OAUTH2

1. 目的

能够让客户端(即WEB应用中的前端服务)或者一些第三方应用能够凭借用户授权后的access\_token去访问属于这个用户的一些资源。只要用户授权一次后，在该token有效期内，客户端可以凭借该access\_token去访问任意的属于该用户的资源。

这样的好处是，用户不需要把自己的用户名和密码给客户端或者第三方应用，也不需要每次访问不同服务器时都要再登录验证一遍，这样的好处就是很方便用作单点登录的设计思想。

1. 模式以及设计原则以及内容

OAUTH2是一种协议，一种思想，它不包括具体的技术实现，比如access\_token它具体应该是怎么构成的？它具体又是如何验证的？这些都没有具体规定。所以也都有相较于不同的技术实现，比如JWT来具体实现access\_token, SpringSecurity oauth2在sprigsecurity基础上具体实现oauth2, oauth2根据流程大概将内容划分为下列这些部分

1. 客户端或第三方应用

它是需要获取到数据的，可以理解为是服务的消费者，它需要从资源服务器那儿去用户的资源再展示，它可以是前后端分离情况下的前端服务，或者是一些第三方应用需要调用本应用服务的一些数据。注意，不能把它理解为浏览器!!!

2) 授权服务器

这个主要是供用户认证用的，当客户端访问资源服务器没带token或者token失效后，就被转到授权服务器，提醒用户输入密码进行验证，验证完之后会回传用户一个code, 之后用户再把这个code给客户端，客户端凭借这个code向授权服务再获取access\_token，这个获取过程不经过用户浏览器且通过SSL加密，保证了安全性，最后给客户端一个token,客户端拿着这个token再去访问资源服务器，经过资源服务器验证后，把资源拿给客户端。之后就一直拿着这个token去访问资源，直到失效。

3) 资源服务器

用于存放用户资源的服务器，可根据客户端传递过来的access\_token自行校验，验证其是否有效。

1. Oauth2验证的具体流程

这里仅就授权码模式做出介绍

如下图



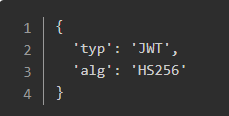
1. 针对access\_token的具体技术实现

JWT协议格式

JWT是针对oauth2中提出的token的协议规定，它把token分成了三部分header playload 以及 secret 三部分。三部分加密后用点隔开，如xxx.xxxxx.xxxxx 下面分开介绍这三部分

1. header

包括两块，申明以及加密算法，示例如下图



申明一般是JWT

加密算法则是具体加密算法的内容

1. payload

包括标准中注册的申明，公有的声明，私有的申明，不重要，略过

1. secret

包括三部分 加密后的header,加密后的payload，再加盐secret。资源服务器就是用这个进行验证，用对称公钥或者私钥进行解密，如果和之前相符，且加盐一样，则有效，所以这个secret一定要资源服务器自己保存，不能泄露给客户端，否则客户端将自己签发access\_token.