Single Node Apache Hadoop Setup and Configuration on Ubuntu 14.04

Data Science Lab, The Department of Computer Science,
KSKV Kachchh University.

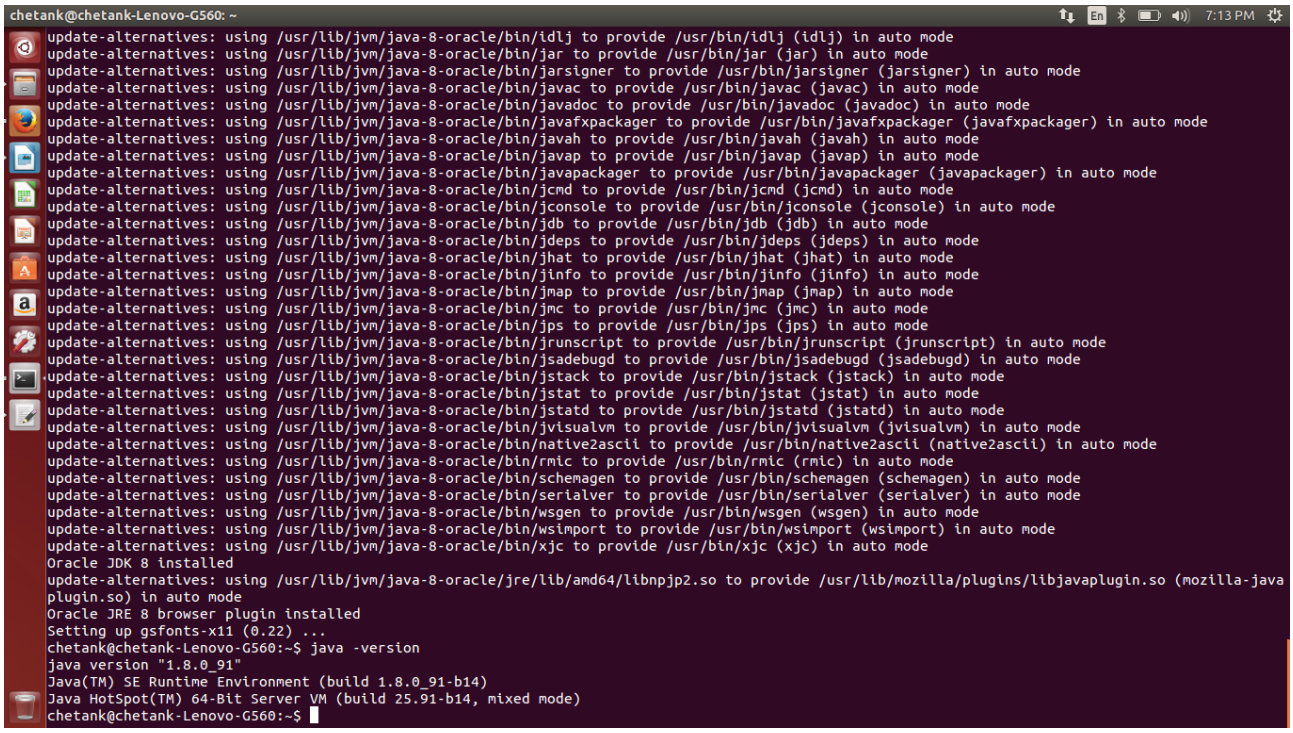
Web: http://cs.kutchuni.edu.in

1) Ubuntu 14.04 64bit Installed.
2) Java 1.8 Installed with following steps

sudo add-apt-repository ppa:webupd8team/java -y
sudo apt-get update
sudo apt-get install oracle-java8-installer

To automatically set up the Java 8 environment variables

Check it
java -version



[Figure 1] Installation of JAVA 1.8 on Ubuntu 14.04

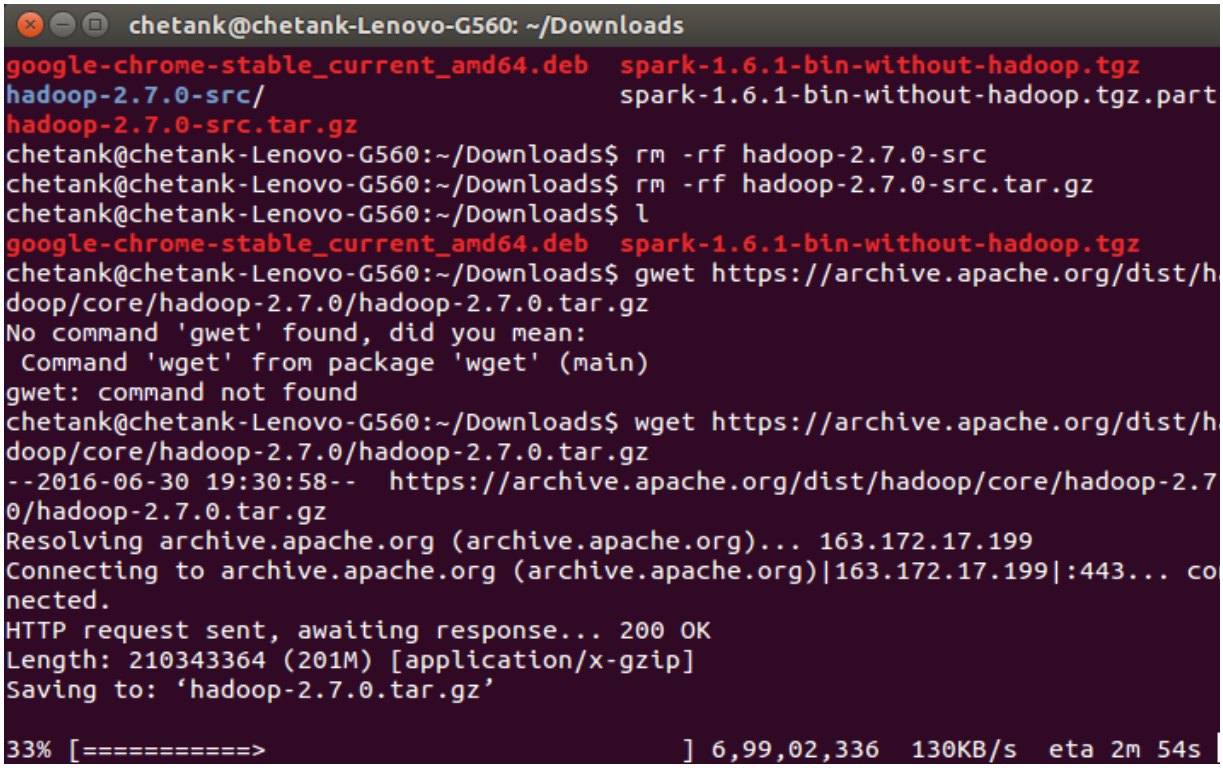
Set JAVA_HOME System environment variable, follow following steps.

1. Sudo gedit ~/.bashrc
2. go to end of the file, write *export JAVA_HOME=/usr/lib/jvm/java-8-oracle*
3. save & close the bashrc file
4. source ~/.bashrc (for saving environment variable permanently)
5. for check – echo $JAVA_HOME will give you entire path.

[Figure 2] Checking JAVA_HOME environment variable

3) **Installation of Hadoop 2.7.0**

Download Hadoop 2.7.0
wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.0/hadoop-2.7.0.tar.gz



