DATA
ENGINEER

INPT
المعهد الوطني للبريد والمواصلات
Institut National des Postes et Télécommunications

TP1 : Apache Hadoop

Rapport

Réalisé par :

Abderrahim ALAKOUCHE
Abdellah AGHLALOU

Encadré par :

Mme. Dounia ZAIDOUNI

PLAN

Introduction

I Installation et configuration d'un nœud unique

II Exécution du programme MapReduce « Word Count » dans le cluster à nœud unique

III Installation et configuration d'un cluster multi-nœuds

IV Exécution du programme MapReduce « Word Count » dans le cluster multi-nœuds

V Extension : benchmark détaillé

Introduction

Hadoop est une infrastructure logicielle à code source libre pour le stockage et le traitement à grande échelle d'ensembles de données dans un environnement informatique distribué. Il est sponsorisé par Apache Software Foundation. Il est conçu pour passer de serveurs uniques à des milliers de machines, chacune offrant des fonctions de calcul et de stockage locales.

Hadoop a été créé par Doug Cutting et Mike Cafarella en 2005.Cutting, qui travaillait chez Yahoo! à l'époque, l'a nommé après l'éléphant de jouet de son fils. Hadoop a été initialement développé pour prendre en charge la distribution du projet de moteur de recherche.[1]

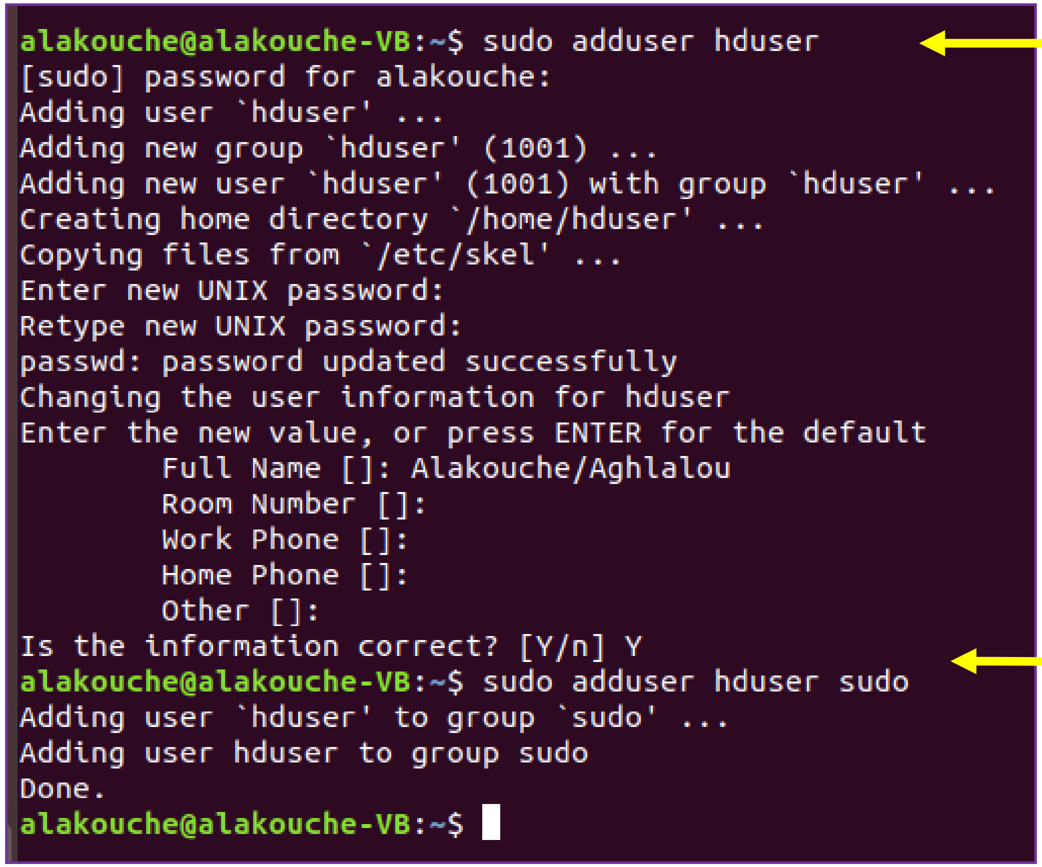
Les objectifs de ce TP sont les suivants :

- Installation et configuration d'un nœud unique d'Apache Hadoop 3.2.1
- Exécution du programme MapReduce « **Word Count** » dans le cluster à nœud unique de Hadoop
- Installation et configuration d'un cluster multi-nœuds d'Apache Hadoop 3.2.1
- Exécution du programme MapReduce « **Word Count** » dans le cluster multi-nœuds de Hadoop
- Faire un benchmark sur les différentes plateformes cloud permettant la création d'un cluster multi-nœuds déjà configuré avec les outils de l'écosystème Hadoop.

