



**JBOSS**

二〇二一年一月

## 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 获取证书 .....	4
2.2 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 ( Jboss )

### 1. 生成证书请求

#### 1.1 安装JDK

安装Jboss 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.globalsign.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
```

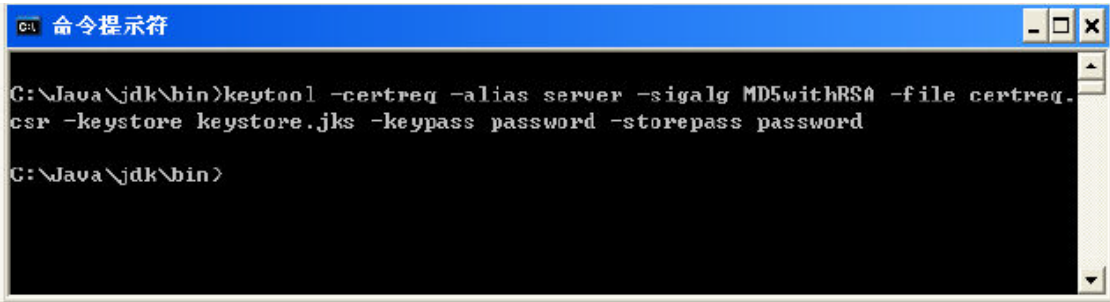
```
CN=cn.globalsign.com, OU=IT Dept, O= GlobalSign China Co., Ltd., L=Shanghai, ST=Shanghai, C=CN  
正确吗?  
[否]: Y
```

```
输入<jboss>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 获取证书

#### 获取中级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（包括中级证书和交叉证书）。

从邮件中获取中级证书和交叉证书：

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 (交叉证书)和 intermediate2.cer (中级证书)文件。

#### 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

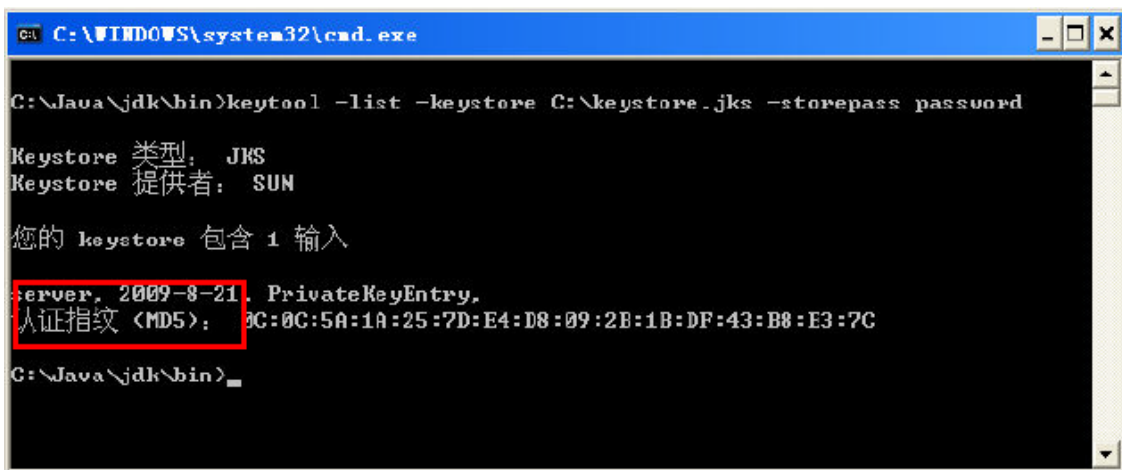
server.cer文件。

### 查看Keystore文件内容

进入 JDK 安装目录下的 bin 目录，运行 keytool 命令。

您的 keystore 密码

`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 文件丢失或生成新的 keystore.jks 文件，都将无法正确导入您的服务器证书。

