EAP 6.1 quickstart on CentOS

[Getting started with Red Hat EAP 6.1 on a CentOS Virtual Machine]

[1/8/2013]

Francesco
Marchioni

Contents

Setting up JBoss EAP 6 on Linux Centos VM

Welcome to the "**JBoss EAP 6 on Linux Centos Virtual Machine**" eBook. The purpose of this mini book is demonstrate how to setup a Centos Virtual Machine featuring the **JBoss Enterprise Application Platform 6.1**.

The **JBoss Enterprise Application Platform (EAP)** is part of the Red Hat *JBoss Enterprise Middleware* portfolio of software. It can be used to build fast, secure and powerful applications built upon open standards, and compliant with the Java Enterprise Edition 6 specification

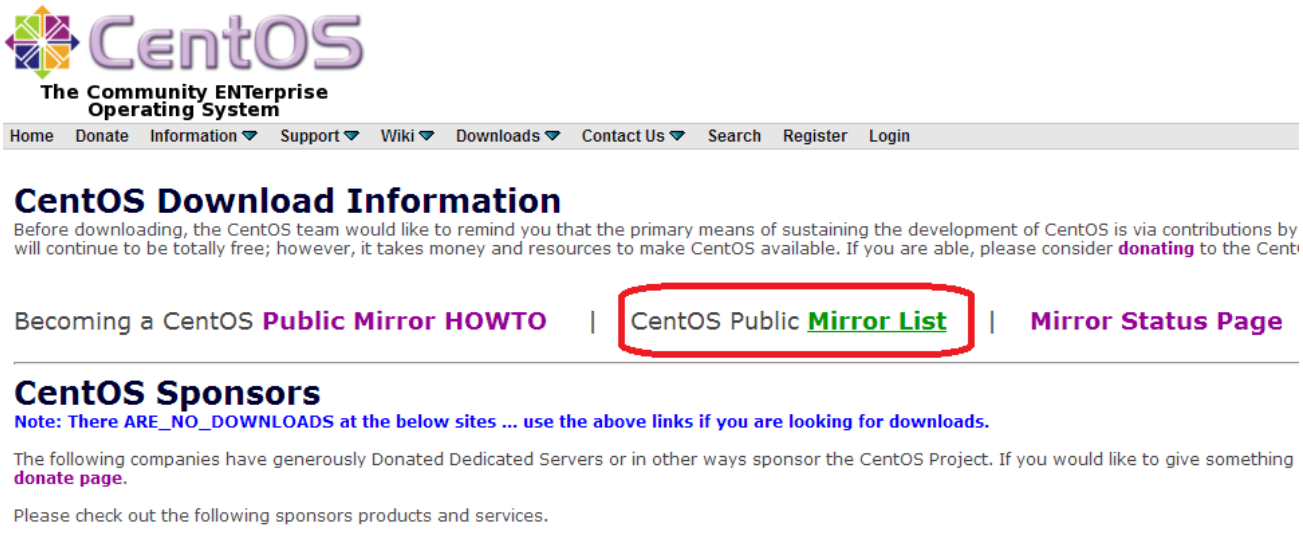
The **Linux CentOS** team is a volunteer group that repackages publicly available Red Hat Enterprise Linux (RHEL) source packages into binaries. The Linux CentOS software is then distributed through various public mirrors. There is no official, commercial support service for CentOS; if this puts you off to use this software in production, you can however use Centos to leverage RHEL skills and also for a setting up a quick demo or test environment.

At the end of this book you will learn some basic skills including:

- Setting up a Linux Virtual Machine using an Industry standard product such as Oracle'sVirtualBox
- Install the JBoss EAP 6.1 and its required software (JDK)
- Configure the JBoss EAP 6.1 as a service

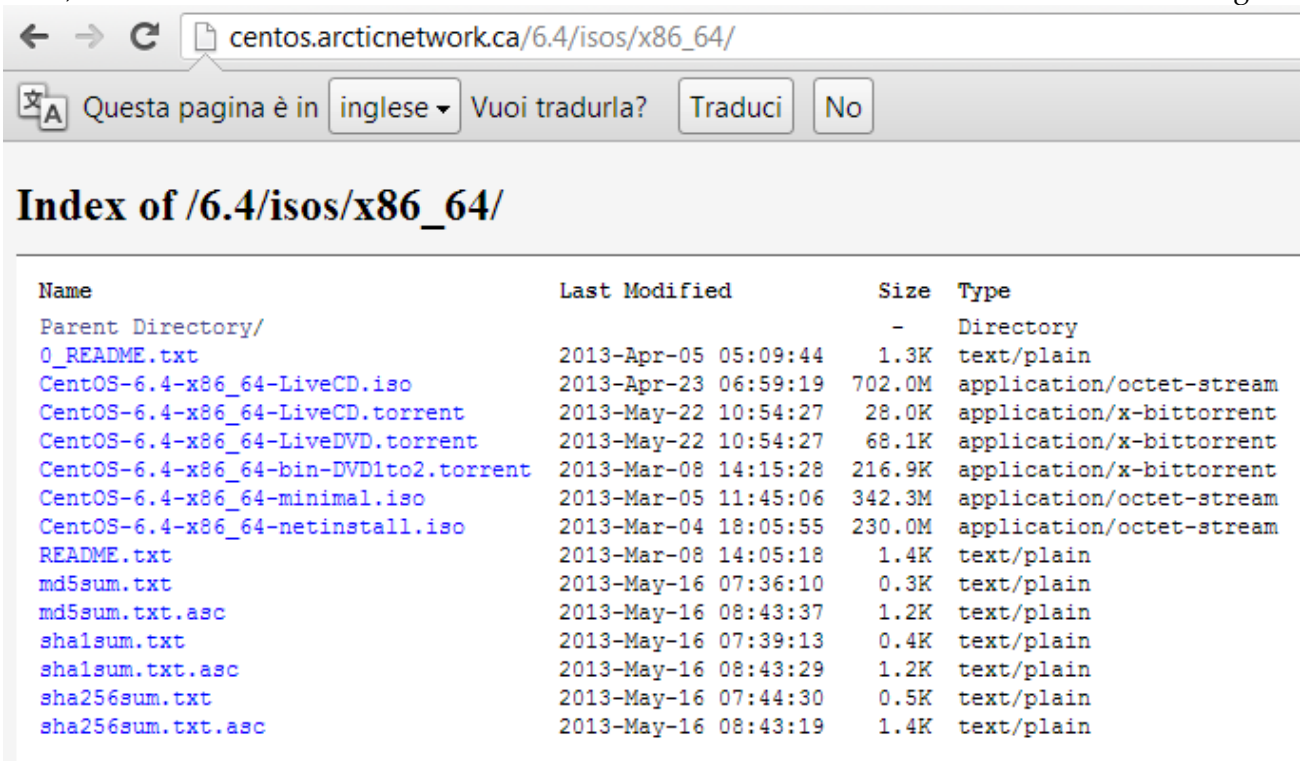
Step 1: Download CentOS distribution

This part will guide you through a step-by-step installation of **Community ENTerprise Operating System 6.4** (CentOS) with screenshots. First off visit http://www.centos.org and download ISO image(s) for your servers architecture (i686, x86_64 etc).



ISO Images are not directly hosted on the CentOS site but you have to pick them up from mirror sites which are reachable from above highlighted link.

