Acme Madrugá - Https Configuration

Configuring our project

To configure our project go to the webapp folder and locate the web.xml file.

We are going to create a new security tag, in this tag we will indicate the name of our project (Acme-Madruga), the url pattern (put "/*" to use https in all our website, not just in the login process), and the transport guarantee constraint "Confidential" to tell the server that the use of SSL is required.

```
🔘 Java EE - Acme-Madruga/src/main/webapp/WEB-INF/web.xml - Eclipse
    Project Explorer 🛭 🖹 🔄 💝 🔼 🗓 😿 server.xml 🔃 web.xml 🗴
   🖃 👺 Acme-Madruga
                                                                                                       <?xml version="1.0" encoding="UTF-8"?>
         ■ AX-WS Web Services
        ■ 23 Deployment Descriptor: Acme-Madruga
■ 29 Java Resources
                                                                                                         * Copyright (C) 2019 Universidad de Sevilla
         7 * The use of this project is hereby constrained to the conditions of the
8 * TDG Licence, a copy of which you may download from
              pom.xml
        src
                                                                                                        * http://www.tdg-seville.info/License.html
                   java

resources
                                                                                                10
                                                                                                119 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.w3.org/2001
                   i 🗁 🗁 webapp
                                                                                                12
                                                                                                             <display-name>Acme-Madruga</display-name>
                                                                                                139 tener>
                              favicon.ico
                                                                                                                  tener-class>org.springframework.web.context.ContextLoaderListener
                                                                                                           </listener>
                         15
                                                                                                               <param-name>contextConfigLocation</param-name>

    ★ Styles

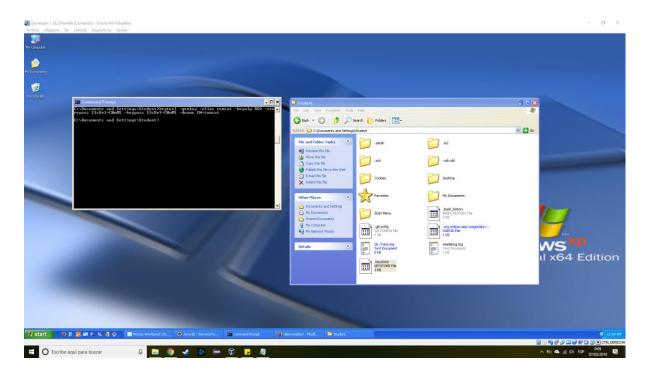
                                                                                                                  19
                                                                                                           </context-param>
                                                                                                20
                              😐 🧁 tags
                                                                                               21<del>0</del>
22<del>0</del>
                                                                                                               <security-constraint>
                                                                                                                   <web-resource-collection>
        🛨 🗁 target
                                                                                                                             <web-resource-name>Acme-Madruga</web-resource-name>
<url-pattern>/*</url-pattern>
                                                                                                23
   </web-resource-collection>
<user-data-constraint>
         □ Improve at localhost-config.
                                                                                               260
                    catalina.policy
                                                                                                                                     <transport-guarantee>CONFIDENTIAL</transport-guarantee>
                   · 📄 catalina.properties
                                                                                                28
                                                                                                                          </user-data-constraint>
                                                                                                               </security-constraint>
                  server.xml
                                                                                               30
                   x tomcat-users.xml
                                                                                               32
33
                  -- 🖈 web.xml
                                                                                                                 <servlet-name>AcmeMadrugaServlet</servlet-name>
                                                                                                                 <servlet-class>org.springframework.web.servlet.DispatcherServlet</ser</pre>
                                                                                               34 <init-param>
                                                                                           Design Source
```

Creating the SSL certificate

To create our certificate we are going to open a command prompt in the location where we are going to stablish the certificate and execute the next command:

keytool -genkey -alias tomcat -keyalg RSA -storepass \$3cUr3=C0mM\$ -keypass \$3cUr3=C0mM\$ -dname

Where keypass is the password of our certificate, in our case we decided to use "\$3cUr3=C0mM\$". It will generate the certificate in the user folder.



