

# Install Custom Tomcat 7 on Azure Website with Java 8

Friday, April 10, 2015 8:01 PM

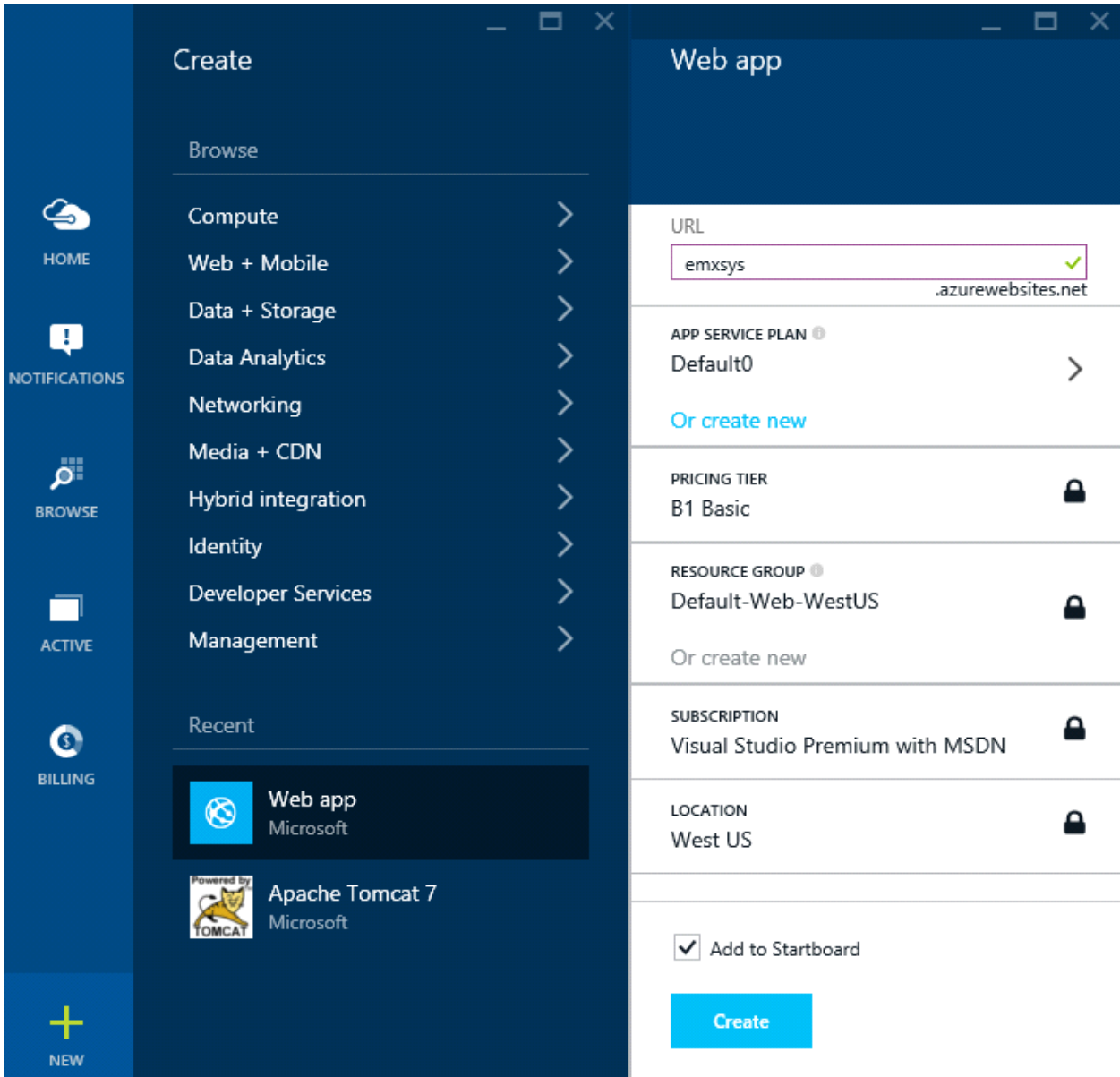
Reference Article: <<http://azure.microsoft.com/en-us/documentation/articles/web-sites-java-custom-upload/>>