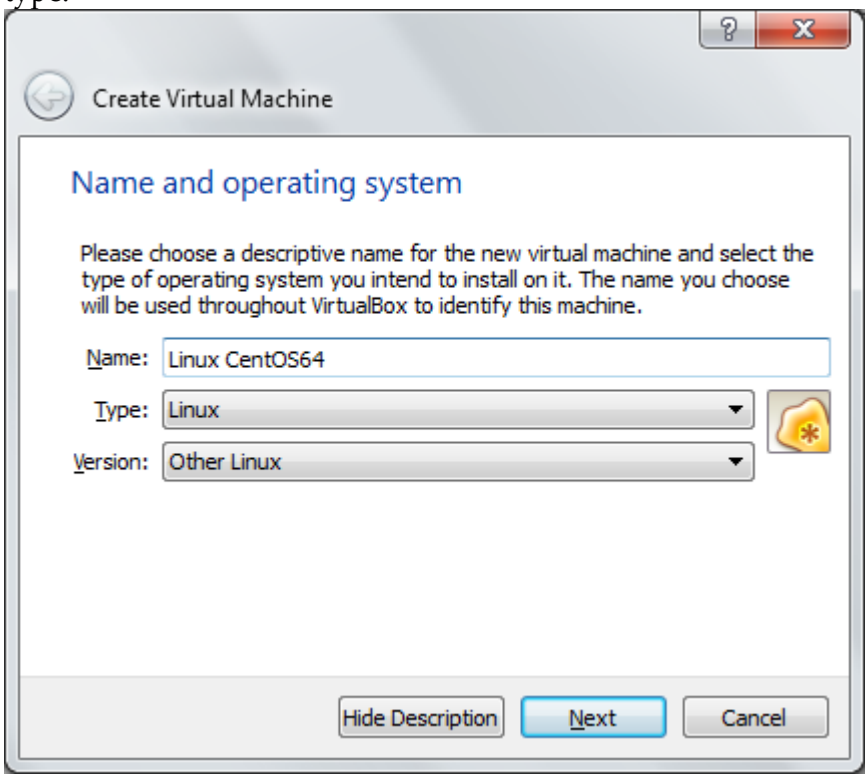
Next, select the latest stable release and choose to download either the LiveCD or the DVD images



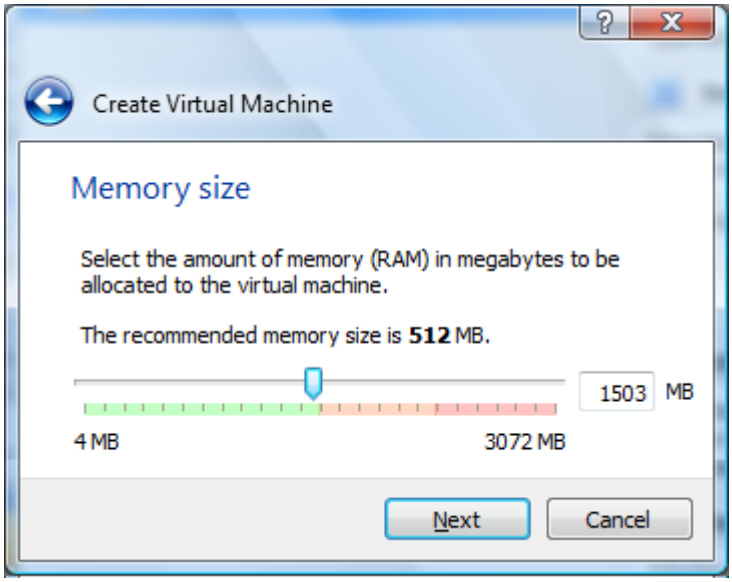
I would recommend the torrent option of downloading especially for the DVD images; this is by far the fastest and safest way as torrents automatically perform md5 check on the download images preventing file corruption (which is common on large file downloads).

Step 2: Create a Virtual Machine

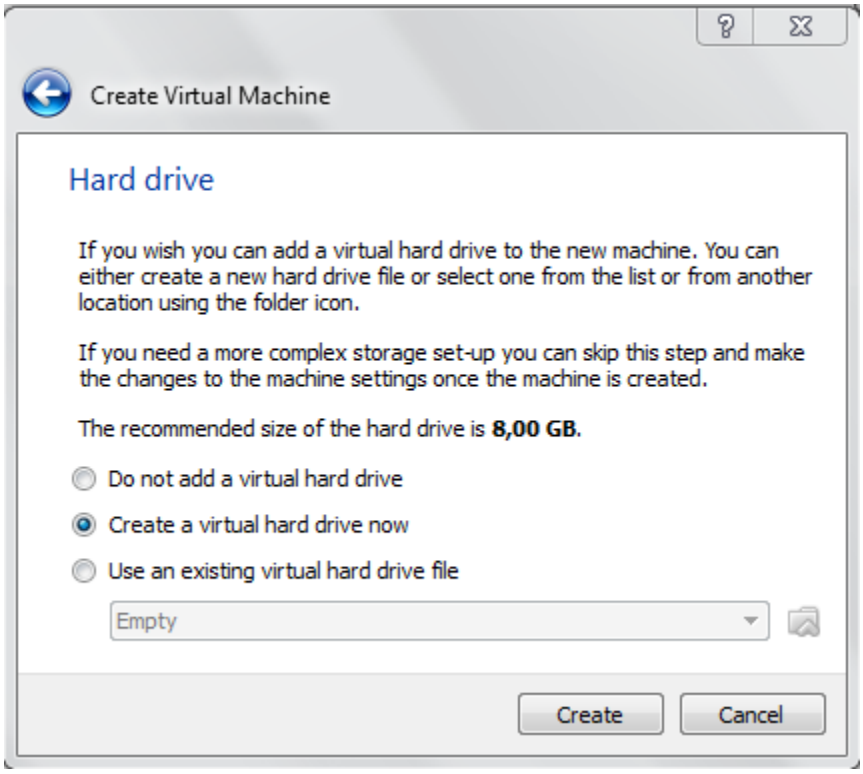
Once downloaded the CD/DVD it's time to create a new Virtual Machine using your favourite VM Tool such as **Oracle's Virtual Box**.
Start Virtual Box and click on the "**New**" button at the top of the Virtual Box Manager window. A wizard will pop up to guide you through setting up a new Virtual Machine (VM). In the first screen enter the VM name and the Operating system you intend to install on it. In our case, we will install a Linux CentoOS which is not mentioned through the defaults; therefore select "**Other Linux**" as type:



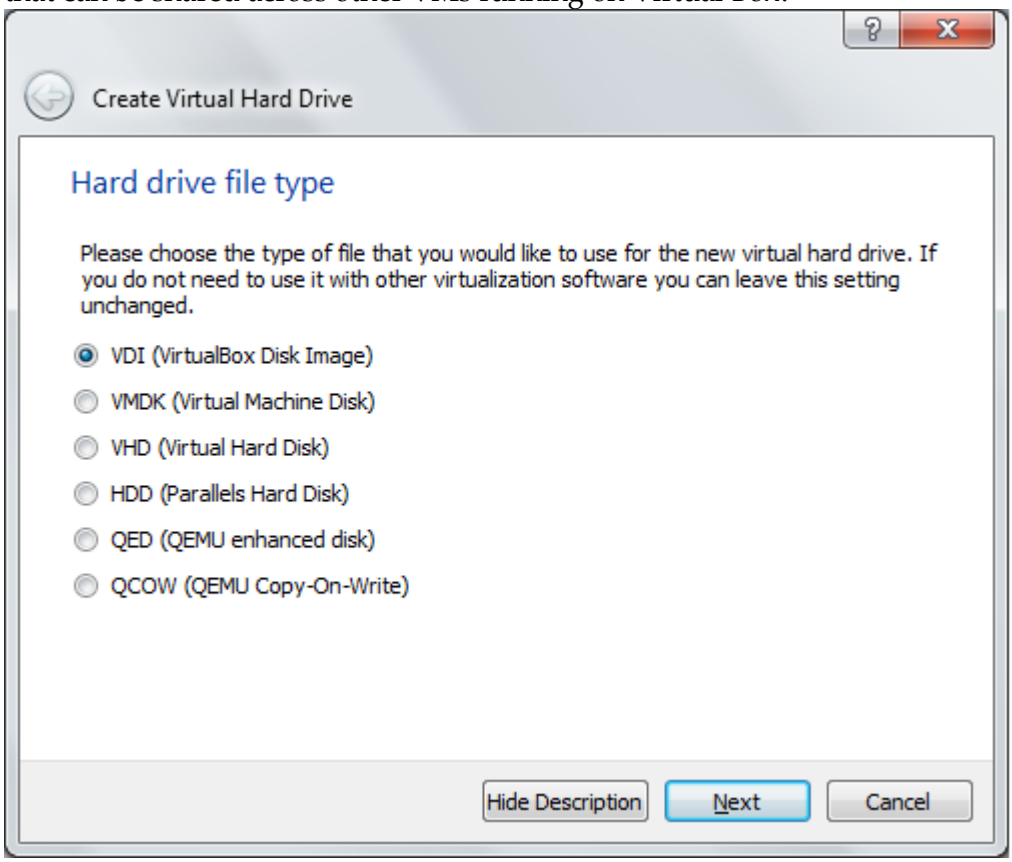
Click Next. In the following screen select the amount of memory which will be assigned to the VM. Since I don't have lots of memory available on this machine, I'll just create a tiny Linux VM with about 1500 MB of RAM. Click Next:



