Exp. No: 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:

```
(base) lab@lab-H410M-H:~$ java -version
openjdk version "11.0.20.1" 2023-08-24
OpenJDK Runtime Environment (build 11.0.20.1+1-post-Ubuntu-0ubuntu120.04)
OpenJDK 64-Bit Server VM (build 11.0.20.1+1-post-Ubuntu-0ubuntu120.04, mixed mode, sharing)
```

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

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

$ 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:nLS3cQzog3Q6+Rv5AZgFaG0PhjaCwj8mPO9XhE3HuE4 hadoop@lab-H410M-H
The key's randomart image is:
+---[RSA 3072]----+
|..   +. o        |
|o.. * =+ +       |
|o .+ +=oB .      |
| + + o E.o o     |
|  = . @ S o o    |
|   .   = = +     |
|  .   . + o      |
|   . .   + .     |
|    .   . .      |
+----[SHA256]-----+
```

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

```
hadoop@lab-H410M-H:~$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz
--2023-11-02 18:06:41--  https://dlcdn.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 730107476 (696M) [application/x-gzip]
Saving to: 'hadoop-3.3.6.tar.gz'

hadoop-3.3.6.tar.gz                          78%[=============================================
```

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:

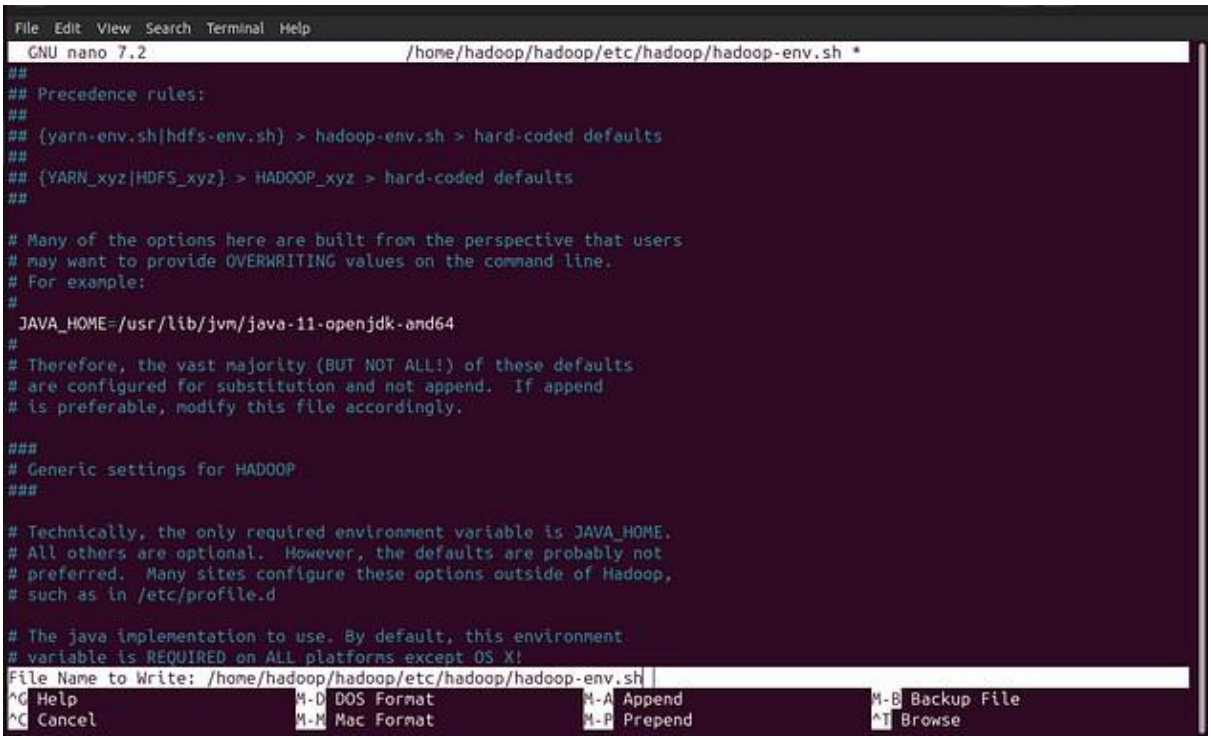
s$ source ~/.bashrc

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:

```
<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:

```
<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:

```
<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

```
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
```

