Exp.1 Downloading and installing Hadoop, Understanding different Hadoop modes, Startup scripts, Configuration files.

AIM:

To Download and install Hadoop, Understanding different Hadoop modes, Startup scripts, Configuration files.

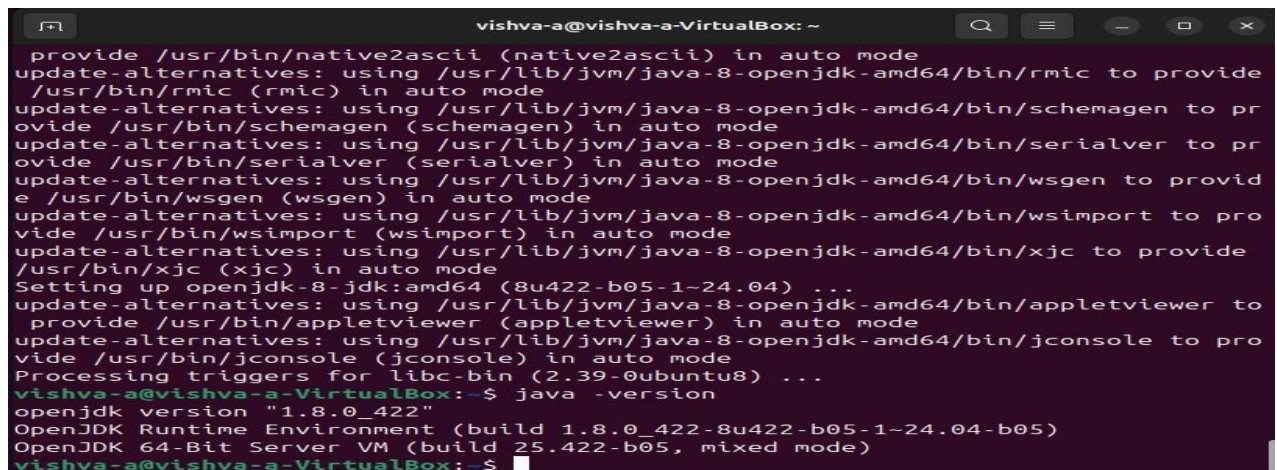**Procedure:**

Step 1 : Install Java Development Kit
The default Ubuntu repositories contain Java 8 and Java 11 both. But, Install 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 installed, verify the installed version of Java with the following command:

$ java -version Output:

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**

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 hadoop

Output:


```
BAD PASSWORD: The password contains the user name in some form
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a diction
ary word
Retype new password:
Sorry, passwords do not match.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for hadoop
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
info: Adding new user `hadoop' to supplemental / extra groups `users' ...
info: Adding user `hadoop' to group `users' ...
vishva-a@vishva-a-VirtualBox:~$ su - hadoop
Password:
hadoop@vishva-a-VirtualBox:~$
```

Step 5 : Switch user

Switch to the newly created hadoop user:

$ su - hadoop

Step 6 : Configure SSH

Now configure password-less SSH access for the newly created hadoop user, so didn't enter the key to save file and passphrase. Generate an SSH keypair (generate Public and Private Key Pairs)first


```
hadoop@vishva-a-VirtualBox:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hadoop@vishva-a-VirtualBox:~$ chmod 640 ~/.ssh/authorized_keys
hadoop@vishva-a-VirtualBox:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:qwEaicW2kGbLyCNuxqstOLGP/tauCfxQH0tg11xNb2c.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
Enter passphrase for key '/home/hadoop/.ssh/id_rsa':
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

308 updates can be applied immediately.
155 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

$ssh-keygen -t rsa

Step 7 : Set permissions :

Next, append the generated public keys from id_rsa.pub to authorized_keys and set proper permission:

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

Step 8 : SSH to the localhost
Next, verify the password less SSH authentication with the following command:

$ 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 switch to hadoop. So, First, change the user to hadoop with the following command: **$ su–hadoop**

Step 10 : Install hadoop

Next, download the latest version of Hadoop using the wget command:

$ wgethttps://downloads.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz Once

downloaded, extract the downloaded file:

$ tar -xvzf hadoop-3.3.6.tar.gz

Next, rename the extracted directory to hadoop:

$ mv hadoop-3.3.6 hadoop

Next, you will need to configure Hadoop and Java Environment Variables on your system. Open the ~/.bashrc file in your favorite text editor. Use nano editior , to pasting the code we use ctrl+shift+v for saving the file ctrl+x and ctrl+y ,then hit enter:

Next, you will need to configure Hadoop and Java Environment Variables on your system.

Open the ~/.bashrc file in your favorite text editor:

$ nano ~/.bashrc

Append the below lines to 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"
```

