

# Lab Manual

## JBoss AS7 (EAP 6)

### 1. Software and Support files provided in JBoss7Lab Folder

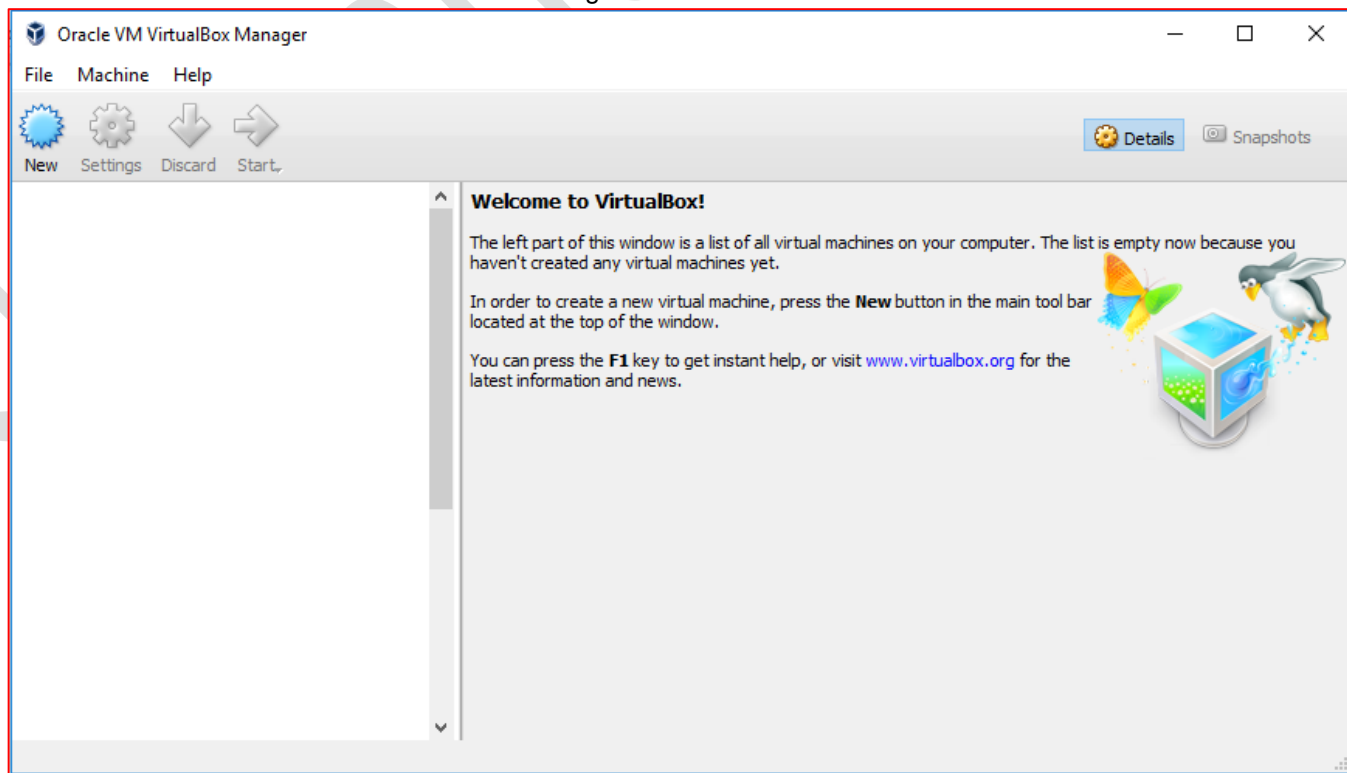
- VM
- Examples
- JBossClass
- CLI Recipes-v28-20180418\_1000.pdf
- devstudio-11.2.0.GA-installer-standalone.jar
- jboss-eap-6.4.0.zip
- jdk-8u77-linux-i586.tar.gz
- npp.6.9.2.Installer.exe
- putty.exe
- VirtualBox-5.1.24-117012-Win.exe
- vmware-player-12.1.1.exe
- WinSCP-5.13-Setup.exe
- Win-Setup (Contains files for windows OS)
- mysql-connector-java-5.1.26.jar

### LAB 1: Install VirtualBox Software

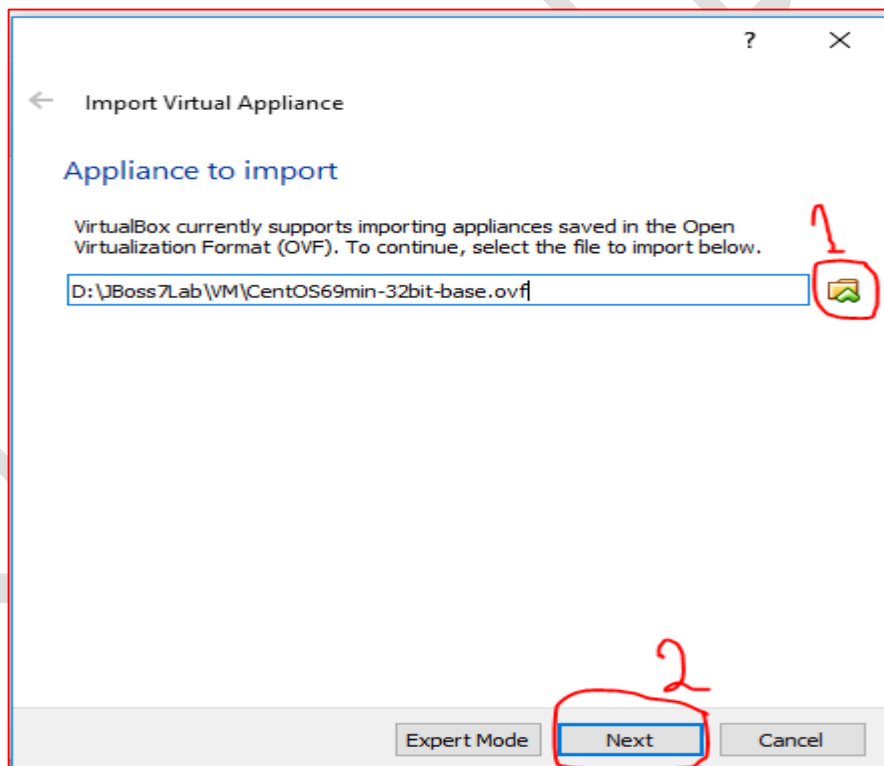
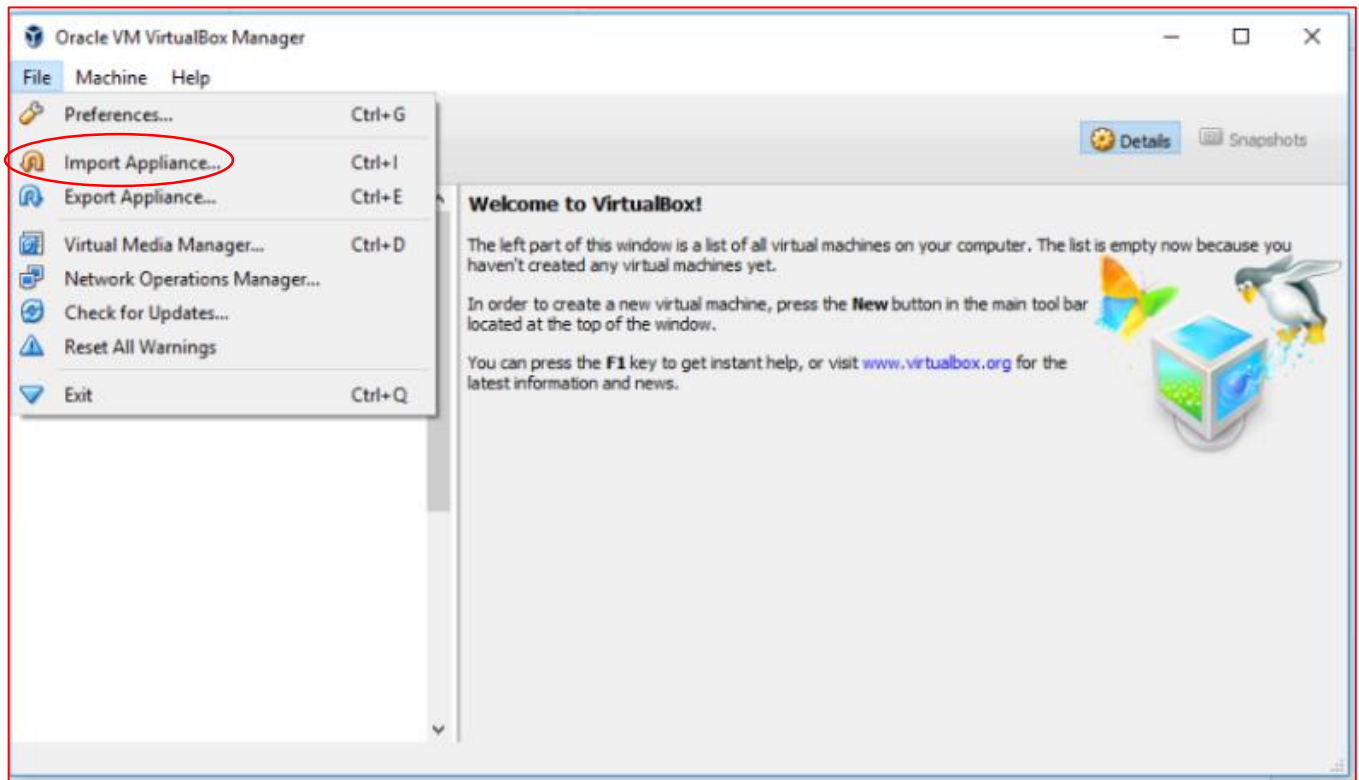
- To install VirtualBox it is necessary to enable virtualisation in your computer (Verify and Enable it)
- Double click "VirtualBox-5.1.24-117012-Win.exe" and follow the On Screen instructions
- Once the installation is complete then start the VirtualBox Manager. See the following Screen



Double click on  It starts VirtualBox Manager as below

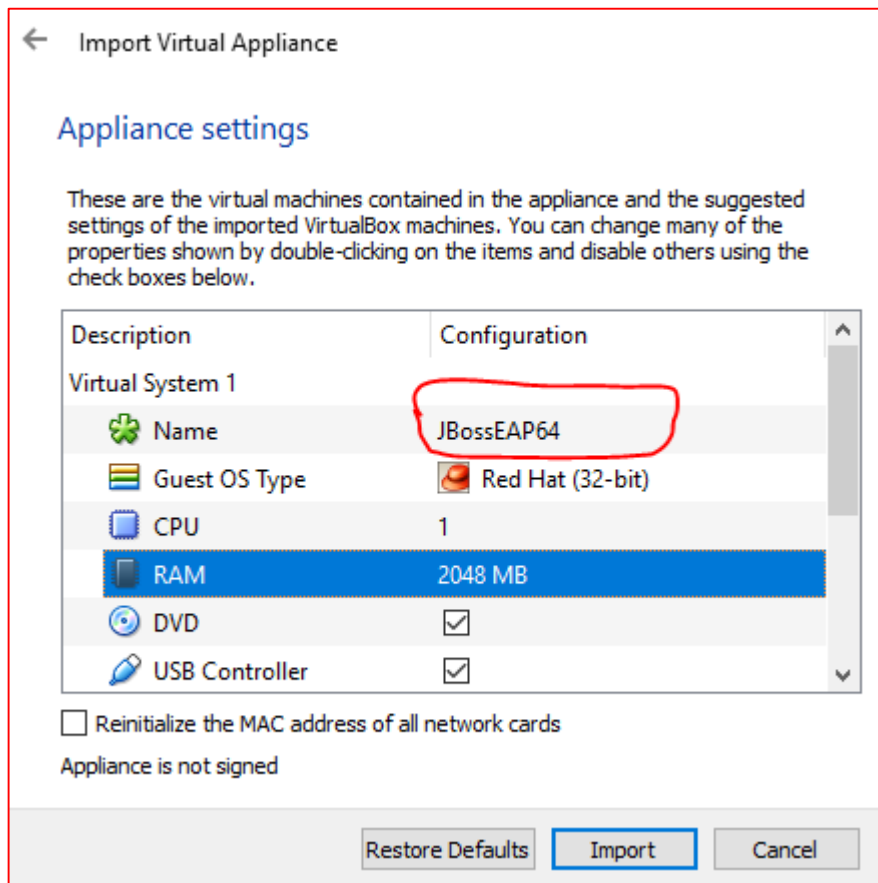


Click on File>Import Appliance and navigate to JBoss7Lab/VM folder and select the .OVF file and click on Next button