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

$ jps

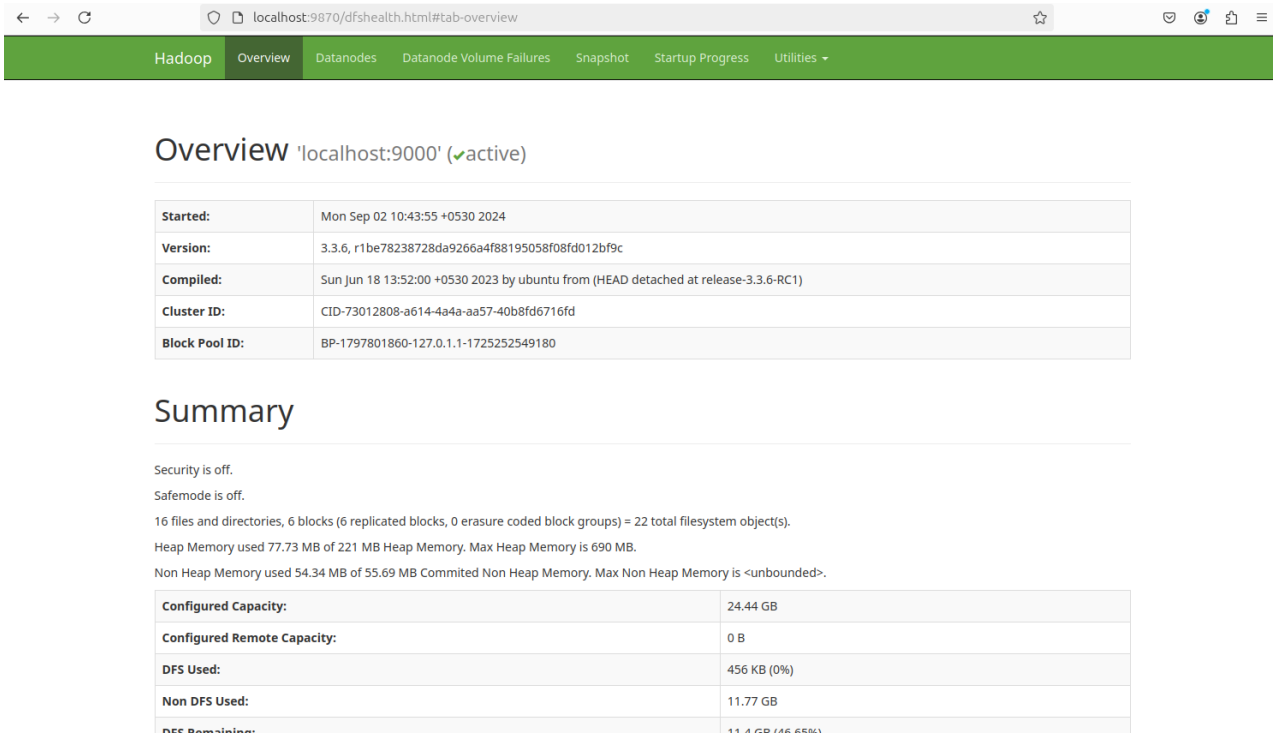
```
hadoop@lab-H410M-H:~/hadoop$ jps
14480 NodeManager
13857 SecondaryNameNode
14722 Jps
14122 ResourceManager
13418 NameNode
13615 DataNode
hadoop@lab-H410M-H:~/hadoop$
```

Step 13 – Access Hadoop Namenode and Resource Manager

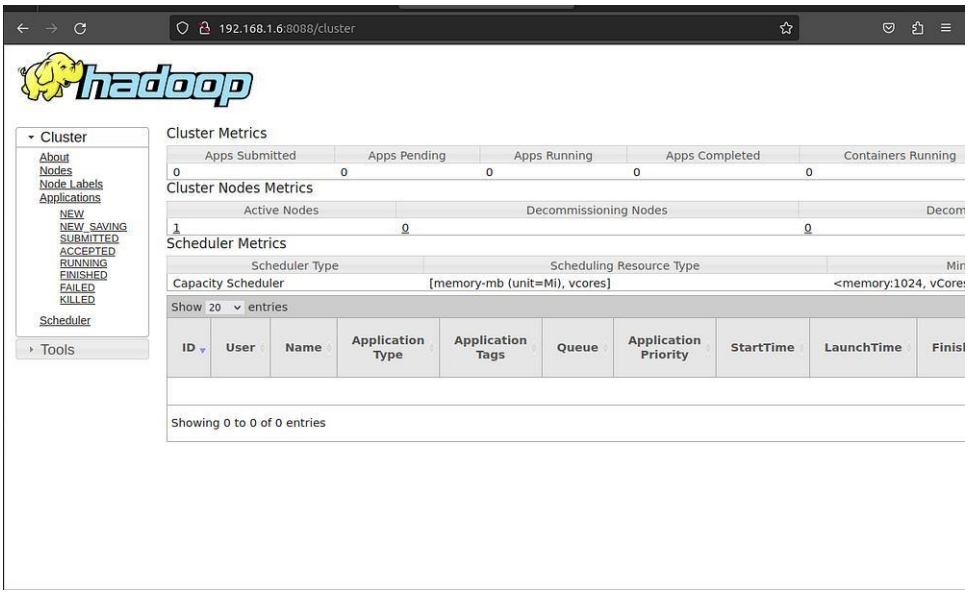
- First we need to know our ipaddress, In Ubuntu we need to install net-tools to run ipconfig command,
  If you installing net-tools for the first time switch to default user:
  $sudo apt install net-tools
- Then run ifconfig command to know our ip address: **ifconfig**