[1]: https://riptutorial.com/fr/hadoop

I Installation et configuration d'un nœud unique

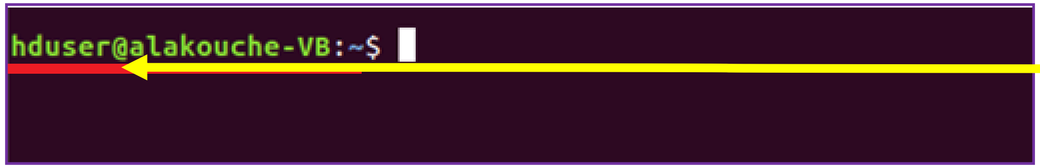
Etape1 : Création d'un utilisateur hduser



```
alakouche@alakouche-VB:~$ sudo adduser hduser
[sudo] password for alakouche:
Adding user `hduser' ...
Adding new group `hduser' (1001) ...
Adding new user `hduser' (1001) with group `hduser' ...
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 []: Alakouche/Aghlalou
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
alakouche@alakouche-VB:~$ sudo adduser hduser sudo
Adding user `hduser' to group `sudo' ...
Adding user hduser to group sudo
Done.
alakouche@alakouche-VB:~$
```

Création d’un compte normal (non root) **hduser**

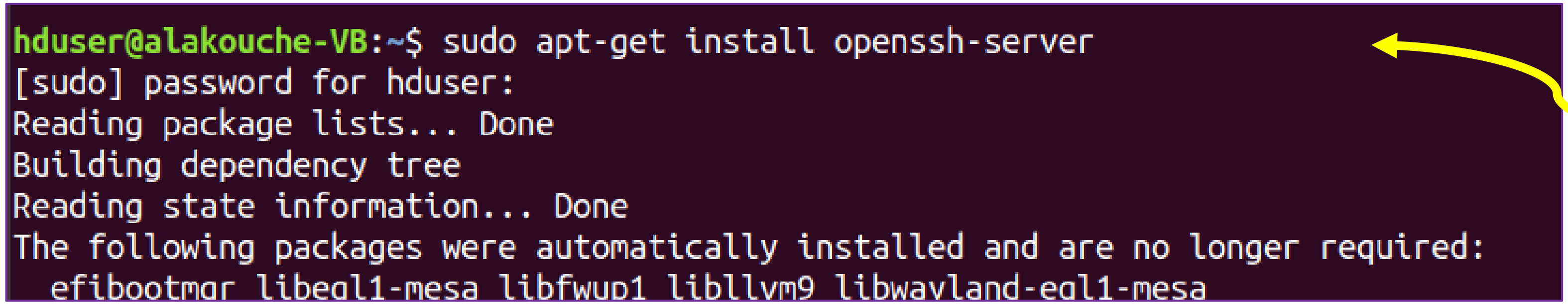
Cette commande va nous permettre d’éviter les erreurs du types « hduser is not in the sudoers file»



```
hduser@alakouche-VB:~$
```

