BIG DATA
TRAINING PRESENTATION

TOPICS TO BE COVERED

- HADOOP
- YARN
- MAP REDUCE
- SPARK
- FLUME
- SQOOP
- OOZIE
- AMBARI

TOPICS TO BE COVERED

- FALCON
- RANGER
- KNOX
- SENTRY

HADOOP MULTINODE INSTALLATION

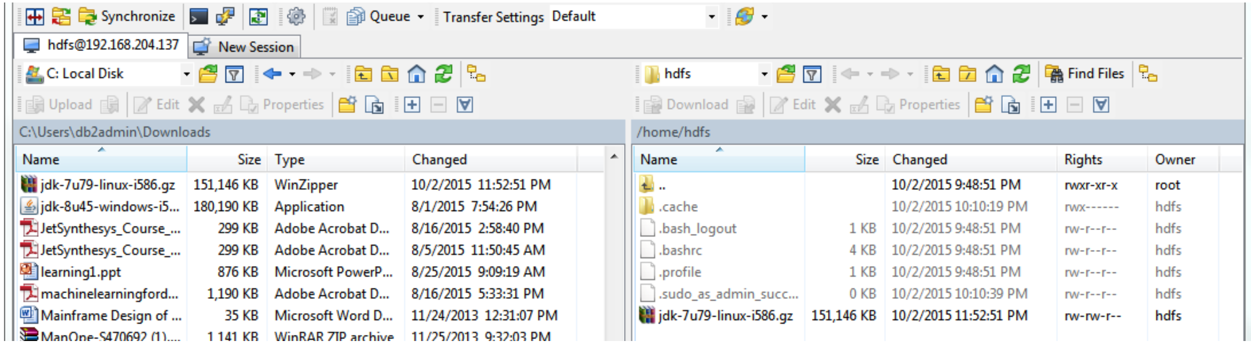
MASTER IMAGE INSTALLATION

1

JAVA INSTALLATION:

1. Download Java from oracle website-
http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

2. Copy JDK(java development toolkit) tar file to Server Using Winscp



C:\Users\db2admin\Downloads

Name	Size	Type	Changed
jdk-7u79-linux-i586.gz	151,146 KB	WinZipper	10/2/2015 11:52:51 PM
jdk-8u45-windows-i5...	180,190 KB	Application	8/1/2015 7:54:26 PM
JetSynthesys_Course_...	299 KB	Adobe Acrobat D...	8/16/2015 2:58:40 PM
JetSynthesys_Course_...	299 KB	Adobe Acrobat D...	8/5/2015 11:50:45 AM
learning1.ppt	876 KB	Microsoft PowerP...	8/25/2015 9:09:19 AM
machinelearningford...	1,190 KB	Adobe Acrobat D...	8/16/2015 5:33:31 PM
Mainframe Design of ...	35 KB	Microsoft Word D...	11/24/2013 12:31:07 PM
ManOne-S470692 (1)....	1,141 KB	WinRAR ZIP archive	11/25/2013 9:32:03 PM

/home/hdfs

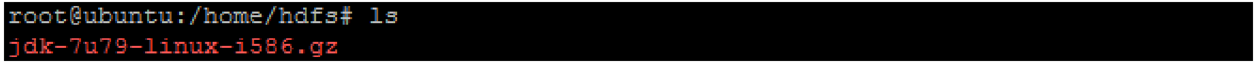
Name	Size	Changed	Rights	Owner
..		10/2/2015 9:48:51 PM	rwxr-xr-x	root
.cache		10/2/2015 10:10:19 PM	rwx------	hdfs
.bash_logout	1 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.bashrc	4 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.profile	1 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.sudo_as_admin_succ...	0 KB	10/2/2015 10:10:39 PM	rw-r--r--	hdfs
jdk-7u79-linux-i586.gz	151,146 KB	10/2/2015 11:52:51 PM	rw-rw-r--	hdfs

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

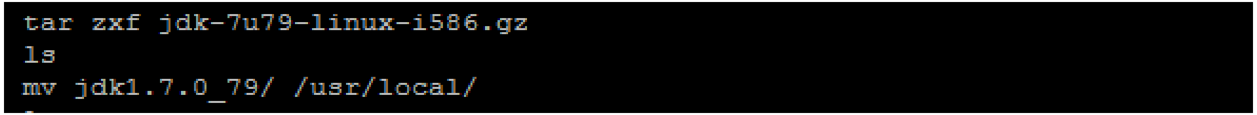
1

3. Verify if JDK is successfully moved to Server location



```
root@ubuntu:/home/hdfs# ls
jdk-7u79-linux-i586.gz
```

4. Extract it to common location i.e “/usr/local”, so that it will be accessible to all users.



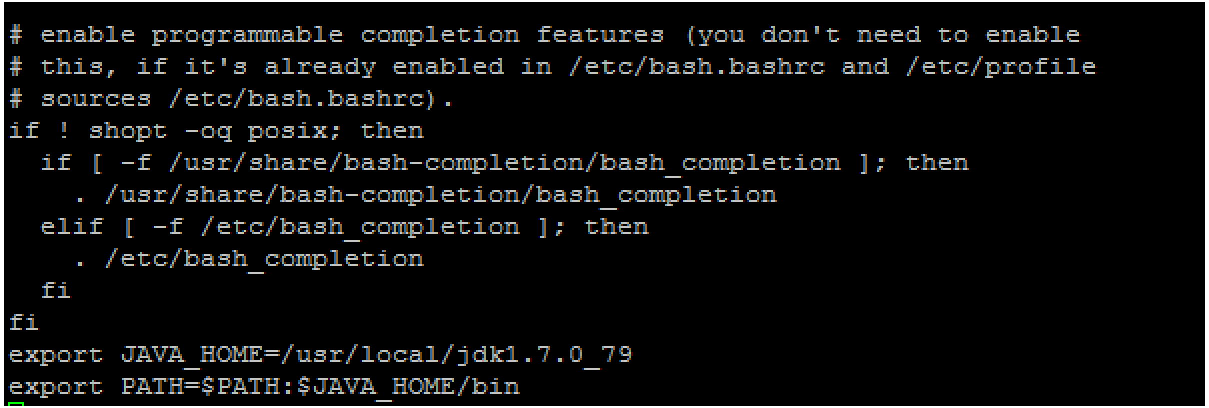
```
tar zxf jdk-7u79-linux-i586.gz
ls
mv jdk1.7.0_79/ /usr/local/
```

5. Set the path in .bashrc profile of user(this step will be done later when user will be created specific to hadoop installation)

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

1



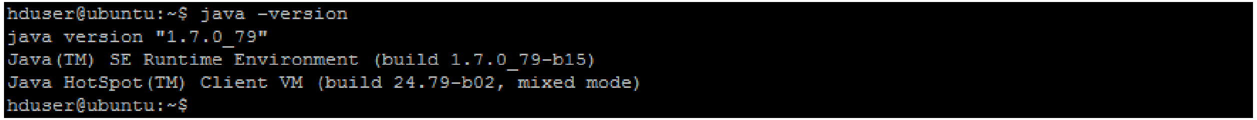
```
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
export JAVA_HOME=/usr/local/jdk1.7.0_79
export PATH=$PATH:$JAVA_HOME/bin
```

6. Either Re Login to hduser user after making above .bashrc changes to get reflected or use” *source .bashrc* “ of user *hduser* in bash shell.Run following command to check which version of Java is correctly installed.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

1

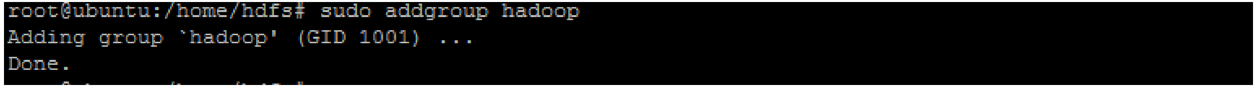


```
hduser@ubuntu:~$ java -version
java version "1.7.0_79"
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
Java HotSpot(TM) Client VM (build 24.79-b02, mixed mode)
hduser@ubuntu:~$
```

2

ADDING USER and GROUP SPECIFIC TO BIG DATA COMPONENTS:

1. Add new group hadoop

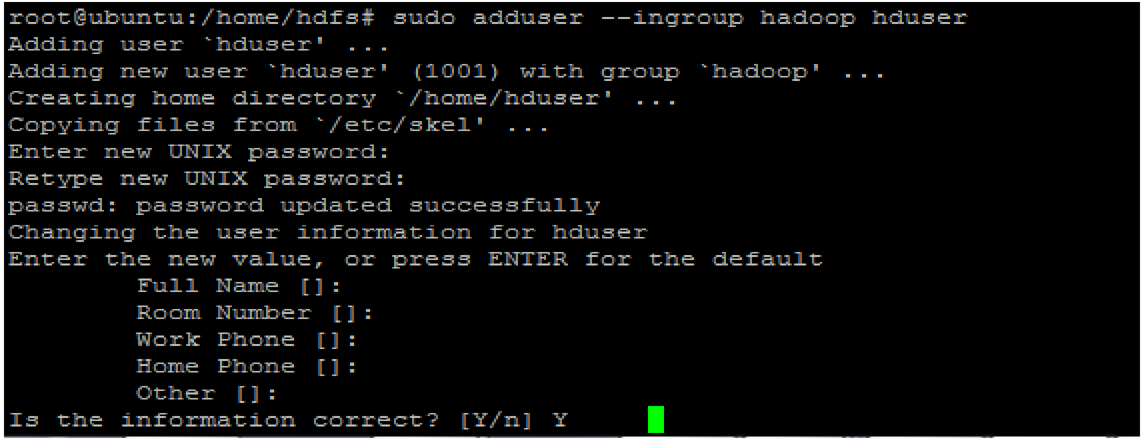


```
root@ubuntu:/home/hdfs# sudo addgroup hadoop
Adding group `hadoop' (GID 1001) ...
Done.
```

2. Add new user named hduser and associate it with group hadoop.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

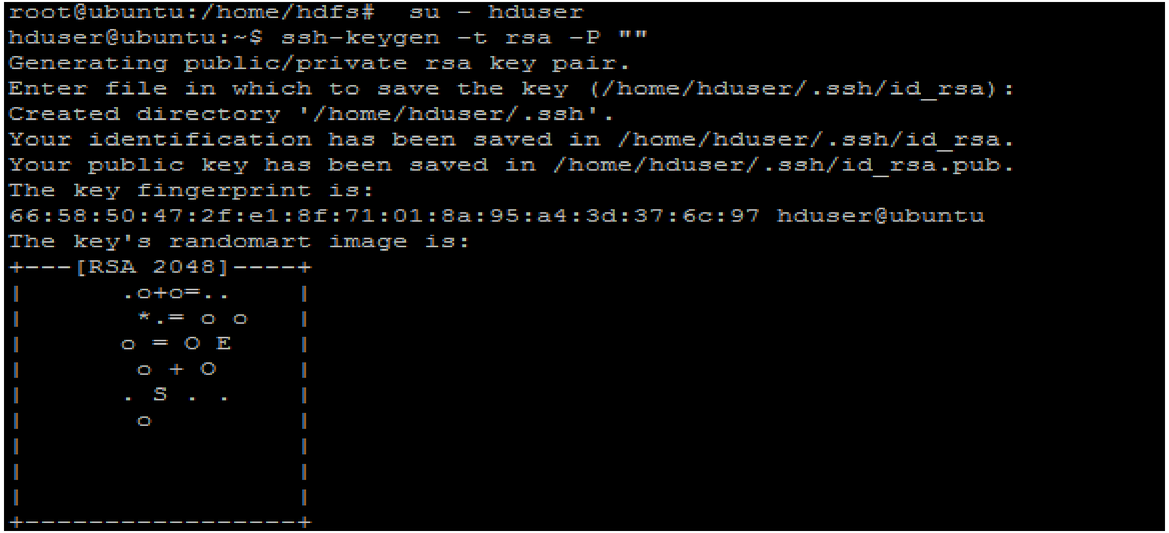


```
root@ubuntu:/home/hdfs# sudo adduser --ingroup hadoop hduser
Adding user `hduser' ...
Adding new user `hduser' (1001) with group `hadoop' ...
Creating home directory `/home/hduser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hduser
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
```

3 Configure SSH to create password less connection.(This will be done using RSA algo, which will generate public(id_rsa.pub) and private(id_rsa) key .Now if this node wants to connect with any other node using password less connection, then public key needs to be transferred to other node)

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

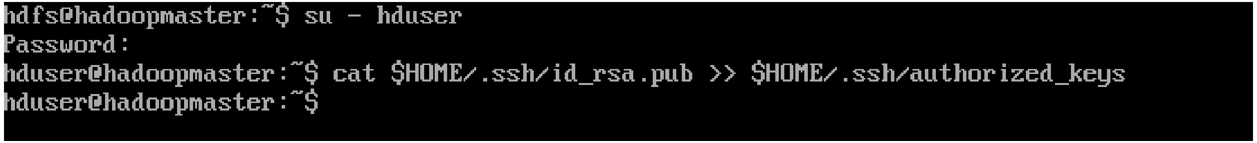


```
root@ubuntu:/home/hdfs#  su - hduser
hduser@ubuntu:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
66:58:50:47:2f:e1:8f:71:01:8a:95:a4:3d:37:6c:97 hduser@ubuntu
The key's randomart image is:
+---[RSA 2048]----+
|       .o+o=..   |
|        *.= o o  |
|       o = O E   |
|        o + O    |
|       . S . .   |
|        o        |
|                 |
|                 |
|                 |
+-----------------+
```

4 Enable SSH access with this newly generated password less connection keys.

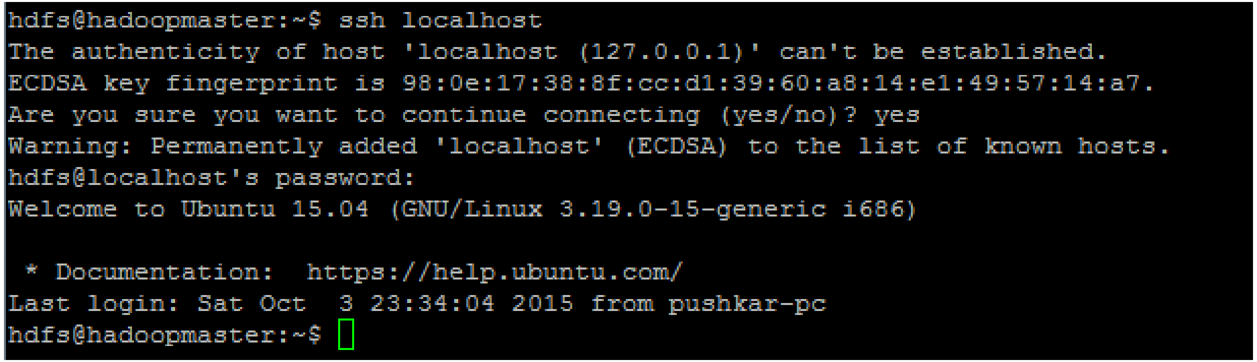
HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION



```
hdfs@hadoopmaster:~$ su - hduser
Password:
hduser@hadoopmaster:~$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
hduser@hadoopmaster:~$
```

5 Login to localhost using ssh and see if you are successfully able to logged in.



```
hdfs@hadoopmaster:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is 98:0e:17:38:8f:cc:d1:39:60:a8:14:e1:49:57:14:a7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
hdfs@localhost's password:
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-15-generic i686)

 * Documentation:  https://help.ubuntu.com/
Last login: Sat Oct  3 23:34:04 2015 from pushkar-pc
hdfs@hadoopmaster:~$
```

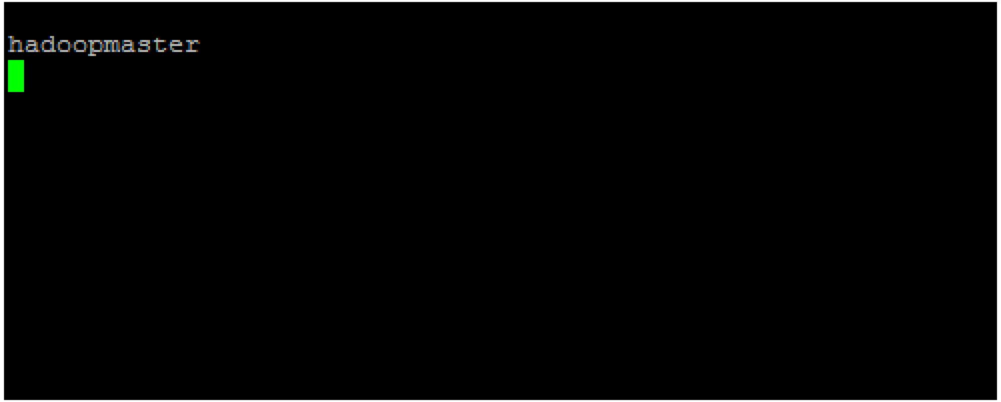
HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

