How to enable SSL (for HTTPS) on JBoss 3.2.5
A definitive step by step guide

dnickull@adobe.com

Preface:

In order to get Adobe liveCycle Policy Server working, Secure Socket Layers (SSL) must be enabled on the application server. The default installation for testing is often JBoss 3.2.5 but no current documentation exists to guide developers through the SSL enablement process. This guide was written to enable anyone to quickly get it done.

Environment:

This "How to" was done using the following configuration:
Macintosh MacBook Pro, Mac OS/X build 10.4.7*
JBoss 3.2.5 downloaded from jboss.org
Java version "1.5.0_06"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_06-112)
Java HotSpot(TM) Client VM (build 1.5.0_06-64, mixed mode, sharing)

Instructions

1. Install the latest JBoss application server.

2. To test if it is working, you must navigate to the ~/install_directory/bin and execute the run shell script by typing in

sh run.sh

The terminal window will tell you its progress as it starts up. Navigate to http://localhost:8080 to make sure it installed correctly.

3. Open a 2nd terminal and generate a key. To make this go quickly, simply cut and paste the following text into your terminal window:

keytool –genkey –alias jboss –keypass rmi+ssl –keyalg RSA –keystore duane.keystore –validity 3650

You will be asked a few questions. Just answer them then hit enter.

```
duane-nickulls-computer:/Applications/jboss-3.2.5/bin nickull$ keytool -genkey -
alias 4jboss -keypass rmi+ssl -keyalg RSA -keystore duane.keystore -validity 365
Enter keystore password:  rmi+ssl
What is your first and last name?
[Unknown]:  Duane Nickull
What is the name of your organizational unit?
[Unknown]:  EDBU
What is the name of your organization?
[Unknown]:  Adobe Systems
What is the name of your City or Locality?
[Unknown]:  Vancouver
What is the name of your State or Province?
[Unknown]:  BC
What is the two-letter country code for this unit?
[Unknown]:  CA
Is CN=Duane Nickull, OU=EDBU, O=Adobe Systems, L=Vancouver, ST=BC, C=CA correct?
[no]:  y
duane-nickulls-computer:/Applications/jboss-3.2.5/bin nickull$ _
```

4. To test to make sure your key got generated, type in the following:

Prompt> keytool –list –v

The v flag is for Verbose to give you the details. You will be prompted for the password you gave earlier. In this example it was “rmi+ssl”

NOTE: It is important to use the “-keystore <keystore>” flag to specify which keystore you wish to use. By default, the Keytool will use the default keystore.

