

## Install JBoss 7.1 on CentOS 6

This post will cover installing JBoss 7.1.1 on CentOS 6.x.

We'll also set up JBoss to run as a service, as well as set up access to the management console

Finally, we will look at how run JBoss on port 80 or, alternatively, placing JBoss behind Apache.

In this post, we will set up the JBoss 7.1.1 in Standalone mode. In a subsequent post we will look at setting up JBoss 7.1.1 in Domain mode.

### Step 1: Install the Java 6 or 7

Firstly, we will need to install Java.

JBoss 7.1.1 will work with JDK 6 or JDK 7.

I'm using JDK 7, update 5.

You can download the JDK here: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

The instructions below will also work with JDK 6, you just need to change the file names accordingly.

Start by creating a directory /usr/java.

```
1. [root@dev2 ~]# mkdir /usr/java
```

Download jdk-7u5-linux-x64.tar.gz (or latest) and save to /usr/java directory you created above.

```
1. [root@dev2 java]# ls
2. jdk-7u5-linux-x64.tar.gz
```

Extract it:

```
1. [root@dev2 java]# tar -zxf jdk-7u5-linux-x64.tar.gz
```

This will create the directory, /usr/java/jdk1.7.0\_05, this will be our JAVA\_HOME.

```
1. [root@dev2 java]# ls
2. jdk1.7.0_05 jdk-7u5-linux-x64.tar.gz
3. [root@dev2 java]# cd jdk1.7.0_05
4. [root@dev2 jdk1.7.0_05]# pwd
5. /usr/java/jdk1.7.0_05
```

*Note: If you decided to use JDK 6 rather than 7 as we did above, simply save the JDK 6 bin file to /opt (or another location), then navigate to /usr/java and issue: 'sh /opt/jdk-6u33-linux-x64.bin' as shown below (substitute whichever version you downloaded). This will create a JAVA Home of /usr/java/jdk1.6.0.33*

```
1. [root@dev2 ~]#cd /usr/java
```

2. [root@dev2 java]# sh /opt/jdk-6u33-linux-x64.bin

## Step 2: Download and Install JBoss 7.1.1 Application Server

Change to the /usr/share directory:

1. [root@dev2 jdk1.7.0\_05]# cd /usr/share

Download jboss-as-7.1.1.Final.zip at <http://www.jboss.org/jbossas/downloads> and save it to /usr/share. Or, use wget:

1. [root@dev2 share]# wget http:

Unzip the file:

1. [root@dev2 share]# unzip -q jboss-as-7.1.1.Final.zip

Rename jboss-as-7.1.1.Final to jboss-as. This isn't strictly necessary, but it will save you the bother of changing the start up script later.

1. [root@dev2 share]# mv jboss-as-7.1.1.Final jboss-as

Our JBOSS\_HOME is '/usr/share/jboss-as'.

1. [root@dev2 share]# cd jboss-as

2. [root@dev2 jboss-as]# pwd

3. /usr/share/jboss-as

## Step 3: Create the user jboss, who will own and run JBoss

Since we will want to run JBoss as a non-root user with minimal privileges, we'll create a user, jboss, who will own the JBoss files and JBoss will run under his account.

To do this, do the following.

Create a new group, jboss, and then create the user jboss and add the user to the jboss group.

1. [root@dev2 share]# groupadd jboss

2. [root@dev2 share]# useradd -s /bin/bash -g jboss jboss

Change ownership of the JBoss home directory, /usr/share/jboss-as so all files are owned by the user jboss we created.

1. [root@dev2 share]# chown -Rf jboss.jboss /usr/share/jboss-as/

## Step 4: Put Java into the path of jboss and root

Now, we need to put Java into the path of the users jboss and root (as well as any other users you like)

The JAVA\_HOME is where we installed the JDK above: /usr/java/jdk1.7.0\_05

Add the following to the .bash\_profile of the user jboss and the user root:

1. JAVA\_HOME=/usr/java/jdk1.7.0\_05

2. export JAVA\_HOME

3. `PATH=$JAVA_HOME/bin:$PATH`

4. `export PATH`

As root, issue `././bash_profile` as below to put JAVA into the path of root immediately.

1. `[root@dev2 ~]# ././bash_profile`

Check that JAVA is now in the path of root by issuing `'java -version'` as below.

