

# TomcatOnMacOS

## Running Tomcat on Mac OS X

For an updated guide for installing Tomcat 5/6 on Mac OS X 10.6 using [MacPorts](http://serverfault.com/questions/183496/full-guide-for-installing-tomcat-on-os-x) check <http://serverfault.com/questions/183496/full-guide-for-installing-tomcat-on-os-x>

[See below for later updates to this 2004 posting]

These notes are the result of several weeks playing with different things and asking a LOT of questions on several mailing lists.

I hope these notes are of use to someone out there.

### THE ENVIRONMENT

Java is installed by default on every MacOS X installation. MacOS X 10.2 (*Jaguar*) comes with Java 1.3; MacOS X 10.3 (*Panther*) comes with Java 1.4. MacOS X 10.4 (*Tiger*) comes with Java 1.4.2 installed, but Java 1.5 can be downloaded. MacOS X 10.5 (*Leopard*) comes with Java 1.5. MacOS X 10.5 (*Snow Leopard*) comes with Java 1.6. It is possible to run Java 1.4 on MacOS X *Jaguar* but it may interfere with operation of the standard environment.

*JAVA\_HOME* is at `/Library/Java/Home`, but this is a link into a directory `/System/Frameworks/JavaJVM.Framework/Versions` which is used to switch between versions easily. There is a link `Current` and `CurrentVersion` which matches up the current Java environment with the appropriate version.

### THE SITUATION

I wanted to learn about Java Servlet Pages (JSP) on my Macintosh (with MacOS X). I installed it the first time on a Windows XP Professional system, but my development machine is a Macintosh.

### THE SOLUTION

Since I had already installed and played with a separate install of Tomcat (the version installed by Apple's XCode was out of date, as was the version I downloaded from another site), I was ready to give up!

In my case I wanted to get rid of all the other Tomcat installations and start from scratch.

#### Step 1: Download Tomcat

Download Jakarta Tomcat from the Tomcat home page, and get the binary version - I downloaded *jakarta-tomcat-5.0.27.tar.gz*.

Extract the files (unless your browser automatically extracts them).

Navigate (using the Finder) to *Macintosh HD > Library*.

If there is not already a folder in *Macintosh HD > Library* called *Tomcat*, then create one.

If there is already a *Tomcat* folder (presumably from an out-of-date installation), delete all the files already in that folder.

#### Step 2: Extract tomcat archive

Copy all the unstuffed files from the *jakarta-tomcat-5.0.27.tar.gz* download and put all the files into the *Tomcat* folder.

I just named it *Tomcat* as it saves some typing in the Terminal 😊

So the directory structure becomes:

*Macintosh HD > Library > Tomcat > <extracted files>*

#### Step 3: Configure tomcat-users.xml

Open the `/Library/Tomcat/conf/tomcat-users.xml` file with a plain text editor. The default Tomcat installation only has the basic roles and users set up.

What is configured as default is:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="role1" password="tomcat" roles="role1"/>
</tomcat-users>
```

An example of what it *should* look like follows:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <role rolename="manager"/>
  <role rolename="admin"/>
  <user username="tomcat" password="tomcat" roles="tomcat,admin,manager"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="role1" password="tomcat" roles="role1"/>
</tomcat-users>
```

In this case a role, a user and password have been set up with the name *tomcat*. You can change this later.

#### Step 4: Start Tomcat

To start Tomcat, open a shell command prompt (using, for instance, the Terminal application).

The path to Tomcat via the Finder is *Macintosh HD > Library > Tomcat*.

But to get to that directory using the Terminal, type in:

```
cd /Library/Tomcat/bin
```

Do an `ls` - you should see a file called `startup.sh`.

Any file in this directory ending in `.sh` can be executed in the terminal by putting a period and a slash before the file name (eg: `startup.sh`). The following example executes the tomcat startup script:

```
./startup.sh && tail -f ../logs/catalina.out
```

Terminal should display four lines looking something like this:

```
Using CATALINA_BASE:  /library/tomcat
Using CATALINA_HOME:  /library/tomcat
Using CATALINA_TMPDIR: /library/tomcat/temp
Using JAVA_HOME:      /Library/Java/Home
Users-Computer:/library/tomcat/bin user$
```

There are some notes and a couple of Preference Panes for automating the starting and stopping of Tomcat. After trying them all on five different computers - this is the simplest!

**Step 5: Test installation** Open a browser window, and enter <http://127.0.0.1:8080> - the default Tomcat page should open.