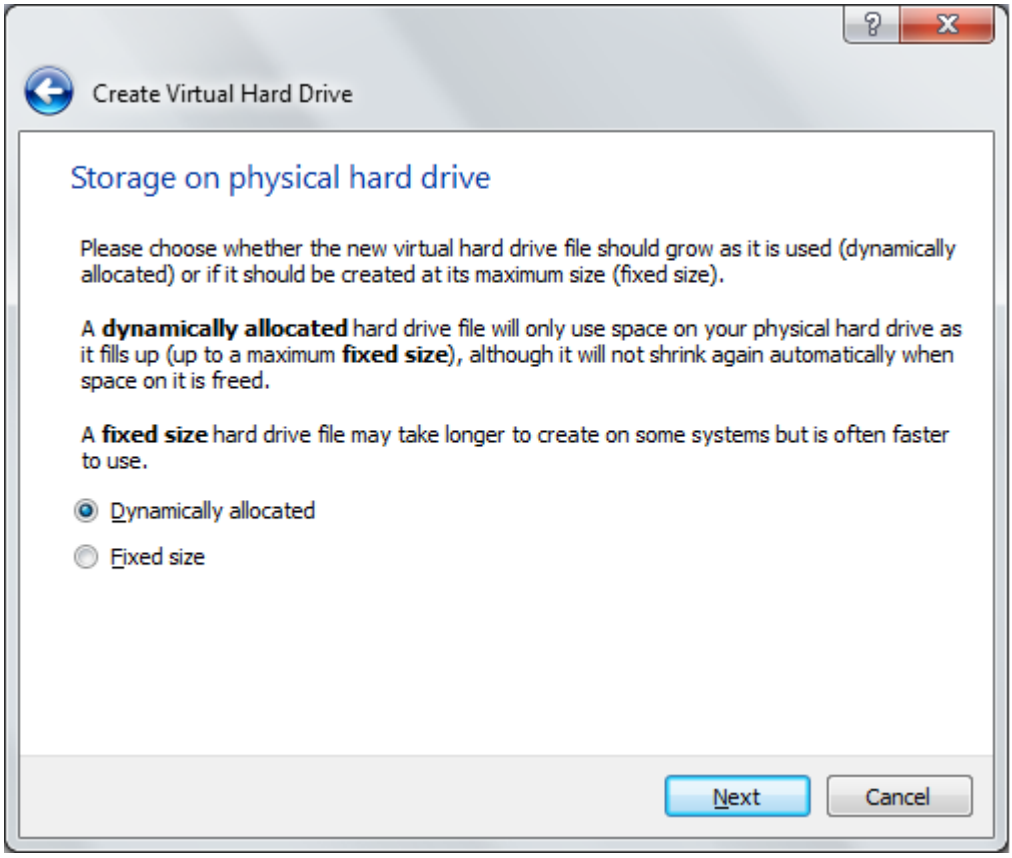
In the next screen we will decide to create an Hard drive for our VM where data will be stored:



Choose Create. In the next window select the default option, that is using a Virtual Box Disk image that can be shared across other VMs running on Virtual Box:



Click Next. This will bring up the Storage Type window:



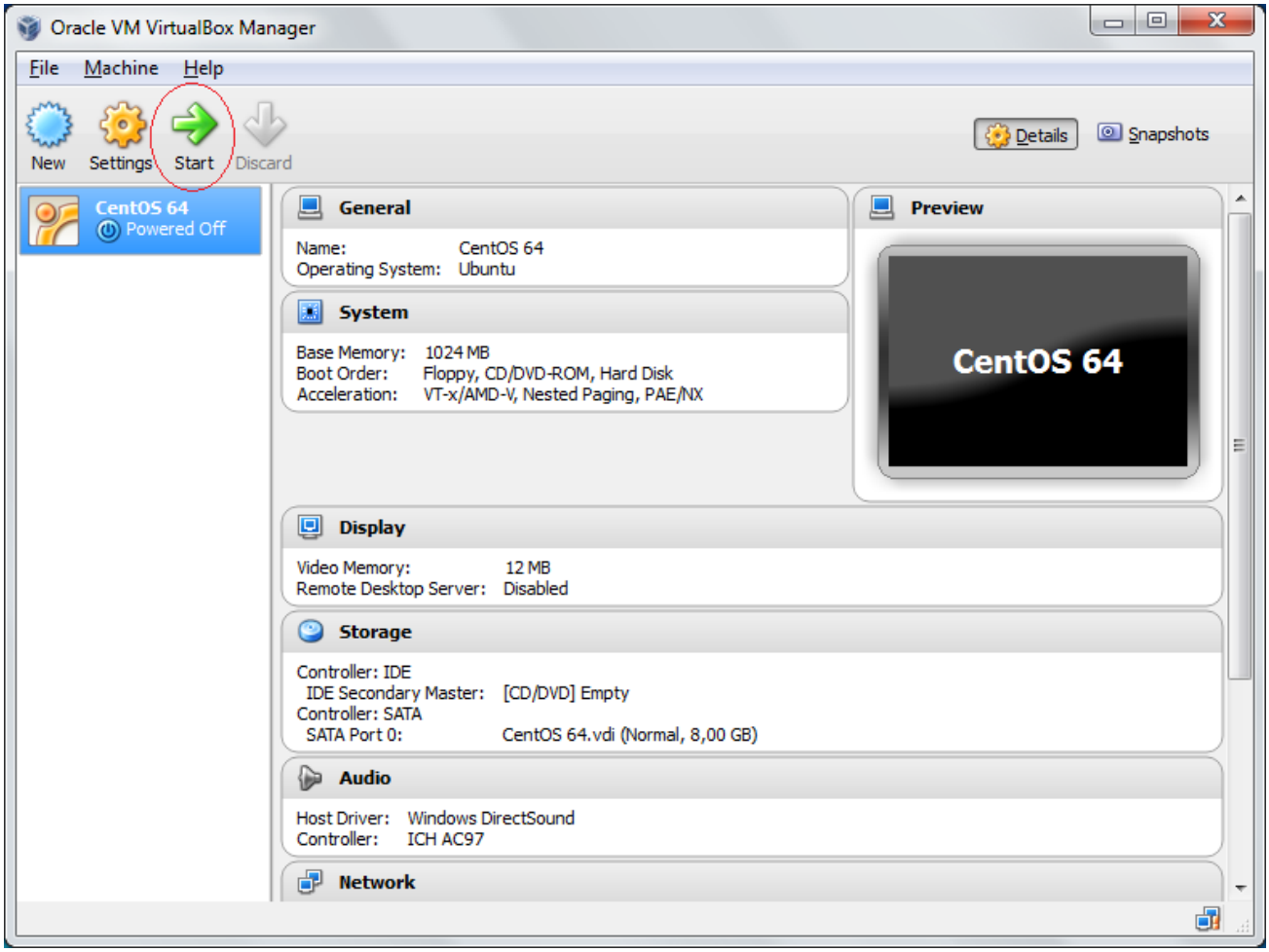
In this screen you have to choose how to store your data on the physical hard drive. As a matter of fact, VirtualBox supports two types of image files:

- A **dynamically allocated** file will only grow in size when the guest actually stores data on its virtual hard disk. It will therefore initially be small on the host hard drive and only later grow to the size specified as it is filled with data.
- A **fixed-size** file will immediately occupy the file specified, even if only a fraction of the virtual hard disk space is actually in use. While occupying much more space, a fixed-size file incurs less overhead and is therefore slightly faster than a dynamically allocated file.

