# 引言

**HTTPS和HTTP的区别**

超文本传输协议HTTP协议被用于在Web浏览器和网站服务器之间传递信息。HTTP协议以明文方式发送内容，不提供任何方式的数据加密，如果攻击者截取了Web浏览器和网站服务器之间的传输报文，就可以直接读懂其中的信息，因此HTTP协议不适合传输一些敏感信息，比如信用卡号、密码等。

为了解决HTTP协议的这一缺陷，需要使用另一种协议：安全套接字层超文本传输协议HTTPS。为了数据传输的安全，HTTPS在HTTP的基础上加入了SSL协议，SSL依靠证书来验证服务器的身份，并为浏览器和服务器之间的通信加密。

HTTPS和HTTP的区别主要为以下四点：

一、https协议需要到ca申请证书，一般免费证书很少，需要交费。

二、http是[超文本传输协议](https://baike.baidu.com/item/%E8%B6%85%E6%96%87%E6%9C%AC%E4%BC%A0%E8%BE%93%E5%8D%8F%E8%AE%AE" \t "_blank)，信息是明文传输，https 则是具有[安全性](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E6%80%A7" \t "_blank)的[ssl](https://baike.baidu.com/item/ssl)加密传输协议。

三、http和https使用的是完全不同的连接方式，用的端口也不一样，前者是80，后者是443。

四、http的连接很简单，是无状态的；HTTPS协议是由SSL+HTTP协议构建的可进行加密传输、身份认证的[网络协议](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%8D%8F%E8%AE%AE" \t "_blank)，比http协议安全。

# 目的

nginx和tomcat配置支持https访问。

由引言可知，要支持https，需要安装证书。

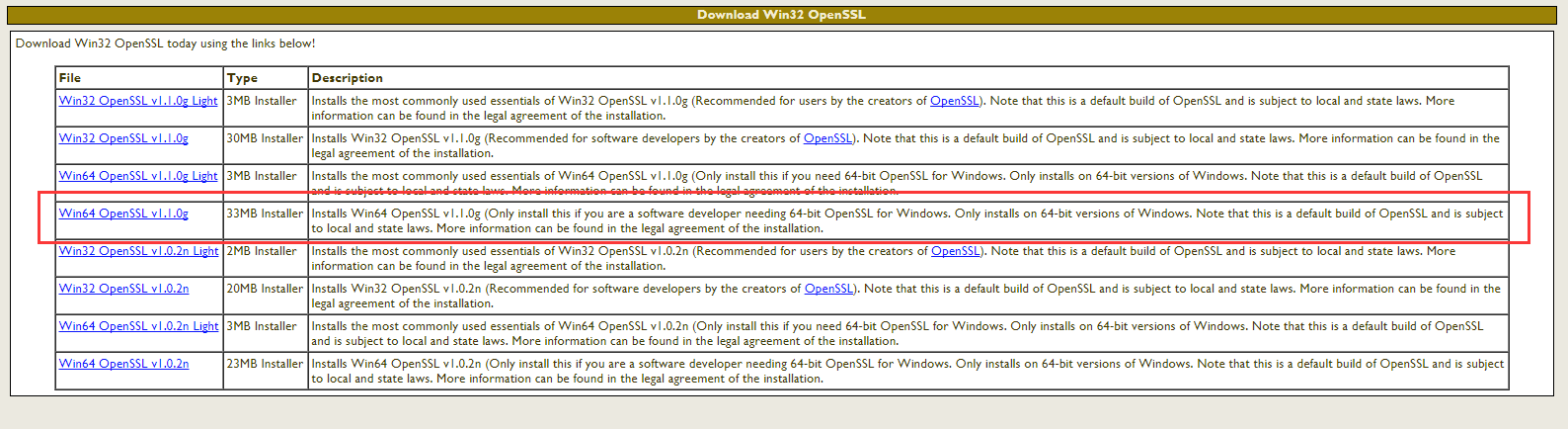
# 配置https

## 1、生成证书For Nginx

以 openssl生成证书为例。

1. **安装openssl**

下载地址：<http://slproweb.com/products/Win32OpenSSL.html>



直接安装Win64OpenSSL-1\_1\_0g.exe即可

1. **配置环境变量OPENSSL\_CONF**

值为openssl.cnf全路径，例：D:\Program Files\OpenSSL-Win64\bin\cnf\openssl.cnf

1. **生成证书**

首先，进入你想创建证书和私钥的目录，例如：

$ cd /usr/local/nginx/conf

1）创建服务器私钥，命令会让你输入一个口令：

$ openssl genrsa -des3 -out server.key 1024

2）创建签名请求的证书（CSR）：

$ openssl req -new -key server.key -out server.csr

3）在加载SSL支持的[Nginx](http://www.centos.bz/category/web-server/nginx/" \o "Nginx)并使用上述私钥时除去必须的口令：

