Apache Hadoop 2.0 Installation and Single Node Cluster Configuration on Ubuntu

A guide to install and setup Single-Node Apache Hadoop 2.0 Cluster

edureka!

© 2014 Brain4ce Education Solutions Pvt. Ltd.

Version 1.0

Apache Hadoop 2.0 Installation and Single Node Cluster Configuration on Ubuntu

A guide to install and setup Single-Node Apache Hadoop 2.0 Cluster

Table of Contents

Software Requirements....4
Hardware Requirements....4
Introduction....5
1. Setting up the Ubuntu Desktop....6
1.1 Creating an Ubuntu VM Player instance....6
1.1.1 Download the VMware image....6
1.1.2 Open the image file....6
1.1.3 Play the Virtual Machine....8
1.1.4 Update the OS packages and their dependencies....10
1.1.5 Install the Java and openssh server for Hadoop 2.2.0....10
1.2 Download the Apache Hadoop 2.2.0 binaries....10
1.2.1 Download the Hadoop package....10
2. Configure the Apache Hadoop 2.2.0 Single Node Server....13
2.1 Update the Configuration files....13
2.1.1 Update ".bashrc" file for user 'ubuntu'....13
2.2 Setup the Hadoop Cluster....15
2.2.1 Configure JAVA_HOME....15
2.2.2 Create NameNode and DataNode directory....15
2.2.3 Configure the Default File system....16
2.2.4 Configure the HDFS....17
2.2.5 Configure YARN framework....19
2.2.6 Configure MapReduce framework....21
2.2.7 Edit /etc/hosts file....22
2.2.9 Creating ssh....23
2.2.10 Moving the key to authorized key:....24

edureka!

2.2.11 Start the DFS services 25
2.2.12 Perform the Health Check 27
3. Installing Pig on Hadoop-2.2.0 28
3.1 Download Pig-0.12.0 28
3.2 Untar the package 29
3.3 Edit .bashrc file 29
3.4 Configure JAVA_HOME.....30
3.5 Start Grunt Shell 31
4. Installing Hive on Hadoop-2.2.0 32
4.1 Download Hive-0.11.0 32
4.2 Untar the package 33
4.3 Edit .bashrc file 33
4.4 Create Hive directories on HDFS 34
4.5 Set Hadoop path in hive-config.sh 34
4.6 Start hive shell 35
5. Installing Hbase on Hadoop-2.2.0 36
5.1 Download Hbase-0.96.2-hadoop2-bin.tar.gz 36
5.2 Untar the package 37
5.3 Edit .bashrc file 37
5.4 Create Hbase directory.....38
5.5 Edit hbase-site.xml 38
5.6 Edit hbase-env.sh 40
5.7 Start Hbase 41
5.8 Start Hbase Shell.....41
6. Installing Sqoop on Hadoop-2.2.0 42
6.1 Download Sqoop-1.4.4 42
6.2 Untar the package 43
6.3 Edit .bashrc file 43
6.4 Download Sqoop Jars 44
6.5 Check Sqoop Installation 45
7. Installing Oozie on Hadoop-2.2.0 46
7.1 Download Oozie-4.0.0 46
7.2 Untar the package 46
7.3 Edit .bashrc file 47
7.4 Make an User oozie 47

edureka!

7.5 Give Permission to Oozie User 48
7.6 Edit core-site.xml.....48
7.7 Download libext.tar.gz 49
7.8 Untar libext.tar.gz.....49
7.9 Move libext Directory to oozie-4.0.0 Directory.....50
7.10 Prepare War File.....50
7.11 Create Oozie Database 51
7.12 Start Oozie 51
7.13 Oozie Web Console 52

edureka!

Software Requirements
✓ VMware Player ✓ 12.0.4 Ubuntu image
Hardware Requirements
✓ Intel Core i3 processor or higher ✓ 8 GB RAM Recommended (will work with 4 GB also) ✓ 100 GB for VM Recommended (By default 40 GB is taken)

edureka!

Introduction

This setup and configuration document is a guide to setup a Single-Node Apache Hadoop 2.0 cluster on an Ubuntu virtual machine on your PC. If you are new to both Ubuntu and Hadoop, this guide comes handy to quickly setup a Single-Node Apache Hadoop 2.0 Cluster on Ubuntu and start your Big Data and Hadoop learning journey.

The guide describes the whole process in two parts:

Section 1: Setting up the Ubuntu OS for Hadoop 2.0

This section describes step by step guide to download, configure an Ubuntu Virtual Machine image in VMPlayer, and provides steps to install pre-requisites for Hadoop Installation on Ubuntu.

Section 2: Installing Apache Hadoop 2.0 and Setting up the Single Node Cluster

This section explains primary Hadoop 2.0 configuration files, Single-Node cluster configuration and Hadoop daemons start and stop process in detail.

Note: The configuration described here is intended for learning purposes only.

edureka!

1. Setting up the Ubuntu Desktop

This section describes the steps to download and create an Ubuntu image on VMware Player.

1.1 Creating an Ubuntu VM Player instance

The first step is to download an Ubuntu image and create an Ubuntu VMPlayer instance.

1.1.1 Download the VMware image

Access the following link and download the 12.0.4 Ubuntu image:

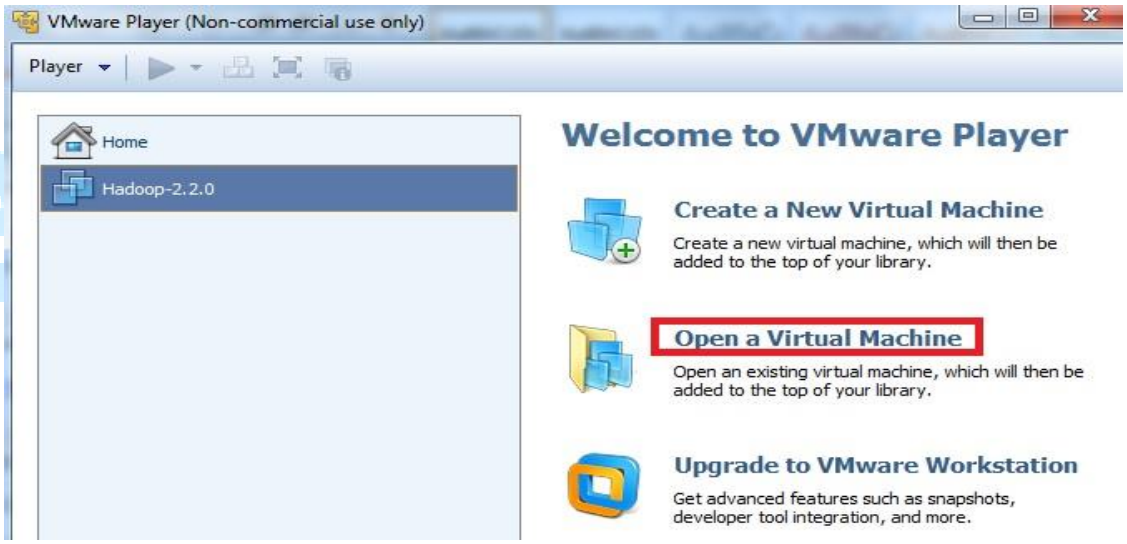
http://www.traffictool.net/vmware/ubuntu1204t.html

1.1.2 Open the image file

Extract the Ubuntu VM image and Open it in VMware Player.

