1. Set up a Ubuntu instance in ec2 with traffic in security group as everywhere
2. Ref doc- http://www.bogotobogo.com/Hadoop/BigData\_hadoop\_Install\_on\_ubuntu\_single\_node\_cluster.php
3. **sudo apt-get update - update the sources list**
4. sudo apt-get install default-jdk - Java installation
5. java –version
6. sudo addgroup Hadoop – add a Hadoop user
7. sudo adduser --ingroup hadoop hduser
8. sudo apt-get install ssh
9. which ssh
10. which sshd
11. login : su hduser
12. pwd :b12345
13. ssh-keygen -t rsa -P ""
14. cat $HOME/.ssh/id\_rsa.pub >> $HOME/.ssh/authorized\_keys
15. ssh localhost
16. tar xvzf hadoop-2.6.0.tar.gz
17. cd /usr/local
18. sudo mkdir hadoop
19. hduser is not in the sudoers file. This incident will be reported. (err)
20. sudo wget <http://mirrors.sonic.net/apache/hadoop/common/hadoop-2.6.0/hadoop-2.6.0.tar.gz>
21. sudo tar xvzf hadoop-2.6.0.tar.gz
22. logout ctrl+d till you get ubuntu@ip$
23. sudo visudo
24. below
25. root ALL=(ALL) ALL
26. add one with your username below that
27. hduser ALL=(ALL) ALL
28. cd /home/hduser/hadoop-2.6.0
29. [hduser@ip-172-31-18-65:~/hadoop-2.6.0$](mailto:hduser@ip-172-31-18-65:~/hadoop-2.6.0$)
30. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ cd /usr/local
31. hduser@ip-172-31-18-65:/usr/local$ sudo mkdir Hadoop
32. hduser@ip-172-31-18-65:/usr/local$ cd /home/hduser/hadoop-2.6.0
33. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo mv \* /usr/local/hadoop
34. sudo chown -R hduser:hadoop /usr/local/Hadoop
35. update-alternatives --config java (to confirm java location)
36. sudo nano ~/.bashrc
37. add the below block at the end of the file

#HADOOP VARIABLES START

export JAVA\_HOME=/usr/lib/jvm/java-7-openjdk-amd64

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

1. source ~/.bashrc
2. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ javac –version
3. which javac
4. readlink -f /usr/bin/javac
5. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
6. add export JAVA\_HOME=/usr/lib/jvm/java-7-openjdk-amd64
7. sudo mkdir -p /app/hadoop/tmp
8. sudo chown hduser:hadoop /app/hadoop/tmp
9. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo nano /usr/local/hadoop/etc/hadoop/core-site.xml
10. add the flowing block of code between configuration tags

<configuration>

<property>

<name>hadoop.tmp.dir</name>

<value>/app/hadoop/tmp</value>

<description>A base for other temporary directories.</description>

</property>

<property>

<name>fs.default.name</name>

<value>hdfs://localhost:54310</value>

<description>The name of the default file system. A URI whose

scheme and authority determine the FileSystem implementation. The

uri's scheme determines the config property (fs.SCHEME.impl) naming

the FileSystem implementation class. The uri's authority is used to

determine the host, port, etc. for a filesystem.</description>

</property>

</configuration>

1. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml
2. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo nano /usr/local/hadoop/etc/hadoop/mapred-site.xml
3. and add the block below within configuration tags

<configuration>

<property>

<name>mapred.job.tracker</name>

<value>localhost:54311</value>

<description>The host and port that the MapReduce job tracker runs

at. If "local", then jobs are run in-process as a single map

and reduce task.

</description>

</property>

</configuration>

1. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo mkdir -p /usr/local/hadoop\_store/hdfs/namenode
2. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo mkdir -p /usr/local/hadoop\_store/hdfs/datanode
3. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo chown -R hduser:hadoop /usr/local/hadoop\_store
4. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml
5. Insert the code below between the configuration tags

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

<description>Default block replication.

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.

