Hands-on 1 (CW) - Hadoop Installation Guide and HDFS

Every step is to be executed on the home directory. Use `cd` to move to home directory.

The commands in the guide use `pes0ug21cs000` as the notation for your username. If you have executed A0 correctly, then this should **be your SRN in lowercase.** This is important since the auto-evaluation depends on it. Verify your username by running `whoami` on the terminal.

```
pes2ug21cs925@pes2ug21cs925:~$ whoami
pes2ug21cs925
```

Change any `/home/pes0ug21cs000/` to `/home/<your SRN>/`

This manual includes steps that you will be doing in the classroom. It assumes that you have completed the downloads and installation steps 1-2 from your home which was sent earlier. If you have not completed these steps, then click here to do so.

Execute the following commands to move to the home directory and updating the package list and the system. This guide assumes that you are working with **Ubuntu or a Debian** based distribution.

```
cd
sudo apt update -y
sudo apt upgrade -y
```

Step number continues from the H1_HOME manual.

Step 3 - Format HDFS NameNode

Before starting Hadoop for the first time, the namenode must be formatted. Use the following command.

```
hdfs namenode -format
```

A `SHUTDOWN` message will signify the end of the formatting process.

If you have reached this stage, it signifies that you have successfully installed hadoop.

```
2024-08-18 20:46:29,680 INFO util.GSet: 1.0% max memory 868 MB = 8.7 MB
2024-08-18 20:46:29,680 INFO util.GSet: capacity      = 2^20 = 1048576 entries
2024-08-18 20:46:29,680 INFO namenode.FSDirectory: ACLs enabled? true
2024-08-18 20:46:29,680 INFO namenode.FSDirectory: POSIX ACL inheritance enabled? true
2024-08-18 20:46:29,680 INFO namenode.FSDirectory: XAttrs enabled? true
2024-08-18 20:46:29,681 INFO namenode.NameNode: Caching file names occurring more than 10 times
2024-08-18 20:46:29,684 INFO snapshot.SnapshotManager: Loaded config captureOpenFiles: false, skipCaptureAccessTimeOnlyChange: false, snapshotDiffAllowSnapRootDescendant: true, maxSnap
shotLimit: 65536
2024-08-18 20:46:29,685 INFO snapshot.SnapshotManager: SkipList is disabled
2024-08-18 20:46:29,687 INFO util.GSet: Computing capacity for map cachedBlocks
2024-08-18 20:46:29,687 INFO util.GSet: VM type       = 64-bit
2024-08-18 20:46:29,687 INFO util.GSet: 0.25% max memory 868 MB = 2.2 MB
2024-08-18 20:46:29,687 INFO util.GSet: capacity      = 2^18 = 262144 entries
2024-08-18 20:46:29,694 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2024-08-18 20:46:29,694 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
2024-08-18 20:46:29,694 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2024-08-18 20:46:29,698 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2024-08-18 20:46:29,698 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
2024-08-18 20:46:29,699 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2024-08-18 20:46:29,699 INFO util.GSet: VM type       = 64-bit
2024-08-18 20:46:29,700 INFO util.GSet: 0.029999999329447746% max memory 868 MB = 266.6 KB
2024-08-18 20:46:29,700 INFO util.GSet: capacity      = 2^15 = 32768 entries
2024-08-18 20:46:29,719 INFO namenode.FSImage: Allocated new BlockPoolId: BP-880327019-127.0.1.1-1724013989716
2024-08-18 20:46:29,733 INFO common.Storage: Storage directory /home/pes2ug21cs925/dfsdata/namenode has been successfully formatted.
2024-08-18 20:46:29,789 INFO namenode.FSImageFormatProtobuf: Saving image file /home/pes2ug21cs925/dfsdata/namenode/current/fsimage.ckpt_0000000000000000000 using no compression
2024-08-18 20:46:29,851 INFO namenode.FSImageFormatProtobuf: Image file /home/pes2ug21cs925/dfsdata/namenode/current/fsimage.ckpt_0000000000000000000 of size 408 bytes saved in 0 secon
ds .
2024-08-18 20:46:29,855 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2024-08-18 20:46:29,867 INFO namenode.FSNamesystem: Stopping services started for active state
2024-08-18 20:46:29,867 INFO namenode.FSNamesystem: Stopping services started for standby state
2024-08-18 20:46:29,870 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2024-08-18 20:46:29,870 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at pes2ug21cs925/127.0.1.1
************************************************************/
```

Step 4 - Starting Hadoop

Navigate to the `hadoop` folder and execute the following commands. `start-all.sh` is a shell script that is used to start all the processes that hadoop requires.

```
cd
cd hadoop-3.3.6/sbin/
./start-all.sh
```

```
pes2ug21cs925@pes2ug21cs925:~/hadoop-3.3.6/sbin$ ./start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as pes2ug21cs925 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 [pes2ug21cs925]
2024-08-18 20:21:37,341 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
```

Type `jps` to find all the Java Processes started by the shell script. ee a total of 6 processes, including the `jps` process.

