  任何机构或者个人都可以申请数字证书，并使用数字证书对网络通信保驾护航。要获得数字证书，首先需要使用数字证书管理工具，如keytool、OpenSSL等，然后构建CSR（Certificate Siging Request，数字证书签发申请），提交给数字证书认证机构进行签名，最终形成数字证书。

Keytool 是一个Java 数据证书的管理工具 ,Keytool 将密钥（key）和证书（certificates）存在一个称为keystore的文件中。

在keystore里，包含两种数据：

(1)密钥实体（Key entity）——密钥（secret key）又或者是私钥和配对公钥（采用非对称加密）   
(2)可信任的证书实体（trusted certificate entries）——只包含公钥

JDK中Keytool常见的命令：

-genkey 在用户主目录中创建一个默认文件".keystore",还会产生一个mykey的别名，mykey中包含用户的公钥、私钥和证书(在没有指定生成位置的情况下,keystore会存在用户系统默认目录)

-alias 产生别名,每个keystore都关联这一个独一无二的alias，这个alias通常不区分大小写

-keystore 指定密钥库的名称(产生的各类信息将不在.keystore文件中)

-keyalg 指定密钥的算法 (如 RSA DSA（如果不指定默认采用DSA）)

-validity 指定创建的证书有效期多少天

-keysize 指定密钥长度

-storepass 指定密钥库的密码(获取keystore信息所需的密码)

-keypass 指定别名条目的密码(私钥的密码)

-dname 指定证书拥有者信息

-list 显示密钥库中的证书信息

-v 显示密钥库中的证书详细信息

-export 将别名指定的证书导出到文件

-file 参数指定导出到文件的文件名

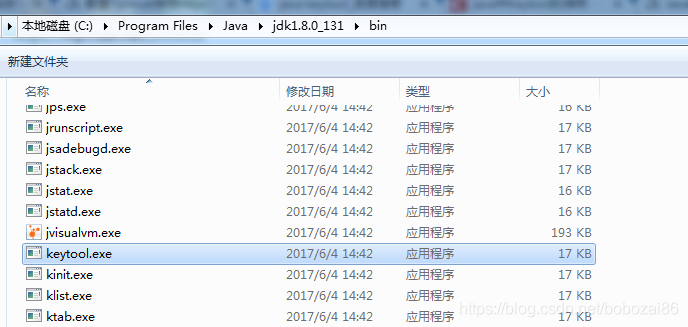
-delete 删除密钥库中某条目

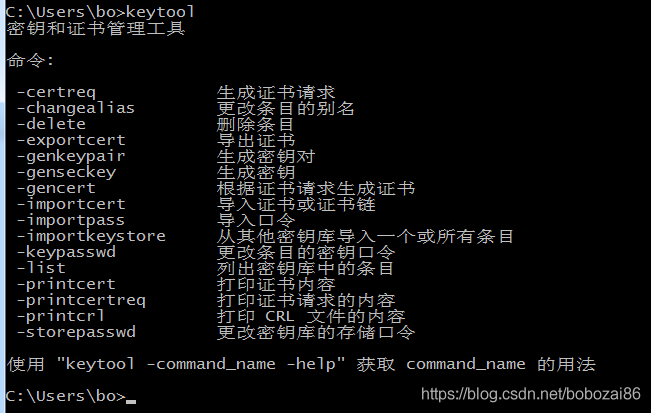
-printcert 查看导出的证书信息

-keypasswd 修改密钥库中指定条目口令

-import 将已签名数字证书导入密钥库

在JDK 1.4以后的版本中都包含了这一工具，它的位置为%JAVA\_HOME%\bin\keytool.exe，如下图：



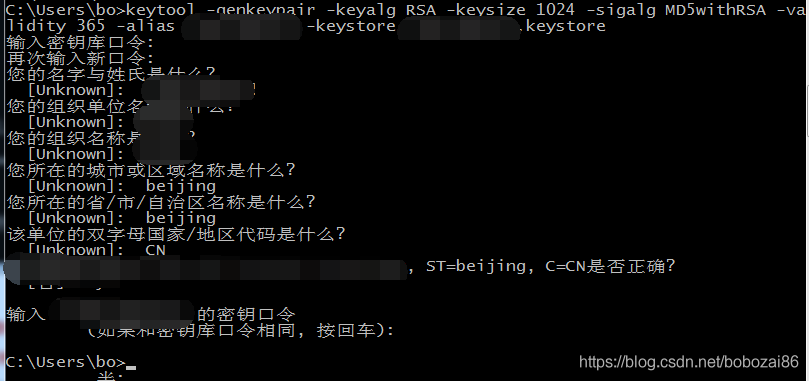


（1）构建自签名证书

       在构建CSR之前，需要先在秘钥库中生成本地数字证书。生成本地数字证书需要提供用户的身份、加密算法、有效期等一些数字证书的基本信息。

keytool -genkeypair -keyalg RSA -keysize 1024 -sigalg MD5withRSA -va

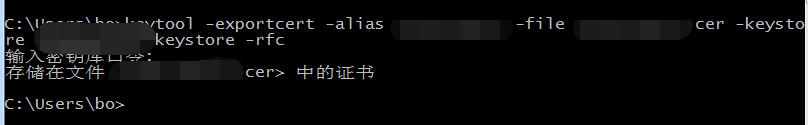
lidity 365 -alias bo -keystore bo.keystore



（2）证书导出

     执行完上述命令之后，我们已经在本地生成了一个数字证书，虽然没有经过证书认证机构进行认证，但是并不影响我们使用，我们可以使用相应的命令导出。

C:\Users\bo>keytool -exportcert -alias bo -file bo.cer -keystore bo.keystore -rfc



（3）导出CSR

       如果想得到证书认证机构的认证，需要导出数字证书并签发申请（CSR），经过认证机构认证并签发后，再将认证后的证书导入到本地秘钥库和信任库。

keytool -certreq -alias bo -keystore bo.keystore -file bo.csr -v

导出CSR后，便可以到VerSign、GeoTrust等权威证书认证机构进行证书认证，但是通过这些认证机构认证的证书往往价格不菲。可以通过一些证书认证机构申请免费的证书，如Cacert。

（4）导入证书

      获得证书认证机构颁发的数字证书后，需要将其导入信任库。

keytool -importcert -trustcacerts -alias bo -file bo.cer -keystore bo.keystore

（5）查看keystore中证书

      通过上面一系列操作导入证书后，便可以通过相关命令，列出keystore中的证书。

keytool -list -alias bo -keystore bo.keystore

keytool没办法签发其它证书，只能只签名生成证书。因此用户如果需要签发证书，需要使用OpenSSL等来进行证书的签发和证书链的管理。

参考：

<https://blog.csdn.net/yinhuaiya/article/details/81774314>

<https://www.cnblogs.com/xdp-gacl/p/3750965.html>