</description>

</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>

</configuration>

1. hduser@ip-172-31-18-65:~/hadoop-2.6.0$ hadoop namenode –format
2. cd /usr/local/hadoop/sbin
3. start-all.sh
4. jps

11698 DataNode

12019 ResourceManager

11885 SecondaryNameNode

12435 Jps

12151 NodeManager

11541 NameNode

1. test netstat -plten | grep java
2. on the browser test <http://ec2-54-200-245-70.us-west-2.compute.amazonaws.com:50070/> web UI
3. test the pi program
4. hduser@ip-172-31-18-65:~$ hadoop jar /usr/local/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.6.0.jar pi 10 1000
5. run wordcount problem
6. move your users.tx(or any other to /home/Ubuntu via filezilla)
7. hduser@ip-172-31-18-65:~$ cp /home/ubuntu/users.txt /home/hduser
8. movinf to Hadoop file system
9. hduser@ip-172-31-18-65:~$ hadoop dfs -copyFromLocal users.txt
10. hduser@ip-172-31-18-65:~$ hadoop dfs -mkdir input
11. hduser@ip-172-31-18-65:~$ hadoop dfs -cp users.txt input
12. hduser@ip-172-31-18-65:~$ hadoop jar /usr/local/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.6.0.jar wordcount input output
13. hduser@ip-172-31-18-65:~$ hadoop dfs –ls ( to view all directories)
14. hduser@ip-172-31-18-65:~$ hadoop dfs -cat output/part-r-00000
15. running mapper.py & reducer.py
16. ftp the quakes.csv file to /home/Ubuntu via filezilla
17. from putty
18. hduser@ip-172-31-18-65:~$ cp quakes.csv /home/hduser
19. hduser@ip-172-31-18-65:~$ hadoop dfs -mkdir input1
20. hduser@ip-172-31-18-65:~$ hadoop dfs -copyFromLocal quakes.csv
21. hduser@ip-172-31-18-65:~$ hadoop dfs -cp quakes.csv input1
22. hduser@ip-172-31-18-65:~$ hadoop dfs -ls input1

Found 1 items

-rw-r--r-- 1 hduser supergroup 1788428 2016-06-19 21:05 input1/quakes.csv

1. write the mapper.py & reducer.py programs in /home/hduser folder
2. hduser@ip-172-31-18-65:~$ sudo nano mapper.py
3. hduser@ip-172-31-18-65:~$ sudo cp mapper.py python
4. hduser@ip-172-31-18-65:~$ sudo cp reducer.py python
5. hduser@ip-172-31-18-65:~$ chmod 0777 python
6. hduser@ip-172-31-18-65:/usr/local/hadoop$ bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -file /home/hduser/python/mapper.py -mapper /home/hduser/python/mapper.py -file /home/hduser/python/reducer.py -reducer /home/hduser/python/reducer.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output1
7. hduser@ip-172-31-18-65:/usr/local/hadoop$ hadoop dfs -cat output8/part-00000
8. hduser@ip-172-31-18-65:/usr/local/hadoop$ hadoop dfs -copyToLocal output8/part-00000
9. ubuntu@ip-172-31-18-65:/usr/local/hadoop$ sudo cat part-00000 > /home/ubuntu/quakesmr.csv
10. from filezilla do ftp

Visualisation

1. Notepad ++ get the html file
2. Edit the file with your csv file
3. Run it on local for which both files should be in same folder
4. <file:///D:/d3js-Data-Visualization-master/d3js-Data-Visualization-master/index.html> in web browser mozilla
5. Go to putty
6. Create a new ec2 instance and set up a Ubuntu user
7. Rename index.html as /var/www/html
8. Follow <http://www.datasciencebytes.com/bytes/2015/02/24/running-a-flask-app-on-aws-ec2/>
9. In my case ran on old ec2-asn3 instance
10. So refer vis-had.txt on desktop
11. sudo cp quakesmr.csv /var/www/html and sudo mv index.html /var/www/html
12. `same folder and should be able to access the graph on
13. <http://ec2-54-201-235-70.us-west-2.compute.amazonaws.com/>

Show OUTPUT

1. Go to putty
2. Su hduser
3. B12345
4. Cd /usr/local/Hadoop //change the location
5. bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -file /home/hduser/python/mapper.py -mapper /home/hduser/python/mapper.py -file /home/hduser/python/reducer.py -reducer /home/hduser/python/reducer.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output\*\*\*
6. output directory should not be opened and should be empty
7. hadoop dfs -cat output\*\*/part-00000
8. hduser@ip-172-31-18-65:~$ hadoop dfs -copyToLocal output10/part-00000 /home/hduser
9. ubuntu@ip-172-31-18-65:/home/hduser$ sudo mv part-00000 /home/ubuntu/test.csv

