Hadoop Multi-Node cluster Installation on GCP

**First we have to create instance:-**

* Click on navigation bar => compute engine => vm instances

Create a instance 1 called **namenode-1**

**machine type** 2vcpus

Os ubuntu 16

Click on create  an instance will be created

Same way we have to create another 2 instances as datanode-1 and datanode-2

Datanode-1 and datanode-2

Machine type: 1cpu with 3.5gb

* You can see the all the instances which are created in vm instances

**Now** from name node instances in click on ssh and select open in browser window

Then the terminal will be opened for namenode-1

From namenode-1 shell login to root user using command

**sudo -i**

**Then create a user using command**

**adduser hduser**

**Set password as hduser**

**=>edit the following file**

**$ nano /etc/ssh/sshd\_config**

**add /update following lines**

PasswordAuthentication yes

ChallengeResponseAuthentication yes

Now we have to restart the sshd service to update changes using command

**service sshd restart**

**Now we have to configure hosts file**

**$ nano etc/hosts**

This will open name node hosts file with its fully qualified domain and internal ip address

In this we have to add datanode-1 and datanode-2 fully qualified domain with ip address info

We have to add

We can check fully qulified domin name using command

$ hostname -f

$hostname -i -----> to get ip address info

We have to this in namenode-1 host file and datanode-1 and datanode-2 host files also

The above steps should be repeated in datanode-1 and datanode-2.

At last before configuring hadoop open namenode-1 terminal and login to hduser using command  **$ su hduser**

**$ cd ~**

**Execute the following command**

**$  ssh-keygen -t rsa -P “”**

The above command will generate a keys

Now we have to copy the keys to datanode-1 and 2

**$ ssh-copy-id-i  /home/hduser/.ssh/id\_rsa.pub hduser@datanode-1**

**$ ssh-copy-id-i  /home/hduser/.ssh/id\_rsa.pub hduser@datanode-2**

**$ ssh-copy-id-i  /home/hduser/.ssh/id\_rsa.pub hduser@namenode-1**

**Now we have to give read/write permissions**

**$ chmod 0600 ~/.ssh/authorized\_keys**

**Now its time to create hadoop clustre on newly created instances:----**

**Step 1**: Download java and Hadoop

1. Get into the namenode -1 instance

2. Switch as root user using the command

sudo -i

3. Download java1.8 and Hadoop 2.7 by using the below command

cd /opt/

wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" http://download.oracle.com/otn-pub/java/jdk/8u131-b11/d54c1d3a095b4ff2b6607d096fa80163/jdk-8u131-linux-x64.tar.gz

wget <https://archive.apache.org/dist/hadoop/core/hadoop-2.7.0/hadoop-2.7.0.tar.gz>

**Step2: Install Java**

1. Extract java tar file and configure java using the below command

tar xzf jdk-8u131-linux-x64.tar.gz

cd /opt/jdk1.8.0\_131

alternatives --install /usr/bin/java java /opt/jdk1.8.0\_131/bin/java 2

 -----*if you get error like alternatives command not found when you run above command then you run command java version you get some versions which are yet to be installed so using command* ***apt install java headless 8 version***  *select versions 8 from options--------*

*sudo update-alternatives --config java*

#alternatives --config java

java -version

**Step3: Configure Hadoop**

1. Extract Hadoop tar file and update Hadoop configuration files using the below command

cd /opt/

tar xzf hadoop-2.7.0.tar.gz

mv hadoop-2.7.0 /usr/local/hadoop

chown -R hduser:hduser /usr/local/hadoop

mkdir -p /usr/local/hadoop\_store/tmp

mkdir -p /usr/local/hadoop\_store/hdfs/namenode

mkdir -p /usr/local/hadoop\_store/hdfs/datanode

mkdir -p /usr/local/hadoop\_store/hdfs/secondarynamenode

chown -R hduser:hduser /usr/local/hadoop\_store

su hduser

2. Edit hdfs-site.xml file and add the given configuration

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

<configuration>

<property>

<name>dfs.replication</name>

<value>3</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>

<property>

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

<value>file:/usr/local/hadoop\_store/hdfs/secondarynamenode</value>

</property>

<property>

<name>dfs.namenode.checkpoint.period</name>

<value>3600</value>

</property>

</configuration>

3. Edit core-site.xml file and add the given configuration

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

<configuration>

<property>

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

<value>/usr/local/hadoop\_store/tmp</value>

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

</property>

<property>

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

<value>hdfs://namenode-1:54310</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>

