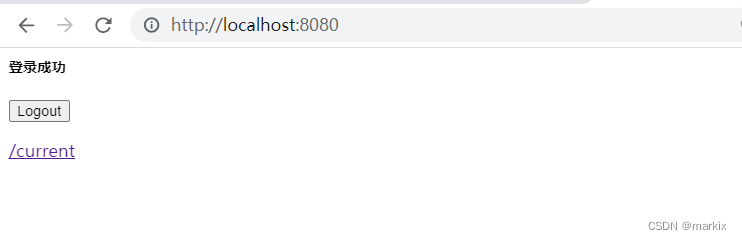
访问首页



点击 OAuth2登录（gitee的client），跳转到gitee用户授权页面，点击“同意授权”



登录成功，跳到我们写的index.html



到这里整个OAuth2登录流程结束了。

当然，你还可以点击 /current，可以看到gitee信息是如何封装成认证信息的。

{"authorities":[{"authority":"ROLE\_USER","attributes":{"id":5624461,"login":"markix","name":"Markix","avatar\_url":"https://gitee.com/assets/no\_portrait.png","url":"https://gitee.com/api/v5/users/markix","html\_url":"https://gitee.com/markix","remark":"","followers\_url":"https://gitee.com/api/v5/users/markix/followers","following\_url":"https://gitee.com/api/v5/users/markix/following\_url{/other\_user}","gists\_url":"https://gitee.com/api/v5/users/markix/gists{/gist\_id}","starred\_url":"https://gitee.com/api/v5/users/markix/starred{/owner}{/repo}","subscriptions\_url":"https://gitee.com/api/v5/users/markix/subscriptions","organizations\_url":"https://gitee.com/api/v5/users/markix/orgs","repos\_url":"https://gitee.com/api/v5/users/markix/repos","events\_url":"https://gitee.com/api/v5/users/markix/events{/privacy}","received\_events\_url":"https://gitee.com/api/v5/users/markix/received\_events","type":"User","blog":null,"weibo":null,"bio":"123123","public\_repos":10,"public\_gists":0,"followers":0,"following":0,"stared":0,"watched":12,"created\_at":"2020-01-10T15:59:58+08:00","updated\_at":"2022-10-28T11:41:46+08:00","email":null}},{"authority":"SCOPE\_user\_info"}],"details":{"remoteAddress":"0:0:0:0:0:0:0:1","sessionId":"28D8B4350D887507368CFDF67B9B8FAD"},"authenticated":true,"principal":{"authorities":[{"authority":"ROLE\_USER","attributes":{"id":5624461,"login":"markix","name":"Markix","avatar\_url":"https://gitee.com/assets/no\_portrait.png","url":"https://gitee.com/api/v5/users/markix","html\_url":"https://gitee.com/markix","remark":"","followers\_url":"https://gitee.com/api/v5/users/markix/followers","following\_url":"https://gitee.com/api/v5/users/markix/following\_url{/other\_user}","gists\_url":"https://gitee.com/api/v5/users/markix/gists{/gist\_id}","starred\_url":"https://gitee.com/api/v5/users/markix/starred{/owner}{/repo}","subscriptions\_url":"https://gitee.com/api/v5/users/markix/subscriptions","organizations\_url":"https://gitee.com/api/v5/users/markix/orgs","repos\_url":"https://gitee.com/api/v5/users/markix/repos","events\_url":"https://gitee.com/api/v5/users/markix/events{/privacy}","received\_events\_url":"https://gitee.com/api/v5/users/markix/received\_events","type":"User","blog":null,"weibo":null,"bio":"123123","public\_repos":10,"public\_gists":0,"followers":0,"following":0,"stared":0,"watched":12,"created\_at":"2020-01-10T15:59:58+08:00","updated\_at":"2022-10-28T11:41:46+08:00","email":null}},{"authority":"SCOPE\_user\_info"}],"attributes":{"id":5624461,"login":"markix","name":"Markix","avatar\_url":"https://gitee.com/assets/no\_portrait.png","url":"https://gitee.com/api/v5/users/markix","html\_url":"https://gitee.com/markix","remark":"","followers\_url":"https://gitee.com/api/v5/users/markix/followers","following\_url":"https://gitee.com/api/v5/users/markix/following\_url{/other\_user}","gists\_url":"https://gitee.com/api/v5/users/markix/gists{/gist\_id}","starred\_url":"https://gitee.com/api/v5/users/markix/starred{/owner}{/repo}","subscriptions\_url":"https://gitee.com/api/v5/users/markix/subscriptions","organizations\_url":"https://gitee.com/api/v5/users/markix/orgs","repos\_url":"https://gitee.com/api/v5/users/markix/repos","events\_url":"https://gitee.com/api/v5/users/markix/events{/privacy}","received\_events\_url":"https://gitee.com/api/v5/users/markix/received\_events","type":"User","blog":null,"weibo":null,"bio":"123123","public\_repos":10,"public\_gists":0,"followers":0,"following":0,"stared":0,"watched":12,"created\_at":"2020-01-10T15:59:58+08:00","updated\_at":"2022-10-28T11:41:46+08:00","email":null},"name":"markix"},"authorizedClientRegistrationId":"gitee-app","credentials":"","name":"markix"}

升级版示例

通过绑定与解绑方式实现三方登录（企业级实现方式），详见案例源码

<https://gitee.com/markix/spring-security-federal-login>

大前提

三方用户需要先和本地用户绑定，三方登录时，实际上是找到对应的本地用户进行后台登录。所以，三方用户登录实际也是用户登录。（认证实体是本地用户） 三方用户和本地用户通过关联表记录绑定关系。

功能点

表单登录

三方帐号和本地用户绑定、解绑

三方帐号登录