Launch Azure Preview

<https://portal.azure.com/>

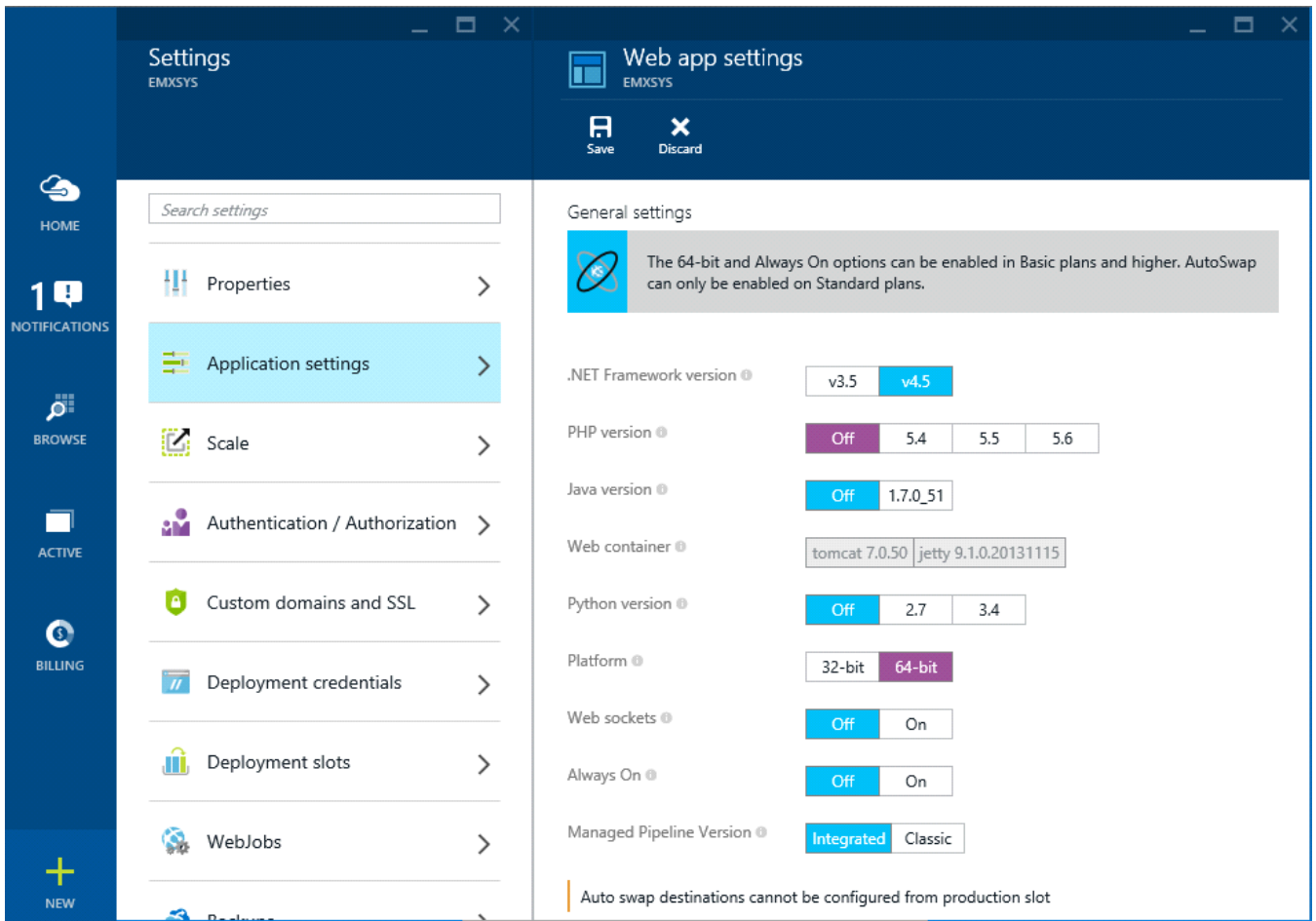
Create a new Web app

Click NEW -> Web app. Enter URL and click Create.