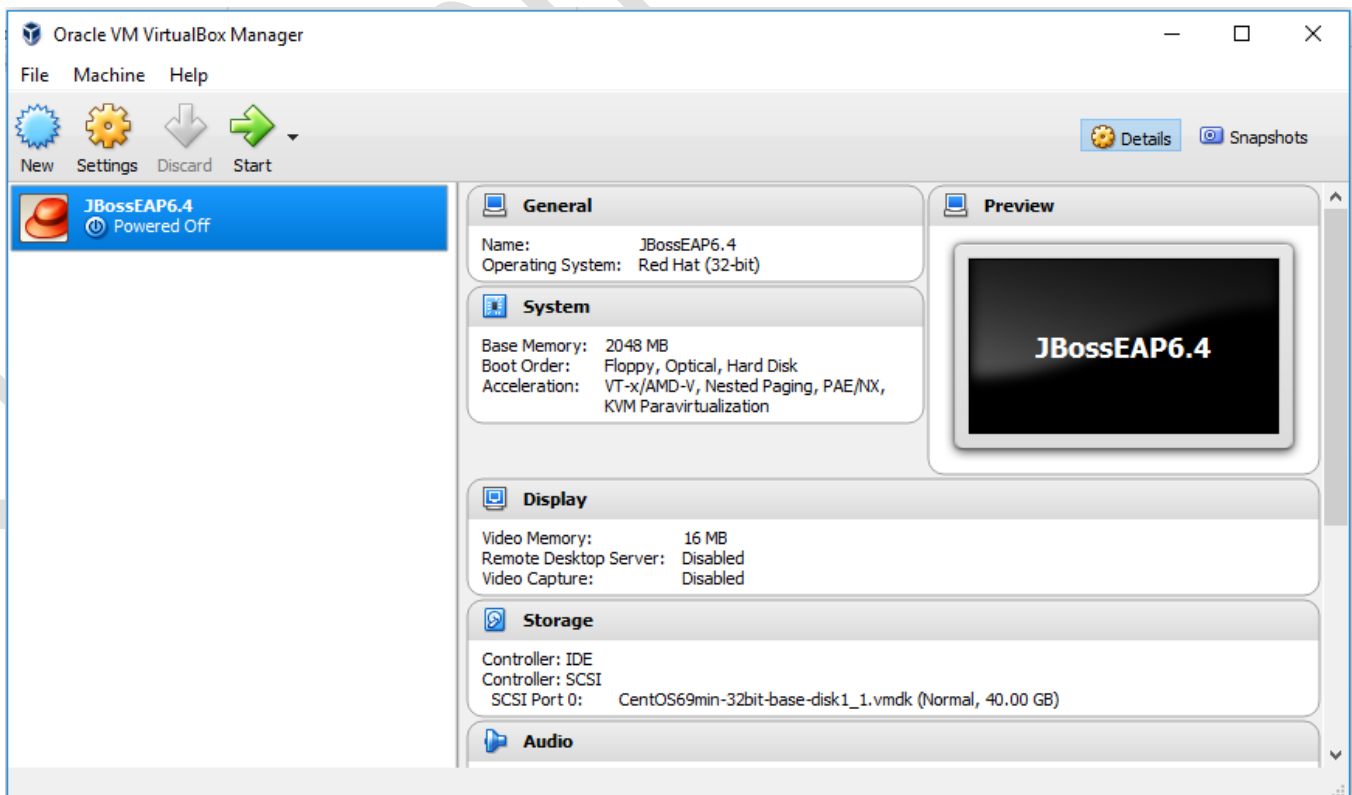


In the Next Screen Change the name of the Linux VM to “JBossEAP64” and then click on

Import

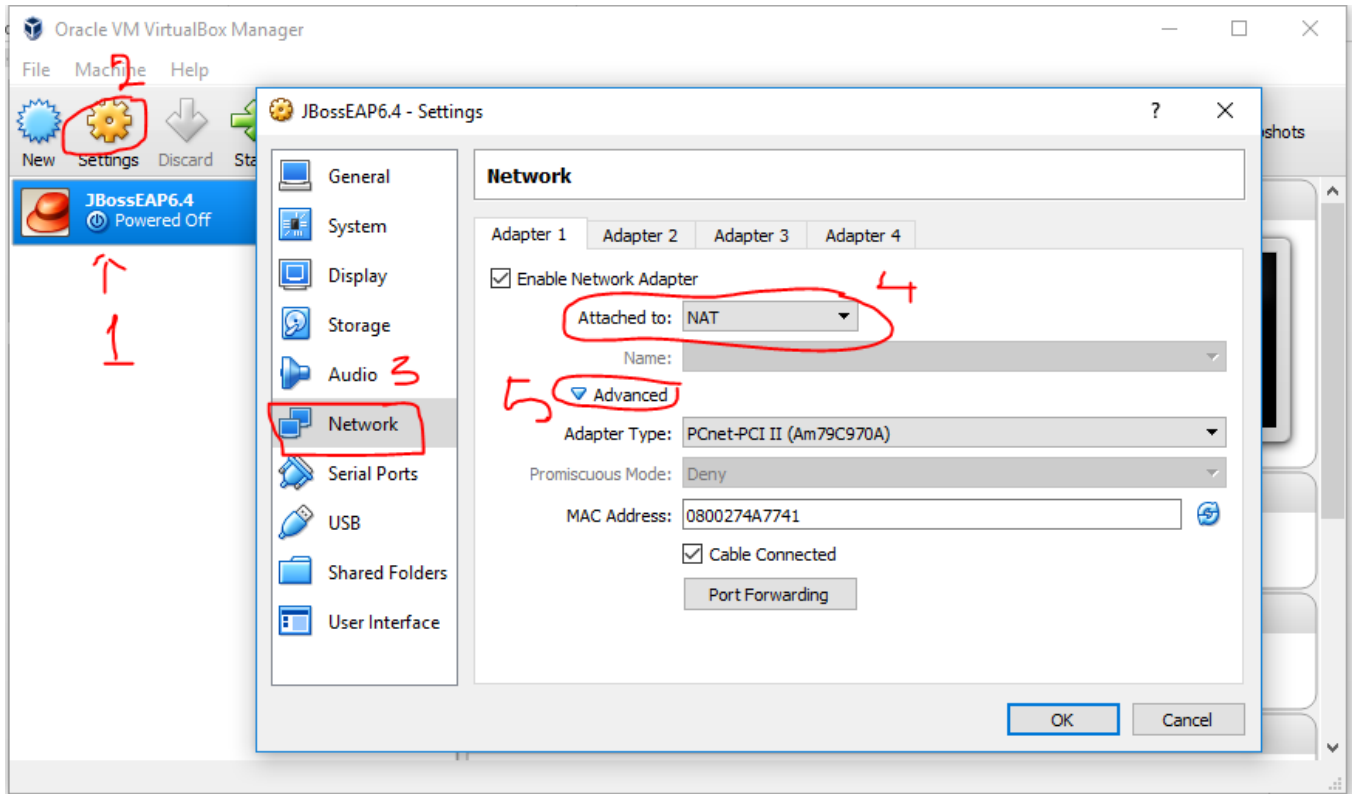


Once it is imported your screen should look like the following:

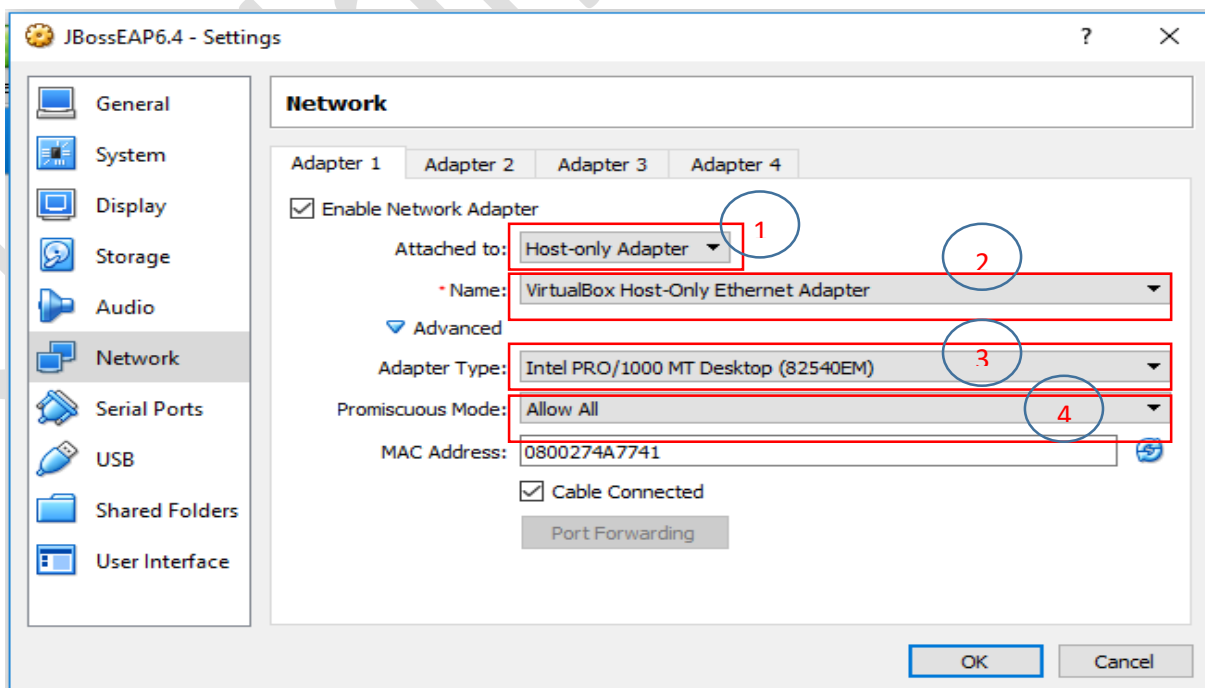


## Modify the Network Adapter

1. Click On the Virtual Machine (JBossEAP64) and then click on the gear button labelled as **settings**. The settings page for the VM opens. Select Network from left Pane(3). Expand the Advanced Option (5)



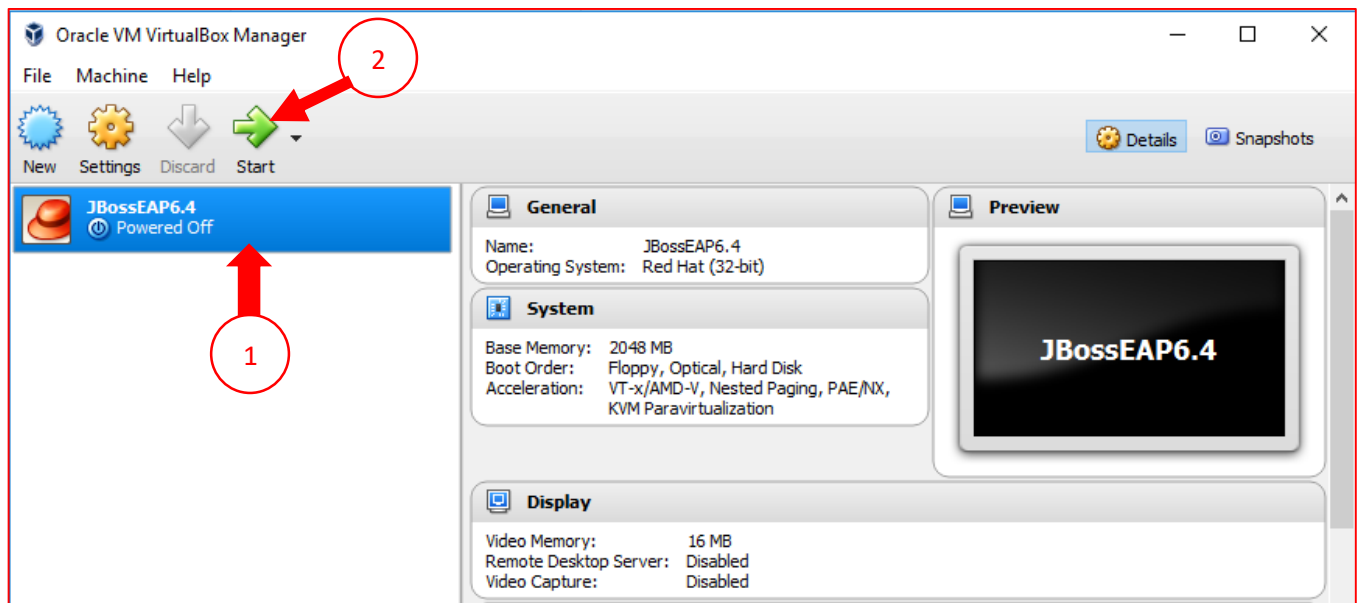
By Default the Network Adapter will be attached to NAT. We need to change it to **Host Only Adapter** for intercommunication between Guest VM and Host OS. Select From dropdown "Host Only Adapter" (1) and make the other changes as per the given screen below:



After making the changes, click on OK Button.

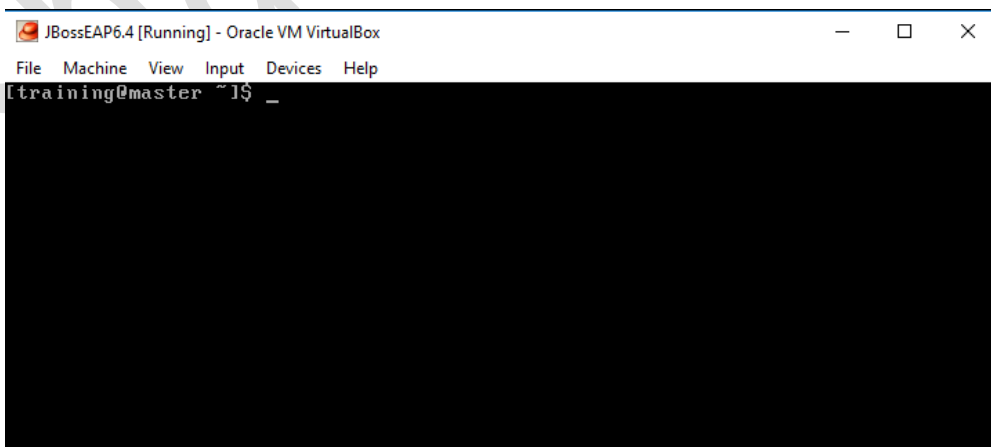
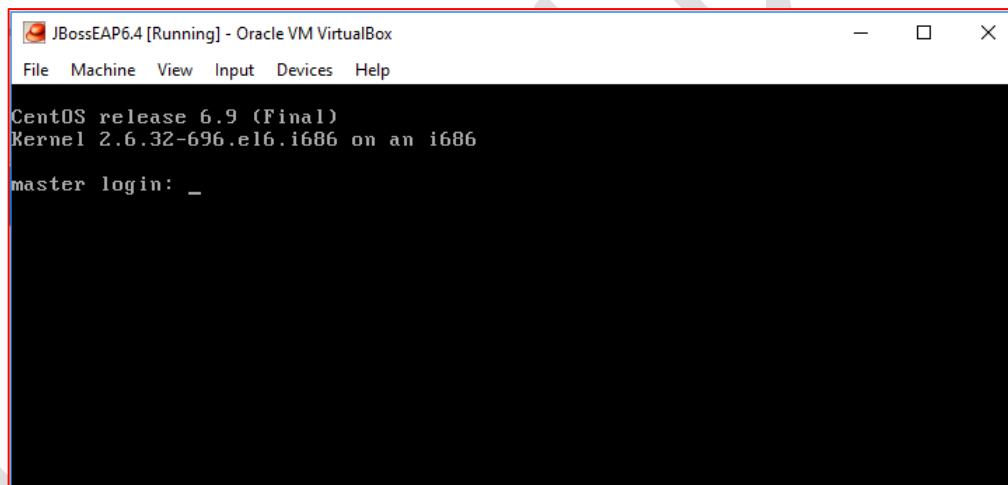
## LAB2: Start the Virtual Machine (Guest OS) and setup the network if required.