Configuring Tomcat

Configuring Tomcat is really easy, go to your server folder an open server.xml, we are going to define a new Connector tag.

This tag already exist in the file but is commented, you just need to add two new params to indicate the location and password of the certificate:

<Connector SSLEnabled="true" clientAuth="false" keystoreFile="\${user.home}/.keystore" keystorePass="\$3cUr3=C0mM\$" maxThreads="150" port="8443" protocol="HTTP/1.1" scheme="https" secure="true" sslProtocol="TLS"/>

```
🔾 Java EE - Servers/Tomcat v7.0 Server at localhost-config/server.xml - Eclipse
        Acme-Madruga

Acme-Madruga

Acme-Madruga

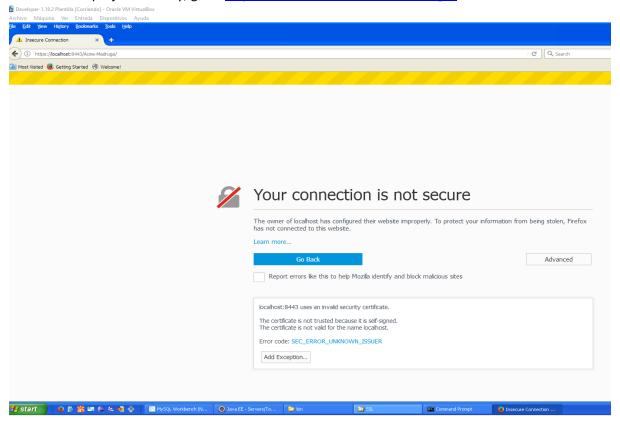
Descriptor: Acme-Madruga

Descriptor: Acme-Madruga

Descriptor: Acme-Madruga
                                                                                                                                                                                           <Connector executor="tomcatThreadPool"</p>
                                                                                                                                                         port="80" protocol="HTTF/1.1"
connectionTimeout="20000"
redirectPort="8443" />
             | avaScript Resources | ava Deployed Resources | available Resources |
                                                                                                                                                                                         ---- 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 -->
                                                                                                                                                                                          --- «Connector SSLEnabled="true" clientAuth="false" keystoreFile="$(user.home)/.keystore" keystorePass="$3cUr3=COmM$" maxThreads="150" port="3443" protocol="HTTF/1.1" scheme="https" secure="true" sslProtocol="TLS"/> <!-- Define an AJP 1.3 Connector on port 8009 --> <Connector port="8009" protocol="AJP/1.3" redirectPort="8443"/>
                                          <!-- An Engine represents the entry point (within Cataling) that processes
every request. The Engine implementation for Tomcat stand alone
analyzes the HTTP headers included with the request, and passes them
on to the appropriate Host (witual host).
Documentation at /docs/config/engine.html -->
                               <Engine name="Catalina" defaultHost="localhost" jvmRoute="jvm1"</pre>
                                                                                                                                                                                          <Engine defaultHost="localhost" name="Catalina">
```