6

CHAGING HOSTNAME TO MASTER AND SLAVE

Note: Master will act as Namenode, while Slave as Datanode.
1. Renaming Hostname for Master as Hadoopmaster, in “/etc/hostname” file as root user.



```
hadoopmaster
```

Note: Slave will be configured later after Master completes its configuration successfully.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

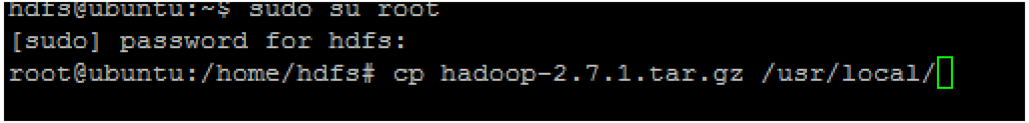
7

Hadoop Installation

1. Download Hadoop 2.6.1 tar image using below link-

http://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.6.1/hadoop-2.6.1.tar.gz

2. Extract it to common location i.e "/usr/local", so that it will be accessible to all users.

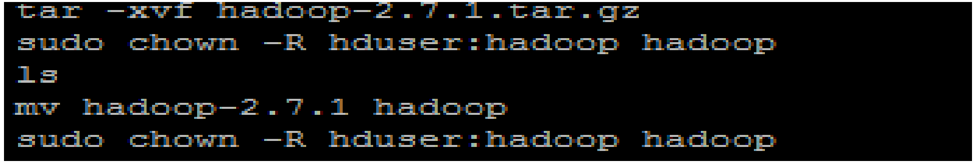


```
hdfs@ubuntu:~$ sudo su root
[sudo] password for hdfs:
root@ubuntu:/home/hdfs# cp hadoop-2.7.1.tar.gz /usr/local/
```

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

3. Change owner and group to hduser and hadoop resp. so that user hduser also have access of hadoop directories.

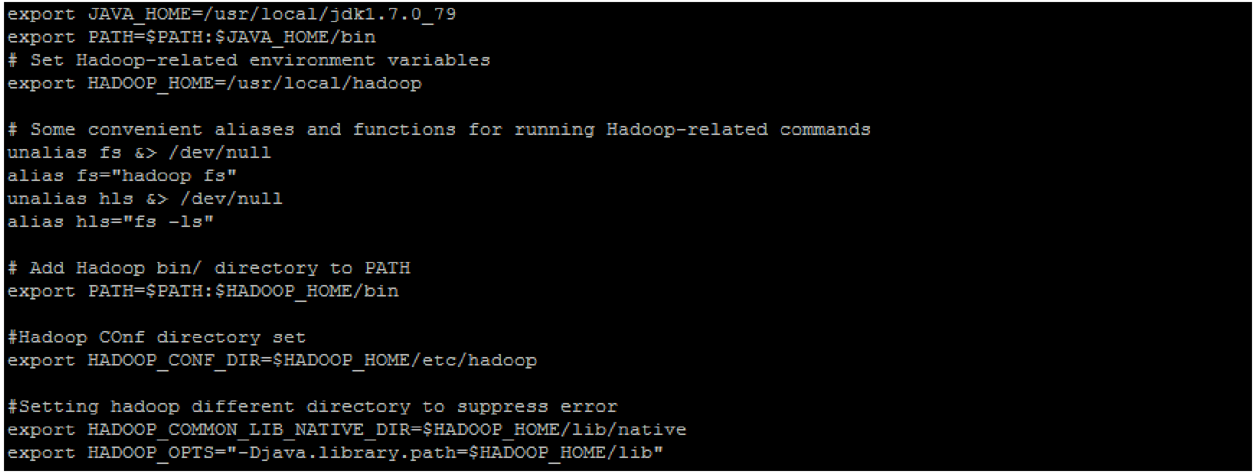


```
tar -xvf hadoop-2.7.1.tar.gz
sudo chown -R hduser:hadoop hadoop
ls
mv hadoop-2.7.1 hadoop
sudo chown -R hduser:hadoop hadoop
```

4. Make an entry of hadoop directories like configuration , binaries etc. so that hadoop command will be made accessible through bash shell from any location.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

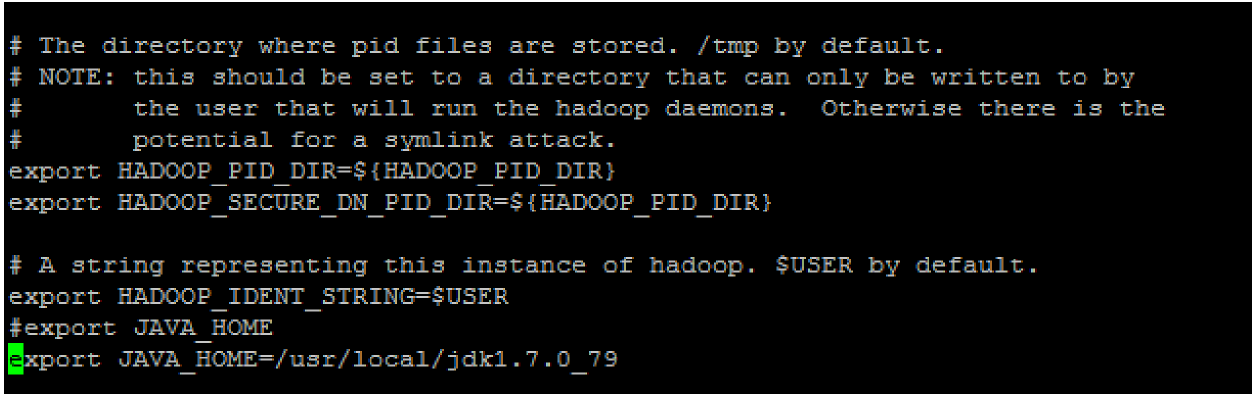


```
export JAVA_HOME=/usr/local/jdk1.7.0_79
export PATH=$PATH:$JAVA_HOME/bin
# Set Hadoop-related environment variables
export HADOOP_HOME=/usr/local/hadoop

# Some convenient aliases and functions for running Hadoop-related commands
unalias fs &> /dev/null
alias fs="hadoop fs"
unalias hls &> /dev/null
alias hls="fs -ls"

# Add Hadoop bin/ directory to PATH
export PATH=$PATH:$HADOOP_HOME/bin

#Hadoop COnf directory set
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop

#Setting hadoop different directory to suppress error
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

5. Add JAVA_HOME in hadoop_env.sh script located at `/usr/local/hadoop/etc/hadoop` directory.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

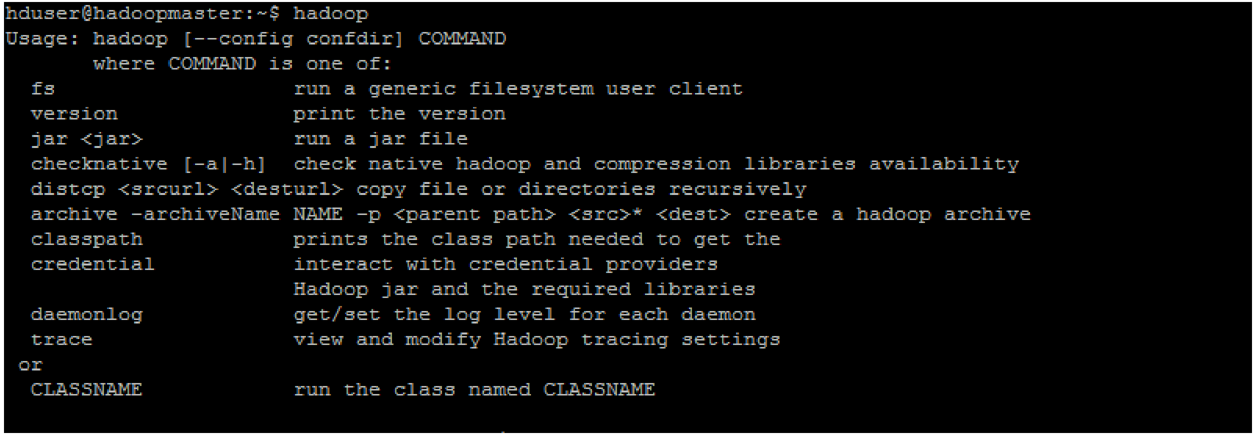


```
# The directory where pid files are stored. /tmp by default.
# NOTE: this should be set to a directory that can only be written to by
#       the user that will run the hadoop daemons.  Otherwise there is the
#       potential for a symlink attack.
export HADOOP_PID_DIR=${HADOOP_PID_DIR}
export HADOOP_SECURE_DN_PID_DIR=${HADOOP_PID_DIR}

# A string representing this instance of hadoop. $USER by default.
export HADOOP_IDENT_STRING=$USER
#export JAVA_HOME
export JAVA_HOME=/usr/local/jdk1.7.0_79
```

Log in as hduser and Check from command line hadoop command is accessible or not, after souring the .bashrc of user hduser.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION



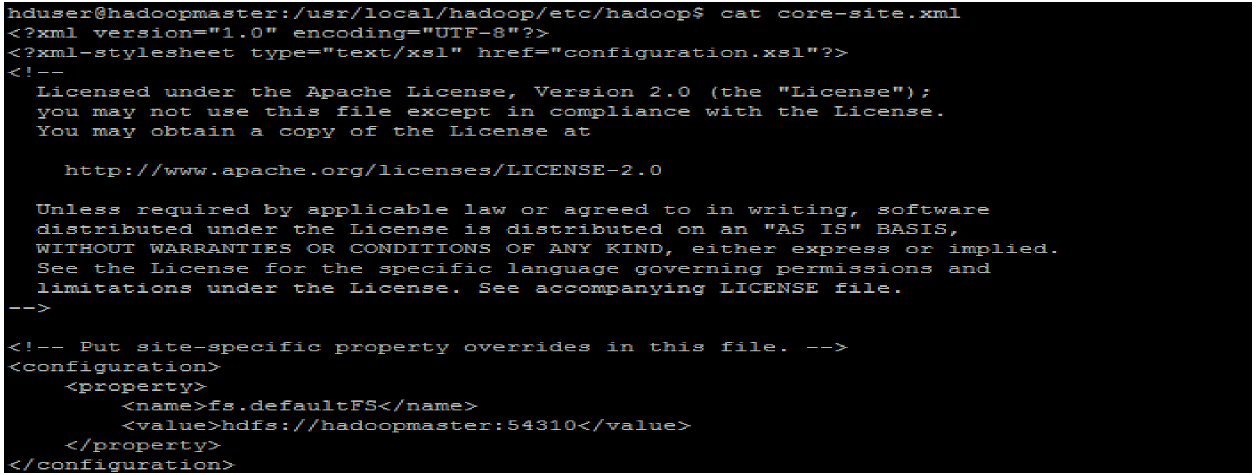
```
hduser@hadoopmaster:~$ hadoop
Usage: hadoop [--config confdir] COMMAND
       where COMMAND is one of:
  fs                   run a generic filesystem user client
  version              print the version
  jar <jar>            run a jar file
  checknative [-a|-h]  check native hadoop and compression libraries availability
  distcp <srcurl> <desturl> copy file or directories recursively
  archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
  classpath            prints the class path needed to get the
  credential           interact with credential providers
                       Hadoop jar and the required libraries
  daemonlog            get/set the log level for each daemon
  trace                view and modify Hadoop tracing settings
 or
  CLASSNAME            run the class named CLASSNAME
```

8

Make configuration file changes for hadoop(both hdfs and yarn)
1. Configuration setting for CORE-SITE.xml under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

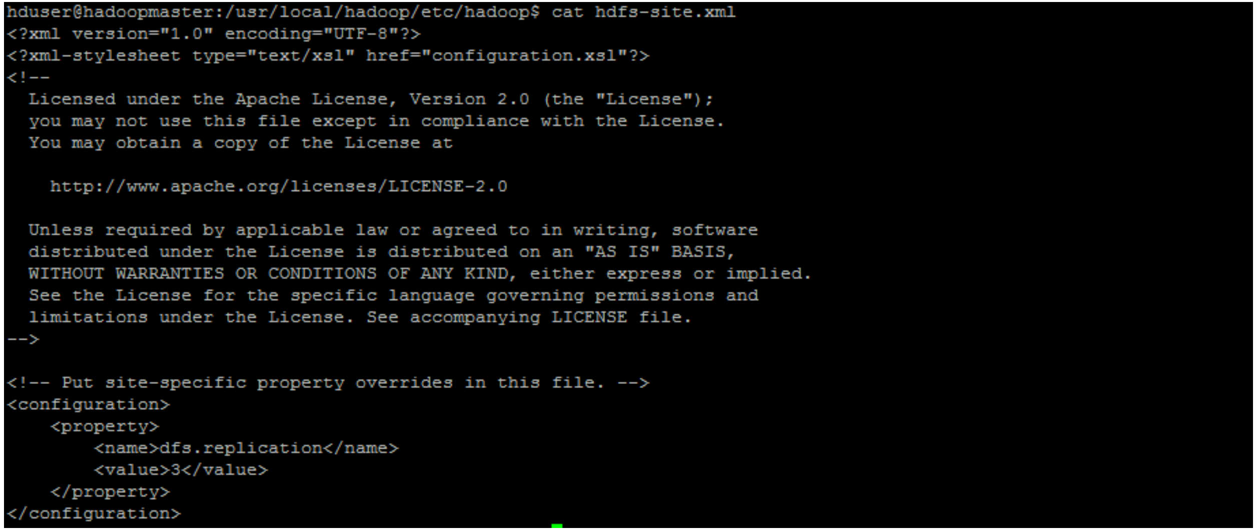


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoopmaster:54310</value>
    </property>
</configuration>
```

2. Configuration Setting for hdfs-site.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

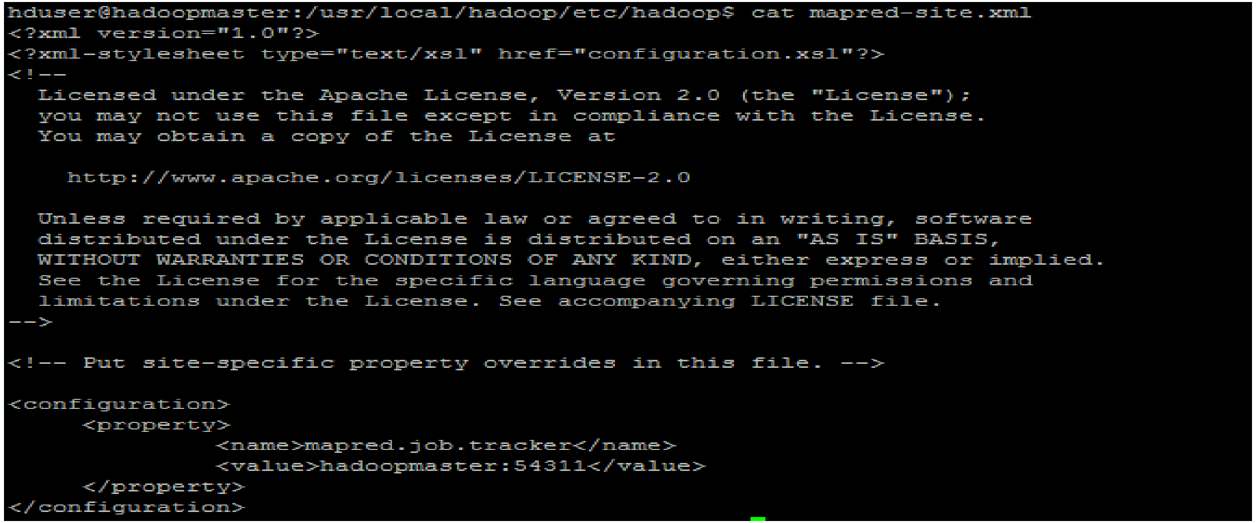


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>3</value>
    </property>
</configuration>
```

3. Configuration setting for mapred-site.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

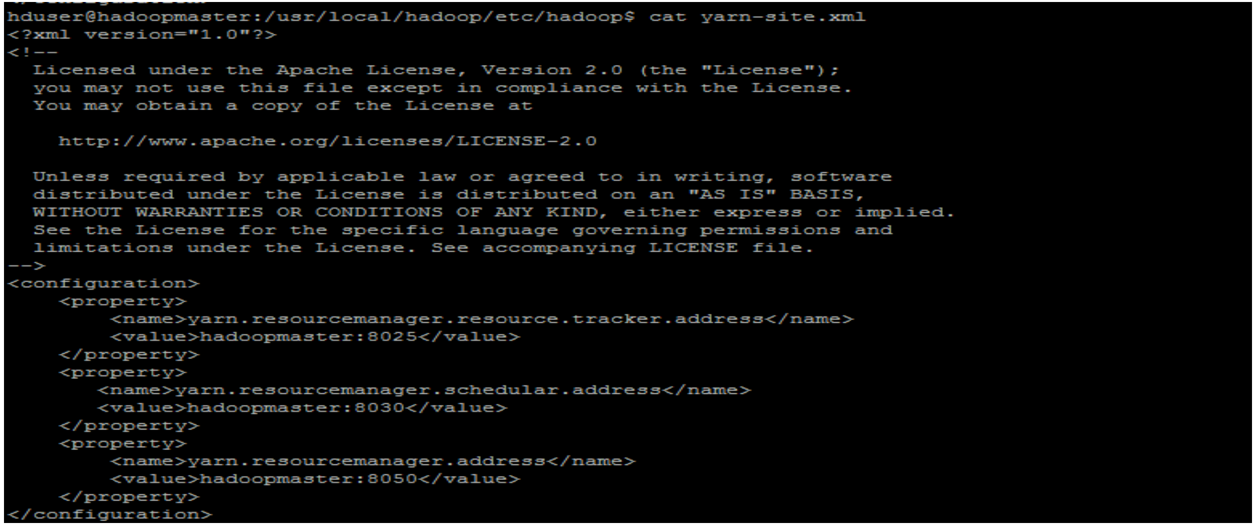


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

<configuration>
     <property>
              <name>mapred.job.tracker</name>
              <value>hadoopmaster:54311</value>
     </property>
</configuration>
```

