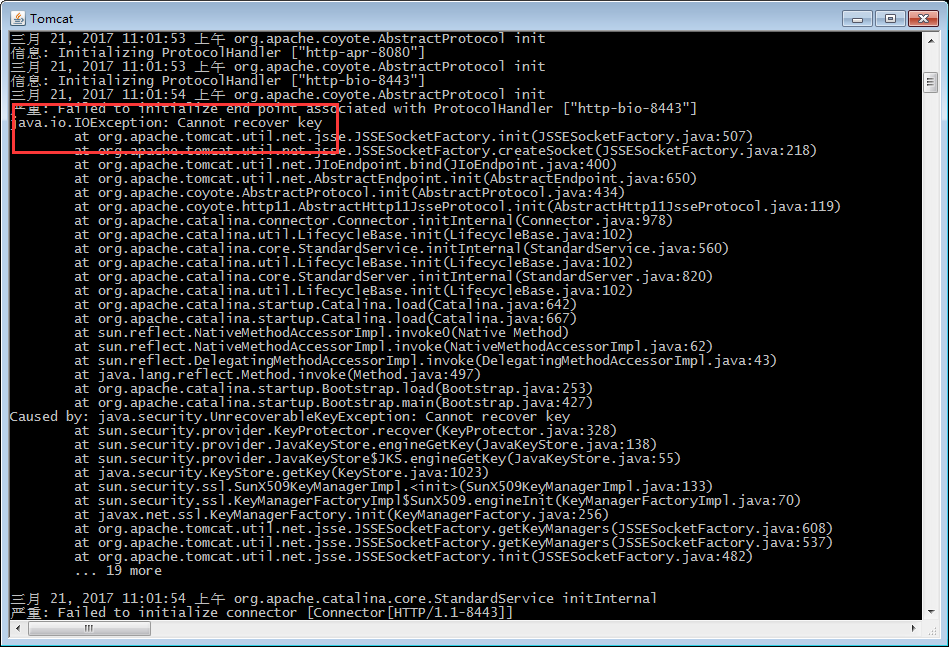
**注：重要步骤“红色”**

1. 第一次执行正确：(产生密钥库文件,生成密钥对参数：-genkeypair。注：生成证书时，CN要和服务器的域名相同，如果在本地测试，则使用localhost。)

**C:\Users\Administrator>keytool -genkey -alias www.beyond.org -validity 3650 -keyalg RSA -sigalg SHA1withRSA -keysize 2048 -keypass keypass -keystore c:\beyond.keystore -storepass storepass -dname "CN=www.beyond.com, OU=beyond ltd, O=beyond, L=GuangZhou, ST=GuangDong, C=cn"**