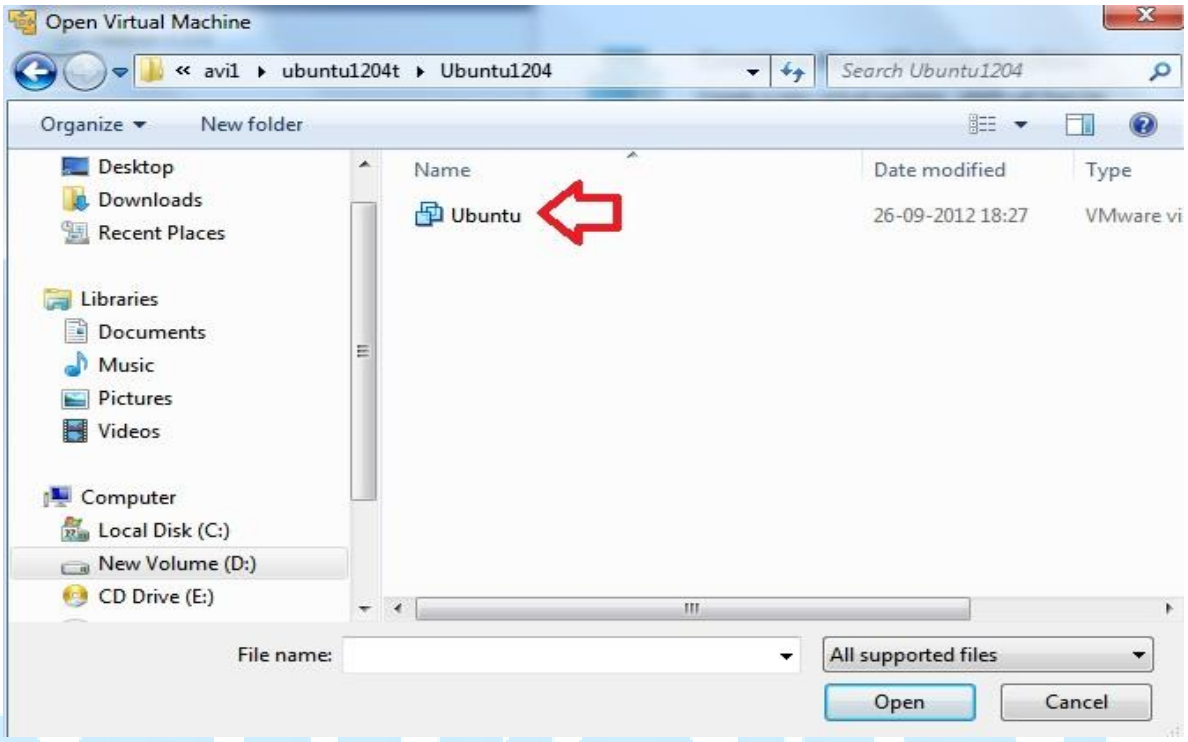
Click open virtual machine and select path where you have extracted the image.

FIGURE 1-1 OPEN THE VM IMAGE



Select the **'.vmx'** file and click **'ok'**.

FIGURE 1-2

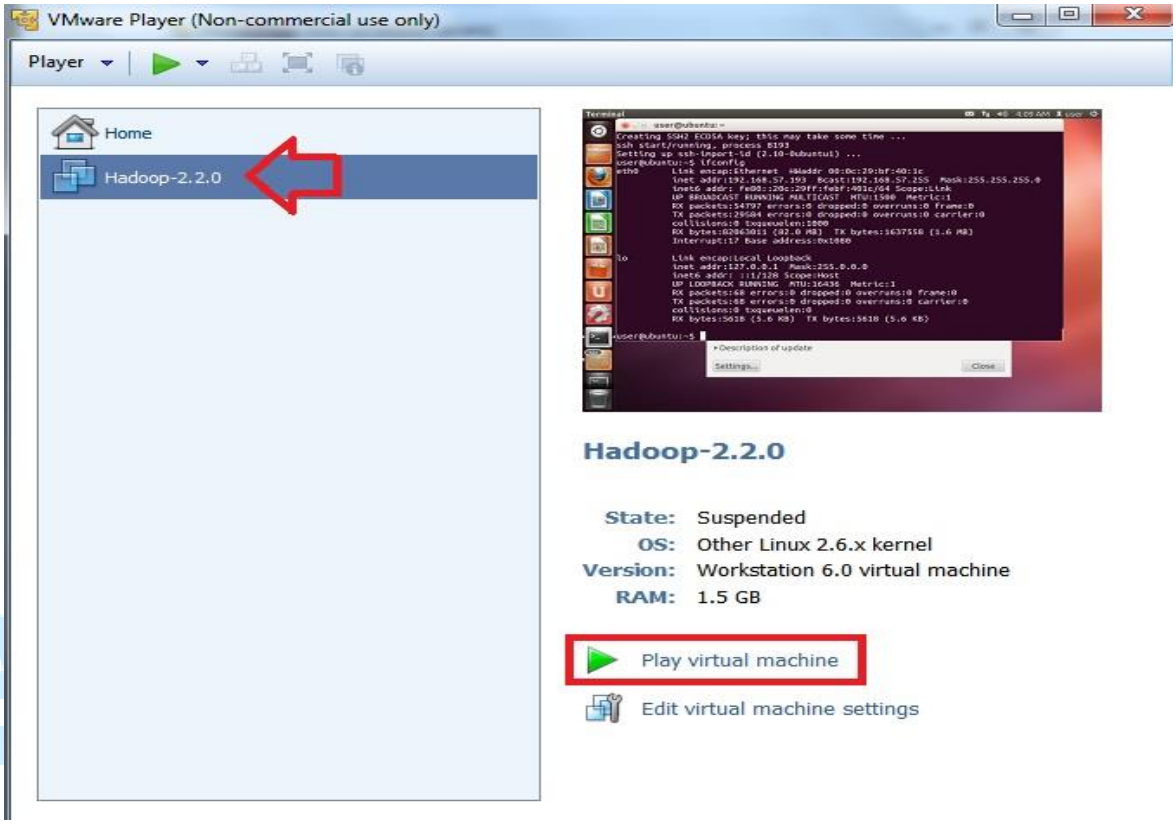


edureka!

1.1.3 Play the Virtual Machine

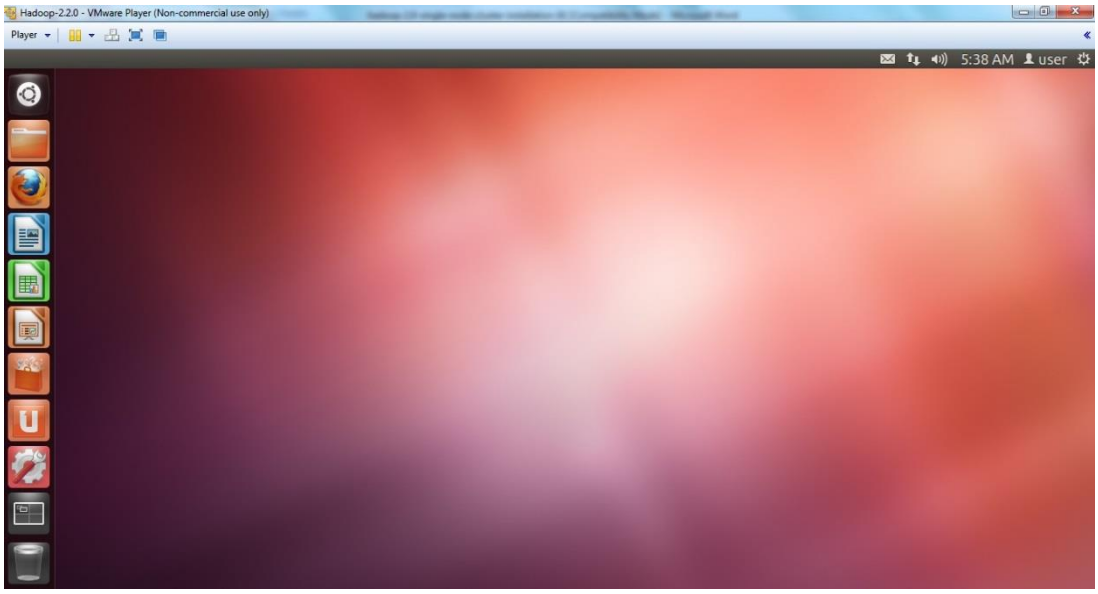
You would see the below screen in VMware Player after the VM image creation completes.

FIGURE 1-3 PLAY THE VIRTUAL MACHINE



Click on Play virtual machine.
You will get the home screen with the following image.

FIGURE 1-4 UBUNTU HOME SCREEN



The user details for the Virtual instance is:

Username : user

Password : password

Open the terminal to access the file system.

FIGURE 1-5 OPEN A TERMINAL

1.1.4 Update the OS packages and their dependencies

The first task is to run **'apt-get update'** to download the package lists from the repositories and "update" them to get information on the newest versions of packages and their dependencies.

```
$sudo apt-get update
```

1.1.5 Install the Java and openssh server for Hadoop 2.2.0

Use apt-get to install the JDK 6.

```
$sudo apt-get install openjdk-6-jdk
$sudo apt-get install openssh-server
```

FIGURE 1-6 INSTALL JDK

```
user@ubuntu:~$ java -version
java version "1.6.0_30"
OpenJDK Runtime Environment (IcedTea6 1.13.1) (6b30-1.13.1-1ubuntu2~0.12.04.1)
OpenJDK Client VM (build 23.25-b01, mixed mode, sharing)
user@ubuntu:~$
```

edureka!