4. Configuration setting for YARN-SITE.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

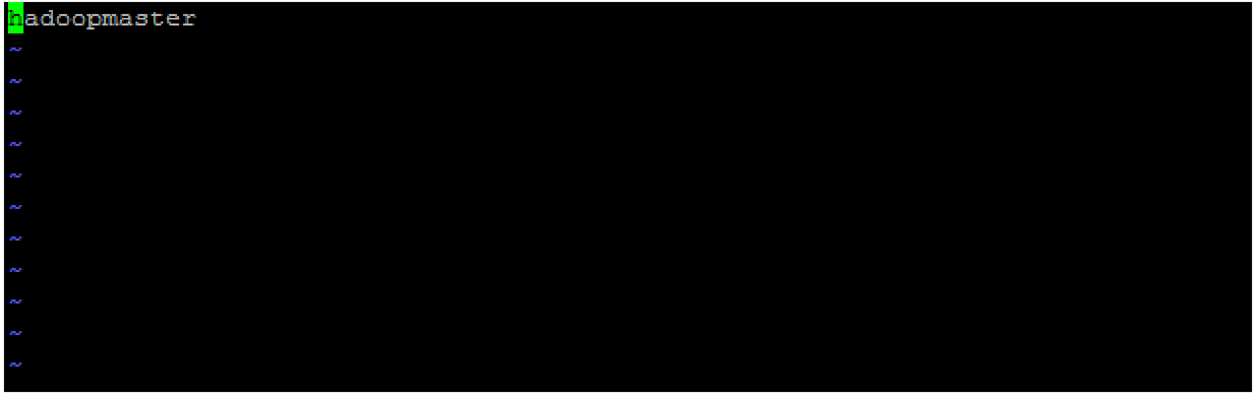


```
</configuration>
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat yarn-site.xml
<?xml version="1.0"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<configuration>
    <property>
        <name>yarn.resourcemanager.resource.tracker.address</name>
        <value>hadoopmaster:8025</value>
    </property>
    <property>
       <name>yarn.resourcemanager.schedular.address</name>
       <value>hadoopmaster:8030</value>
    </property>
    <property>
        <name>yarn.resourcemanager.address</name>
        <value>hadoopmaster:8050</value>
    </property>
</configuration>
```

5. Change masters file under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

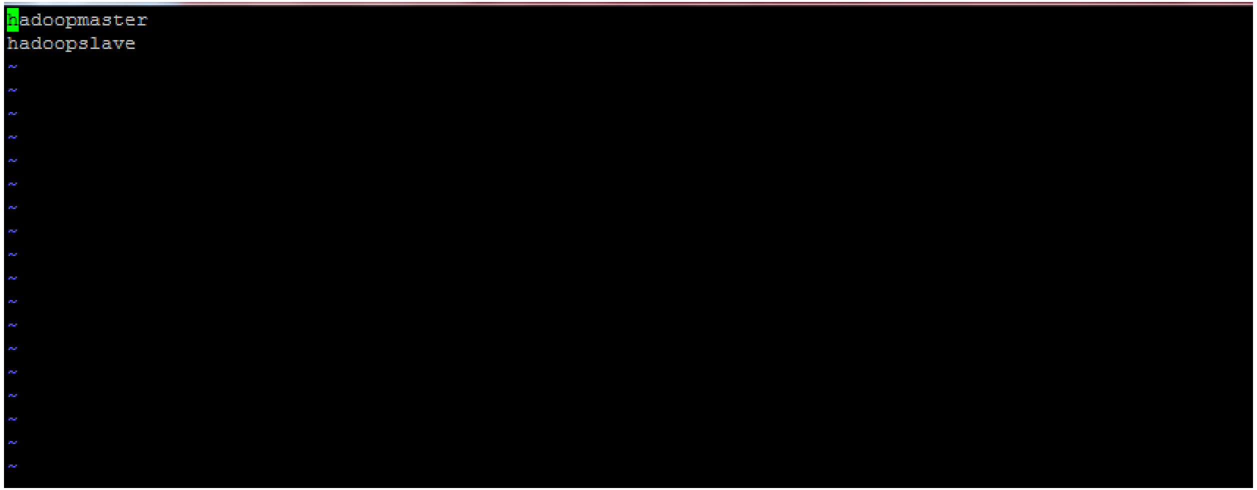


```
hadoopmaster
~
~
~
~
~
~
~
~
~
~
~
```

6.Change slaves file under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

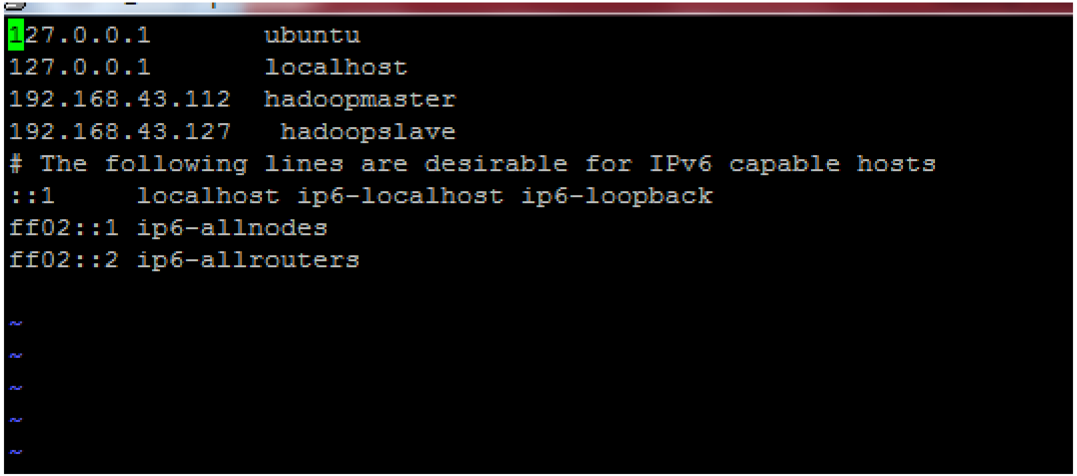


```
hadoopmaster
hadoopslave
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

7. Change /etc/hosts file and make an entry of Master and slave nodes(Logging as root user), in following ways.

HADOOP MULTINODE INSTALLATION

MASTER IMAGE INSTALLATION

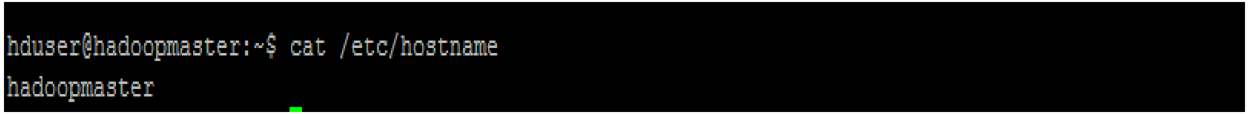


```
127.0.0.1         ubuntu
127.0.0.1         localhost
192.168.43.112  hadoopmaster
192.168.43.127   hadoopslave
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

~
~
~
~
~
```

8. Change hostname to hadoopmaster in /etc/hostname file (Logging as root user)

HADOOP MULTINODE INSTALLATION

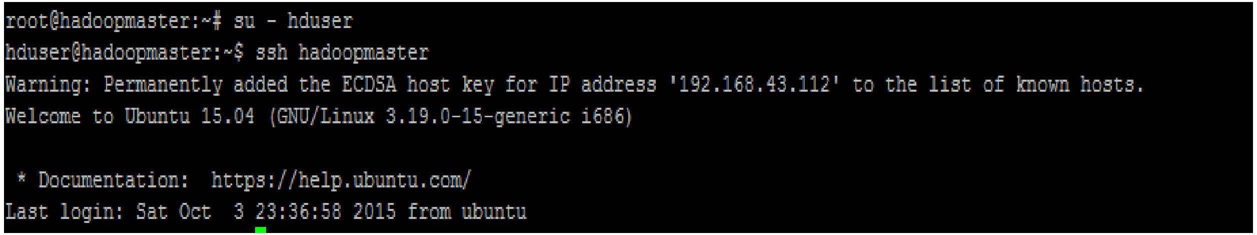
MASTER IMAGE INSTALLATION



```
hduser@hadoopmaster:~$ cat /etc/hostname
hadoopmaster
```

9. Reboot the system so that changes are reflected.

10. Perform and SSH to hadoopmaster(it should be prompting for password as password less connection is already established).



```
root@hadoopmaster:~# su - hduser
hduser@hadoopmaster:~$ ssh hadoopmaster
Warning: Permanently added the ECDSA host key for IP address '192.168.43.112' to the list of known hosts.
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-15-generic i686)

 * Documentation:  https://help.ubuntu.com/
Last login: Sat Oct  3 23:36:58 2015 from ubuntu
```

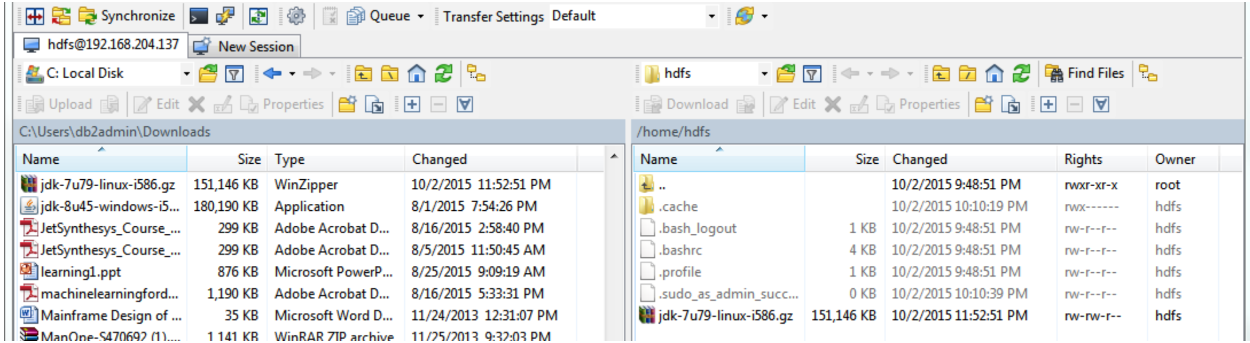
HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

1

JAVA INSTALLATION:
1. Download Java from oracle website-
http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

2. Copy JDK(java development toolkit) tar file to Server Using Winscp



Transfer Settings Default

C:\Users\db2admin\Downloads

Name	Size	Type	Changed
jdk-7u79-linux-i586.gz	151,146 KB	WinZipper	10/2/2015 11:52:51 PM
jdk-8u45-windows-i5...	180,190 KB	Application	8/1/2015 7:54:26 PM
JetSynthesys_Course_...	299 KB	Adobe Acrobat D...	8/16/2015 2:58:40 PM
JetSynthesys_Course_...	299 KB	Adobe Acrobat D...	8/5/2015 11:50:45 AM
learning1.ppt	876 KB	Microsoft PowerP...	8/25/2015 9:09:19 AM
machinelearningford...	1,190 KB	Adobe Acrobat D...	8/16/2015 5:33:31 PM
Mainframe Design of ...	35 KB	Microsoft Word D...	11/24/2013 12:31:07 PM
ManOne-S470692 (1)....	1,141 KB	WinRAR ZIP archive	11/25/2013 9:32:03 PM

/home/hdfs

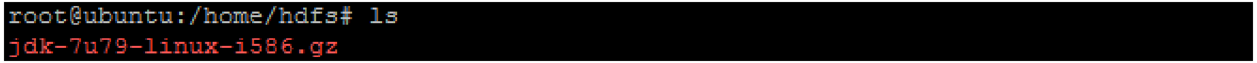
Name	Size	Changed	Rights	Owner
..		10/2/2015 9:48:51 PM	rwxr-xr-x	root
.cache		10/2/2015 10:10:19 PM	rwx------	hdfs
.bash_logout	1 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.bashrc	4 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.profile	1 KB	10/2/2015 9:48:51 PM	rw-r--r--	hdfs
.sudo_as_admin_succ...	0 KB	10/2/2015 10:10:39 PM	rw-r--r--	hdfs
jdk-7u79-linux-i586.gz	151,146 KB	10/2/2015 11:52:51 PM	rw-rw-r--	hdfs

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

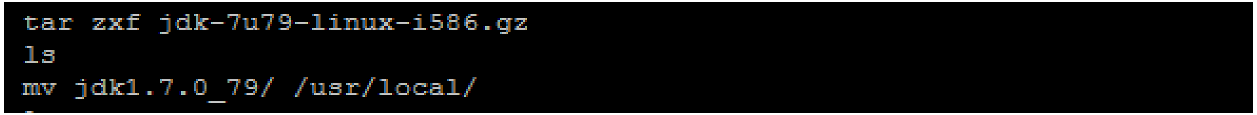
1

3. Verify if JDK is successfully moved to Server location



```
root@ubuntu:/home/hdfs# ls
jdk-7u79-linux-i586.gz
```

4. Extract it to common location i.e "/usr/local", so that it will be accessible to all users.



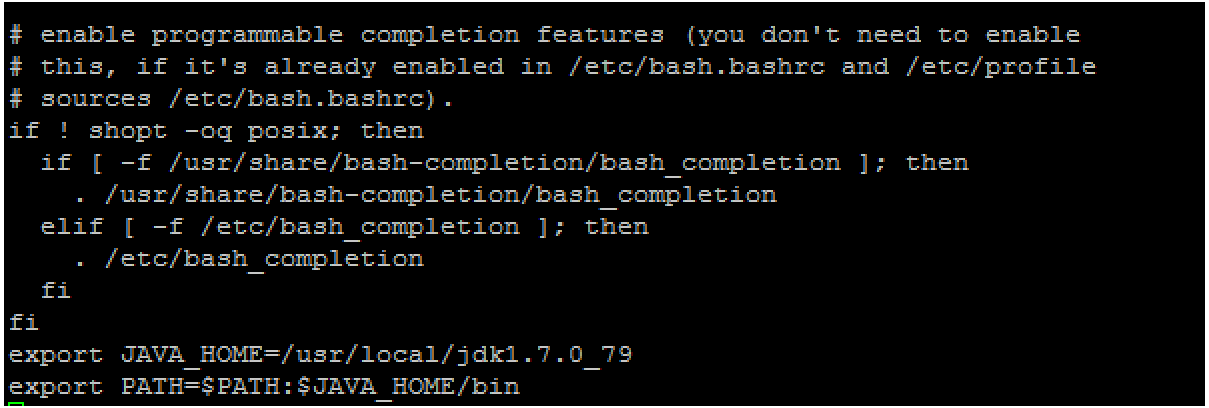
```
tar zxf jdk-7u79-linux-i586.gz
ls
mv jdk1.7.0_79/ /usr/local/
```

5. Set the path in .bashrc profile of user(this step will be done later when user will be created specific to hadoop installation)

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

1



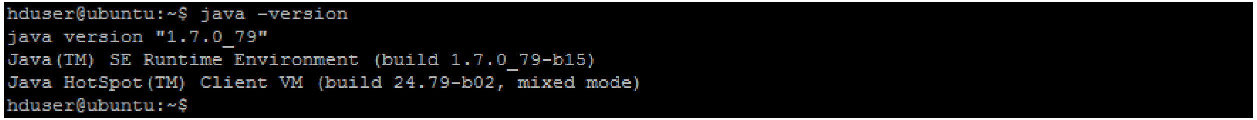
```
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
export JAVA_HOME=/usr/local/jdk1.7.0_79
export PATH=$PATH:$JAVA_HOME/bin
```

6. Either Re Login to hduser user after making above .bashrc changes to get reflected or use” *source .bashrc* “ of user *hduser* in bash shell.Run following command to check which version of Java is correctly installed.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

1

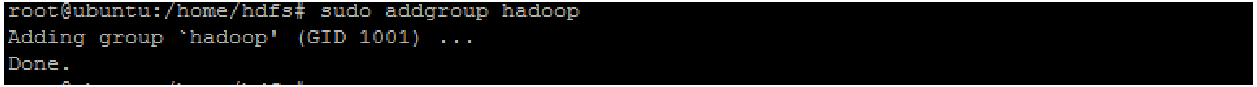


