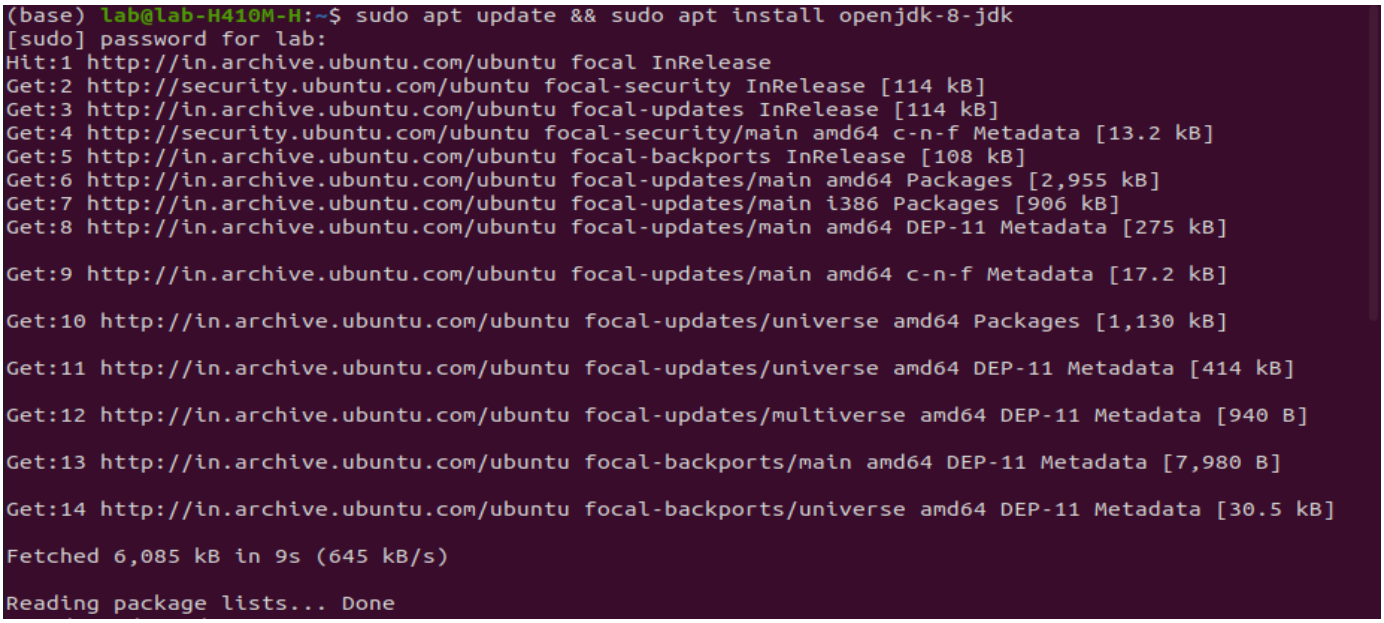
Demystifying Hadoop Installation: A Comprehensive Guide

Step 1 : Install Java Development Kit

The default Ubuntu repositories contain Java 8 and Java 11 both. I am using Java 8 because hive only works on this version.Use the following command to install it.

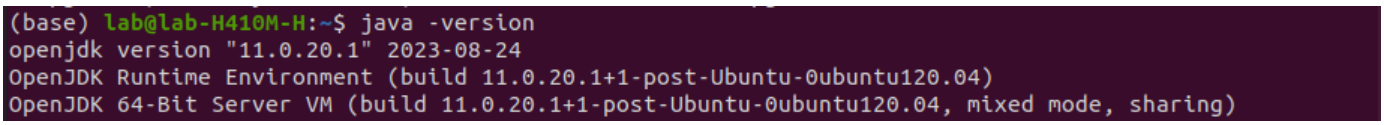
```
sudo apt update && sudo apt install openjdk-8-jdk
```



Step 2 : Verify the Java version :

Once you have successfully installed it, check the current Java version:

```
java -version
```



Step 3 : Install SSH :