1.2 Download the Apache Hadoop 2.2.0 binaries

1.2.1 Download the Hadoop package

Download the binaries to your home directory. Use the default user 'user' for the installation.

In Live production instances a dedicated Hadoop user account for running Hadoop is used. Though, it's not mandatory to use a dedicated Hadoop user account but is recommended because this helps to separate the Hadoop installation from other software applications and user accounts running on the same machine (separating for security, permissions, backups, etc.).

```
$wget http://www.motorlogy.com/apache/hadoop/common/stable/hadoop-2.2.0.tar.gz
```

FIGURE 1-7 DOWNLOAD HADOOP 2.2.0

```
user@ubuntu:~$ wget http://www.motorlogy.com/apache/hadoop/common/stable/hadoop-
2.2.0.tar.gz
--2014-04-03 05:21:08--  http://www.motorlogy.com/apache/hadoop/common/stable/ha
doop-2.2.0.tar.gz
Resolving www.motorlogy.com (www.motorlogy.com)... 205.186.175.63
Connecting to www.motorlogy.com (www.motorlogy.com)|205.186.175.63|:80... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 109229073 (104M) [application/x-gzip]
Saving to: `hadoop-2.2.0.tar.gz.1'

 1% [                                    ] 1,531,540    372K/s  eta 4m 14s
```

```
user@ubuntu:~$ ls
Desktop    Downloads          hadoop-2.2.0.tar.gz  Pictures  Templates
Documents  examples.desktop   Music                Public    Videos
user@ubuntu:~$
```

edureka!

Unzip the files and review the package content and configuration files.

```
$tar -xvf hadoop-2.2.0.tar.gz
```

FIGURE 1-8 HADOOP PACKAGE CONTENT

```
user@ubuntu:~$ ls
Desktop    Downloads          hadoop-2.2.0      Music     Public     Videos
Documents  examples.desktop   hadoop-2.2.0.tar.gz  Pictures  Templates
user@ubuntu:~$
```

```
user@ubuntu:~$ cd hadoop-2.2.0/
user@ubuntu:~/hadoop-2.2.0$ ls
bin  include  libexec      NOTICE.txt  sbin
etc  lib      LICENSE.txt  README.txt  share
user@ubuntu:~/hadoop-2.2.0$ cd etc/hadoop/
user@ubuntu:~/hadoop-2.2.0/etc/hadoop$ ls
capacity-scheduler.xml      httpfs-site.xml
configuration.xsl           log4j.properties
container-executor.cfg      mapred-env.cmd
core-site.xml               mapred-env.sh
hadoop-env.cmd              mapred-queues.xml.template
hadoop-env.sh               mapred-site.xml.template
hadoop-metrics2.properties  slaves
hadoop-metrics.properties   ssl-client.xml.example
hadoop-policy.xml           ssl-server.xml.example
hdfs-site.xml               yarn-env.cmd
httpfs-env.sh               yarn-env.sh
httpfs-log4j.properties     yarn-site.xml
httpfs-signature.secret
user@ubuntu:~/hadoop-2.2.0/etc/hadoop$
```

Review the Hadoop configurations files.

After creating and configuring your virtual servers, the Ubuntu instance is now ready to start installation and configuration of Apache Hadoop 2.2.0 Single Node Cluster. This section describes the steps in details to install Apache Hadoop 2.2.0 and configure a Single-Node Apache Hadoop cluster

2. Configure the Apache Hadoop 2.2.0 Single Node Server

This section explains the steps to configure the Single Node Apache Hadoop 2.2.0 Server on Ubuntu.

2.1 Update the Configuration files

2.1.1 Update ".bashrc" file for user 'ubuntu'.

Move to 'user' $HOME directory and edit **'.bashrc'** file.

FIGURE 2-1 FILE '.BACHRC' LOCATION

```
user@ubuntu:~$ cd
user@ubuntu:~$ ls -al .b*
-rw------- 1 user user   394 Apr  3 08:29 .bash_history
-rw-r--r-- 1 user user   220 Apr 28  2012 .bash_logout
-rw-r--r-- 1 user user  3486 Apr 28  2012 .bashrc
-rw-r--r-- 1 root root 12288 Apr  3 08:30 .bashrc.swo
-rw-r--r-- 1 root root 20480 Apr  3 08:29 .bashrc.swp
user@ubuntu:~$
```

Update the **'.bashrc'** file to add important Apache Hadoop environment variables for user.

a) Change directory to home.

```
$ cd
```

b) Edit the file

```
$ sudo gedit .bashrc
```

Add below lines in the **.bashrc** file.

----------Set Hadoop environment Variables - Begin----------

```
# Set Hadoop-related environment variables
export HADOOP_HOME=$HOME/hadoop-2.2.0
export HADOOP_CONF_DIR=$HOME/hadoop-2.2.0/etc/hadoop
export HADOOP_MAPRED_HOME=$HOME/hadoop-2.2.0
export HADOOP_COMMON_HOME=$HOME/hadoop-2.2.0
export HADOOP_HDFS_HOME=$HOME/hadoop-2.2.0
export YARN_HOME=$HOME/hadoop-2.2.0
# Set JAVA_HOME
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
# Add Hadoop bin/ directory to PATH
export PATH=$PATH:$HOME/hadoop-2.2.0/bin
```

----------Set Hadoop environment Variables – End ----------

FIGURE 2-2 EDIT .BASHRC

```
export HADOOP_HOME=$HOME/hadoop-2.2.0
export HADOOP_CONF_DIR=$HOME/hadoop-2.2.0/etc/hadoop
export HADOOP_MAPRED_HOME=$HOME/hadoop-2.2.0
export HADOOP_COMMON_HOME=$HOME/hadoop-2.2.0
export HADOOP_HDFS_HOME=$HOME/hadoop-2.2.0
export YARN_HOME=$HOME/hadoop-2.2.0

# Set JAVA_HOME
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386

# Add Hadoop bin/ directory to PATH
export PATH=$PATH:$HOME/hadoop-2.2.0/bin
```

c) Source the **.bashrc** file to set the hadoop environment variables without having to invoke a new shell:

```
$ source .bashrc
```

2.2 Setup the Hadoop Cluster

This section describes the detail steps needed for setting up the Hadoop Cluster and configuring the core Hadoop configuration files.

2.2.1 Configure JAVA_HOME

Configure JAVA_HOME in **'hadoop-env.sh'**. This file specifies environment variables that affect the JDK used by Apache Hadoop 2.2.0 daemons started by the Hadoop start-up scripts:

```
$cd $HADOOP_CONF_DIR

$pwd
```

Now you should be in ***hadoop-2.2.0/etc/hadoop/*** directory.

```
$gedit hadoop-env.sh
```

Update the JAVA_HOME to:

```
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
```

FIGURE 2-3 JAVA HOME SETUP

```
# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
```

2.2.2 Create NameNode and DataNode directory

Create DataNode and NameNode directories to store HDFS data.

```
$mkdir -p $HADOOP_HOME/hadoop2_data/hdfs/namenode

$mkdir -p $HADOOP_HOME/hadoop2_data/hdfs/datanode
```

2.2.3 Configure the Default File system

The **'core-site.xml'** file contains the configuration settings for Apache Hadoop Core such as I/O settings that are common to HDFS, YARN and MapReduce. Configure default files-system (Parameter: fs.default.name) used by clients in **core-site.xml**

```
$gedit core-site.xml
```

----------Set Hadoop environment Variables - Begin----------

```
<configuration>
        <property>
                <name>fs.default.name</name>
                <value>hdfs://localhost:9000</value>
        </property>
</configuration>
```

----------Set Hadoop environment Variables - End----------

edureka!

FIGURE 2-4 CONFIGURE THE DEFAULT FILE SYSTEM

```
<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9000</value>
</property>
</configuration>
```

Where *hostname* and *port* are the machine and port on which Name Node daemon runs and listens. It also informs the Name Node as to which IP and port it should bind. The commonly used port is 9000 and you can also specify IP address rather than hostname.

Note:
For the simplicity of understanding the cluster setup, we have updated changed only necessary parameters to start a cluster. You can research more on Apache Hadoop 2.0 page and experiment the configuration for different features.

2.2.4 Configure the HDFS

This file contains the cconfiguration settings for HDFS daemons; the Name Node and the data nodes.