1. `[root@dev2 ~]# java -version`

2. `java version "1.7.0_05"`

3. `Java(TM) SE Runtime Environment (build 1.7.0_05-b06)`

4. `Java HotSpot(TM) 64-Bit Server VM (build 23.1-b03, mixed mode)`

5. `[root@dev2 ~]#`

We need to have JAVA in the path of user root for when we add a Management User for JBoss console in Step 8 below.

Finally, switch to the user jboss and issue `'java -version'` to verify that Java is now in the path of user jboss.

1. `[root@dev2 ~]# su - jboss`

2. `[jboss@dev2 ~]$ java -version`

3. `java version "1.7.0_05"`

4. `Java(TM) SE Runtime Environment (build 1.7.0_05-b06)`

5. `Java HotSpot(TM) 64-Bit Server VM (build 23.1-b03, mixed mode)`

6. `[jboss@dev2 ~]$`

### **Step 5: Create a start/stop/restart script for JBoss.**

To create our JBoss script, we will copy the `jboss-as-standalone.sh` script located under `/usr/share/jboss-as/bin/init.d`

As root, copy `jboss-as-standalone.sh` to `/etc/init.d` and rename it `jboss`

.

1. `[root@dev2 ~]# cd /usr/share/jboss-as/bin/init.d`

2. `[root@dev2 bin]# cp jboss-as-standalone.sh /etc/init.d/jboss`

The only change I needed to make two changes to the script

The first was to change line #5 below from it's original `'# chkconfig: - 80 20'` to `'# chkconfig: 234 80 20'` The second was to set the `JBOSS_USER`.

To set the `JBOSS_USER`, add the following lines to script:

```
JBOSS_USER=jboss
```

```
export JBOSS_USER
```

You can add the lines just under `'export JAVA_HOME'` on line 18 or so below. So the beginning of your script will look like this:

```

1. #!/bin/sh
2. #
3. # JBoss standalone control script
4. #
5. # chkconfig: 234 80 20
6. # description: JBoss AS Standalone
7. # processname: standalone
8. # pidfile: /var/run/jboss-as/jboss-as-standalone.pid
9. # config: /etc/jboss-as/jboss-as.conf
10. # Source function library.
11. . /etc/init.d/functions
12. # Load Java configuration.
13. [ -r /etc/java/java.conf ] && . /etc/java/java.conf
14. export JAVA_HOME
15. JBOSS_USER=jboss
16. export JBOSS_USER
17. # Load JBoss AS init.d configuration.
18. if [ -z "$JBOSS_CONF" ]; then
19.     JBOSS_CONF="/etc/jboss-as/jboss-as.conf"
20. fi
21. [ -r "$JBOSS_CONF" ] && . "${JBOSS_CONF}"
22. # Set defaults.
23. if [ -z "$JBOSS_HOME" ]; then
24.     JBOSS_HOME=/usr/share/jboss-as
25. fi
26. export JBOSS_HOME

```

### **Step 6: Run JBoss as a Service.**

To run JBoss as a service and enable start up at boot, make the script we created above executable and add it to our chkconfig so it starts at boot.

1. [root@dev2 init.d]# chmod 755 jboss
2. [root@dev2 init.d]# chkconfig --add jboss
3. [root@dev2 init.d]# chkconfig --level 234 jboss on

We should now be able to Start, Stop, and Restart JBoss as a service.

Start JBoss (JBoss can take some time to start, but it is faster than JBoss 6).:

1. [root@dev2 init.d]# service jboss start

2. Starting jboss-as: [ OK ]

3. [root@dev2 init.d]#

Stop JBoss:

1. [root@dev2 init.d]# service jboss stop

2. Stopping jboss-as: \*\*\* JBossAS process (25794) received TERM signal \*\*\*

3. [ OK ]

### Step 7: Change bind address to make JBoss accessible.

By default, JBoss 7.1.1 is bound to the loopback IP of 127.0.0.1, so if we want to make it available on the web, we need to change this.

Locate standalone.xml under /usr/share/jboss-as/standalone/configuration/.

Open standalone.xml in vi or a text editor and look for the public interfaces node as shown below.

```
<interface name="public">
<inet-address value="${jboss.bind.address:127.0.0.1}"/>
</interface>
```

To make JBoss publicly accessible, change 127.0.0.1 to either 0.0.0.0 to allow access on all interfaces or to your public IP.