Here my ip address is 192.168.1.6.

- To access the Namenode, open your web browser and visit the URL http://your-serverip:9870.
- You should see the following screen:
  http://192.168.1.6:9870



To access Resource Manage, open your web browser and visit the URL http://your-serverip:8088. You should see the following screen: http://192.168.16:8088



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:

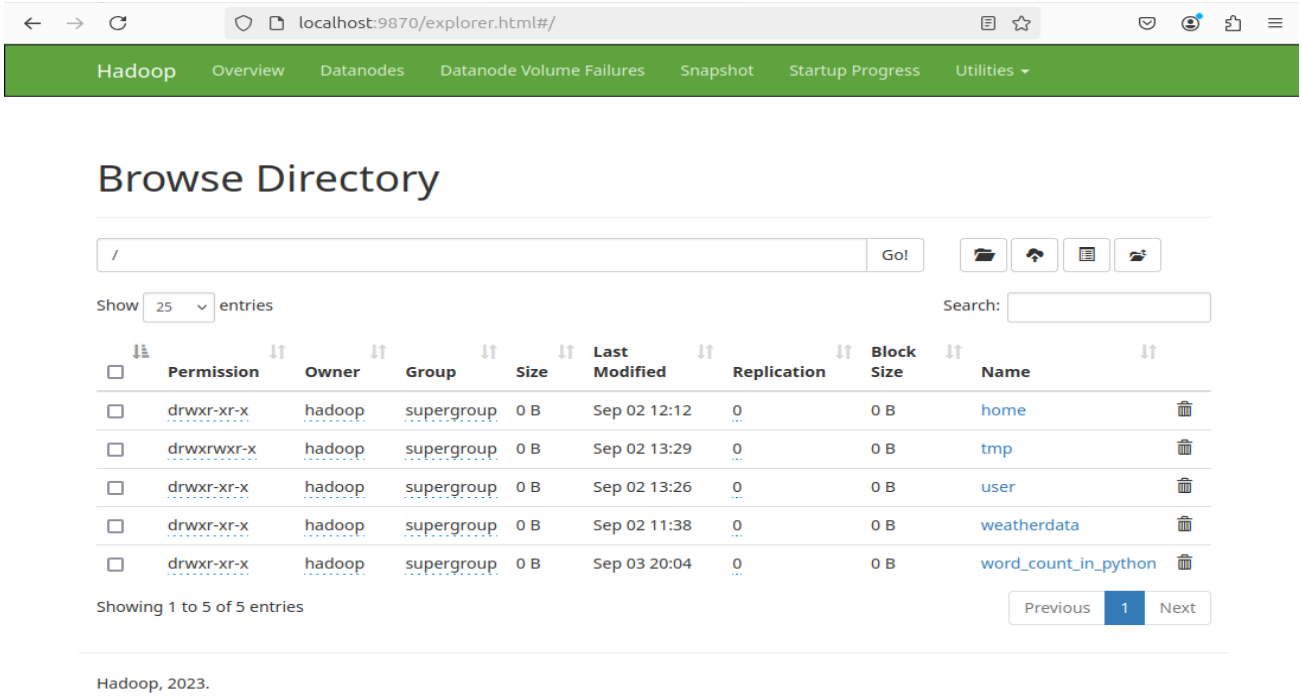
```
$ hdfsdfs -mkdir /test1
$ hdfsdfs -mkdir /logs
```

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

```
hadoop@lab-H410M-H:~/hadoop$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as hadoop in 10 seconds
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [lab-H410M-H]
Stopping nodemanagers
Stopping resourcemanager
hadoop@lab-H410M-H:~/hadoop$
```

Result:

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