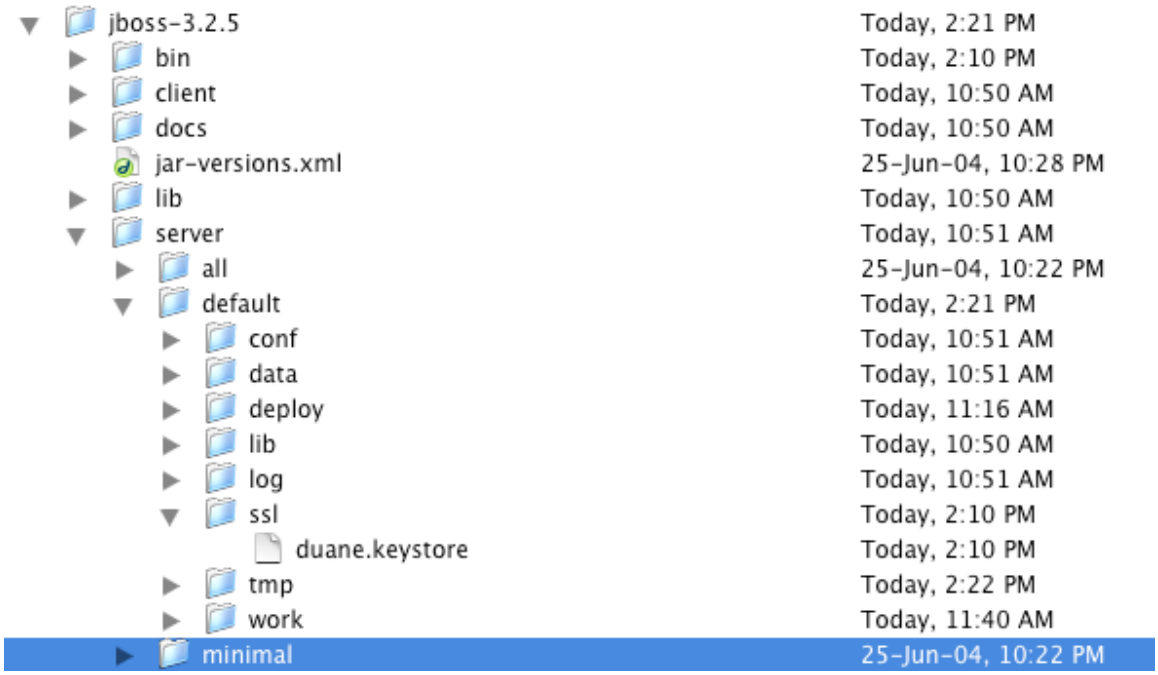
```
duane-nickulls-computer:/ nickull$ keytool -list -v -keystore duane
Enter keystore password:  rmi+ssl

Keystore type: jks
Keystore provider: SUN

Your keystore contains 1 entry

Alias name: duane-key
Creation date: Sep 13, 2006
Entry type: keyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=Duane Nickull, OU=EDBU, O=Adobe, L=Vancouver, ST=BC, C=CA
Issuer: CN=Duane Nickull, OU=EDBU, O=Adobe, L=Vancouver, ST=BC, C=CA
Serial number: 45088135
Valid from: Wed Sep 13 15:07:49 PDT 2006 until: Tue Dec 12 14:07:49 PST 2006