Configure the Website

Set for 64 bit



### Prerequisite: Download the Windows x64 Java 8 Server JRE

Note: I was unable to get the server to startup using JRE 8\_u40, but does work with the first JRE 8, both the server version and the standard version. You will have to go to the Oracle Java Archives to download earlier JREs:

<http://www.oracle.com/technetwork/java/javase/downloads/java-archive-javase8-2177648.html>

<http://www.oracle.com/technetwork/java/javase/downloads/server-jre8-downloads-2133154.html>

### Prerequisite: Download the Tomcat 7 Windows x64 Distribution

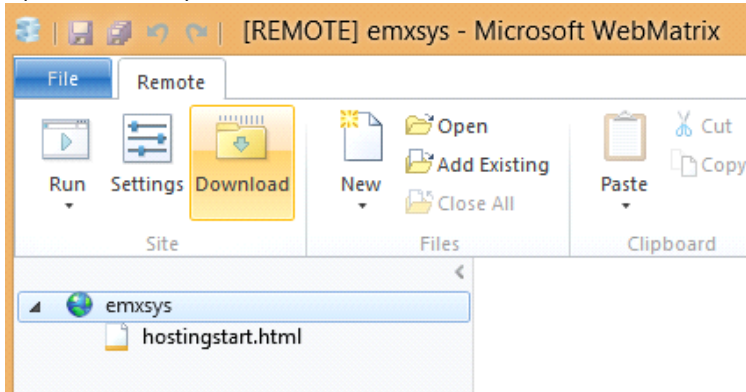
<https://tomcat.apache.org/download-70.cgi>

<http://download.nextag.com/apache/tomcat/tomcat-7/v7.0.59/bin/apache-tomcat-7.0.59-windows-x64.zip>

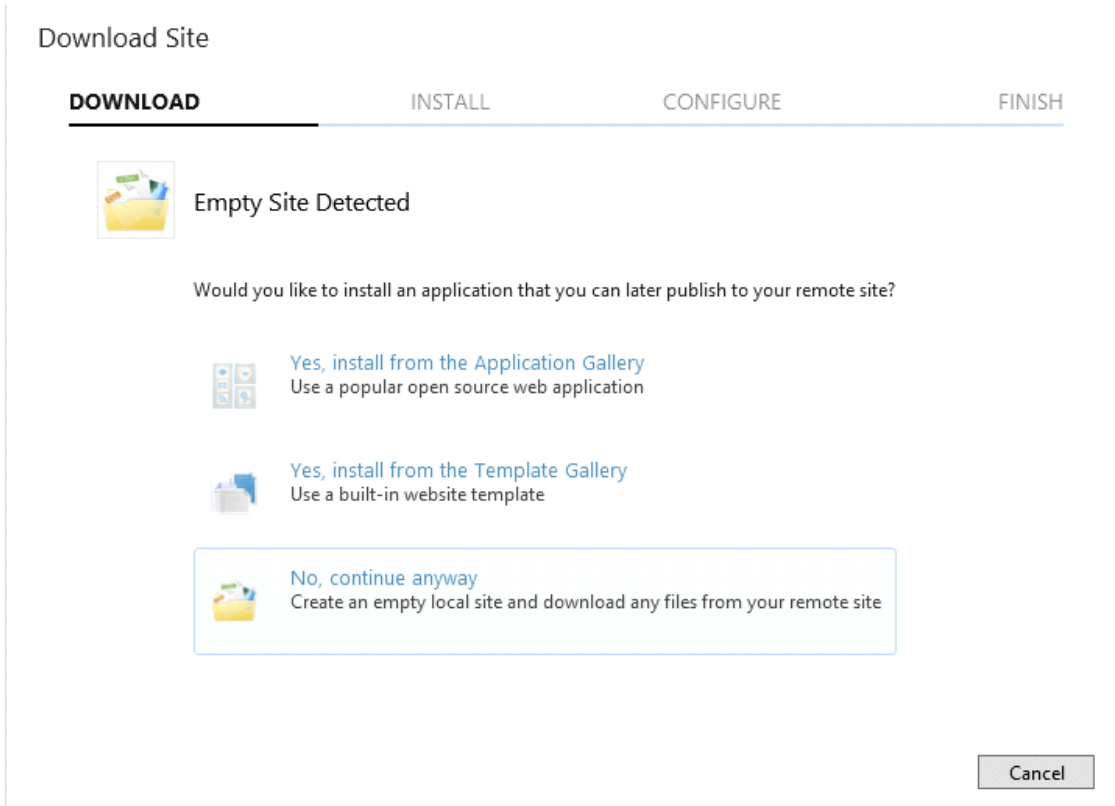
<http://mirror.reverse.net/pub/apache/tomcat/tomcat-7/v7.0.61/bin/apache-tomcat-7.0.61-windows-x64.zip>