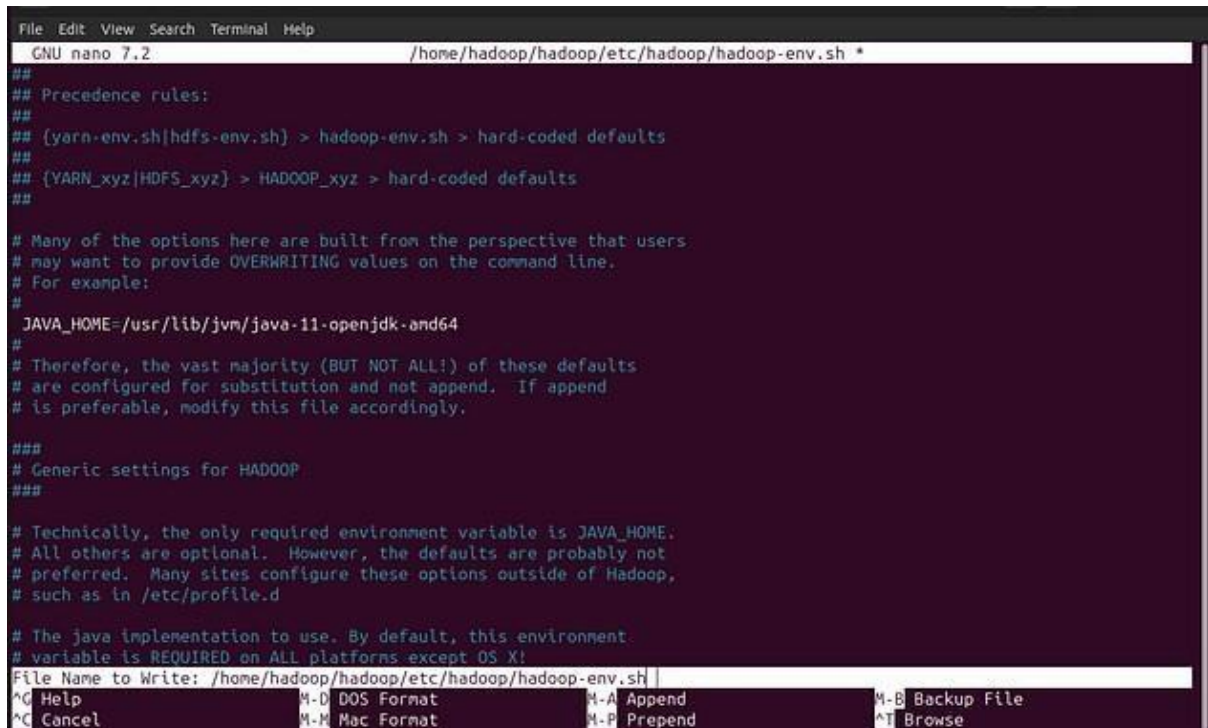Save and close the file. Then, activate the environment variables with the following command:

s$ source ~/.bashrc

Next, open the Hadoop environment variable file: **$ 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
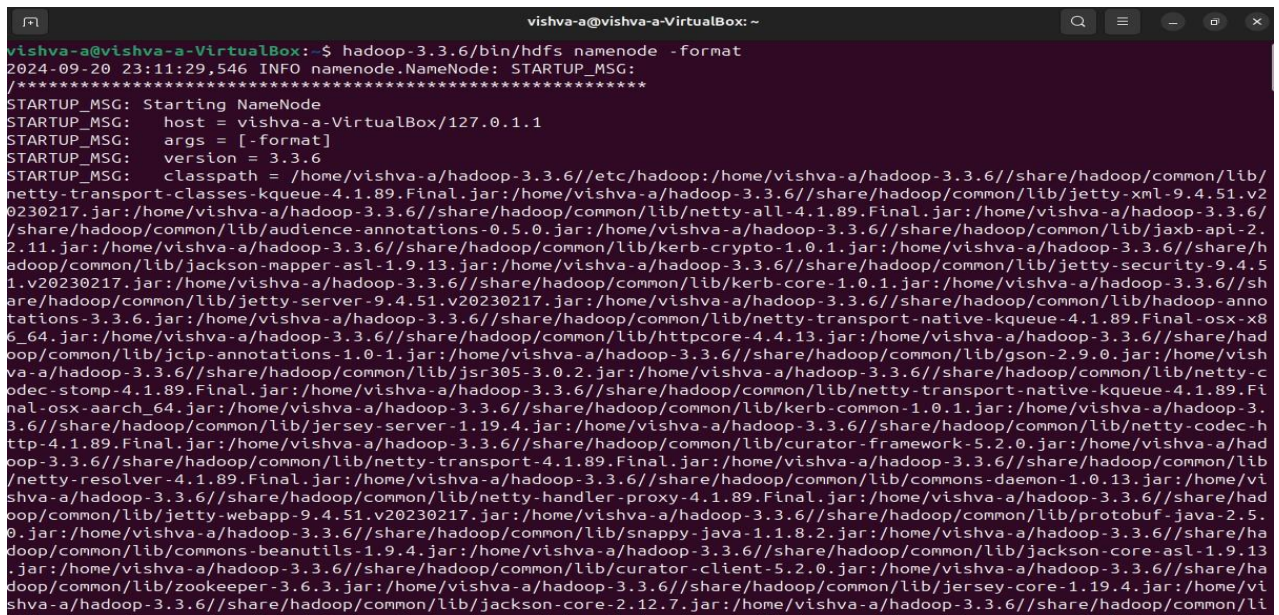

Save and close the file when you are finished.

Step 11 : Configuring Hadoop :
First, you will need to create the namenode and datanode directories inside the Hadoop user home directory. Run 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:

```xml
<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>
```

- Then, edit the mapred-site.xml file:
  $nano $HADOOP_HOME/etc/hadoop/mapred-site.xml

- Make the following changes:

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

- Then, edit the yarn-site.xml file:
  $nano $HADOOP HOME/etc/hadoop/yarn-site.xml
- Make the following changes:

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

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 "

Then start the Hadoop cluster with the following command. **$
start-all.sh**

```
vishva-a@vishva-a-VirtualBox:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as vishva-a 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 [vishva-a-VirtualBox]
Starting resourcemanager
Starting nodemanagers
```

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

$ jps

```
vishva-a@vishva-a-VirtualBox:~$ jps
3426 SecondaryNameNode
3971 Jps
3717 ResourceManager
3238 DataNode
3082 NameNode
3835 NodeManager
```

Step 13 – Access Hadoop Namenode and Resource Manager

• First we need to know our ipaddress, In Ubuntu we need to install net-tools to run ipconfig command,

  If you installing net-tools for the first time switch to default user:

  $sudo apt install net-tools

• Then run ifconfig command to know our ip address: **ifconfig**

```
                        hadoop@vishva-a-VirtualBox: ~
  command 'iconfig' from deb ipmiutil (3.1.9-3)
  command 'hipconfig' from deb hipcc (5.2.3-12)
  command 'iwconfig' from deb wireless-tools (30~pre9-13.1ubuntu4)
Try: apt install <deb name>
hadoop@vishva-a-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe0d:e52f  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:0d:e5:2f  txqueuelen 1000  (Ethernet)
        RX packets 522125  bytes 781215302 (781.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 175427  bytes 11160991 (11.1 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 1097  bytes 139599 (139.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1097  bytes 139599 (139.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

hadoop@vishva-a-VirtualBox:~$
```

Here my ip address is 10.0.2.15.

• To access the Namenode, open your web browser and visit the URL http://yourserverip:9870.