[Figure 3] Downloading Apache Hadoop 2.7.0

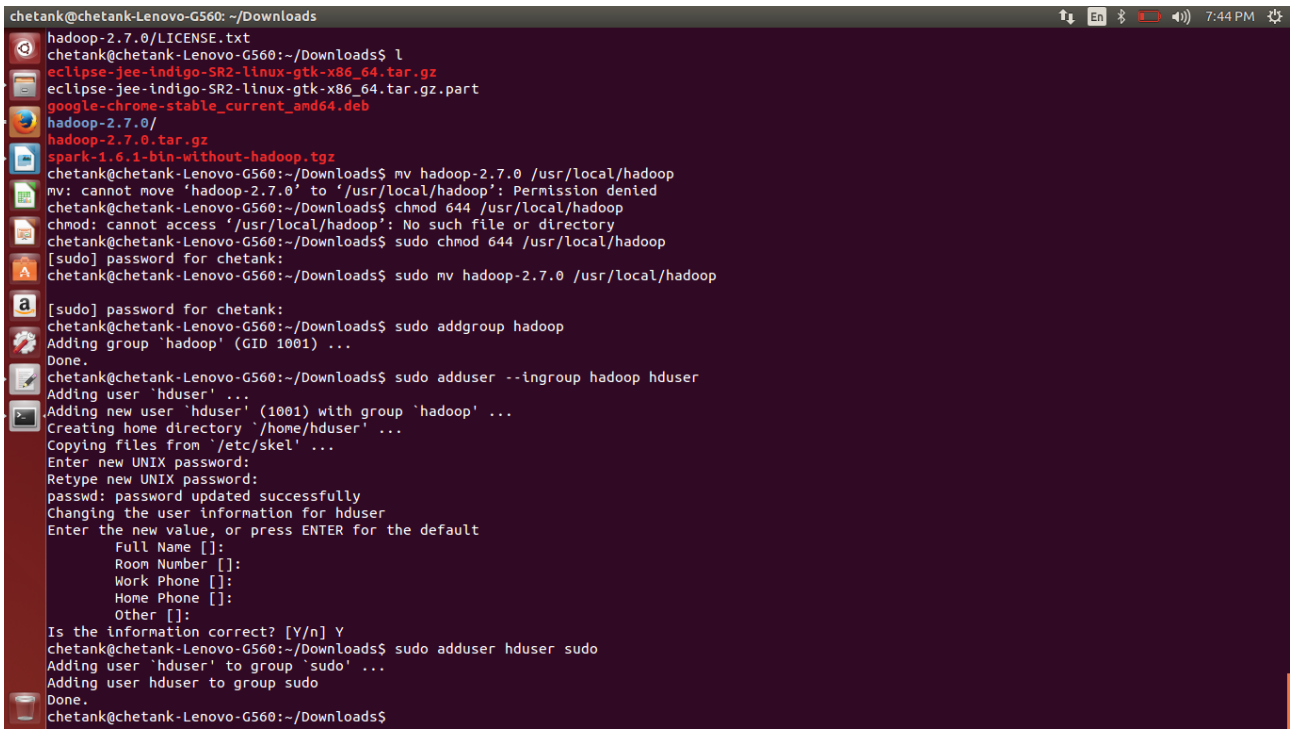
Uncompress the tar file and move to /usr/local/hadoop

$ tar xfz hadoop-2.7.0.tar.gz
$ sudo mv hadoop-2.7.0 /usr/local/hadoop

4) Add Hadoop Group and User

To avoid security issues, it’s always a good practice to setup new Hadoop user group and user account to deal with all Hadoop related activities. We will create **hadoop** as system group and **hduser** as system user.

$ sudo addgroup hadoop
$ sudo adduser --ingroup hadoop hduser
$ sudo adduser hduser sudo

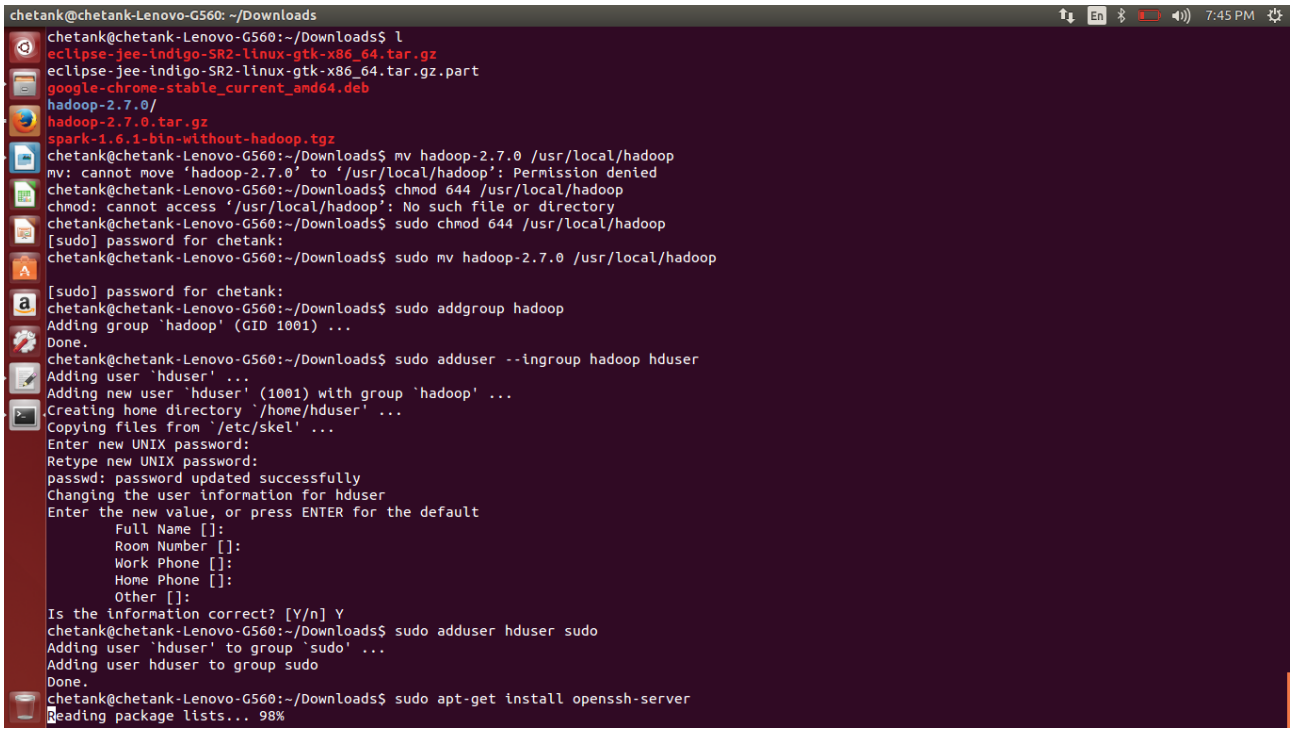


[Figure 4] Adding Hadoop group and hduser in ubuntu

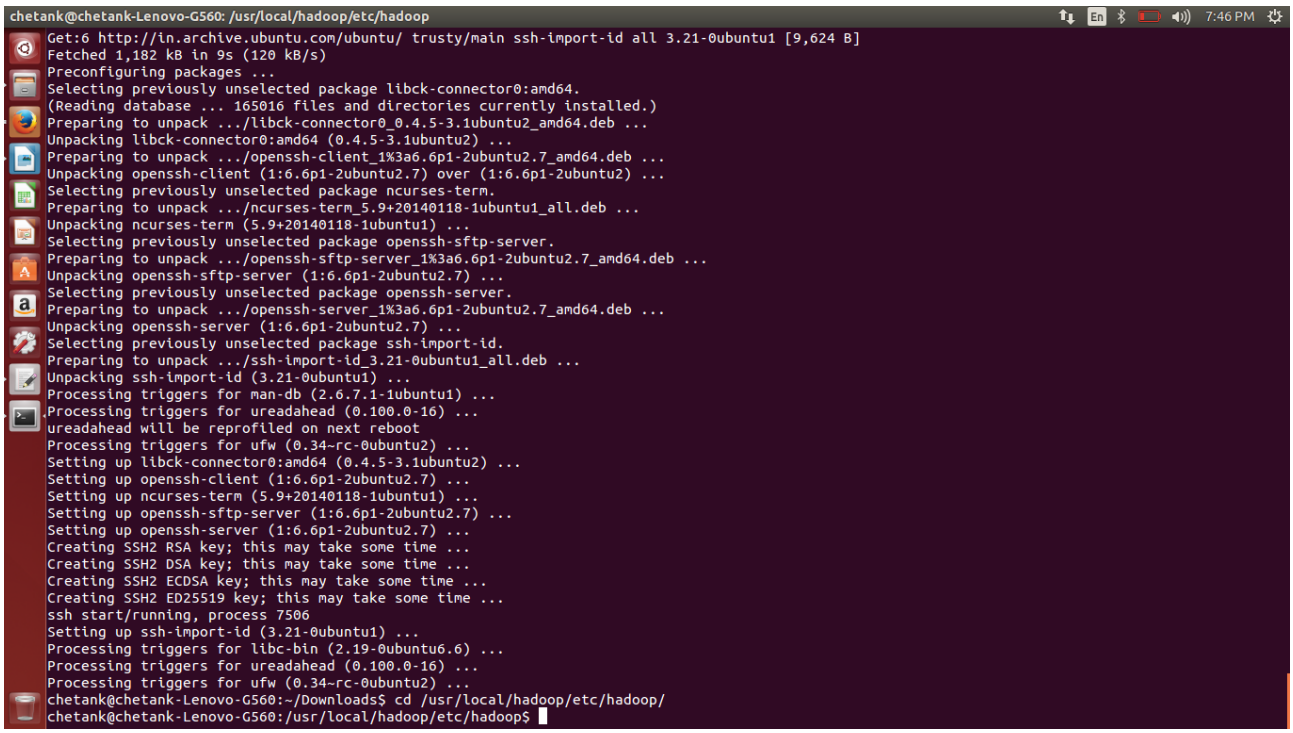
5) Installing SSH

SSH ("Secure SHell") is a protocol for securely accessing one machine from another. Hadoop uses SSH for accessing another slaves nodes to start and manage all HDFS and MapReduce daemons. (i.e Daemons – Build it processes of the application service)

