一旦CAS SERVER验证成功后，我们就会跳转到客户端中去。那么跳转到客户端去后，客户端要怎么获取用户信息呢？

其实验证成功，跳转客户端这个过程中，CAS SERVER 会返回登录的相关信息给客户端，客户端只要进行获取，就能知道登录的具体是哪个用户了。不过CAS 默认只返回用户账号给客户端，那么怎么定义CAS SERVER返回的信息呢？

先了解下有关相关的几个接口

Credentials

　　Credentials （org.jasig.cas.authentication.Credentials）接口，我们在上一篇其实有使用过，我们当时有用过一个叫 UsernamePasswordCredential 的类，就是实现了Credentials接口。这个接口是用来定义我们登录页上输入的认证信息的，比如用户名、密码、验证码等，可以理解为用户认证的相关凭据。

Principal

　　Principal (org.jasig.cas.authentication.principal.Principal) 接口，这个主要是用来保存用户认证后的用户信息，信息保存在一个Map中。

IPersonAttributeDao

　　IPersonAttributeDao (org.jasig.services.persondir.IPersonAttributeDao) 接口，这个是用来定义我们需要返回给客户端相关信息的接口，CAS SERVER 默认有提供许多实现，比如

• LdapPersonAttributeDao ：通过查询 LDAP 目录 ，来返回信息

• SingleRowJdbcPersonAttributeDao ： 通过JDBC SQL查询，来返回信息

等等，还有许多，大家可以参考源码中的实现，CAS SERVER提供了各种功能的实现，有时候我们可以直接使用这个现成的就行了。

PrincipalResolver

　　PrincipalResolver(org.jasig.cas.authentication.principal.PrincipalResolver) 接口,上面有说到 Credentials 是从登录页面上进行获取相关用户信息的。那么认证成功后，怎么把Credentials里面的信息转换到 Principal 中呢，这就是这个接口的作用了。由于认证本身是没有返回用户信息的，只是确定认证是通过还是没有通过。这时还要用到我们上面的IPersonAttributeDao 接口，在这接口中我们就可以定义我们需要返回的信息了。

这接口中有两个方法

• resolve ： 解析Credentials中的信息，返回 Principal 接口

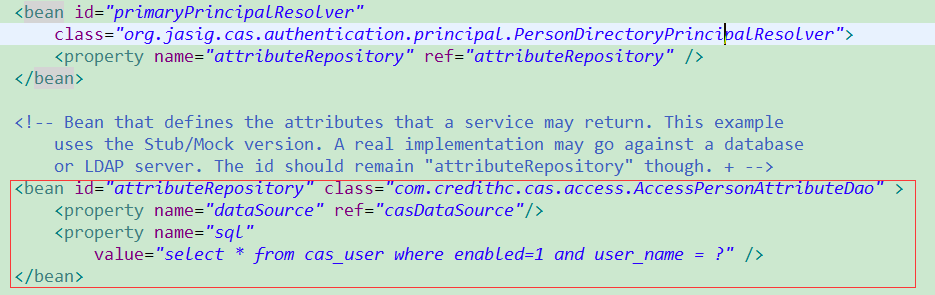
• supports ： 判断Credentials 是否支持 Principal 协议。

ps: 在3.x版本中没有 PrincipalResolver接口，对应的是CredentialsToPrincipalResolver， PrincipalResolver这个是在4.0版本中加入的，大家要注意。

## 流程

相关接口讲解后，大家应该对怎么返回信息有个大概的思路了。没错就是实现上面所说的 IPersonAttributeDao 、PrincipalResolver 接口 。下面根据代码讲解下具体的一个流程：

首先打开 deployerConfigContext.xml 文件，看下面的定义：



//PersonDirectoryPrincipalResolver 部分源码





具体流程：

1.从上面的deployerConfigContext.xml 配置我们可以看到，CAS 默认配置了一个叫做 PersonDirectoryPrincipalResolver 的类，在 这个类的 resolve 方法中有调用 extractPrincipalId 这个方法，这个方法传入一个Credentials 类型的参数，默认调用的是Credentials 的getId() 方法，CAS默认是返回用户的userName,即登录账号。不过getId() 这个方法的实现我们可以在上一章中指定的UsernamePasswordCredential 类中自定义，一般是定义成返回用户的userId或者其他唯一键，因为我们如果知道了用户的userId，那么就可以根据这个从数据库中查询中用户的一些具体信息了，进而就可以组成我们需要返回的信息。