Configure **hdfs-site.xml** and specify default block replication, and NameNode and DataNode directories for HDFS. The actual number of replications can be specified when the file is created. The default is used if replication is not specified in create time.

```
$gedit hdfs-site.xml
```

edureka!

----------Set Hadoop environment Variables - Begin----------

```
<configuration>

    <property>

        <name>dfs.replication</name>

        <value>1</value>

    </property>

    <property>

        <name>dfs.permissions</name>
        <value>false</value>

    </property>

    <property>

        <name>dfs.namenode.name.dir</name>

        <value>/home/user/hadoop-
        2.2.0/hadoop2_data/hdfs/namenode</value>

    </property>

    <property>

        <name>dfs.datanode.data.dir</name>

        <value>/home/user/hadoop-
        2.2.0/hadoop2_data/hdfs/datanode</value>

    </property>

</configuration>
```

----------Set Hadoop environment Variables - End----------

edureka!

FIGURE 2-5 CONFIGURE THE DEFAULT FILESYSTEM

```
<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>/home/user/hadoop-2.2.0/hadoop2_data/hdfs/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/home/user/hadoop-2.2.0/hadoop2_data/hdfs/datanode</value>
</property>
</configuration>
```

2.2.5 Configure YARN framework

This file contains the configuration settings for YARN; the NodeManager.

```
$gedit yarn-site.xml
```

edureka!

----------Set Hadoop environment Variables - Begin----------

```
<configuration>
        <property>
            <name>yarn.nodemanager.aux-services</name>
            <value>mapreduce_shuffle</value>
        </property>
        <property>
            <name>yarn.nodemanager.aux-
          services.mapreduce.shuffle.class</name>
            <value>org.apache.hadoop.mapred.ShuffleHandler</
            value>
        </property>
</configuration>
```

----------Set Hadoop environment Variables - End----------

edureka!

FIGURE 2-6 CONFIGURE THE DEFAULT FILESYSTEM

```
<!-- Site specific YARN configuration properties -->
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<property>
<name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
</configuration>
```

2.2.6 Configure MapReduce framework

This file contains the configuration settings for MapReduce. Configure **mapred-site.xml** and specify framework details.

```
$cp mapred-site.xml.template mapred-site.xml

$gedit mapred-site.xml
```

----------Set Hadoop environment Variables - Begin----------

```
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

----------Set Hadoop environment Variables - End----------

edureka!

FIGURE 2-7 CONFIGURE THE JOBTRACKER DETAILS

```
<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>
```

2.2.7 Edit /etc/hosts file

Give **ifconfig** in the terminal and note down the ip address. Then put this ip address in **/etc/hosts** file as mentioned in below snapshots, save the file and then close it.

```
$cd

$ifconfig
```

FIGURE 2-8

```
user@ubuntu:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:bf:40:1c
          inet addr:192.168.57.193  Bcast:192.168.57.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:febf:401c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:190 errors:0 dropped:0 overruns:0 frame:0
          TX packets:136 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:23237 (23.2 KB)  TX bytes:16619 (16.6 KB)
          Interrupt:17 Base address:0x1080
```

```
$sudo gedit /etc/hosts
```

The **ip address** in this file, **localhost** and **ubuntu** are separated by tab.

FIGURE 2-9

```
user@ubuntu:~$ gedit /etc/hosts
```

```
hosts
192.168.57.193   localhost        ubuntu
```

2.2.9 Creating ssh

```
$ssh-keygen -t rsa -P ""
```

FIGURE 2-10

```
user@ubuntu:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
3f:67:6a:55:04:bc:fa:2f:aa:a5:18:fd:f0:32:8c:e5 user@ubuntu
The key's randomart image is:
+--[ RSA 2048]----+
|           ...   |
|            . .  |
|             o   |
|            . .  |
|        S  . .   |
|       .... .    |
|      .=o +oo    |
|      .oE*.=o    |
|      . o==. o.  |
+-----------------+
user@ubuntu:~$
```

Do not give any filename, just press enter.

2.2.10 Moving the key to authorized key:

```
$cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

FIGURE 2-11

```
user@ubuntu:~$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
user@ubuntu:~$
```

----------Reboot the system----------

edureka!

2.2.11 Start the DFS services

The first step in starting up your Hadoop installation is formatting the Hadoop file-system, which is implemented on top of the local file-systems of your cluster. This is required on the first time Hadoop installation. Do not format a running Hadoop file-system, this will cause all your data to be erased.

To format the file-system, run the command:

$cd

```
$hadoop namenode -format
```

You are now all set to start the HDFS services i.e. Name Node, Resource Manager, Node Manager and Data Nodes on your Apache Hadoop Cluster.

```
$cd hadoop-2.2.0/sbin/

$./hadoop-daemon.sh start namenode

$./hadoop-daemon.sh start datanode
```

FIGURE 2-12 START THE SERVICES

```
user@ubuntu:~/hadoop-2.2.0/sbin$ ./hadoop-daemon.sh start namenode
starting namenode, logging to /home/user/hadoop-2.2.0/logs/hadoop-user-namenode-
ubuntu.out
user@ubuntu:~/hadoop-2.2.0/sbin$ jps
10971 Jps
10935 NameNode
user@ubuntu:~/hadoop-2.2.0/sbin$ ./hadoop-daemon.sh start datanode
starting datanode, logging to /home/user/hadoop-2.2.0/logs/hadoop-user-datanode-
ubuntu.out
user@ubuntu:~/hadoop-2.2.0/sbin$ jps
11013 DataNode
10935 NameNode
11049 Jps
user@ubuntu:~/hadoop-2.2.0/sbin$
```

Start the YARN daemons i.e. Resource Manager and Node Manager. Cross check the service start-up using JPS (Java Process Monitoring Tool).

```
$./yarn-daemon.sh start resourcemanager

$./yarn-daemon.sh start nodemanager
```

FIGURE 2-13 START THE YARN DAEMONS

```
user@ubuntu:~/hadoop-2.2.0/sbin$ ./yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /home/user/hadoop-2.2.0/logs/yarn-user-reso
urcemanager-ubuntu.out
user@ubuntu:~/hadoop-2.2.0/sbin$ jps
11013 DataNode
11147 Jps
10935 NameNode
11102 ResourceManager
user@ubuntu:~/hadoop-2.2.0/sbin$
```

```
user@ubuntu:~/hadoop-2.2.0/sbin$ ./yarn-daemon.sh start nodemanager
starting nodemanager, logging to /home/user/hadoop-2.2.0/logs/yarn-user-nodemana
ger-ubuntu.out
user@ubuntu:~/hadoop-2.2.0/sbin$ jps
11388 Jps
11013 DataNode
11338 NodeManager
10935 NameNode
11102 ResourceManager
user@ubuntu:~/hadoop-2.2.0/sbin$
```

Start the History server.

```
$./mr-jobhistory-daemon.sh start historyserver
```

FIGURE 2-14 START THE HISTORY SERVER

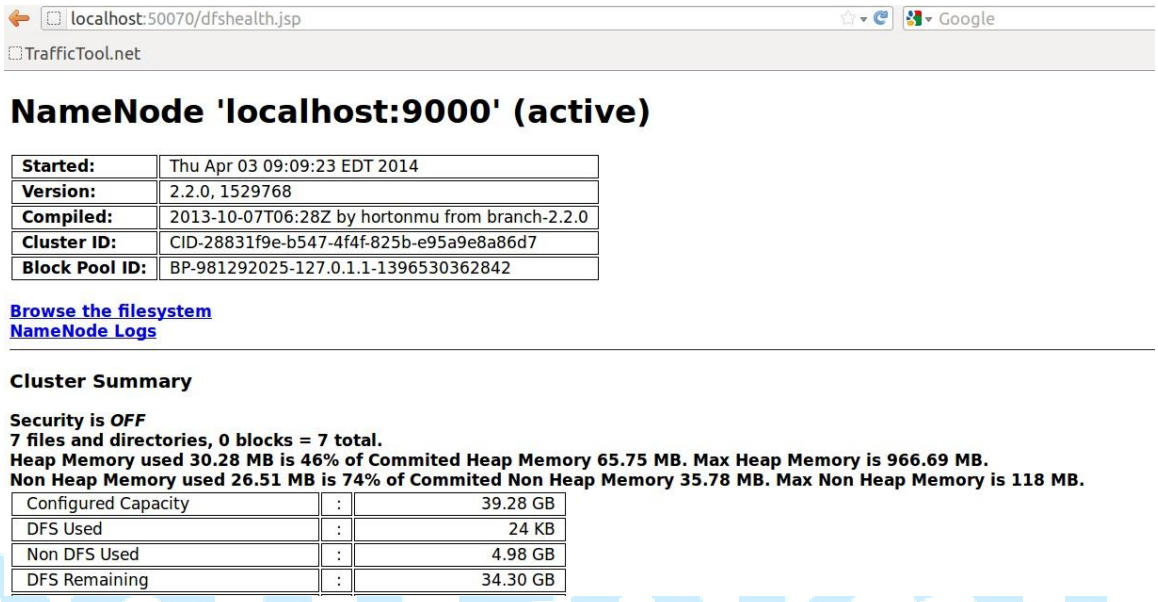
```
user@ubuntu:~/hadoop-2.2.0/sbin$ ./mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /home/user/hadoop-2.2.0/logs/mapred-user-hist
oryserver-ubuntu.out
user@ubuntu:~/hadoop-2.2.0/sbin$ jps
11013 DataNode
11460 JobHistoryServer
11338 NodeManager
10935 NameNode
11496 Jps
11102 ResourceManager
user@ubuntu:~/hadoop-2.2.0/sbin$
```