```
hduser@ubuntu:~$ java -version
java version "1.7.0_79"
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
Java HotSpot(TM) Client VM (build 24.79-b02, mixed mode)
hduser@ubuntu:~$
```

2

ADDING USER and GROUP SPECIFIC TO BIG DATA COMPONENTS:

1. Add new group hadoop

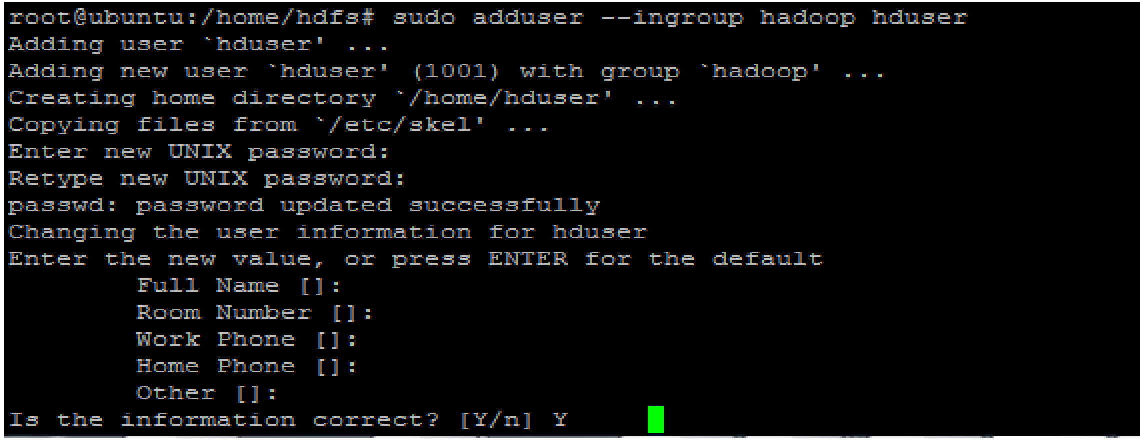


```
root@ubuntu:/home/hdfs# sudo addgroup hadoop
Adding group `hadoop' (GID 1001) ...
Done.
```

2. Add new user named hduser and associate it with group hadoop.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

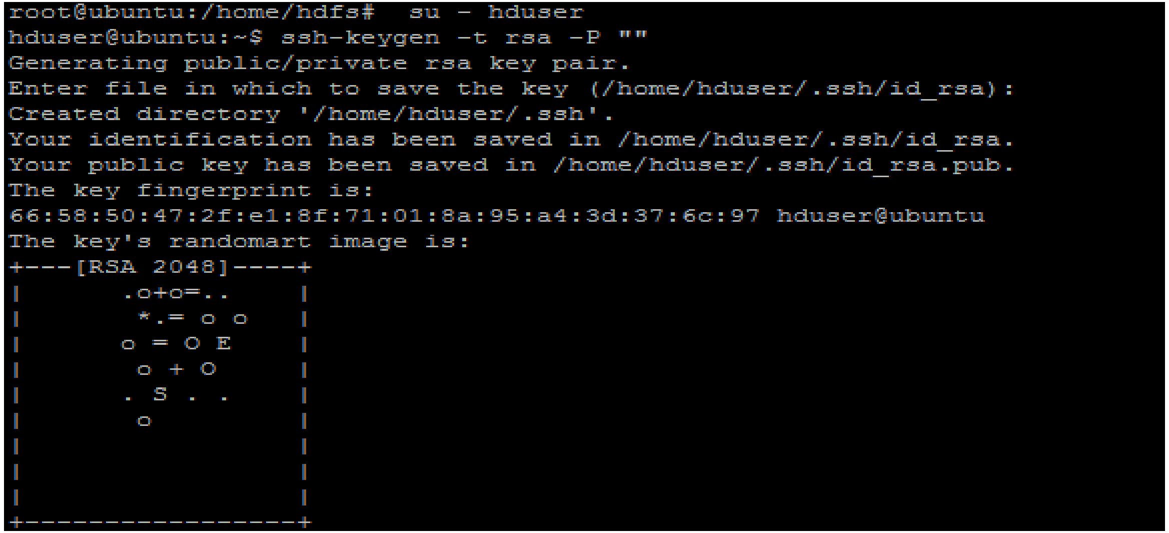


```
root@ubuntu:/home/hdfs# sudo adduser --ingroup hadoop hduser
Adding user `hduser' ...
Adding new user `hduser' (1001) with group `hadoop' ...
Creating home directory `/home/hduser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hduser
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
```

3 Configure SSH to create password less connection.(This will be done using RSA algo, which will generate public(id_rsa.pub) and private(id_rsa) key .Now if this node wants to connect with any other node using password less connection, then public key needs to be transferred to other node)

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

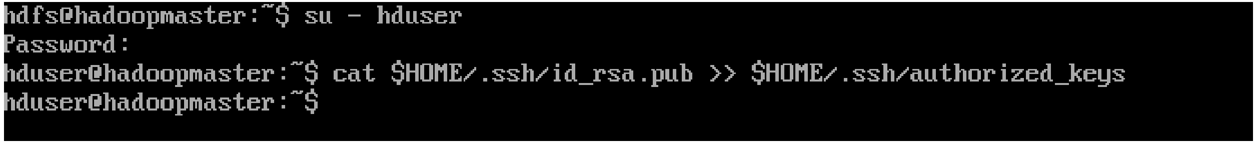


```
root@ubuntu:/home/hdfs#  su - hduser
hduser@ubuntu:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
66:58:50:47:2f:e1:8f:71:01:8a:95:a4:3d:37:6c:97 hduser@ubuntu
The key's randomart image is:
+---[RSA 2048]----+
|      .o+o=..    |
|       *.= o o   |
|      o = O E    |
|       o + O     |
|      . S . .    |
|       o         |
|                 |
|                 |
|                 |
+-----------------+
```

4 Enable SSH access with this newly generated password less connection keys.

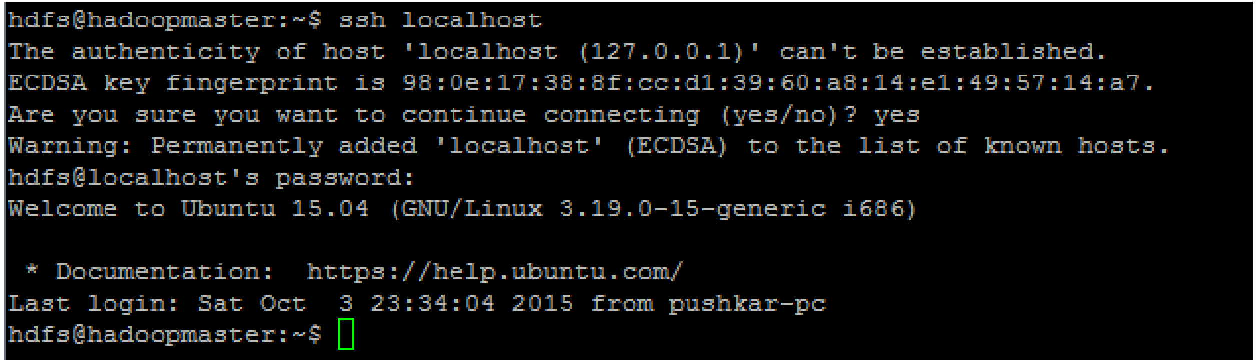
HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION



```
hdfs@hadoopmaster:~$ su - hduser
Password:
hduser@hadoopmaster:~$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
hduser@hadoopmaster:~$
```

5 Login to localhost using ssh and see if you are successfully able to logged in.



```
hdfs@hadoopmaster:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is 98:0e:17:38:8f:cc:d1:39:60:a8:14:e1:49:57:14:a7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
hdfs@localhost's password:
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-15-generic i686)

 * Documentation:  https://help.ubuntu.com/
Last login: Sat Oct  3 23:34:04 2015 from pushkar-pc
hdfs@hadoopmaster:~$
```

HADOOP MULTINODE INSTALLATION

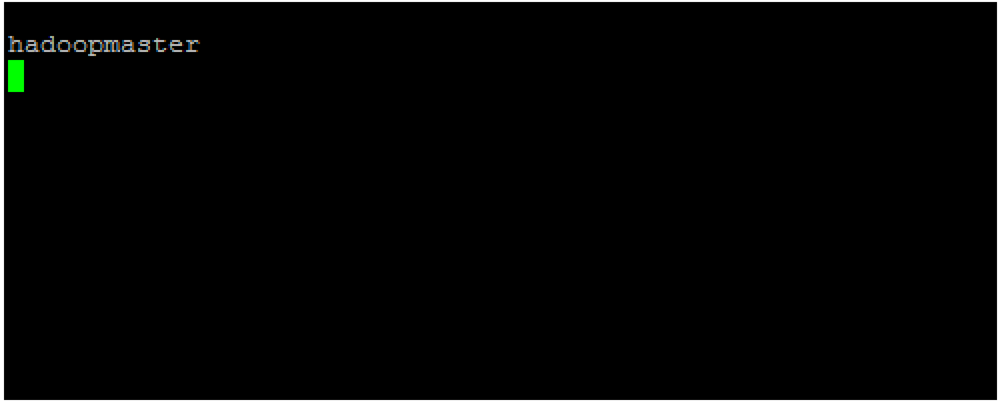
SLAVE IMAGE INSTALLATION

6

CHAGING HOSTNAME TO MASTER AND SLAVE

Note: Master will act as Namenode, while Slave as Datanode.

1. Renaming Hostname for Master as Hadoopmaster, in “/etc/hostname” file as root user.



```
hadoopmaster
```

Note: Slave will be configured later after Master completes its configuration successfully.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

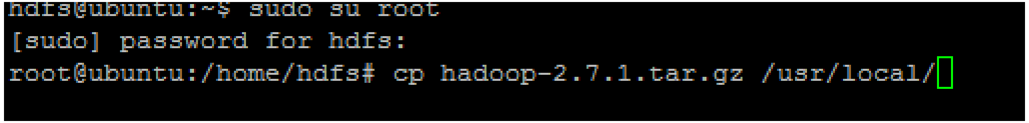
7

Hadoop Installation

1. Download Hadoop 2.6.1 tar image using below link-

http://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.6.1/hadoop-2.6.1.tar.gz

2. Extract it to common location i.e "/usr/local", so that it will be accessible to all users.

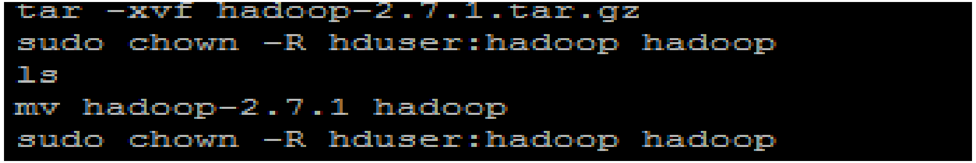


```
hdfs@ubuntu:~$ sudo su root
[sudo] password for hdfs:
root@ubuntu:/home/hdfs# cp hadoop-2.7.1.tar.gz /usr/local/
```

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

3. Change owner and group to hduser and hadoop resp. so that user hduser also have access of hadoop directories.

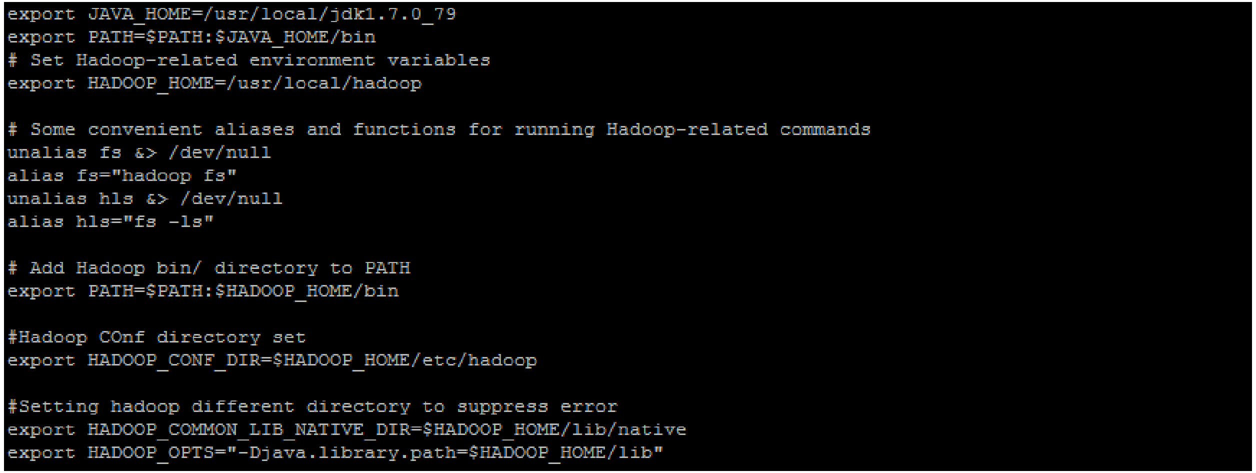


```
tar -xvf hadoop-2.7.1.tar.gz
sudo chown -R hduser:hadoop hadoop
ls
mv hadoop-2.7.1 hadoop
sudo chown -R hduser:hadoop hadoop
```

4. Make an entry of hadoop directories like configuration , binaries etc. so that hadoop command will be made accessible through bash shell from any location.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

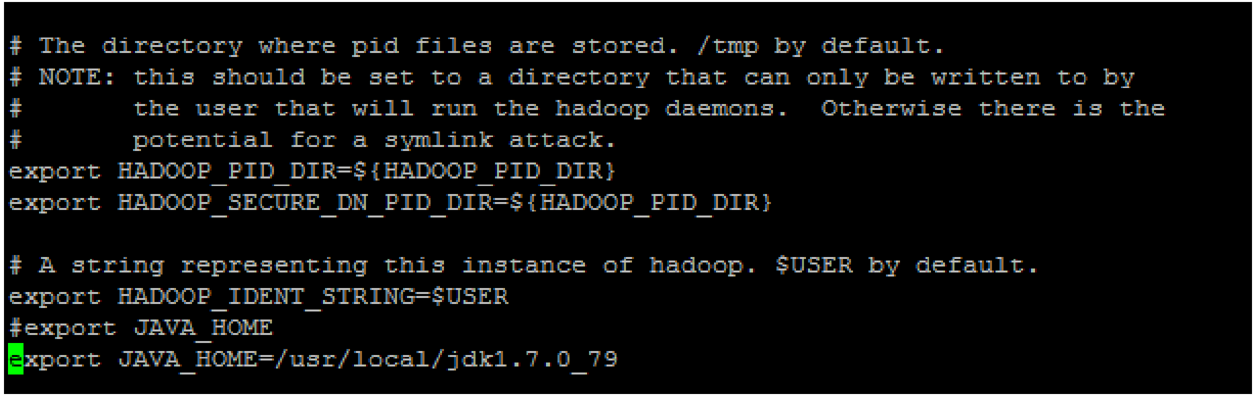


```
export JAVA_HOME=/usr/local/jdk1.7.0_79
export PATH=$PATH:$JAVA_HOME/bin
# Set Hadoop-related environment variables
export HADOOP_HOME=/usr/local/hadoop