Note that the order of the items and the process IDs will be different

```
2994 DataNode
3219 SecondaryNameNode
3927 Jps
3431 ResourceManager
2856 NameNode
3566 NodeManager
```

```
pes2ug21cs925@pes2ug21cs925:~/hadoop-3.3.6/sbin$ jps
7985 NodeManager
8181 Jps
7030 NameNode
7640 ResourceManager
7435 SecondaryNameNode
7214 DataNode
```

Step 5 - Accessing Hadoop from the Browser

You can access Hadoop on `localhost` on the following ports

- NameNode - http://localhost:9870
- DataNode - http://localhost:9864
- YARN Manager - http://localhost:8088

Step 6 - Hadoop Examples

We will be using the Wordcount example to demonstrate the usage of Hadoop. Create a text file named `input.txt` with any content you want. Next, we will put this to the HDFS folder `/example` with the following command.

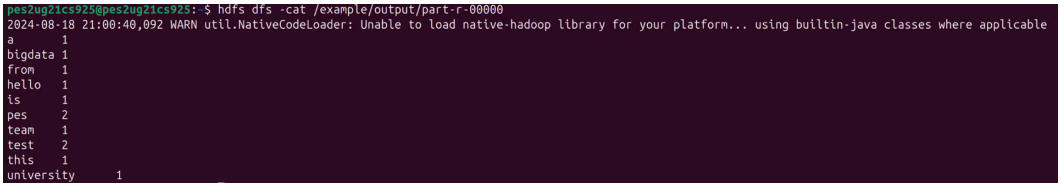
```
cd
hdfs dfs -mkdir /example
hdfs dfs -put input.txt /example
```

Run the following command for the wordcount example.

```
hadoop jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-e
```

 ck the output with the following command.

```
hdfs dfs -cat /example/output/part-r-00000
```



Step 7 - Simulating MapReduce in Local System

Download and extract the zip file:

CW_1.zip 0.8KB

Install dos2unix if you haven't before

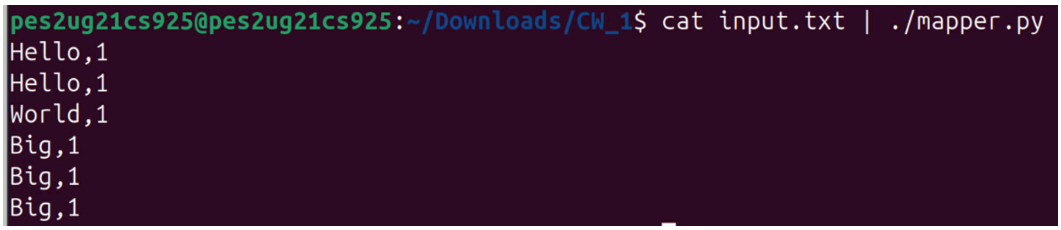
```
sudo apt install dos2unix
```

Then **change your directory** to the directory where you extracted the above .zip file and convert the files to unix format using the following command :

```
dos2unix *.py
```

Pass input to a mapper file using `echo` command

```
cat input.txt | ./mapper.py
```



You can then sort the output from the mapper file using :

```
cat input.txt | ./mapper.py | sort
```

```
pes2ug21cs925@pes2ug21cs925:~/Downloads/CW_1$ cat input.txt | ./mapper.py | sort
Big,1
Big,1
Big,1
Hello,1
Hello,1
World,1
```

Now, finally call the reducer file

```
cat input.txt | ./mapper.py | sort | ./reducer.py
```

```
pes2ug21cs925@pes2ug21cs925:~/Downloads/CW_1$ cat input.txt | ./mapper.py | sort | ./reducer.py
Big 3
Hello 2
World 1
```

> 💡 *Note :*
> *If you're getting a* `bash: ./mapper.py: Permission denied` *error, run* `chmod +x mapper.py`
> *Similarly, for* `bash: ./reducer.py: Permission denied` *error, run* `chmod +x reducer.py`

Step 8 - Running Custom Word Count

Now, we will run a sample HDFS command to calculate the frequency of a particular word in a text file using our own mapper and reducer files.

```
hdfs dfs -mkdir /cw # creating a directory in hadoop dfs
hdfs dfs -put ./input.txt /cw # adding a file to hadoop dfs
chmod +x *.py # granting permissions
```

Next, run the following command to run the wordcount program.

```
hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3
```

To check the output, execute the following command.

```
hdfs dfs -cat /cw/output-text/part-00000
```


Your output should look like this :

```
varunc@varunc:~/testing/CW_1$ hdfs dfs -cat /cw/output-text-7/part-00000
2024-08-24 12:52:56,944 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes
 where applicable
Big 3
Hello 2
World 1
```

FAQs :

- ## If you cannot see datanode, try the following steps :

Feel free to add in any issues you're encountering. If you fixed any issue you think others can encounter, feel free to contribute here