### Download new Website with WebMatrix3

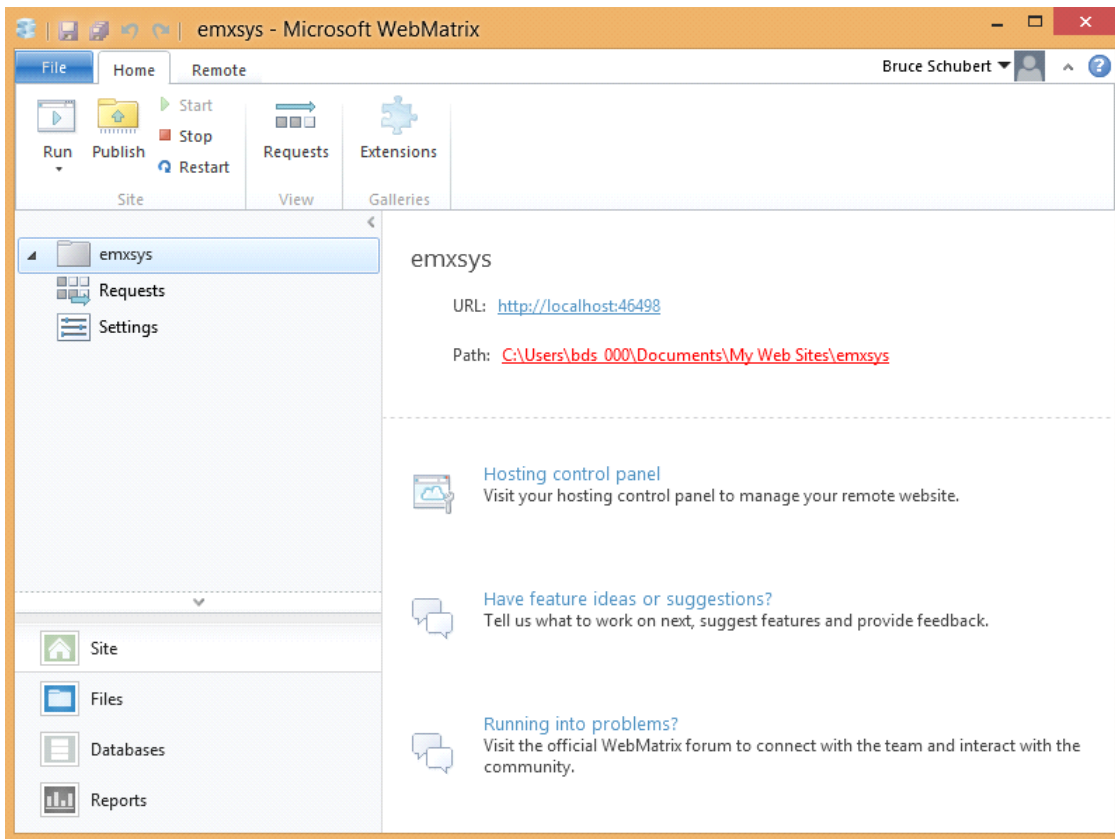
Open the recently created Azure Web site in WebMatrix3 and download to local disk.