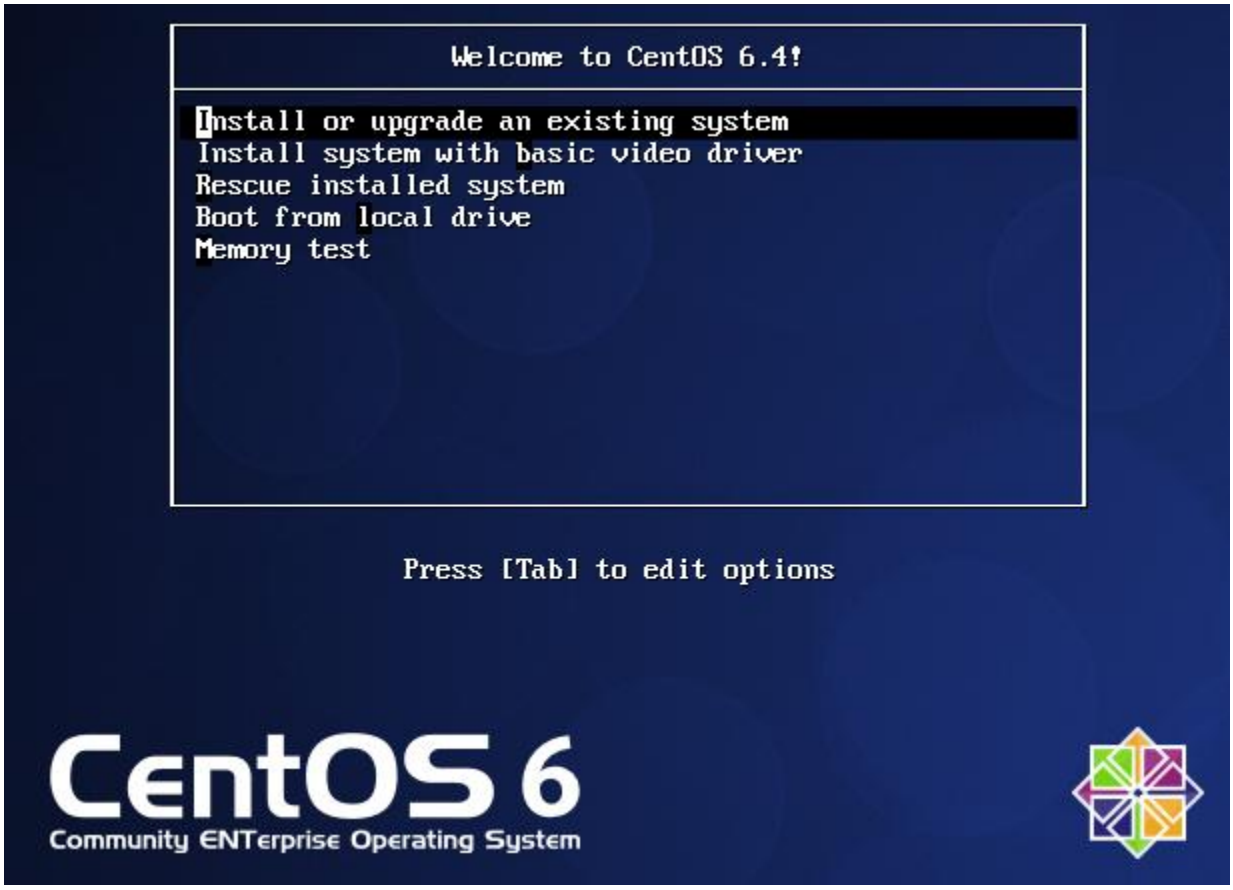
We will choose the first option which will prevent your hard disk growing up too much, provided that you have little data stored on it. Fine, now your VM Creation is complete.

Step 3: Installing CentOS

Now it's time to boot your Virtual Machine. Click on the Start button and the CentOS installation will start.

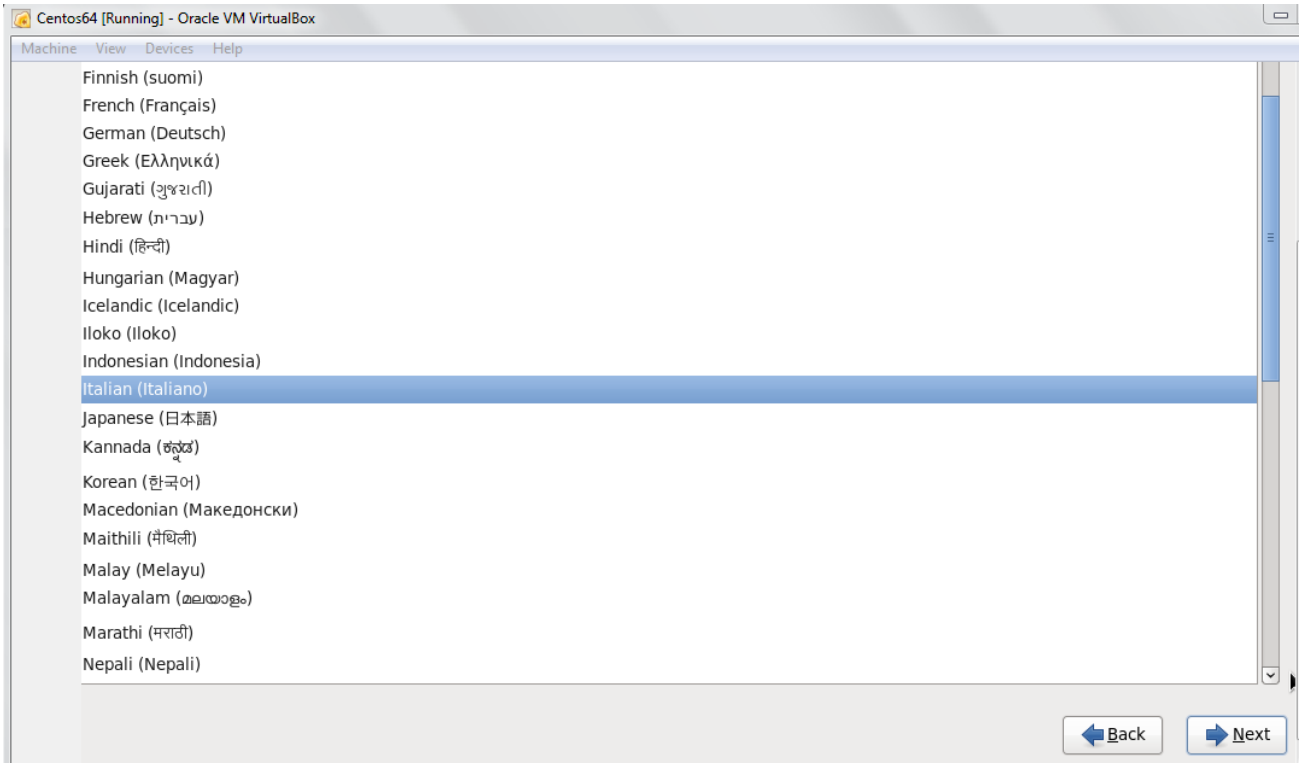


As first step you will be requested to point out where your ISO files are. Next, the CentOS installation will start as displayed by the following picture:

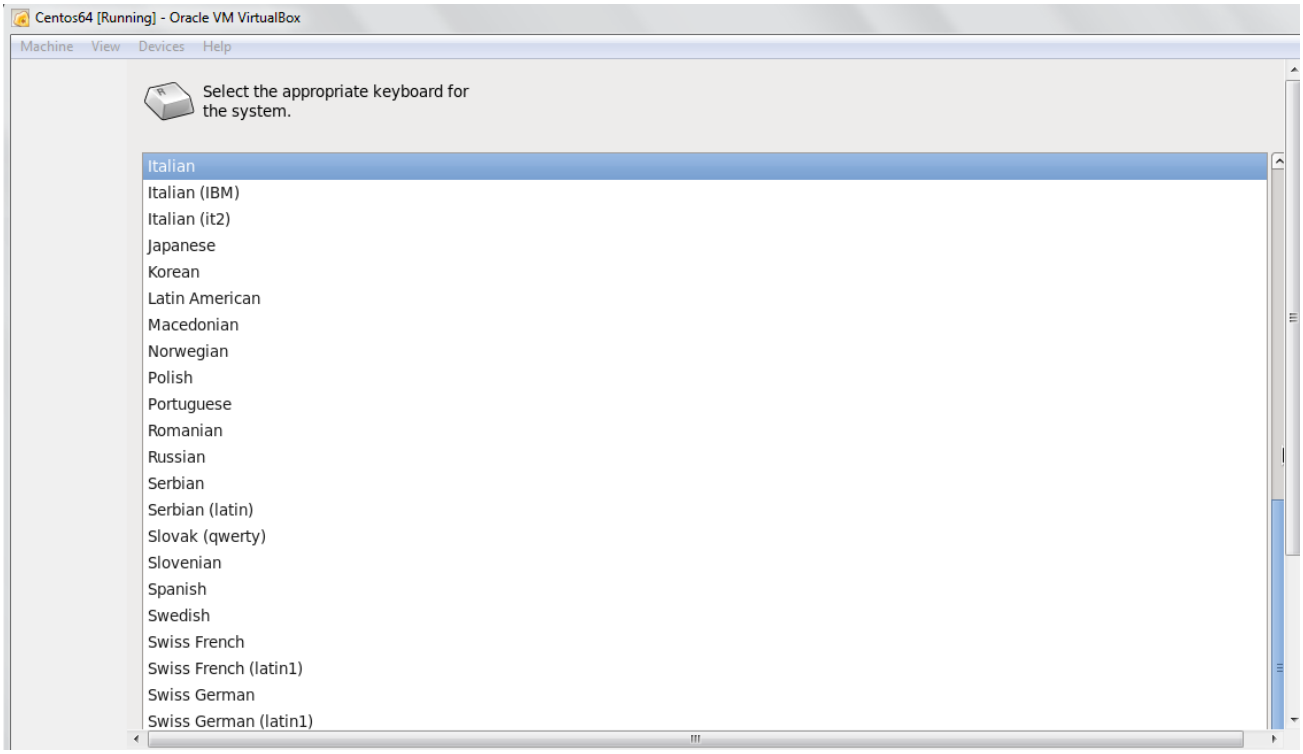


As we created a Media Device using the Virtual Box tool, the installer will formerly request a Disc Drive Check:

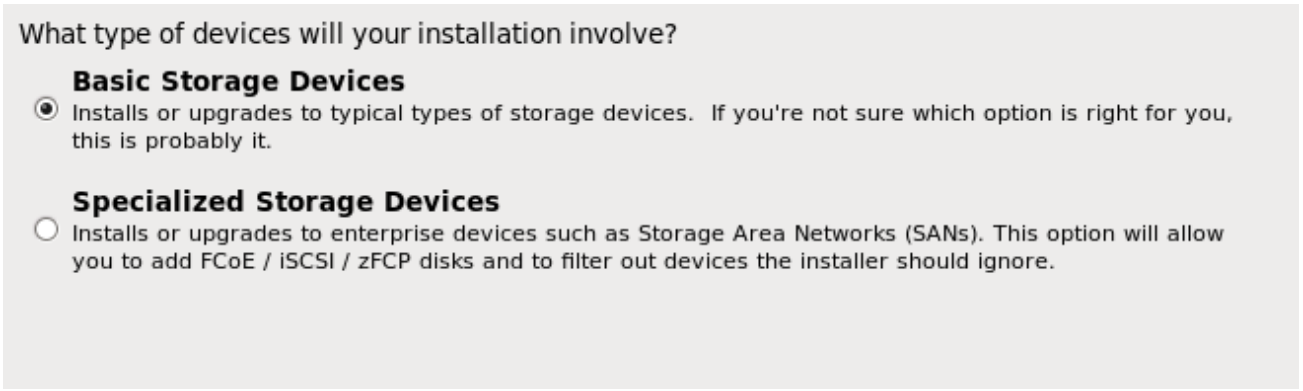
We are confident enough to skip the device test so just hit the Skip button which will take you to the actual installation. The installation is pretty simple: you have at first to enter the language that will be used across the installation:



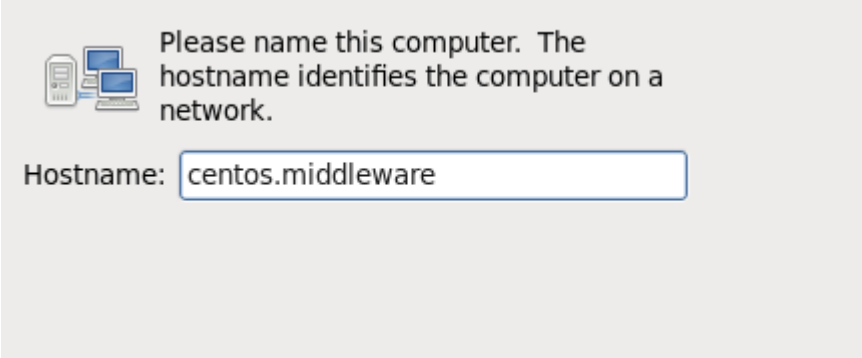
Next, enter the appropriate keyboard settings as shown by the following picture:



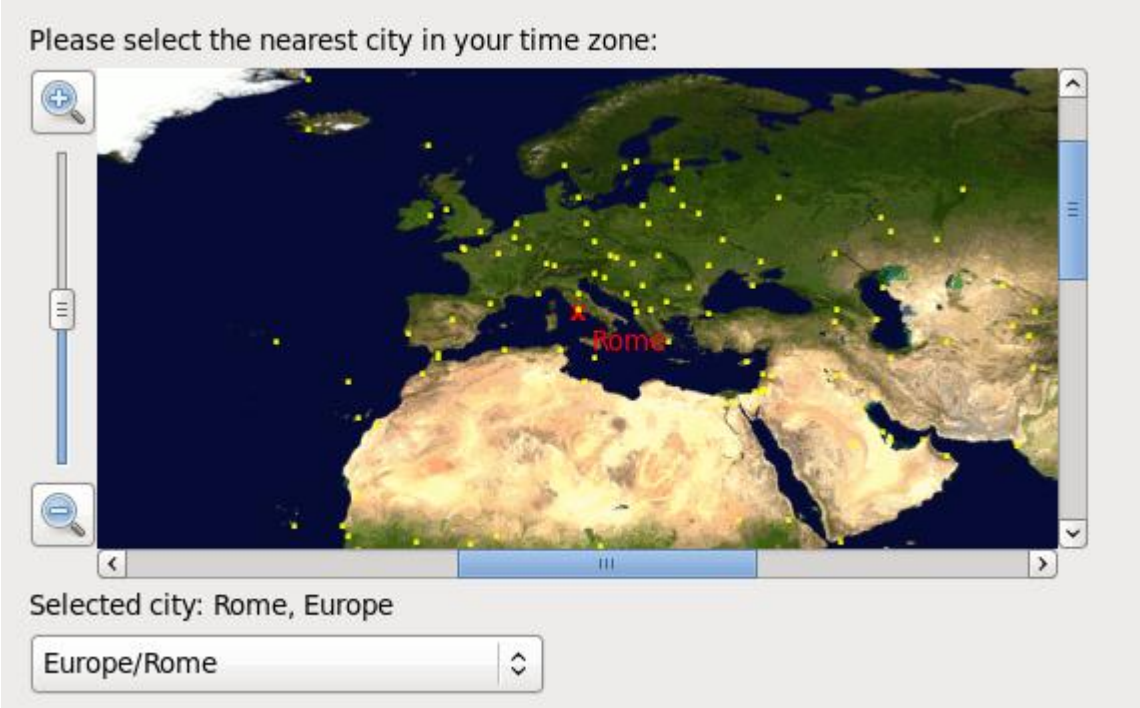
In the following step you will select which kind of devices you are going to install. As we don't need to install a Specialized Storage such as a SAN, just select a **Basic Storage Device**:



