

金三银四Java面试突击专题

安全验证篇

=== 图灵： 楼兰 ===

一、什么是认证和授权？如何设计一个权限认证框架？

认证：就是对系统访问者的身份进行确认。 用户名密码登录、 二维码登录、手机短信登录、指纹、刷脸。。。

授权：就是对系统访问者的行为进行控制。授权通常是在认证之后，对系统内的用户隐私数据进行保护。后台接口访问权限、前台控件的访问权限。

RBAC模型： 主体 -> 角色 -> 资源 -> 访问系统的行为。

认证和授权也是对一个权限认证框架进行扩展的两个主要的方面。

二、Cookie和Session有什么区别？

三、如果没有Cookie,Session还能进行身份验证吗？

当服务器tomcat第一次接收到客户端的请求时，会开辟一块独立的session空间，建立一个session对象，同时会生成一个session id，通过响应头的方式保存到客户端浏览器的cookie当中。以后客户端的每次请求，都会在请求头部带上这个session id，这样就可以对应上服务端的一些会话的相关信息，比如用户的登录状态。

如果没有客户端的Cookie，Session是无法进行身份验证的。

当服务端从单体应用升级为分布式之后，cookie+session这种机制要怎么扩展？

1、session黏贴：在负载均衡中，通过一个机制保证同一个客户端的所有请求都会转发到同一个tomcat实例当中。问题：当这个tomcat实例出现问题之后，请求就会被转发到其他实例，这时候用户的session信息就丢了。

2、session复制：当一个tomcat实例上保存了session信息后，主动将session复制到集群中的其他实例。问题：复制是需要时间的，在复制过程中，容易产生session信息丢失。

3、session共享：就是将服务端的session信息保存到一个第三方中，比如Redis。

四、什么是CSRF攻击？如何防止？

CSRF：Cross Site Request Forgery 跨站请求伪造

一个正常的请求会将合法用户的session id保存到浏览器的cookie。这时候，如果用户在浏览器中打开另一个tab页，那这个tab页也是可以获得浏览器的cookie。黑客就可以利用这个cookie信息进行攻击。

攻击过程：

1、某银行网站A可以以GET请求的方式发起转账操作。 www.xxx.com/transfer.do?accountNum=100&money=1000 accountNum表示目标账户。这个请求肯定是需要登录才可以正常访问的。

2、攻击者在某个论坛或者网站上，上传一个图片，链接地址是 www.xxx.com/transfer.do?accountNum=888&money=10000 其中这个accountNum就是攻击者自己的银行账户。

3、如果有一个用户，登录了银行网站，然后又打开浏览器的另一个tab页，点击了这个图片。这时，银行就会受理到一个带了正确cookie的请求，就会完成转账。用户的钱就被盗了。

CSRF方式方式：

1、尽量使用POST请求，限制GET请求。POST请求可以带请求体，攻击者就不容易伪造出请求。

2、将cookie设置为HttpOnly：`response.setHeader("Set-Cookie","cookieName=cookieValue;HttpOnly")`。

3、增加token;