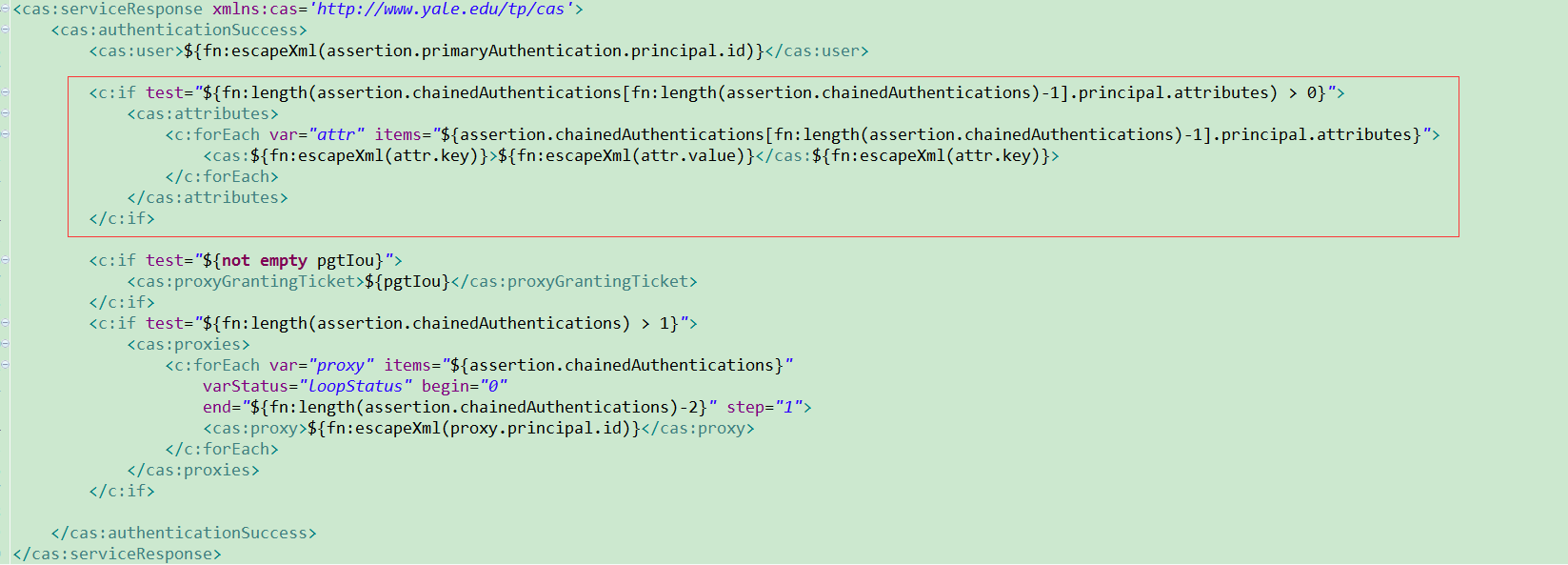
2. 继续往下看源码，接着在 PersonDirectoryPrincipalResolver 中有注入一个 attributeRepository 属性，这个就是上面的IPersonAttributeDao 接口，然后在resolve 方法中调用了 IPersonAttributeDao 接口 的getPerson方法，还传入了一个参数principalId，其实这个传入的参数就是我们上面 getId() 返回的值。

所以其实我们只要实现我们需要的 IPersonAttributeDao 就可以了。 下面给一个简单的IPersonAttributeDao 例子：



3. 修改完成后，我们还需要在

\WEB-INF\view\jsp\protocol\2.0\casServiceValidationSuccess.jsp）添加一段代码：



4.客户端接收信息，在java中可以通过下面的方式获取

AttributePrincipal principal = (AttributePrincipal) request.getUserPrincipal();

Map attributes = principal.getAttributes();

String xxx=attributes .get("xxx");