# Some convenient aliases and functions for running Hadoop-related commands
unalias fs &> /dev/null
alias fs="hadoop fs"
unalias hls &> /dev/null
alias hls="fs -ls"

# Add Hadoop bin/ directory to PATH
export PATH=$PATH:$HADOOP_HOME/bin

#Hadoop COnf directory set
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop

#Setting hadoop different directory to suppress error
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

5. Add JAVA_HOME in hadoop_env.sh script located at `/usr/local/hadoop/etc/hadoop` directory.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

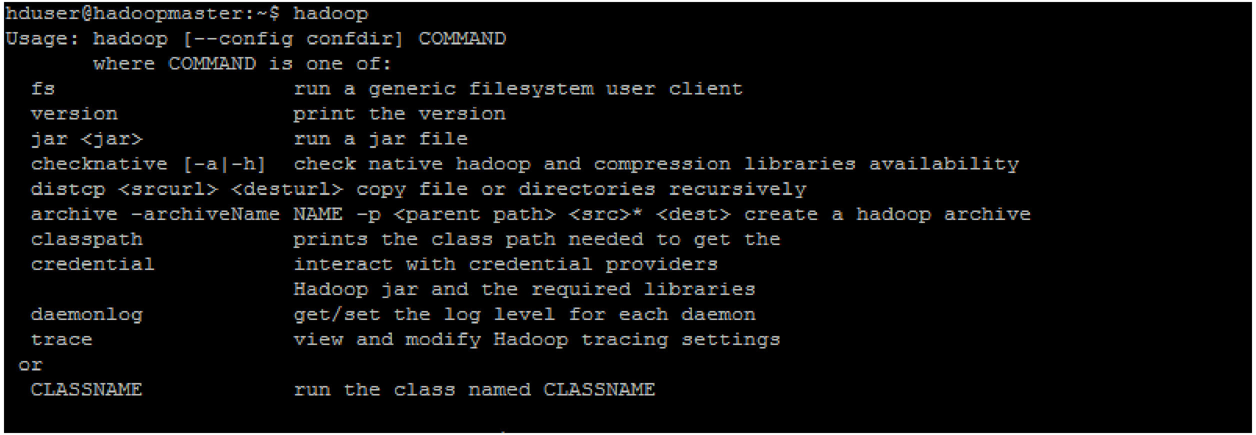


```
# The directory where pid files are stored. /tmp by default.
# NOTE: this should be set to a directory that can only be written to by
#       the user that will run the hadoop daemons.  Otherwise there is the
#       potential for a symlink attack.
export HADOOP_PID_DIR=${HADOOP_PID_DIR}
export HADOOP_SECURE_DN_PID_DIR=${HADOOP_PID_DIR}

# A string representing this instance of hadoop. $USER by default.
export HADOOP_IDENT_STRING=$USER
#export JAVA_HOME
export JAVA_HOME=/usr/local/jdk1.7.0_79
```

Log in as hduser and Check from command line hadoop command is accessible or not, after souring the .bashrc of user hduser.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION



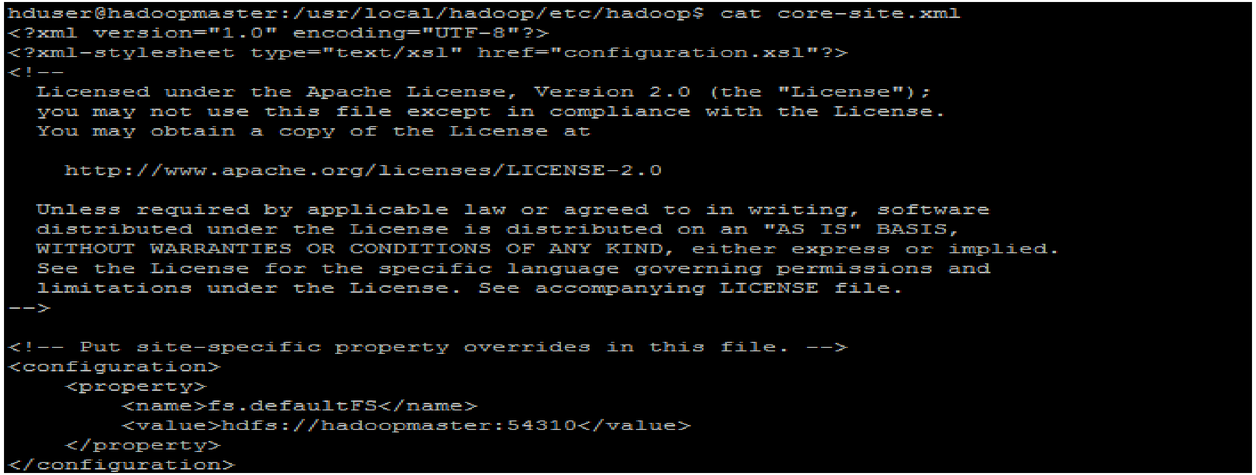
```
hduser@hadoopmaster:~$ hadoop
Usage: hadoop [--config confdir] COMMAND
       where COMMAND is one of:
  fs                   run a generic filesystem user client
  version              print the version
  jar <jar>            run a jar file
  checknative [-a|-h]  check native hadoop and compression libraries availability
  distcp <srcurl> <desturl> copy file or directories recursively
  archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
  classpath            prints the class path needed to get the
  credential           interact with credential providers
                       Hadoop jar and the required libraries
  daemonlog            get/set the log level for each daemon
  trace                view and modify Hadoop tracing settings
 or
  CLASSNAME            run the class named CLASSNAME
```

8

Make configuration file changes for hadoop(both hdfs and yarn)
1. Configuration setting for CORE-SITE.xml under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

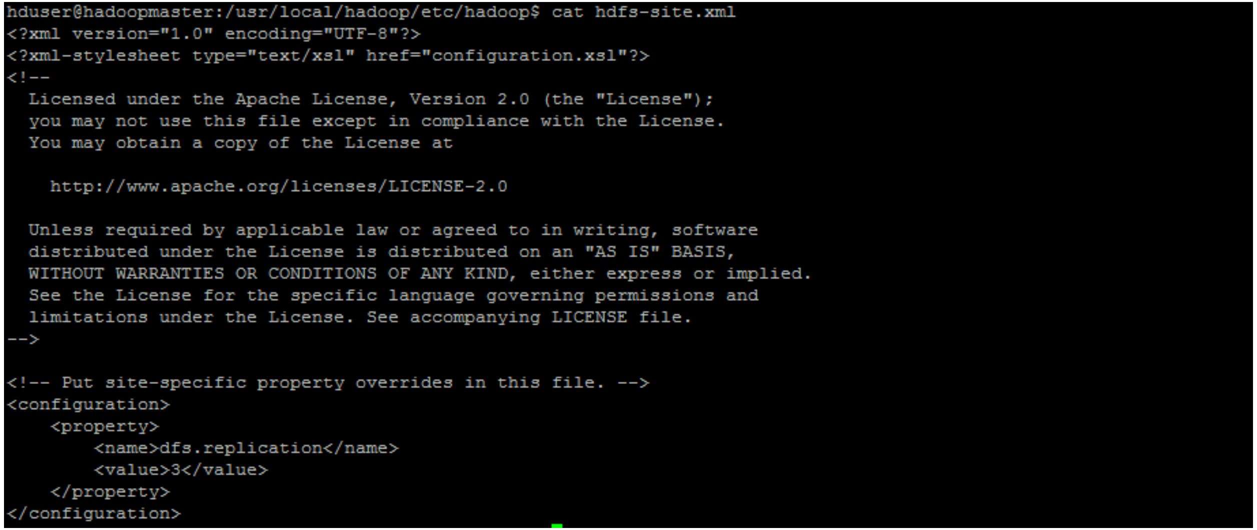


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoopmaster:54310</value>
    </property>
</configuration>
```

2. Configuration Setting for hdfs-site.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

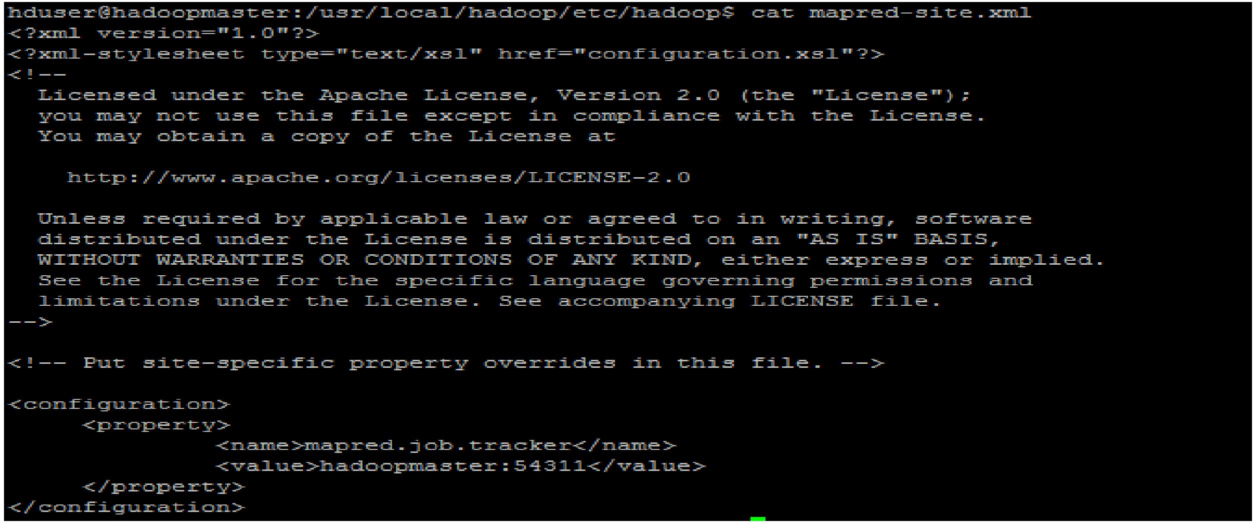


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>3</value>
    </property>
</configuration>
```

3. Configuration setting for mapred-site.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

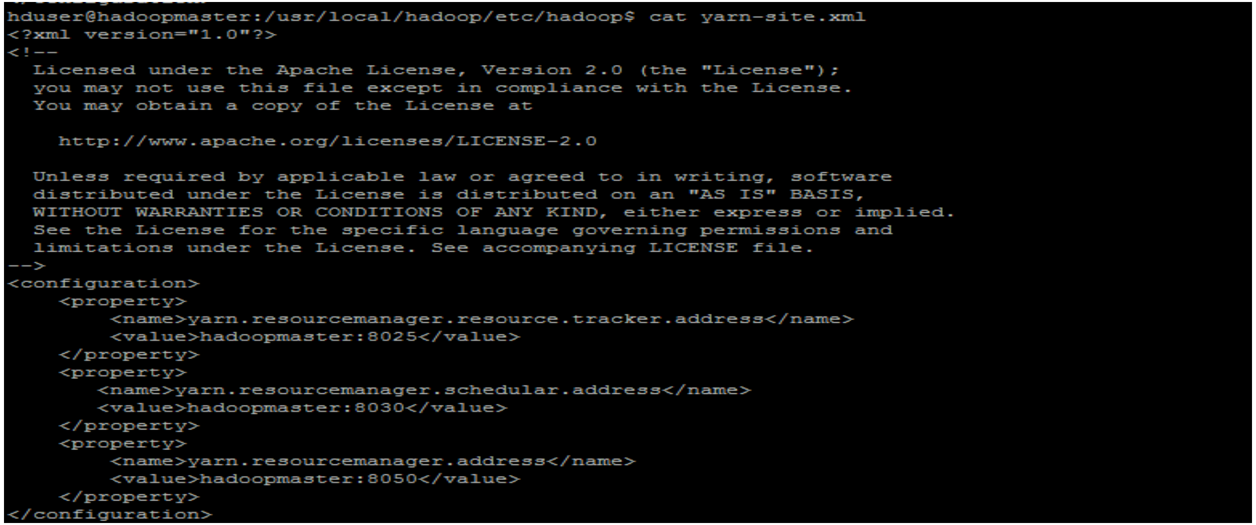


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

<configuration>
     <property>
              <name>mapred.job.tracker</name>
              <value>hadoopmaster:54311</value>
     </property>
</configuration>
```

4. Configuration setting for YARN-SITE.xml under “/usr/local/hadoop/etc/hadoop”.

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

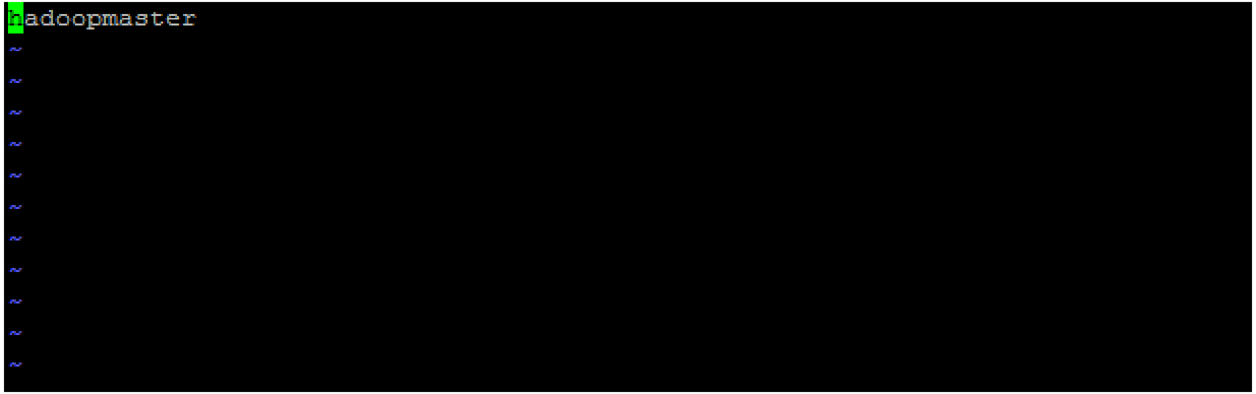


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cat yarn-site.xml
<?xml version="1.0"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<configuration>
    <property>
        <name>yarn.resourcemanager.resource.tracker.address</name>
        <value>hadoopmaster:8025</value>
    </property>
    <property>
       <name>yarn.resourcemanager.schedular.address</name>
       <value>hadoopmaster:8030</value>
    </property>
    <property>
        <name>yarn.resourcemanager.address</name>
        <value>hadoopmaster:8050</value>
    </property>
</configuration>
```

5. Change masters file under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

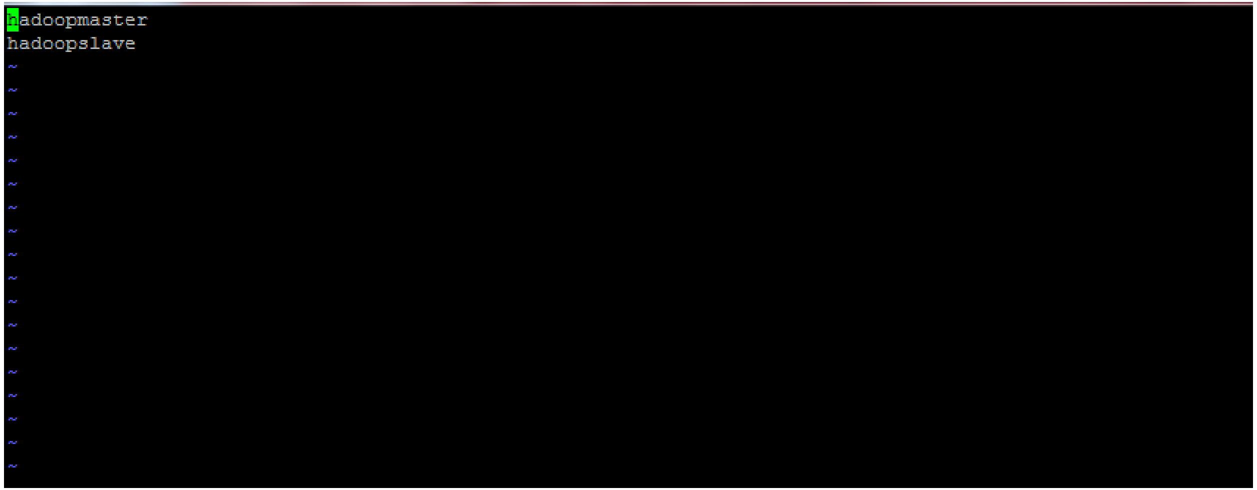


```
hadoopmaster
~
~
~
~
~
~
~
~
~
~
~
```

6.Change slaves file under “/usr/local/hadoop/etc/hadoop”