If you click the *Tomcat Administrator* or *Tomcat Manager* links in the upper left hand of the default Tomcat page, you will be asked for a user name and password.

As mentioned above, use *tomcat* for the user name, and *tomcat* for the password.

By : Brynley Blake (31 August 2004)

With *considerable* help from Joachim, Tracy, Tom, Samuel and Jonel from the Apple Web and Java Developer mailing lists.

## Updated for 2006

Basil Bourque says:

Today (2006-04-05) I discovered a somewhat different recipe to running Tomcat 5.5.16 on Mac OS X 10.4.6.

- Today's Tomcat (v5.5) wants to run on Java 5, but Apple's command line defaults to Java 4 (1.4.2). Surprisingly, Tomcat's script works around this.
- I did not need to set any environment variables (JAVA\_HOME).
- But I did need to fix vital file permissions in the downloaded Tomcat folder.

## Also Updated in 2010

By Pid: I removed some of the original statements about setting JAVA\_HOME as they did not match my experience of installing Tomcat on OS X. It's also certainly overkill to install an application just to be able to set an environment variable.

### (1) Upgrade to Mac OS X 10.4.6

- I do not know if 10.4.6 is required, but it sure made an easy Tomcat install.
- Apply all of Apple's Software Updates, especially the Java 5 update.

### (2) Download apache-tomcat-5.5.16

- Unzip.
- Move it to your home folder.

### (3) Fix the Unix file permissions

- Download the freeware BatChmod, a GUI wrapper around the Unix "chmod" command.

<http://macchampion.com/arbysoft/>

- Drag and drop the entire Tomcat folder onto the BatChmod icon.
- Check \*all\* the checkboxes.
- Click "Apply" and give your Administrator password.

### (4) Start Tomcat

- Launch /Applications/Utilities/Terminal.
- Navigate to the Tomcat folder (apache-tomcat-5.5.16):

```
cd ~/apache-tomcat-5.5.16
```

- Run the startup script:

```
./bin/startup.sh
```

- You should see something like this:

```
Using CATALINA_BASE:   /Users/basilbourque/apache-tomcat-5.5.16
Using CATALINA_HOME:   /Users/basilbourque/apache-tomcat-5.5.16
Using CATALINA_TMPDIR: /Users/basilbourque/apache-tomcat-5.5.16/temp
Using JRE_HOME:        /System/Library/Frameworks/JavaVM.framework/Versions/1.5/Home
```

- Notice that somehow the startup script(s) have used Java 5 on Mac OS X even though it is not the default Java runtime at the command line. I hope to parse that script later to discover how they did this.

### (5) Test Tomcat

- Launch a web browser such as Safari.
- Goto <http://127.0.0.1:8080/>
- You should see the cute Tomcat logo on the welcome page.

You are technically done at this point. But as a good practice, I also did the following additional steps.

### (6) Start the Firewall

- System Preferences > Sharing > Firewall > Start
- Also, click the Advanced button on that panel, to (a) block UDP, and (b) start logging. It is fascinating to see how quickly a computer directly on the Internet starts to get queried/tested/attacked. Open /Applications/Utilities/Console to see the log.

### (7) Enable Port-Forwarding

- Add a rule to do port-forwarding from port 80 to Tomcat's default port 8080. Type this in the Terminal:

```
sudo ipfw add 100 fwd 127.0.0.1,8080 tcp from any to any 80 in
```

Or use this nifty program, another GUI wrapper around the ipfw command line, Simple Port Forwarder: <http://www.4dresources.com/files/>

The Sharing SysPref will get cranky when you add an ipfw rule behind its back; it disables its user interface. To use the Sharing SysPref again you'll have to clear that rule, the one we added and numbered 100. To delete a rule, either read the ipfw man page, or use Simple Port Forwarder again. Quit the System Preferences program, and re-launch it to re-enable its Firewall panel.

## (8) Test Tomcat

- Launch a web browser such as Safari.
- Goto: <http://127.0.0.1/>

Notice that we left off the port number this time, so the web browser defaults to port 80. Or you can explicitly say port 80: <http://127.0.0.1:80/>

- You should see the cute Tomcat logo on the welcome page.
- To be sure it is working, and not cached by the browser:
- Try another browser.
- Click the Reload button in the toolbar of the browser.
- Choose the Reload command, such as View > Reload Page in Safari.

## (9) Walk the dog

- Enjoy an all-too-rare moment of computing satisfaction.

As best as I can recall, that is all I had to do today. I hope I am not steering you wrong. I did assume at first that Java 5 would not be used by default, so I wasted time messing with the path (\$PATH) and setting the environment variable JAVA\_HOME. But I undid those, as they failed. I then tried the simple steps above, and it worked.