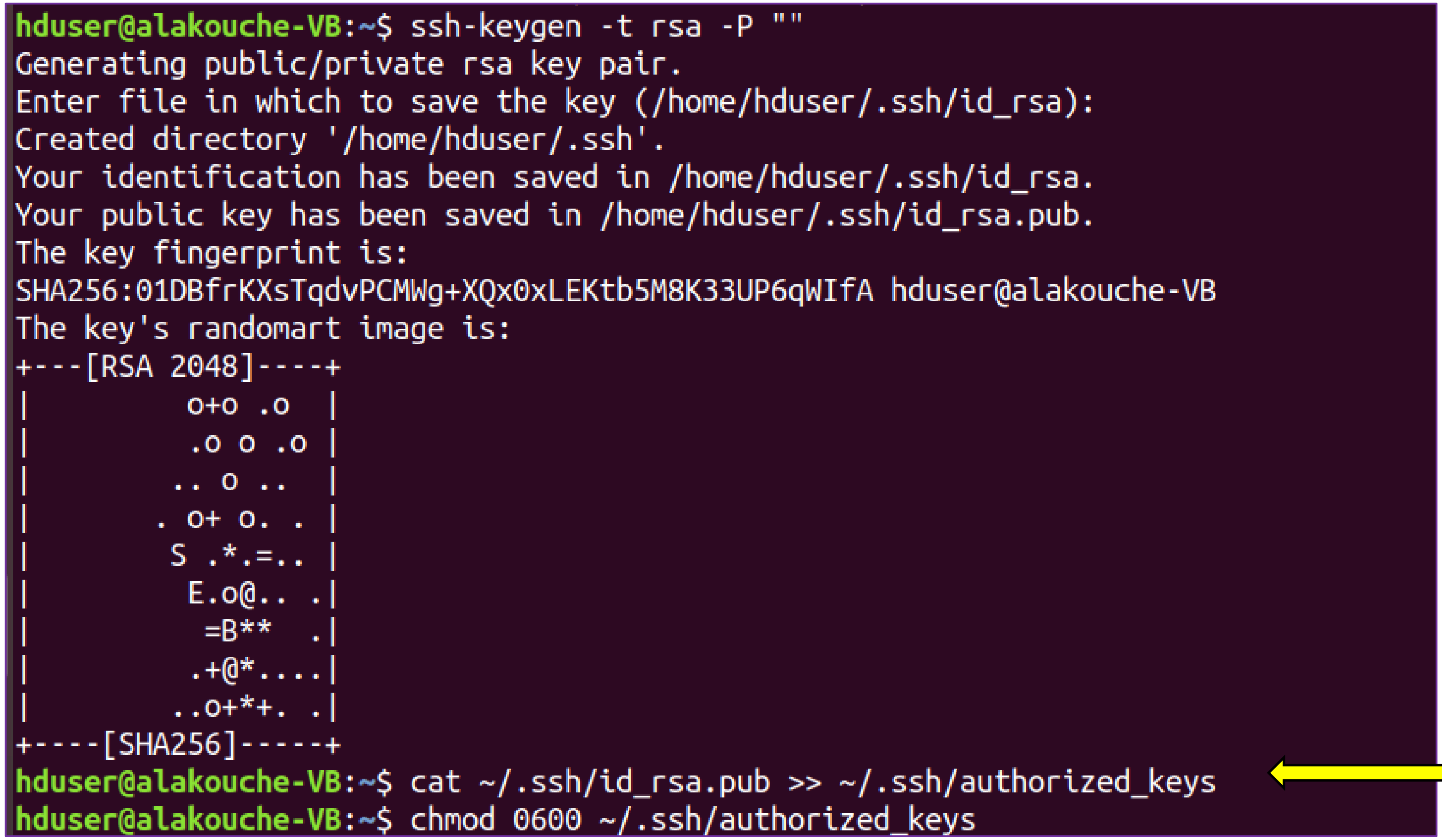
On redémarre la machine virtuelle afin de basculer vers le compte **hduser**

Etape2 : Mise en place de la clé ssh



```
hduser@alakouche-VB:~$ sudo apt-get install openssh-server
[sudo] password for hduser:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  efibootmgr libegl1-mesa libfwupd1 libllvm9 libwayland-egl1-mesa
```

Installation de paquet nécessaire pour **ssh.**

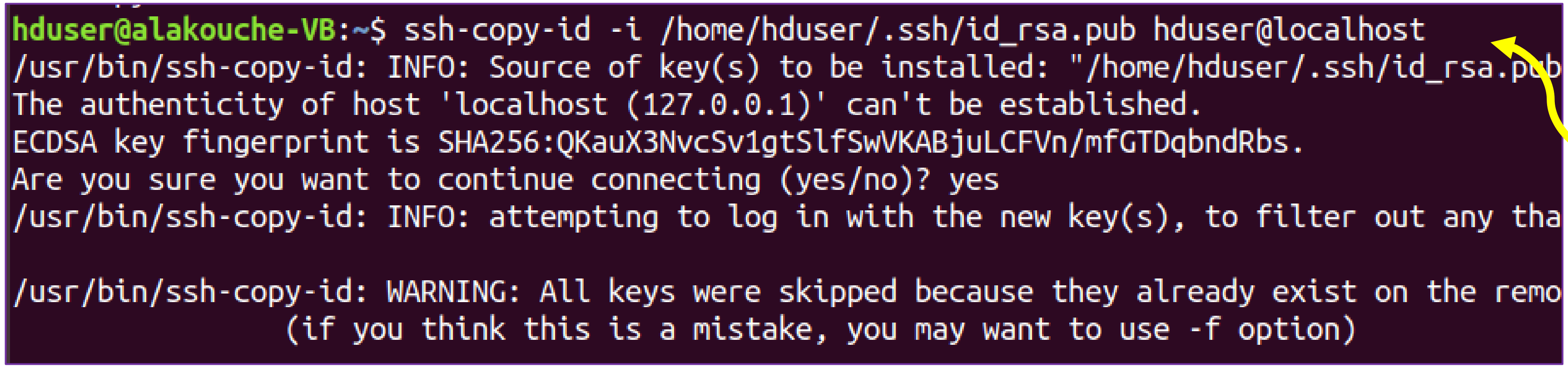


```
hduser@alakouche-VB:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:01DBfrKXsTqdvPCMWg+XQx0xLEKtb5M8K33UP6qWIfA hduser@alakouche-VB
The key's randomart image is:
+---[RSA 2048]----+
|         o+o .o  |
|         .o o .o |
|        .. o ..  |
|       . o+ o. . |
|        S .*.=.. |
|         E.o@.. .|
|          =B**  .|
|         .+@*....|
|        ..o+*+. .|
+----[SHA256]-----+
hduser@alakouche-VB:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hduser@alakouche-VB:~$ chmod 0600 ~/.ssh/authorized_keys
```

Mettre en place la clé **ssh** pour son propre compte.