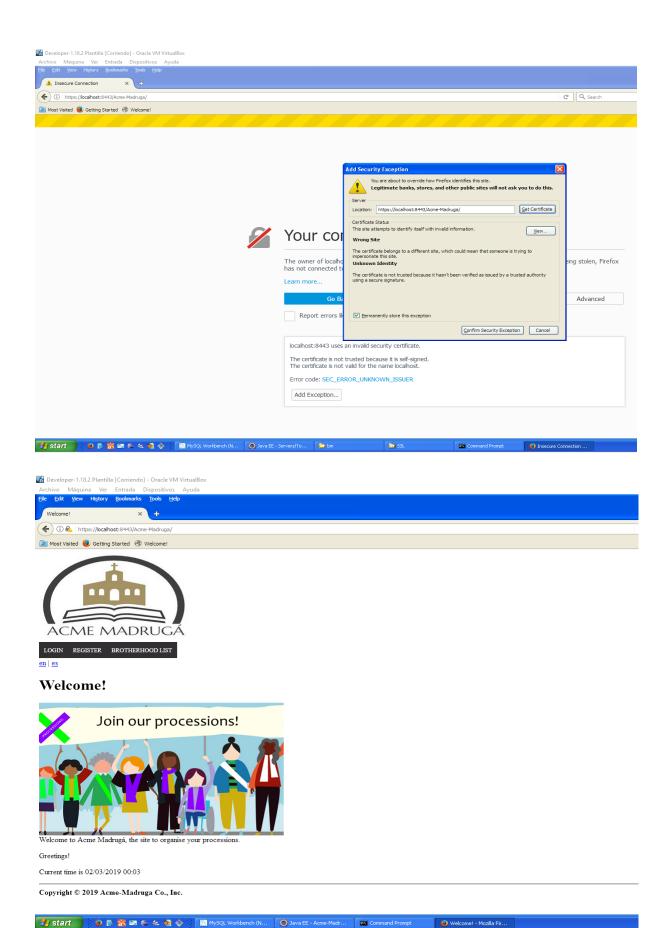
Checking the results

Now that our project is ready, go to http://localhost:8080/Acme-Madruga/



If we see this page, we succeeded configuring https for our application, but since we created our own certificate, the browser will warn us that is probably an insecure page.

Add the exception to the browser and confirm the security exception.



Sample-project

Following this same steps will configure https to the sample-project:

Tomcat server.xml:

```
Java EE - Servers/Tomcat v7.0 Server at localhost-config/server.xml - Eclipse
  ile Edit Navigate Search Project Run Design Window Help
  □ 🔄 👂 🗸
 Sample-Project-1.19

Scrapbook

Servers
                                             ☐ / Formula Tomat v7.0 Server at localhost-cont
         77
78
79
80
81
82
83
84
85
86
87
                                             <Connector SSLEnabled="true" clientAuth="false" keystoreFile="${user.home}/.keystore" keystorePass="$3cUr3=COmM$
maxThreads="150" port="8443" protocol="HTTP/1.1" scheme="https" secure="true" sslProtocol="TLS"/>
                                              <!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443"/>
                                     890
                                             <!-- An Engine represents the entry point (within Catalina) that processes
                                                   every request. The Engine implementation for Tomcat stand alone analyzes the HTTP headers included with the request, and passes them on to the appropriate Host (virtual host).

Documentation at /docs/config/engine.html -->
```

Sample project web.xml:

```
🧔 Java EE - Sample-Project-1.19/src/main/webapp/WEB-INF/web.xml - Eclipse
    File Edit Source Navigate Search Project Run Window Help
     Project Ex 🛭 🏗 Type Hier 🗀 🗓 🗓 server.xml 🔯 web.xml 🖂
                                                                                                                            | Seventh | New July |
                                                                   🖃 🔛 Sample-Project-1.19
             ■ ■ JavaScript Resources
                                                                                                                                                       withs="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" xsi:schemalocation="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
td="WebApp_ID" version="2.5"
td="WebApp
              ■ □ Deployed Resources
              □ 🤛 src
                                                                                                                                                    <display-name>Sample Project</display-name>

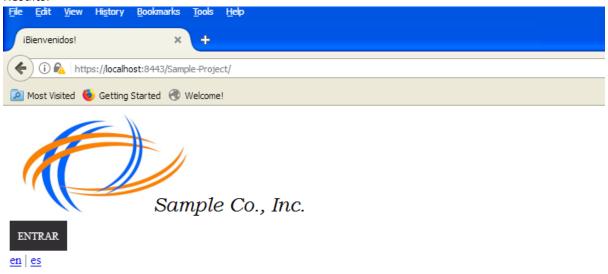
                       亩 🗁 main
                               java

resources
                                                                                                                                                                       contextLoaderListener/listener-class>
                                                                                                                           11
                                                                                                                           12
13
14<sup>-</sup>
                                                                                                                                                     </listener>
                                😑 🧁 webapp
                                                                                                                                                    <security-constraint>
                                         15⊖
16
17

<
                                         scripts
                                                                                                                                                                    </web-resource-collection>
                                         20
21
22
23
                                                                                                                                                                                      <transport-guarantee>CONFIDENTIAL/transport-guarantee>
                                                 🗷 🧁 tags
                                                                                                                                                                      </user-data-constraint>
                                                     web.xml
                                                                                                                                                        </security-constraint>
              249
                                                                                                                                                      <context-param>
     <param-value>classpath:spring/servlet.xml</param-value>
             = E Tomcat v7.0 Server at localhost-conf
                                                                                                                                                        </context-param>
                                 catalina.policy
                                                                                                                                                          <servlet-name>SampleServlet</servlet-name>
<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
                                                                                                                          29
                                 x context.xml
                                                                                                                    tomcat-users.xml
```