So, for example, if your public IP is 173.194.35.177, you would change it as so:

```
1. <interface name="management">
2.     "${jboss.bind.address.management:127.0.0.1}"/>
3. </interface>
4. <interface name="public">
5.     "${jboss.bind.address:173.194.35.177}"/>
6. </interface>
7. this if the jacob subsystem is added -->
8. <interface name="unsecure">
9.     ~ Used for IIOP sockets in the standard configuration.
10.    ~ To secure JacORB you need to setup SSL
11.    -->
12.     "${jboss.bind.address.unsecure:127.0.0.1}"/>
13. </interface>
```

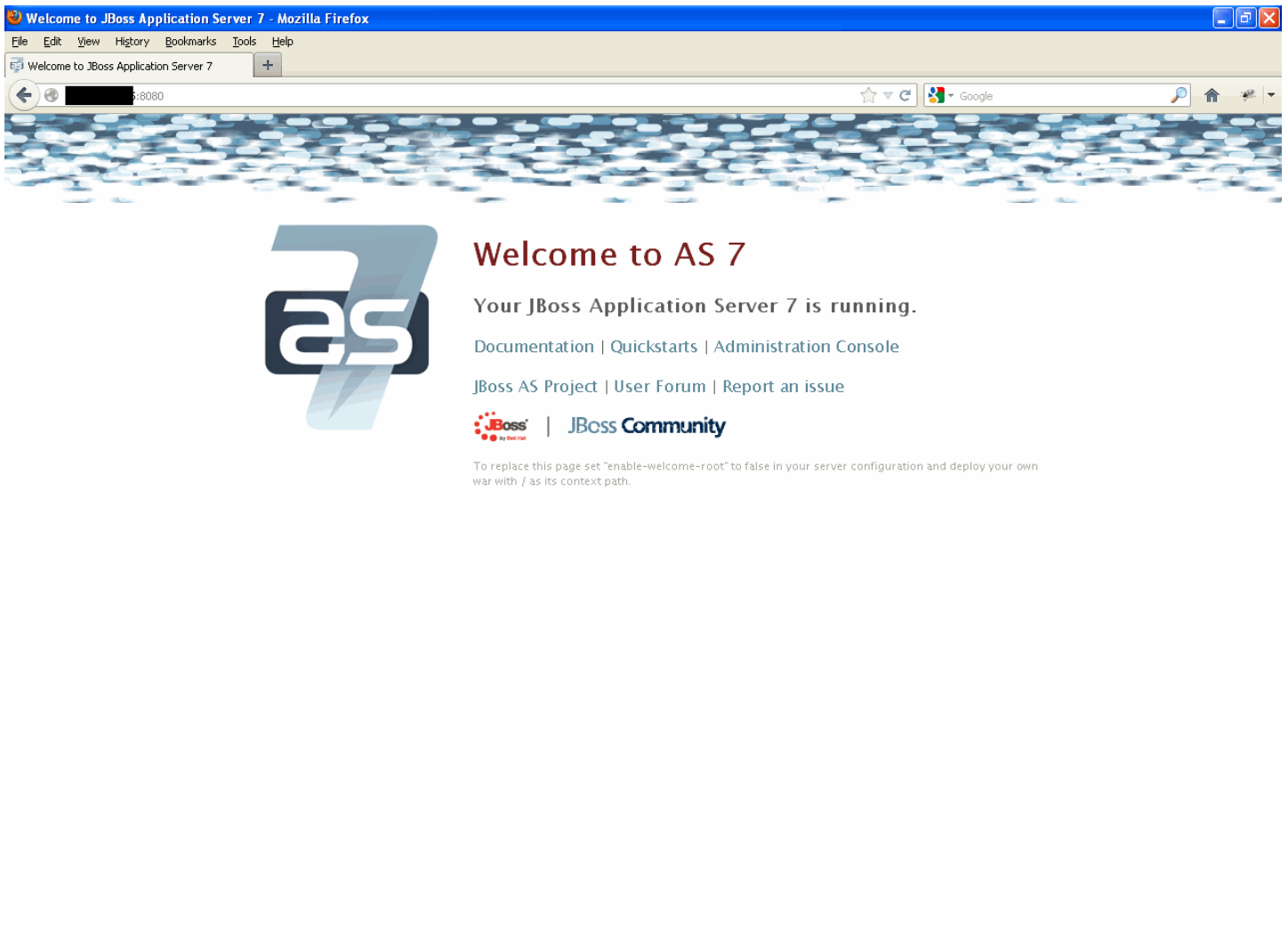
Again, if you wish to have JBoss publicly accessible on all interfaces, use 0.0.0.0 in place of your IP above.

