Spring security3 在BPM3 平台中功能实现

# 一个用户在同一个时间点只允许登录一次

## 简要说明

在系统登录的时候，我们要求一个帐号在系统中只能登录一次。比如用户admin从A点登录，另外一个人从B点登录，这个时候A登录的用户自动退出系统，并且通知用户已从另外的系统登录。

## 实现方法

修改spring security 配置文件。

<security:session-management invalid-session-url="/login.jsp">

<security:concurrency-control error-if-maximum-exceeded="false" max-sessions="1" expired-url="/login.jsp"/>

</security:session-management>

配置说明：

error-if-maximum-exceeded：

设置成false表示如果用户登录超过同时登录的次数后，将之前的用户踢出去，系统可以登录。

设置为true则抛出错误，不能登录。

max-sessions：设置为1，表示一个帐号在某个时刻只能登录一次。

invalid-session-url：表示如果登录下一个用户登录后，前一个登录用户跳转到的地址。

### 直接使用spring的过滤器。

直接使用spring的过滤器。

Spring 实现有个过滤器叫做org.springframework.security.web.authentication. UsernamePasswordAuthenticationFilter

这个过滤器拦截地址：j\_spring\_security\_check 实现用户登录。

在登录界面中的表单是有要求的：

用户名表单名称 为 : j\_username

密码表单名为: j\_password

表单的action地址为：j\_spring\_security\_check 。

这样在登录表单输入用户名和密码后，调用后端的实现UserDetailsService接口的类获取用户并做验证。在BPMX3平台中这个接口有 SysUserDao 实现。

另外有一个问题：如果需要加上验证码如何实现呢？

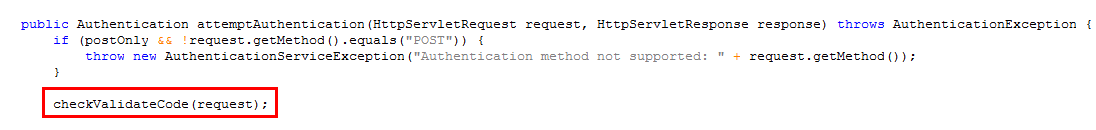
最简单的办法：

修改UsernamePasswordAuthenticationFilter这个过滤器的源码。

修改如下：

在这个类中添加如下方法：





在attemptAuthentication 方法中增加这个方法的调用。

在app-security.xml修改如下：



### 扩展UsernamePasswordAuthenticationFilter实现。

上面的方法可以满足实现验证码的需求，但不是最好的解决办法，因为修改了源码，最好是不动他的源码，我们通过扩展他的过滤器来实现是更好的办法。

### 自己写控制器调用

在bpmx3 中实现的方法是通过自己写控制器来实现登陆的。

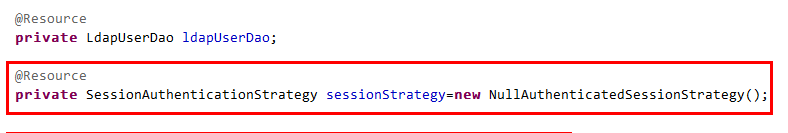


红框中的代码是必须的，之前因为没有这段代码实现不了踢出前面登录用户的功能。sessionStrategy 是session策略，配置了



之后这个session为ConcurrentSessionControlStrategy。这个有兴趣的可以研究一下这个代码（org.springframework.security.web.authentication.session. ConcurrentSessionControlStrategy）。

这个策略通过如下代码注入：



# 多个用户登录界面实现

## 简要说明

一般情况下我们的登录入口只用一个就可以了，有些情况下可能需要多个入口，比如手机端登录需要一个登录界面，这种情况下就有了多个入口。

## 实现方法

实现接口AuthenticationEntryPoint，在平台中实现类如下：

com.hotent.platform.web.filter.MultipleAuthenticationLoginEntry

这个类的实现的思路是：

1.遍历配置的入口策略，如果当前输入的URL符合某个策略的情况，则返回该策略的跳转地址。

2.如果没有策略满足则返回默认的登录地址。

在平台中配置如下：



# 切换用户

## 简要说明

在平台中管理员可能希望不用通过登录，直接使用其他用户的身份，查看该用户的数据。比如在测试流程的时候，通过切换用户进行流程审批。这样大大提高了测试效率。

## 实现方法

spring security 提供了这种实现，我们自定义一个过滤器。

com.hotent.core.web.filter.HtSwitchUserFilter 继承 SwitchUserFilter。

这个过滤器拦截地址：

j\_spring\_security\_switch\_user ：需要切换用户

j\_spring\_security\_exit\_user ：需要退出切换

过滤器实现的逻辑：

切换用户时

记录一个cookie表示当前处于切换用户的状态，表示不能再次切换，并且可退出切换。

退出切换时

删除该cookie。

配置方法如下：

