

## tomcat中开启的对SSL(https)的支持

打开conf/server.xml会发现下面一段配置被注释着:

```
<!--  
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"  
    maxThreads="150" scheme="https" secure="true"  
    clientAuth="false" sslProtocol="TLS" />  
-->
```

这段配置可以让tomcat支持SSL连接。默认是注释的。取消他的注释, 并按需修改。

secure必须设置为true, scheme必须设计为https

如果你更改了SSL端口8443为其他端口, 你必须修改其他非SSL的redirectPort, 因为非SSL连接会重定向那些需要SSL安全约束的用户请求到你所修改的那个端口。

去掉注释, 并启动tomcat, 输入 <https://localhost:8443> 就可以看到SSL加密效果。 8443是配置的ssl请求连接端口

### 配置Keystores

要使用ssl connector, 必须先创建一个keystore。他包含了服务器中被客户端用于验证服务器的数字证书。一旦客户端接受了这个证书, 客户端就可以使用public key去加密他们要发送的数据。而服务器, 拥有一个private key, 作为唯一解密数据的密钥。

要创建keystore, 可以使用JAVA\_HOME/bin下的keytool

```
keytool -genkey -alias tomcat -keyalg RSA
```

其中:

-genkey: 创建一个public-private key pair

-alias tomcat: 用户别名为tomcat

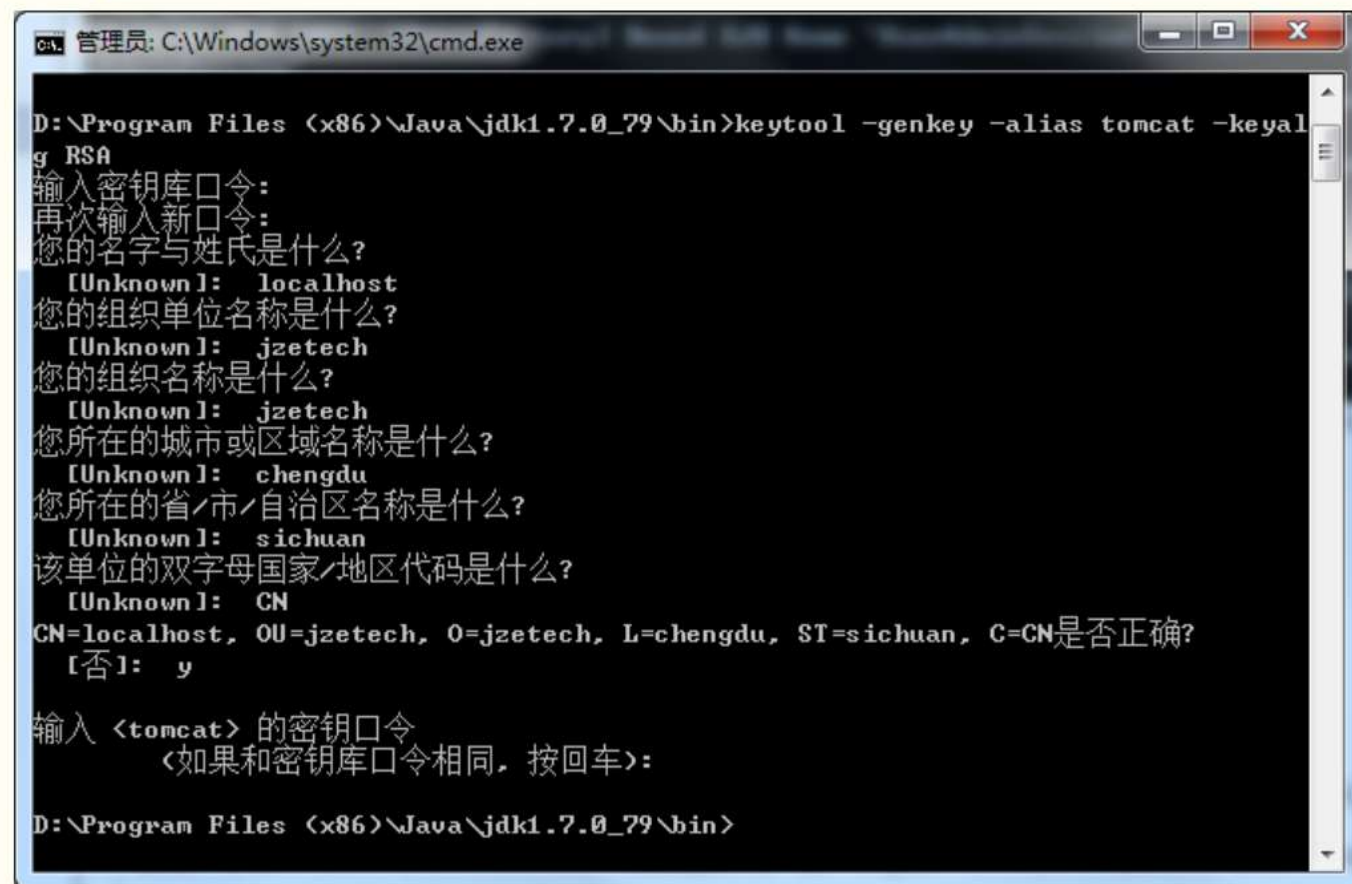
-keyalg RSA: 使用RSA算法。 MD5算法也是被支持的, 但是建议使用RSA获得更好的兼容。

上面那句话, 创建了一个自签证书 (self-signed certificate)

运行上面这段命令后, 如果是第一次使用, 会让你输入密钥库口令, 这个是仓库口令, 然后输入个人资料, 最后为tomcat用户设置口令, 如果想设置为和仓库口令一致, 直接回车即可。

最后, 会在C:\Users\Administrator (当前登录用户名) 下生成一个.keystore的文件, 保存tomcat等key信息。

如下图所示:



```
管理员: C:\Windows\system32\cmd.exe

D:\Program Files (x86)\Java\jdk1.7.0_79\bin>keytool -genkey -alias tomcat -keyalg RSA
输入密钥库口令:
再次输入新口令:
您的名字与姓氏是什么?
[Unknown]: localhost
您的组织单位名称是什么?
[Unknown]: jzotech
您的组织名称是什么?
[Unknown]: jzotech
您所在的城市或区域名称是什么?
[Unknown]: chengdu
您所在的省/市/自治区名称是什么?
[Unknown]: sichuan
该单位的双字母国家/地区代码是什么?
[Unknown]: CN
CN=localhost, OU=jzotech, O=jzotech, L=chengdu, ST=sichuan, C=CN是否正确?
[否]: y
输入 <tomcat> 的密钥口令
如果和密钥库口令相同, 按回车:
D:\Program Files (x86)\Java\jdk1.7.0_79\bin>
```

如果想使用keystore, 那么将connector修改如下:

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"  
    maxThreads="150" scheme="https" secure="true"  
    clientAuth="false" sslProtocol="TLS" keystorePass="你刚才为tomcat用户添加的口令"/>
```