$sudo apt-get install openssh-server



[Figure 5] Install openssh-server



[Figure 6] Install openssh-server

6) Switch to Hadoop User account

$ su – hduser

7) Configure Hadoop

To complete the setup of Hadoop, the following files will have to be modified:

1) ~/.bashrc
2) /usr/local/hadoop/etc/hadoop/hadoop-env.sh
3) /usr/local/hadoop/etc/hadoop/core-site.xml
4) /usr/local/hadoop/etc/hadoop/yarn-site.xml
5) /usr/local/hadoop/etc/hadoop/mapred-site.xml
6) /usr/local/hadoop/etc/hadoop/hdfs-site.xml

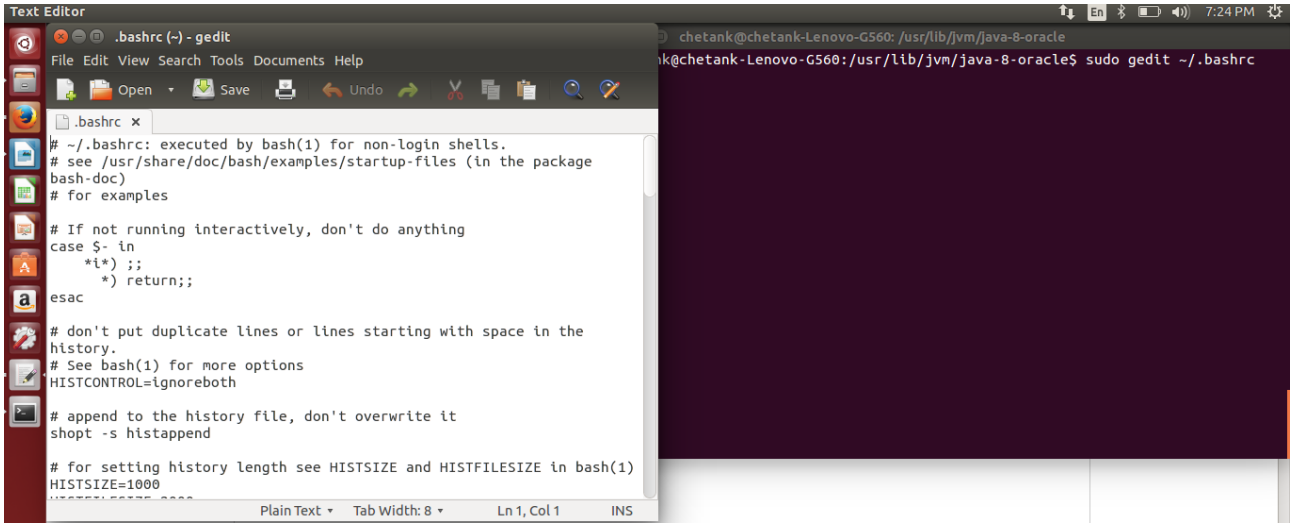
Let's start configuring one by one.

7.1) ~/.bashrc

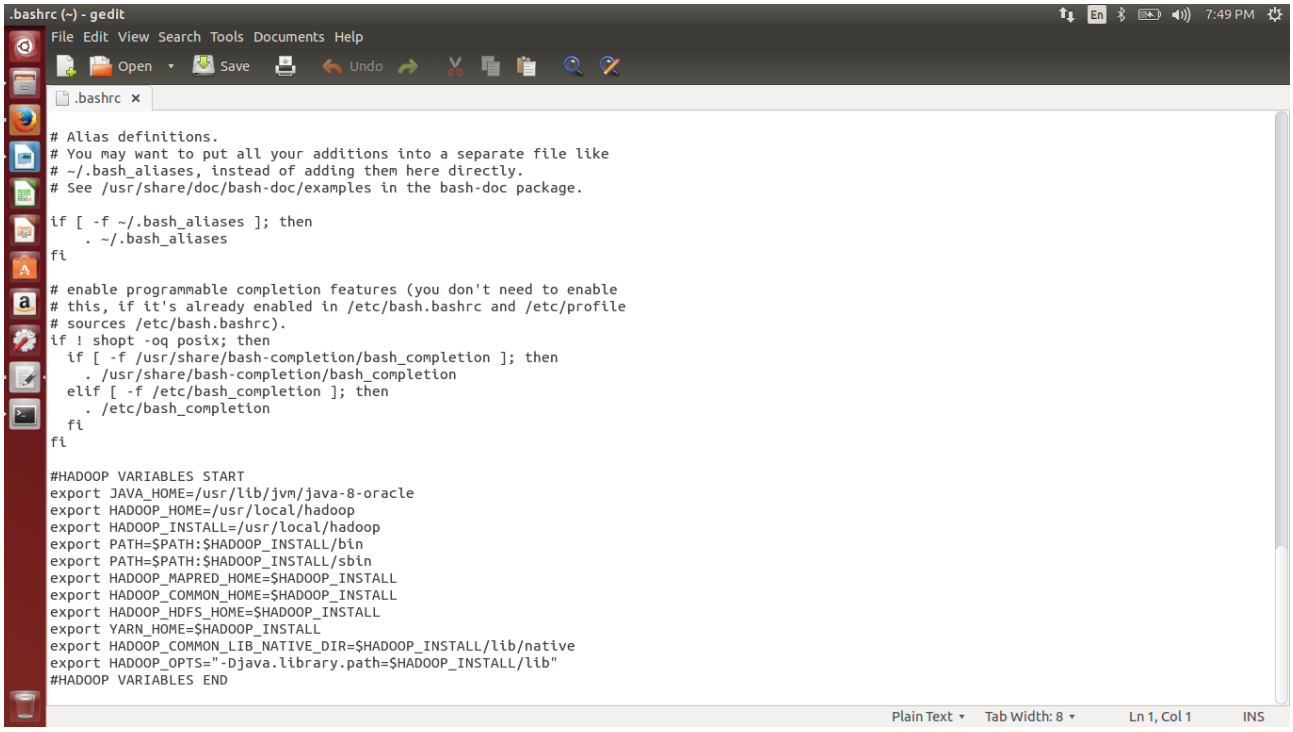
open the ~/.bashrc and paste the following at the end.Putting the above content in the .bashrc file ensures that these variables are always available when your VPS starts up.

#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL

```
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END
```



[Figure 7] Configure environment variable with ~/.bashrc



[Figure 8] Hadoop environment variables with ~/.bashrc

[Figure 9] Save environment variables with ~/.bashrc

7.2) Edit hadoop-env.sh

/usr/local/hadoop/etc/hadoop /hadoop-env.sh

Change JAVA_HOME variable into

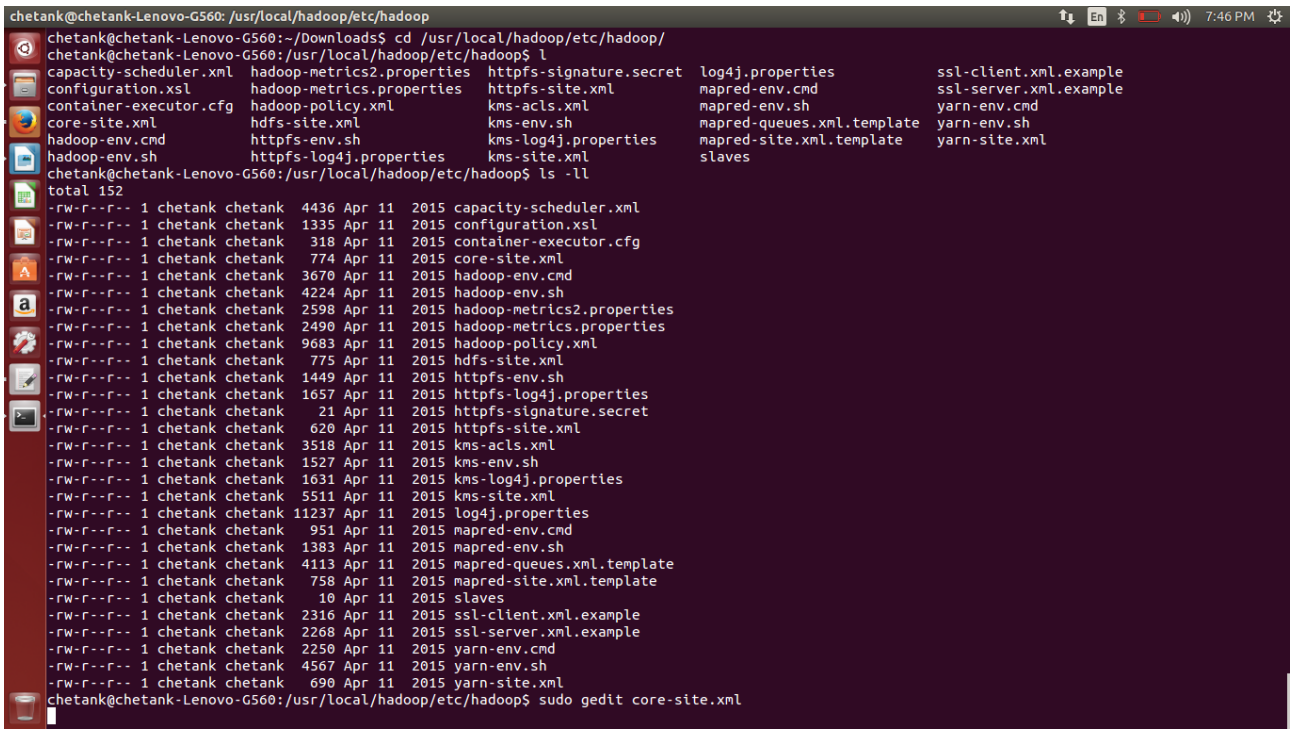
export JAVA_HOME=/usr/lib/jvm/java-8-oracle