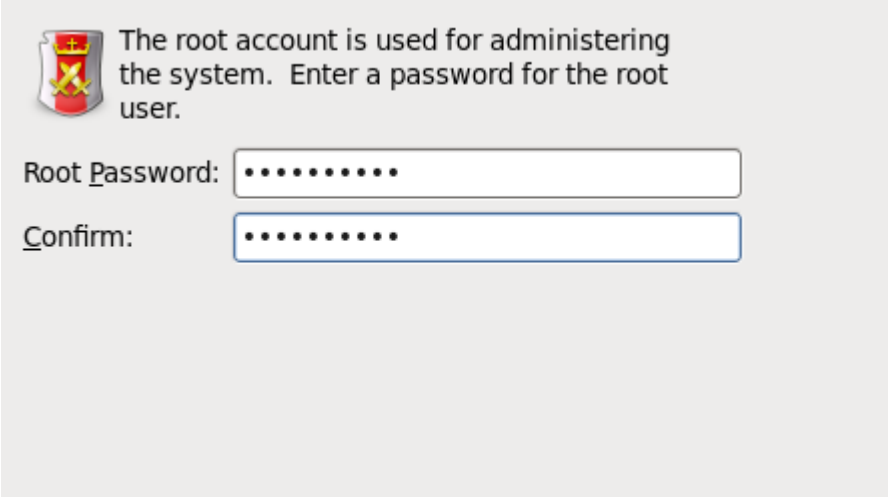
Next, enter the **Hostname** information as shown by the following snapshot:



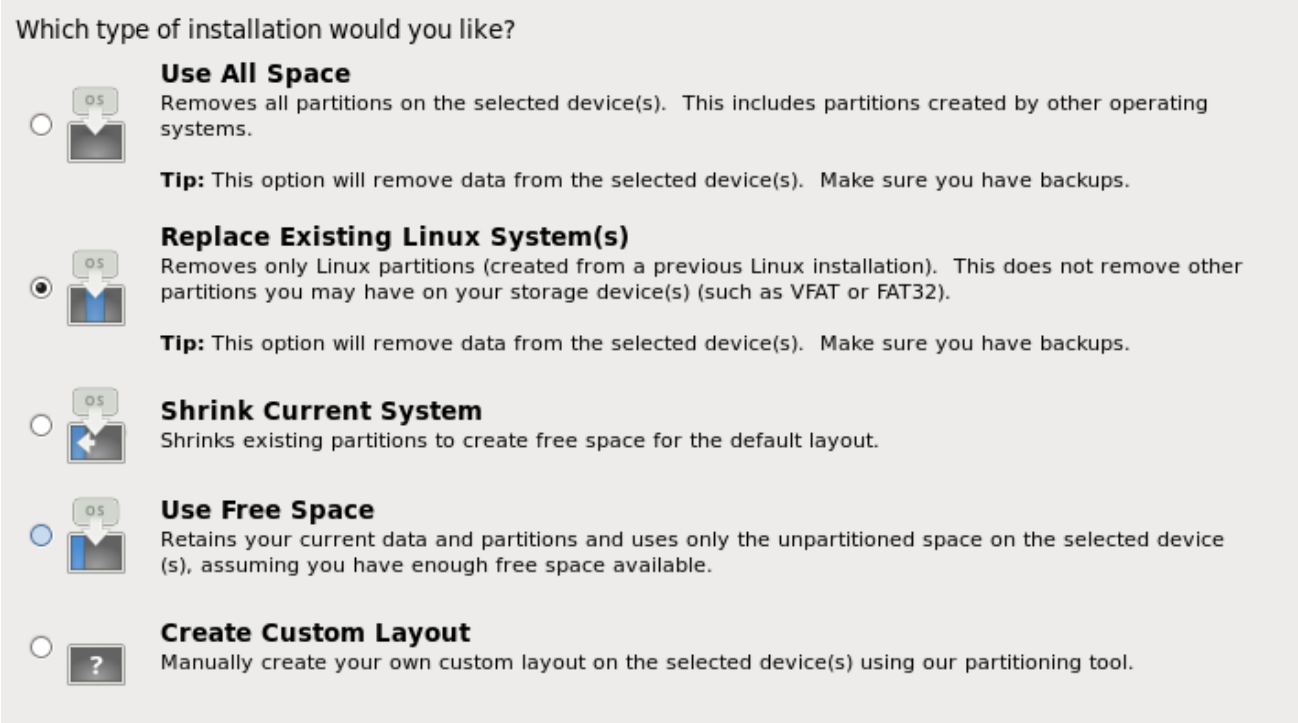
In the next screen enter the appropriate Time zone for your CentOS machine:



Next step will be entering the password for the super user:



Finally, before writing changes to disk, select the type of installation that will be used. Being a Virtual Machine Installation you can either **Use All Space** of **Replace Existing Linux Systems**:



Beware that selecting the "Use All Space" on a physical machine will erase all existing partitions!

At the end of the installation, you will need to reboot your Virtual Machine.

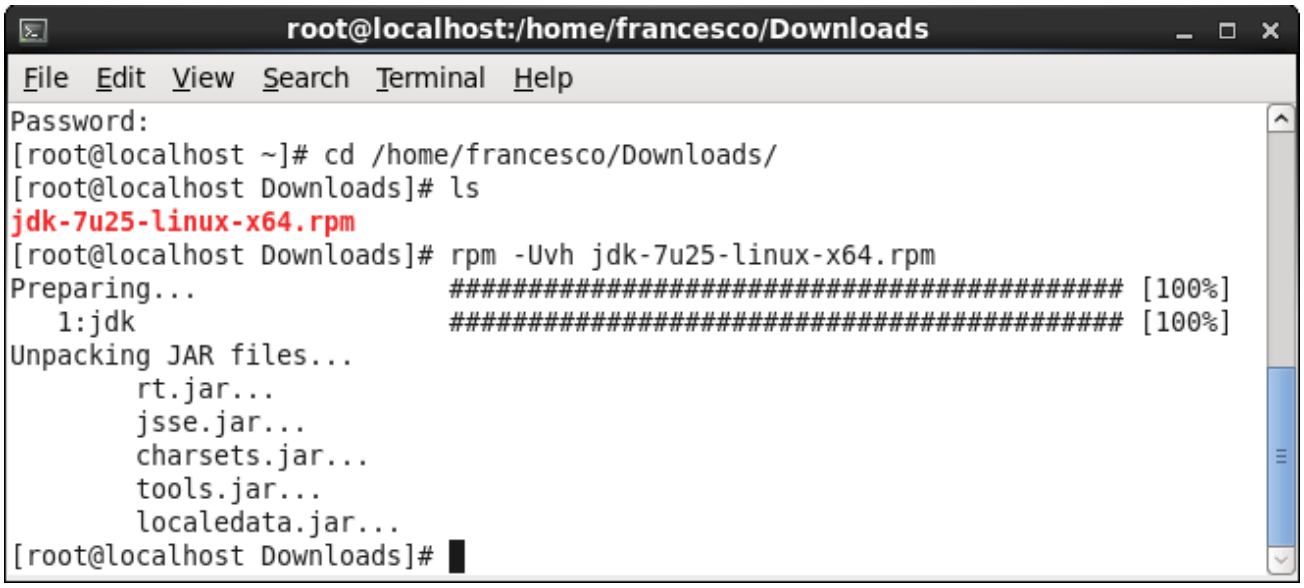
Step 4: Download and install the JDK

Once that you have logged in, the first step will be installing the Java Virtual Machine which is a pre-requisite for running JBoss EAP. Pickup the latest JDK stable release from: http://www.oracle.com/technetwork/java/javase/downloads/index.html

