

Exercise 22: How to use SSL to work with a secure connection

Part 1 - Configure a secure connection

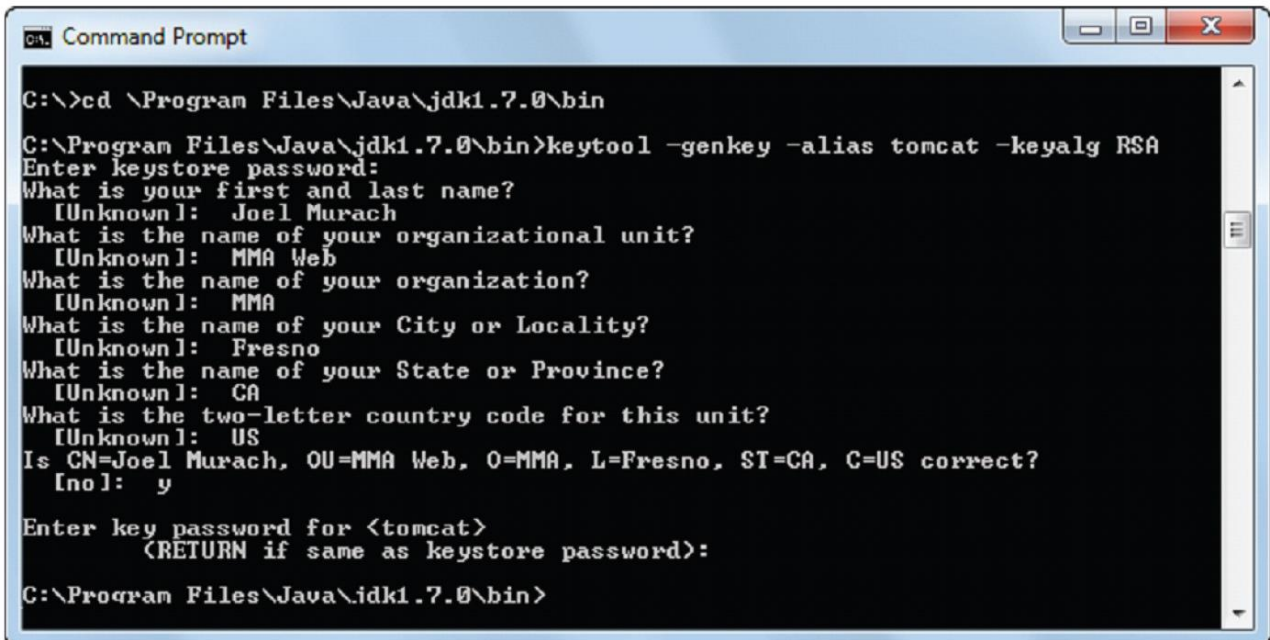
1. Use a Command Prompt (PC) or Terminal window to create a self-signed certificate.

Hint: use keytool application in [jdk directory]\bin

cd [jdk directory]\bin

keytool -genkey -alias tomcat -keyalg RSA

Hint: keytool creates a keystore file named .keystore in your home directory



```
Command Prompt

C:\>cd \Program Files\Java\jdk1.7.0\bin

C:\Program Files\Java\jdk1.7.0\bin>keytool -genkey -alias tomcat -keyalg RSA
Enter keystore password:
What is your first and last name?
  [Unknown]:  Joel Murach
What is the name of your organizational unit?
  [Unknown]:  MMA Web
What is the name of your organization?
  [Unknown]:  MMA
What is the name of your City or Locality?
  [Unknown]:  Fresno
What is the name of your State or Province?
  [Unknown]:  CA
What is the two-letter country code for this unit?
  [Unknown]:  US
Is CN=Joel Murach, OU=MMA Web, O=MMA, L=Fresno, ST=CA, C=US correct?
  [no]:  y

Enter key password for <tomcat>
  (RETURN if same as keystore password):

C:\Program Files\Java\jdk1.7.0\bin>
```

2. Edit Tomcat's server.xml file so it includes a Connector element for secure connection.

Hint: uncomment Connector element and add keystoreFile and keystorePass attributes

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
    SSLEnabled="true" maxThreads="150" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystoreFile="${user.home}/.keystore" keystorePass="changeit"/>
```

3. Restart Tomcat

Part2 - Test the secure connection

4. Enter the URL `https://localhost:8443` into your browser. This should display a warning page.
5. Read the warning page but proceed anyway. This should use a secure connection to display the page you requested, the default Tomcat page. If this doesn't work, you need to troubleshoot the problem and test it again.

Question 1: Explain why you see the warning page.

Part3 - Experiment with a secure application

6. Download `ch15email.zip` from Canvas and open it.
7. Open the `index.jsp` file and review the code for the links.
Question 2: Which link uses a secure connection and which one uses an un-secure connection? Justify your answer.
8. Run this application.
9. (***) Click on the "First connection". Take a screenshot from your browser including the address bar of the browser.
10. Read the warning page, but continue anyway. This should display the "Join our email list" page.
11. (***) Click on the lock icon and use the resulting menu to display the secure certificate. (Take a screenshot from the certificate)
12. Use the application to join the email list.
Question 3: Does this application use a secure connection or an unsecure connection to send the user data to the server? How can you understand?
13. Click the Return button to return the `index.jsp` page.
Question 4: Is this page using a secure connection or unsecure connection?
14. Read the certificate and then return to the application
15. Click on the "Second connection" link.
Question 5: Is this link using a secure connection or an unsecure connection? (Based on the URL in the address bar of the browser)

Deliverables:

1. Create a PDF document and include screenshot(s) and all of your answers to the questions.
2. Upload the document on Canvas.