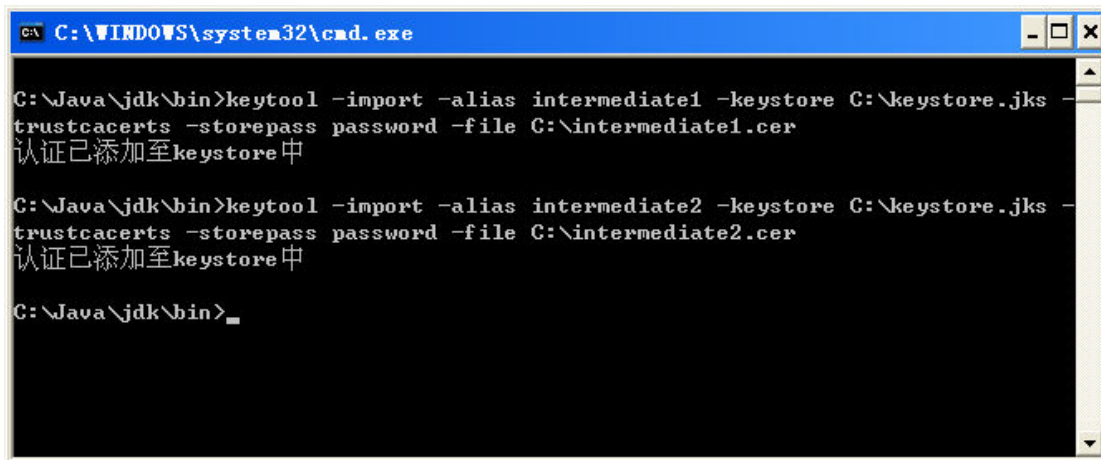
## 2.2 导入证书

导入第一张中级 CA 证书（交叉证书）

`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（中级证书）

`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

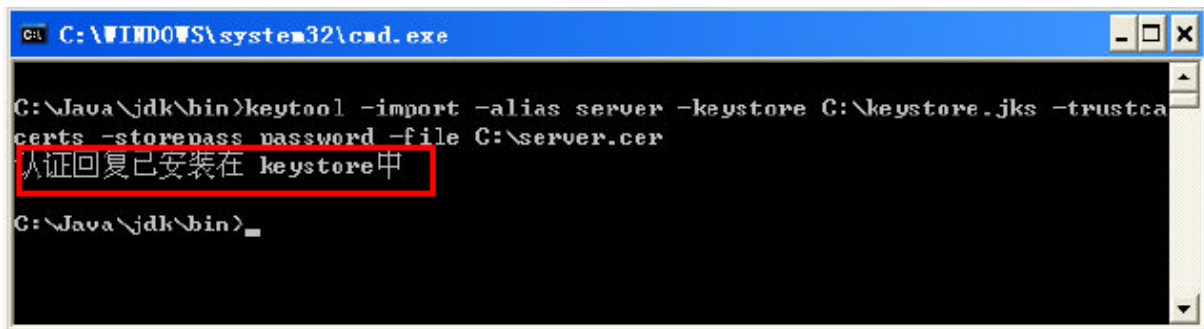
C:\Java\jdk\bin>keytool -import -alias intermediate2 -keystore C:\keystore.jks -trustcacerts -storepass password -file C:\intermediate2.cer
认证已添加至keystore中

C:\Java\jdk\bin>_
```

导入服务器证书

私钥的别名

```
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 -trustcacerts -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): EC:E2:EE:0B: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

打开Jboss 安装目录下server\default\deploy\jbossweb.sar 目录中的server.xml 文件，找到并修改以下内容

```
<!-- SSL/TLS Connector configuration using the admin devl guide keystore

<Connector protocol="HTTP/1.1" SSLEnabled="true"
port="8443" address="{jboss.bind.address}"
scheme="https" secure="true" clientAuth="false"
keystoreFile="{jboss.server.home.dir}/conf/chap8.keystore"
keystorePass="rmi+ssl" sslProtocol = "TLS" />

-->
```

修改为

```
<Connector protocol="HTTP/1.1" SSLEnabled="true"
```

SSL 访问端口

```
port="443" address="{jboss.bind.address}"  
  
scheme="https" secure="true" clientAuth="false"  
  
keystoreFile=" C:\keystore.jks "  
  
keystorePass="password" 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 protocol="HTTP/1.1" SSLEnabled="true"  
  
port="443" address="{jboss.bind.address}"  
  
scheme="https" secure="true" clientAuth="false"  
  
keystoreFile="C:\keystore.jks" keystorePass="password"  
  
truststoreFile="conf\keystore.jks" truststorePass="password"  
  
sslProtocol = "TLS" />
```

### 3.3 访问测试

重启Tomcat，访问https://youdomain: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。

请注意，此文档会不定期更新！

GlobalSign China Co., Ltd

环玺信息科技（上海）有限公司

2021年 1 月