HADOOP MULTINODE INSTALLATION

SLAVE IMAGE INSTALLATION

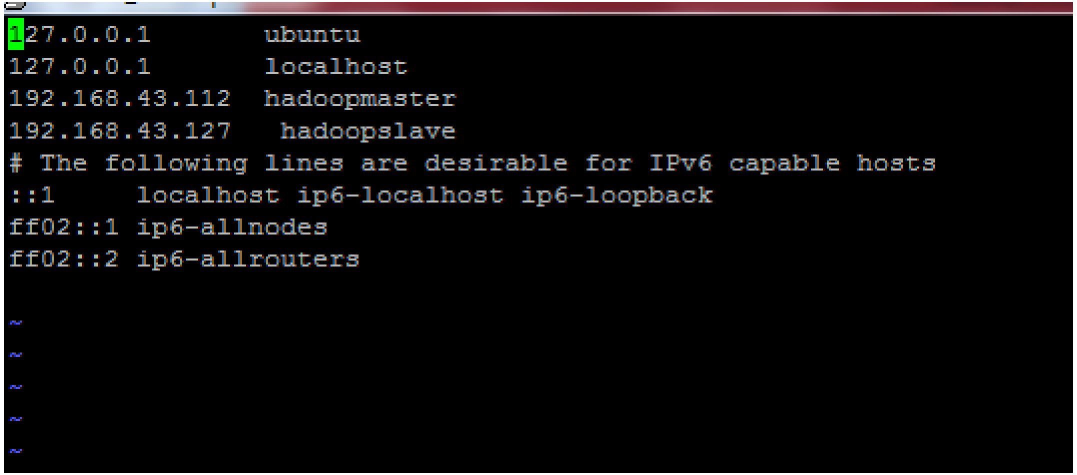


```
hadoopmaster
hadoopslave
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

7. Change /etc/hosts file and make an entry of Master and slave nodes(Logging as root user), in following ways.

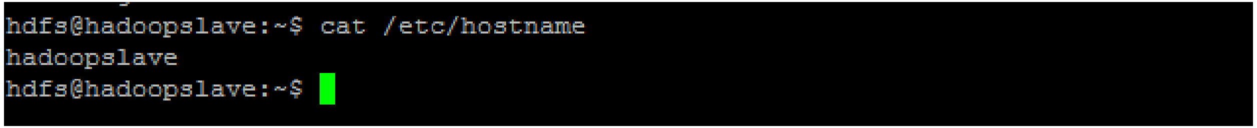
HADOOP MULTINODE INSTALLATION

SlAVE IMAGE INSTALLATION



```
127.0.0.1       ubuntu
127.0.0.1       localhost
192.168.43.112  hadoopmaster
192.168.43.127   hadoopslave
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

~
~
~
~
~
```

8. Change hostname to hadoopslave in /etc/hostname file (Logging as root user)

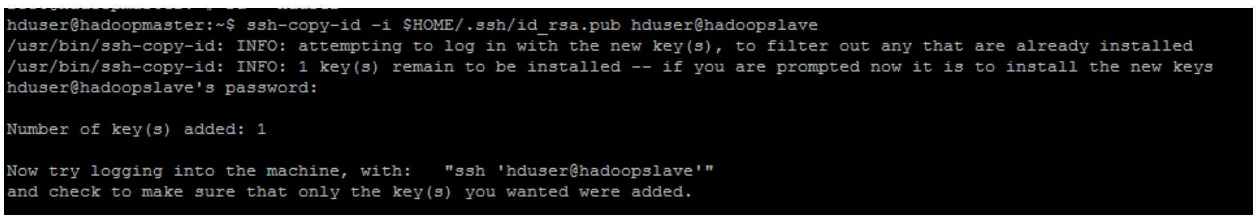


```
hdfs@hadoopslave:~$ cat /etc/hostname
hadoopslave
hdfs@hadoopslave:~$
```

HADOOP MULTINODE INSTALLATION

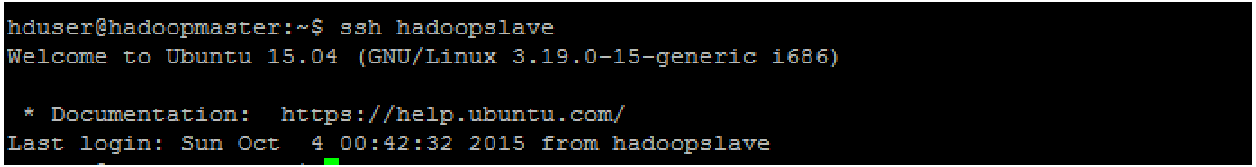
SLAVE MASTER INSTALLATION

1. Adding public key to slave hduser.



```
hduser@hadoopmaster:~$ ssh-copy-id -i $HOME/.ssh/id_rsa.pub hduser@hadoopslave
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
hduser@hadoopslave's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'hduser@hadoopslave'"
and check to make sure that only the key(s) you wanted were added.
```

2. Login to hduser@hadoopslave from master.



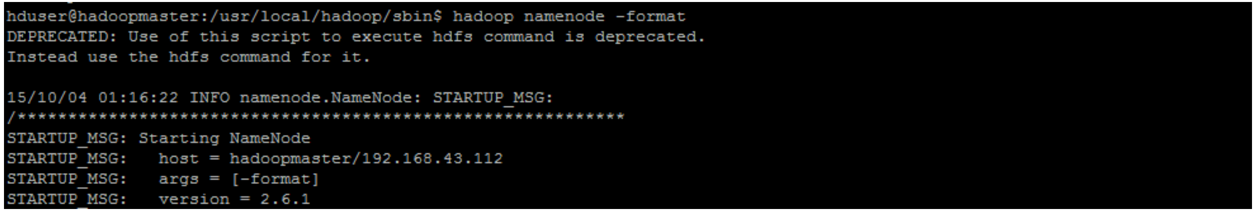
```
hduser@hadoopmaster:~$ ssh hadoopslave
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-15-generic i686)

 * Documentation:  https://help.ubuntu.com/
Last login: Sun Oct  4 00:42:32 2015 from hadoopslave
```

HADOOP MULTINODE INSTALLATION

SLAVE MASTER INSTALLATION

3. FORMATTING THE HDFS FILESYSTEM VIA NAMNODE.

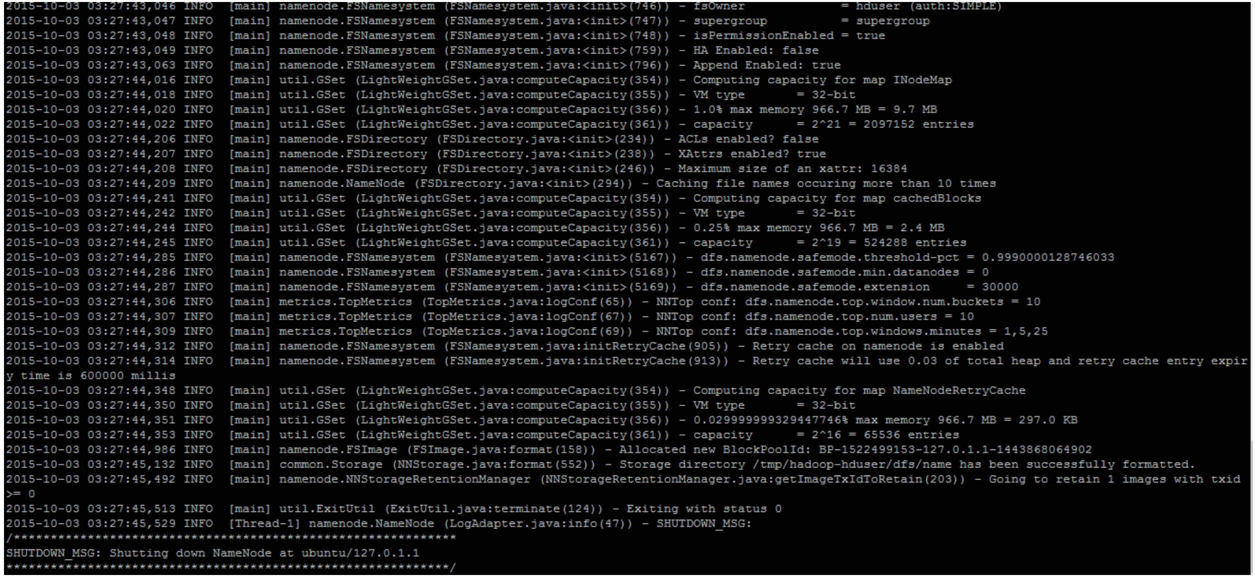
Before we start our new multi-node cluster, we must format Hadoop’s distributed filesystem (HDFS) via the NameNode. You need to do this the first time you set up an Hadoop cluster. *Warning: Do not format a running cluster because this will erase all existing data in the HDFS filesytem!*
To format the filesystem (which simply initializes the directory specified by the dfs.name.dir variable on the NameNode), run the command-



```
hduser@hadoopmaster:/usr/local/hadoop/sbin$ hadoop namenode -format
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

15/10/04 01:16:22 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = hadoopmaster/192.168.43.112
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.6.1
```

HADOOP MULTINODE INSTALLATION

SLAVE MASTER INSTALLATION



```
2015-10-03 03:27:43,046 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(746)) - fsOwner             = hduser (auth:SIMPLE)
2015-10-03 03:27:43,047 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(747)) - supergroup          = supergroup
2015-10-03 03:27:43,048 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(748)) - isPermissionEnabled = true
2015-10-03 03:27:43,049 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(759)) - HA Enabled: false
2015-10-03 03:27:43,063 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(796)) - Append Enabled: true
2015-10-03 03:27:44,016 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(354)) - Computing capacity for map INodeMap
2015-10-03 03:27:44,018 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(355)) - VM type       = 32-bit
2015-10-03 03:27:44,020 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(356)) - 1.0% max memory 966.7 MB = 9.7 MB
2015-10-03 03:27:44,022 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(361)) - capacity      = 2^21 = 2097152 entries
2015-10-03 03:27:44,206 INFO  [main] namenode.FSDirectory (FSDirectory.java:<init>(234)) - ACLs enabled? false
2015-10-03 03:27:44,207 INFO  [main] namenode.FSDirectory (FSDirectory.java:<init>(238)) - XAttrs enabled? true
2015-10-03 03:27:44,208 INFO  [main] namenode.FSDirectory (FSDirectory.java:<init>(246)) - Maximum size of an xattr: 16384
2015-10-03 03:27:44,209 INFO  [main] namenode.NameNode (FSDirectory.java:<init>(294)) - Caching file names occuring more than 10 times
2015-10-03 03:27:44,241 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(354)) - Computing capacity for map cachedBlocks
2015-10-03 03:27:44,242 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(355)) - VM type       = 32-bit
2015-10-03 03:27:44,244 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(356)) - 0.25% max memory 966.7 MB = 2.4 MB
2015-10-03 03:27:44,245 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(361)) - capacity      = 2^19 = 524288 entries
2015-10-03 03:27:44,285 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(5167)) - dfs.namenode.safemode.threshold-pct = 0.9990000128746033
2015-10-03 03:27:44,286 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(5168)) - dfs.namenode.safemode.min.datanodes = 0
2015-10-03 03:27:44,287 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:<init>(5169)) - dfs.namenode.safemode.extension     = 30000
2015-10-03 03:27:44,306 INFO  [main] metrics.TopMetrics (TopMetrics.java:logConf(65)) - NNTop conf: dfs.namenode.top.window.num.buckets = 10
2015-10-03 03:27:44,307 INFO  [main] metrics.TopMetrics (TopMetrics.java:logConf(67)) - NNTop conf: dfs.namenode.top.num.users = 10
2015-10-03 03:27:44,309 INFO  [main] metrics.TopMetrics (TopMetrics.java:logConf(69)) - NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2015-10-03 03:27:44,312 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:initRetryCache(905)) - Retry cache on namenode is enabled
2015-10-03 03:27:44,314 INFO  [main] namenode.FSNamesystem (FSNamesystem.java:initRetryCache(913)) - Retry cache will use 0.03 of total heap and retry cache entry expir
y time is 600000 millis
2015-10-03 03:27:44,348 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(354)) - Computing capacity for map NameNodeRetryCache
2015-10-03 03:27:44,350 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(355)) - VM type       = 32-bit
2015-10-03 03:27:44,351 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(356)) - 0.029999999329447746% max memory 966.7 MB = 297.0 KB
2015-10-03 03:27:44,353 INFO  [main] util.GSet (LightWeightGSet.java:computeCapacity(361)) - capacity      = 2^16 = 65536 entries
2015-10-03 03:27:44,986 INFO  [main] namenode.FSImage (FSImage.java:format(158)) - Allocated new BlockPoolId: BP-1522499153-127.0.1.1-1443868064902
2015-10-03 03:27:45,132 INFO  [main] common.Storage (NNStorage.java:format(552)) - Storage directory /tmp/hadoop-hduser/dfs/name has been successfully formatted.
2015-10-03 03:27:45,492 INFO  [main] namenode.NNStorageRetentionManager (NNStorageRetentionManager.java:getImageTxIdToRetain(203)) - Going to retain 1 images with txid
>= 0
2015-10-03 03:27:45,513 INFO  [main] util.ExitUtil (ExitUtil.java:terminate(124)) - Exiting with status 0
2015-10-03 03:27:45,529 INFO  [Thread-1] namenode.NameNode (LogAdapter.java:info(47)) - SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at ubuntu/127.0.1.1
************************************************************/
```

HADOOP MULTINODE INSTALLATION

SLAVE MASTER INSTALLATION

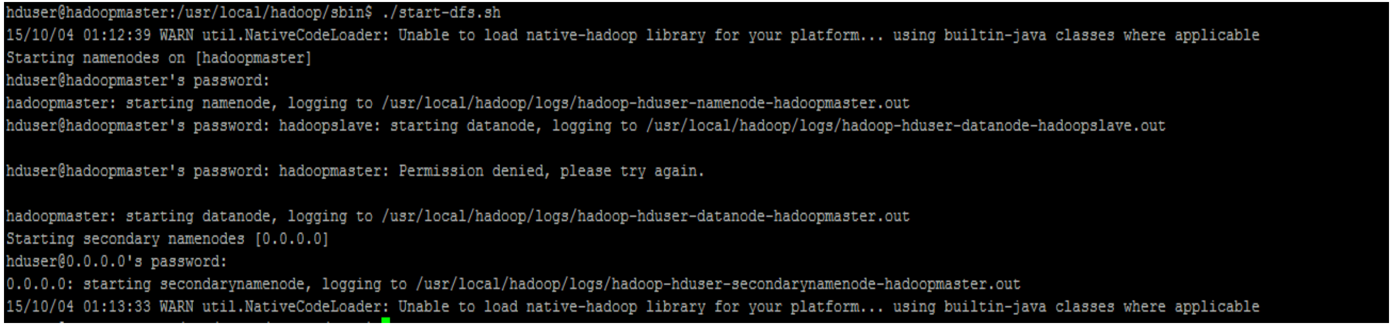
Background: The HDFS name table is stored on the NameNode's (here: master) local filesystem in the directory specified by dfs.name.dir. The name table is used by the NameNode to store tracking and coordination information for the DataNodes

4. START HDFS DAEMON IN MASTER

Run the command bin/start-dfs.sh on the machine you want the (primary) NameNode to run on. This will bring up HDFS with the NameNode running on the machine you ran the previous command on, and DataNodes on the machines listed in the conf/slaves file.(after successful run of start-dfs.sh run start-yarn.sh)

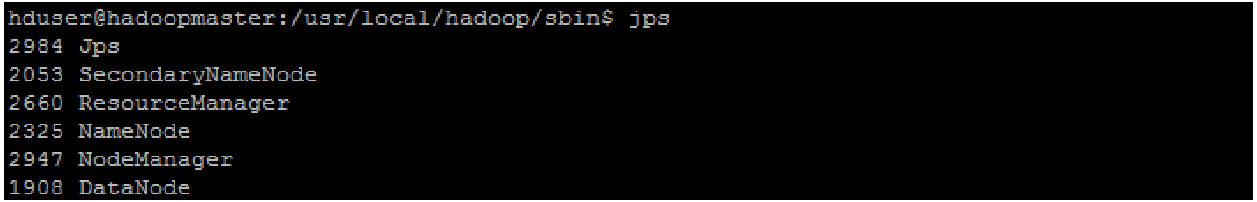
HADOOP MULTINODE INSTALLATION

SLAVE MASTER INSTALLATION



