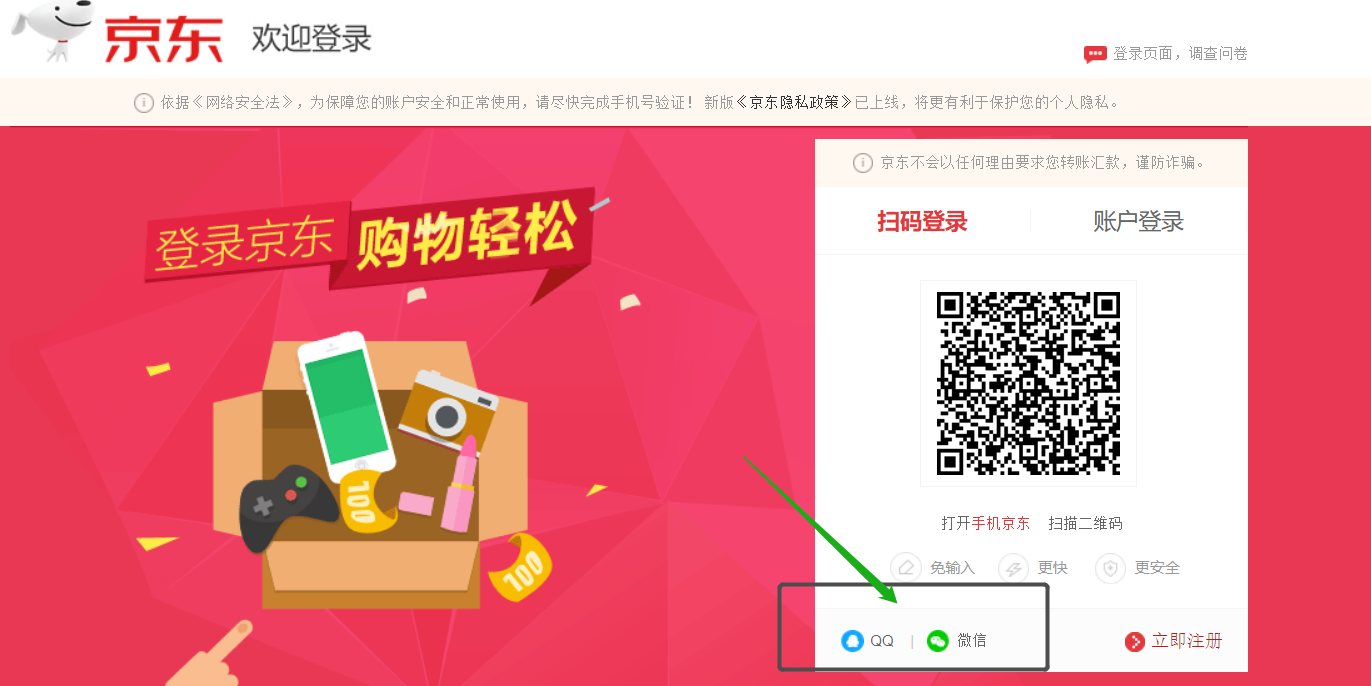
# Oauth2学习

**定义：**第三方认证技术方案最主要是解决认证协议的通用标准问题，因为要实现跨系统认证，各系统之间要 遵循一定的接口协议。只是一个协议，定义统一接口的协议。

重要：

下面是京东的登录页面，可以看到他是支持微信或者qq一键登录的



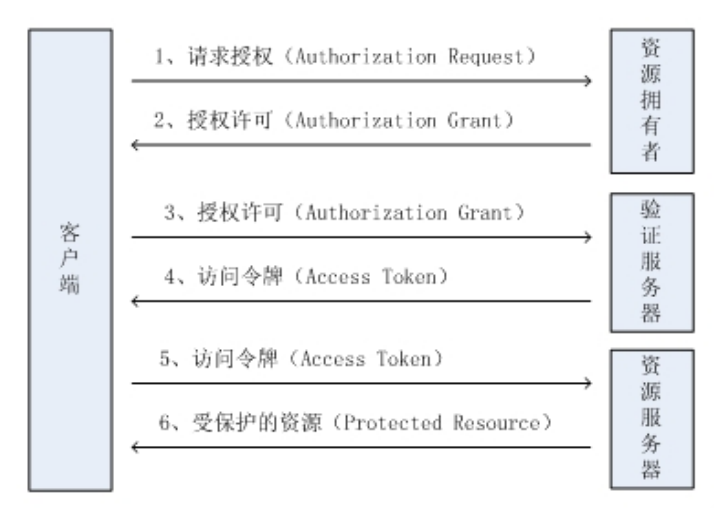
当我点击微信之后，会出现一个微信扫描二维码，如下：

其实，这就是要微信给授权。

然后需要用户扫码进行一系列的授权，就可以成功登录。



下面这个是oauth2的工作流程，是可以实现上面的那一系列操作的。



**描述流程：**

1. 客户端通过http请求访问资源拥有者(即用户)请求授权；
2. 资源拥有者给予客户端权限，即授权许可证；
3. 客户端拿着授权许可证再去请求认证服务器(即授权服务器)，请求认证授权；
4. 认证服务器给予客户端一个授权的访问令牌token；
5. 客户端拿着这个令牌去访问资源服务器；
6. 资源服务器通过token认证返回给客户端想要访问的资源；