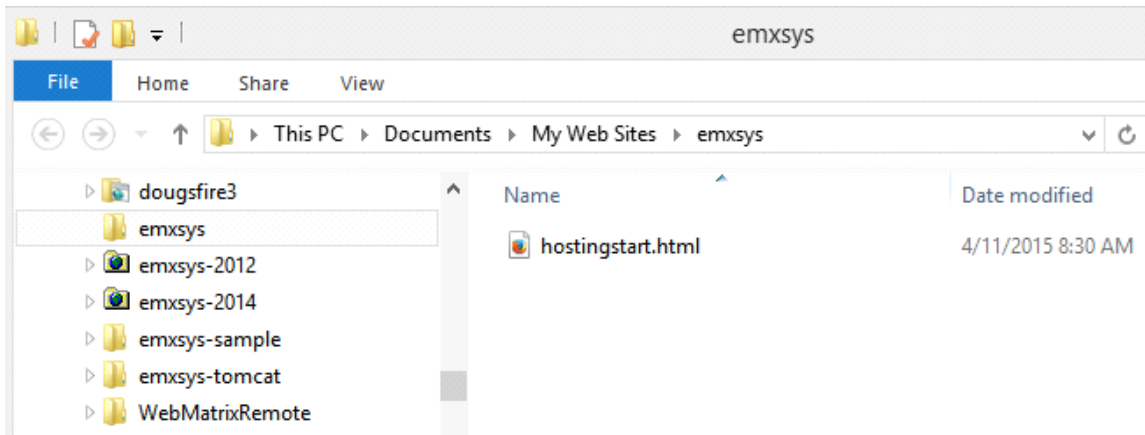
Create an empty local website.



★ Open the local website Site (creates the local file structure) via Site -> Path link.

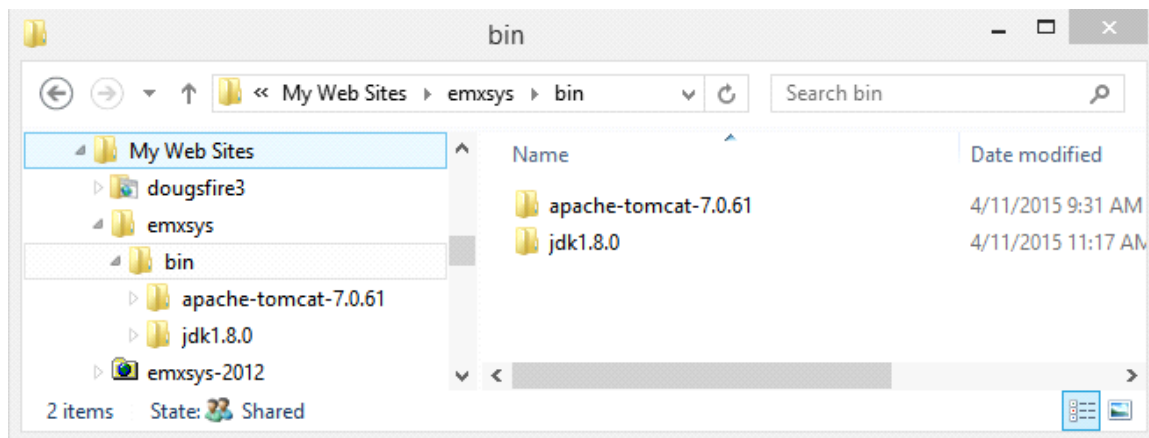


Here's the new local website with the default landing page:



