

**2.主要功能**

**shiro主要有三大功能模块：**

**1. Subject：主体，一般指用户。**

**2. SecurityManager：安全管理器，管理所有Subject，可以配合内部安全组件。(类似于SpringMVC中的DispatcherServlet)**

**3. Realms：用于进行权限信息的验证，一般需要自己实现。**

**3.细分功能**

**1. Authentication：身份认证/登录(账号密码验证)。**

**2. Authorization：授权，即角色或者权限验证。**

**3. Session Manager：会话管理，用户登录后的session相关管理。**

Shiro认证流程

1. 获取当前的Subject 。调用SecurityUtils.getSubject();
2. 测试当前的用户是否己经认证，即是否已经登录。调用Subject的isAuthenticated()
3. 若没有被认证，则把用户名和密码封装为UsernamePasswrodToken对象

1）、创建一个表单页面

2）、把请求提交到springMVC的handler

3）、获取用户名和密码。

4、执行登录：调用Subject的login(AuthenticationToken)方法。

5、自定义Realm的方法，从数据库中获取对应的记录、返回给Shiro.

1）、实际上需要继承org.apache.shiro.ream.AuthenticationRealm类

2）、实现doGetAuthenticationInfo(AuthenticationToken)方法。

6、由shiro完成密码的比对。密码的比对是通过AuthenticatingRealm的 credentialsMatcher属性来进行密码的比对。

1. 如何把一个字符串加密为MD5

**new** SimpleHash(hashAlgorithmName, credentials, salt, hashIterations);

1. 替换当前Realm 的credentialsMatcher属性为