Certificate fingerprints:
         MD5:  F6:B3:D1:1A:8A:C5:CA:32:35:3E:F2:EB:1E:F6:96:0B
         SHA1: 2E:F0:4D:41:FD:B0:3A:D7:4A:C2:AF:A8:1B:20:7D:74:D1:53:3F:97

*******************************************
*******************************************

duane-nickulls-computer:/ nickull$
```

You now have a key to use.

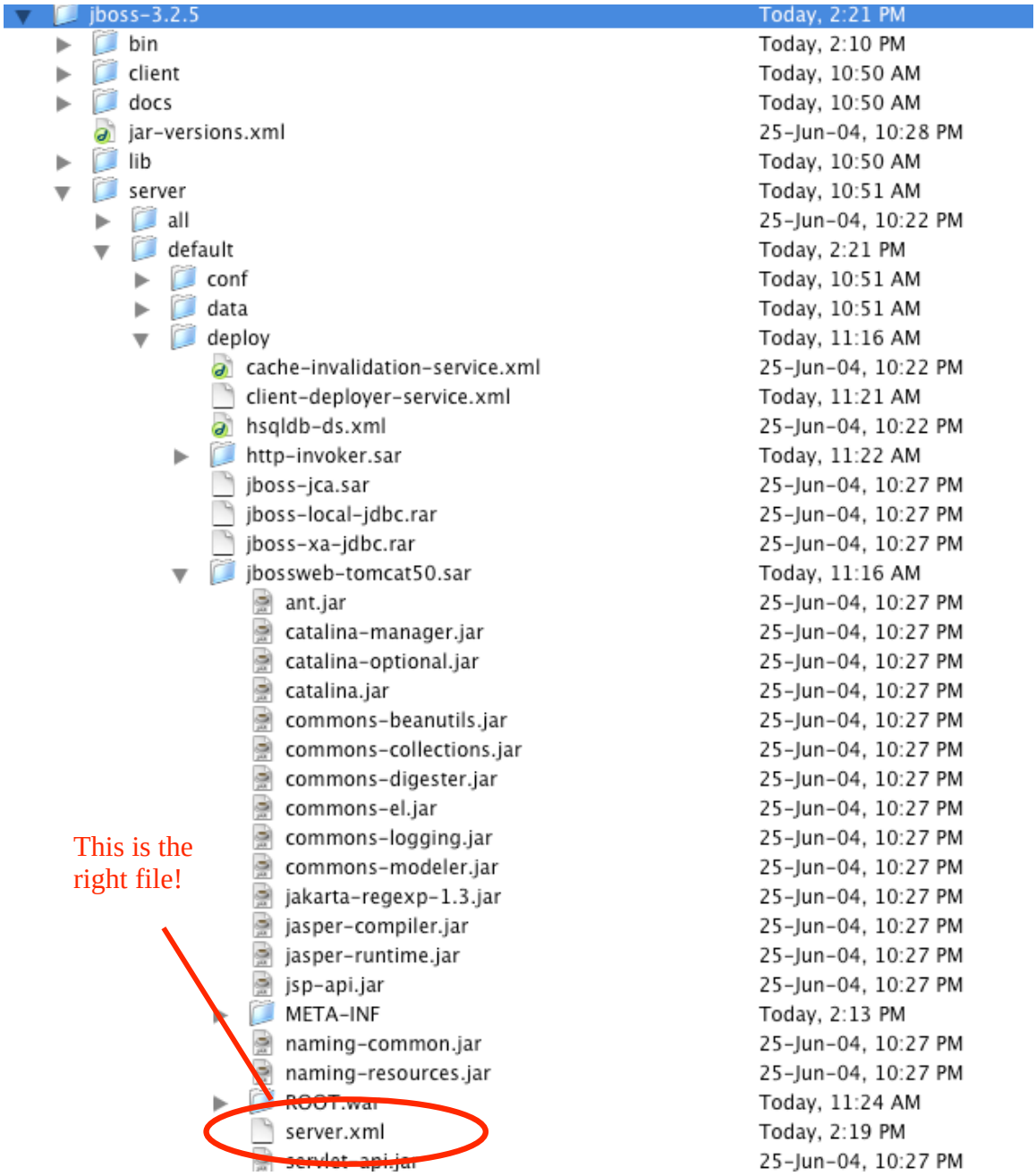
5. The next step is to copy the keystore (duane.keystore) to the correct directory where it will be called during runtime. You can do this with Finder or via the terminal. If you do it via the terminal, simply 'cp' it to:

$JBOSS_HOME/server/default/ssl/

Note that the "ssl" directory will not be there by default so you must create it. You should see the following structure in your Finder:



6. The next step is to modify the server.xml file within the $JBOSS_HOME/server/default/deploy/jbossweb-tomcat50.sar folder. This is the server.xml file that must tell the application server where to find the keystore and how to access it during startup.



7. Open the server.xml file with a text editor of your choice and uncomment the block of XML to declare the SSL. This is easy and you should only have to modify two pieces. The first one is the actual path to the keystore file relative to the jboss home directory; the second is the password required for the application server to access the key.

```
<!-- SSL/TLS Connector configuration using the admin devl guide keystore-->
<Connector port="8443" address="${jboss.bind.address}"
     maxThreads="100" minSpareThreads="5" maxSpareThreads="15"
     scheme="https" secure="true" clientAuth="false"
     keystoreFile="${jboss.server.home.dir}/ssl/duane.keystore"
     keystorePass="rmi+ssl" sslProtocol = "TLS" />
<!--end ssl-->
```

Note that the KeystoreURL and the KeyStorePass must be the same as the values you used when you generated the key.

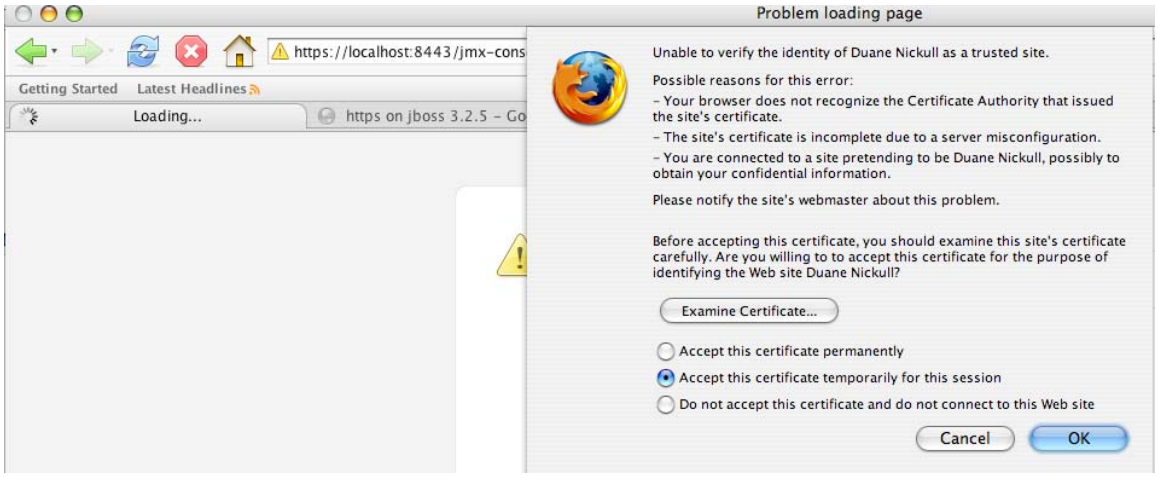
8. Save this file then navigate to your $JBOSS_HOME/bin directory and start the server. On Mac the syntax is "sh run.sh". You should see jBoss Start up.

   NOTE: By default, I have not entered any specific configuration of JBoss and apply this only to the default jboss instance. If you want to enable ssl on other instances of JBoss, you will have to modify the files under those server directories.

9. One error that may occur due to different class loaders is that the jboss instance will be unable to find the keystore. Examine the resulting trace to the shell to determine the problem. For example, the output below clearly shows that the JBoss server could not find the keystore because it wasn't in the directory where it was looking. Modify the server.xml files as required to help JBoss find the keystore.

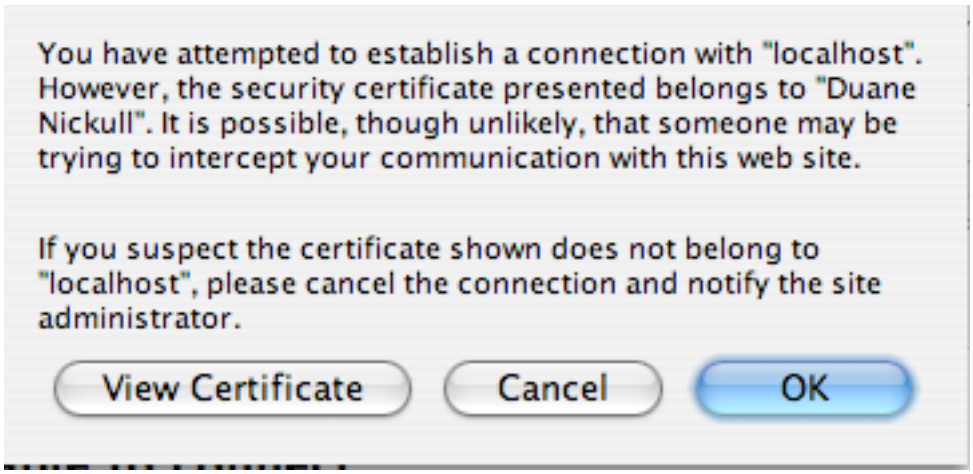
```
14:19:52,530 WARN  [Tomcat5] Failed to startConnectors
LifecycleException:  Protocol handler start failed: java.io.FileNotFoundExceptio
n: /Applications/jboss-3.2.5/server/default/ssl/duane.keystore (No such file or
directory)
        at org.apache.coyote.tomcat5.CoyoteConnector.start(CoyoteConnector.java:
1529)
```

Grab your browser and hit https://localhost:8443



You should see the following message warning that the certificate is a bit dodgy. Since it is your certificate you just made, you can accept it permanently and you shouldn't see this warning any more.

A second warning will appear like the one below. Click "OK".



Voila! JBoss using SSL


JBoss JMX Management Console
https://localhost:8443/jmx-console/
Getting Started
Latest Headlines
JBoss JMX Management Console
https on jboss 3.2.5 - Google Se...
Nexaweb WebEx Enterprise Site
JMX Agent View
duane-nickulls-computer.local
ObjectName Filter (e.g. "jboss:*", "*:service=invoker,*"):
Apply Filter
Clear Filter
Catalina
type=Server