**注：1. 生成keystore文件时：如果-keypass 与 –storepass不一致，keystore配置到tomcat 启动时报错：**



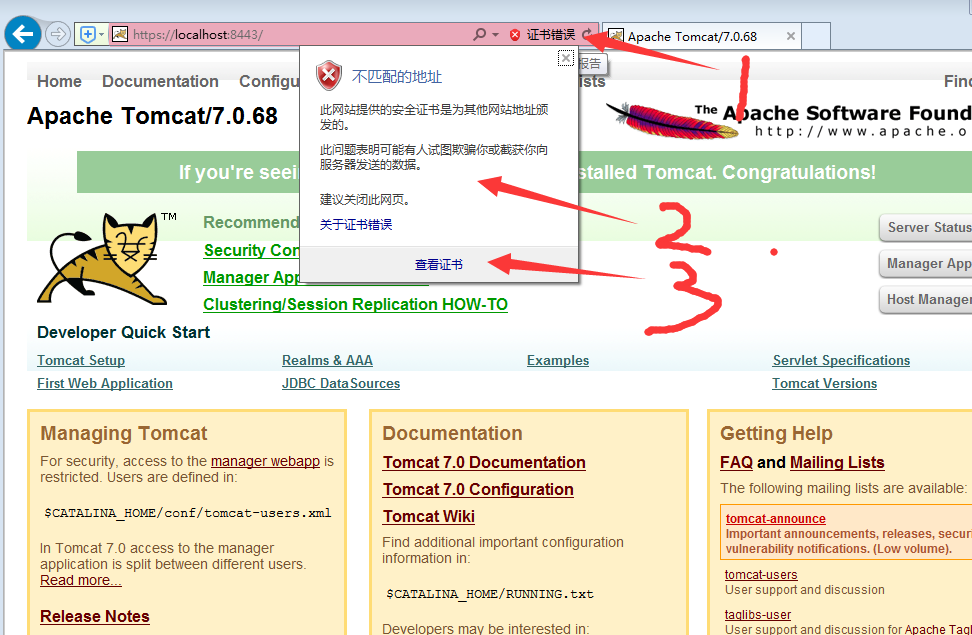
同理：

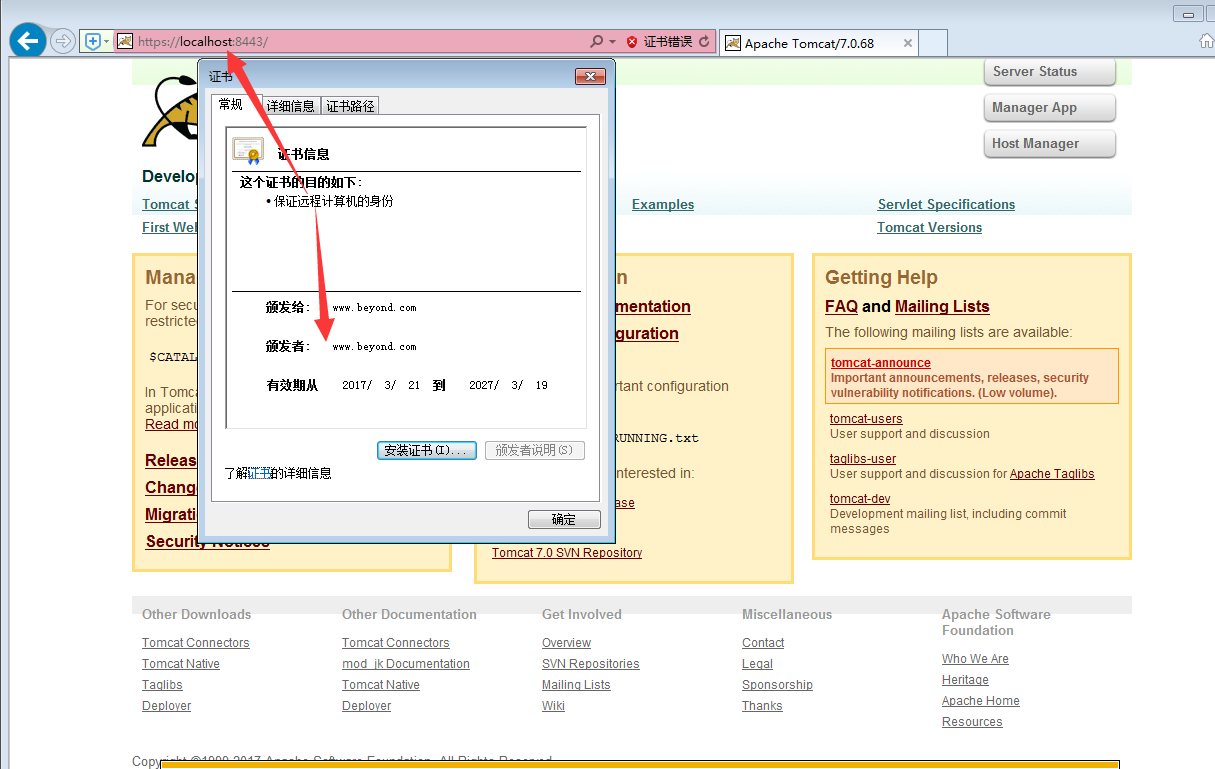
[**Java**](http://lib.csdn.net/base/javase)**代码通过keystore文件获取私钥**privateKey**报错:**

java.security.UnrecoverableKeyException: **Cannot recover key**

参考：<http://blog.csdn.net/venndian/article/details/54410919>

**注：2. 生成keystore文件时,** 如果颁发者与访问网站地址不一致。部署到tomcat后，访问时会报**：**





2. 第二次重复执行抛错：

C:\Users\Administrator>keytool -genkey -alias www.beyond.org -validity 3650 -keyalg RSA -sigalg SHA1withRSA -keysize 2048 -keypass keypass -keystore c:\beyond.keystore -storepass storepass -dname "CN=www.beyond.com, OU=beyond ltd, O=beyond, L=GuangZhou, ST=GuangDong, C=cn"

keytool 错误: java.lang.Exception: 未生成密钥对, 别名 <www.beyond.org.cn> 已经存在

3. 第三次修改别名执行正确：(复制执行文件不会被覆盖，只是更新文件，在密钥库文件增加一个证书)