SSH (Secure Shell) installation is vital for Hadoop as it enables secure communication between nodes in the Hadoop cluster. This ensures data integrity, confidentiality, and allows for efficient distributed processing of data across the

cluster.

```
sudo apt install ssh
```

```
(base) lab@lab-H410M-H:~$ sudo apt install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
ssh is already the newest version (1:8.2p1-4ubuntu0.9).
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gir1.2-goa-1.0 gstreamer1.0-vaapi libfwupdplugin1
  libgstreamer-plugins-bad1.0-0 libllvm11 libnvidia-cfg1-470 libnvidia-common-470
  libnvidia-decode-470 libnvidia-encode-470 libnvidia-extra-470 libnvidia-fbc1-470 libnvidia-gl-470
  libnvidia-ifr1-470 libva-wayland2 libx11-xcb1:i386 libxmlb1 libxnvctrl0 nvidia-compute-utils-470
  nvidia-prime nvidia-settings nvidia-utils-470 screen-resolution-extra
  xserver-xorg-video-nvidia-470
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 30 not upgraded.
```

Step 4 : Create the Hadoop user :

All the Hadoop components will run as the user that you create for Apache Hadoop, and the user will also be used for logging in to Hadoop's web interface.

Run the command to create user and set password :

```
sudo adduser user_name
```

give the name user_name whatever you want to give in place of user_name and also enter the password. (Remember the password).

```
(base) lab@lab-H410M-H:~$ sudo adduser user_name
Adding user `user_name' ...
Adding new group `user_name' (1002) ...
Adding new user `user_name' (1002) with group `user_name' ...
Creating home directory `/home/user_name' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
```

Step 5 : Switch user :

Switch to the newly created Hadoop user:

```
su - hadoop
```

```
(base) lab@lab-H410M-H:~$ su - hadoop
Password:
hadoop@lab-H410M-H:~$
```

Here user name that I have set is “hadoop”, you write the user name and password that you gave.

Step 6 : Configure SSH :

Now configure password-less SSH access for the newly created Hadoop user, so there is no need to enter key making it able to save file and passphrase. Generate an SSH key pair first:

```
ssh-keygen -t rsa
```

```
hadoop@lab-H410M-H:~$ ssh-keygen -t rsa
```

```
hadoop@sanjay-VirtualBox:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
Created directory '/home/hadoop/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:yu8Hsie3mbQ7UifnfH6iam4kFLRRbEb9zVYGUtbaYyg hadoop@sanjay-VirtualBox
The key's randomart image is:
+---[RSA 3072]----+
|     .o+o.  ..+o |
|      .o+ .  o  +|
|      .+   . o.= |
|      .    E..=o.|
|     .  S   ... .|
|     .o.= o      |
|      o*.B       |
|      +o*++ o .  |
|       X@*.+.o   |
+----[SHA256]-----+
hadoop@sanjay-VirtualBox:~$
```

Step 7 : Set permissions :

Copy the generated public key to the authorized key file and set the proper permissions:

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
chmod 640 ~/.ssh/authorized_keys
```

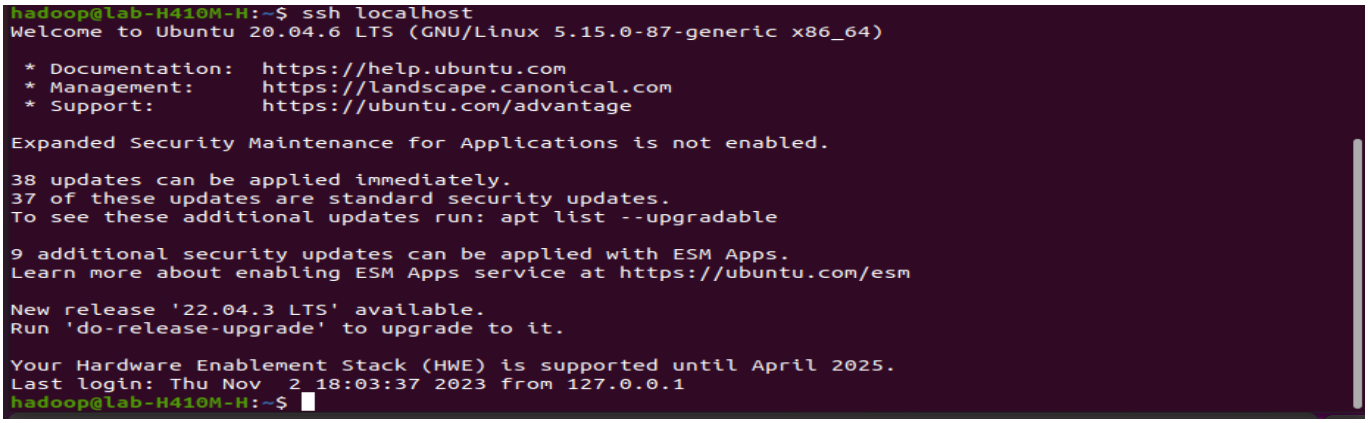
```
hadoop@lab-H410M-H:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hadoop@lab-H410M-H:~$ chmod 640 ~/.ssh/authorized_keys
```

Step 8 : SSH to the localhost :