1. Select the VM (JBossEAP64) and click on Start button as shown:



The VM Starts and prompts for user name and password. Enter

**User Name: training || Password: training**



## Check the network adapter and reconfigure if required. OR Skipped to Lab 3

\$ifconfig

```
[training@master ~]$  
[training@master ~]$ ifconfig  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
          UP LOOPBACK RUNNING  MTU:65536  Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:0  
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

```
[training@master ~]$ _
```

We do not find any adapter containing the IP address for the VM. It's only the localhost with an adapter "lo". We need to configure the network adapter.

Use the command "**ip addr**" to list all installed Adapters. We see an adapter installed as "eth1" but not configured. Let's configure it.

```
[training@master ~]$  
[training@master ~]$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
    inet6 ::1/128 scope host  
    valid_lft forever preferred_lft forever  
2: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000  
    link/ether 08:00:27:4a:77:41 brd ff:ff:ff:ff:ff:ff  
[training@master ~]$ _
```

Use the following commands to configure and start the adapter named "**eth1**"

1. Note down the MAC address of eth1 (08:00:27:4a:77:41) [your value will be different]
2. Execute the following commands

```
$cd /etc/sysconfig/network-scripts
```

```
$sudo cp ifcfg-eth0 ifcfg-eth1
```

Edit the file **ifcfg-eth1** using vi and update the **DEVICE=eth1** and **HWADDR =08:00:27:4a:77:41**  
(please note your adapter hardware address may be different)

```
$sudo vi ifcfg-eth1
```

```
DEVICE=eth1
HWADDR=08:00:27:4a:77:41_
TYPE=Ethernet
UUID=74df55c6-4302-4259-8554-59ae9a91bbd3
ONBOOT=yes
NM_CONTROLLED=yes
BOOTPROTO=dhcp
```

Save the file and restart the network service and verify the ip address of the eth1 adapter

```
$sudo service network restart
```

```
[training@master network-scripts]$ sudo service network restart
[sudo] password for training:
Shutting down loopback interface: [ OK ]
Bringing up loopback interface: [ OK ]
Bringing up interface eth0: Device eth0 does not seem to be present, delaying i
nitialization.
[FAILED]
Bringing up interface eth1:
Determining IP information for eth1... done.
[ OK ]
[training@master network-scripts]$ _
```

**\$ifconfig**

```
[training@master ~]$
[training@master ~]$ ifconfig
eth1      Link encap:Ethernet  HWaddr 08:00:27:4A:77:41
          inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe4a:7741/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3268 (3.1 KiB)  TX bytes:1272 (1.2 KiB)

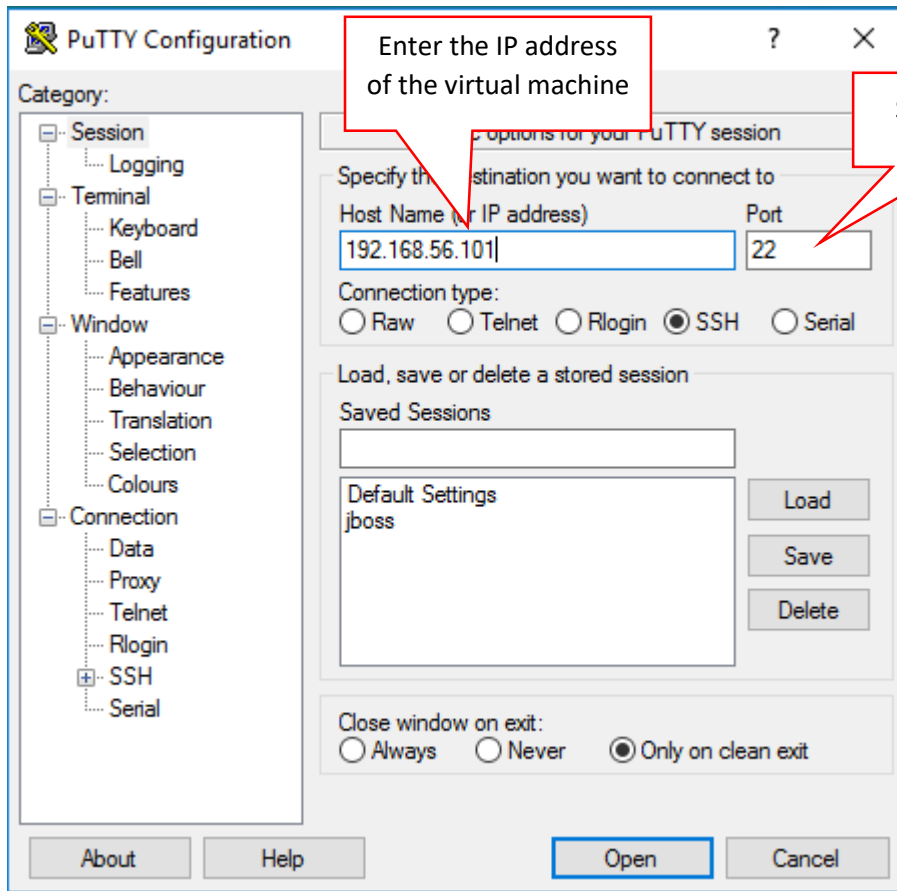
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

IP Address for the VM Now is 192.168.56.101

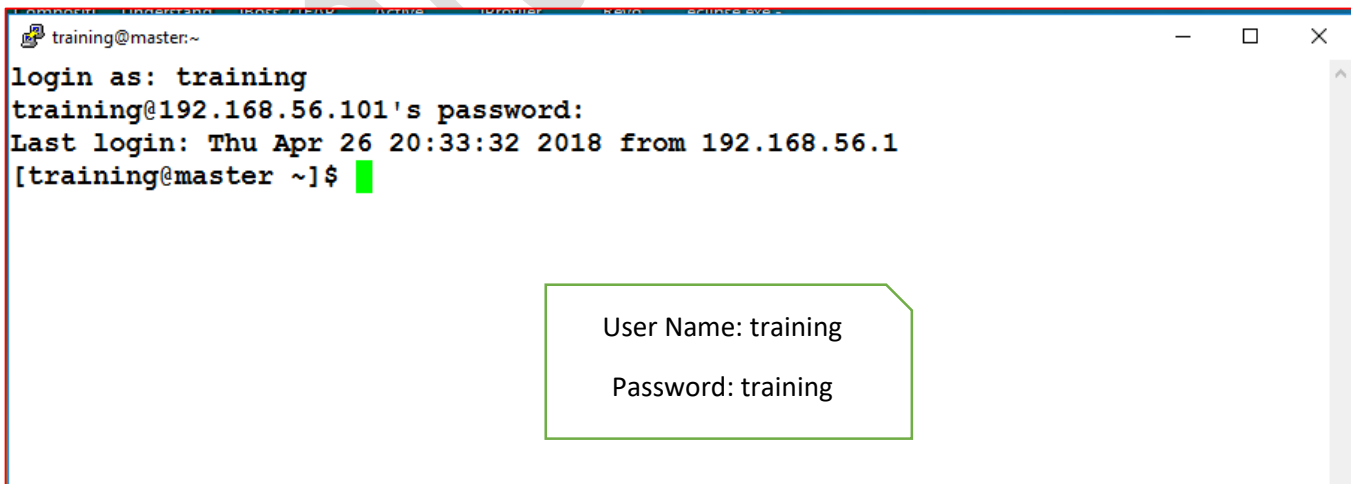
## LAB3: Connect the VM using Putty and WinSCP

### 1. Putty Connection:

Double click on putty.exe. Putty opens and the following screen appears:



Provide the IP address and SSH port and press “Open”

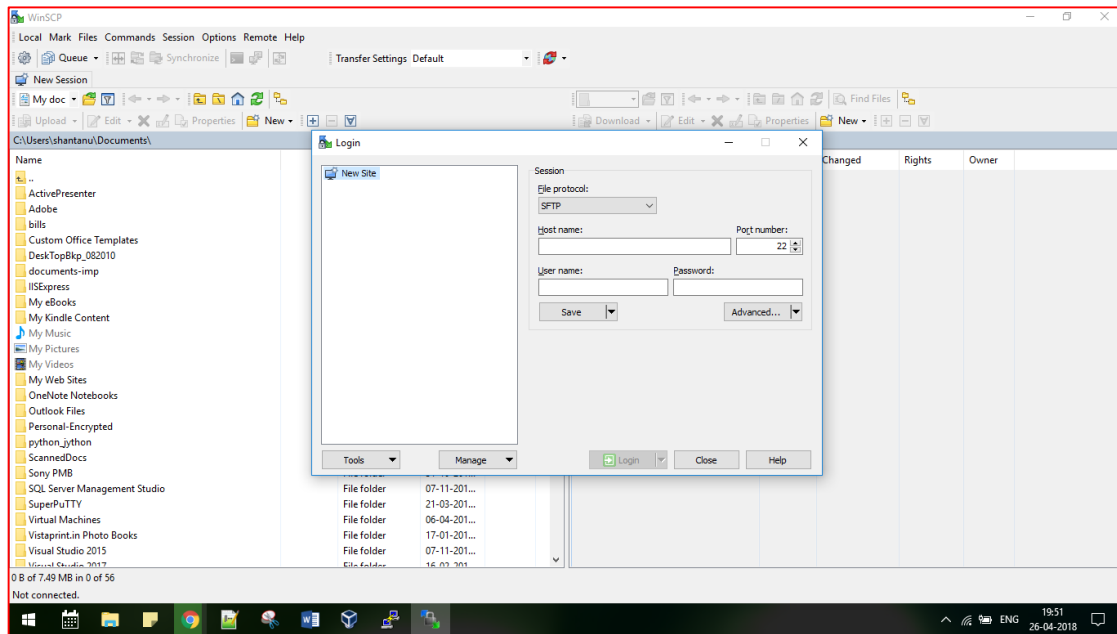


**YOU ARE NOW CONNECTED TO THE VM USING PUTTY**

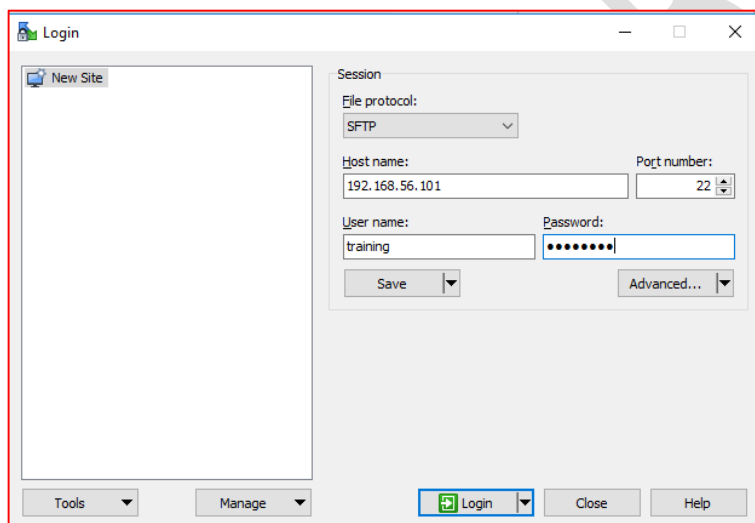


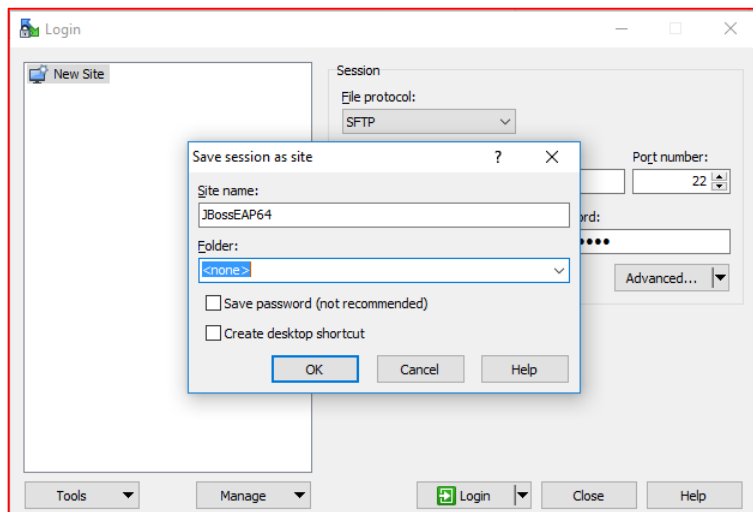
## 2. WinSCP Connection: (Install WinSCP, if required and then do the following)

Double Click on WinSCP and it opens as follows:

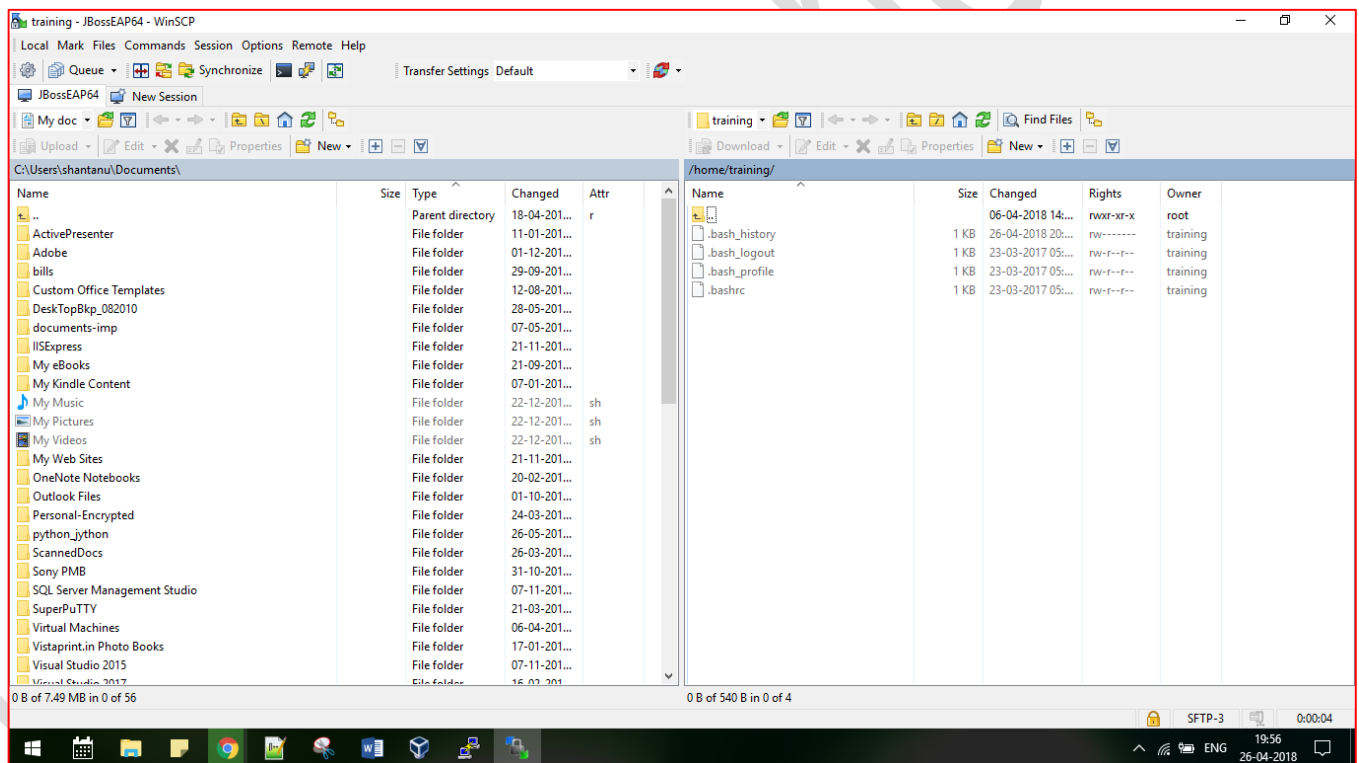


Provide IP Address, username and password and click on save. Save the configuration as JBossEAp64.





**After Successful Connection WinSCP opens as follows:**



**Now you are connected to the VM using WinSCP. You can transfer and edit files using WinSCP**

## LAB 4: Install Java

1. Use WinSCP to transfer “jdk-8u77-linux-i586.tar.gz” to /home/training/ directory of the Linux VM

Execute the following commands:

```
$cd  
$sudo mkdir /usr/java  
$sudo tar zxvf jdk-8u77-linux-i586.tar.gz -C /usr/java/  
$sudo ln -s /usr/java/jdk1.8.0_77 /usr/java/latest  
$ sudo vi /etc/profile.d/javaenv.sh
```

when the file opens, enter the following to set JAVA\_HOME and Java Path

```
export JAVA_HOME=/usr/java/latest  
export PATH=$JAVA_HOME/bin:$PATH
```

Save the file and check the correctness by using source command

```
$source /etc/profile.d/javaenv.sh
```

**Note: If you get some error after running this command, please correct them and re run the same command to recheck**

## LAB 5: Install, Start and Shutdown JBossEAP 6 (Standalone Mode)

1. Copy the file "jboss-eap-6.4.0.zip" to VM path "/home/training/" using WinSCP
2. Use the following commands to install JBoss-eap-6.4.0

```
$sudo mkdir /opt/production

$sudo chown -R training:training /opt/production

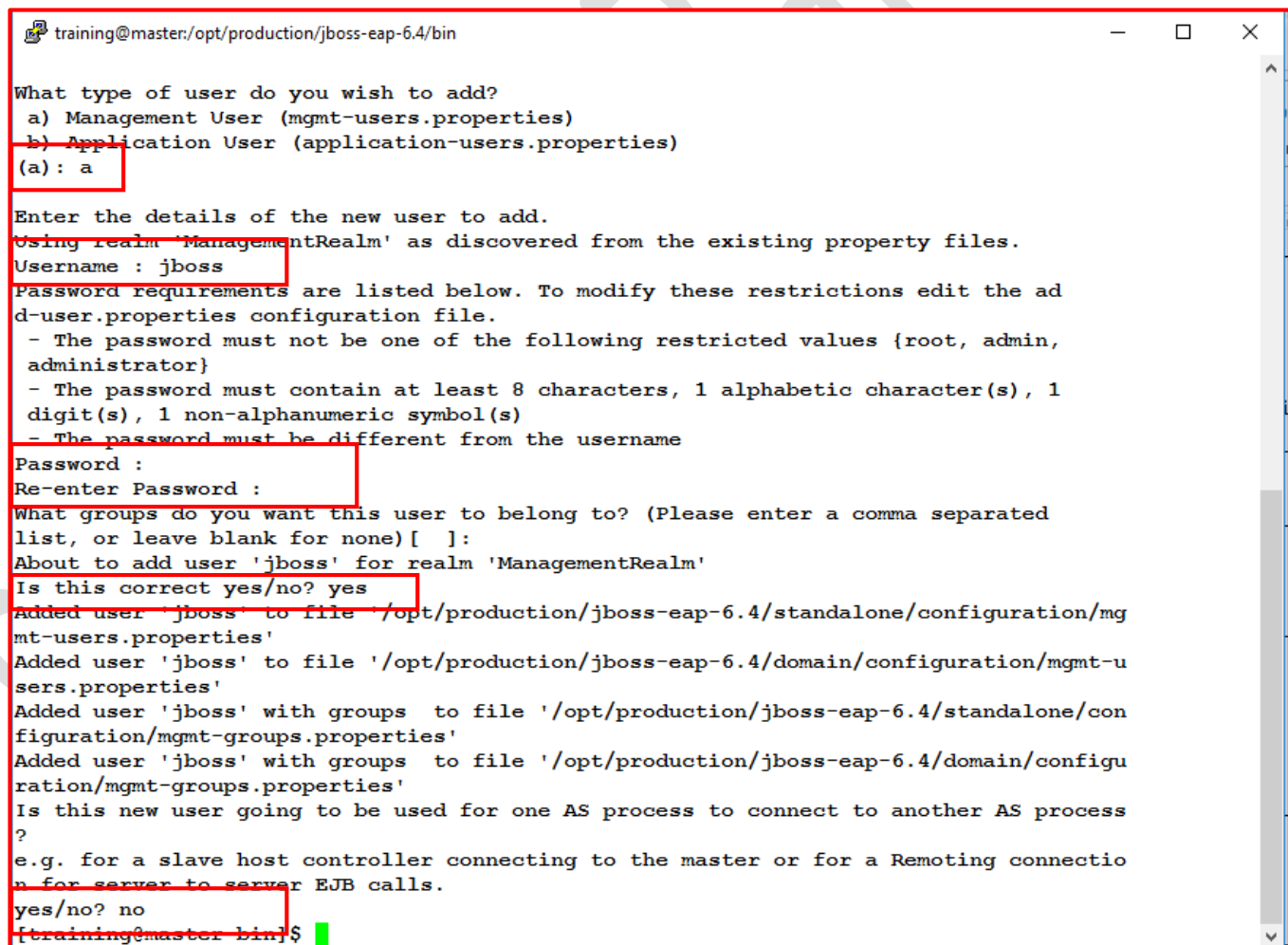
$cd

$unzip /home/training/jboss-eap-6.4.0.zip -d /opt/production/
```

3. Create a JBoss-EAP administrator (Management User):

```
$cd /opt/production/jboss-eap-6.4/bin

$ ./add-user.sh
```



```
training@master:/opt/production/jboss-eap-6.4/bin

What type of user do you wish to add?
a) Management User (mgmt-users.properties)
b) Application User (application-users.properties)
(a): a

Enter the details of the new user to add.
Using realm 'ManagementRealm' as discovered from the existing property files.
Username : jboss
Password requirements are listed below. To modify these restrictions edit the add-user.properties configuration file.
- The password must not be one of the following restricted values {root, admin, administrator}
- The password must contain at least 8 characters, 1 alphabetic character(s), 1 digit(s), 1 non-alphanumeric symbol(s)
- The password must be different from the username
Password :
Re-enter Password :
What groups do you want this user to belong to? (Please enter a comma separated list, or leave blank for none)[ ]:
About to add user 'jboss' for realm 'ManagementRealm'
Is this correct yes/no? yes
Added user 'jboss' to file '/opt/production/jboss-eap-6.4/standalone/configuration/mgmt-users.properties'
Added user 'jboss' to file '/opt/production/jboss-eap-6.4/domain/configuration/mgmt-users.properties'
Added user 'jboss' with groups to file '/opt/production/jboss-eap-6.4/standalone/configuration/mgmt-groups.properties'
Added user 'jboss' with groups to file '/opt/production/jboss-eap-6.4/domain/configuration/mgmt-groups.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.
yes/no? no
[training@master bin]$
```

Start JBoss Server in Standalone mode

```
$cd /opt/production/jboss-eap-6.4/bin  
$./standalone.sh
```

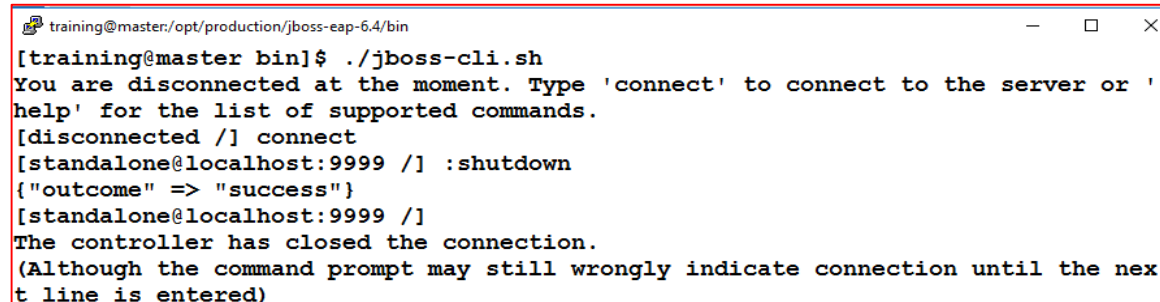
```
training@master:/opt/production/jboss-eap-6.4/bin  
1-2) JBWEB003001: Coyote HTTP/1.1 initializing on : http-/127.0.0.1:8080  
02:08:36,465 INFO [org.apache.coyote.http11.Http11Protocol] (MSC service thread  
1-2) JBWEB003000: Coyote HTTP/1.1 starting on: http-/127.0.0.1:8080  
02:08:37,297 WARN [com.arjuna.ats.arjuna] (MSC service thread 1-2) ARJUNA012210  
: Unable to use InetAddress.getLocalHost() to resolve address.  
02:08:37,903 INFO [org.jboss.ws.common.management] (MSC service thread 1-1) JBW  
S022052: Starting JBoss Web Services - Stack CXF Server 4.3.4.Final-redhat-1  
02:08:37,982 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread  
1-2) JBAS015012: Started FileSystemDeploymentService for directory /opt/producti  
on/jboss-eap-6.4/standalone/deployments  
02:08:38,252 INFO [org.jboss.as.remoting] (MSC service thread 1-1) JBAS017100:  
Listening on 127.0.0.1:9999  
02:08:38,268 INFO [org.jboss.as.remoting] (MSC service thread 1-1) JBAS017100:  
Listening on 127.0.0.1:4447  
02:08:38,738 INFO [org.jboss.as.connector.subsystems.datasources] (MSC service  
thread 1-1) JBAS010400: Bound data source [java:jboss/datasources/ExampleDS]  
02:08:39,061 INFO [org.jboss.as] (Controller Boot Thread) JBAS015961: Http mana  
gement interface listening on http://127.0.0.1:9990/management  
02:08:39,064 INFO [org.jboss.as] (Controller Boot Thread) JBAS015951: Admin con  
sole listening on http://127.0.0.1:9990  
02:08:39,069 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: JBoss EAP  
6.4.0.GA (AS 7.5.0.Final-redhat-21) started in 18062ms - Started 153 of 191 ser  
vices (57 services are lazy, passive or on-demand)
```

## Shutdown JBoss EAP:

There is no shutdown command as script. We need to use Jboss CLI as below:

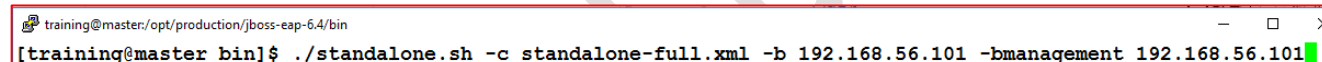
```
$cd /opt/production/jboss-eap-6.4/bin
```

```
$. /jboss-cli.sh
```




```
training@master:/opt/production/jboss-eap-6.4/bin
[training@master bin]$ ./jboss-cli.sh
You are disconnected at the moment. Type 'connect' to connect to the server or '
help' for the list of supported commands.
[disconnected /] connect
[standalone@localhost:9999 /] :shutdown
{"outcome" => "success"}
[standalone@localhost:9999 /]
The controller has closed the connection.
(Although the command prompt may still wrongly indicate connection until the nex
t line is entered)
```

## Run JBoss-EAP on particular IP address to be accessed from Host OS:



```
training@master:/opt/production/jboss-eap-6.4/bin
[training@master bin]$ ./standalone.sh -c standalone-full.xml -b 192.168.56.101 -bmanagement 192.168.56.101
```