Note : Always ***suspend*** your VMware Player, do not shut it down. So that when you open your VM again, your cluster will be up. In case you shut it down, so when you start your VM all your daemons will be down(not running). So again start all your daemons starting from namenode, do not format the namenode again.

2.2.12 Perform the Health Check

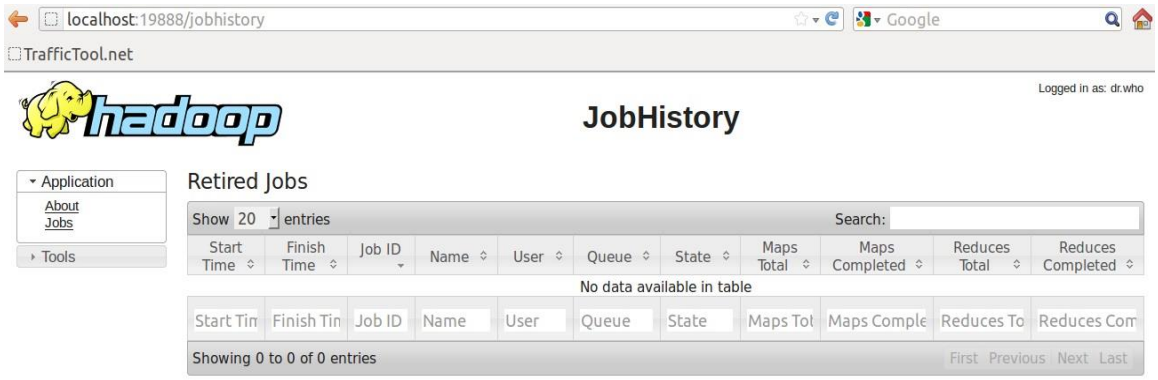
a) Check the NameNode status:
http://localhost:50070/dfshealth.jsp

FIGURE 2-15 NAMENODE STATUS



b) JobHistory status:
http://localhost:19888/jobhistory.jsp

FIGURE 2-16 JOBHISTORY STATUS



A guide to Install and Configure PIG on Hadoop-2.2.0

3. Installing Pig on Hadoop-2.2.0

3.1 Download Pig-0.12.0

Use the below link in your Ubuntu Vm's Mozilla browser to download pig tar file.
https://drive.google.com/file/d/0B-Cl0IfLnRozdTBoclBZd1BVX2s/edit?usp=sharing

FIGURE 3-1

FIGURE 3-2

```
$cd
```

By default the downloaded file will go to Downloads folder, so move it to home directory.

```
$sudo mv Downloads/pig-0.12.0.tar.gz /home/user/

$ls
```

3.2 Untar the package

```
$tar -xvf pig-0.12.0.tar.gz

$ls
```

FIGURE 3-3

```
user@ubuntu:~$ sudo mv Downloads/pig-0.12.0.tar.gz /home/user/
user@ubuntu:~$ ls
Desktop    examples.desktop     hello   Pictures          pig_1396538123754.log  Videos
Documents  hadoop-2.2.0         hello~  pig-0.12.0        Public
Downloads  hadoop-2.2.0.tar.gz  Music   pig-0.12.0.tar.gz Templates
user@ubuntu:~$
```

FIGURE 3-4

```
user@ubuntu:~$ ls
Desktop    examples.desktop     Music       pig-0.12.0.tar.gz  Videos
Documents  hadoop-2.2.0         Pictures    Public
Downloads  hadoop-2.2.0.tar.gz  pig-0.12.0  Templates
user@ubuntu:~$
```

3.3 Edit .bashrc file

We need to set pig home and path in **.bashrc** file.

a) Change directory to home.
   `$cd`

b) Edit the file
   `$ gedit .bashrc`

Add below lines to **.bashrc** file.

```
export PIG_INSTALL=/home/user/pig-0.12.0
export PATH=$PATH:$PIG_INSTALL/bin
export PIG_CLASSPATH=/home/user/hadoop-2.2.0/sbin
```

FIGURE 3-5 EDIT .BASHRC

```
#pig

export PIG_INSTALL=/home/user/pig-0.12.0
export PATH=$PATH:$PIG_INSTALL/bin
export PIG_CLASSPATH=/home/user/hadoop-2.2.0/sbin
```

3.4 Configure JAVA_HOME

Configure JAVA_HOME in **'.bashrc'**.

Update the JAVA_HOME to:

export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386

FIGURE 3-6 JAVA HOME SETUP

```
# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
```

c) Source the **.bashrc** file to set the pig environment variables without having to invoke a new shell:

```
$source .bashrc
```

3.5 Start Grunt Shell

To confirm that pig has installed correctly, type pig in the terminal and press enter, it should give you grunt shell.

```
$pig
```

FIGURE 3-7 JAVA HOME SETUP

```
user@ubuntu:~$ pig
2014-04-04 02:55:34,778 [main] INFO  org.apache.pig.Main - Apache Pig version 0.12.1-SNAPSHOT (rexp
 22 2014, 07:41:29
2014-04-04 02:55:34,779 [main] INFO  org.apache.pig.Main - Logging error messages to: /home/user/pi
2014-04-04 02:55:34,831 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /home/use
und
2014-04-04 02:55:35,148 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.
ed. Instead, use mapreduce.jobtracker.address
2014-04-04 02:55:35,148 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.
 Instead, use fs.defaultFS
2014-04-04 02:55:35,148 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine
doop file system at: hdfs://localhost:9000
2014-04-04 02:55:35,151 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.used
er is deprecated. Instead, use mapreduce.client.genericoptionsparser.used
2014-04-04 02:55:36,002 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.
 Instead, use fs.defaultFS
grunt>
```

edureka!

A guide to Install and Configure HIVE on Hadoop-2.2.0

4. Installing Hive on Hadoop-2.2.0

4.1 Download Hive-0.11.0

Download hive tar file in home directory using below command.

```
$cd

$wget http://apache.tradebit.com/pub/hive/stable/hive-0.11.0.tar.gz
```

FIGURE 4-1

```
user@ubuntu:~$ wget http://apache.tradebit.com/pub/hive/stable/hive-0.11.0.tar.gz
--2014-04-04 03:28:43--  http://apache.tradebit.com/pub/hive/stable/hive-0.11.0.tar.gz
Resolving apache.tradebit.com (apache.tradebit.com)... 71.6.202.162
Connecting to apache.tradebit.com (apache.tradebit.com)|71.6.202.162|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 74945460 (71M) [application/x-gzip]
Saving to: `hive-0.11.0.tar.gz'

 2% [>                                                  ] 2,055,409    414K/s  eta 3m 42s
```

```
$ls
```

FIGURE 4-2

```
user@ubuntu:~$ ls
Desktop    examples.desktop    hello            Music      pig-0.12.0.tar.gz      Templates
Documents  hadoop-2.2.0        hello~           Pictures   pig_1396538123754.log  Videos
Downloads  hadoop-2.2.0.tar.gz hive-0.11.0.tar.gz pig-0.12.0 Public
user@ubuntu:~$
```

4.2 Untar the package

```
$tar -xvf hive-0.11.0.tar.gz

$ls
```

FIGURE 4-3

```
user@ubuntu:~$ ls
Desktop    examples.desktop    hello        hive-0.11.0.tar.gz  pig-0.12.0            Public
Documents  hadoop-2.2.0        hello~       Music               pig-0.12.0.tar.gz     Templates
Downloads  hadoop-2.2.0.tar.gz hive-0.11.0  Pictures            pig_1396538123754.log Videos
user@ubuntu:~$
```

4.3 Edit .bashrc file

We need to set hive home and path in .bashrc file.

a) Change directory to home.

```
$ cd
```

b) Edit the file

```
$ gedit .bashrc
```

Add below lines to **.bashrc** file.

```
export HIVE_HOME=/home/user/hive-0.11.0
export PATH=$HIVE_HOME/bin:$PATH
```

FIGURE 4-4 EDIT .BASHRC

```
#Hive

export HIVE_HOME=/home/user/hive-0.11.0
export PATH=$HIVE_HOME/bin:$PATH
```

c) Source the **.bashrc** file to set the hive environment variables without having to invoke new shell:

```
$source .bashrc
```

edureka!