在请求中放入一个攻击者无法伪造的信息，并且该信息不存在于cookie当中。

这也是Spring Security框架中采用的防范方式。

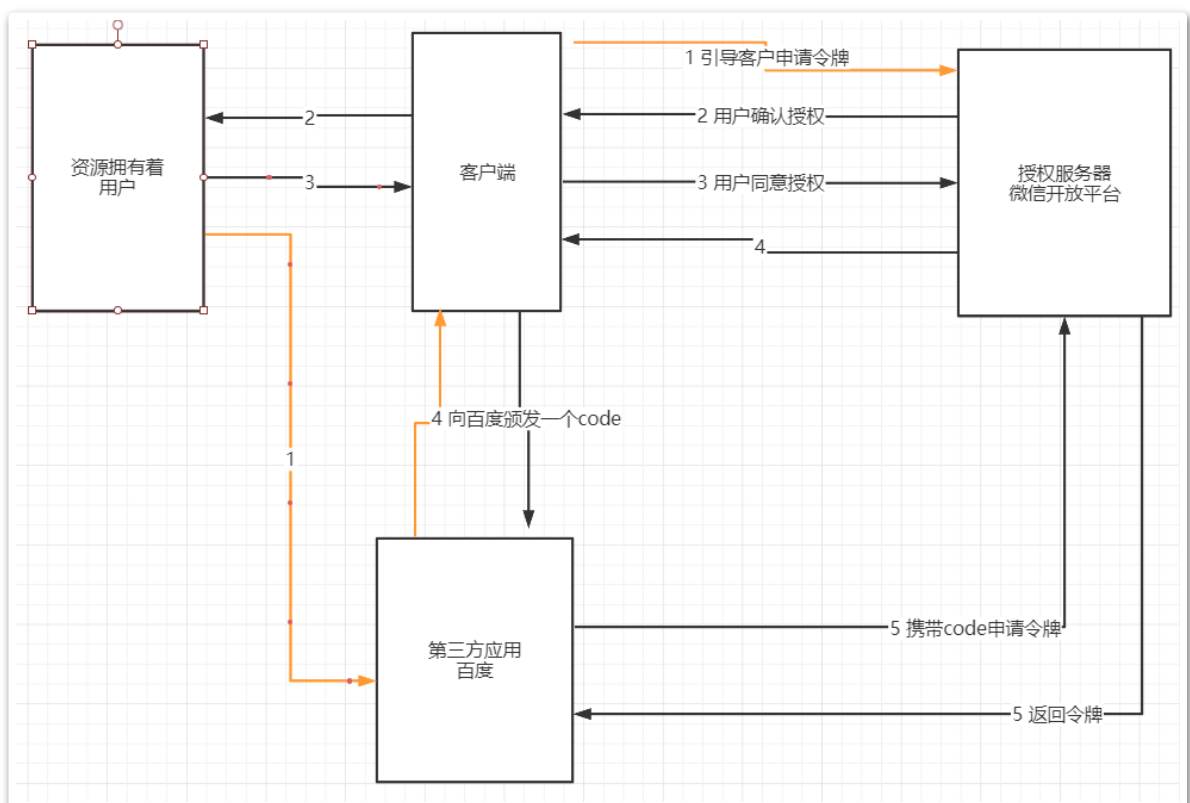
五、什么是OAuth2.0协议？有哪几种认证方式？什么是JWT令牌？和普通令牌有什么区别？

OAuth2.0是一个开放标准，允许用户授权第三方应用程序访问他们存储在另外的服务提供者上的信息，而不需要将用户名和密码提供给第三方应用或分享他们数据的所有内容。