Save your changes, start up JBoss, and check it is publicly accessible via <http://yourIP:8080>

1. [root@dev2 init.d]# service jboss start

2. Starting jboss-as: [ OK ]

Woo-hoo! JBoss 7:



### Step 8: Accessing the JBoss 7 Admin Console and adding a Management User

If you try to access the JBoss Admin Console via the link on the home page (or by going directly to <http://YourIP:9990>, you will be greeted with the following:



## Welcome to AS 7

Your JBoss Application Server 7 is running.

However you have not yet added any users to be able to access the admin console.

To add a new user execute the add-user.sh script within the bin folder of your AS 7 installation and enter the requested information.

By default the realm name used by AS 7 is "ManagementRealm" this is already selected by default.

```
darran@localhost:~/src/jbossas7/jboss-as/build/target/jboss-as-7.1.0.Alpha2-SNAPSHOT/bin
File Edit View Search Terminal Help
[darran@localhost bin]$ ./add-user.sh
Enter details of new user to add.
Realm (ManagementRealm) :
Username : myNewUser
Password :
Re-enter Password :
About to add user 'myNewUser' for realm 'ManagementRealm'
Is this correct yes/no? yes
Added user 'myNewUser' to file '/home/darran/src/jbossas7/jboss-as/build/target
/jboss-as-7.1.0.Alpha2-SNAPSHOT-standalone/configuration/mgmt-users.properties'
Added user 'myNewUser' to file '/home/darran/src/jbossas7/jboss-as/build/target
/jboss-as-7.1.0.Alpha2-SNAPSHOT-standalone/configuration/mgmt-users.properties'
[darran@localhost bin]$
```

After you have added the user follow this link to [Try Again](#).

While the error page tells you how to add a user, you will also need to update the standalone.xml as we did earlier if you want to allow access over your public IP.

Let's start with adding a Management user as shown on the error page:

As root, navigate to /usr/share/jboss-as/bin

1. [root@dev2 etc]# cd /usr/share/jboss-as/bin

Issue './add-user.sh' to run the script to add a Management user. Follow the prompts. Some defaults are provided.

1. [root@dev2 bin]# ./add-user.sh
2. What type of user do you wish to add?
3. a) Management User (mgmt-users.properties)
4. b) Application User (application-users.properties)
5. (a):
6. Enter the details of the new user to add.
7. Realm (ManagementRealm) :
8. Username : david
9. Password :
10. Re-enter Password :
11. About to add user 'david' for realm 'ManagementRealm'
12. Is this correct yes/no? yes