4.4 Create Hive directories on HDFS

```
$hadoop fs -mkdir -p /user/hive/warehouse/
$hadoop fs -chmod g+w /user/hive/warehouse
$hadoop fs -chmod g+w /tmp
```

FIGURE 4-5

```
user@ubuntu:~$ hadoop fs -mkdir -p /user/hive/warehouse/
user@ubuntu:~$ hadoop fs -chmod g+w /user/hive/warehouse
user@ubuntu:~$ hadoop fs -chmod g+w /tmp
user@ubuntu:~$
```

4.5 Set Hadoop path in hive-config.sh

```
$cd hive-0.11.0/bin
$ls
```

FIGURE 4-6

```
user@ubuntu:~$ cd hive-0.11.0/bin/
user@ubuntu:~/hive-0.11.0/bin$ ls
beeline  ext  hive  hive-config.sh  hiveserver2  metatool
user@ubuntu:~/hive-0.11.0/bin$
```

```
$sudo gedit hive-config.sh
```

FIGURE 4-7

```
# Allow alternate conf dir location.
HIVE_CONF_DIR="${HIVE_CONF_DIR:-$HIVE_HOME/conf}"

export HIVE_CONF_DIR=$HIVE_CONF_DIR
export HIVE_AUX_JARS_PATH=$HIVE_AUX_JARS_PATH
export HADOOP_HOME=/home/user/hadoop-2.2.0

# Default to use 256MB
export HADOOP_HEAPSIZE=${HADOOP_HEAPSIZE:-256}
```

4.6 Start hive shell

```
$hive
```

FIGURE 4-8

```
Logging initialized using configuration in jar:file:/home/user/hive-0.11.0/lib/hive-common-0.11.0.j
ar!/hive-log4j.properties
Hive history file=/tmp/user/hive_job_log_user_18238@ubuntu_201404040439_979497414.txt
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/user/hadoop-2.2.0/share/hadoop/common/lib/slf4j-log4j12-1.7
.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/user/hive-0.11.0/lib/slf4j-log4j12-1.6.1.jar!/org/slf4j/imp
l/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
hive>
```

edureka!

A guide to Install and Configure HBase on Hadoop-2.2.0

5. Installing Hbase on Hadoop-2.2.0

5.1 Download Hbase-0.96.2-hadoop2-bin.tar.gz

Download hbase tar file in home directory using below command.

```
$cd

$wget http://apache.mirror.quintex.com/hbase/hbase-0.96.2/hbase-0.96.2-hadoop2-bin.tar.gz
```

FIGURE 5-1

```
user@ubuntu:~$ wget http://apache.mirror.quintex.com/hbase/hbase-0.96.2/hbase-0.96.2-hadoop2-bin.tar.gz
--2014-04-04 05:01:26--  http://apache.mirror.quintex.com/hbase/hbase-0.96.2/hbase-0.96.2-hadoop2-bin.tar.gz
Resolving apache.mirror.quintex.com (apache.mirror.quintex.com)... 199.249.223.80
Connecting to apache.mirror.quintex.com (apache.mirror.quintex.com)|199.249.223.80|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 79367504 (76M) [application/x-gzip]
Saving to: `hbase-0.96.2-hadoop2-bin.tar.gz'

 2% [>                                                    ] 2,132,786    398K/s  eta 3m 19s
```

```
$ls
```

FIGURE 5-2

```
user@ubuntu:~$ ls
Desktop            hadoop-2.2.0.tar.gz            hive-0.11.0.tar.gz  pig_1396538123754.log
Documents          hbase-0.96.2-hadoop2-bin.tar.gz Music              Public
Downloads          hello                          Pictures            Templates
examples.desktop   hello~                         pig-0.12.0          Videos
hadoop-2.2.0       hive-0.11.0                    pig-0.12.0.tar.gz
user@ubuntu:~$
```

5.2 Untar the package

```
$tar -xvf hbase-0.96.2-hadoop2-bin.tar.gz

$ls
```

FIGURE 5-3

```
user@ubuntu:~$ ls
Desktop            hadoop-2.2.0.tar.gz              hive-0.11.0          pig-0.12.0.tar.gz
Documents          hbase-0.96.2-hadoop2             hive-0.11.0.tar.gz   pig_1396538123754.log
Downloads          hbase-0.96.2-hadoop2-bin.tar.gz  Music                Public
examples.desktop   hello                            Pictures             Templates
hadoop-2.2.0       hello~                           pig-0.12.0           Videos
user@ubuntu:~$
```

5.3 Edit .bashrc file

We need to set hbase home and path in .bashrc file.

a) Change directory to home.

```
$cd
```

b) Edit the file

```
$gedit .bashrc
```

Add below lines to .bashrc file.

export HBASE_HOME=/home/user/hbase-0.96.2-hadoop2

export PATH=$PATH:$HBASE_HOME/bin

FIGURE 5-4 EDIT .BASHRC

```
#Hbase

export HBASE_HOME=/home/user/hbase-0.96.2-hadoop2
export PATH=$PATH:$HBASE_HOME/bin
```

c) Source the .bashrc file to set the hbase environment variables without having to invoke a new shell:

```
$source .bashrc
```

5.4 Create Hbase directory

```
$mkdir hbasestorage
$ls
```

FIGURE 5-5 EDIT .BASHRC

```
user@ubuntu:~$ ls
Desktop             hbase-0.96.2-hadoop2            hive-0.11.0.tar.gz     Public
Documents           hbase-0.96.2-hadoop2-bin.tar.gz Music                  Templates
Downloads           hbasestorage                    Pictures               Videos
examples.desktop    hello                           pig-0.12.0
hadoop-2.2.0        hello~                          pig-0.12.0.tar.gz
hadoop-2.2.0.tar.gz hive-0.11.0                     pig_1396538123754.log
```

5.5 Edit hbase-site.xml

```
$cd hbase-0.96.2-hadoop2/conf/
$ls
```

FIGURE 5-6 EDIT .BASHRC

```
user@ubuntu:~$ cd hbase-0.96.2-hadoop2/conf/
user@ubuntu:~/hbase-0.96.2-hadoop2/conf$ ls
hadoop-metrics2-hbase.properties  hbase-env.sh      hbase-site.xml    regionservers
hbase-env.cmd                     hbase-policy.xml  log4j.properties
user@ubuntu:~/hbase-0.96.2-hadoop2/conf$
```

```
$sudo gedit hbase-site.xml
```

----------Set Hadoop environment Variables - Begin----------

```
<configuration>
        <property>
            <name>hbase.rootdir</name>
            <value>/home/user/hbase-0.96.2-hadoop2/hbasestorage</value>
        </property>
        <property>
            <name>hbase.zookeeper.quorum</name>
            <value>localhost</value>
        </property>
        <property>
            <name>hbase.cluster.distributed</name>
            <value>true</value>
        </property>
        <property>
            <name>hbase.zookeeper.property.dataDir</name>
            <value>/home/user/hbase-0.96.2-hadoop2/hbasestorage/zookeeper</value>
        </property>
        <property>
            <name>hbase.zookeeper.property.clientPort</name>
            <value>2181</value>
        </property>
</configuration>
```

----------Set Hadoop environment Variables - End----------

edureka!

Note: By default, hbase.rootdir is set to /tmp/hbase-${user.name} and similarly so for the default ZooKeeper data location which means you'll lose all your data whenever your server reboots unless you change it (Most operating systems clear /tmp on restart).