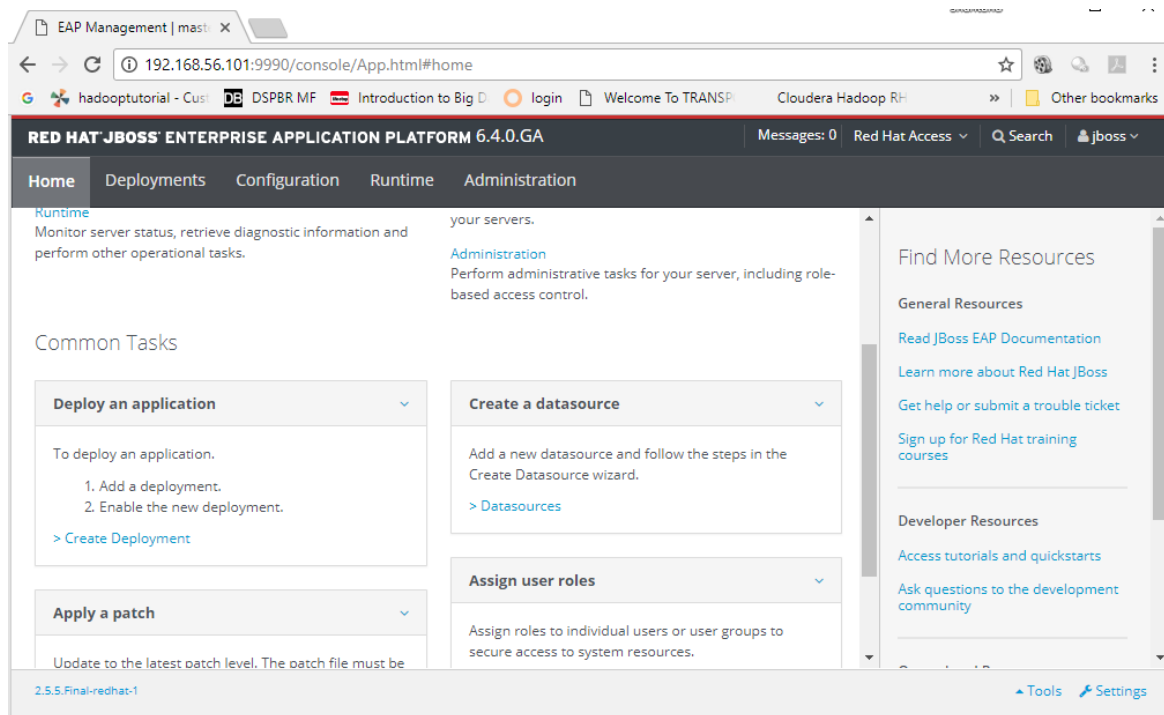
```
training@master:/opt/production/jboss-eap-6.4/bin
BWBE003001: Coyote HTTP/1.1 initializing on : http-/192.168.56.101:8080
03:00:51,005 INFO [org.apache.coyote.http11.Http11Protocol] (MSC service thread 1-1) JB
BWBE003000: Coyote HTTP/1.1 starting on: http-/192.168.56.101:8080
03:00:51,806 WARN [com.arjuna.ats.arjuna] (MSC service thread 1-1) ARJUNA012210: Unabl
e to use InetAddress.getLocalHost() to resolve address.
03:00:52,453 INFO [org.jboss.ws.common.management] (MSC service thread 1-2) JBWS022052
: Starting JBoss Web Services - Stack CXF Server 4.3.4.Final-redhat-1
03:00:52,477 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread 1-1) JB
AS015012: Started FileSystemDeploymentService for directory /opt/production/jboss-eap-6
.4/standalone/deployments
03:00:52,722 INFO [org.jboss.as.remoting] (MSC service thread 1-1) JBAS017100: Listeni
ng on 192.168.56.101:9999
03:00:52,734 INFO [org.jboss.as] (MSC service thread 1-1) JBAS017100: Listeni
ng on 192.168.56.101:4447
03:00:53,127 INFO [org.jboss.as.connector.subsystems.datasources] (MSC service thread
1-1) JBAS010400: Bound data source [java:jboss/datasources/ExampleDS]
03:00:53,486 INFO [org.jboss.as] (Controller Boot Thread) JBAS015961: Http management
interface listening on http://192.168.56.101:9990/management
03:00:53,487 INFO [org.jboss.as] (Controller Boot Thread) JBAS015951: Admin console li
stening on http://192.168.56.101:9990
03:00:53,497 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: JBoss EAP 6.4.0.
GA (AS 7.5.0.Final-redhat-21) started in 16206ms - Started 153 of 191 services (57 serv
ices are lazy, passive or on-demand)
```

### 1. Switch to Host OS (Windows in our case)

2. Open a Browser and type the JBossEAP management address <http://192.168.56.101:9990/>

When prompted for username and password enter:

Username: jboss Password: welcome@123 (your password may be different!!)



## LAB 6: Deploy Applications to JBoss EAP (Standalone Mode)

1. Copy **JBoss7Lab/JBossClass** folder to **VM's /home/training/** folder using WinSCP
2. Deployment of applications can be done in following ways
  - a. Drop-In deployment
  - b. Deployment of applications using admin console
  - c. Deployment Using JBoss CLI
  - d. Exploded Deployment

### 3. Drop-In Deployment

- a. The applications are deployed by copying the application archive to the server's deployment folder i.e. `$JBOSS_HOME/standalone/deployments` folder

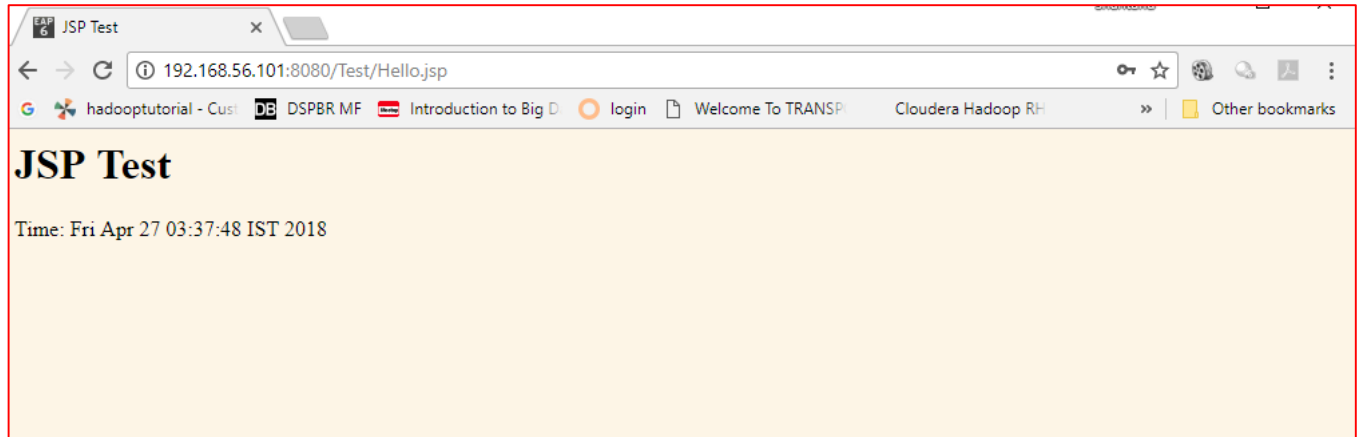
```
$scp /home/training/JBossClass/Test.war /opt/production/Jboss-eap-6.4/standalone/deployments/
```

If the deployment is successful, you will see the notification in the server's logging console.

```
03:34:38,900 INFO [org.jboss.as.server.deployment] (MSC service thread 1-1) JBAS015876 : Starting deployment of "Test.war" (runtime-name: "Test.war")
03:34:40,353 INFO [org.jboss.web] (ServerService Thread Pool -- 55) JBAS018210: Register web context: /Test
03:34:41,490 INFO [org.jboss.as.server] (DeploymentScanner-threads - 1) JBAS015859: Deployed "Test.war" (runtime-name : "Test.war")
```

Access the application from browser by entering the url as

<http://192.168.56.101:8080/Test/Hello.jsp>



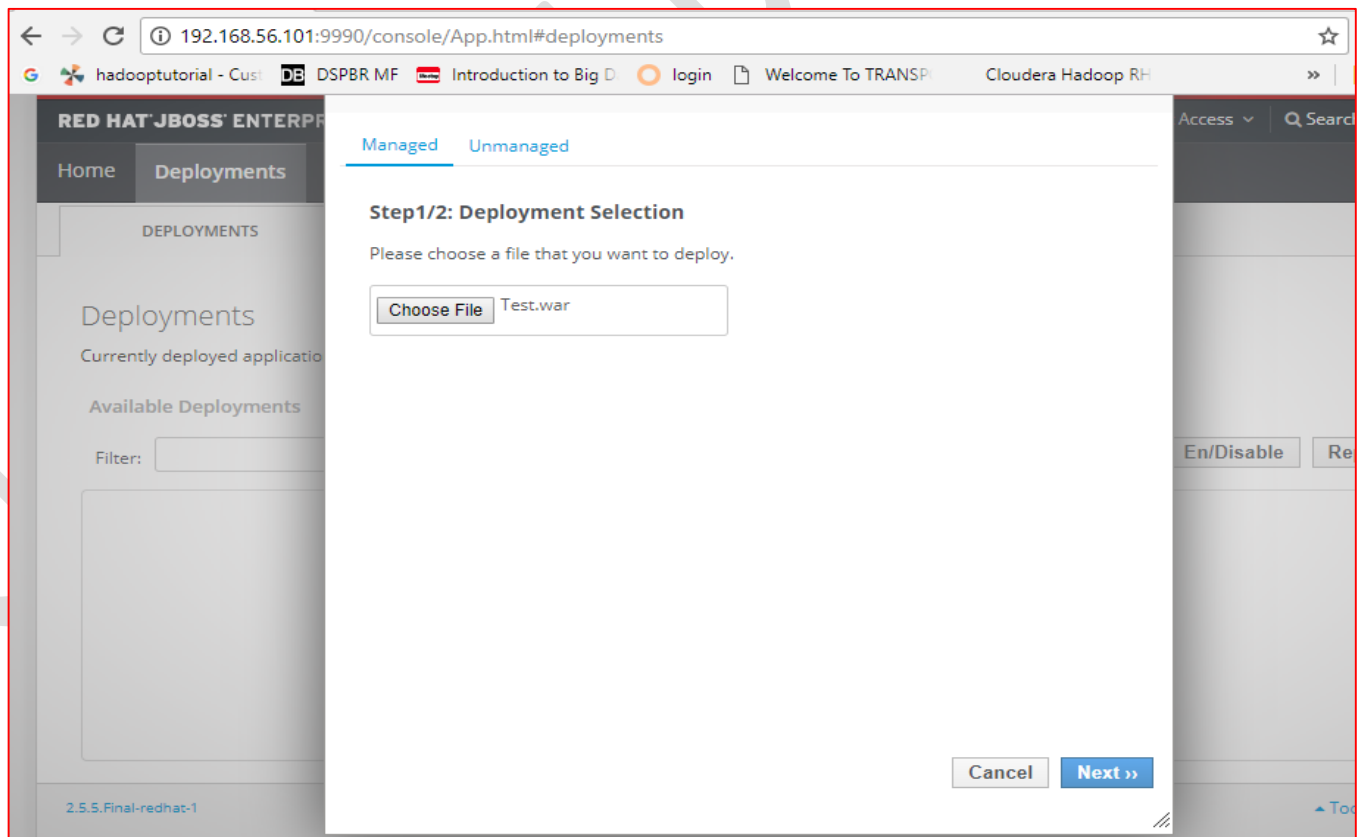
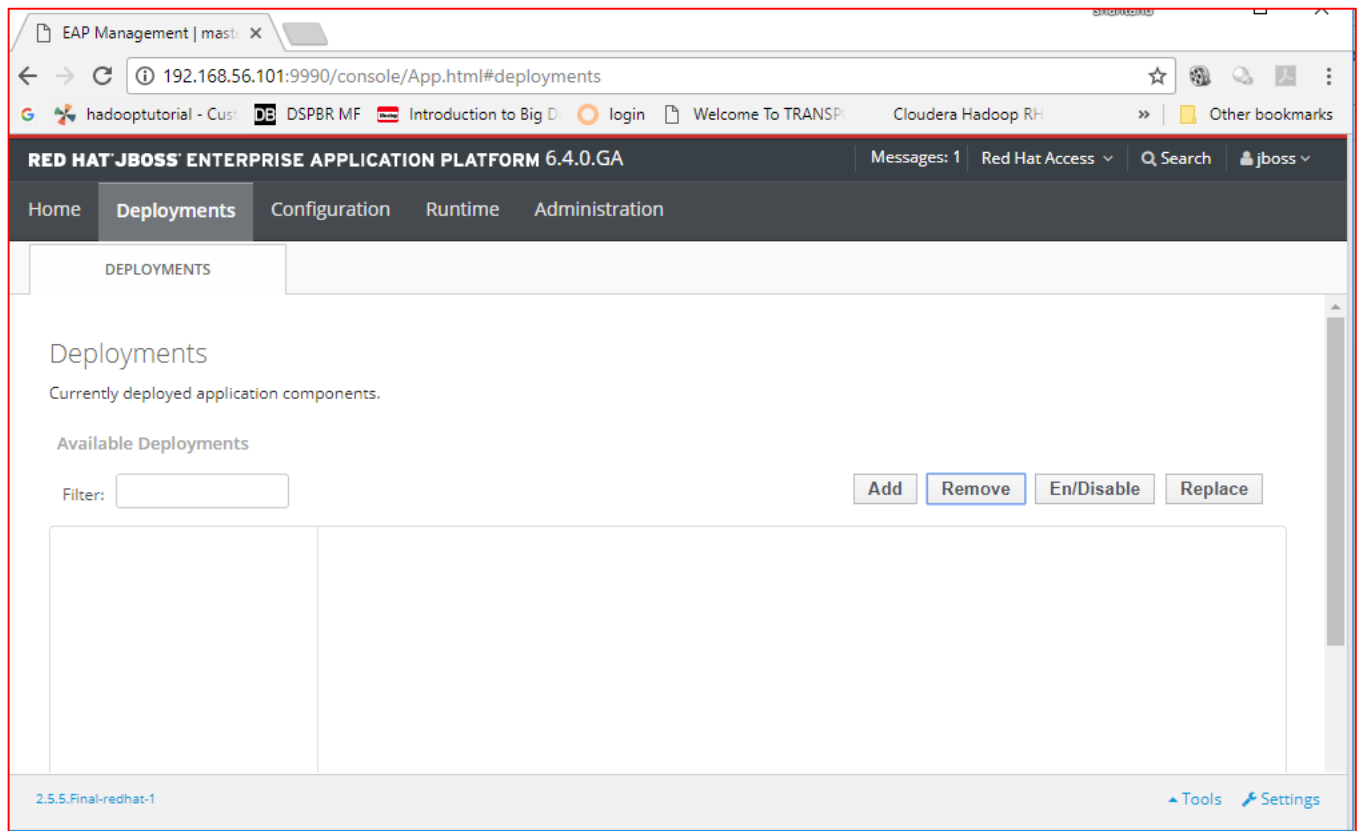
Using the same procedure you may deploy other type of archives e.g. .war, .ear and .jar etc.