The certificate is the same that we used before in Acme-Madruga.

Results:



¡Bienvenidos!

¡Saludos John Doe!

Dia y hora actuales: 07/03/2019 01:01

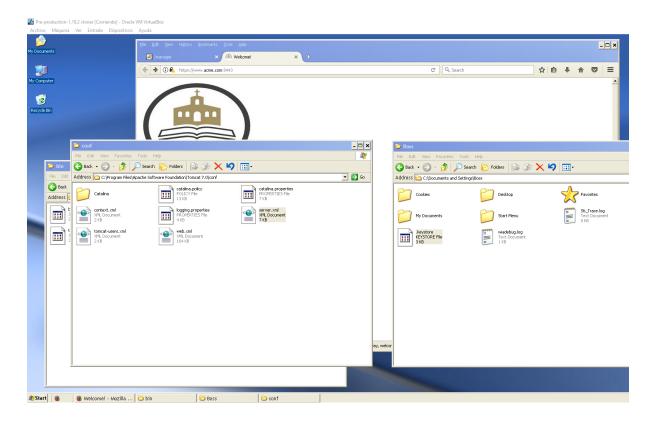
Politica de privacidad - Terminos y condiciones - Sobre nosotros - Copyright © 2019 Sample Co., Inc.

If you want to check this results please use the sample project and sample workspace that we attached to item 2 and move the certificate (inside sample workspace folder) to the users folder, in our case is C:\Documents and Settings\Student.

Pre-Production

To deploy with https configuration in the pre-production machine we need to do some additional steps.

First, copy the key generated previously (.keystore) and put it in C:\Documents and Settings\Boss. Once you do this, go to C:\Program Files\Apache Software Foundation\Tomcat 7.0\conf\server.xml and open it with notepad.

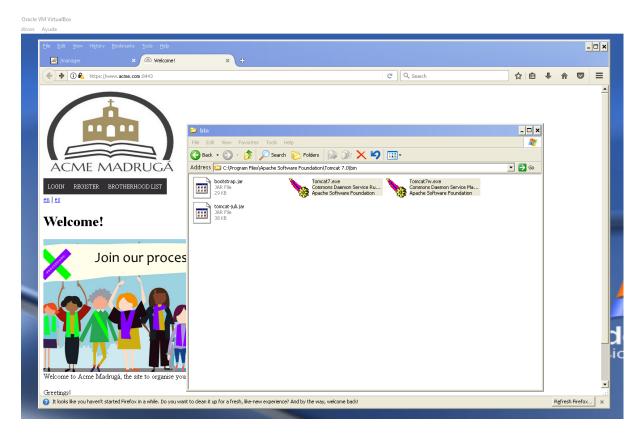


This file is practically the same file we modified in the developer machine but now we need to uncomment and additional line. Make sure you add the parameters keystore and keypass to the connector port like this:

```
<Connector executor="tomcatThreadPool"
    port="80" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />

<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystoreFile="C:\Documents and Settings\Boss\.keystore"
keystorePass="$3cUr3=C0mM$" />
```

Now we need to restart Tomcat. Go to C:\Program Files\Apache Software Foundation\Tomcat 7.0\bin and execute TomCat7w.exe and Tomcat7.exe , in this order, to restart the server.



Create the database and deploy the war like always and open www.acme.com in the browser, it will change automatically to https://www.acme.com:8443/ allowing us to use https.