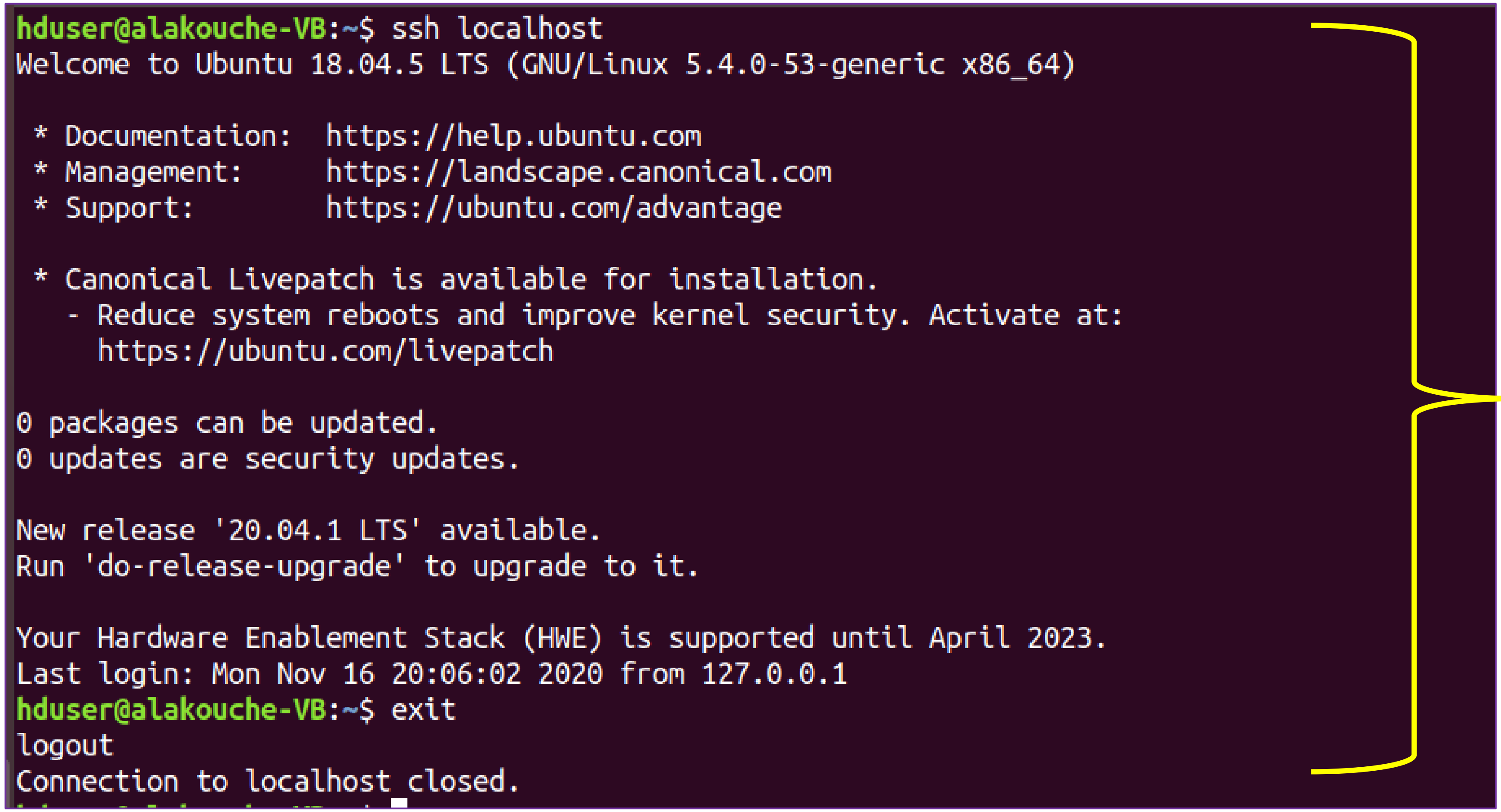
Hadoop nécessite un accès SSH pour gérer les différents nœuds. Bien que nous soyons dans une configuration simple nœud, nous avons besoin de configurer l'accès vers localhost pour l'utilisateur hduser que nous venons de créer précédemment.