13. Added user 'david' to file '/usr/share/jboss-as/standalone/configuration/mgmt-users.properties'

While you can now access the Admin console on localhost (127.0.0.1), if you want to access it publicly, we need to update `standalone.xml` under `/usr/share/jboss-as/standalone/configuration/` as we did earlier in Step 7.

```
<interface name="management">
<inet-address value="$jboss.bind.address.management:127.0.0.1"/>
</interface>
```

```

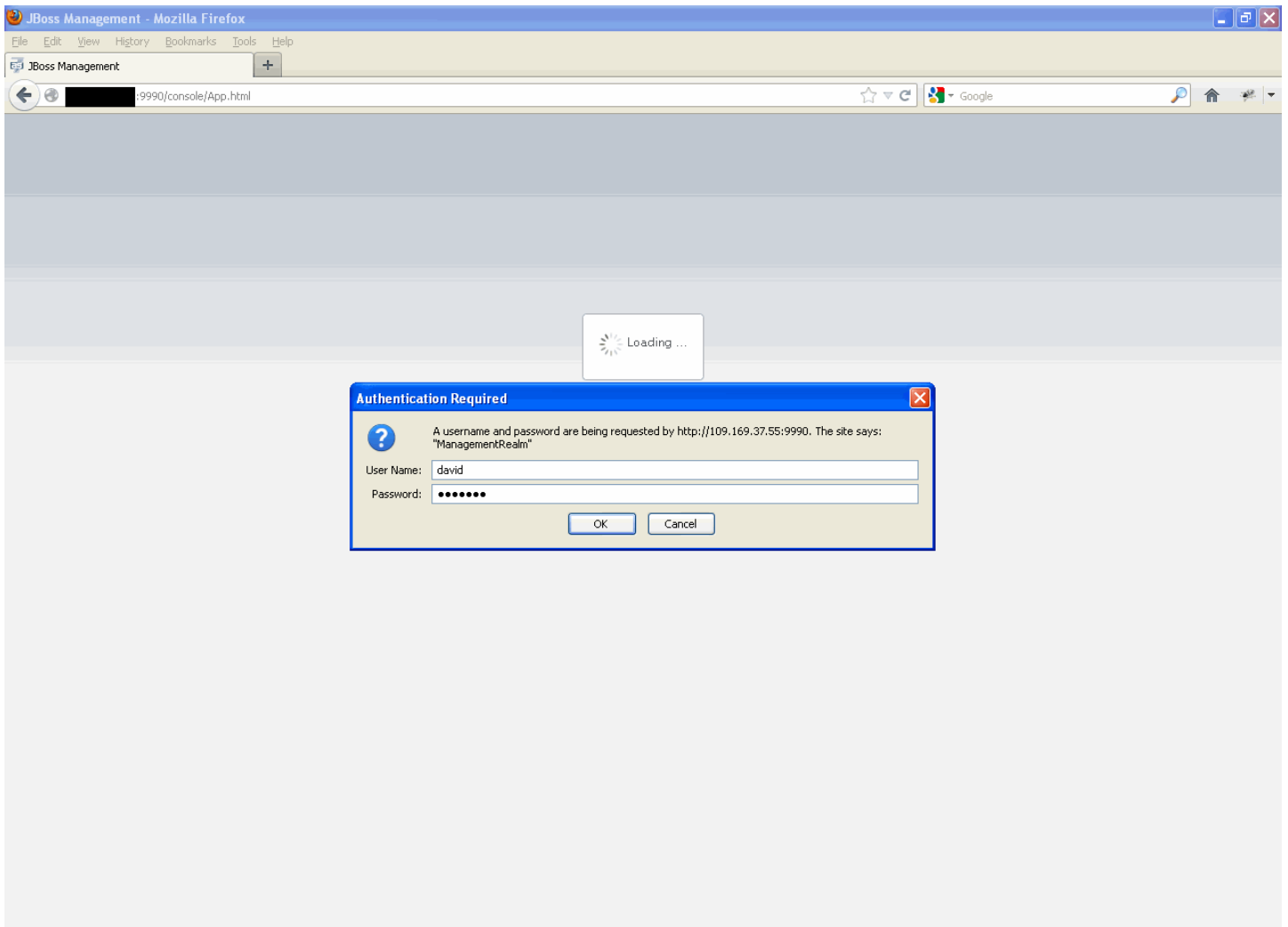
1.      <interface name="management">
2.          "${jboss.bind.address.management:173.194.35.177}"/>
3.      </interface>
4.      <interface name="public">
5.          "${jboss.bind.address:173.194.35.177}"/>
6.      </interface>
7.      this if the jacobd subsystem is added -->
8.      <interface name="unsecure">
9.          ~ Used for IIOP sockets in the standard configuration.
10.         ~           To secure JacORB you need to setup SSL
11.         -->
12.         "${jboss.bind.address.unsecure:127.0.0.1}"/>
13.     </interface>