Linux distributions are available as tar.gz or rpm. We will choose to download the rpm package which will be faster to install

Once downloaded, as root execute:

```
[root@dev2 ~]# rpm -Uvh jdk-7u25-linux-x64.rpm
```



Now verify that the JDK has been correctly installed by issuing:

```
[root@dev2 ~]# java -version
java version "1.7.0_25"
Java(TM) SE Runtime Environment (build 1.7.0_25-b11)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 23.25-b01, mixed mode)
```

Step 5: Download and Install JBoss EAP 6.1.0

As next step, download the EAP 6.1.0 from http://www.jboss.org/jbossas/downloads

In order to download the EAP 6 you need formerly to register on www.jboss.org and accept to use the Red Hat subscription form.

Once downloaded, copy the zip file under the /usr/share which is the recommended file system for JBoss EAP:

```
[root@dev2 ~]# mv jboss-eap-6.1.0.zip /usr/share
```

Now enter the /usr/share folder:

```
[root@dev2 ~]# cd /usr/share
```

Unzip the jboss-eap archive using a tool like the **jar** utility which is part of the JDK

```
[root@dev2 share]# jar xvf jboss-eap-6.1.0.zip
```

Create user and group

We suggest creating an user named **jboss** which belongs to the **jboss** group that will be in charge to start/stop the application server

```
[root@dev2 share]# groupadd jboss
[root@dev2 share]# useradd -s /bin/bash -g jboss jboss
```

In order to make the EAP platform usable for the jboss user, you have to change the ownership of the whole jboss-eap-6.1 folder

```
[root@dev2 share]# chown -Rf jboss.jboss /usr/share/jboss-eap-6.1
```

Fine; now before starting the EAP we have to set the JAVA_HOME environment variable and include it as part of the system PATH. Let's do it for both the user root and the user jboss:

```
[root@dev2 share]# cd ~
[root@dev2 ~]# vi .bash_profile
```

Enter the following profile (adjust the path to the name and location of your JDK):

```
JAVA_HOME=/usr/java/jdk1.7.0_25
export JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

Perform the same steps for the user jboss:

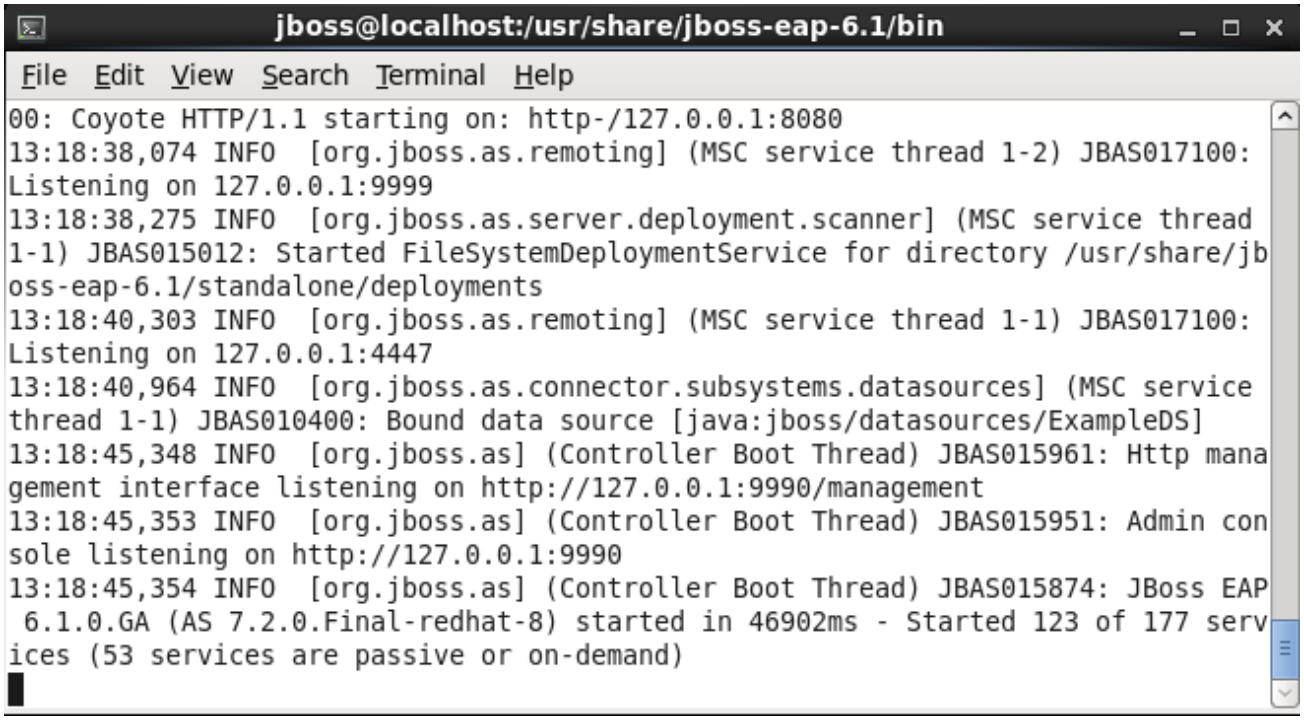
```
[root@dev2 share]# su - jboss
[jboss@dev2 ~]# vi .profile
```

Enter the same information in it:

```
JAVA_HOME=/usr/java/jdk1.7.0_25
export JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

Now read the profile environment and verify that the EAP 6.1 starts correctly:

```
[jboss@dev2 ~]# source .profile
[jboss@dev2 share]# cd /usr/share/jboss-eap-6.1
[jboss@dev2 jboss-eap-6.1]# chmod -R 755 *.sh
[jboss@dev2 jboss-eap-6.1]# cd bin
[jboss@dev2 share]# ./standalone.sh
```



Step 6: Install JBoss EAP as a service.

Once that your environment is ready, we will configure to boot the EAP as a service. This is a two-step process which requires at first copying the **JBOSS_HOME/bin/jboss-as-standalone.sh** under the **/etc/init.d** folder (For convenience we will rename the script as "jboss")

```
[jboss@dev2 ~]# su - root
[root@dev2 bin]# cp /usr/share/jboss-eap-6.1/bin/init.d/jboss-as-standalone.sh
/etc/init.d/jboss
```

At first set permissions for the script file so that it can be executed by jboss user

```
[root@dev2 bin]# chmod 755 /etc/init.d/jboss
```

Now, we need to do some small adaptations to this file so that it will be usable by the **chkconfig** tool.

Chkconfig command is used to setup, view, or change services that are configured to start automatically during the system startup.

