

Cas 整合 shiro

1添加依赖

其他依赖请自行添加,此处仅 shiro 整合 cas 依赖

```
<dependency>
           <groupId>org.apache.shiro</groupId>
           <artifactId>shiro-all</artifactId>
           <version>1.3.2
       </dependency>
       <!-- https://mvnrepository.com/artifact/org.apache.shiro/shiro-cas -->
       <dependency>
           <groupId>org.apache.shiro
           <artifactId>shiro-cas</artifactId>
           <version>1.3.2
       </dependency>
       <!-- https://mvnrepository.com/artifact/org.apache.shiro/shiro-spring -->
       <dependency>
           <groupId>org.apache.shiro</groupId>
           <artifactId>shiro-spring</artifactId>
           <version>1.3.2
       </dependency>
<!--此处 shiro 缓存使用的是 ehcache, 实际开发请使用自己的替代-->
       <dependency>
           <groupId>net.sf.ehcache/groupId>
           <artifactId>ehcache</artifactId>
           <version>2.10.5
       </dependency>
```

2 自定义 shiro realm

注意需要继承CasRealm

```
public class MzCasRealm extends CasRealm {

    /**

    * 授权访问控制,用于对用户进行的操作进行人证授权,证明该用户是否允许进行当前操作,如
访问某个链接,某个资源文件等

    */
    @Override
    protected AuthorizationInfo doGetAuthorizationInfo(PrincipalCollection
principals) {
```