7.3) core-site.xml

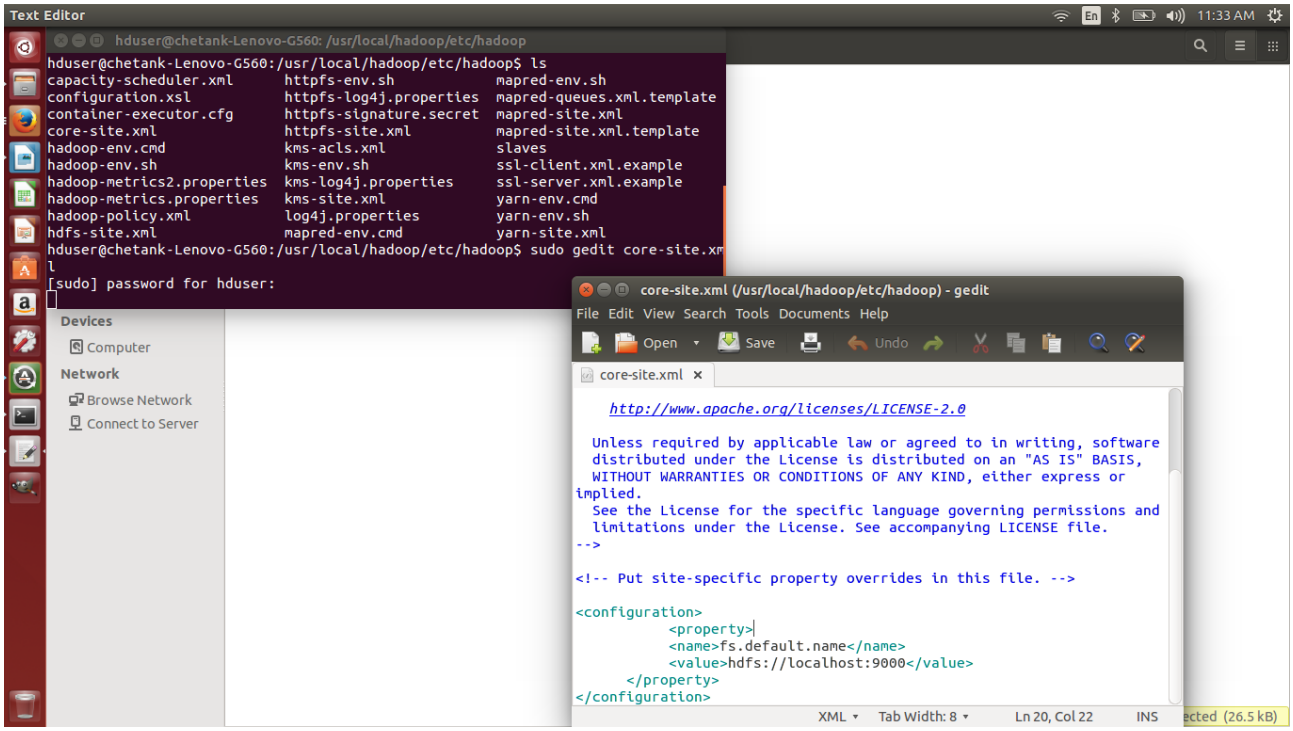
/usr/local/hadoop/etc/hadoop/core-site.xml

The /usr/local/hadoop/etc/hadoop/core-site.xml file contains configuration properties that Hadoop uses when starting up. This file can be used to override the default settings that Hadoop starts with. Enter the following content in between the tag <configuration></configuration>

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



[Figure 10] Opening core-site.xml from /usr/local/hadoop/etc/hadoop



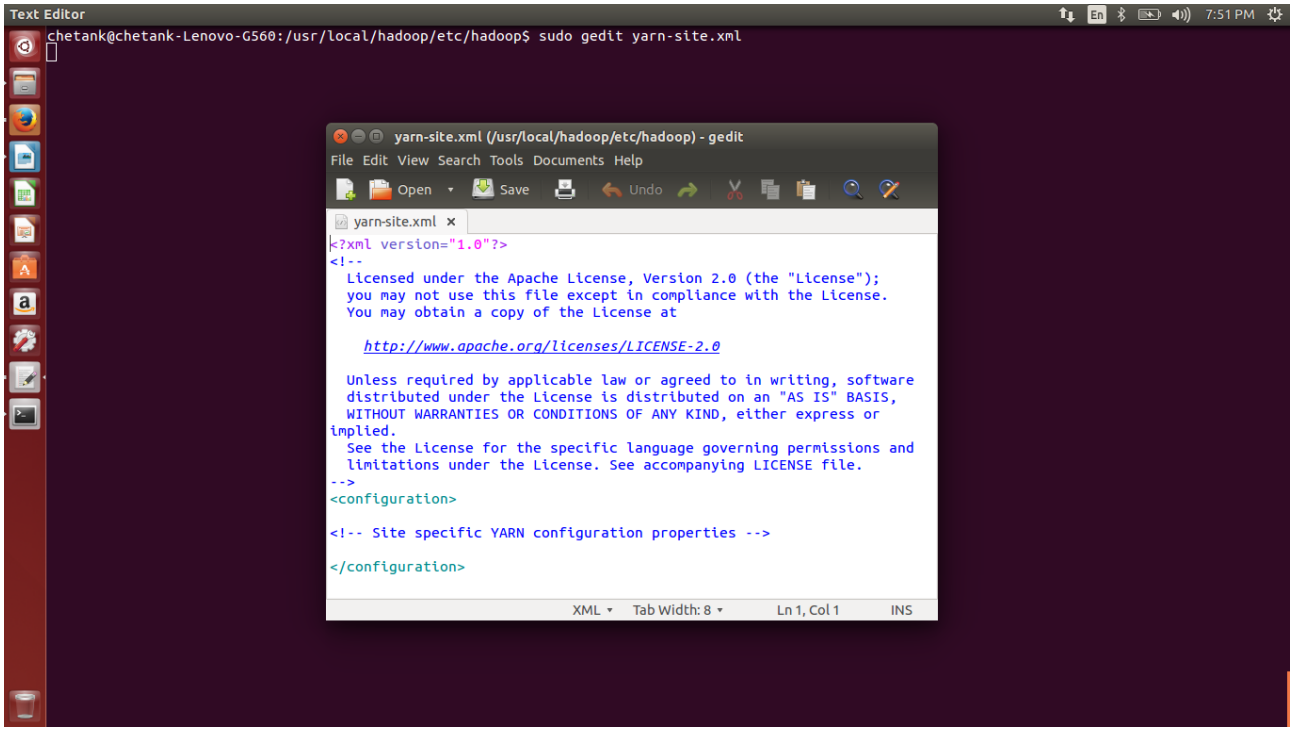
[Figure 11] Content of core-site.xml after adding property in configuration.