Here are is an excerpt of the first part of the file, with the required changes highlighted:

```
#!/bin/sh
#
# JBoss standalone control script
#
# chkconfig: 2345 80 20
# description: JBoss AS Standalone
# processname: standalone
# pidfile: /var/run/jboss-as/jboss-as-standalone.pid
# config: /etc/jboss-as/jboss-as.conf

# Source function library.
. /etc/init.d/functions

# Load Java configuration.
[ -r /etc/java/java.conf ] && . /etc/java/java.conf
export JAVA_HOME

JBOSS_USER=jboss
```

This are the service boot levels which need to be specified if you want that chkconfig is able to validate this file

This is the system user that will have the ownership of the JVM process

```
export JBOSS_USER



JBOSS_HOME=/usr/share/jboss-eap-6.1
export JBOSS_HOME
```

Once done with the changes, save the file. Next, we will use the chkconfig command to install the EAP as a service. The first command will add the jboss shell script to the chkconfig

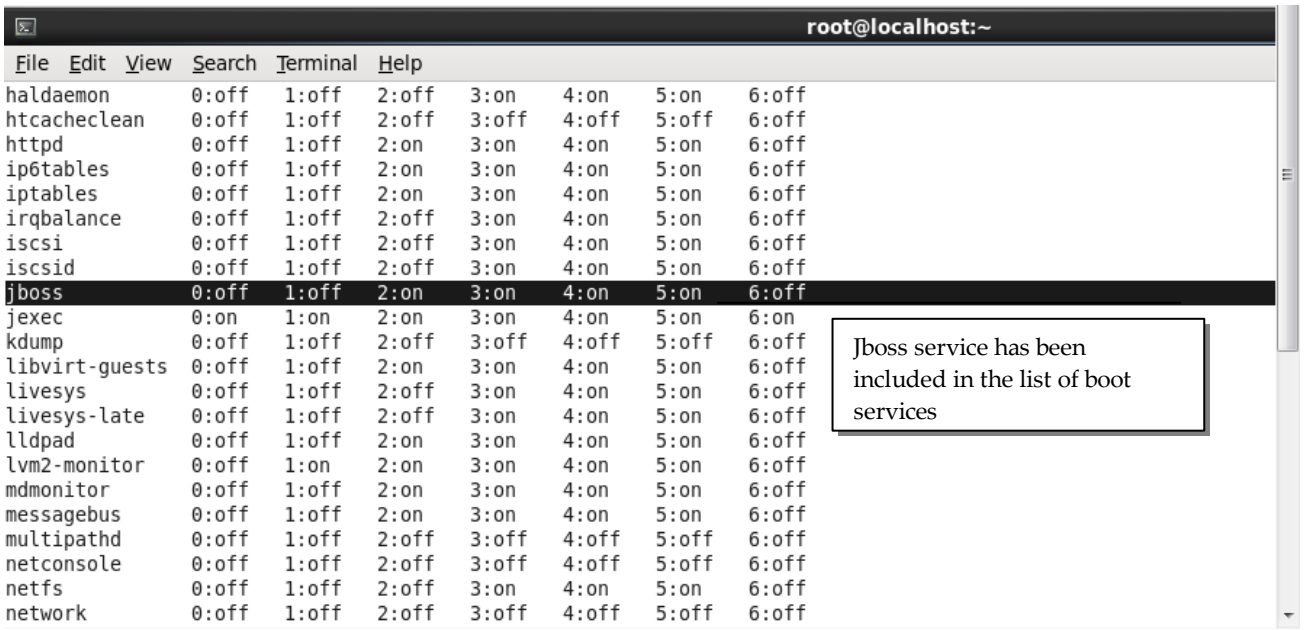
```
[root@dev2 bin]# cd /etc/init.d/
[root@dev2 init.d]# chkconfig --add jboss
```

The next one sets the boot level where the service will be started

```
[root@dev2 init.d]# chkconfig --level 2345 jboss on
```

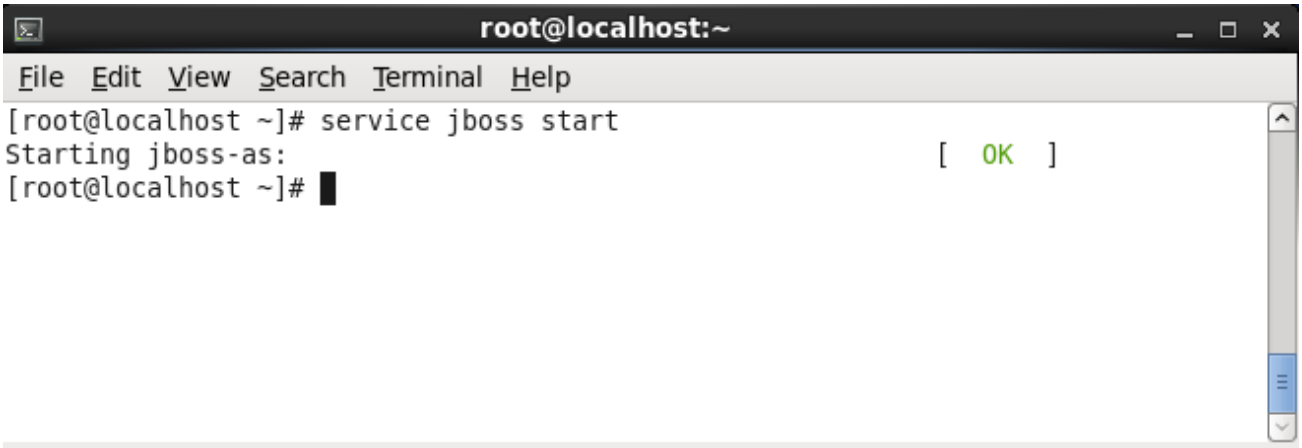
Now, verify that the jboss service has been correctly included in the chkconfig list:

```
[root@dev2 init.d]# chkconfig --list
```



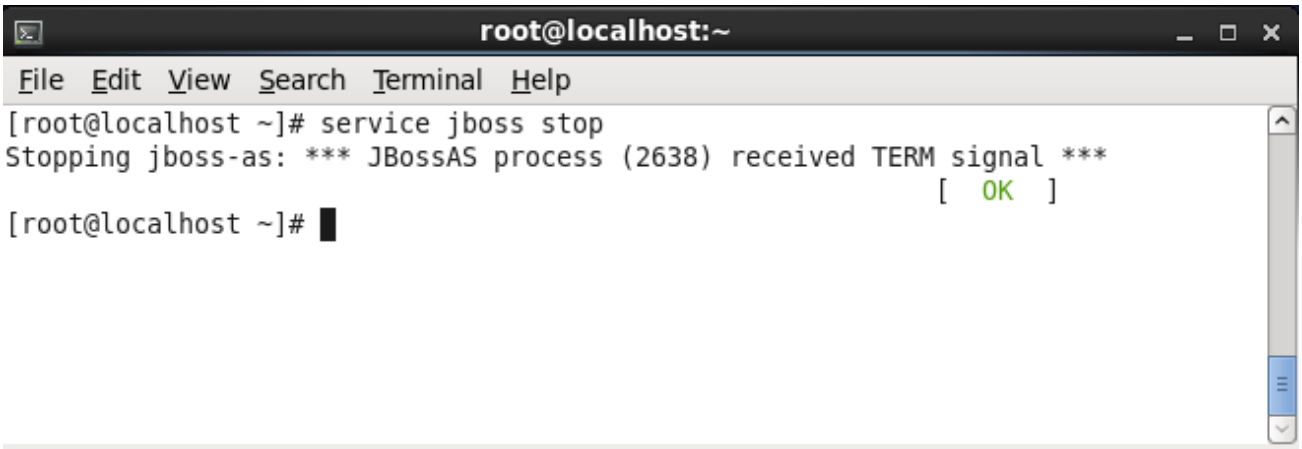
Let's test if the service starts correctly:

```
service jboss start
```



And here's the corresponding service stopping:

```
service jboss stop
```



As a last tip, add the ownership of the /var/log/jboss-as/ folder to the jboss user, otherwise you will get some annoying log errors when the service is started directly by the jboss user.

```
chown -Rf jboss:jboss /var/log/jboss-as/
```

That's all. Now your VM will automatically start the EAP 6.1 as soon as the system is booting.