**注意：*为了防止藕荷，认证服务器一般是单独的，和资源服务器不在一个系统。***

**角色**

**客户端**

本身不存储资源，需要通过资源拥有者的授权去请求资源服务器的资源，比如：Android客户端、Web客户端（浏览器端）、微信客户端等。

**资源拥有者**

通常为用户，也可以是应用程序，即该资源的拥有者。

**授权服务器（也称认证服务器）**

用来对资源拥有的身份进行认证、对访问资源进行授权。客户端要想访问资源需要通过认证服务器由资源拥有者授权后方可访问。

**资源服务器**

存储资源的服务器，比如，网站用户管理服务器存储了网站用户信息，网站相册服务器存储了用户的相册信息，微信的资源服务存储了微信的用户信息等。客户端最终访问资源服务器获取资源信息。

**常用术语**

**客户凭证(client Credentials) ：**客户端的clientId和密码用于认证客户

**令牌(tokens) ：**授权服务器在接收到客户请求后，颁发的访问令牌

**作用域(scopes) ：**客户请求访问令牌时，由资源拥有者额外指定的细分权限(permission)

**令牌类型**

**授权码 ：**仅用于授权码授权类型，用于交换获取访问令牌和刷新令牌

**访问令牌 ：**用于代表一个用户或服务直接去访问受保护的资源

**刷新令牌 ：**用于去授权服务器获取一个刷新访问令牌

**BearerToken ：**不管谁拿到Token都可以访问资源，类似现金

**Proof of Possession(PoP) Token ：**可以校验client是否对Token有明确的拥有权

特点

**授权模式**

授权模式一共有4种：

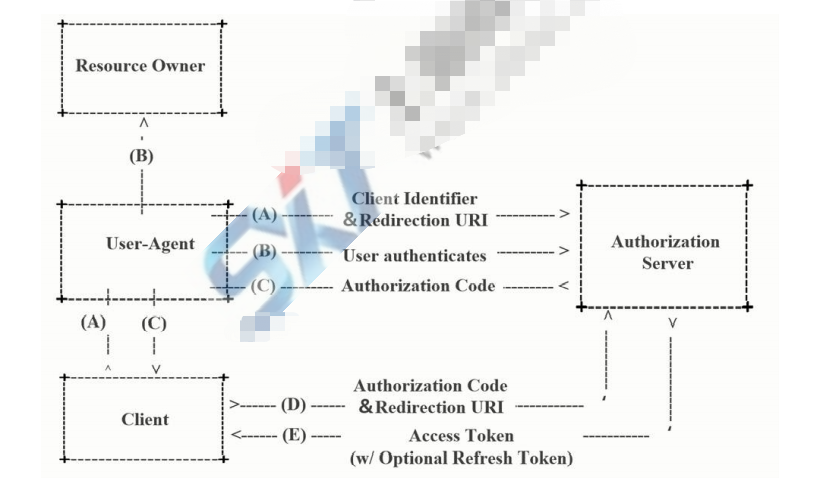
1、授权码模式（Authorization Code）----这是最安全，也是经常用到的。

2、简化授权模式（Implicit）

3、密码模式（Resource Owner PasswordCredentials）

4、客户端模式（Client Credentials）

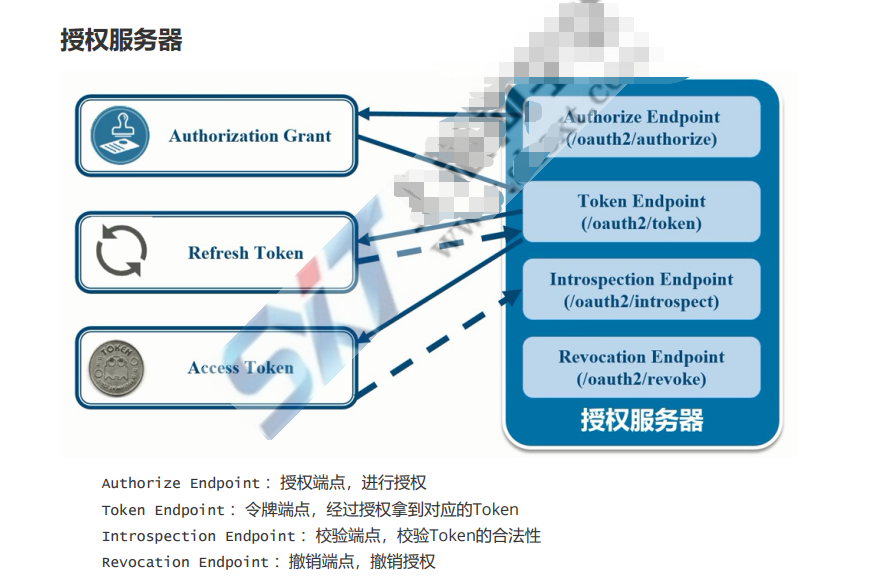
这里放一下**授权码模式**的流程图（摘自别人的资料）：



此流程的调用方式：**ABCDE**

1. 客户端（client）通过浏览器（User-Agent）去访问授权服务器（Authorization Server）。
2. 客户端会通过浏览器去访问资源拥有者（Resource Owner,即用户）进行授权认证。
3. 客户端会带着资源拥有者给的用户认证，通过浏览器去访问授权服务器，此时，授权服务器会给浏览器返回一个授权码。并且浏览器会将授权码返回给客户端。
4. 客户端带着授权码再去访问授权服务器，授权服务器通过校验会返回给客户端一个令牌（Access token和Refresh token）。
5. 最后客户端拿着这个令牌去访问资源服务器。

**Spring Security Oauth2**



重点需要理解上面的4个端点。