**1. Generate Keystore**

$Tomcat\bin**>**keytool -genkey -alias mkyong -keyalg RSA -keystore c:\mkyongkeystore

Enter keystore password:

Re-enter new password:

What is your first and **last** name?

**[**Unknown**]**: yong mook kim

What is the name of your organizational unit?

**//**omitted to save space

**[**no**]**: **yes**

Enter key password **for** **<**mkyong**>**

**(**RETURN **if** same **as** keystore password**)**:

Re-enter new password:

$Tomcat\bin**>**

#### 2. Connector in server.xml

Next, locate your Tomcat’s server configuration file at $Tomcat\conf\server.xml, modify it by adding a **connector** element to support for SSL or https connection.

File : $Tomcat\conf\server.xml

**//**...

**<!**-- Define a SSL HTTP**/**1.1 Connector on port 8443

This connector uses the JSSE configuration, when using APR, the

connector should be using the OpenSSL style configuration

described **in** the APR documentation --**>**

After the 8080 Port

**<**Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"

maxThreads="150" scheme="https" secure="true"

clientAuth="false" sslProtocol="TLS"

keystoreFile="c:\mkyongkeystore"

keystorePass="password" **/>**

### 4 – BONUS - Configuring your app to work with SSL (access through https://localhost:8443/yourApp)

To force your web application to work with SSL, you simply need to add the following code to your**web.xml** file (before web-app tag ends):

[view source](http://java.dzone.com/articles/setting-ssl-tomcat-5-minutes#viewSource)

[print](http://java.dzone.com/articles/setting-ssl-tomcat-5-minutes#printSource)[?](http://java.dzone.com/articles/setting-ssl-tomcat-5-minutes#about)

01.

02.<security-constraint>

03.<web-resource-collection>

04.<web-resource-name>securedapp</web-resource-name>

05.<url-pattern>/\*</url-pattern>

06.</web-resource-collection>

07.<user-data-constraint>

08.<transport-guarantee>CONFIDENTIAL</transport-guarantee>

09.</user-data-constraint>

10.</security-constraint>

The ***url pattern*** is set to **/\*** so any page/resource from your application is secure (it can be only accessed with **https**). The ***transport-guarantee*** tag is set to **CONFIDENTIAL**to make sure your app will work on **SSL**.

If you want to turn off the SSL, you don’t need to delete the code above from web.xml, simply change**CONFIDENTIAL** to **NONE**.