```
hduser@hadoopmaster:/usr/local/hadoop/sbin$ ./start-dfs.sh
15/10/04 01:12:39 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [hadoopmaster]
hduser@hadoopmaster's password:
hadoopmaster: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-namenode-hadoopmaster.out
hduser@hadoopmaster's password: hadoopslave: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-hadoopslave.out

hduser@hadoopmaster's password: hadoopmaster: Permission denied, please try again.

hadoopmaster: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-hadoopmaster.out
Starting secondary namenodes [0.0.0.0]
hduser@0.0.0.0's password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hduser-secondarynamenode-hadoopmaster.out
15/10/04 01:13:33 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

5. Use JPS to see if all services are running successfully or not.(both YARN and HDFS)

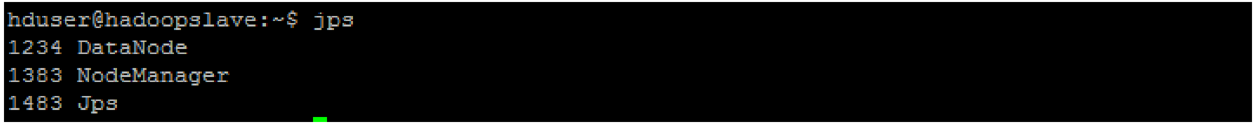


```
hduser@hadoopmaster:/usr/local/hadoop/sbin$ jps
2984 Jps
2053 SecondaryNameNode
2660 ResourceManager
2325 NameNode
2947 NodeManager
1908 DataNode
```

HADOOP MULTINODE INSTALLATION

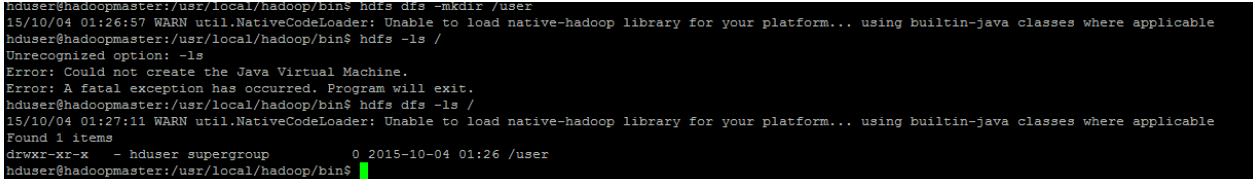
SLAVE MASTER INSTALLATION

6. At the same time check in slave machine all services running using jps-



```
hduser@hadoopslave:~$ jps
1234 DataNode
1383 NodeManager
1483 Jps
```

7. CREATING HADOOP DIRECTORY FROM MASTER

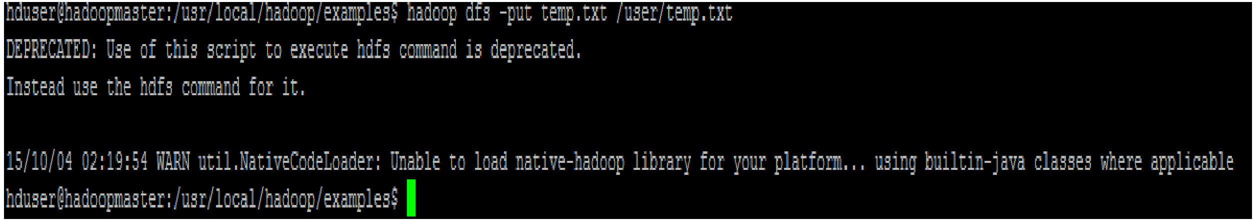


```
hduser@hadoopmaster:/usr/local/hadoop/bin$ hdfs dfs -mkdir /user
15/10/04 01:26:57 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@hadoopmaster:/usr/local/hadoop/bin$ hdfs -ls /
Unrecognized option: -ls
Error: Could not create the Java Virtual Machine.
Error: A fatal exception has occurred. Program will exit.
hduser@hadoopmaster:/usr/local/hadoop/bin$ hdfs dfs -ls /
15/10/04 01:27:11 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x   - hduser supergroup          0 2015-10-04 01:26 /user
hduser@hadoopmaster:/usr/local/hadoop/bin$
```

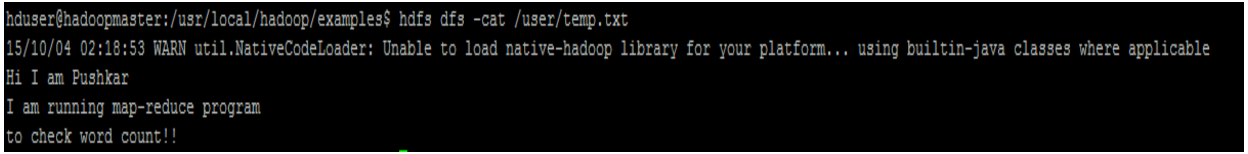
HADOOP MULTINODE INSTALLATION

SLAVE MASTER INSTALLATION

8. Creating a file and putting it to HDFS-



```
hduser@hadoopmaster:/usr/local/hadoop/examples$ hadoop dfs -put temp.txt /user/temp.txt
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

15/10/04 02:19:54 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@hadoopmaster:/usr/local/hadoop/examples$
```

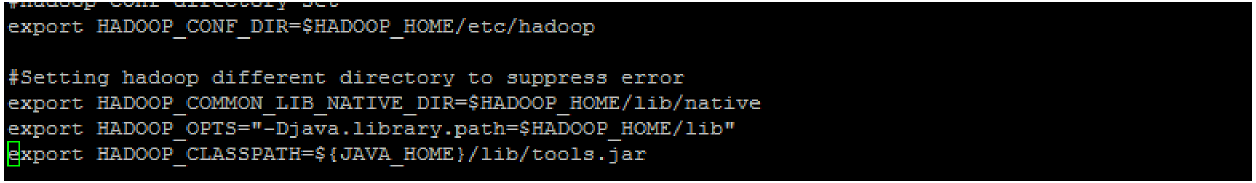


```
hduser@hadoopmaster:/usr/local/hadoop/examples$ hdfs dfs -cat /user/temp.txt
15/10/04 02:18:53 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Hi I am Pushkar
I am running map-reduce program
to check word count!!
```

HADOOP MULTINODE INSTALLATION

RUNNING MAPREDUCE JOB-

1.Make an entry of class path of Java in .bashrc



```
#hadoop conf directory set
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop

#Setting hadoop different directory to suppress error
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
```

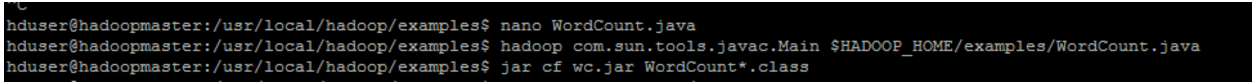
2. Create a directory named examples under $HADOOP_HOME, and keep WordCount.java in that directory**.**

WordCount.java

HADOOP MULTINODE INSTALLATION

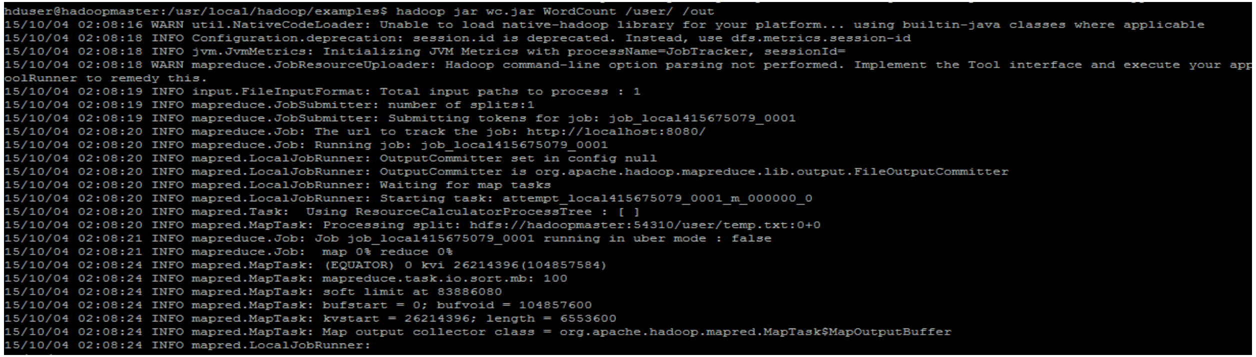
RUNNING MAPREDUCE JOB-

3. Complile and create executable jar file



```
^C
hduser@hadoopmaster:/usr/local/hadoop/examples$ nano WordCount.java
hduser@hadoopmaster:/usr/local/hadoop/examples$ hadoop com.sun.tools.javac.Main $HADOOP_HOME/examples/WordCount.java
hduser@hadoopmaster:/usr/local/hadoop/examples$ jar cf wc.jar WordCount*.class
```

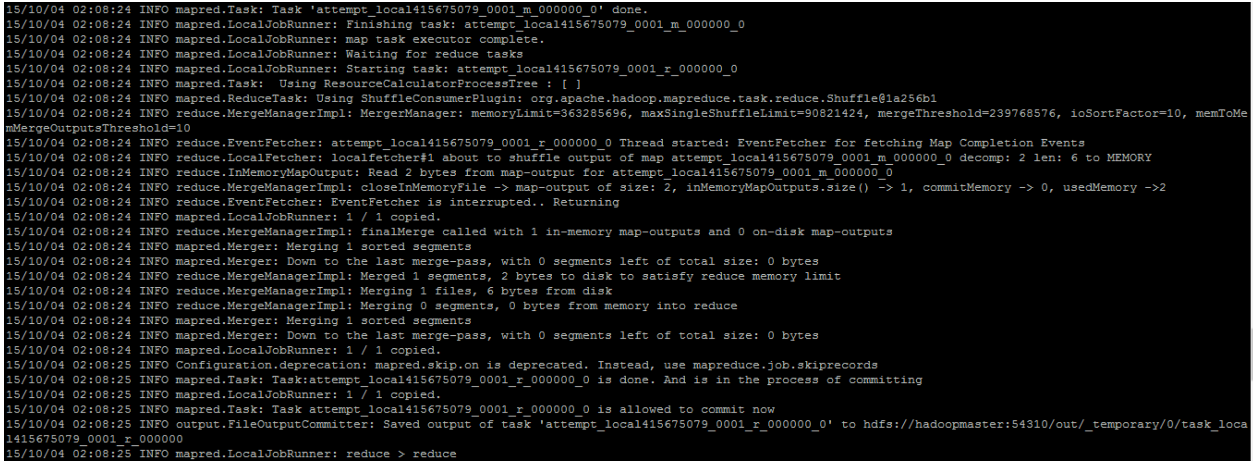
4. Run mapreduce job, by providing input file as temp.txt



```
hduser@hadoopmaster:/usr/local/hadoop/examples$ hadoop jar wc.jar WordCount /user/ /out
15/10/04 02:08:16 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
15/10/04 02:08:18 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
15/10/04 02:08:18 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
15/10/04 02:08:18 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your app
oolRunner to remedy this.
15/10/04 02:08:19 INFO input.FileInputFormat: Total input paths to process : 1
15/10/04 02:08:19 INFO mapreduce.JobSubmitter: number of splits:1
15/10/04 02:08:19 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local415675079_0001
15/10/04 02:08:20 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
15/10/04 02:08:20 INFO mapreduce.Job: Running job: job_local415675079_0001
15/10/04 02:08:20 INFO mapred.LocalJobRunner: OutputCommitter set in config null
15/10/04 02:08:20 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
15/10/04 02:08:20 INFO mapred.LocalJobRunner: Waiting for map tasks
15/10/04 02:08:20 INFO mapred.LocalJobRunner: Starting task: attempt_local415675079_0001_m_000000_0
15/10/04 02:08:20 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
15/10/04 02:08:20 INFO mapred.MapTask: Processing split: hdfs://hadoopmaster:54310/user/temp.txt:0+0
15/10/04 02:08:21 INFO mapreduce.Job: Job job_local415675079_0001 running in uber mode : false
15/10/04 02:08:21 INFO mapreduce.Job:  map 0% reduce 0%
15/10/04 02:08:24 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
15/10/04 02:08:24 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
15/10/04 02:08:24 INFO mapred.MapTask: soft limit at 83886080
15/10/04 02:08:24 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
15/10/04 02:08:24 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
15/10/04 02:08:24 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
15/10/04 02:08:24 INFO mapred.LocalJobRunner:
```

HADOOP MULTINODE INSTALLATION

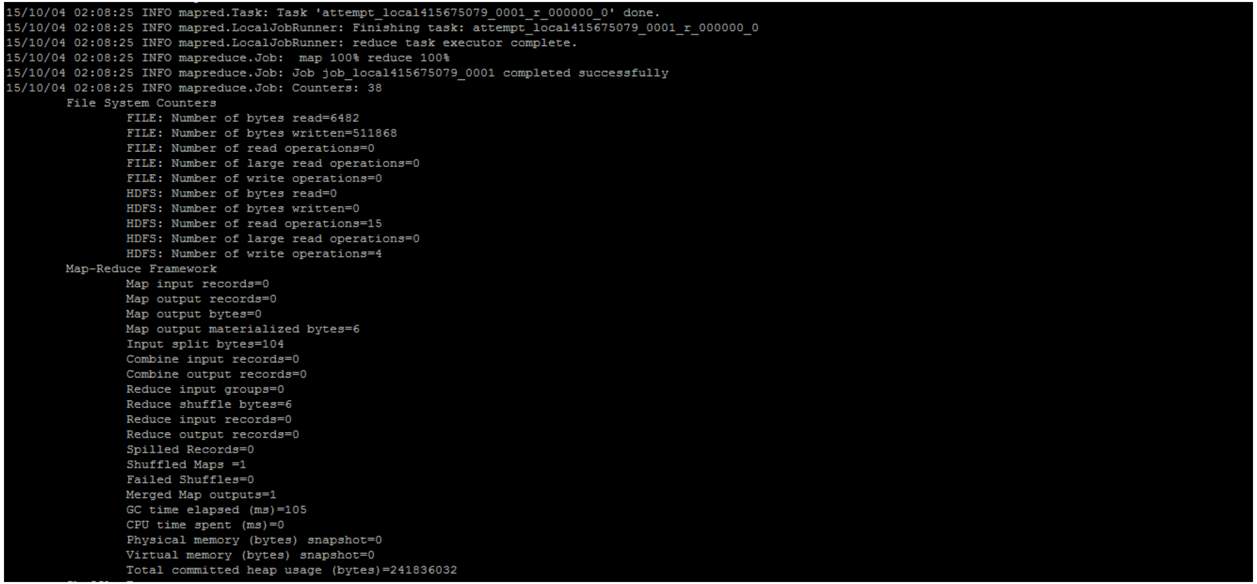
RUNNING MAPREDUCE JOB-



```
15/10/04 02:08:24 INFO mapred.Task: Task 'attempt_local415675079_0001_m_000000_0' done.
15/10/04 02:08:24 INFO mapred.LocalJobRunner: Finishing task: attempt_local415675079_0001_m_000000_0
15/10/04 02:08:24 INFO mapred.LocalJobRunner: map task executor complete.
15/10/04 02:08:24 INFO mapred.LocalJobRunner: Waiting for reduce tasks
15/10/04 02:08:24 INFO mapred.LocalJobRunner: Starting task: attempt_local415675079_0001_r_000000_0
15/10/04 02:08:24 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
15/10/04 02:08:24 INFO mapred.ReduceTask: Using ShuffleConsumerPlugin: org.apache.hadoop.mapreduce.task.reduce.Shuffle@1a256b1
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: MergerManager: memoryLimit=363285696, maxSingleShuffleLimit=90821424, mergeThreshold=239768576, ioSortFactor=10, memToMe
mMergeOutputsThreshold=10
15/10/04 02:08:24 INFO reduce.EventFetcher: attempt_local415675079_0001_r_000000_0 Thread started: EventFetcher for fetching Map Completion Events
15/10/04 02:08:24 INFO reduce.LocalFetcher: localfetcher#1 about to shuffle output of map attempt_local415675079_0001_m_000000_0 decomp: 2 len: 6 to MEMORY
15/10/04 02:08:24 INFO reduce.InMemoryMapOutput: Read 2 bytes from map-output for attempt_local415675079_0001_m_000000_0
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: closeInMemoryFile -> map-output of size: 2, inMemoryMapOutputs.size() -> 1, commitMemory -> 0, usedMemory ->2
15/10/04 02:08:24 INFO reduce.EventFetcher: EventFetcher is interrupted.. Returning
15/10/04 02:08:24 INFO mapred.LocalJobRunner: 1 / 1 copied.
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: finalMerge called with 1 in-memory map-outputs and 0 on-disk map-outputs
15/10/04 02:08:24 INFO mapred.Merger: Merging 1 sorted segments
15/10/04 02:08:24 INFO mapred.Merger: Down to the last merge-pass, with 0 segments left of total size: 0 bytes
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: Merged 1 segments, 2 bytes to disk to satisfy reduce memory limit
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: Merging 1 files, 6 bytes from disk
15/10/04 02:08:24 INFO reduce.MergeManagerImpl: Merging 0 segments, 0 bytes from memory into reduce
15/10/04 02:08:24 INFO mapred.Merger: Merging 1 sorted segments
15/10/04 02:08:24 INFO mapred.Merger: Down to the last merge-pass, with 0 segments left of total size: 0 bytes
15/10/04 02:08:24 INFO mapred.LocalJobRunner: 1 / 1 copied.
15/10/04 02:08:25 INFO Configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skiprecords
15/10/04 02:08:25 INFO mapred.Task: Task:attempt_local415675079_0001_r_000000_0 is done. And is in the process of committing
15/10/04 02:08:25 INFO mapred.LocalJobRunner: 1 / 1 copied.
15/10/04 02:08:25 INFO mapred.Task: Task attempt_local415675079_0001_r_000000_0 is allowed to commit now
15/10/04 02:08:25 INFO output.FileOutputCommitter: Saved output of task 'attempt_local415675079_0001_r_000000_0' to hdfs://hadoopmaster:54310/out/_temporary/0/task_loca
l415675079_0001_r_000000
15/10/04 02:08:25 INFO mapred.LocalJobRunner: reduce > reduce
```

