（1）Java认证与授权服务

（2）Java密钥框架（加密解密）

（3）处理公钥和数字凭证的公钥基础设施

（4）Java安全套接字扩展

（1）Java认证与授权（红色表示只存在一个对象）：

（1）LoginContext对象：协调整个登录过程

（2）LoginModule对象：封装不同的登录逻辑

（2-1）CallBackHandler对象：域用户交互的对象（获取数据）

（3）Configuration对象：在不修改源代码的前提下添加新的登录模块LoginModule

（4）Subject对象：表示一个登录的实体（一个用户） 在LoginContext中

（5）Principal对象：表示一个身份标识 在LoginContext中



（2）权限，策略，保护域

（1）Permission对象：表示权限 最重要的implies()方法 进行权限的验证

（2）Policy对象：策略对象 在同一时间只能有一个策略对象

（以文件的形式）



启动jvm是，设置启动参数：java.security.policy=xxx/xxxx/xxx.policy

（3）ProtectionDomain保护域：组成：（1）CodeSource 表示代码的来源

（2）PermissionCollection 权限集合

如果想在运行至保护与可以动态的改变，需要添加几个参数：

（1）ClassLoader

（2）Principal接口的实现数组

（4）访问控制权限：

Java.lang .SecurityManager（原先的API）

AccessController（新加API，对原先的SecurityManager的修改，没有替代）

（5）访问控制上下文：

AccessControlContext类中封装了上下文信息，可以在不同的线程间通过 AccessControlContext进行权限检验

守卫对象：如果在使用访问控制上下文是不是一个好选择，可以使用GuardedObject， 