```
ssh localhost
```

You will be asked to authenticate hosts by adding RSA keys to known hosts. Type yes

and hit Enter to authenticate the localhost.



Step 9 : Switch user Again:

Again switch to hadoop(user name)

```
su - hadoop
```

Step 10 : Install hadoop

- Download hadoop 3.3.6

```
wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz
```

- Once you've downloaded the file, you can unzip it to a folder.

```
tar -xvzf hadoop-3.3.6.tar.gz
```

- Rename the extracted folder to remove version information. This is an optional step, but if you don't want to rename, then adjust the remaining configuration paths

```
mv hadoop-3.3.6 hadoop
```

- Next, I configured the Hadoop and Java Environment Variables on your system. Opening the ~/.bashrc file using nano editior. In nano editor: copy-past: ctrl+shit+c & ctrl+shirt+v, save: ctrl + x.

```
nano ~/.bashrc
```

- Append the below lines to the file.

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export HADOOP_HOME=/home/hadoop/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
```

- Load the above configuration in the current environment.

```
source ~/.bashrc
```

- You also need to configure JAVA_HOME in hadoop-env.sh file. Edit the Hadoop environment variable file in the text editor:

```
nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
```

Search for the “export JAVA_HOME” and configure it .

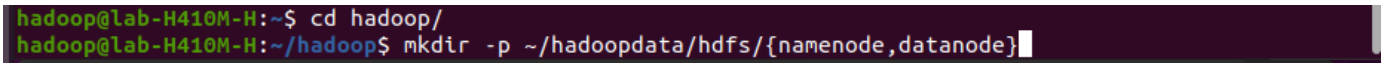
```
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

Step 11 : Configuring Hadoop :

- To configure, I created the namenode and datanode directories inside the Hadoop user home directory. I wrote the following command to create both directories:

```
cd hadoop/
```

```
mkdir -p ~/hadoopdata/hdfs/{namenode,datanode}
```



- Next, edit the core-site.xml file and update with your system hostname:

```
nano $HADOOP_HOME/etc/hadoop/core-site.xml
```

Change the following name as per your system hostname:

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

Save and close the file.

- Then, edit the hdfs-site.xml file:

```
nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml
```

- Change the NameNode and DataNode directory paths as shown below:

```
<configuration>
<property>

<name>dfs.replication</name>
<value>1</value>
</property>

<property>
<name>dfs.namenode.name.dir</name>
<value>file:///home/hadoop/hadoopdata/hdfs/namenode</value>
</property>
```