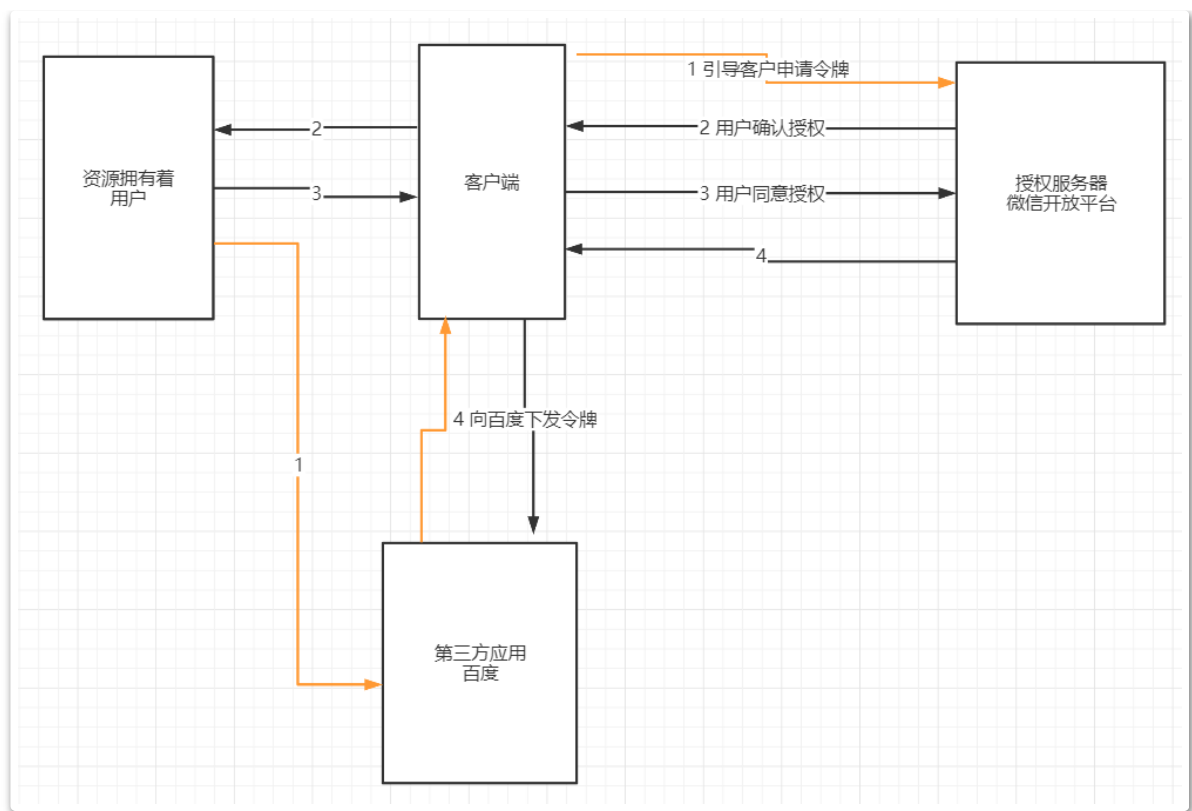
OAuth2.0协议的认证流程，简单理解，就是允许我们将之前的授权和认证过程交给一个独立的第三方进行担保。