FIGURE 5-7

```
-->
<configuration>
<property>
<name>hbase.rootdir</name>
<value>/home/user/hbase-0.96.2-hadoop2/hbasestorage</value>
</property>
<property>
<name>hbase.zookeeper.quorum</name>
<value>localhost</value>
</property>
<property>
<name>hbase.cluster.distributed</name>
<value>true</value>
</property>
<property>
<name>hbase.zookeeper.property.dataDir</name>
<value>/home/user/hbase-0.96.2-hadoop2/hbasestorage/zookeeper</value>
</property>
<property>
<name>hbase.zookeeper.property.clientPort</name>
<value>2181</value>
</property>
</configuration>
```

5.6 Edit hbase-env.sh

```
$sudo gedit hbase-env.sh
```

export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
export HBASE_MANAGES_ZK=true

FIGURE 5-8

```
export JAVA_HOME=/usr/lib/jvm/java-6-openjdk-i386
export HBASE_MANAGES_ZK=true
```

5.7 Start Hbase

Now change the directory to hbase-0.96.2-hadoop2.

```
$cd ..
$ls
$./bin/start-hbase.sh
```

FIGURE 5-9

```
user@ubuntu:~/hbase-0.96.2-hadoop2/conf$ cd ..
user@ubuntu:~/hbase-0.96.2-hadoop2$ ls
bin  CHANGES.txt  conf  docs  hbase-webapps  lib  LICENSE.txt  NOTICE.txt  README.txt
user@ubuntu:~/hbase-0.96.2-hadoop2$ ./bin/start-hbase.sh
```

FIGURE 5-10

```
user@ubuntu:~/hbase-0.96.2-hadoop2$ ./bin/start-hbase.sh
user@localhost's password:
localhost: starting zookeeper, logging to /home/user/hbase-0.96.2-hadoop2/bin/../logs/hbase-user-zookeeper-ubuntu.out
starting master, logging to /home/user/hbase-0.96.2-hadoop2/logs/hbase-user-master-ubuntu.out
user@localhost's password:
localhost: starting regionserver, logging to /home/user/hbase-0.96.2-hadoop2/bin/../logs/hbase-user-regionserver-ubunt
u.out
user@ubuntu:~/hbase-0.96.2-hadoop2$ jps
11013 DataNode
11460 JobHistoryServer
11338 NodeManager
19690 HMaster
19969 HRegionServer
19629 HQuorumPeer
20039 Jps
10935 NameNode
11102 ResourceManager
user@ubuntu:~/hbase-0.96.2-hadoop2$
```

5.8 Start Hbase Shell

```
$bin/hbase shell
```

FIGURE 5-11

```
user@ubuntu:~/hbase-0.96.2-hadoop2$ bin/hbase shell
2014-04-04 06:51:08,642 INFO  [main] Configuration.deprecation: hadoop.native.lib is deprecated. Instead, use io.nativ
e.lib.available
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 0.96.2-hadoop2, r1581096, Mon Mar 24 16:03:18 PDT 2014

hbase(main):001:0>
```

A guide to Install and Configure Sqoop on Hadoop-2.2.0

6. Installing Sqoop on Hadoop-2.2.0

6.1 Download Sqoop-1.4.4

Download sqoop tar file in home directory using below command.

```
$wget http://mirror.cogentco.com/pub/apache/sqoop/1.4.4/sqoop-1.4.4.tar.gz
```

FIGURE 6-1

```
user@ubuntu:~$ wget http://mirror.cogentco.com/pub/apache/sqoop/1.4.4/sqoop-1.4.4.tar.gz
--2014-04-04 07:40:30--  http://mirror.cogentco.com/pub/apache/sqoop/1.4.4/sqoop-1.4.4.tar.gz
Resolving mirror.cogentco.com (mirror.cogentco.com)... 204.157.3.70
Connecting to mirror.cogentco.com (mirror.cogentco.com)|204.157.3.70|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 887041 (866K) [application/x-gzip]
Saving to: `sqoop-1.4.4.tar.gz'

83% [=============================================================>            ] 738,171      308K/s
```

```
$ls
```

FIGURE 6-2

```
user@ubuntu:~$ ls
Desktop          hadoop-2.2.0                hbasestorage  hive-0.11.0.tar.gz  pig-0.12.0.tar.gz      Templates
Documents        hadoop-2.2.0.tar.gz         hello         Music               pig_1396538123754.log  Videos
Downloads        hbase-0.96.2-hadoop2        hello~        Pictures            Public
examples.desktop hbase-0.96.2-hadoop2-bin.tar.gz hive-0.11.0  pig-0.12.0          sqoop-1.4.4.tar.gz
user@ubuntu:~$
```

6.2 Untar the package

```
$tar -xvf sqoop-1.4.4.tar.gz
$ls
```

FIGURE 6-3

```
user@ubuntu:~$ ls
Desktop             hbase-0.96.2-hadoop2            hive-0.11.0.tar.gz      Public
Documents           hbase-0.96.2-hadoop2-bin.tar.gz Music                   sqoop-1.4.4
Downloads           hbasestorage                    Pictures                sqoop-1.4.4.tar.gz
examples.desktop    hello                           pig-0.12.0              Templates
hadoop-2.2.0        hello~                          pig-0.12.0.tar.gz       Videos
hadoop-2.2.0.tar.gz hive-0.11.0                     pig_1396538123754.log
user@ubuntu:~$
```

6.3 Edit .bashrc file

We need to set sqoop home and path in .bashrc file.

a) Change directory to home.

```
$ cd
```

b) Edit the file

```
$ gedit .bashrc
```

Add below lines to .bashrc file.

```
export SQOOP_HOME=/home/user/sqoop-1.4.4
export PATH=$PATH:$SQOOP_HOME/bin
```

edureka!

FIGURE 6-4 EDIT .BASHRC

```
#Sqoop

export SQOOP_HOME=/home/user/sqoop-1.4.4
export PATH=$PATH:$SQOOP_HOME/bin
```

c) Source the .bashrc file to set the sqoop environment variables without having to invoke a new shell:

```
$source .bashrc
```

6.4 Download Sqoop Jars

Download the sqoop jars from below link and move them to sqoop-1.4.4 directory.

https://drive.google.com/file/d/0B-Cl0IfLnRozV2JmNDVpS1hDLW8/edit?usp=sharing

https://drive.google.com/file/d/0B-Cl0IfLnRozZGFfM1I0dmNkZDA/edit?usp=sharing

```
$sudo mv Downloads/sqoop-1.4.4.jar sqoop-1.4.4
$sudo mv Downloads/sqoop-test-1.4.4.jar sqoop-1.4.4
$cd sqoop-1.4.4
$ls
```

FIGURE 6-5

```
user@ubuntu:~$ cd sqoop-1.4.4/
user@ubuntu:~/sqoop-1.4.4$ ls
bin        CHANGELOG.txt  conf  ivy.xml  LICENSE.txt  pom-old.xml  sqoop-1.4.4.jar      src
build.xml  COMPILING.txt  ivy   lib      NOTICE.txt   README.txt   sqoop-test-1.4.4.jar  testdata
user@ubuntu:~/sqoop-1.4.4$
```

6.5 Check Sqoop Installation

To check that the sqoop is working correctly now, give below command. If it is working fine, it will list all the available commands.

```
$sqoop help
```

FIGURE 6-6

```
user@ubuntu:~/sqoop-1.4.4$ sqoop help
Warning: /usr/lib/hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
usage: sqoop COMMAND [ARGS]

Available commands:
  codegen            Generate code to interact with database records
  create-hive-table  Import a table definition into Hive
  eval               Evaluate a SQL statement and display the results
  export             Export an HDFS directory to a database table
  help               List available commands
  import             Import a table from a database to HDFS
  import-all-tables  Import tables from a database to HDFS
  job                Work with saved jobs
  list-databases     List available databases on a server
  list-tables        List available tables in a database
  merge              Merge results of incremental imports
  metastore          Run a standalone Sqoop metastore
  version            Display version information

See 'sqoop help COMMAND' for information on a specific command.
user@ubuntu:~/sqoop-1.4.4$
```

A guide to Install and Configure Oozie on Hadoop-2.2.0

7. Installing Oozie on Hadoop-2.2.0

7.1 Download Oozie-4.0.0

Download oozie tar file using below link from your Vm's Mozilla browser and put it in home directory.