Autoriser l'accès au SSH de la machine avec cette nouvelle clé fraîchement créée



```
hduser@alakouche-VB:~$ ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub hduser@localhost
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hduser/.ssh/id_rsa.pub
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:QKauX3NvcSv1gtSlfSwVKABjuLCFVn/mfGTDqbndRbs.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any tha

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remo
                (if you think this is a mistake, you may want to use -f option)
```

On copie la clé public sur le serveur localhost

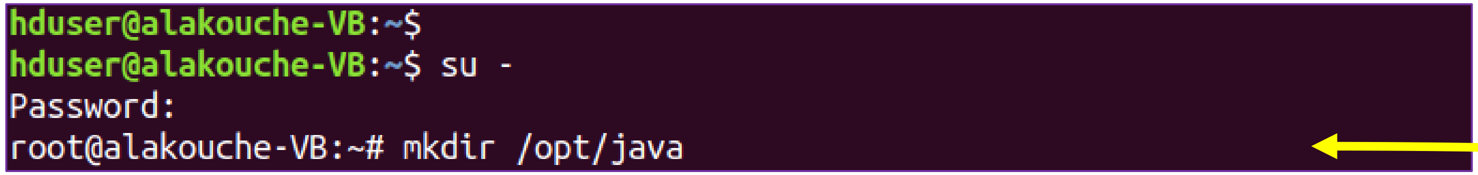


```
hduser@alakouche-VB:~$ ssh localhost
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-53-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Mon Nov 16 20:06:02 2020 from 127.0.0.1
hduser@alakouche-VB:~$ exit
logout
Connection to localhost closed.
```

On teste la connexion à localhost

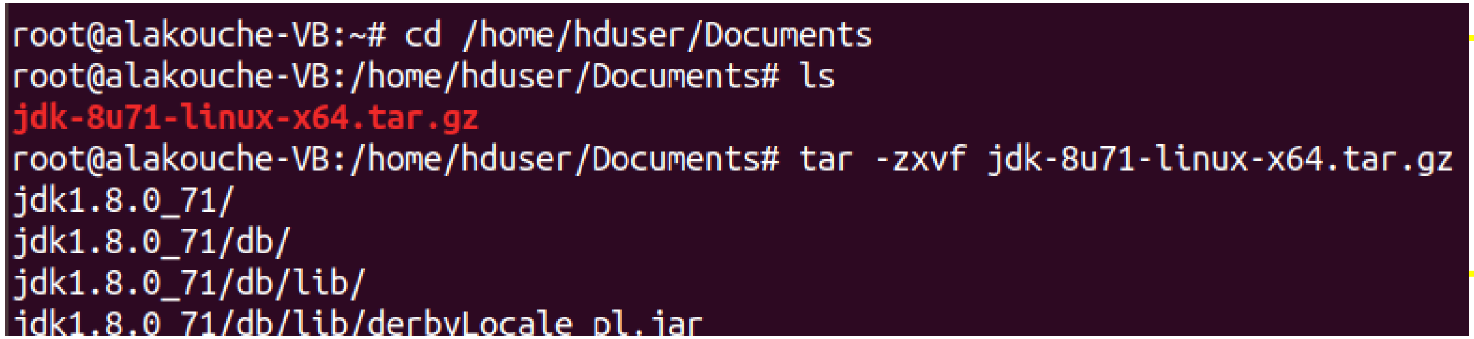
Etape 3 : Installation de JAVA 8

- Hadoop nécessite l’installation de Java. Pour ce TP, la version 8 de Java sera utilisée via la distribution **OpenJDK**



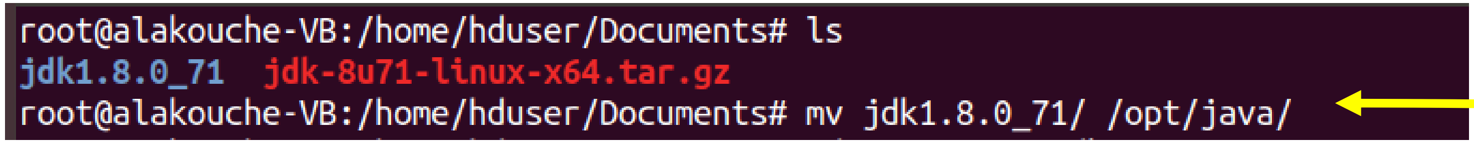
```
hduser@alakouche-VB:~$
hduser@alakouche-VB:~$ su -
Password:
root@alakouche-VB:~# mkdir /opt/java
```

Création de répertoire **/opt/java**



```
root@alakouche-VB:~# cd /home/hduser/Documents
root@alakouche-VB:/home/hduser/Documents# ls
jdk-8u71-linux-x64.tar.gz
root@alakouche-VB:/home/hduser/Documents# tar -zxvf jdk-8u71-linux-x64.tar.gz
jdk1.8.0_71/
jdk1.8.0_71/db/
jdk1.8.0_71/db/lib/
jdk1.8.0_71/db/lib/derbyLocale_pl.jar
```

