



## 服务器证书安装配置指南

### Tomcat 4

Q/ GlobalSign China-QI-XX-YY

GlobalSign 数字证书颁发机构  
环玺信息科技（上海）有限公司

2018 年 1 月

## 目 录

1. 生成证书请求 .....	3
1.1 安装 JDK .....	3
1.2 生成 keystore 文件.....	3
1.3 生成证书请求文件(CSR) .....	4
2. 导入服务器证书 .....	4
2.1 获取服务器证书中级 CA 证书.....	4
2.2 获取服务器证书.....	4
2.3 查看 Keystore 文件内容.....	4
2.4 导入证书.....	5
3. 安装服务器证书 .....	6
3.1 单向认证的配置.....	6
3.2 双向认证的配置.....	7
3.3 访问测试.....	7
4. 服务器证书的备份及恢复 .....	7
4.1 服务器证书的备份.....	7
4.2 服务器证书的恢复.....	7

## 服务器证书安装配置指南 (Tomcat 4)

### 1. 生成证书请求

#### 1.1 安装 JDK

安装 Tomcat 需要 JDK 支持。如果您还没有 JDK 的安装，则可以参考 Java SE Development Kit (JDK) 下载。下载地址：  
<http://java.sun.com/javase/downloads/index.jsp>

#### 1.2 生成 keystore 文件

生成密钥库文件 keystore.jks 需要使用 JDK 的 keytool 工具。命令行进入 JDK 下的 bin 目录，运行 keytool 命令。（示例中粗体部分为可自定义部分，请根据实际配置情况作相应调整）

```
keytool -genkey -alias server -keyalg RSA -keysize 2048 -keystore keystore.jks
-storepass password
```

```
keytool -genkey -alias weblogic -keyalg RSA -keysize 2048 -keystore c:\ca\keystore.jks
输入 keystore 密码: *****
```

```
您的名字与姓氏是什么？
[Unknown]:  cn.globalseign.com
您的组织单位名称是什么？
[Unknown]:  IT Dept.
您的组织名称是什么？
[Unknown]:  GlobalSign China Co., Ltd.
您所在的城市或区域名称是什么？
[Unknown]:  Shanghai
您所在的州或省份名称是什么？
[Unknown]:  Shanghai
该单位的两字母国家代码是什么
[Unknown]:  CN
```

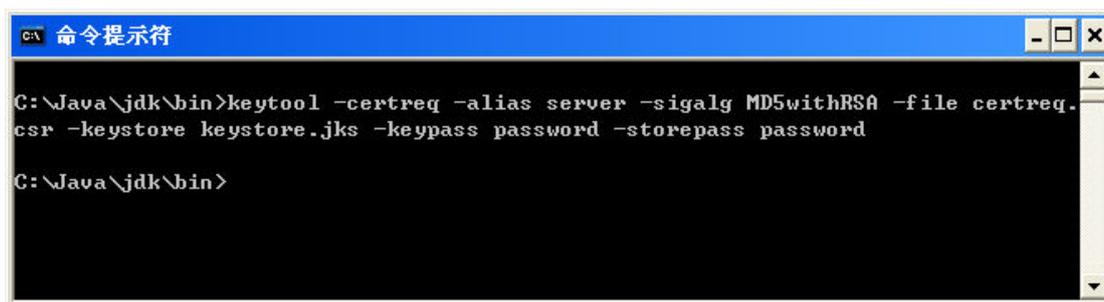
```
CN=cn.globalseign.com, OU=IT Dept, O= GlobalSign China Co., Ltd., L=Shanghai, ST=Shanghai, C=CN
正确吗？
[否]:  Y
```

```
输入<tomcat>的主密码
(如果和 keystore 密码相同，按回车):
```

以上命令中，**server** 为私钥别名(-alias)，生成的 **keystore.jks** 文件默认放在命令行当前路径下。

## 1.3 生成证书请求文件(CSR)

```
Keytool -certreq -alias server -sigalg MD5withRSA -file certreq.csr -keystore keystore.jks -keypass password -storepass password
```



备份密钥库文件 **keystore.jks**，并稍后提交证书请求文件 **certreq.csr**，等待证书签发。

## 2. 导入服务器证书

### 2.1 获取服务器证书中级 CA 证书

为保障服务器证书在 IE7 以下客户端的兼容性，服务器证书需要安装两张中级 CA 证书。  
从邮件中获取中级 CA 证书：

将证书签发邮件中的从 BEGIN 到 END 结束的两张中级 CA 证书内容（包括“-----BEGIN CERTIFICATE-----”和“-----END CERTIFICATE-----”）分别粘贴到记事本等文本编辑器中，并修改文件扩展名，保存为 **intermediate1.cer** 和 **intermediate2.cer** 文件。

### 2.2 获取服务器证书

将证书签发邮件中的从 BEGIN 到 END 结束的服务器证书内容（包括“-----BEGIN CERTIFICATE-----”和“-----END CERTIFICATE-----”）粘贴到记事本等文本编辑器中，并修改文件扩展名，保存为 **server.cer** 文件

### 2.3 查看 Keystore 文件内容

您的 keystore 密码

进入 JDK 安装目录下的 bin 目录，运行 keytool 命令。

```
keytool -list -keystore C:\keystore.jks -storepass password
```

```

C:\WINDOWS\system32\cmd.exe

C:\Java\jdk\bin>keytool -list -keystore C:\keystore.jks -storepass password

Keystore 类型: JKS
Keystore 提供者: SUN

您的 keystore 包含 1 输入

server, 2009-8-21, PrivateKeyEntry,
人证指纹 (MD5): 0C:0C:5A:1A:25:7D:E4:D8:09:2B:1B:DF:43:B8:E3:7C

C:\Java\jdk\bin>