## Copy the Tomcat and JRE distributions to the new Website

1. Create a **bin** folder in under My Web Sites\<website>
2. Unzip the Apache Tomcat 7 distribution to **bin**
3. Unzip the Java 8 Server JRE to **bin**: jdk1.8.0
  - a. *Reminder: jdk1.8.0\_40 did not work*



## Create the web.config file

1. Delete the **hostingstart.html** landing page.
2. Create a new **web.config** file in the website root. Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <system.webServer>
    <handlers>
      <add name="httpPlatformHandler" path="*" verb="*"
          modules="httpPlatformHandler" resourceType="Unspecified" />
    </handlers>
    <!--
    HttpPlatform configuration
    - - - - -
    processPath      - path to the executable or script that will
                      launch a process listening for Http requests.
                      examples:
                        1. %JAVA_HOME%\bin\java.exe
                        2. d:\home\site\wwwroot\bin\tomcat\bin\startup.bat
                        3. d:\home\site\wwwroot\bin\tomcat\bin\catalina.bat

    arguments        - (default="") arguments to the executable or
                      script specified in 'processPath' setting.
                      examples:
                        1. processPath="d:\home\site\wwwroot\bin\tomcat\bin\catalina.bat"
                           arguments="start"
```

```

2. processPath="%JAVA_HOME\bin\java.exe"
   arguments="-Djava.net.preferIPv4Stack=true
             -Djetty.port=%HTTP_PLATFORM_PORT%
             -Djetty.base=&quot;d:\home\site\wwwroot\bin\jetty-
distribution-9.1.0.v20131115&quot;
             -jar &quot;d:\home\site\wwwroot\bin\jetty-
distribution-9.1.0.v20131115\start.jar&quot;";

startupTimeLimit - (default=10seconds) Duration for which
                  HttpPlatformHandler will wait for the executable/script
                  to start a process listening on the port.
                  If this time limit is exceeded, httpPlatformHandler will
                  kill the process and try to launch it again
                  'startupRetryCount' times.

startupRetryCount - (default=10) Number of times HttpPlatformHandler will try
                  to launch process specified in processPath.
                  (please see startupTimeLimit for more details).

rapidFailsPerMinute - (default=10) Number of times the process specified in processPath
                  is allowed to crash per minute. If this limit is exceeded,
                  HttpPlatformHandler will stop launching the process for that minute.

requestTimeout - (default="00:02:00") Duration for which HttpPlatformHandler will
                wait for a response from the process listening on
                %HTTP_PLATFORM_PORT%.

stdoutLogEnabled - (default="true") stdout and stderr of the process specified in
                'processPath' setting will be redirected to a file specified in
                'stdoutLogFile' (see below).

stdoutLogFile - (default='d:\home\LogFiles\httpPlatformStdout.log')
               absolute file path to the file where stdout/stderr from process
               specified in processPath will be logged.

```

NOTE: %HTTP\_PLATFORM\_PORT% is a special placeholder which needs to be specified either as part of 'arguments' or part of the httpPlatform 'environmentVariable' list - This will be replaced by an internally generated PORT by HttpPlatformHandler so that the process specified in 'processPath' setting can listen on this port.