```
<property>
<name>dfs.datanode.data.dir</name>
<value>file:///home/hadoop/hadoopdata/hdfs/datanode</value>
</property>
</configuration>
```

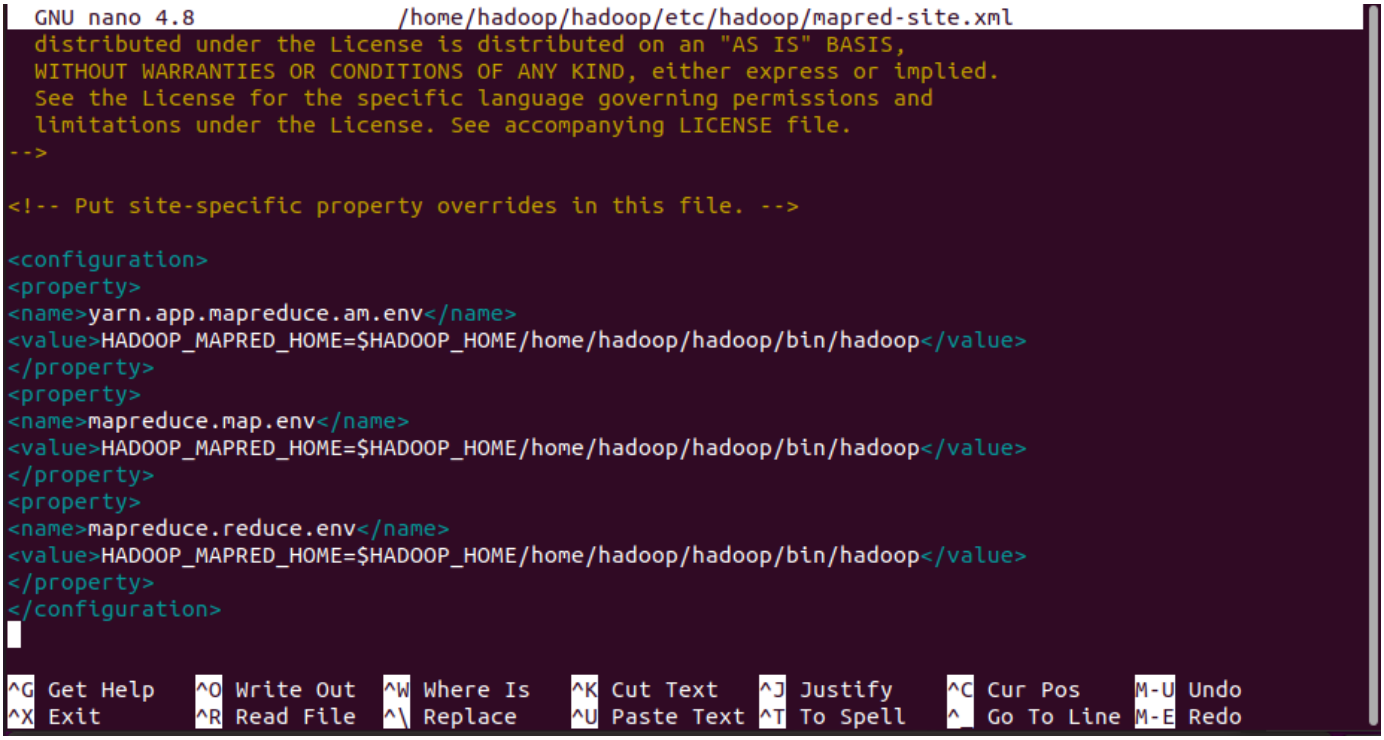
```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
                                [ Read 32 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell  ^_ Go To Line M-E Redo
```

- Then, edit the mapred-site.xml file:

```
nano $HADOOP_HOME/etc/hadoop/mapred-site.xml
```

- Make the following changes

```
<configuration>
<property>
<name>yarn.app.mapreduce.am.env</name>
<value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</val
</property>
<property>
<name>mapreduce.map.env</name>
<value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</val
</property>
<property>
<name>mapreduce.reduce.env</name>
<value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</val
</property>
</configuration>
```



- Then, edit the yarn-site.xml file:

```
nano $HADOOP_HOME/etc/hadoop/yarn-site.xml
```

- Make the following changes:

```
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
</configuration>
```

```
 GNU nano 4.8                 /home/hadoop/hadoop/etc/hadoop/yarn-site.xml
<?xml version="1.0"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<!-- Site specific YARN configuration properties -->

</configuration>


^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```

Save the file and close it .