7. 4) yarn-site.xml

/usr/local/hadoop/etc/hadoop/yarn-site.xml

The /usr/local/hadoop/etc/hadoop/yarn-site.xml file contains configuration properties that MapReduce uses when starting up. This file can be used to override the default settings that MapReduce starts with. Enter the following content in between the tag <configuration></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>
```



[Figure 12] Opening yarn-site.xml from /usr/local/hadoop/etc/hadoop

7.5) mapred-site.xml

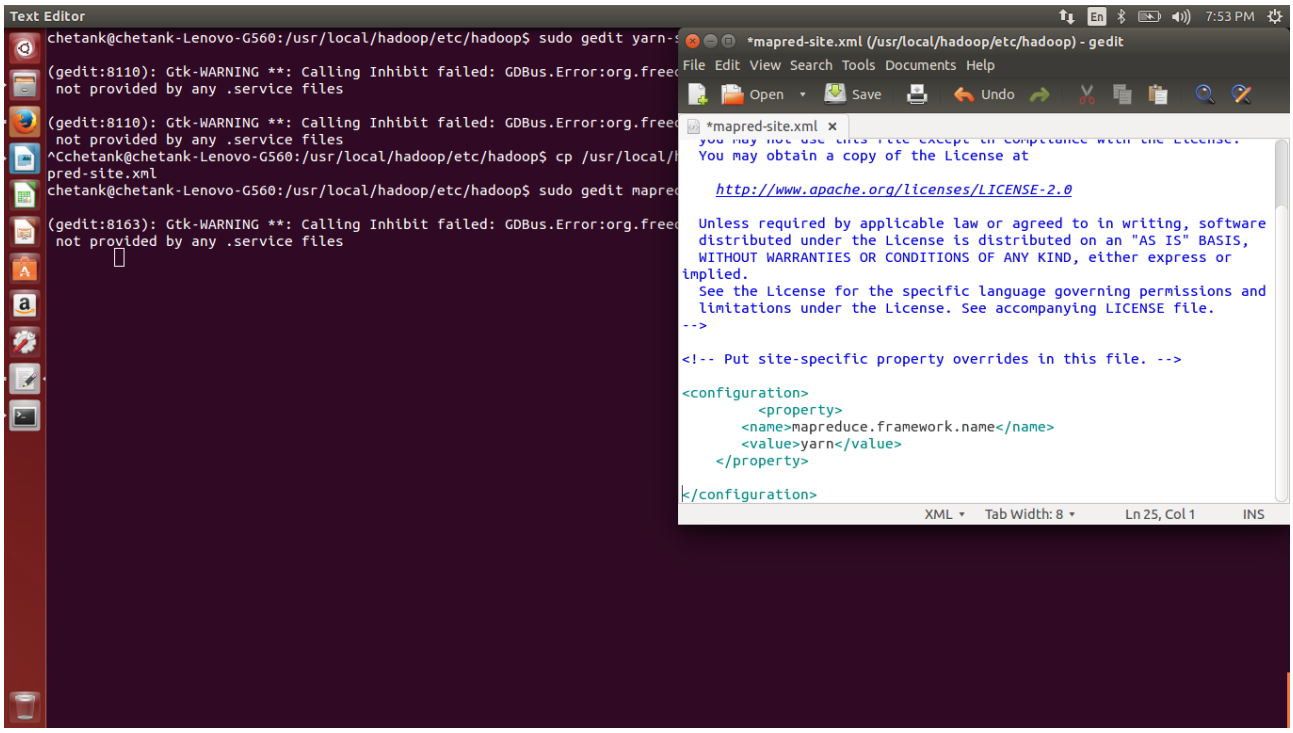
/usr/local/hadoop/etc/hadoop/mapred-site.xml

By default, the /usr/local/hadoop/etc/hadoop/ folder contains the /usr/local/hadoop/etc/hadoop/mapred-site.xml.template file which has to be renamed/copied with the name mapred-site.xml. This file is used to specify which framework is being used for MapReduce.

cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml

Then open /usr/local/hadoop/etc/hadoop/mapred-site.xml and enter the following content in between the tag <configuration></configuration>

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



[Figure 13] Configuration of mapred-site.xml

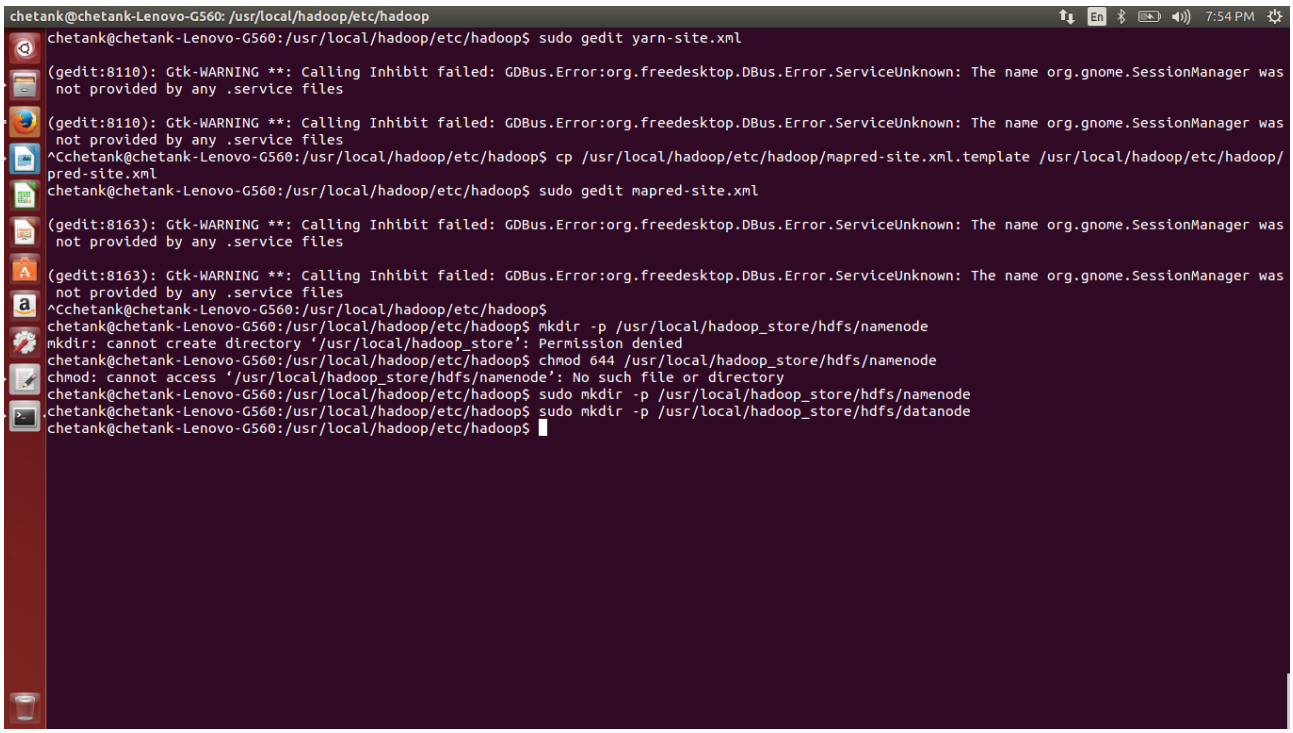
7.6) hdfs-site.xml

/usr/local/hadoop/etc/hadoop/hdfs-site.xml

The /usr/local/hadoop/etc/hadoop/hdfs-site.xml has to be configured for each host in the cluster that is being used. It is used to specify the directories which will be used as the namenode and the datanode on that host.

Before editing this file, we need to create two directories which will contain the namenode and the datanode for this Hadoop installation.

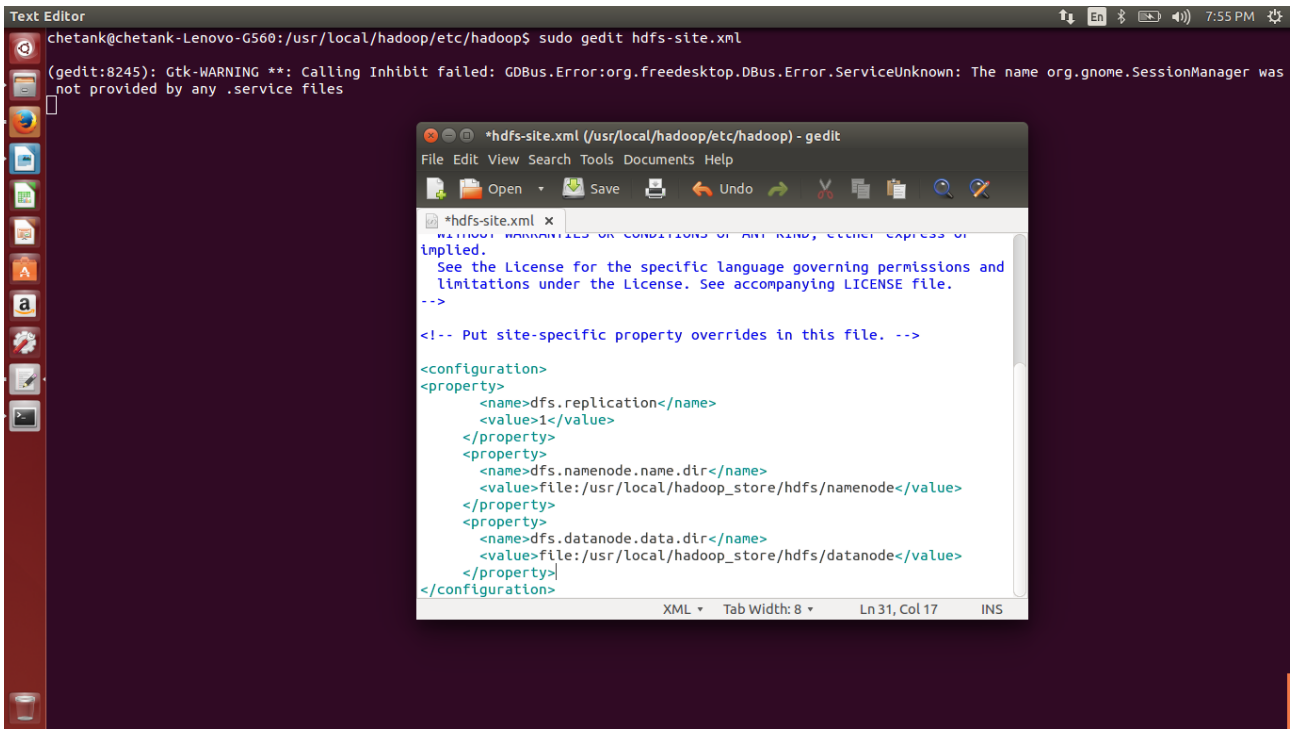
```
sudo mkdir -p /usr/local/hadoop_store/hdfs/namenode
sudo mkdir -p /usr/local/hadoop_store/hdfs/datanode
```



[Figure 14] Creation of namenode and datanode directories.

Now open /usr/local/hadoop/etc/hadoop/hdfs-site.xml and enter the following content in between the tag <configuration></configuration>