```

-->
<httpPlatform processPath="d:\home\site\wwwroot\bin\apache-tomcat-7.0.61\bin\startup.bat">
  <environmentVariables>
    <environmentVariable name="CATALINA_OPTS" value="-Dport.http=%HTTP_PLATFORM_PORT%" />
    <environmentVariable name="CATALINA_HOME" value="d:\home\site\wwwroot\bin\apache-tomcat-7.0.61" />
    <environmentVariable name="JAVA_OPTS" value="-Djava.net.preferIPv4Stack=true -
Dsun.java2d.d3d=false" />
    <!--
    <environmentVariable name="JAVA_HOME" value="d:\home\site\wwwroot\bin\jdk1.8.0" />
    -->
    <environmentVariable name="JRE_HOME" value="d:\home\site\wwwroot\bin\jdk1.8.0\jre" />
  </environmentVariables>
</httpPlatform>
</system.webServer>
</configuration>

```

## Edit the server.xml page

- The HTTP port used by the Java process is dynamically assigned. The process must use the port from the environment variable HTTP\_PLATFORM\_PORT.
- All listen ports other than the single HTTP listener should be disabled. In Tomcat, that includes the shutdown, HTTPS, and AJP ports.
- The container needs to be configured for IPv4 traffic only.

```

<?xml version='1.0' encoding='utf-8'?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

```

<http://www.apache.org/licenses/LICENSE-2.0>

```

Unless required by applicable law or agreed to in writing, software
distributed under the license is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the license for the specific language governing permissions and
limitations under the License.
-->
<!-- Note: A "Server" is not itself a "Container", so you may not
      define subcomponents such as "Valves" at this level.
      Documentation at /docs/config/server.html
-->
<Server port="-1" shutdown="SHUTDOWN">
  <Listener className="org.apache.catalina.startup.VersionLoggerListener" />
  <!-- Security listener. Documentation at /docs/config/listeners.html
  <Listener className="org.apache.catalina.security.SecurityListener" />
  -->
  <!--APR library loader. Documentation at /docs/apr.html -->
  <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
  <!--Initialize Jasper prior to webapps are loaded. Documentation at /docs/jasper-howto.html -->
  <Listener className="org.apache.catalina.core.JasperListener" />
  <!-- Prevent memory leaks due to use of particular java/javax APIs-->
  <Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener" />
  <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
  <Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener" />

  <!-- Global JNDI resources
      Documentation at /docs/jndi-resources-howto.html
  -->
  <GlobalNamingResources>
    <!-- Editable user database that can also be used by
          UserDatabaseRealm to authenticate users
    -->
    <Resource name="UserDatabase" auth="Container"
              type="org.apache.catalina.UserDatabase"
              description="User database that can be updated and saved"
              factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
              pathname="conf/tomcat-users.xml" />
  </GlobalNamingResources>

  <!-- A "Service" is a collection of one or more "Connectors" that share
        a single "Container" Note: A "Service" is not itself a "Container",
        so you may not define subcomponents such as "Valves" at this level.
        Documentation at /docs/config/service.html
  -->
  <Service name="Catalina">

    <!--The connectors can use a shared executor, you can define one or more named thread pools-->
    <!--
    <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
              maxThreads="150" minSpareThreads="4"/>
    -->

    <!-- A "Connector" represents an endpoint by which requests are received
          and responses are returned. Documentation at :
          Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
          Java AJP Connector: /docs/config/ajp.html
          APR (HTTP/AJP) Connector: /docs/apr.html
          Define a non-SSL HTTP/1.1 Connector on port 8080
    -->
    <Connector port="{port.http}" protocol="HTTP/1.1"
              connectionTimeout="20000"
              redirectPort="8443" />
    <!-- A "Connector" using the shared thread pool-->
    <!--
    <Connector executor="tomcatThreadPool"
              port="8080" protocol="HTTP/1.1"
              connectionTimeout="20000"
              redirectPort="8443" />
    -->
    <!-- Define a SSL HTTP/1.1 Connector on port 8443
          This connector uses the BIO implementation that requires the JSSE
          style configuration. When using the APR/native implementation, the
          OpenSSL style configuration is required as described in the APR/native
          documentation -->
    <!--
    <Connector port="8443" protocol="org.apache.coyote.http11.Http11Protocol"

```

```

        maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
        clientAuth="false" 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 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 -->

<!-- You should set jvmRoute to support load-balancing via AJP ie :
<Engine name="Catalina" defaultHost="localhost" jvmRoute="jvm1">
-->
<Engine name="Catalina" defaultHost="localhost">