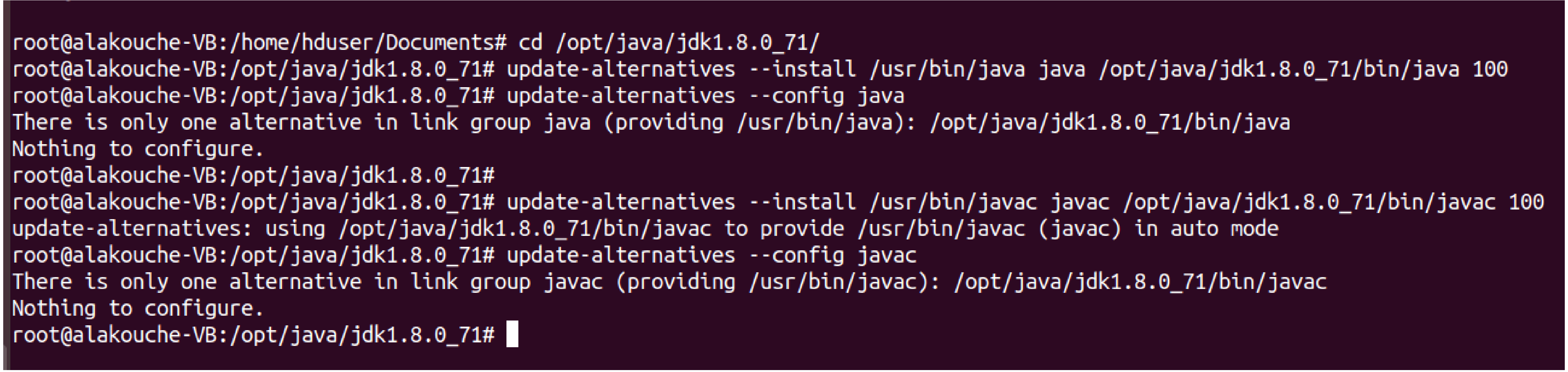
On va ensuite décompresser l’archive jdk8u71linuxx64.tar.gz



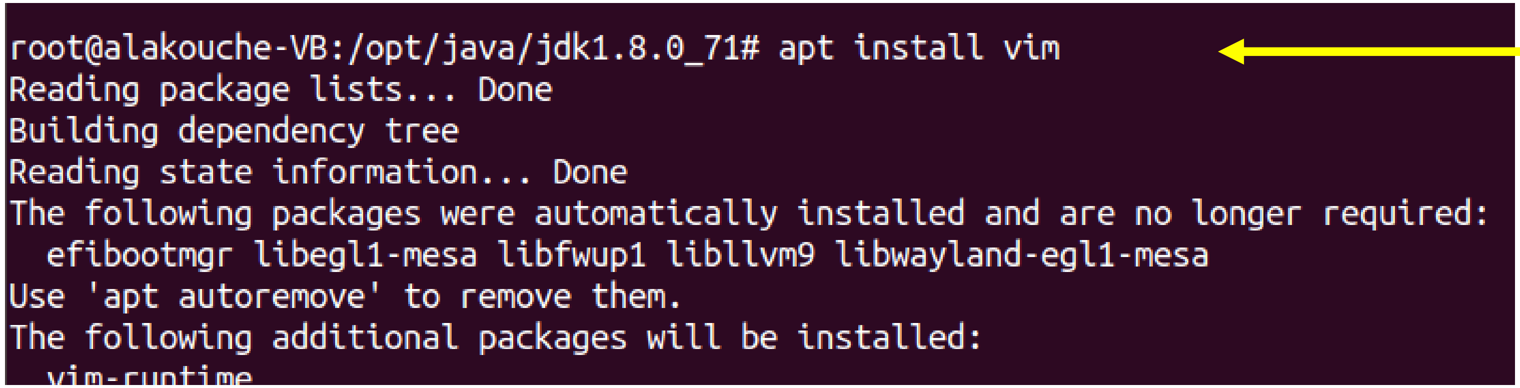
```
root@alakouche-VB:/home/hduser/Documents# ls
jdk1.8.0_71  jdk-8u71-linux-x64.tar.gz
root@alakouche-VB:/home/hduser/Documents# mv jdk1.8.0_71/ /opt/java/
```

On déplace le jdk vers /opt/java/

On informe le système où java et ses exécutables sont installés.