C:\Users\Administrator>keytool -genkey -alias www.beyond.org.cn -validity 3650 -keyalg RSA -sigalg SHA1withRSA -keysize 2048 -keypass keypass -keystore c:\beyond.keystore -storepass storepass -dname "CN=www.beyond.com, OU=beyond ltd, O=beyond, L=GuangZhou, ST=GuangDong, C=cn"

4. 更改密钥库文件的密码：（参数：-storepasswd 更改密钥库的存储口令。参数：-storepass 输入密钥库的老密码）

C:\Users\Administrator>keytool -storepasswd -keystore c:/beyond.keystore -storepass storepass

5. 列表显示密钥库里所有证书：

C:\Users\Administrator>keytool -list -v -keystore c:/beyond.keystore -storepass storepass

6. 从密钥库文件 导出 指定别名的 证书文件：www.beyond.org (导出文件若不指定目录，默认导出到系统当前用户目录下。证书文件提供给客户端)

**C:\Users\Administrator>keytool -export -alias www.beyond.org -keystore c:/beyond.keystore -storepass storepass -file c:/beyond.org.cer**

7. 打印证书内容：c:/beyond.org.cer

C:\Users\Administrator>keytool -printcert -file c:/beyond.org.cer

8. 从密钥库文件 导出 指定别名的 证书文件：www.beyond.org.cn

C:\Users\Administrator>keytool -export -alias www.beyond.org.cn -keystore c:/beyond.keystore -storepass storepass -file c:/beyond.org.cn.cer

8. 打印证书内容：c:/beyond.org.cn.cer

C:\Users\Administrator>keytool -printcert -file c:/beyond.org.cn.cer

9. 列表显示密钥库里所有证书：

C:\Users\Administrator>keytool -list -v -keystore c:/beyond.keystore -storepass storepass

10. 删除密钥库的证书：www.beyond.org.cn

C:\Users\Administrator>keytool -delete -alias www.beyond.org.cn -keystore c:\beyond.keystore -storepass storepass

11. 列表显示密钥库里所有证书：

C:\Users\Administrator>keytool -list -v -keystore c:/beyond.keystore -storepass storepass

一般生成证书公司自己项目用于加密码传输，生成证书即可。在程序中调用。

若要CA认证机构认证的执行步骤：参考：<http://www.cnblogs.com/SirSmith/p/4996392.html>

全球CA认证三大巨头:Verisign、Thawte、GeoTrust；国内：中国人民银行联合12家银行建立的金融CFCA安全认证中心、中国电信认证中心（CTCA）以及海关认证中心（SCCA）、国家外贸部EDI中心建立的国富安CA安全认证中心等。

查看客户端已安装的证书，显示太多列表时，可以输出到一个文件,若不指定文件目录，默认输出到系统用户目录下。

输入列表命令后，不用写"-storepass xxx"，回车后提示输入密码时,直接按回车(不用输入口令，输入口令会报错)，界面只是提示"尚未经过验证\*\*\*密钥库口令",但证书列表信息还是输出到文件，在文件中查看要导入的证书是否导入。

C:\Users\Administrator>keytool -list -v -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" > log-beyond.txt

12. 在客户端执行导入，把服务器端下发的 证书文件，导入到客户端JVM的 密钥库文件: "%JAVA\_HOME%/jre/lib/security/cacerts",密钥库的密码，默认是 changeit。参数：-trustcacerts 受信任的证书。-storepass 客户端密钥库密码。

( 默认密码：-storepass changeit -keypass changeit )

或删除密钥库文件("%JAVA\_HOME%/jre/lib/security/cacerts")，然后再执行导入。注意，删除密钥库，原密钥库的证书就没有了。

**C:\Users\Administrator>keytool -import -trustcacerts -alias www.beyond.org -file c:/beyond.org.cer -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit**