```

查询到 PrivateKeyEntry 属性的私钥别名(alias)为 **server**。记住该别名，在稍后导入服务器证书时需要用到。(示例中粗体部分为可自定义部分，请根据实际配置情况作相应调整。)

注意，导入证书时，一定要使用生成证书请求文件时生成的 keystore.jks 文件。keystore.jks 文件丢失或重新生成 keystore.jks 文件，都将无法正确导入您的服务器证书。

## 2.4 导入证书

导入第一张中级 CA 证书

```
keytool -import -alias intermediate1 -keystore C:\keystore.jks -trustcacerts
-storepass password -file C:\intermediate1.cer
```

导入第二张中级 CA 证书

```
keytool -import -alias intermediate2 -keystore C:\keystore.jks -trustcacerts
-storepass password -file C:\intermediate2.cer
```

```

C:\WINDOWS\system32\cmd.exe

C:\Java\jdk\bin>keytool -import -alias intermediate1 -keystore C:\keystore.jks -
trustcacerts -storepass password -file C:\intermediate1.cer
认证已添加至 keystore 中

C:\Java\jdk\bin>keytool -import -alias intermediate2 -keystore C:\keystore.jks -
trustcacerts -storepass password -file C:\intermediate2.cer
认证已添加至 keystore 中

C:\Java\jdk\bin>

```

导入服务器证书

```
keytool -import -alias server -keystore C:\keystore.jks -trustcacerts -storepass
password -file C:\server.cer
```

```

C:\WINDOWS\system32\cmd.exe

C:\Java\jdk\bin>keytool -import -alias server -keystore C:\keystore.jks -trustca
certs -storepass password -file C:\server.cer
认证回复已安装在 keystore 中

C:\Java\jdk\bin>_

```

导入服务器证书时，服务器证书的别名必须和私钥别名一致。请留意导入中级 CA 证书和导入服务器证书时的提示信息，如果您在导入服务器证书时使用的别名与私钥别名不一致，将提示“认证已添加至 keystore 中”而不是应有的“认证回复已安装在 keystore 中”。

证书导入完成，运行 keytool 命令，再次查看 keystore 文件内容  
keytool -list -keystore C:\keystore.jks -storepass password

```

C:\WINDOWS\system32\cmd.exe

C:\Java\jdk\bin>keytool -list -keystore C:\keystore.jks -storepass password

Keystore 类型: JKS
Keystore 提供者: SUN

您的 keystore 包含 3 输入

intermediate2, 2009-8-21, trustedCertEntry,
认证指纹 (MD5): CA:D5:A7:99:DD:90:93:60:B8:7C:31:9B:DE:D5:F3:2F
intermediate1, 2009-8-21, trustedCertEntry,
认证指纹 (MD5): FC:E3:FD:0D:2D:90:EA:EE:43:17:63:DC:2F:70:2E:4A
server, 2009-8-21, PrivateKeyEntry,
认证指纹 (MD5): 25:55:7C:12:50:1E:86:D9:64:61:2F:66:97:B8:D2:1A

C:\Java\jdk\bin>

```

## 3. 安装服务器证书

### 3.1 单向认证的配置

复制已正确导入认证回复的 keystore.jks 文件到 Tomcat 安装目录下的 conf 目录。打开 conf 目录下的 server.xml 文件，找到并修改以下内容

```

<!--
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8443" enableLookups="true" scheme="https" secure="true"
acceptCount="100"
useURISValidationHack="false" disableUploadTimeout="true"
clientAuth="false" sslProtocol="TLS" />
-->

```

修改为

SSL 访问端口

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
  port="443" enableLookups="true" scheme="https" secure="true"
  acceptCount="100"
  useURIVValidationHack="false" disableUploadTimeout="true"
  keystoreFile="conf\keystore.jks" keystorePass="password"
  clientAuth="false" sslProtocol="TLS" />
```

默认的 SSL 访问端口号为 443，如果使用其他端口号，则您需要使用 `https://yourdomain:port` 的方式来访问您的站点。

## 3.2 双向认证的配置

配置双向认证时，您还需要指定客户端认证的信任库文件。客户端认证信任库文件（`truststoreFile`）可以和服务器证书密钥库文件（`keystoreFile`）为同一个文件，也可以进行独立配置。

示例中使用相同的密钥库文件。您需要首先将客户端认证的根证书以及中级 CA 证书导入到客户端认证信任库。

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
  port="443" enableLookups="true" scheme="https" secure="true"
  acceptCount="100"
  useURIVValidationHack="false" disableUploadTimeout="true"
  keystoreFile="conf\keystore.jks" keystorePass="password"
  truststoreFile="conf\keystore.jks" truststorePass="password"
  clientAuth="true" sslProtocol="TLS" />
```

## 3.3 访问测试

重启 Tomcat，访问 `https://yourdomain:port`，测试证书的安装。

## 4. 服务器证书的备份及恢复

在您成功的安装和配置了服务器证书之后，请务必依据下面的操作流程，备份好您的服务器证书，以防证书丢失给您带来不便。

### 4.1 服务器证书的备份

备份服务器证书密钥库文件 `keystore.jks` 文件即可完成服务器证书的备份操作。

### 4.2 服务器证书的恢复

请参照服务器证书安装部分，将服务器证书密钥库 `keystore.jks` 文件恢复到您的服务器

上，并修改配置文件，恢复服务器证书的应用。