Step 12 : Start Hadoop cluster:

- Before starting the Hadoop cluster. You will need to format the Namenode as a hadoop user.
- Run the following command to format the Hadoop Namenode:

```
hdfs namenode -format
```

- Once the namenode directory is successfully formatted with hdfs file system, you will see the message “Storage directory /home/hadoop/hadoopdata/hdfs/namenode has been successfully formatted“.

```
hadoop@lab-H410M-H:~/hadoop$ hdfs namenode -format
2023-11-02 21:33:40,591 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = lab-H410M-H/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.3.6
STARTUP_MSG:   classpath = /home/hadoop/hadoop/etc/hadoop:/home/hadoop/hadoop/share/hadoop/common/lib
/kerb-admin-1.0.1.jar:/home/hadoop/hadoop/share/hadoop/common/lib/httpcore-4.4.13.jar:/home/hadoop/ha
```

```
2023-11-02 21:33:44,287 INFO namenode.FSNamesystem: Stopping services started for active state
2023-11-02 21:33:44,288 INFO namenode.FSNamesystem: Stopping services started for standby state
2023-11-02 21:33:44,289 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdo
wn.
2023-11-02 21:33:44,289 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at lab-H410M-H/127.0.1.1
************************************************************/
hadoop@lab-H410M-H:~/hadoop$
```

```
2023-09-10 13:07:27,704 INFO snapshot.SnapshotManager: Loaded config captureOpenFiles: false, skipCaptureAccessTimeOnly
Change: false, snapshotDiffAllowSnapRootDescendant: true, maxSnapshotLimit: 65536
2023-09-10 13:07:27,712 INFO snapshot.SnapshotManager: SkipList is disabled
2023-09-10 13:07:27,727 INFO util.GSet: Computing capacity for map cachedBlocks
2023-09-10 13:07:27,727 INFO util.GSet: VM type       = 64-bit
2023-09-10 13:07:27,727 INFO util.GSet: 0.25% max memory 748 MB = 1.9 MB
2023-09-10 13:07:27,728 INFO util.GSet: capacity      = 2^18 = 262144 entries
2023-09-10 13:07:27,744 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2023-09-10 13:07:27,751 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
2023-09-10 13:07:27,752 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2023-09-10 13:07:27,755 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2023-09-10 13:07:27,756 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expir
y time is 600000 millis
2023-09-10 13:07:27,761 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2023-09-10 13:07:27,761 INFO util.GSet: VM type       = 64-bit
2023-09-10 13:07:27,761 INFO util.GSet: 0.029999999329447746% max memory 748 MB = 229.8 KB
2023-09-10 13:07:27,761 INFO util.GSet: capacity      = 2^15 = 32768 entries
2023-09-10 13:07:27,804 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1272319295-127.0.1.1-1694331447796
2023-09-10 13:07:27,847 INFO common.Storage: Storage directory /home/hadoop/hadoopdata/hdfs/namenode has been successfu
lly formatted.
2023-09-10 13:07:27,899 INFO namenode.FSImageFormatProtobuf: Saving image file /home/hadoop/hadoopdata/hdfs/namenode/cu
rrent/fsimage.ckpt_0000000000000000000 using no compression
2023-09-10 13:07:28,040 INFO namenode.FSImageFormatProtobuf: Image file /home/hadoop/hadoopdata/hdfs/namenode/current/f
simage.ckpt_0000000000000000000 of size 401 bytes saved in 0 seconds .
2023-09-10 13:07:28,054 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2023-09-10 13:07:28,074 INFO namenode.FSNamesystem: Stopping services started for active state
2023-09-10 13:07:28,078 INFO namenode.FSNamesystem: Stopping services started for standby state
2023-09-10 13:07:28,087 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2023-09-10 13:07:28,088 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at sanjay-VirtualBox/127.0.1.1
************************************************************/
hadoop@sanjay-VirtualBox:~/hadoop$
```

- Then start the Hadoop cluster with the following command.

```
start-all.sh
```

```
hadoop@lab-H410M-H:~/hadoop$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [lab-H410M-H]
Starting resourcemanager
Starting nodemanagers
hadoop@lab-H410M-H:~/hadoop$
```