13. 显示密钥库证书：

参数：-v 详细显示输出 密钥库 证书(不能和-rfc共用)。使用： -list -v

C:\Users\Administrator>keytool -list -v -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit

参数：-rfc 以 RFC 样式输出 密钥库 证书(不能和-v共用)。使用： -list -rfc

C:\Users\Administrator>keytool -list -rfc -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit

参数：-list 简要列表显示 密钥库 证书。使用： -list

C:\Users\Administrator>keytool -list -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit

打印证书信息：

C:\Users\Administrator>keytool -printcert -file c:/beyond.org.cer

14. 删除JVM 密钥库文件的：www.beyond.org 证书

C:\Users\Administrator>keytool -delete -alias www.beyond.org -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit

生成的密钥库，部署到tomcat:

修改tomcat路径conf目录下的server.xml文件，去掉注释并修改以下内容：

<Connector port="8443" protocol="org.apache.coyote.http11.Http11Protocol"

maxThreads="150" SSLEnabled="true" scheme="https" secure="true"

clientAuth="false" sslProtocol="TLS" keystoreFile="c:\beyond.keystore" keystorePass="keypass\_and\_storepass\_is\_same" />

参考：<http://576604498.iteye.com/blog/1838145>

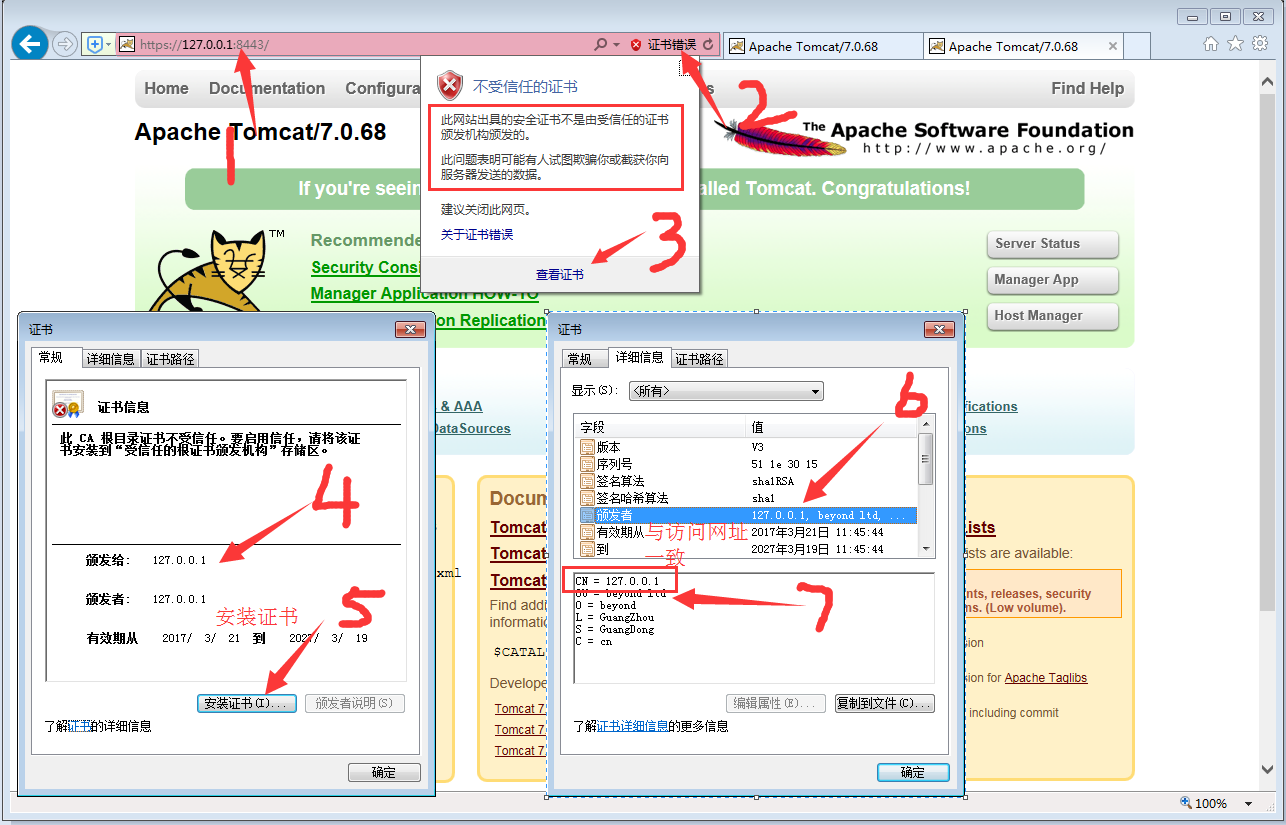
**完整一次过的步骤：**

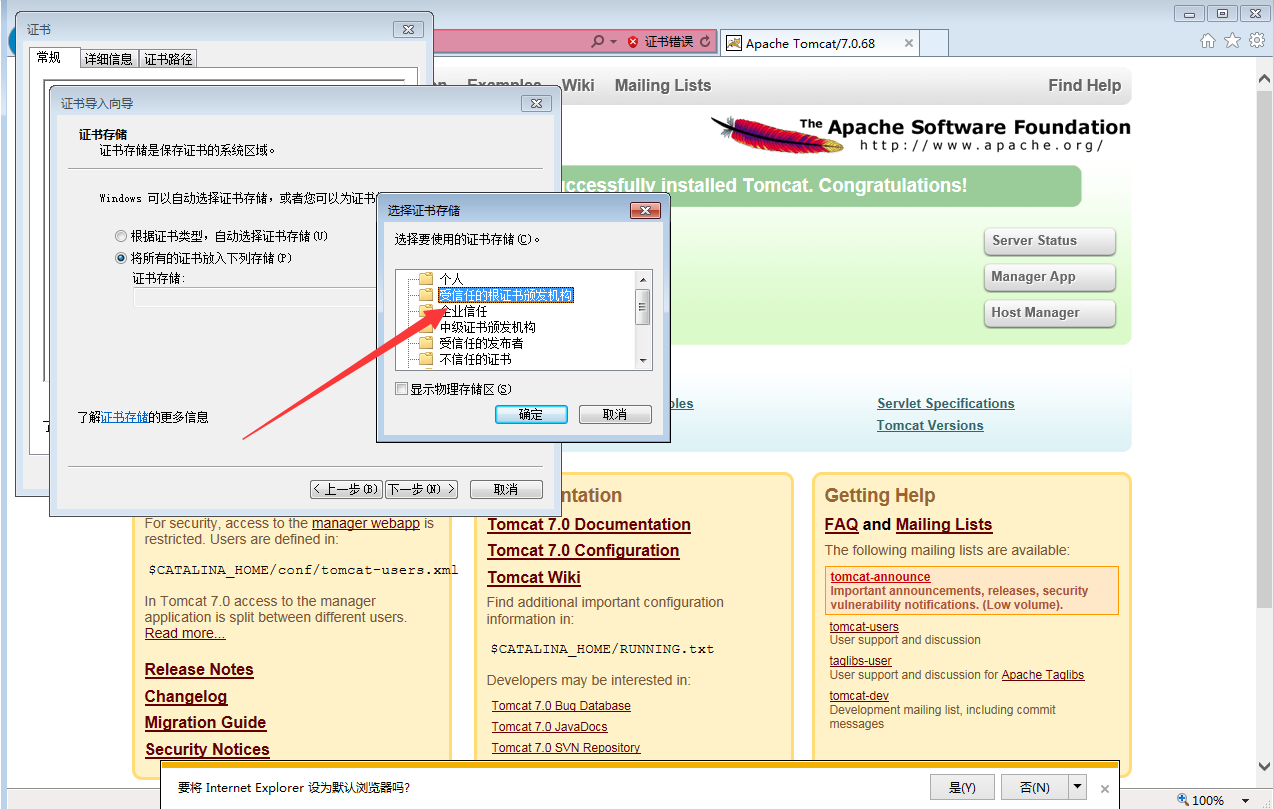
**生成密钥库注意两点：1. -keypass与-storepass一致，部署到tomcat不报错。 2. CN的值 与 访问网站的域名要一致，IE安装证书之后，不再拦截。**

**C:\Users\Administrator>keytool -genkey -alias www.beyond.org -validity 3650 -keyalg RSA -sigalg SHA1withRSA -keysize 2048 -keypass keypass\_and\_storepass\_is\_same -keystore c:\beyond.keystore -storepass keypass\_and\_storepass\_is\_same -dname "CN=127.0.0.1, OU=beyond ltd, O=beyond, L=GuangZhou, ST=GuangDong, C=cn"**

