Experiment 2: HDFS Commands

2.1 Hadoop3.2.1 installation steps on Ubuntu 20.04 LTS

sudo apt remove openjdk-11*(if needed not mandatory)

Step 1: Installation of openJDK-8

sudo apt install openjdk-8-jdk openjdk-8-jre java –version (to check java version)

Install vim editor and opensshserver and client by giving below command

sudo apt install vim openssh-server openssh-client

Step 2: Adding the Jdk path to the path variable Find the java path by giving below commands whereis java

which java

readlink -f /usr/bin/javac

Open ~/.bashrc and add the jdk path

$ sudo vim ~/.bashrc

#go to the last line and add the following

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64 export PATH=$PATH:$JAVA_HOME

or

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64 export PATH=$PATH:/usr/lib/jvm/java-8-openjdk-amd64 ##save and exit

Inform the OS about the modification

$ source ~/.bashrc

Give the below commands and check whether the path has been set right or not

$ echo $JAVA_HOME

$ echo $PATH

Step 3: Add a dedicated user for the HADOOP

sudo adduser hadoop

sudo usermod -aG sudo hadoop

Step 4: Once the user is added, login to the user "Hadoop" to generate the ssh key for passwordless login (hadoop@machinename)

$ sudo su -hadoop

$ ssh-keygen -t rsa

$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

$ chmod 0600 ~/.ssh/authorized_keys

Check the login to localhost using ssh is valid

$ ssh localhost

IMPORTANT once the connection is made, logout from ssh

$ exit

Step 5: Download the latest binary from Hadoop site " hadoop-3.2.1.tar.gz "

Extract in downloads using extract here

From the same window got to new terminal and give the following command

$sudo mv hadoop-3.2.1 /usr/local/hadoop

$sudo vim /etc/profile.d/hadoop_java.sh

it will open an empty file add the below contents in the file export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64 export HADOOP_HOME=/usr/local/hadoop

export HADOOP_HDFS_HOME=$HADOOP_HOME export HADOOP_MAPRED_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native

export PATH=$PATH:$JAVA_HOME/bin:$HADOOP_HOME/bin:$HADOOP_HOME/sbin export HADOOP_OPTS="$HADOOP_OPTS -Djava.library.path=$HADOOP_HOME/lib/native"

Step 6 : now come back to the hadoop terminal

$source /etc/profile.d/hadoop_java.sh

$echo $HADOOP_HOME

$echo $PATH

$hadoop version

$hdfs version

now we have to give the ownership of the hadoop folder to the hadoop user now

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

get into /usr/local/hadoop

Specify JAVA_HOME in hadoop-env.sh (/usr/local/hadoop/etc/hadoop)

cd etc ls

cd hadoop ls

$ vim hadoop-env.sh

Add the following line in java implementation

export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64 (54 line) Save and exit

Change ur current working director to /usr/local/hadoop

& create 2 directories htemp and hdfs and 2 subdirectories namenode and datanode inside hdfs

mkdir htemp

mkdir hdfs

cd hdfs

mkdir datanode namenode

ls

$ sudo chown -R hadoop:hadoop /usr/local/hadoop/hdfs

$ sudo chown -R hadoop:hadoop /usr/local/hadoop/htemp

Step 7: Modify core-site.xml to setup web portal for hadoop

cd .. ls

cd etc/hadoop ls

vim core-site.xml

Add the following lines to it

<configuration>

<property>

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

<value>hdfs://localhost:9000</value>

<description>The default file system URI</description>

</property>

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

<value>/usr/local/hadoop/htemp</value>

</property>

</configuration>

Step 8: Modify hdfs-site.xml to setup namenode and datanode path and replication factor

vim hdfs-site.xml

Modify hdfs-site.xml and add the following lines inside

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

<value>file:/usr/local/hadoop/hdfs/namenode</value>

</property>

<property>

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

<value>file:/usr/local/hadoop/hdfs/datanode</value>

</property>

</configuration>

Step 9: Configure the mapreduce framework by editing the mapred-site.xml Modify the mapred-site.xml and add the following lines

<configuration>

<property>

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

</property>

<property>

<name>mapreduce.application.classpath</name>

<value>$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/*:$HADOOP_MAPRED_
HOME/share/hadoop/mapreduce/lib/*</value>

</property>

</configuration>

Step 10: Configure the YARN resource manager by editing the yarn-site.xml

<configuration>

<property>

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

<value>mapreduce_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.env-whitelist</name>

<value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP

_CONF_DIR,CLASSPATH_PREPEND_DISTCACHE,HADOOP_YARN_HOME,HADOO

P_MAPRED_HOME</value>

</property>

</configuration>

Step 11 :namenode format

hdfs namenode –format

Step 12 : to start all hadoop daemon process

Change path to /usr/local/hadoop/sbin

./start-all.sh

jps