- You can now check the status of all Hadoop services using the jps command:

```
jps
```

```
hadoop@lab-H410M-H:~/hadoop$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [lab-H410M-H]
Starting resourcemanager
Starting nodemanagers
hadoop@lab-H410M-H:~/hadoop$
```

Step 13 : Access Hadoop Namenode and Resource Manager :

- First I need to know our ip address,In Ubuntu I need to install net-tools to run ipconfig command, as I was installing net-tools for the first time I was required

to switch to default user :

```
sudo apt install net-tools
```

```
hadoop@lab-H410M-H:~/hadoop$ sudo apt install net-tools
[sudo] password for hadoop:
hadoop is not in the sudoers file.  This incident will be reported.
```

```
hadoop@lab-H410M-H:~/hadoop$ exit
logout
hadoop@lab-H410M-H:~$
```

```
hadoop@lab-H410M-H:~$ exit
logout
Connection to localhost closed.
hadoop@lab-H410M-H:~$ exit
logout
(base) lab@lab-H410M-H:~$
```

```
(base) lab@lab-H410M-H:~$ sudo apt install net-tools
[sudo] password for lab:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gir1.2-goa-1.0 gstreamer1.0-vaapi libfwupdplugin1
  libgstreamer-plugins-bad1.0-0 libllvm11 libnvidia-cfg1-470 libnvidia-common-470
  libnvidia-decode-470 libnvidia-encode-470 libnvidia-extra-470 libnvidia-fbc1-470 libnvidia-gl-470
  libnvidia-ifr1-470 libva-wayland2 libx11-xcb1:i386 libxmlb1 libxnvctrl0 nvidia-compute-utils-470
  nvidia-prime nvidia-settings nvidia-utils-470 screen-resolution-extra
  xserver-xorg-video-nvidia-470
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 30 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20180626.aebd88e-1ubuntu1 [196 kB]
Fetched 196 kB in 2s (121 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 226749 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
(base) lab@lab-H410M-H:~$
```

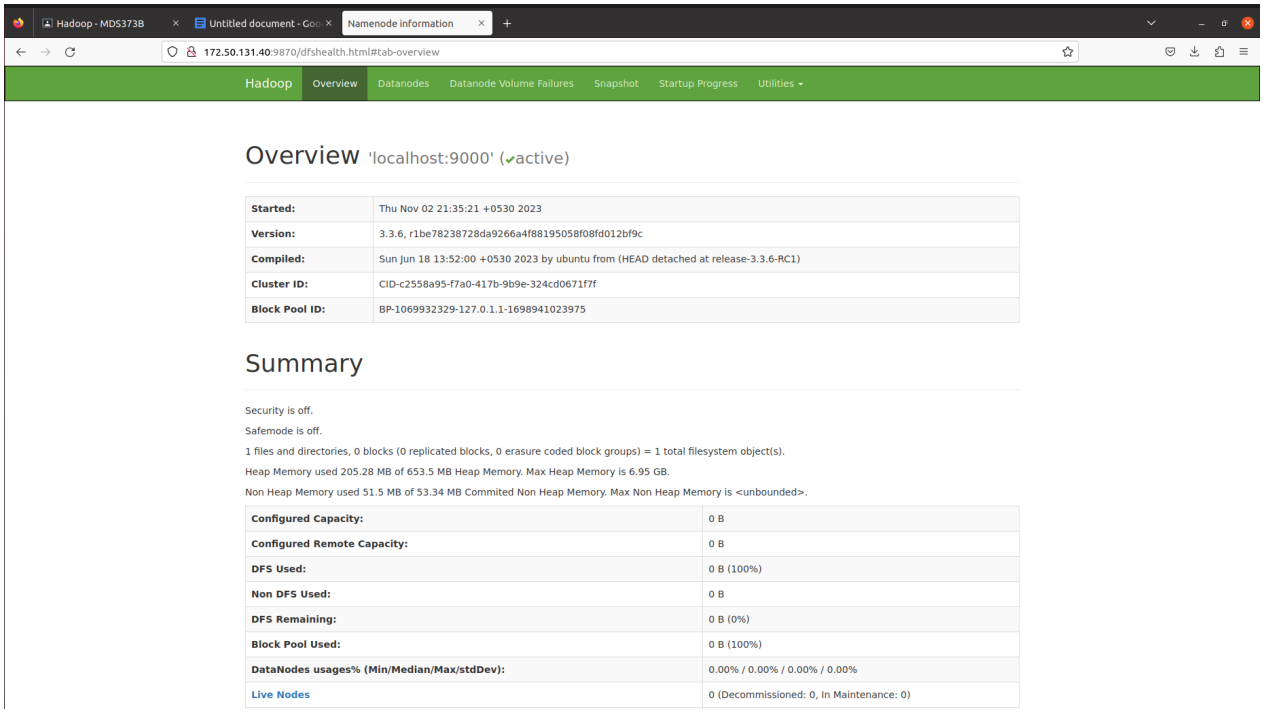
- Then run ifconfig command to know get to know my ip address:

```
ifconfig
```

```
(base) lab@lab-H410M-H:~$ ifconfig
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.50.131.40  netmask 255.255.255.0  broadcast 172.50.131.255
        inet6 fe80::5e14:a075:f430:b9f4  prefixlen 64  scopeid 0x20<link>
        ether 18:c0:4d:b7:b6:18  txqueuelen 1000  (Ethernet)
        RX packets 958057  bytes 1149698037 (1.1 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 561400  bytes 111236487 (111.2 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 107958  bytes 18693144 (18.6 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 107958  bytes 18693144 (18.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

(base) lab@lab-H410M-H:~$
```