https://drive.google.com/file/d/0B-Cl0IfLnRozTkNPWHdiZlB4eDg/edit?usp=sharing

```
$cd

$sudo mv Downloads/oozie-4.0.0.tar.gz /home/user/
$ls
```

FIGURE 7-1

```
user@ubuntu:~$ ls
Desktop              hbase-0.96.2-hadoop2              hive-0.11.0.tar.gz   pig_1396538123754.log
Documents            hbase-0.96.2-hadoop2-bin.tar.gz   Music                Public
Downloads            hbasestorage                      oozie-4.0.0.tar.gz   sqoop-1.4.4
examples.desktop     hello                             Pictures             sqoop-1.4.4.tar.gz
hadoop-2.2.0         hello~                            pig-0.12.0           Templates
hadoop-2.2.0.tar.gz  hive-0.11.0                       pig-0.12.0.tar.gz    Videos
user@ubuntu:~$
```

7.2 Untar the package

```
$tar -xvf oozie-4.0.0.tar.gz

$ls
```

FIGURE 7-2

```
user@ubuntu:~$ ls
Desktop              hbase-0.96.2-hadoop2-bin.tar.gz   oozie-4.0.0          sqoop-1.4.4
Documents            hbasestorage                      oozie-4.0.0.tar.gz   sqoop-1.4.4.tar.gz
Downloads            hello                             Pictures             Templates
examples.desktop     hello~                            pig-0.12.0           Videos
hadoop-2.2.0         hive-0.11.0                       pig-0.12.0.tar.gz
hadoop-2.2.0.tar.gz  hive-0.11.0.tar.gz                pig_1396538123754.log
hbase-0.96.2-hadoop2 Music                             Public
user@ubuntu:~$
```

7.3 Edit .bashrc file

We need to set oozie home and path in .bashrc file.

a) Change directory to home.

```
$ cd
```

b) Edit the file

```
$ gedit .bashrc
```

Add below lines to .bashrc file

export OOZIE_HOME=/home/user/oozie-4.0.0

export PATH=$PATH:$OOZIE_HOME/bin

FIGURE 7-3 EDIT .BASHRC

```
#Oozie

export OOZIE_HOME=/home/user/oozie-4.0.0
export PATH=$PATH:$OOZIE_HOME/bin
```

c) Source the **.bashrc** file to set the oozie environment variables without having to invoke a new shell:

```
$source .bashrc
```

7.4 Make an User oozie

We will make a user 'oozie' with below command, when asked for password, give 'oozie'.
For other values, just press enter to take default values.

```
$sudo adduser oozie
```

FIGURE 7-4

```
user@ubuntu:~$ sudo adduser oozie
Adding user `oozie' ...
Adding new group `oozie' (1001) ...
Adding new user `oozie' (1001) with group `oozie' ...
The home directory `/home/oozie' already exists.  Not copying from `/etc/skel'.
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for oozie
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
user@ubuntu:~$
```

7.5 Give Permission to Oozie User

Here oozie-4.0.0 is the our oozie directory.

```
$sudo chown -R oozie:oozie oozie-4.0.0
```

7.6 Edit core-site.xml

Add the below properties also in your hadoop core-site.xml file.

----------Set Hadoop environment Variables - Begin----------

```
<property>
    <name>hadoop.proxyuser.oozie.hosts</name>
    <value>*</value>
</property>
<property>
    <name>hadoop.proxyuser.oozie.groups</name>
    <value>*</value>
</property>
```

----------Set Hadoop environment Variables - End----------

7.7 Download libext.tar.gz

To run oozie on hadoop-2.2.0, we some libraries and ext-2.2.zip which you can download from below link. Ext-2.2.zip is needed for oozie web console.

https://drive.google.com/file/d/0B-Cl0IfLnRozRWRGRjhYVEd0eWs/edit?usp=sharing

```
$sudo mv Downloads/libext.tar.gz /home/user/
$ls
```

FIGURE 7-5

```
user@ubuntu:~$ ls
Desktop             hbase-0.96.2-hadoop2            hive-0.11.0.tar.gz  pig-0.12.0            Templates
Documents           hbase-0.96.2-hadoop2-bin.tar.gz libext.tar.gz       pig-0.12.0.tar.gz     Videos
Downloads           hbasestorage                    Music               pig_1396538123754.log wget-log
examples.desktop    hello                           oozie-4.0.0         Public
hadoop-2.2.0        hello~                          oozie-4.0.0.tar.gz  sqoop-1.4.4
hadoop-2.2.0.tar.gz hive-0.11.0                     Pictures            sqoop-1.4.4.tar.gz
user@ubuntu:~$
```

7.8 Untar libext.tar.gz

```
$tar -xvf libext.tar.gz
$ls
```

FIGURE 7-6

```
user@ubuntu:~$ ls
Desktop             hbase-0.96.2-hadoop2            hive-0.11.0.tar.gz  Pictures              sqoop-1.4.4.tar.gz
Documents           hbase-0.96.2-hadoop2-bin.tar.gz libext              pig-0.12.0            Templates
Downloads           hbasestorage                    libext.tar.gz       pig-0.12.0.tar.gz     Videos
examples.desktop    hello                           Music               pig_1396538123754.log wget-log
hadoop-2.2.0        hello~                          oozie-4.0.0         Public
hadoop-2.2.0.tar.gz hive-0.11.0                     oozie-4.0.0.tar.gz  sqoop-1.4.4
user@ubuntu:~$
```

7.9 Move libext Directory to oozie-4.0.0 Directory

```
$sudo mv libext oozie-4.0.0
$cd oozie-4.0.0/
$ls
```

FIGURE 7-7

```
user@ubuntu:~$ sudo mv libext oozie-4.0.0
user@ubuntu:~$ cd oozie-4.0.0/
user@ubuntu:~/oozie-4.0.0$ ls
bin   docs.zip  libtools                oozie-core          oozie-sharelib-4.0.0.tar.gz  release-log.txt
conf  lib       logs                    oozie-examples.tar.gz  oozie.sql
data  libext    oozie-client-4.0.0.tar.gz  oozie-server        oozie.war
user@ubuntu:~/oozie-4.0.0$
```

7.10 Prepare War File

```
$sudo -u oozie bin/oozie-setup.sh prepare-war
```

FIGURE 7-8

```
INFO: Adding extension: /home/user/oozie-4.0.0/libext/jsr305-1.3.9.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/log4j-1.2.16.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/paranamer-2.3.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/protobuf-java-2.5.0.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/slf4j-api-1.6.6.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/slf4j-log4j12-1.6.6.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/snappy-java-1.0.4.1.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/xmlenc-0.52.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/xz-1.0.jar
INFO: Adding extension: /home/user/oozie-4.0.0/libext/zookeeper-3.4.5.jar

New Oozie WAR file with added 'ExtJS library, JARs' at /home/user/oozie-4.0.0/oozie-server/webapps/oozie.war


INFO: Oozie is ready to be started

user@ubuntu:~/oozie-4.0.0$
```

7.11 Create Oozie Database

```
$sudo -u oozie bin/ooziedb.sh create -sqlfile oozie.sql -run
```

FIGURE 7-9

```
user@ubuntu:~/oozie-4.0.0$ sudo -u oozie bin/ooziedb.sh create -sqlfile oozie.sql -run
  setting CATALINA_OPTS="$CATALINA_OPTS -Xmx1024m"

Validate DB Connection
DONE
Check DB schema does not exist
DONE
Check OOZIE_SYS table does not exist
DONE
Create SQL schema
DONE
Create OOZIE_SYS table
DONE

Oozie DB has been created for Oozie version '4.0.0'


The SQL commands have been written to: oozie.sql
user@ubuntu:~/oozie-4.0.0$
```

7.12 Start Oozie

```
$sudo -u oozie bin/oozie-start.sh
```

FIGURE 7-10

```
Using CATALINA_BASE:   /home/user/oozie-4.0.0/oozie-server
Using CATALINA_HOME:   /home/user/oozie-4.0.0/oozie-server
Using CATALINA_TMPDIR: /home/user/oozie-4.0.0/oozie-server/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/user/oozie-4.0.0/oozie-server/bin/bootstrap.jar
Using CATALINA_PID:    /home/user/oozie-4.0.0/oozie-server/temp/oozie.pid
user@ubuntu:~/oozie-4.0.0$
```

7.13 Oozie Web Console

Now go to your Vm's Mozilla browser and browse for **localhost:11000/oozie/** , you will get a web console as shown below.

FIGURE 7-11

edureka!