    <!--For clustering, please take a look at documentation at:
        /docs/cluster-howto.html (simple how to)
        /docs/config/cluster.html (reference documentation) -->
    <!--
    <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
    -->

    <!-- Use the LockOutRealm to prevent attempts to guess user passwords
        via a brute-force attack -->
    <Realm className="org.apache.catalina.realm.LockOutRealm">
        <!-- This Realm uses the UserDatabase configured in the global JNDI
            resources under the key "UserDatabase". Any edits
            that are performed against this UserDatabase are immediately
            available for use by the Realm. -->
        <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
            resourceName="UserDatabase"/>
    </Realm>

    <Host name="localhost" appBase="webapps"
        unpackWARs="true" autoDeploy="true">

        <!-- SingleSignOn valve, share authentication between web applications
            Documentation at: /docs/config/valve.html -->
        <!--
        <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
        -->

        <!-- Access log processes all example.
            Documentation at: /docs/config/valve.html
            Note: The pattern used is equivalent to using pattern="common" -->
        <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
            prefix="localhost_access_log." suffix=".txt"
            pattern="%h %l %u %t &quot;%r&quot; %s %b" />

    </Host>
</Engine>
</Service>
</Server>

```

## Edit the logging.properties

Change the paths from \${catalina.base}/logs to d:/home/LogFiles

```

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the license. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

```

```

handlers = 1catalina.org.apache.juli.FileHandler, 2localhost.org.apache.juli.FileHandler,
3manager.org.apache.juli.FileHandler, 4host-manager.org.apache.juli.FileHandler, java.util.logging.ConsoleHandler

.handlers = 1catalina.org.apache.juli.FileHandler, java.util.logging.ConsoleHandler

#####
# Handler specific properties.
# Describes specific configuration info for Handlers.
#####

1catalina.org.apache.juli.FileHandler.level = FINE
1catalina.org.apache.juli.FileHandler.directory = d:/home/LogFiles
1catalina.org.apache.juli.FileHandler.prefix = catalina.

2localhost.org.apache.juli.FileHandler.level = FINE
2localhost.org.apache.juli.FileHandler.directory = d:/home/LogFiles
2localhost.org.apache.juli.FileHandler.prefix = localhost.

3manager.org.apache.juli.FileHandler.level = FINE
3manager.org.apache.juli.FileHandler.directory = d:/home/LogFiles
3manager.org.apache.juli.FileHandler.prefix = manager.

4host-manager.org.apache.juli.FileHandler.level = FINE
4host-manager.org.apache.juli.FileHandler.directory = d:/home/LogFiles
4host-manager.org.apache.juli.FileHandler.prefix = host-manager.

java.util.logging.ConsoleHandler.level = FINE
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter

#####
# Facility specific properties.
# Provides extra control for each logger.
#####

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].handlers = 2localhost.org.apache.juli.FileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].handlers =
3manager.org.apache.juli.FileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].handlers = 4host-
manager.org.apache.juli.FileHandler

# For example, set the org.apache.catalina.util.LifecycleBase logger to log
# each component that extends LifecycleBase changing state:
#org.apache.catalina.util.LifecycleBase.level = FINE

# To see debug messages in TldLocationsCache, uncomment the following line:
#org.apache.jasper.compiler.TldLocationsCache.level = FINE

```

## Publish the Website!

Verify the JVM version in the Server Status page:



The Apache Software Foundation  
<http://www.apache.org/>

**Server Status**

**Manager**

[List Applications](#) | 
 [HTML Manager Help](#) | 
 [Manager Help](#) | 
 [Complete Server Status](#)

**Server Information**

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/7.0.61	1.8.0-b132	Oracle Corporation	Windows Server 2012	6.2	amd64	RD00155D50B936	127.0.0.1

<http://emxsys.azurewebsites.net/manager/status>

4/11/2015 12:20 PM - Screen Clipping