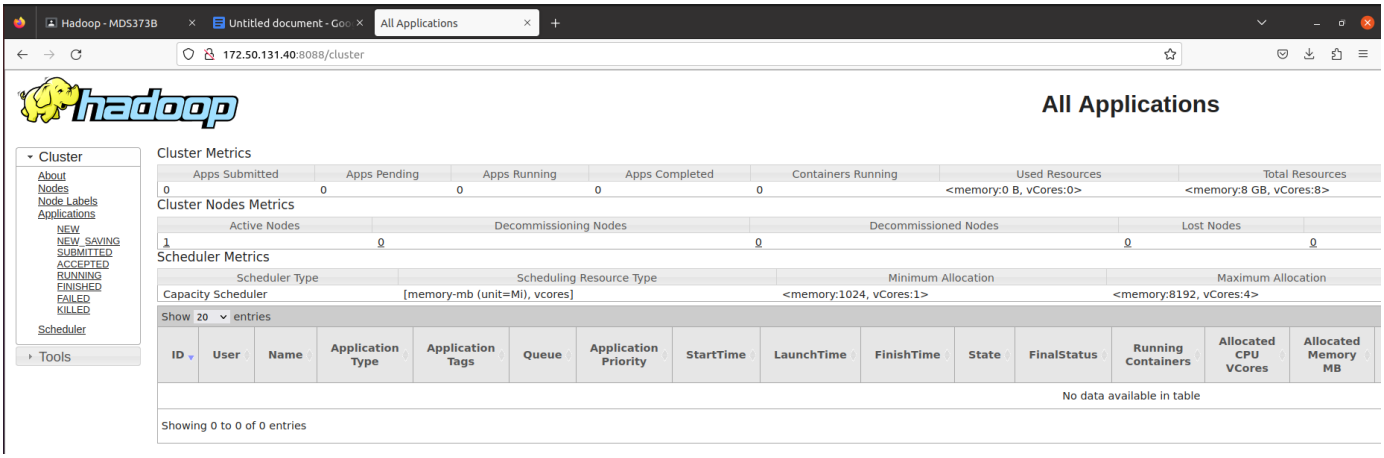
Here my ip address is 172.50.131.40

- To access the Namenode, I opened the web browser and visit the URL http://my- server-ip:9870. following screen is visible:



- To access Resource Manage, opened the web browser and visit the URL http://your-server-ip:8088.

http://192.168.1.6:8088



Step 13 :Verify the Hadoop Cluster :

At this point, the Hadoop cluster is installed and configured. Next, we will create some directories in the HDFS filesystem to test the Hadoop.