4. Edit map-site.xml file and add the given configuration

vi /usr/local/hadoop/etc/hadoop/map-site.xml

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

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

<value>namenode-1:54311</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>

5. Edit yarn-site.xml file and add the given configuration

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

<configuration>

<!-- Site specific YARN configuration properties -->

<property>

<name>yarn.resourcemanager.hostname</name>

<value>namenode-1</value>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

</configuration>

6. Add java path in hadoop-env.sh file

echo 'export JAVA\_HOME=/opt/jdk1.8.0\_131' >> /usr/local/hadoop/etc/hadoop/hadoop-env.sh

**Step4: Configure environment variable**

1. To run Hadoop services, we need to add hadoop and java path in the .basrhc file

su hduser

vi /home/hduser/.bashrc

export HADOOP\_PREFIX=/usr/local/hadoop

export HADOOP\_HOME=/usr/local/hadoop

export HADOOP\_MAPRED\_HOME=${HADOOP\_HOME}

export HADOOP\_COMMON\_HOME=${HADOOP\_HOME}

export HADOOP\_HDFS\_HOME=${HADOOP\_HOME}

export YARN\_HOME=${HADOOP\_HOME}

export HADOOP\_CONF\_DIR=${HADOOP\_HOME}/etc/hadoop

# Native Path

export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=${HADOOP\_PREFIX}/lib/native

export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_PREFIX/lib"

export PATH=$PATH:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin

export JAVA\_HOME=/opt/jdk1.8.0\_131

export JRE\_HOME=/opt/jdk1.8.0\_131/jre

export PATH=$PATH:/opt/jdk1.8.0\_131/bin:/opt/jdk1.8.0\_131/jre/bin

**Step5: Copy the Java tar file and Hadoop folder into another 2 instances**

1. Copy java tar file into another 2 instances

su hduser

cd /opt/

scp jdk-8u131-linux-x64.tar.gz datanode-1:/home/hduser/

scp jdk-8u131-linux-x64.tar.gz datanode-2:/home/hduser/

2. Copy Hadoop folder into another 2 instances

cd /usr/local

scp -r hadoop datanode-1:/home/hduser/

scp -r hadoop\_store datanode-1:/home/hduser/

scp -r /home/hduser/.bashrc datanode-1:/home/hduser/

scp -r /usr/local/hadoop datanode-2:/home/hduser/

scp -r /usr/local/hadoop\_store datanode-2:/home/hduser/

scp -r /home/hduser/.bashrc datanode-2:/home/hduser/

**Step6: Get into another 2 instances , move folders in respective path and install java**

1. Create and give permissions to the folders for Hadoop disk storage and processing

sudo -i

mv /home/hduser/jdk-8u131-linux-x64.tar.gz /opt/

mv /home/hduser/hadoop /usr/local/

mv /home/hduser/hadoop\_store /usr/local/

chown -R hduser:hduser /usr/local/hadoop

chown -R hduser:hduser /usr/local/hadoop\_store

2. Extract java tar file and configure java using the below command

cd /opt

tar xzf jdk-8u131-linux-x64.tar.gz

cd jdk1.8.0\_131

apt install openjdk-8-jre-headless

alternatives --install /usr/bin/java java /opt/jdk1.8.0\_131/bin/java 2

alternatives --config java

Note: Do step6 in both datanode-1 and datnode-2

**Step7: Format namenode and start all Hadoop services**

1. Add the datanodes instance name in the slaves file

vi /usr/local/hadoop/etc/hadoop/slaves

namenode-1

datanode-1

datanode-2

2. Get into the namenode instance and format namenode

su hduser

hadoop namenode -format

3. Start storage and yarn services

start-dfs.sh

start-yarn.sh