To open mapper/reducer.py

1. ubuntu@ip-172-31-18-65:/home/hduser$ su hduser
2. hduser@ip-172-31-18-65:~$ sudo nano /home/hduser/python/mapper.py
3. to save ctrl-o
4. to exit ctrl-X //for nano editor cd /home/hduser/python

SUPPOSE WE NEED TO GIVE SOME VALUES AND RUN ON COMMAND PROMPT,THEN USE

1. hduser@ip-172-31-18-65:/usr/local/hadoop$ bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -file /home/hduser/python/mapper1.py -mapper '/home/hduser/python/mapper1.py 3' -file /home/hduser/python/reducer1.py -reducer /home/hduser/python/reducer1.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output31
2. hduser@ip-172-31-18-65:/usr/local/hadoop$ hadoop dfs -cat output31/part-00000
3. do soo from 1 to 15

**code for mapper // earthquakes**

#!/usr/bin/python

import sys

input=int(sys.argv[1])

# input comes from STDIN (standard input)

for line in sys.stdin:

line = line.strip()

line = line.split(",")

if len(line)>0:

print '%s\t%s' % (line[0].split('T')[0], line[input])

STARTING OF QUIZ

1. given csv file
2. login to hduser
3. move your haduser.csv(or any other to /home/Ubuntu via filezilla)
4. hduser@ip-172-31-18-65:~$ cp /home/ubuntu/haduser.csv /home/hduser
5. movinf to Hadoop file system
6. hduser@ip-172-31-18-65:~$ hadoop dfs -copyFromLocal haduser.csv
7. hduser@ip-172-31-18-65:~$ hadoop dfs -mkdir input3
8. hduser@ip-172-31-18-65:~$ hadoop dfs -cp haduser.csv input3
9. hduser@ip-172-31-18-65:/home/ubuntu$ hadoop dfs –ls
10. hduser@ip-172-31-18-65:/home/ubuntu$ cd /home/hduser/python
11. hduser@ip-172-31-18-65:~/python$ ls
12. hduser@ip-172-31-18-65:~/python$ cp mapper.py mapper1.py
13. hduser@ip-172-31-18-65:~/python$ cp reducer.py reducer1.py
14. hduser@ip-172-31-18-65:~/python$ sudo nano mapper1.py
15. bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -file /home/hduser/python/mapper.py -mapper /home/hduser/python/mapper.py -file /home/hduser/python/reducer.py -reducer /home/hduser/python/reducer.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output\*\*\*
16. output directory should not be opened and should be empty
17. hadoop dfs -cat output\*\*/part-00000
18. hduser@ip-172-31-18-65:~$ hadoop dfs -copyToLocal output10/part-00000 /home/hduser
19. ubuntu@ip-172-31-18-65:/home/hduser$ sudo mv part-00000 /home/ubuntu/test.csv
20. hduser@ip-172-31-18-65:~/python$ python hadooptask.py

////// hduser@ip-172-31-18-65:~/python$ sudo nano hadooptask.py

MULTIPLE MAPPERS AND REDUCERS

1. hduser@ip-172-31-18-65:~/python$ /usr/local/hadoop/bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -D mapred.map.tasks=2 -D mapred.reduce.tasks=4 -file /home/hduser/python/mapper1.py -mapper '/home/hduser/python/mapper1.py 3' -file /home/hduser/python/reducer1.py -reducer /home/hduser/python/reducer1.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output50
2. hduser@ip-172-31-18-65:~/python$ /usr/local/hadoop/bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.0.jar -D mapred.map.tasks=2 -D mapred.reduce.tasks=4 -file /home/hduser/python/mapper1.py -mapper '/home/hduser/python/mapper1.py 3' -file /home/hduser/python/reducer1.py -reducer /home/hduser/python/reducer1.py -input /user/hduser/input1/quakes.csv -output /user/hduser/output51

//mapper and reducer above

WEB APPLICATION

1. send the csv file in filezilla from local to /home/Ubuntu
2. flaskapp.py rename and send the new one and send the index.html rename in the template
3. in putty connect to asn3 and run
4. sudo apachectl restart
5. go to web browser and run
6. <http://ec2-54-201-235-70.us-west-2.compute.amazonaws.com/>
7. Programs are in C:\Users\Brinda\PycharmProjects\hadoopweb… for editing

CHECKING LOGS

http://ec2-54-200-245-70.us-west-2.compute.amazonaws.com:50070