- Let’s create some directories in the HDFS filesystem using the following command:

```
hdfs dfs -mkdir /test1
hdfs dfs -mkdir /logs
```

```
hadoop@lab-H410M-H:~$ hdfs dfs -mkdir /test1
hadoop@lab-H410M-H:~$ hdfs dfs -mkdir /logs
```

- Next, run the following command to list the above directory:

```
hdfs dfs -ls /
```

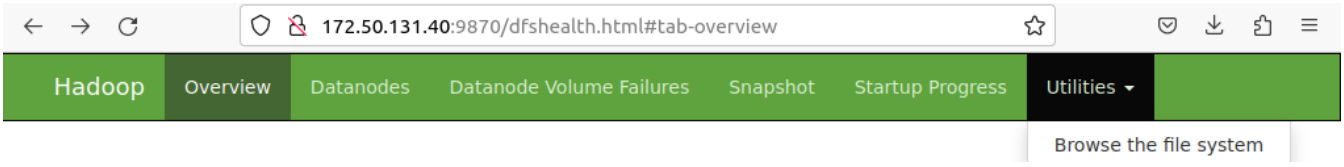
You should get the following output:

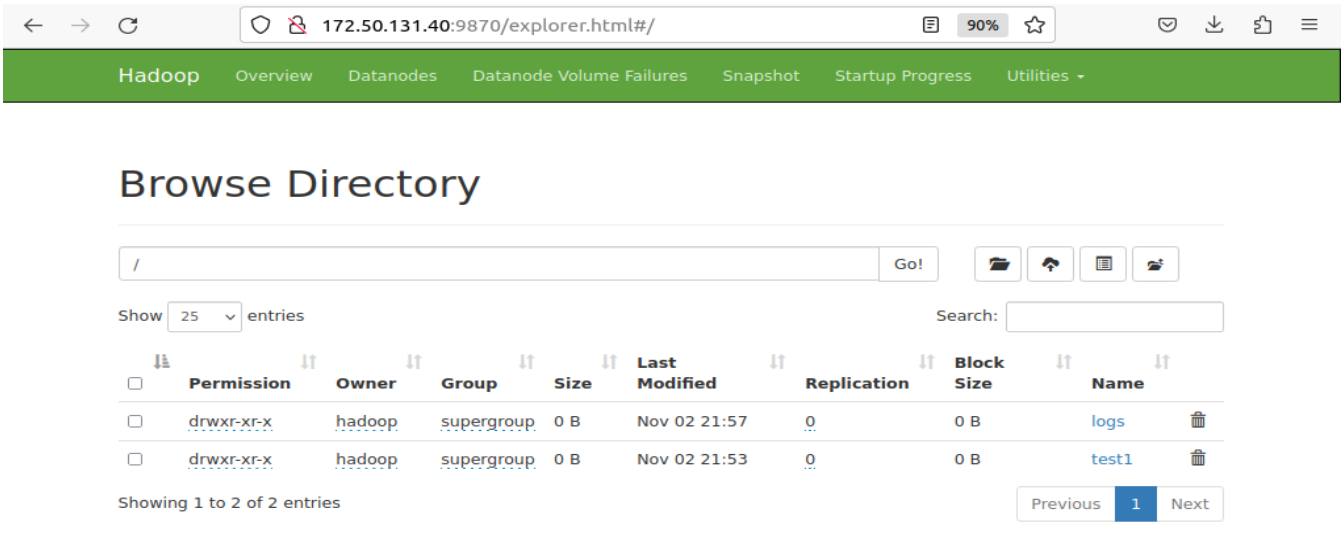
- Also, put some files to hadoop file system. For the example, putting log files from host machine to hadoop file system.

```
hdfs dfs -put /var/log/* /logs/
```

You can also verify the above files and directory in the Hadoop web interface.

Go to the web interface, click on the Utilities => Browse the file system. You should see your directories which you have created earlier in the following screen:





Step 14 : To stop hadoop services :

To stop the Hadoop service, run the following command as a hadoop user:

```
stop-all.sh
```