I hope to refresh my own web site with this info: <http://www.BasilBourque.org/>

Tip: You can drag and drop from the Finder to the Terminal, to avoid typing file and folder names.

--Basil Bourque

## Updated for 2008

Michael Valentiner says:

Today (2008-02-03) I verified running Tomcat 6.0.14 on Mac OS X 10.5.1.

Like Basil, I did not need to set up JAVA\_HOME and I did need to fix Unix file permissions.

Unlike Basil, the System Preferences > Sharing > Firewall has moved to System Preferences > Security > Firewall. I was able to enable logging, but haven't figured out how to block UDP.

## Updated for 2009

Chris Latko says:

To have Tomcat 6.0.x launch on boot on Mac OS X 10.5.x, you need to add a LaunchDaemon. Create a file called org.apache.tomcat.plist in /Library /LaunchDaemons/ with the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Disabled</key>
    <false/>
    <key>Label</key>
    <string>com.apache.tomcat</string>
    <key>ProgramArguments</key>
    <array>
        <string>/Library/Tomcat/bin/startup.sh</string>
    </array>
    <key>RunAtLoad</key>
    <true/>
</dict>
</plist>
```

## Updated for 2009

Greg Woolsey says:

For Tomcat 5.5.x (and probably 6.0, as the scripts don't appear to have changed much) you need a slightly modified plist, that calls catalina.sh in a manner compatible with the requirements of launchd: <http://developer.apple.com/MacOSX/launchd.html>. Specifically, the script must be run with the "run" parameter instead of "start", as start backgrounds the Java process and exits the script. This causes launchd to think the service is done, and kills the java process as part of it's cleanup.

My plist assumes Tomcat is installed in /Library/Tomcat/tomcat-version, with a symbolic link called "Home" pointing to the version to use.

Save this file as /Library/LaunchDaemons/org.apache.tomcat.plist:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Disabled</key>
    <false/>
    <key>Label</key>
    <string>org.apache.tomcat</string>
    <key>ProgramArguments</key>
    <array>
        <string>/Library/Tomcat/Home/bin/catalina.sh</string>
        <string>run</string>
    </array>
    <key>RunAtLoad</key>
    <true/>
</dict>
</plist>
```

If you have a web app that doesn't want to shut down nicely (my company's product is one), then to avoid the need to force quit Tomcat just to turn off your Mac, use a startup script instead of a daemon plist.

Add this line to /etc/hostconfig as root:

**TOMCAT=**~~YES~~

Then, as root, create /Library/StartupItems/Tomcat, writable only by the owner (root).

Inside this, create two files - Tomcat (script) and StartupParameters.plist. Again, make them writable only by root, and the script executable by everyone.

Contents of StartupParameters.plist:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Description</key>
    <string>Tomcat Server</string>
    <key>OrderPreference</key>
    <string>Late</string>
    <key>Provides</key>
    <array>
        <string>Tomcat</string>
    </array>
</dict>
</plist>
```

Contents of the Tomcat script file:

```

#!/bin/sh
#
# /Library/StartupItems/Tomcat/Tomcat
#
# A script to automatically start up Tomcat on system bootup
# for Mac OS X. This is actually just a wrapper script around
# the standard catalina.sh script, which is included in
# the distribution.
#

# Suppress the annoying "$1: unbound variable" error when no option
# was given
if [ -z $1 ] ; then
    echo "Usage: $0 [start|stop|restart] "
    exit 1
fi

# Source the common setup functions for startup scripts
test -r /etc/rc.common || exit 1
. /etc/rc.common

# The path to the catalina.sh script.
# The currently used version is in /Library/Tomcat/Home/bin
SCRIPT="/Library/Tomcat/Home/bin/catalina.sh"

# file to hold the process ID on start so it can be killed by stop.
export CATALINA_PID="/Library/Tomcat/Home/server.pid"

StartService ()
{
    if [ "${TOMCAT:=NO}" = "YES" ] ; then
        ConsoleMessage "Starting Tomcat server"
        $SCRIPT start > /dev/null 2>&1
    fi
}

StopService ()
{
    ConsoleMessage "Stopping Tomcat server"
    $SCRIPT stop -force > /dev/null 2>&1
}

RestartService ()
{
    ConsoleMessage "Restarting Tomcat server"
    StopService
    StartService
}

if test -x $SCRIPT ; then
    RunService "$1"
else
    ConsoleMessage "Could not find Tomcat control script!"
fi

```