$ cp server.key server.key.org

$ openssl rsa -in server.key.org -out server.key

4）最后标记证书使用上述私钥和CSR：

$ openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

## 2、配置

Nginx和tomcat集成支持https访问，有2种配置方案

方案1：Tomcat自身安装了证书，nginx直接跳转到tomcat的https地址。

方案2：Tomcat不安装证书，仅配置对https的支持，由nginx跳转到tomcat的http地址。

### 1、方案一

Tomcat自身安装证书，nginx直接跳转到tomcat的https地址

#### 1、Tomcat安装https证书

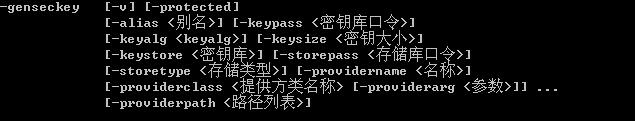
1、java环境：因为SUN公司提供了制作证书的工具keytool。

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。

2、创建证书的命令

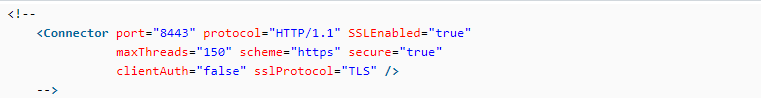
keytool -genkeypair -alias "tomcat" -keyalg "RSA" -keystore "f:\tomcat.keystore"

参数的意思如下：

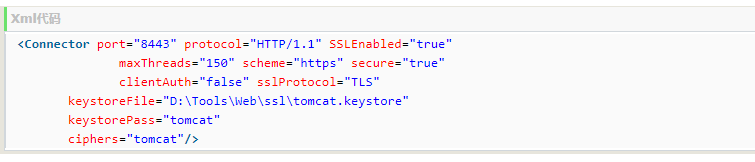


3、修改server.xml

找到如下已经被注释的代码：



去掉注释，修改为



|  |  |
| --- | --- |
| 属性 | 描述 |
| clientAuth | 如果设为true，表示Tomcat要求所有的SSL客户出示安全证书，对SSL客户进行身份验证 |
| keystoreFile | 指定keystore文件的存放位置，可以指定绝对路径，也可以指定相对于<CATALINA\_HOME>（Tomcat安装目录）环境变量 的相对路径。如果此项没有设定，默认情况下，Tomcat将从当前操作系统用户的用户目录下读取名为“.keystore”的文件。 |
| keystorePass | 指定keystore的密码，如果此项没有设定，在默认情况下，Tomcat将使用“changeit”作为默认密码。 |
| sslProtocol | 指定套接字（Socket）使用的加密/解密协议，默认值为TLS，用户不应该修改这个默认值。 |
| ciphers | 指定套接字可用的用于加密的密码清单，多个密码间以逗号（,）分隔。如果此项没有设定，在默认情况下，套接字可以使用任意一个可用的密码。 |

#### 2、Nginx直接跳转Tomcat的https地址

server {

#监听443端口

listen 443 ssl;

server\_name localhost 127.0.0.1 192.168.3.162;

#配置SSL证书和密钥

ssl\_certificate D:/server.crt;

ssl\_certificate\_key D:/server.key;

#可选配置

#ssl\_session\_timeout 5m;

#ssl\_protocols SSLv2 SSLv3 TLSv1;

#ssl\_ciphers ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP;

#ssl\_prefer\_server\_ciphers on;

location /ou/ {

proxy\_pass <https://192.158.43.23>:8000;

proxy\_set\_header X-Real-IP $remote\_addr;

#此处需加上:$server\_port，防止跳转到默认80端口

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_pass\_request\_headers on;

proxy\_redirect off;

proxy\_connect\_timeout 240;

proxy\_send\_timeout 240;

proxy\_read\_timeout 240;

}

}

### 2、方案二

Tomcat不安装证书，仅配置对https的支持，由nginx跳转到tomcat的http地址。

#### 1、Tomcat配置https支持

仅配置对https的支持，修改server.xml增加下图所示内容



<Valve className="org.apache.catalina.valves.RemoteIpValve"

remoteIpHeader="x-forwarded-for"

remoteIpProxiesHeader="x-forwarded-by"

protocolHeader="x-forwarded-proto"

protocolHeaderHttpsValue="https"

/>

#### 2、Nginx跳转Tomcat的http地址

server {

#监听443端口

listen 443 ssl;

server\_name localhost 127.0.0.1 192.168.3.162;

#配置SSL证书和密钥

ssl\_certificate D:/server.crt;

ssl\_certificate\_key D:/server.key;

#可选配置

#ssl\_session\_timeout 5m;

#ssl\_protocols SSLv2 SSLv3 TLSv1;

#ssl\_ciphers ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP;

#ssl\_prefer\_server\_ciphers on;

location /ou/ {

proxy\_pass <http://192.158.43.23>:8000;

proxy\_set\_header X-Real-IP $remote\_addr;

#此处需加上:$server\_port，防止跳转到默认80端口

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_pass\_request\_headers on;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_redirect off;

proxy\_connect\_timeout 240;

proxy\_send\_timeout 240;

proxy\_read\_timeout 240;

}

}