**非平台3.1和3.2需要使用此方法**

**登陆用户名和密码加密提交解决方案**

方案：

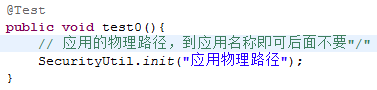
采用客户端加密，服务端解密的方式。嗯~~ 很简短，流程就是这样

方法：

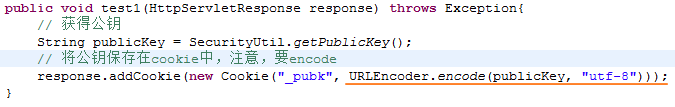
1. 将jar文件放到lib目录
2. 将js文件引入到登陆页面的jsp中
3. 去掉本来登录页中的用户名密码input的name（这样就不会提交input的值）
4. 在登陆jsp中建立用户名、密码隐藏域input，input的name为登陆程序需要接收的name（也就是3中去掉的name用在这里的隐藏域input中）
5. 在登陆前，用js加密3中的用户名密码，放到对应的4中的隐藏域中
6. 在java程序中将获取到的加密的用户名密码，用jar里面的方法解密成明文就可以了

编码：

1. Jar初始化，在使用加密之前我们先要对jar进行初始化



1. 在进入登陆页时，需要使用jar中的方法获取公钥encode之后并存储在cookie中。Js加密的时候要用。



Key必须叫\_pubk

建议cookie保存 60 \* 60 \* 24 \* 365（一年）

1. 登陆页面嵌入2个js

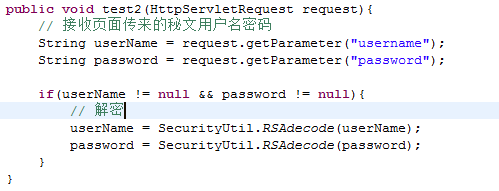
jsencrypt.min.js

rsa\_util.js

1. 在登录页面建立用户名密码隐藏域，并且去掉原用户名密码input的name。在提交前将用户名密码的明文进行js加密放入隐藏域中。这时候就可以提交了



1. 服务器端解密，调用jar里面的方法就可以解密成明文



注意：

依赖的js和jar都在文件夹里面，请根据项目需要添加