OFEDU 干锋教育

```
String username = (String) principals.getPrimaryPrincipal();
      此处实现自己的业务逻辑 获取用户的权限和角色
       return new SimpleAuthorizationInfo();
   }
   /**
    * 验证用户身份
    */
   @Override
   protected AuthenticationInfo doGetAuthenticationInfo(AuthenticationToken
token) throws AuthenticationException {
       return super.doGetAuthenticationInfo(token);//使用 cas 的认证方式
   }
   @Override
   public void clearCachedAuthorizationInfo(PrincipalCollection principals) {
       super.clearCachedAuthorizationInfo(principals);
   }
   @Override
   public void clearCachedAuthenticationInfo(PrincipalCollection principals) {
       super.clearCachedAuthenticationInfo(principals);
   }
   @Override
   public void clearCache(PrincipalCollection principals) {
       super.clearCache(principals);
   }
   public void clearAllCachedAuthorizationInfo() {
       getAuthorizationCache().clear();
   }
   public void clearAllCachedAuthenticationInfo() {
       getAuthenticationCache().clear();
   public void clearAllCachedKickoutInfo() {
   }
   public void clearAllCache() {
       clearAllCachedAuthenticationInfo();
       clearAllCachedAuthorizationInfo();
       clearAllCachedKickoutInfo();
   }
}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"</pre>
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:aop="http://www.springframework.org/schema/aop"
      xmlns:context="http://www.springframework.org/schema/context"
       xmlns:util="http://www.springframework.org/schema/util"
      xsi:schemaLocation="http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/aop
       http://www.springframework.org/schema/aop/spring-aop.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context.xsd
http://www.springframework.org/schema/util
http://www.springframework.org/schema/util/spring-util.xsd">
    <bean id="shiroFilter"</pre>
class="org.apache.shiro.spring.web.ShiroFilterFactoryBean">
        cproperty name="securityManager" ref="securityManager" />
       <!-- 设定角色的登录链接,这里为cas登录页面的链接可配置回调地址 -->
        cproperty name="loginUrl" value="http://localhost:8080/cas/login?
service=http://localhost:8081/shiro-cas" />
        cproperty name="successUrl" value="/index.htm"></property>
        cproperty name="filters">
            <util:map>
                <entry key="casFilter" value-ref="casFilter" />
                <entry key="logout" value-ref="logout" />
            </util:map>
       </property>
        cproperty name="filterChainDefinitions">
            <value>
                /shiro-cas=casFilter
                /logout.html=logout
                /login.html=authc
                /xzgh/**=authc
               /tjfx/**=authc
               /gxfw/**=authc
                /xtsz/**=authc
               /region/**=authc
                /shareApi/**=authc
                /**=anon
            </value>
       </property>
    </hean>
    <!-- shiro-cas登录过滤器 -->
    <bean id="casFilter" class="org.apache.shiro.cas.CasFilter">
       <!-- 配置验证错误时的失败页面 , 这里配置为登录页面 -->
```

```
OFEDU 干锋教育
```

```
cproperty name="failureUrl" value="http://localhost:8080/cas/login?
service=http://localhost:8081/shiro-cas" />
    </bean>
    <!-- 退出登录过滤器 -->
    <bean id="logout" class="org.apache.shiro.web.filter.authc.LogoutFilter">
          <property name="redirectUrl" value="http://localhost:8080/cas/logout?</pre>
service=http://localhost:8081/shiro-cas" />
    </bean>
    <!-- 自定义casRealm -->
    <bean id="casRealm" class="com.qianfeng.cas.shiro.realm.MzCasRealm">
         <!-- <pre><!-- <pre>c!-- c!-- c!-- c!-- cellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcellcell</
         <!-- 配置cas服务器地址 -->
         <!-- 客户端的回调地址设置,必须和上面的shiro-cas过滤器casFilter拦截的地址一致 -
         cproperty name="casService" value="http://localhost:8081/shiro-cas" />
    </bean>
    <!--缓存机制 -->
    <bean id="cacheManager" class="org.apache.shiro.cache.ehcache.EhCacheManager">
          cproperty name="cacheManagerConfigFile" value="classpath:ehcache-
shiro.xml" />
    </bean>
    <bean id="securityManager"</pre>
class="org.apache.shiro.web.mgt.DefaultWebSecurityManager">
         cproperty name="realm" ref="casRealm" />
         cproperty name="subjectFactory" ref="casSubjectFactory" />
          cproperty name="cacheManager" ref="cacheManager" />
    </bean>
     <!-- 如果要实现cas的remember me的功能,需要用到下面这个bean,并设置到
securityManager的subjectFactory中 -->
    <bean id="casSubjectFactory" class="org.apache.shiro.cas.CasSubjectFactory" />
     <bean id="lifecycleBeanPostProcessor"</pre>
class="org.apache.shiro.spring.LifecycleBeanPostProcessor" />
    <bean
class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">
         cproperty name="staticMethod"
value="org.apache.shiro.SecurityUtils.setSecurityManager" />
          cproperty name="arguments" ref="securityManager" />
    </bean>
</beans>
```

4 ehcache 配置文件

```
<?xml version="1.1" encoding="UTF-8"?>
<ehcache name="shirocache">
    <diskStore path="java.io.tmpdir"/>
    <defaultCache
            maxElementsInMemory="2000"
            eternal="false"
            timeToIdleSeconds="120"
            timeToLiveSeconds="120"
            overflowToDisk="true"
    />
    <!--
           <cache name="diskCache"</pre>
               maxEntriesLocalHeap="2000"
               eternal="false"
               timeToIdleSeconds="300"
               timeToLiveSeconds="0"
               overflowToDisk="false"
               statistics="true">
        </cache> -->
    <cache name="passwordRetryCache"</pre>
           maxElementsInMemory="2000"
           eternal="false"
           timeToIdleSeconds="300"
          timeToLiveSeconds="0"
           overflowToDisk="false"
    </cache>
    <cache name="authorizationCache"</pre>
           maxElementsInMemory="2000"
           eternal="false"
           timeToIdleSeconds="1800"
           timeToLiveSeconds="0"
           overflowToDisk="false"
    </cache>
    <cache name="authenticationCache"</pre>
           maxElementsInMemory="2000"
           eternal="false"
           timeToIdleSeconds="1800"
           timeToLiveSeconds="0"
           overflowToDisk="false"
    </cache>
```

5 启动访问

访问一个需要登录的页面,会自动跳转到 cas 的登录页面,登录成功后会自动跳回