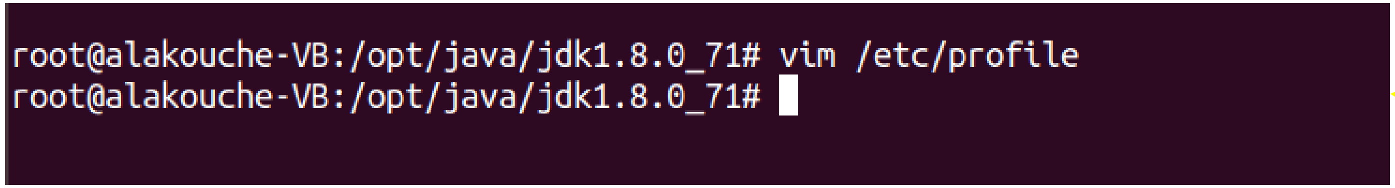


```
root@alakouche-VB:/home/hduser/Documents# cd /opt/java/jdk1.8.0_71/
root@alakouche-VB:/opt/java/jdk1.8.0_71# update-alternatives --install /usr/bin/java java /opt/java/jdk1.8.0_71/bin/java 100
root@alakouche-VB:/opt/java/jdk1.8.0_71# update-alternatives --config java
There is only one alternative in link group java (providing /usr/bin/java): /opt/java/jdk1.8.0_71/bin/java
Nothing to configure.
root@alakouche-VB:/opt/java/jdk1.8.0_71#
root@alakouche-VB:/opt/java/jdk1.8.0_71# update-alternatives --install /usr/bin/javac javac /opt/java/jdk1.8.0_71/bin/javac 100
update-alternatives: using /opt/java/jdk1.8.0_71/bin/javac to provide /usr/bin/javac (javac) in auto mode
root@alakouche-VB:/opt/java/jdk1.8.0_71# update-alternatives --config javac
There is only one alternative in link group javac (providing /usr/bin/javac): /opt/java/jdk1.8.0_71/bin/javac
Nothing to configure.
root@alakouche-VB:/opt/java/jdk1.8.0_71#
```

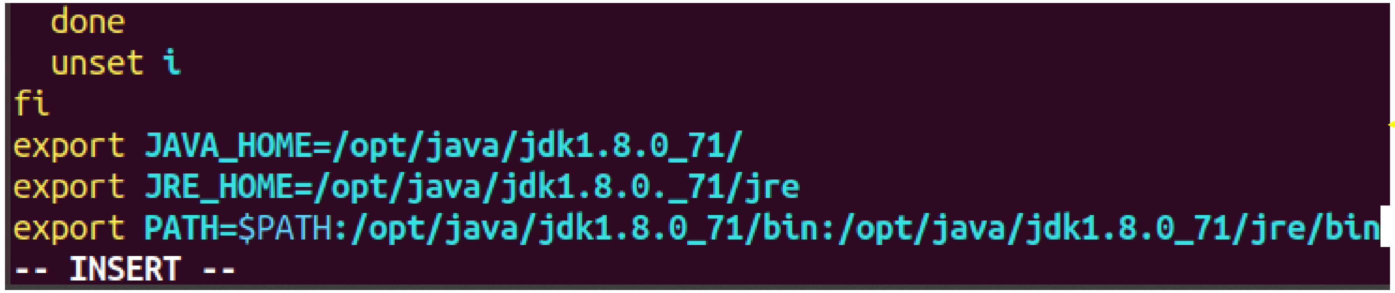


```
root@alakouche-VB:/opt/java/jdk1.8.0_71# apt install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  efibootmgr libegl1-mesa libfwup1 libllvm9 libwayland-egl1-mesa
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  vim-runtime
```

Installation de l'éditeur de texte « **Vim** ».



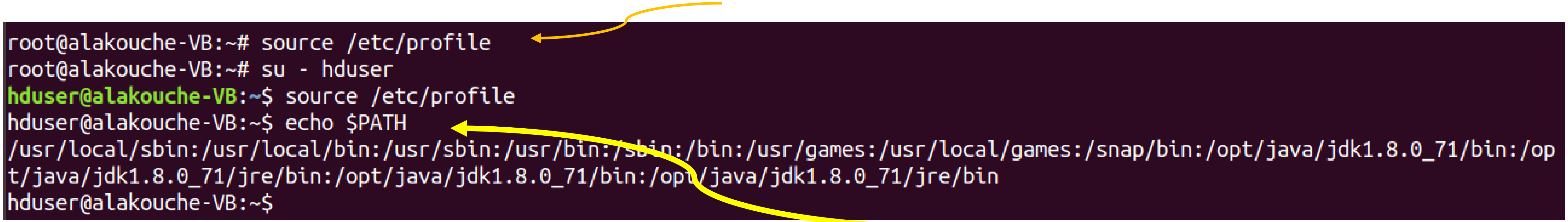
```
root@alakouche-VB:/opt/java/jdk1.8.0_71# vim /etc/profile
root@alakouche-VB:/opt/java/jdk1.8.0_71#
```



```
  done
  unset i
fi
export JAVA_HOME=/opt/java/jdk1.8.0_71/
export JRE_HOME=/opt/java/jdk1.8.0._71/jre
export PATH=$PATH:/opt/java/jdk1.8.0_71/bin:/opt/java/jdk1.8.0_71/jre/bin
-- INSERT --
```

Mettre en place de manière permanente les variables d'environnement JAVA pour tous les utilisateurs