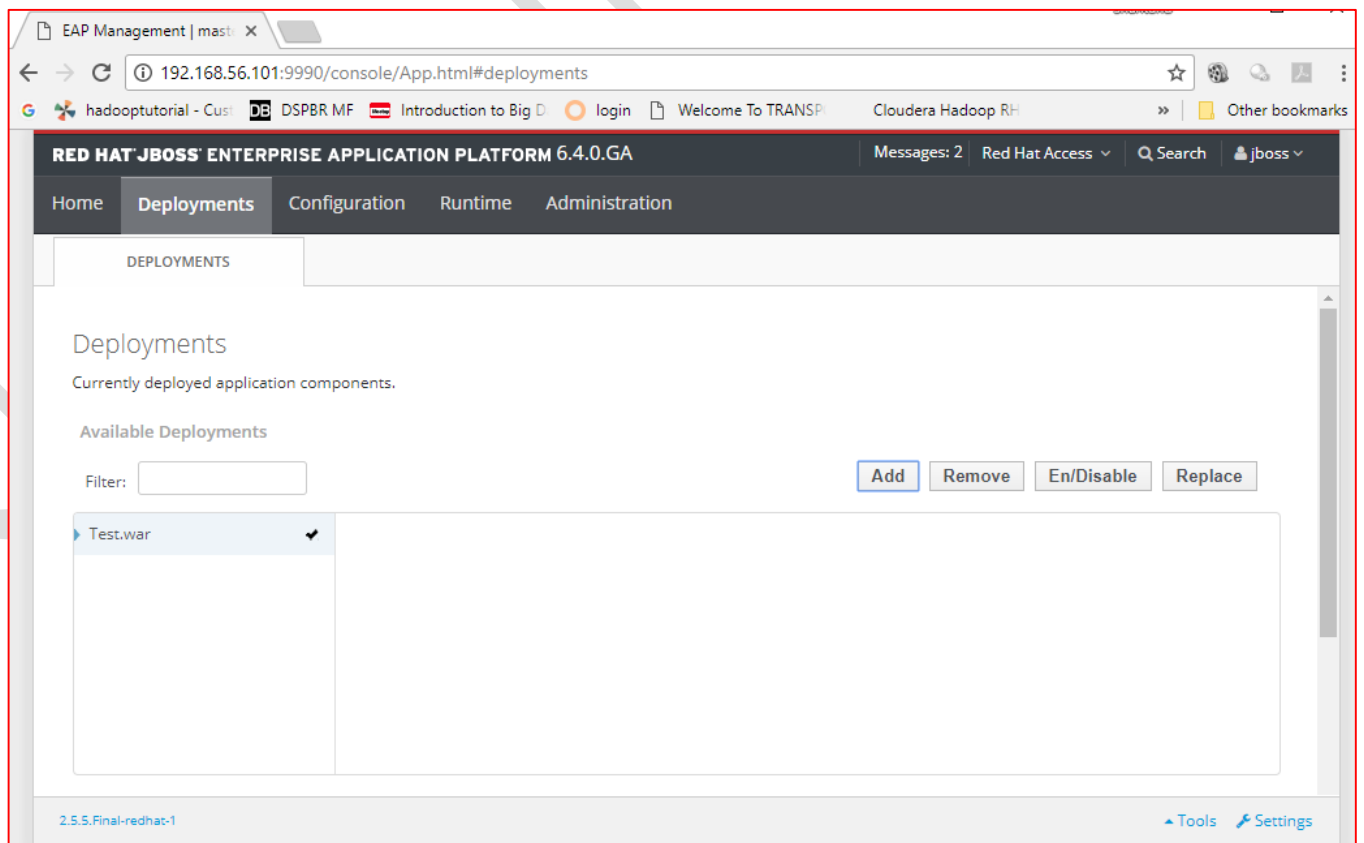
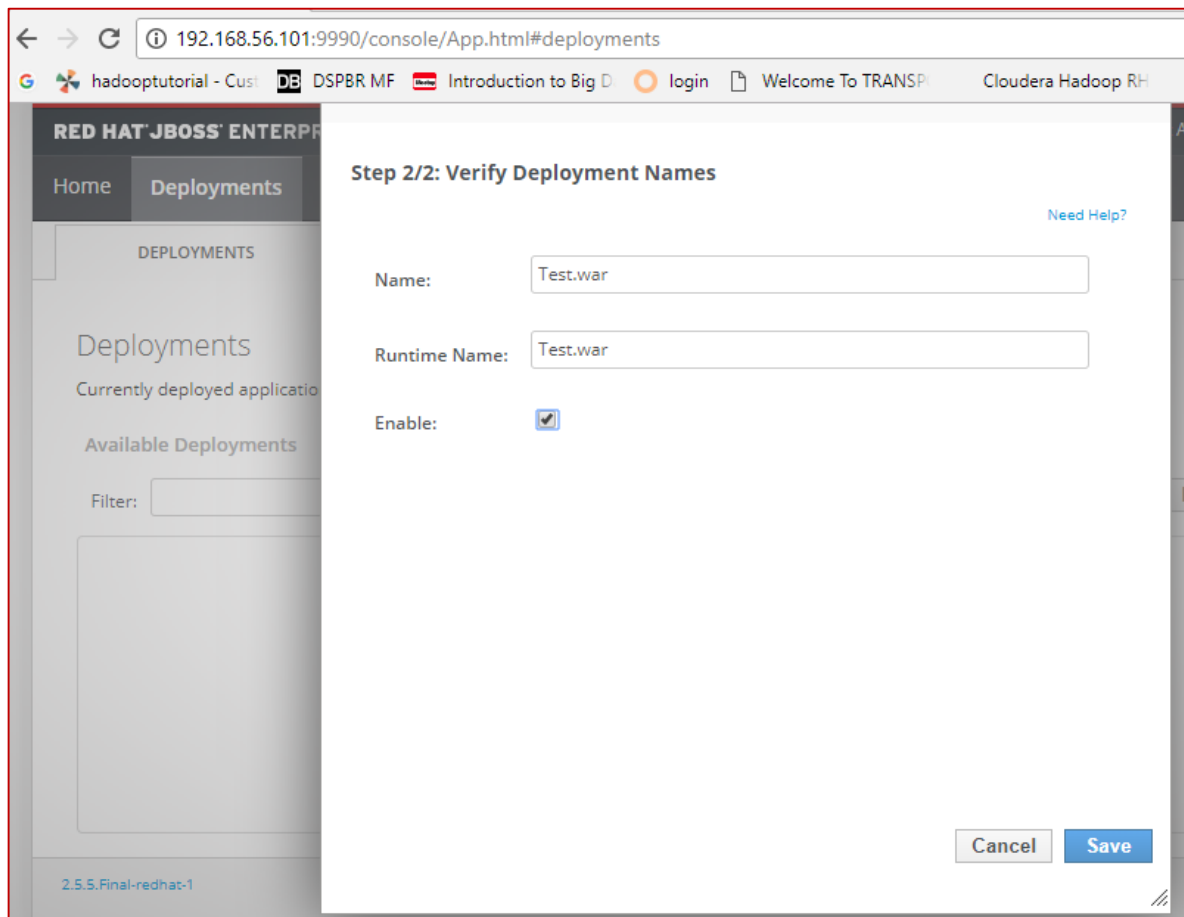
#### 4. Deployment using Admin Console (Web Console)

(We are going to reuse the same Test.war file for demonstration. Please remove the app from the deployment folder prior to redeployment)

1. Start JBossEAP Server (If already not started)
2. Open the AdminConsole using a browser and typing the url <http://192.168.56.101:9990> and login to the Admin Console.
3. Select Deployments Tab from the Admin Console and click on Add button in the right Pane
4. Browse to the folder and select Test.war in the next screen and press Next button.
5. Check enable option and press Save button. Your Application is now deployed.







## 5. Deployment using JBoss CLI

Use the following commands to start CLI and connect to the server and then deploy the application. List the deployments.

```
[training@master ~]$ cd /opt/production/jboss-eap-6.4/bin/
[training@master bin]$
[training@master bin]$ ./jboss-cli.sh
You are disconnected at the moment. Type 'connect' to connect to the server or 'help' for the list of supported commands.
[disconnected /]
[disconnected /] connect 192.168.56.101:9999
[standalone@192.168.56.101:9999 /]
[standalone@192.168.56.101:9999 /] deploy /home/training/JBossClass/Test.war
[standalone@192.168.56.101:9999 /]
[standalone@192.168.56.101:9999 /]
[standalone@192.168.56.101:9999 /] deployment-info
NAME          RUNTIME-NAME  PERSISTENT  ENABLED  STATUS
Test.war Test.war      true        true     OK
[standalone@192.168.56.101:9999 /] █
```

To Undeploy you may use:

```
[standalone@192.168.56.101:9999 /] deployment-info
NAME          RUNTIME-NAME  PERSISTENT  ENABLED  STATUS
Test.war Test.war      true        true     OK
[standalone@192.168.56.101:9999 /]
[standalone@192.168.56.101:9999 /] undeploy Test.war
[standalone@192.168.56.101:9999 /]
[standalone@192.168.56.101:9999 /] deployment-info
[standalone@192.168.56.101:9999 /] █
```

## 6. Exploded Deployment in JBoss

- Unzip the Test.war file as Test1.war folder

```
[training@master ~]$ unzip JBossClass/Test.war -d JBossClass/Test1.war/
Archive:  JBossClass/Test.war
  creating: JBossClass/Test1.war/META-INF/
 extracting: JBossClass/Test1.war/META-INF/MANIFEST.MF
 extracting: JBossClass/Test1.war/Hello.html
 extracting: JBossClass/Test1.war/Hello.jsp
  creating: JBossClass/Test1.war/WEB-INF/
  creating: JBossClass/Test1.war/WEB-INF/classes/
  creating: JBossClass/Test1.war/WEB-INF/lib/
 extracting: JBossClass/Test1.war/WEB-INF/web.xml
 inflating: JBossClass/Test1.war/WEB-INF/weblogic.xml
[training@master ~]$
```

- Copy the Folder “Test1.war” to \$JBASS\_HOME/standalone/deployments/ folder
- The Exploded folder will be detected by the JBoss Server but it will not be deployed automatically. We need to trigger the deployment with a file named as “Test1.war.dodeploy”.

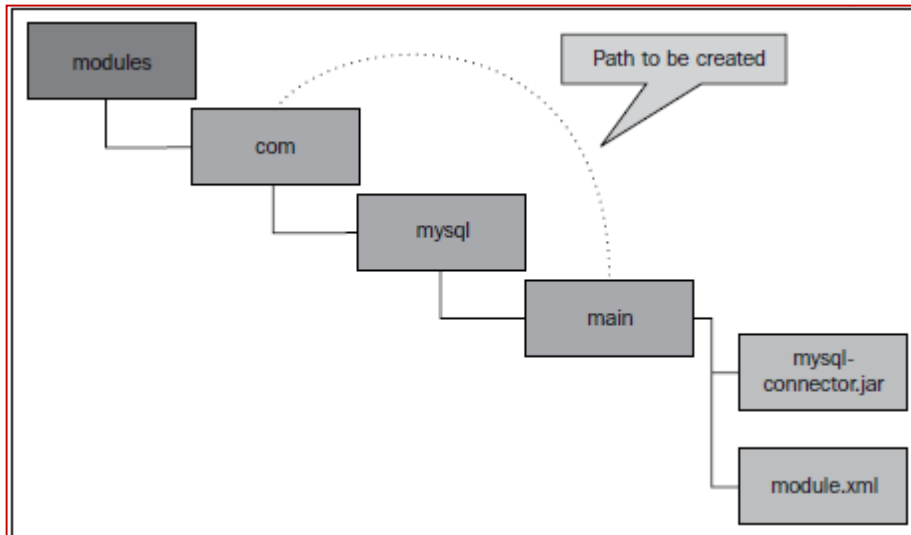
```
18:57:57,694 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: JBoss EAP 6.4.0.GA (AS 7.5.0.Final-redhat-21) started in 25985ms - Started 153 of 191 services (57 services are lazy, passive or on-demand)
18:58:56,448 INFO [org.jboss.as.server.deployment.scanner] (DeploymentScanner-threads - 1) JBAS015003: Found Test1.war in deployment directory. To trigger deployment create a file called Test1.war.dodeploy
```

```
[training@master ~]$ cp -r JBossClass/Test1.war /opt/production/jboss-eap-6.4/standalone/deployments/
[training@master ~]$ touch /opt/production/jboss-eap-6.4/standalone/deployments/Test1.war.dodeploy
[training@master ~]$
```

```
19:00:22,639 INFO [org.jboss.as.server.deployment] (MSC service thread 1-2) JBAS015876: Starting deployment of "Test1.war" (runtime-name: "Test1.war")
19:00:24,389 INFO [org.jboss.web] (ServerService Thread Pool -- 55) JBAS018210: Register web context: /Test1
19:00:25,747 INFO [org.jboss.as.server] (DeploymentScanner-threads - 1) JBAS015859: Deployed "Test1.war" (runtime-name : "Test1.war")
```

## LAB 7: Create a DataSource

- We will configure mysql datasource with JBoss 7(EAP6)
- We will add the driver as a module in JBoss Server
- Modules are added to jboss as  
`<JBOSS_HOME>/modules/system/layers/base/YourModule/main`
- Create a folder structure as `<JBOSS_HOME>/modules/com/mysql/main`



- Our module will be “**com.mysql**”
- Copy the driver file for mysql to  
`<JBOSS_HOME>/modules/system/layers/base/com/mysql/main`
- Create an xml file named “**module.xml**” in the same folder

```
[training@master ~]$ cd /opt/production/jboss-eap-6.4/
[training@master jboss-eap-6.4]$ pwd
/opt/production/jboss-eap-6.4
[training@master jboss-eap-6.4]$ mkdir -p modules/system/layers/base/com/mysql/main
[training@master jboss-eap-6.4]$ cp /home/training/mysql-connector-java-5.1.26.jar modules/system/layers/
base/com/mysql/main/
[training@master jboss-eap-6.4]$ ^C
[training@master jboss-eap-6.4]$ vi modules/system/layers/base/com/mysql/main/module.xml
[training@master jboss-eap-6.4]$
```

h. Add the following entries in the `module.xml` file and save the file

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.1" name="com.mysql">
  <properties>
    <property name="jboss.api" value="unsupported"/>
  </properties>
  <resources>
    <resource-root path="mysql-connector-java-5.1.26.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
    <module name="javax.servlet.api" optional="true"/>
  </dependencies>
</module>
```

- i. Open `$JBOSS_HOME/standalone/standalone.xml` and locate the **subsystem** for **datasources**
- j. Add the following xml snippet in the marked location as shown below, between `<drivers></drivers>` tag.