HADOOP MULTINODE INSTALLATION

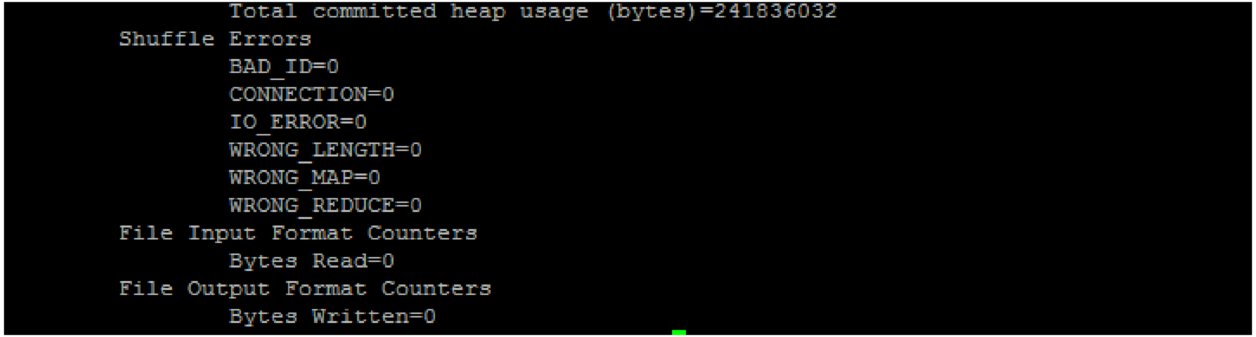
RUNNING MAPREDUCE JOB-



```
15/10/04 02:08:25 INFO mapred.Task: Task 'attempt_local415675079_0001_r_000000_0' done.
15/10/04 02:08:25 INFO mapred.LocalJobRunner: Finishing task: attempt_local415675079_0001_r_000000_0
15/10/04 02:08:25 INFO mapred.LocalJobRunner: reduce task executor complete.
15/10/04 02:08:25 INFO mapreduce.Job:  map 100% reduce 100%
15/10/04 02:08:25 INFO mapreduce.Job: Job job_local415675079_0001 completed successfully
15/10/04 02:08:25 INFO mapreduce.Job: Counters: 38
        File System Counters
                FILE: Number of bytes read=6482
                FILE: Number of bytes written=511868
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=0
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=15
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=4
        Map-Reduce Framework
                Map input records=0
                Map output records=0
                Map output bytes=0
                Map output materialized bytes=6
                Input split bytes=104
                Combine input records=0
                Combine output records=0
                Reduce input groups=0
                Reduce shuffle bytes=6
                Reduce input records=0
                Reduce output records=0
                Spilled Records=0
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=105
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=241836032
```

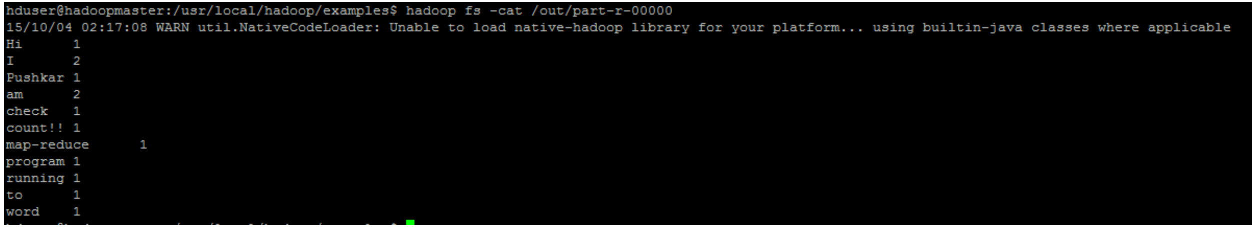
HADOOP MULTINODE INSTALLATION

RUNNING MAPREDUCE JOB-



```
                Total committed heap usage (bytes)=241836032
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
```

5. Map reduce word count example output-

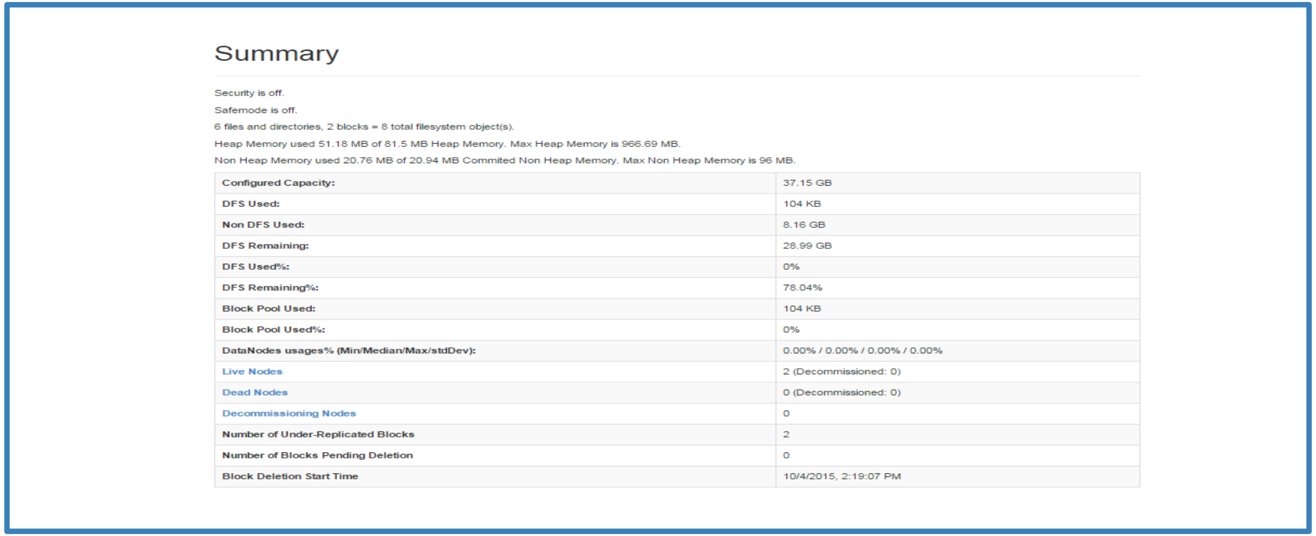


```
hduser@hadoopmaster:/usr/local/hadoop/examples$ hadoop fs -cat /out/part-r-00000
15/10/04 02:17:08 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Hi      1
I       2
Pushkar 1
am      2
check   1
count!! 1
map-reduce      1
program 1
running 1
to      1
word    1
```

HADOOP MULTINODE INSTALLATION

WEBUI-

1. Accessing Namenode (50070) at http://<masternodeip>/50070



Summary

Security is off.

Safemode is off.

6 files and directories, 2 blocks = 8 total filesystem object(s).

Heap Memory used 51.18 MB of 81.5 MB Heap Memory. Max Heap Memory is 966.69 MB.

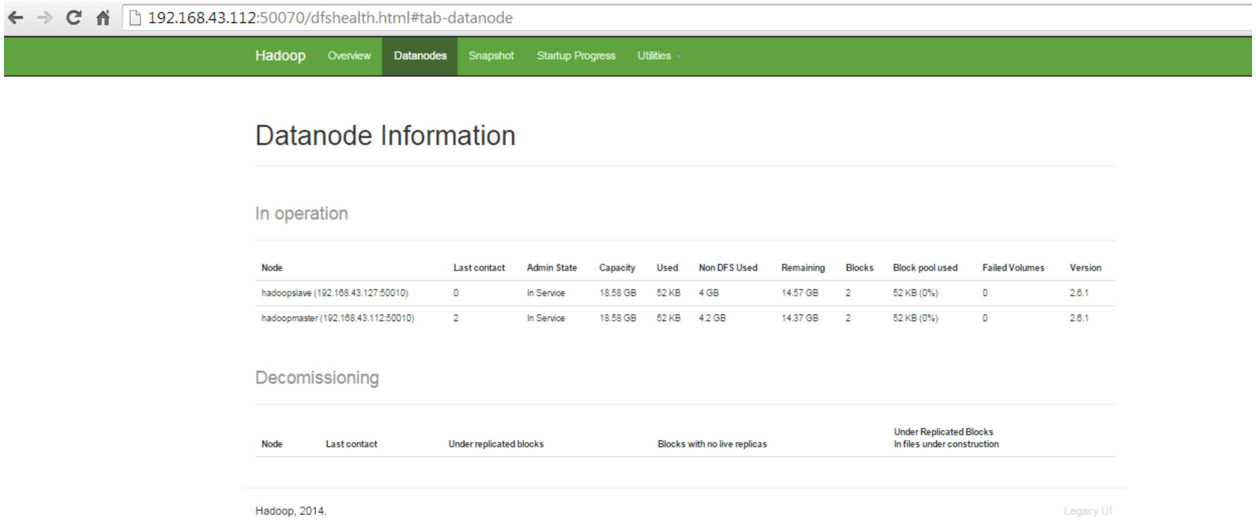
Non Heap Memory used 20.76 MB of 20.94 MB Commited Non Heap Memory. Max Non Heap Memory is 96 MB.

Configured Capacity:	37.15 GB
DFS Used:	104 KB
Non DFS Used:	8.16 GB
DFS Remaining:	28.99 GB
DFS Used%:	0%
DFS Remaining%:	78.04%
Block Pool Used:	104 KB
Block Pool Used%:	0%
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	2 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)
Decommissioning Nodes	0
Number of Under-Replicated Blocks	2
Number of Blocks Pending Deletion	0
Block Deletion Start Time	10/4/2015, 2:19:07 PM

HADOOP MULTINODE INSTALLATION

WEBUI-

2. Datanode information by clicking on Live nodes.



Datanode Information

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
hadoopslave (192.168.43.127:50010)	0	In Service	18.58 GB	52 KB	4 GB	14.57 GB	2	52 KB (0%)	0	2.6.1
hadoopmaster (192.168.43.112:50010)	2	In Service	18.58 GB	52 KB	4.2 GB	14.37 GB	2	52 KB (0%)	0	2.6.1

Decomissioning

Node	Last contact	Under replicated blocks	Blocks with no live replicas	Under Replicated Blocks In files under construction

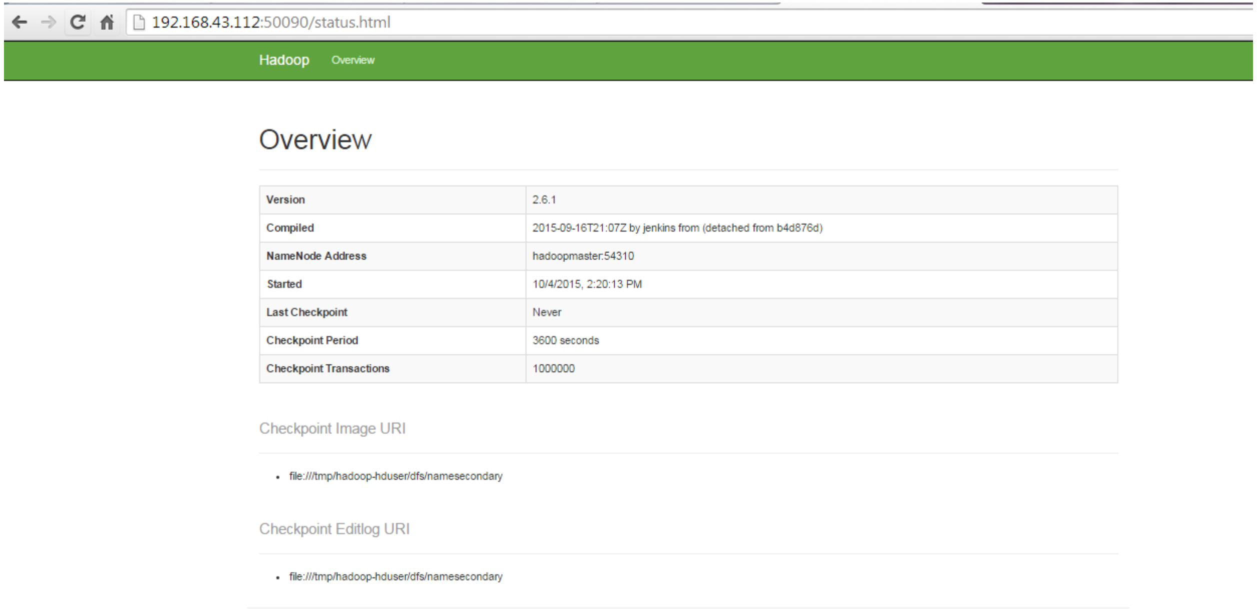
Hadoop, 2014.

Legacy UI

HADOOP MULTINODE INSTALLATION

WEBUI-

3. Secondary Namenode-



Overview

Version	2.6.1
Compiled	2015-09-16T21:07Z by jenkins from (detached from b4d876d)
NameNode Address	hadoopmaster:54310
Started	10/4/2015, 2:20:13 PM
Last Checkpoint	Never
Checkpoint Period	3600 seconds
Checkpoint Transactions	1000000

Checkpoint Image URI

- file:///tmp/hadoop-hduser/dfs/namesecondary

Checkpoint Editlog URI

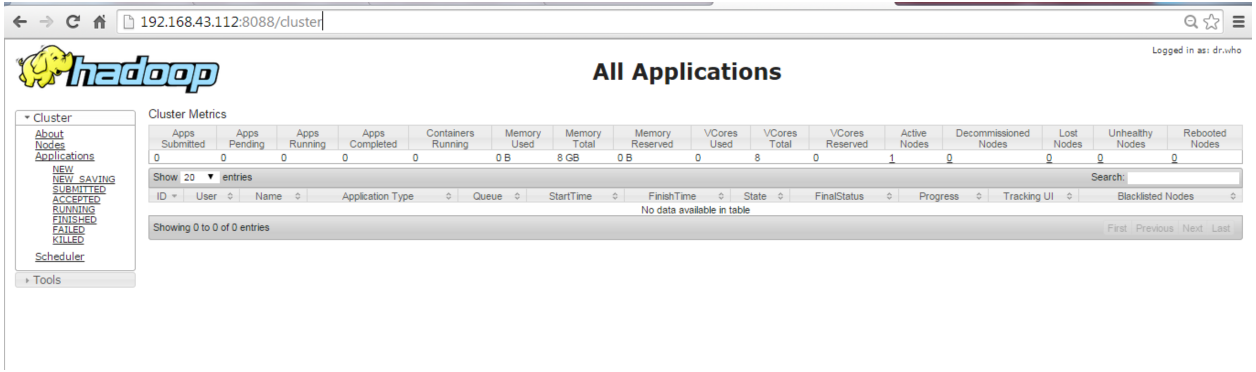
- file:///tmp/hadoop-hduser/dfs/namesecondary

HADOOP MULTINODE INSTALLATION

WEBUI-

4. Yarn Application

.



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
0	0	0	0	0	0 B	8 GB	0 B	0	8	0	1	0	0	0	0

Show 20 entries Search:

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Progress	Tracking UI	Blacklisted Nodes
No data available in table											

Showing 0 to 0 of 0 entries