```

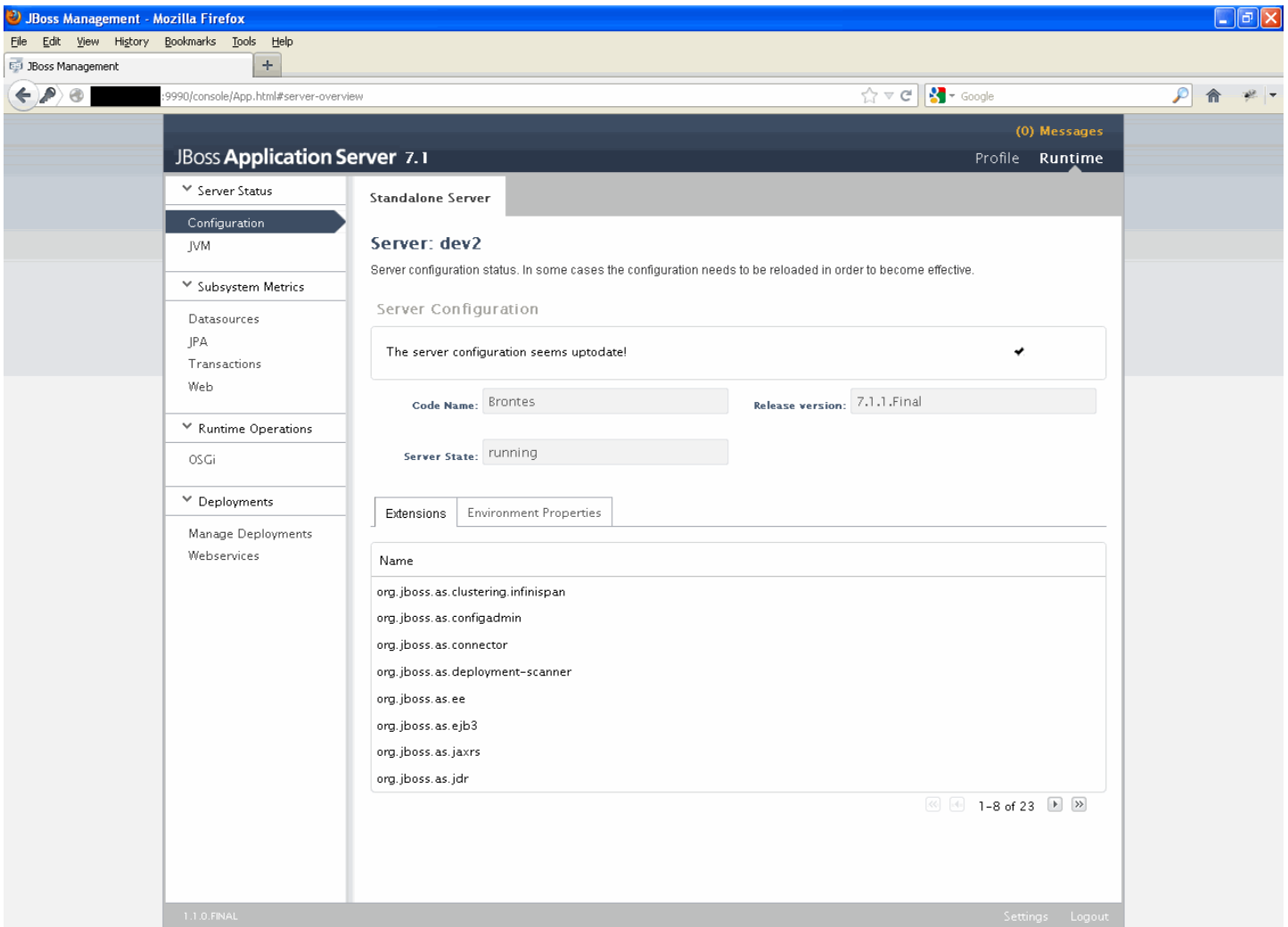
1. [root@dev2 init.d]# service jboss stop
2. Stopping jboss-as: \*\*\* JBossAS process (25794) received TERM signal \*\*\*
3. [ OK ]
4. [root@dev2 init.d]# service jboss start
5. Starting jboss-as: [ OK ]

Now, navigate back to `http://YourIP:9990` and you should be prompted for the Management user credentials you just created:





Below, we are now logged in to the Admin console as the Management user we created:



## Step 9 (Optional): Running JBoss on Port 80 .

To run services below port 1024 as user other than root, you can use port forwarding.

You can do this by adding the following to your IP tables:

1. `[root@sv2 ~]# iptables -t nat -A PREROUTING -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 8080`
2. `[root@sv2 ~]# iptables -t nat -A PREROUTING -p udp -m udp --dport 80 -j REDIRECT --to-ports 8080`

Be sure to save and restart your IP Tables.

## Step 10 (Optional): Putting Apache in Front of JBoss .

As an alternative to running JBoss on port 80, if you have Apache in front of JBoss, you can use mod\_proxy as well as ajp connector to map your domain to your JBoss applications using an Apache vhost as shown below:

While both Tomcat and JBoss have improved their standalone performance, I still prefer to have Apache in front for a number of reasons.