```
<driver name="mysql" module="com.mysql">
  <driver-class>com.mysql.jdbc.Driver</driver-class>
</driver>
```

```
<subsystem xmlns="urn:jboss:domain:datasources:1.2">
  <datasources>
    <datasource jndi-name="java:jboss/datasources/ExampleDS" pool-name="ExampleDS" enabled="true"
      use-java-context="true">
      <connection-url>jdbc:h2:mem:test;DB_CLOSE_DELAY=-1;DB_CLOSE_ON_EXIT=FALSE</connection-url>
      <driver>h2</driver>
      <security>
        <user-name>sa</user-name>
        <password>sa</password>
      </security>
    </datasource>
    <drivers>
      <driver name="h2" module="com.h2database.h2">
        <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>
      </driver>
    </drivers>
  </datasources>
</subsystem>
```

Insert the driver xml snippet here

Now your added driver and module are ready to be used.

## 7.1 As the driver is ready to be used. We have 3 different ways to create a Datasource.

- Using the Server admin Console
- Using the Server's Configuration file (standalone-\*.xml or domain.xml)
- Using JBoss CLI

## 7.2 Create a Datasource Using Server's Admin Console

1. Open the admin console using a browser at <http://192.168.56.101:9990> and login.
2. Select the **configuration** tab in the Admin Console
3. Select Connector/datasources from left pane and click on **add** button in the right pane to start the process
4. Follow the screens below:

The screenshot displays the Red Hat JBoss Enterprise Application Platform 6.4.0.GA Admin Console. The top navigation bar includes 'Home', 'Deployments', 'Configuration' (selected), 'Runtime', and 'Administration'. The left sidebar shows a tree view with 'Subsystems' expanded, containing 'Connector', 'JCA', 'Datasources' (highlighted with a red arrow), 'Resource Adapters', 'Mail', 'Container', 'Core', 'Infinispan', 'Security', and 'Web'. Below this is 'General Configuration' with 'Interfaces', 'Socket Binding', 'Paths', and 'System Properties'. The main content area is titled 'JDBC Datasources' and shows 'JDBC datasource configurations.' It includes a table of 'Available Datasources' with columns 'Name' and 'Enabled?'. The table lists 'ExampleDS' with JNDI 'java:jboss/datasources/ExampleDS' and is enabled. A red arrow points to the 'Add' button above the table. Below the table are tabs for 'Attributes', 'Connection', 'Pool', 'Security', 'Properties', 'Validation', and 'Timeouts'. The 'Attributes' tab is active, showing an 'Edit' button and fields for 'Name' (ExampleDS), 'JNDI' (java:jboss/datasources/ExampleDS), and 'Is enabled?' (true). A 'Need Help?' link is also present.

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	✓

Edit	
Name:	ExampleDS
JNDI:	java:jboss/datasources/ExampleDS
Is enabled?:	true

Create Datasource

Step 1/3: Datasource Attributes

Need Help?

Name:

mysqlDs

JNDI Name:

java:jboss/mysql/DS1

Cancel

Next »

Create Datasource

Step 2/3: JDBC Driver

Select one of the installed JDBC driver. Don't see your driver? Please make sure it's deployed as a module and properly registered.

Detected Driver

Specify Driver

Name

h2

mysql

<<

<

1-2 of 2

>

>>

Cancel

Next »



Create Datasource

Step 3/3: Connection Settings

Need Help?

Connection URL:

Username:

Password:

Security Domain:

Create Datasource

Step 3/3: Connection Settings

Need Help?

Test Connection

**i Successfully created JDBC connection.**

Successfully connected to database mysqlds.

Home

Deployments

Configuration

Runtime

Administration

Subsystems



DATASOURCES

XA DATASOURCES

- > Connector
- > Container
- > Core
- > Infinispan
- > Security
- > Web

General Configuration

- Interfaces
- Socket Binding
- Paths
- System Properties

## JDBC Datasources

JDBC datasource configurations.

Available Datasources

Add

Remove

Disable

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	✓
mysqlDS	java:jboss/mysql/DS1	✓

&lt;&lt; &lt; 1-2 of 2 &gt; &gt;&gt;

Attributes

Connection

Pool

Security

Properties

Validation

Timeouts

[Need Help?](#)

Edit

Name: mysqlDS

JNDI: java:jboss/mysql/DS1

## 7.3 Create a Datasource using the Server's Configuration file (standalone-\*.xml or domain.xml)

Modify your server's config file "standalone\*.xml" or "domain.xml"

Modify the section `<subsystem xmlns="urn:jboss:domain:datasources:1.0">`

Insert the following xml after `<datasources>` element

```
<datasource jta="false" jndi-name="java:jboss/mysql/DS1" pool-name="mysqlDs" enabled="true"
use-ccm="false">
  <connection-url>jdbc:mysql://localhost:3306/demodb</connection-url>
  <driver-class>com.mysql.jdbc.Driver</driver-class>
  <driver>mysql</driver>
  <security>
    <user-name>root</user-name>
    <password>root</password>
  </security>
  <validation>
    <validate-on-match>false</validate-on-match>
    <background-validation>false</background-validation>
  </validation>
  <statement>
    <share-prepared-statements>false</share-prepared-statements>
  </statement>
</datasource>
```

- Restart your Server.

## 8. JMS Configuration: (JMS is supported in **standalone-full\*.xml** configuration)

### 8.1 Configure HornetQ Server

1. By default Security is enabled in HornetQ and it raises an exception as Failed to create session : HornetQException[errorCode=105 message=Unable to validate user: null] if any JMS resource is accessed without proper security setup.
2. To correct this problem you need to add the following

```
<subsystem xmlns="urn:jboss:domain:messaging:1.1">
  <hornetq-server>
    <persistence-enabled>true</persistence-enabled>
    <journal-file-size>102400</journal-file-size>
    <journal-min-files>2</journal-min-files>
    <security-enabled>false</security-enabled>
  </hornetq-server>
</subsystem>
```

### 8.2 Create Connection Factory and JMS Destinations

1. Create Connection Factory:

The screenshot displays the Red Hat JBoss Enterprise Application Platform 6.4.0.GA Configuration console. The top navigation bar includes 'Home', 'Deployments', 'Configuration' (highlighted with a red circle and the number 1), 'Runtime', and 'Administration'. The left sidebar shows a tree view of subsystems, with 'Messaging' expanded and 'Destinations' (highlighted with a red circle and the number 2) selected. The main content area is titled 'MESSAGING DESTINATIONS' and 'JMS Messaging Provider'. It instructs the user to 'Please choose a provider from below for specific settings.' and lists 'Available Messaging Provider' as 'default'. A table with columns 'Name' and 'Option' (highlighted with a red circle and the number 3) shows the 'default' provider with a 'View >' link. Below the table, the 'Attributes' section is visible, showing settings like 'Allow fallback: true', 'Async connection execution enabled: true', and 'Backup: false'. The top right of the console shows 'Messages: 0', 'Red Hat Access', a search bar, and a user profile 'jboss'.

RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA

Messages: 0Red Hat Access 🔍 Search👤 jboss

HomeDeploymentsConfigurationRuntimeAdministration

Subsystems<<

MESSAGING DESTINATIONS

< BackQueues/Topics**Connection Factories**Security SettingsAddress SettingsDiverts

Connection Factories: Provider default

Connection factories for applications. Used to connect to the server using the JMS API.

Connection Factories

AddRemove

Name	JNDI
InVmConnectionFactory	java:/ConnectionFactory
RemoteConnectionFactory	java:jboss/exported/jms/RemoteConnectionFactory

<<<1-2 of 2>>>

CommonConnection Management

Need Help?

Edit

Name:RemoteConnectionFactory

Create Connection Factory

Need Help?

Name:

CustomConnectionFactory

JNDI Name:

java:jboss/exported/jms/ConnectionFactory

Connector:

netty

Cancel

Save

RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA Messages: 1 Red Hat Access Search jboss

Home Deployments Configuration Runtime Administration

Subsystems

- Connector
- Container
- Core
- Infinispan
- Messaging
  - Clustering
  - Connections
  - Destinations**
- Security
- Web

General Configuration

- Interfaces
- Socket Binding
- Paths
- System Properties

MESSAGING DESTINATIONS

Back Queues/Topics **Connection Factories** Security Settings Address Settings Diverts

### Connection Factories: Provider default

Connection factories for applications. Used to connect to the server using the JMS API.

Connection Factories

Add Remove

Name	JNDI
InVmConnectionFactory	java:/ConnectionFactory
RemoteConnectionFactory	java:jboss/exported/jms/RemoteConnectionFactory
CustomConnectionFactory	java:jboss/exported/jms/ConnectionFactory

1-3 of 3

Common **Connection Management**

Need Help?

Edit

### 8.3. Create a Queue

RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA Messages: 0 Red Hat Access Search jboss

Home Deployments Configuration Runtime Administration

Subsystems

- Connector
- Container
- Core
- Infinispan
- Messaging
  - Clustering
  - Connections
  - Destinations**
- Security
- Web

General Configuration

- Interfaces
- Socket Binding
- Paths
- System Properties

MESSAGING DESTINATIONS

Back **Queues/Topics** Connection Factories Security Settings Address Settings Diverts

### JMS Endpoints: Provider default

Queue and Topic destinations.

**Queues** Topics

Add Remove

Name	JNDI
ExpiryQueue	[java:/jms/queue/ExpiryQueue]
DLQ	[java:/jms/queue/DLQ]

1-2 of 2

Queues are read-only after creation

Need Help?

Name: ExpiryQueue

JNDI Names: java:/jms/queue/ExpiryQueue

Annotations:

### Create JMS Queue

Need Help?

Name:

DemoQueue

JNDI Names:

java:jboss/exported/jms/DemoQueue

One item per line

Durable?:

☐

Selector:

Cancel

Save

RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA

Messages: 2 Red Hat Access Search jboss

Home Deployments Configuration Runtime Administration

Subsystems

Connector

Container

Core

Infinispan

▼ Messaging

Clustering

Connections

Destinations

Security

Web

General Configuration

Interfaces

Socket Binding

Paths

System Properties

MESSAGING DESTINATIONS

Back Queues/Topics Connection Factories Security Settings Address Settings Diverts

Queue and Topic Destinations

Queues Topics

Add Remove

Name	JNDI
ExpiryQueue	[java:/jms/queue/ExpiryQueue]
DLQ	[java:/jms/queue/DLQ]
DemoQueue	[java:jboss/exported/jms/DemoQueue]

1-3 of 3

Queues are read-only after creation

Need Help?

Name:	ExpiryQueue
JNDI Names:	java:/jms/queue/ExpiryQueue
Durable?:	true

## 8.4. Create a Topic

RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA Messages: 2 Red Hat Access Search jboss

Home Deployments **Configuration** Runtime Administration

Subsystems <<

MESSAGING DESTINATIONS

< Back Queues/Topics Connection Factories Security Settings Address Settings Diverts

JMS Endpoints: Provider default

Queue and Topic destinations.

Queues Topics

Add Remove

Name	JNDI
No Items!	

General Configuration << < > >>

Create JMS Topic

Need Help?

Name: DemoTopic

JNDI Names: java:boss/exported/jms/DemoTopic

One item per line

Cancel Save



RED HAT JBOSS® ENTERPRISE APPLICATION PLATFORM 6.4.0.GA Messages: 3 Red Hat Access Search jboss

Home Deployments **Configuration** Runtime Administration

Subsystems << MESSAGING DESTINATIONS

< Back Queues/Topics Connection Factories Security Settings Address Settings Diverts

JMS Endpoints: Provider default

Queue and Topic destinations.

Queues Topics

Add Remove

Name	JNDI
DemoTopic	[java:jboss/exported/jms/DemoTopic]

<< < 1-1 of 1 > >>

Edit

Name: DemoTopic

JNDI Names: java:jboss/exported/jms/DemoTopic

General Configuration

Interfaces

Socket Binding

Paths

## 8.5 Alternative JMS Configuration

Note: Alternatively you may create a JMS Destination with an independent file as follows:

Just add this file to deployment folder

File name: **test-jms.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<messaging-deployment xmlns="urn:jboss:messaging-deployment:1.0">
  <hornetq-server>
    <jms-destinations>
      <jms-queue name="QueueOne">
        <entry name="java:jboss/exported/TestQueueOne"/>
        <durable>true</durable>
      </jms-queue>
      <jms-topic name="TopicOne">
        <entry name="java:jboss/exported/TestTopicOne"/>
      </jms-topic>
    </jms-destinations>
  </hornetq-server>
</messaging-deployment>
```

## 9. LAB 9: Security Configuration

### 9.1 Configure SSL on JBoss 7:

1. Follow the Instructors guidance to create the Self Signed Certificate and the keystore.
2. Open <JBOSSE\_HOME><your config><config-file>
3. Navigate to the web subsystem configuration
4. In the connector section add the following xml snippet (Update the key store file location)

```
<connector name="https" protocol="HTTP/1.1" socket-binding="https" scheme="https"
secure="true">
```

```
  <ssl name="https" key-alias ="mykey" password="welcome1" certificate-key-file="C:\jboss-
as-7.1.1.Final\ssl\mykeys.jks"/>
```

```
</connector>
```

**Note:** Follow the Web Subsystem xsd for further configuration.

### 9.2 Security Domain Configuration for Applications and resources

Every security domain configuration is available in JBoss Server with the following JNDI name pattern:

`java:/jaas/<name of the domain in config file>`

e.g.

`java:/jaas/mydomain`

#### 9.2.1 Security domain based on .properties files

UserRoles Login Module (please configure properties files required)

**myusers.properties** file:

```
#username=password mapping
scott=scott123
arun=arun123
shantanu=shan123
```

**myroles.properties** file:

```
#username=role mapping
scott=manager
arun=user
shantanu=manager
```

Add the following entry in security-domains subsystem:

```
<security-domain name="mydomain" cache-type="default">
  <authentication>
    <login-module code="UsersRoles" flag="required">
      <module-option name="usersProperties"
value="\${jboss.server.config.dir}/myusers.properties"/>
      <module-option name="rolesProperties"
value="\${jboss.server.config.dir}/myroles.properties"/>
    </login-module>
  </authentication>
</security-domain>
```

## 9.2.2 Security domain based on Database

**Note** In order to get working with this configuration, you first have to create the required tables and insert some sample data in it:

```
create table users(username varchar(32) primary key,password varchar(32));
create table roles(username varchar(32),role varchar(20));
insert into users(username,password) values('scott','scott123');
insert into users(username,password) values('pavan','pavan123');
insert into users(username,password) values('shantanu','shan123');
insert into roles(username,role) values('scott','manager');
insert into roles(username,role) values('pavan','user');
insert into roles(username,role) values('shantanu','manager');
```

Database Server Login Module:

```
<security-domain name="mydbdomain" cache-type="default">
  <authentication>
    <login-module code="Database" flag="required">
      <module-option name="dsJndiName" value="<DataSource JNDI Name>"/>
      <module-option name="principalsQuery"
value="select password from users where username=?"/>
      <module-option name="rolesQuery"
value="select role, 'Roles' from roles where username=?"/>
    </login-module>
  </authentication>
</security-domain>
```

### 9.2.3 Security domain based on LDAP Server (refer to the associated document)

```
<security-domain name="ldapdomain" cache-type="default">
  <authentication>
    <login-module code="LdapExtended" flag="required">
      <module-option name="throwValidateError" value="true"/>
      <module-option name="java.naming.factory.initial"
value="com.sun.jndi.ldap.LdapCtxFactory"/>
      <module-option name="java.naming.provider.url"
value="ldap://192.168.40.1:10389"/>
      <module-option name="java.naming.security.authentication"
value="simple"/>
      <module-option name="bindDN" value="uid=admin,ou=system"/>
      <module-option name="bindCredential" value="secret"/>
      <module-option name="baseCtxDN" value="ou=users,dc=jboss,dc=com"/>
      <module-option name="baseFilter" value="(uid={0})"/>
      <module-option name="rolesCtxDN" value="ou=roles,dc=jboss,dc=com"/>
      <module-option name="roleFilter" value="(uniqueMember={1})"/>
      <module-option name="roleAttributeID" value="cn"/>
      <module-option name="searchScope" value="ONELEVEL_SCOPE"/>
      <module-option name="allowEmptyPasswords" value="true"/>
    </login-module>
  </authentication>
</security-domain>
```

### LDIF File Data: (also available in Jboss7Lab/JBossClass/security folder)

version: 1

dn: dc=jboss,dc=com  
objectclass: top  
objectclass: domain  
dc: jboss

dn: ou=users,dc=jboss,dc=com  
objectClass: organizationalUnit  
objectClass: top  
ou: users

dn: ou=roles,dc=jboss,dc=com  
objectClass: organizationalUnit  
objectClass: top  
ou: roles

dn: uid=shantanu,ou=users,dc=jboss,dc=com  
objectClass: organizationalPerson  
objectClass: person  
objectClass: inetOrgPerson

objectClass: top  
cn: shantanu  
sn: banerjee  
uid: shantanu  
userPassword:: e1NTSEF9dkdHaHVjQVNPATeXWGZBQ3RGZzU5cDZ4dEw5MTdRRIR4NGtSU0E9P  
Q==

dn: uid=pavan,ou=users,dc=jboss,dc=com  
objectClass: organizationalPerson  
objectClass: person  
objectClass: inetOrgPerson  
objectClass: top  
cn: pavan  
sn: kumar  
uid: pavan  
userPassword:: e1NTSEF9dkdHaHVjQVNPATeXWGZBQ3RGZzU5cDZ4dEw5MTdRRIR4NGtSU0E9P  
Q==

dn: cn=user,ou=roles,dc=jboss,dc=com  
objectClass: groupOfUniqueNames  
objectClass: top  
cn: user  
uniqueMember: uid=pavan,ou=users,dc=jboss,dc=com

dn: cn=admin,ou=roles,dc=jboss,dc=com  
objectClass: groupOfUniqueNames  
objectClass: top  
cn: admin  
uniqueMember: uid=pavan,ou=users,dc=jboss,dc=com  
uniqueMember: uid=shantanu,ou=users,dc=jboss,dc=com

dn: cn=manager,ou=roles,dc=jboss,dc=com  
objectClass: groupOfUniqueNames  
objectClass: top  
cn: manager  
uniqueMember: uid=pavan,ou=users,dc=jboss,dc=com

dn: uid=arun,ou=users,dc=jboss,dc=com  
objectClass: organizationalPerson  
objectClass: person  
objectClass: inetOrgPerson  
objectClass: top  
cn: arun  
sn: kumar  
uid: arun  
userPassword:: e1NTSEF9dkdHaHVjQVNPATeXWGZBQ3RGZzU5cDZ4dEw5MTdRRIR4NGtSU0E9P  
Q==

## 9.2.4 Using Encrypted Password for Datasource

1. Generate the encrypted password for the database using picketbox encryption utility provided with JBoss as follows:

```
$export JBOSS_HOME=/opt/production/jboss-eap-6.4

$export
CLASSPATH=$JBOSS_HOME/modules/system/layers/base/org/picketbox/main/picketbox-4.1.1.Final-redhat-1.jar

$export CLASSPATH=$CLASSPATH:$JBOSS_HOME
/modules/system/layers/base/org/jboss/logging/main/jboss-logging-3.1.4.GA-redhat-2.jar

$java org.picketbox.datasource.security.SecureIdentityLoginModule <your
clear text password here>
```

2. Copy the generated encrypted password and keep it safe.
3. Go to the security subsystem and create a security domain as given below:

```
<security-domain name="encrypted-ds-pwd" cache-type="default">
  <authentication>
    <login-module code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
      <module-option name="username" value="<DB User Name>"/>
      <module-option name="password" value="<Encrypted Password>"/>
    <module-option name="managedConnectionFactoryName"
value="jboss.jca:service=LocalTxCM,name=<Your Connection Pool Name>"/>
    </login-module>
  </authentication>
</security-domain>
```

4. Make the necessary changes in the datasource configuration (replace username and password with the security domain as shown below . Refer to Lab part No. 7.3). Your Updated Security configuration should look like the following:

```
<security>
<!--
      <user-name>root</user-name>
      <password>root</password>
-->
  <security-domain>encrypted-ds</security-domain>
</security>
```

5. Restart your server and verify using the application which uses the datasource (e.g. TestDataSource.war)

## 10. LAB 10: Creating a cluster in Standalone Mode

### 10.1 Cluster in the same Box

#### 1. Create 2 copies of standalone folder as

- i. Standalone-node1
- ii. Standalone-node2

#### 2. Start JBoss node1 as

```
./standalone.sh -c standalone-ha.xml -b <IP> -bmanagement <IP> -u 230.0.0.4 -  
Djboss.server.base.dir=../standalone-node1 -Djboss.node.name=node1 -  
Djboss.socket.binding.port-offset=100
```

#### 3. Start JBoss node2 as :

```
./standalone.sh -c standalone-ha.xml -b <IP> -bmanagement <IP> -u 230.0.0.4 -  
Djboss.server.base.dir=../standalone-node2 -Djboss.node.name=node2 -  
Djboss.socket.binding.port-offset=200
```

4. As JBoss7 cluster is on-demand, you will not see any indication that a cluster is formed. You will be able to get information from the cluster when you deploy a clustered application.

## 11. LAB 11: Configure Apache Web Server as Load Balancer

#### 1. Assumptions

2. We have already started 2 JBoss nodes with ha /full-ha profiles having port offsets 100 and 200 respectively
3. The “**shoppingcart.war**” application is deployed in both the JBoss Nodes
4. JBoss Node1: name=node1, ajp port=8109
5. JBoss Node2: name=node2, ajp port=8209
6. IP address of the system: 192.168.56.101
7. Apache Web Server is already installed
8. We are configuring mod\_jk as load balancer

Copy the necessary files:

```
$ cd /home/training/JBossClass/cluster/  
$ pwd  
/home/training/JBossClass/cluster  
$ sudo cp mod-jk-config/mod-jk.conf /etc/httpd/conf/  
$ sudo cp mod-jk-config/workers.properties /etc/httpd/conf/  
$ sudo cp mod-jk-32bit_apache2.2/mod_jk.so /etc/httpd/modules/  
$ sudo vi /etc/httpd/conf/workers.properties
```

The edited “workers.properties” file should look like the following:

```
worker.list=loadbalancer,status

worker.node1.port=8109
worker.node1.host=192.168.56.101
worker.node1.type=ajp13
worker.node1.lbfactor=1

worker.node2.port=8209
worker.node2.host=192.168.56.101
worker.node2.type=ajp13
worker.node2.lbfactor=1

worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=node1,node2

# Status worker for managing load balancer
worker.status.type=status
```

Edit mod-jk.conf file using vi

```
$sudo vi /etc/httpd/conf/mod-jk.conf
```

The edited file should look like the following:

```
# Load mod_jk module
# Specify the filename of the mod_jk lib
LoadModule jk_module modules/mod_jk.so
# Where to find workers.properties
JkWorkersFile conf/workers.properties
# Where to put jk logs
JkLogFile logs/mod_jk.log
# Set the jk log level [debug/error/info]
JkLogLevel info
# Select the log format
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
# JkOptions indicates to send SSK KEY SIZE
#JkOptions +ForwardKeySize +ForwardURICompat -
#ForwardDirectories
# JkRequestLogFormat
JkRequestLogFormat "%w %V %r"
JkMount /shoppingcart/* loadbalancer
```

We have changed  
only the application  
context in JkMount  
part



Open `/etc/httpd/conf/httpd.conf` file using vi and go to the end of the file(**Shift+g**) and make the following entry

**Include conf/mod-jk.conf**

Restart Apache Web Server and access the clustered application at

<http://192.168.56.101/shoppingcart/>

**-:END:-**

## **mod\_cluster as load balancer** **!! Optional !!**

**Configure mod\_cluster with the above cluster: (Mod cluster in JBoss.7.1.1 contains a bug bug JBPAPP-7516 and automatic advertise mode will not work and you need to provide proxy-list to Jboss node)**

### **1. Setup at httpd side:**

- Download, install and configure your httpd server according to your platform
- Get the shared object library from mod cluster download site [http://www.jboss.org/mod\\_cluster](http://www.jboss.org/mod_cluster)
- We are using mod cluster 1.2.0.Final version
- Copy the required .so files (as per the docs) to httpd/modules folder.
- Copy the following text at the bottom of the conf/httpd.conf file
- Our apache server is listening on 8888 port, please update IP address as per system IP.

```
##### mod_cluster Setting - STARTED #####
LoadModule slotmem_module modules/mod_slotmem.so
LoadModule manager_module modules/mod_manager.so
LoadModule proxy_cluster_module modules/mod_proxy_cluster.so
LoadModule advertise_module modules/mod_advertise.so

#Listen 1.1.1.1:80

<VirtualHost *:8888>
  <Directory />
    Order deny,allow
    Allow from all
  </Directory>
  <Location /mod_cluster-manager>
    SetHandler mod_cluster-manager
    Order deny,allow
    Allow from all
  </Location>
  EnableMCPMReceive
  KeepAliveTimeout 60
  ManagerBalancerName mycluster
  #AdvertiseGroup 224.0.1.105:23364
```

```
AdvertiseFrequency 2
    ServerAdvertise On
</VirtualHost>
##### mod_cluster Setting - ENDED #####
```

- g. Start apache server
- h. You are done!

## 2. JBoss setup:

- a. Open <JBoss\_HOME>/standalone-node1/standalone-ha.xml
- b. Update the <server> element with node name as standalone-node1  

```
<server name="standalone-node1" xmlns="urn:jboss:domain:1.2">
```
- c. Set proxy-list="your apache IP:apacheport" in modcluster subsystem as  

```
<subsystem xmlns="urn:jboss:domain:modcluster:1.0">
    <mod-cluster-config advertise-socket="modcluster" proxy-
list="192.168.135.134:8888">
        <dynamic-load-provider>
            <load-metric type="busyness"/>
        </dynamic-load-provider>
    </mod-cluster-config>
</subsystem>
```
- d. Update web subsystem with instance id as  

```
<subsystem xmlns="urn:jboss:domain:web:1.1" default-virtual-server="default-
host" instance-id="${jboss.node.name}" native="false">
```

Repeat the steps a-d for node2 also.  
Restart JBoss nodes. Enjoy!