```
<property>
      <name>dfs.replication</name>
      <value>1</value>
</property>
<property>
      <name>dfs.namenode.name.dir</name>
      <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
</property>
<property>
      <name>dfs.datanode.data.dir</name>
      <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
</property>
```



[Figure 15] Configuration of Namenode and Datanode

Now Make sure that dir has right owner and permission /usr/local/hadoop_store.

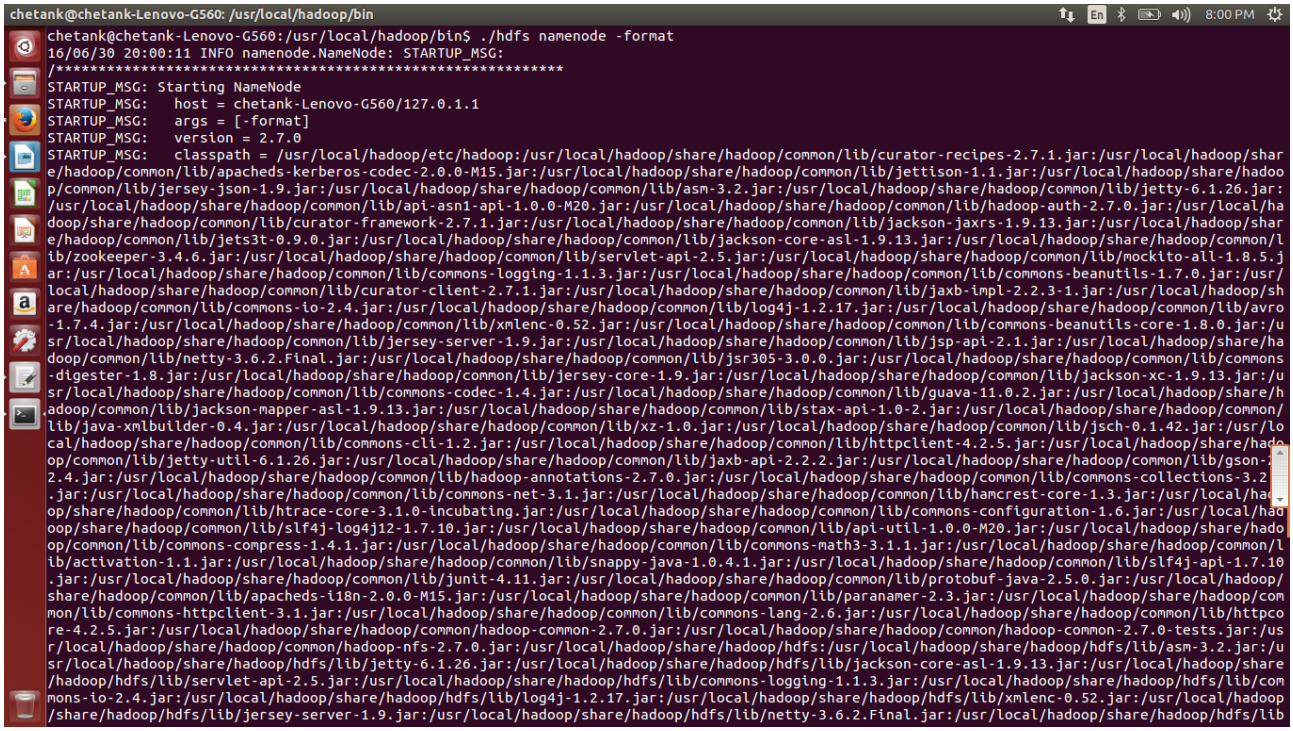
Sudo chown hduser:hadoop -R /usr/local/hadoop_store
sudo chmod 777 -R /usr/local/hadoop_store

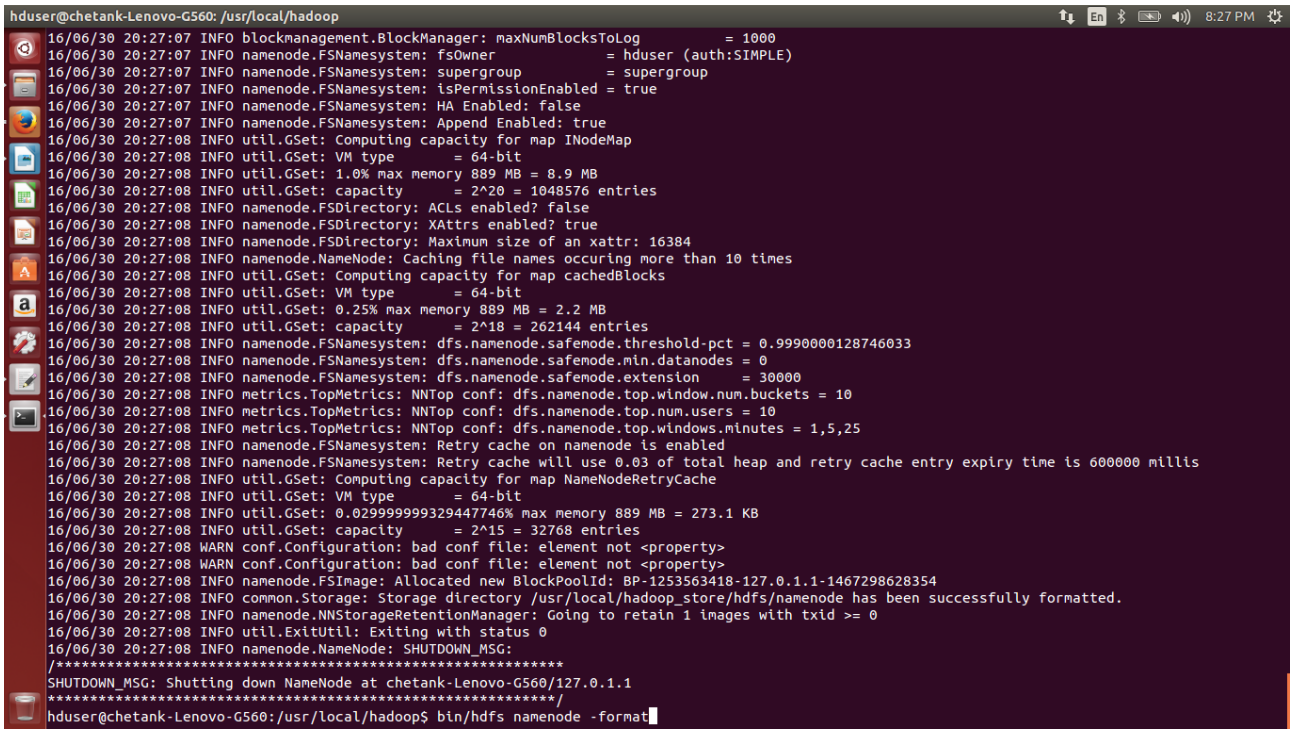
[Figure 16] Checking permission and owner for datanode and namenode directory.

7. 7) Format Namenode

cd /usr/local/hadoop/
bin/hdfs namenode -format

[Figure 17] Formatting namenode method 1

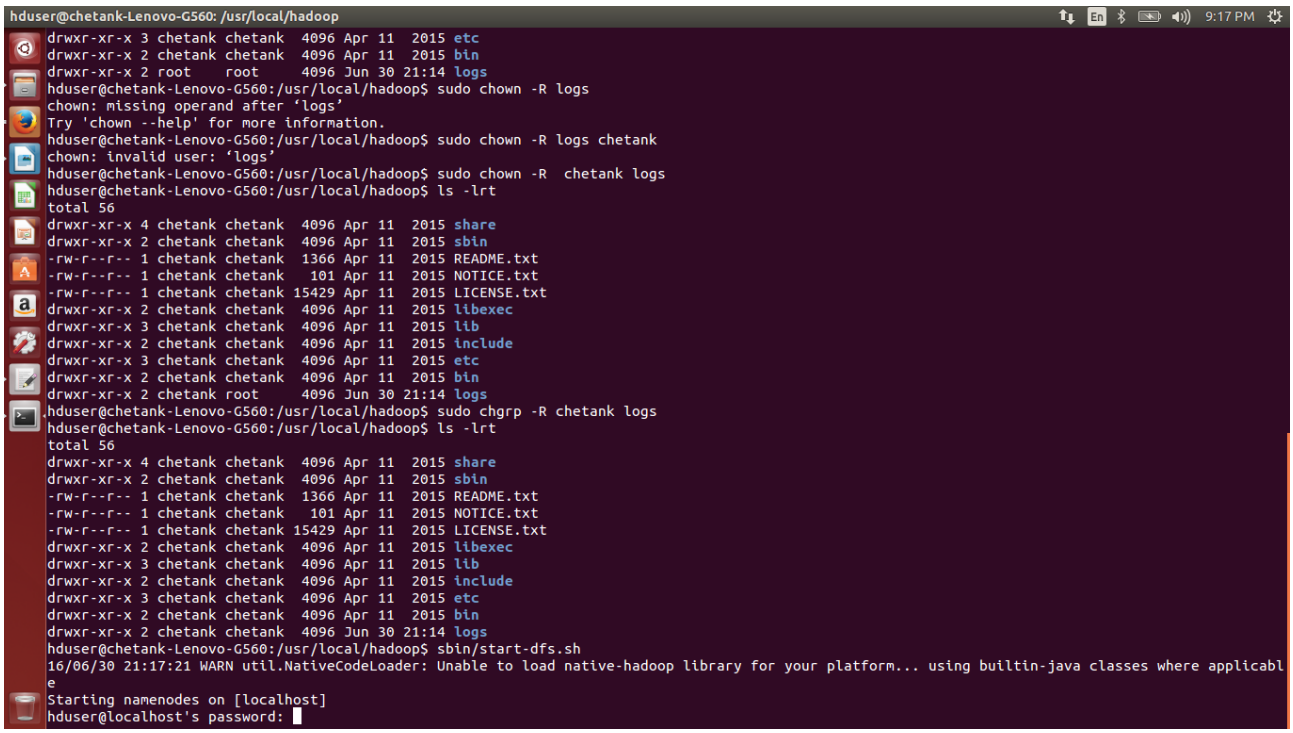