• You should see the following screen:

  http://10.0.2.15:9870

To access Resource Manage, open your web browser and visit the URL http://yourserverip:8088.
You should see the following screen: http://10.0.2.15:8088

Step 14 – 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:

```
$ hdfsdfs  -mkdir /test1
$ hdfsdfs  -mkdir /logs
```

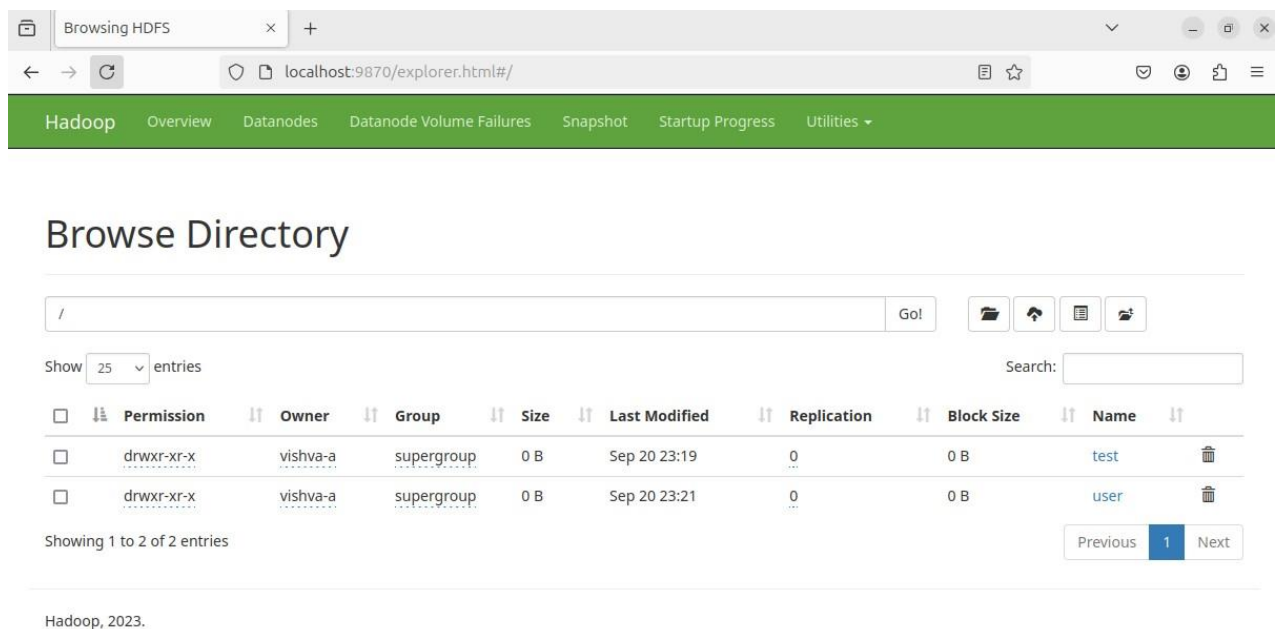Next, run the following command to list the above directory:

```
vishva-a@vishva-a-VirtualBox:~$ hadoop fs -mkdir /test
vishva-a@vishva-a-VirtualBox:~$ hadoop fs -mkdir /user
```

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 Namenode 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 15 – Stop Hadoop Cluster

To stop the Hadoop all services, run the following command:

$ stop-all.sh

```
vishva-a@vishva-a-VirtualBox:~$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as vishva-a in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
localhost: WARNING: datanode did not stop gracefully after 5 seconds: Trying to kill with kill -9
Stopping secondary namenodes [vishva-a-VirtualBox]
vishva-a-VirtualBox: WARNING: secondarynamenode did not stop gracefully after 5 seconds: Trying to kill with kill -9
Stopping nodemanagers
localhost: WARNING: nodemanager did not stop gracefully after 5 seconds: Trying to kill with kill -9
Stopping resourcemanager
WARNING: resourcemanager did not stop gracefully after 5 seconds: Trying to kill with kill -9
```

Result:

The step-by-step installation and configuration of Hadoop on Ubutu linux system have been successfully completed.