

# 优乐选商户平台自定义认证

## 修改web.xml

修改youlexuanshopweb\src\main\webapp\WEB-INF\web.xml

```
<!--spring安全过滤器注入到这里，动态加载过滤器-->
<filter>
    <filter-name>springSecurityFilterChain</filter-name>
    <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-
class>
</filter>
<filter-mapping>
    <filter-name>springSecurityFilterChain</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>
```

## 添加安全配置

添加youlexuanshopweb\src\main\resources\spring-security.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/security"
    xmlns:dubbo="http://code.alibabatech.com/schema/dubbo"
    xmlns:beans="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans.xsd
        http://code.alibabatech.com/schema/dubbo
        http://code.alibabatech.com/schema/dubbo/dubbo.xsd
        http://www.springframework.org/schema/security
        http://www.springframework.org/schema/security/spring-security.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context.xsd">
    <!--配置忽略规则-->
    <http pattern="/favicon.ico" security="none"></http>
    <!--TODO 注意：“/”不要丢了-->
    <http pattern="/register.html" security="none"></http>
    <http pattern="/shoplogin.html" security="none"></http>
    <http pattern="/css/**" security="none"></http>
    <http pattern="/img/**" security="none"></http>
    <http pattern="/js/**" security="none"></http>
    <http pattern="/plugins/**" security="none"></http>

    <!--配置拦截规则-->
    <http use-expressions="false">
        <!--拦截地址以及需要的权限-->
        <intercept-url pattern="/**" access="ROLE_SELLER"></intercept-url>
        <!--如果不具备该角色，则跳转到登录页-->
```

```

        <form-login login-page="/shoplogin.html" authentication-failure-
url="/shoplogin.html"
                    default-target-url="/admin/index.html" always-use-default-
target="true"
        ></form-login>
        <logout></logout>
        <!--关闭跨站攻击校验-->
        <csrf disabled="true"></csrf>
        <!--开启框架验证，子页面具有和父页面相同的验证规则-->
        <headers>
            <!--SAMEORIGIN: frame页面的地址只能为同源域名下的页面-->
            <frame-options policy="SAMEORIGIN"/>
        </headers>
    </http>
    <context:property-placeholder location="classpath*:properties/*.properties">
</context:property-placeholder>
    <!--注册客户端名称-->
    <dubbo:application name="youlexuan-shop-web"></dubbo:application>
    <!--注册服务器地址-->
    <dubbo:registry address="${dubbo.address}"></dubbo:registry>
    <!--通过配置方式-->
    <dubbo:reference interface="com.xxx.proj.service.SellerService"
id="sellerService"></dubbo:reference>

    <beans:bean class="com.xxx.proj.service.impl.UserDetailServiceImpl"
id="userDetailServiceImpl">
        <beans:property name="sellerService" ref="sellerService">
</beans:property>
</beans:bean>
    <!--配置认证管理器-->
    <authentication-manager>
        <authentication-provider user-service-ref="userDetailServiceImpl">
            </authentication-provider>
        </authentication-manager>
    </beans:beans>

```

## 添加安全服务自定义实现类

创建youlexuanshopweb\src\main\java\com\xxx\proj\service\impl\UserDetailServiceImpl.java

```

package com.xxx.proj.service.impl;

import com.alibaba.dubbo.config.annotation.Reference;
import com.xxx.proj.pojo.TbSeller;
import com.xxx.proj.service.SellerService;
import org.springframework.security.core.authority.SimpleGrantedAuthority;
import org.springframework.security.core.userdetails.User;
import org.springframework.security.core.userdetails.UserDetails;
import org.springframework.security.core.userdetails.UserDetailsService;
import org.springframework.security.core.userdetails.UsernameNotFoundException;

import java.util.ArrayList;

public class UserDetailServiceImpl implements UserDetailsService {

```

```

private SellerService sellerService;
//添加setter用于手工注入
public void setSellerService(SellerService sellerService) {
    this.sellerService = sellerService;
}
@Override
public UserDetails loadUserByUsername(String sellerId) throws
UsernameNotFoundException {
    //创建权限集合
    ArrayList<SimpleGrantedAuthority> list = new ArrayList<>();
    list.add(new SimpleGrantedAuthority("ROLE_SELLER"));
    //查询登录用户信息
    TbSeller seller = sellerService.findOne(sellerId);
    if ("1".equals(seller.getStatus())) {
        return new User(sellerId, seller.getPassword(), list);
    }
    return null;
}
}

```

## 登录页

```

<form class="sui-form" action="/login" method="post" id="loginForm">
    <div class="input-prepend"><span class="add-on loginname"></span>
        <input id="prependedInput" type="text" name="username" placeholder="邮箱/
用户名/手机号" class="span2 input-xfat">
    </div>
    <div class="input-prepend"><span class="add-on loginpwd"></span>
        <input id="prependedInput" type="password" name="password"
placeholder="请输入密码" class="span2 input-xfat">
    </div>
    <div class="setting">
        <label class="checkbox inline"><input name="m1" type="checkbox"
value="2" checked="">自动登录</label>
        <span class="forget">忘记密码? </span>
    </div>
    <div class="logged">
        <a class="sui-btn btn-block btn-xlarge btn-danger"
onclick="document:loginForm.submit()" target="_blank">登&nbsp;&nbsp;&nbsp;录</a>
    </div>
</form>

```