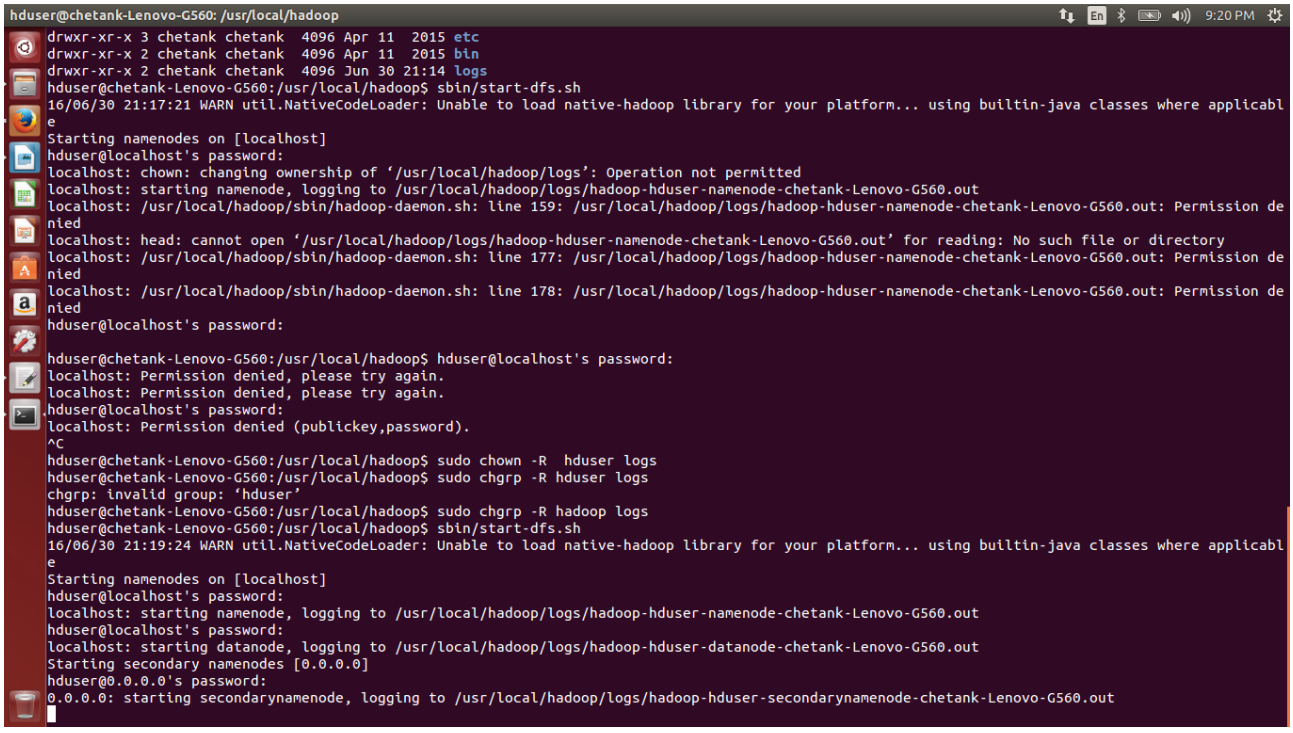
[Figure 18] Formatting namenode method 2

7.8) Start all Hadoop daemons

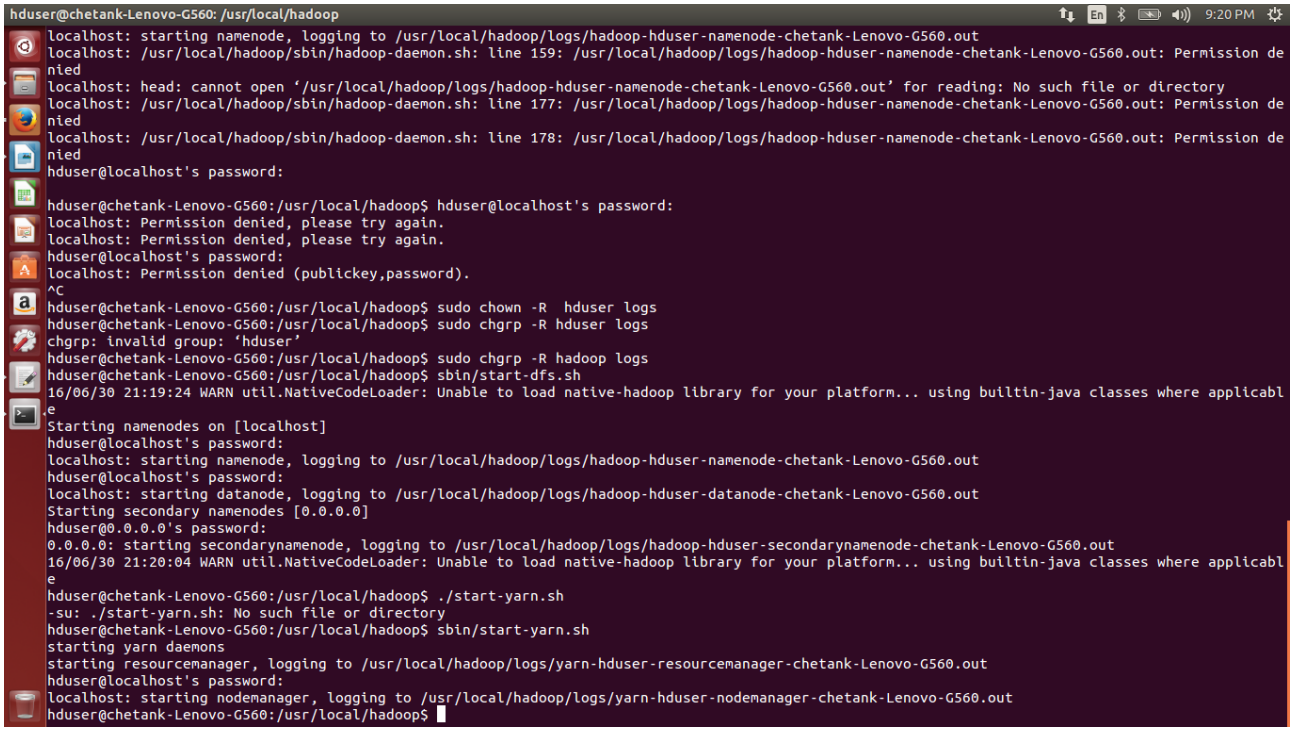
cd /usr/local/hadoop/
sbin/start-dfs.sh



[Figure 18] Starting hadoop deamon services - dfs



7.9) sbin/start-yarn.sh



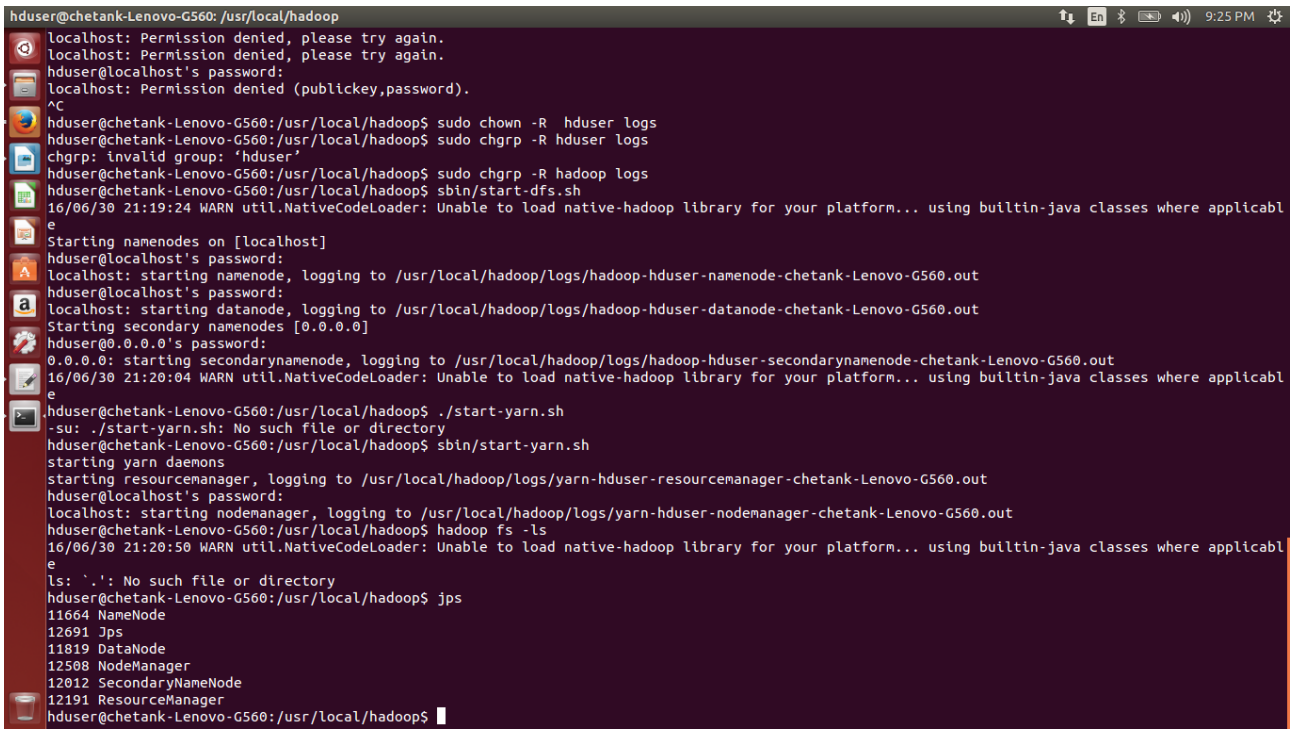
[Figure 19] Starting hadoop deamon services – yarn

Note: in single node cluster , there is no need to start yarn, but for learning purpose we can start yarn.

7.10) Check your hadoop Services

$ jps

If everything goes well, you will see below matrix.

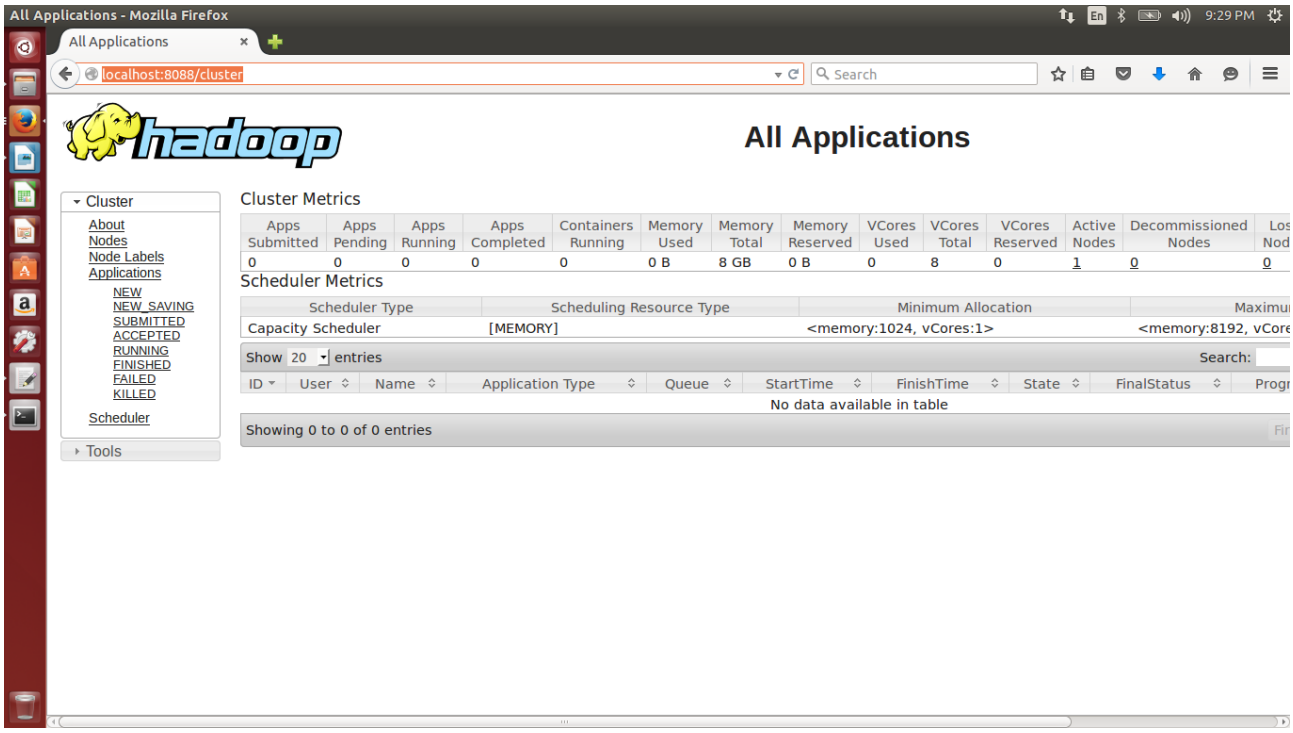


[Figure 20] Apache Hadoop Services status

7.11) Hadoop Cluster Applications UI

For Hadoop Cluster Applications UI, to check running jobs.

Open: http://localhost:8088
it will redirect to http://localhost:8088/cluster
where you can see matrix of running jobs.



[Figure 21] Apache Hadoop Cluster Application UI

Errors might occures:

1) while starting start-dfs.sh service, it might throws permission denieded at namenode/datanode-something.log

Solution:
mkdir /usr/local/hadoop/logs
sudo chown -R hduser /usr/local/hadoop/
sudo chgrp -R hadoop /usr/local/hadoop/

Note: namenode and datanode directories should be created under **hduser** user and **hadoop** group.

Read more:

[1] HDFS Architecture
Ref. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html

HDFS Acess and API

[2] HDFS Commands Guide
Ref. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HDFSCommands.html
[3] JAVA API
Ref. http://hadoop.apache.org/docs/current/api/org/apache/hadoop/fs/FileSystem.html
[4] C/libhdfs
Ref. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/LibHdfs.html
[5] WebHDFS API
Ref. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/WebHDFS.html