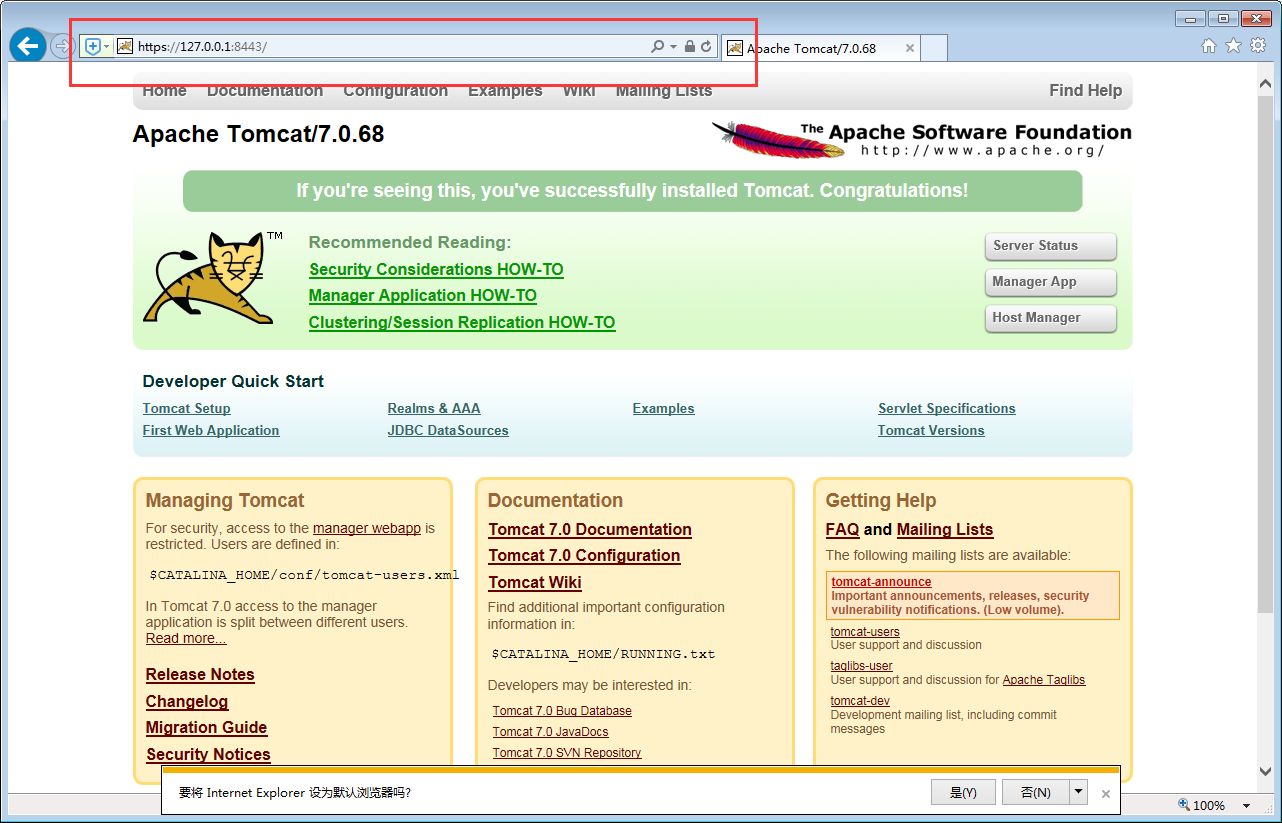
**C:\Users\Administrator>keytool -export -alias www.beyond.org -keystore c:/beyond.keystore -storepass keypass\_and\_storepass\_is\_same -file c:/beyond.org.cer**

**C:\Users\Administrator>keytool -import -trustcacerts -alias www.beyond.org -file c:/beyond.org.cer -keystore "%JAVA\_HOME%/jre/lib/security/cacerts" -storepass changeit**





**通过把证书安装到受信任区，重新打开浏览器，https访问不再提示。**



**//todo现在只生成了密钥库-genkey， 接下来要使用java代码加密码。使用-genkeypair 生成密钥对**