Note that when using proxy, if you will be proxy to localhost (which you should be), you will need to set your public interface bind address to either 0.0.0.0 (for all interfaces) or to 127.0.0.1 for localhost only.

1. ServerAdmin admin@domain.com
2. ServerName domain.com

3. ServerAlias www.domain.com
4. ProxyRequests Off
5. ProxyPreserveHost On
6. Order allow,deny
7. Allow from all
8. ProxyPass / http:
9. ProxyPassReverse / http:
10. ErrorLog logs/domain.com-error\_log
11. CustomLog logs/domain.com-access\_log common

Alternatively, with the AJP connector enabled, you can use ajp as well:

1. ServerAdmin admin@domain.com
2. ServerName domain.com
3. ServerAlias www.domain.com
4. ProxyRequests Off
5. ProxyPreserveHost On
6. Order allow,deny
7. Allow from all
8. ProxyPass / ajp:
9. ProxyPassReverse / ajp:
10. ErrorLog logs/domain.com-error\_log
11. CustomLog logs/domain.com-access\_log common

The AJP connector is NOT enabled by default. To enable the AJP connector:

1. Log into the Admin console
2. Click Profile on the top right.
3. On the left menu, go to Web > Servlet/HTTP.
3. Click on the "add" button at right.

JBoss Application Server 7.1

(0) Messages

ProfileRuntime

▼ Profile

+

 Core
 

+

 Connector
 

JCA
 

Datasources

Resource Adapters

Mail

+

 Container
 

+

 Security
 

+

 Web
 

Servlet/HTTP

Web Services

+

 OSGi
 

+

 Infinispan

▼ General Configuration

Interfaces

Socket Binding

System Properties

Servlet

Servlet/HTTP Configuration

The configuration of the jboss.web subsystem.

Edit

Disabled?: false

Development?: false

▶ Advanced

Connectors

Virtual Servers

Add

Remove

Name	Protocol	Enabled?
▶ http	HTTP/1.1	✓

Edit

Name: http

Socket Binding: http

Protocol: HTTP/1.1

Scheme: http

Enabled?: true

1.1.0.FINAL

SettingsLogout

4. Enter the following in the Create Connector dialogue box as shown below.

5. Click Save.

Create Connector

Name:

ajp

Socket Binding:

ajp

Protocol:

AJP/1.3

Scheme:

http

Enabled?:

☒

Save

Cancel

The AJP connector is now enabled.

JBoss Application Server 7.1

Added Connector ajp (1) Messages

Profile Runtime

Profile

Core

Connector

JCA

Datasources

Resource Adapters

Mail

Container

Security

Web

Servlet/HTTP

Web Services

OSGi

Infinispan

General Configuration

Interfaces

Socket Binding

System Properties

Servlet

### Servlet/HTTP Configuration

The configuration of the jboss.web subsystem.

Edit

Disabled?: false

Development?: false

Advanced

Connectors Virtual Servers

Add Remove

Name	Protocol	Enabled?
ajp	AJP/1.3	✓
http	HTTP/1.1	✓

Edit

Name: http

Socket Binding: http

Protocol: HTTP/1.1

Scheme: http

Enabled?: true

1.1.0.FINAL

Settings Logout

One final note above the vhost examples above (proxy and ajp).

In both vhost examples above, we are "mapping" the domain to the root.

If we wish to map to an application such as domain.com/myapp, we can add some rewrite as shown below.

This will rewrite all requests for domain.com to domain.com/myapp.

1. ServerAdmin admin@domain.com
2. ServerName domain.com
3. ServerAlias www.domain.com
4. RewriteEngine On
5. RewriteRule ^/\$ myapp/ [R=301]
6. ProxyRequests Off
7. ProxyPreserveHost On
8. Order allow,deny

9. Allow from all
10. ProxyPass / ajp:
11. ProxyPassReverse / ajp:
12. ErrorLog logs/domain.com-error\_log
13. CustomLog logs/domain.com-access\_log common

[JBoss 7.1.1 Docs](#)

## WildFly 9 and 10 Hosting

Private Instance  
PostgreSQL 9.5 and MySQL 5.6  
Unlimited Domains

**\$25 per month!**

intoGres

Posted at [05:14PM Jul 23, 2012](#) by David in JBoss | | Tags: [java](#) [centos](#) [jboss](#) | Export to:

