\*Hadoop and Vm ware installation\*

- First VM ware player download and then install.

- Take >=4GB RAM for VM ware player.

- After ubuntu download (version as per our requirement)

- Create New virtual machine in VM ware player & install ubuntu OS.

# How to install ubuntu OS.

1. In New virtual machine browse ubuntu any drive, except C drive.

2. Click 'Next'.

3. Create Username and Password.(Username- hduser)

4. Click 'Next'.

5. Take upto 40gb memory to ubuntu.

6. Click 'Next'.

7. After Click 'Finish'.

8. Then your ubuntu OS will install in few minutes.

- When display ubuntu window then open terminal window by using right click.

- Then check terminal window is running or not by using different linux command like- 'pwd','cd' etc.

- To check the ubuntu version this "lsb\_release-a" command is used.

Installing Java

Hadoop framework is written in Java!!

user1@localhost:~$ cd ~

pwd

# Update the source list

user@laptop(local directory):~$ sudo apt-get update

# The OpenJDK project is the default version of Java

# that is provided from a supported Ubuntu repository.

user@laptop:~$ sudo apt-get install default-jdk

user@laptop:~$ java -version

java version "1.8.0\_65"

OpenJDK Runtime Environment (IcedTea 2.5.3) (7u71-2.5.3-0ubuntu0.14.04.1)

OpenJDK 64-Bit Server VM (build 24.65-b04, mixed mode)

Adding a dedicated Hadoop user

user@laptop:~$ sudo addgroup hadoop

Adding group `hadoop' (GID 1002) ...

Done.

user@laptop:~$ 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

Installing SSH

ssh has two main components:

1. ssh : The command we use to connect to remote machines - the client.

2. sshd : The daemon that is running on the server and allows clients to connect to the server.

The ssh is pre-enabled on Linux, but in order to start sshd daemon, we need to install ssh first. Use this command to do that :

user@laptop:~$ sudo apt-get install ssh

This will install ssh on our machine. If we get something similar to the following, we can think it is setup properly:

user@laptop:~$ which ssh

/usr/bin/ssh

user@laptop:~$ which sshd

/usr/sbin/sshd

Create and Setup SSH Certificates

Hadoop requires SSH access to manage its nodes, i.e. remote machines plus our local machine. For our single-node setup of Hadoop, we therefore need to configure SSH access to localhost.

So, we need to have SSH up and running on our machine and configured it to allow SSH public key authentication.

Hadoop uses SSH (to access its nodes) which would normally require the user to enter a password. However, this requirement can be eliminated by creating and setting up SSH certificates using the following commands. If asked for a filename just leave it blank and press the enter key to continue.

user@laptop:~$ su hduser

Password:

hduser@laptop:/home/user$ cd ~

hduser@laptop:~$ 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.pub

Your public key has been saved in /home/hduser/.ssh/id\_rsa.pub.

The key fingerprint is:

50:6b:f3:fc:0f:32:bf:30:79:c2:41:71:26:cc:7d:e3 hduser@laptop

The key's randomart image is:

+--[ RSA 2048]----+

| .oo.o |

| . .o=. o |

| . + . o . |

| o = E |

| S + |

| . + |

| O + |

| O o |

| o.. |

+-----------------+

hduser@laptop:~$ cat $HOME/.ssh/id\_rsa.pub >> $HOME/.ssh/authorized\_keys

The second command adds the newly created key to the list of authorized keys so that Hadoop can use ssh without prompting for a password.

We can check if ssh works:

hduser@laptop:~$ ssh localhost

The authenticity of host 'localhost (127.0.0.1)' can't be established.

ECDSA key fingerprint is e1:8b:a0:a5:75:ef:f4:b4:5e:a9:ed:be:64:be:5c:2f.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.

Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-40-generic x86\_64)

...

-After 'winscp, software is download and install

which is used for extracting file from windows to ubuntu.

- In this software select 'SCP' protocol

- For writing host name first check your config by using 'ifconfig' command in ubuntu terminal window.

- then write your ip address in host name.

- last write ur user name and password

# Hadoop installation.

- First move your tar file from window to ubuntu by using 'winscp' s/w

- Then after extract

cd /home/hduser

tar -xzvf hadoop-2.7.1.tar.gz

Configure the Hadoop environment variables in ~/.bashrc (for Ubuntu) or ~/.bash\_profile (for CentOS) using any text editor.

sudo nano ~/.bashrc

Append the following variables in this file:

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

export HADOOP\_PREFIX="/home/hduser/hadoop-2.7.1/"

export PATH=$PATH:$HADOOP\_PREFIX/bin

export PATH=$PATH:$HADOOP\_PREFIX/sbin

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

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

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

export YARN\_HOME=${HADOOP\_PREFIX}

After saving the file, run the source command to refresh the values of the environment variables.

source ~/.bashrc

Edit the /home/hduser/hadoop-2.7.1/etc/hadoop/core-site.xml file and add the Hadoop HDFS URI (NameNode and its port) property under configuration tag.

<property>

<name>fs.defaultFS</name>

<value>hdfs://192.168.56.123:8020</value>

<final>true</final>

</property>

Edit the /home/hduser/hadoop-2.7.1/etc/hadoop/hdfs-site.xml file and add the Hadoop HDFS properties:

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

<value>file:///home/hduser/hadoop-2.7.1/hadoop\_data/dfs/namenode</value>

</property>

<property>

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

<value>file:///home/hduser/hadoop-2.7.1/hadoop\_data/dfs/datanode</value>

</property>

</configuration>

Edit the /home/hduser/hadoop-2.7.1/etc/hadoop/mapred-site.xml file and specify the MapReduce framework as YARN.

<configuration>

<property>

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

<value>yarn</value>

</property>

</configuration>

Edit the /home/hduser/hadoop-2.7.1/etc/hadoop/yarn-site.xml file and specify the YARN properties.

<configuration>

<property>

<name>yarn.resourcemanager.address</name>

<value>192.168.56.123:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>192.168.56.123:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>192.168.56.123:8031</value>

</property>

<property>

<name>yarn.resourcemanager.admin.address</name>

<value>192.168.56.123:8033</value>

</property>

<property>

<name>yarn.resourcemanager.webapp.address</name>

<value>192.168.56.123:8088</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>

<property>

<name>yarn.resourcemanager.scheduler.class</name>

<value>org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler</value>

</property>

</configuration>

Hadoop requires JAVA\_HOME environment variable. You can check the value of JAVA\_HOME in your system using the following command:

echo $JAVA\_HOME

Edit the /home/hduser/hadoop-2.7.1/etc/hadoop/hadoop-env.sh file and specify the JAVA\_HOME for Hadoop.

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

Last step before running the Hadoop services is to format the NameNode. Before executing the format command, make sure that the dfs.namenode.name.dir and dfs.datanode.data.dir directories specified in hdfs-site.xml file does not exists. The hdfs command exists in the /home/hduser/hadoop-2.7.1/bin directory.

hdfs namenode -format

- Then run'jps' command.

- After run 'start-all.sh' command.

- Then all nodes will running.

- if suppose one of node is not running then they run by using manualy means following service command.

- Then also node is not running then following steps are used

STEP 1 stop hadoop

hduser@prayagupd$ /usr/local/hadoop-2.2.0/sbin/stop-dfs.sh

STEP 2 remove tmp folder

hduser@prayagupd$ sudo rm -rf /app/hadoop/tmp/

STEP 3 create /app/hadoop/tmp/

hduser@prayagupd$ sudo mkdir -p /app/hadoop/tmp

hduser@prayagupd$ sudo chown hduser:hadoop /app/hadoop/tmp

hduser@prayagupd$ sudo chmod 750 /app/hadoop/tmp

STEP 4 format namenode

hduser@prayagupd$ hdfs namenode -format

STEP 5 start dfs

hduser@prayagupd$ /usr/local/hadoop-2.2.0/sbin/start-dfs.sh

STEP 6 check jps

hduser@prayagupd$ $ jps

11342 Jps

10804 DataNode

11110 SecondaryNameNode

10558 NameNode

Start Hadoop Services using the Hadoop 2 Scripts in the /home/hduser/hadoop-2.7.1/sbin/ directory.

Service Command

Namenode hadoop-daemon.sh start namenode

Datanode hadoop-daemon.sh start datanode

Resourcemanager yarn-daemon.sh start resourcemanager

Nodemanager yarn-daemon.sh start nodemanager

- Then check hadoop is running or not by using browser,

# How to check the hadoop is running or not

- First open the browser then type "localhost:50070/" then press enter.

- Then this window will vissible



- Finally your Hadoop is installed.