OAuth2.0协议有四种认证方式：

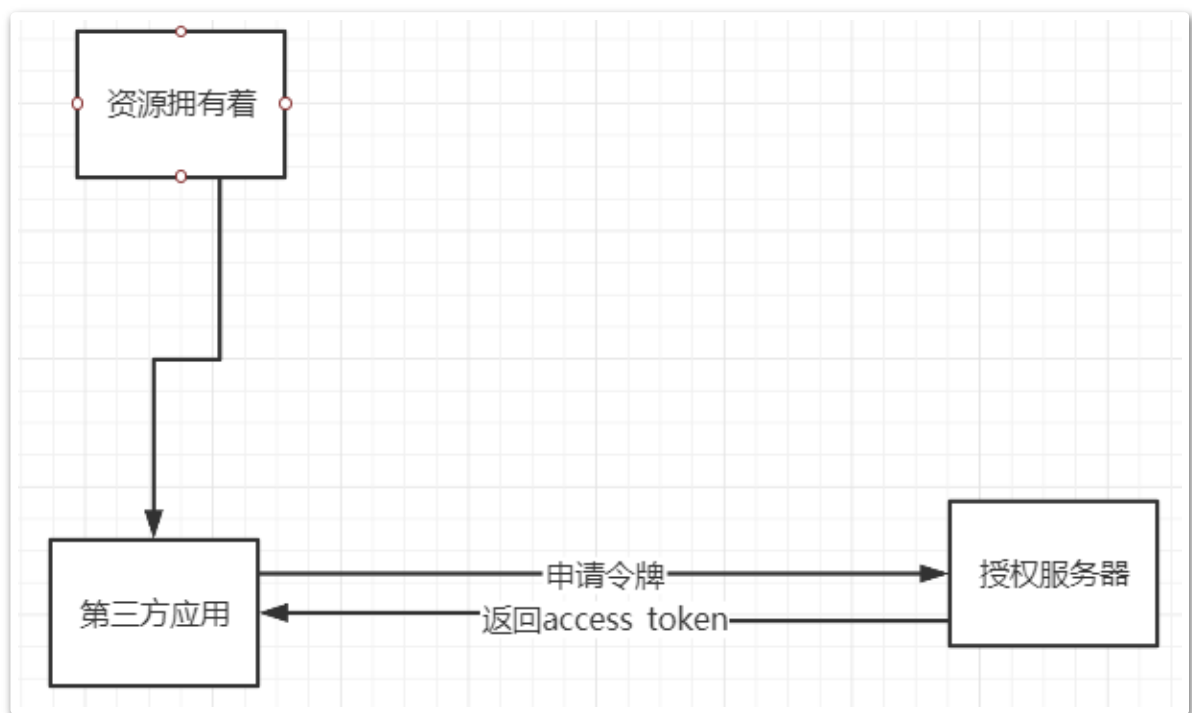
1、授权码模式



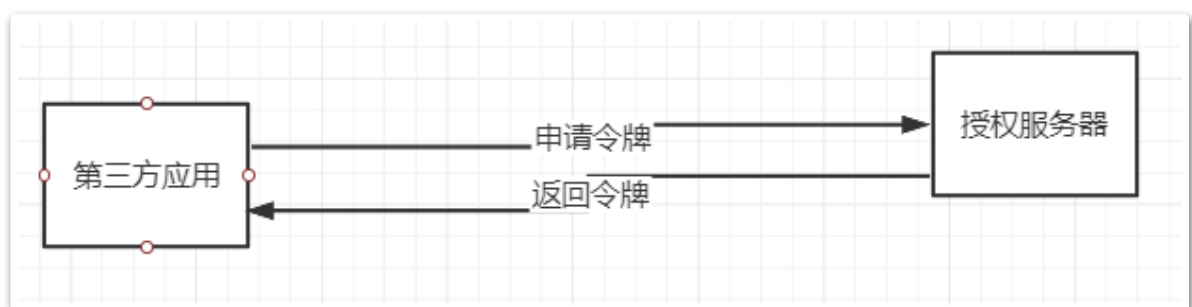
2、简化模式



3、密码模式



4、客户端模式



在梳理OAuth2.0协议流程的过程中，其实有一个主线，就是三方参与者之间的信任程度。

普通令牌： b9f2eaa1-8715-4f03-86c7-06bf757a5f7c

普通令牌只是一个随机的字符串，没有特殊的意义。这就意味着，当客户带上令牌去访问应用的接口时，应用本身无法判断这个令牌是否正确，他就需要到授权服务器上去判断令牌是否有效。在高并发场景下，检查令牌的网络请求就有可能成为一个性能瓶颈。

改良的方式就是JWT令牌。将令牌对应的相关信息全部冗余到令牌本身，这样资源服务器就不再需要发送请求给授权服务器去检查令牌了，他自己就可以读取到令牌的授权信息。JWT令牌的本质就是一个加密的字符串！！

JWT令牌：

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiIsc2FsYXJ5Il0sInVzZXJfbmFtZSI6ImFkbWlulwiw2NvcGUiOlsiYWxsIl0sImV4cCI6MTYxNjY3MjM3OCwiYXV0aG9yaXRpZXMiOlsibW9iaWxliiwic2FsYXJ5Il0sImp0aSI6ImI1MDg2OWE0LTlzZmEtNDg2Yy1hZGJlTlJNTlImMjRiMDY4YSIsImNsaWVudF9pZCI6ImMxIn0.tJ5d7RBKpj8d6w7826OqS6_2pDf_ZXvwkJHMO2uPVAg

六、什么是SSO？与OAuth2.0有什么关系？

OAuth2.0的使用场景通常称为联合登录。 **一处注册，多处使用。**

SSO Single Sign On 单点登录。 **一处登录，多处同时登录。**

SSO的实现关键是将Session信息集中存储。Spring Security

七、如何设计一个开放授权平台？

开放授权平台也可以按照认证和授权两个方向来梳理。

1、认证： 就可以按照OAuth2.0协议来规划认证的过程。

2、授权：首先需要待接入的第三方应用在开放授权平台进行注册，注册需要提供几个必要的信息 clintID, 消息推送地址，密钥(一对公私钥，私钥由授权平台自己保存，公钥分发给第三方应用)。

然后，第三方应用引导客户发起请求时，采用公钥进行参数加密，授权开放平台使用对应的私钥解密。

接下来：授权开放平台同步响应第三方应用的只是消息是否处理成功的结果。而真正的业务数据由授权开放平台异步推动给第三方应用预留的推送地址。