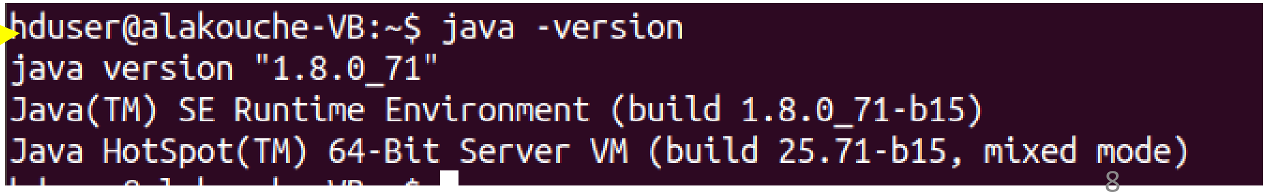
On recharge le fichier **/etc/profile**



```
root@alakouche-VB:~# source /etc/profile
root@alakouche-VB:~# su - hduser
hduser@alakouche-VB:~$ source /etc/profile
hduser@alakouche-VB:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/opt/java/jdk1.8.0_71/bin:/op
t/java/jdk1.8.0_71/jre/bin:/opt/java/jdk1.8.0_71/bin:/opt/java/jdk1.8.0_71/jre/bin
hduser@alakouche-VB:~$
```

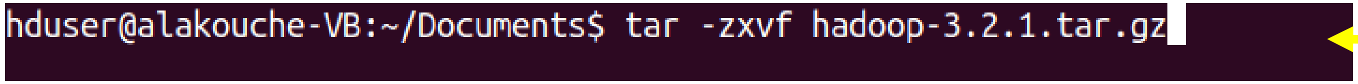
On teste la mise en place des variables d'environnement dans le terminal hadoop

Assurent que la version Java est correctement installée



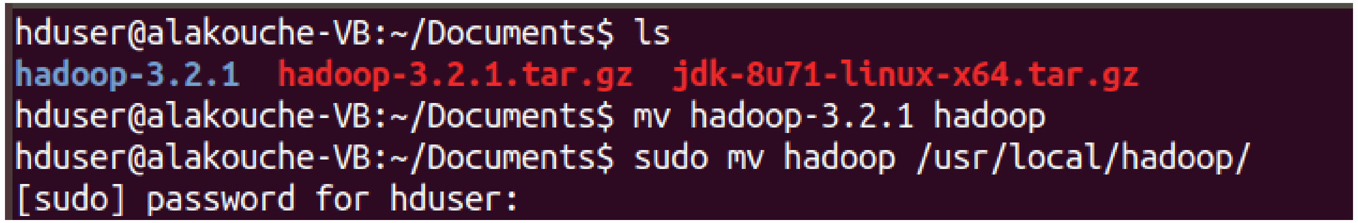
```
hduser@alakouche-VB:~$ java -version
java version "1.8.0_71"
Java(TM) SE Runtime Environment (build 1.8.0_71-b15)
Java HotSpot(TM) 64-Bit Server VM (build 25.71-b15, mixed mode)
hduser@alakouche-VB:~$
```

Etape 4 : Installation d'Apache Hadoop 3.2.1

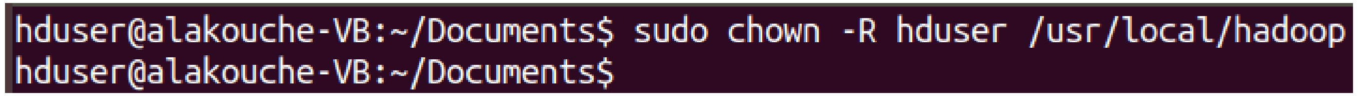


```
hduser@alakouche-VB:~/Documents$ tar -zxvf hadoop-3.2.1.tar.gz
```

On décompresse l’archive : hadoop3.1.2.tar.gz

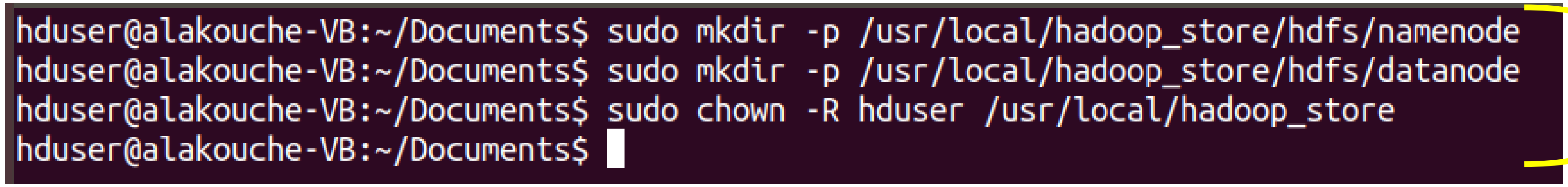


```
hduser@alakouche-VB:~/Documents$ ls
hadoop-3.2.1  hadoop-3.2.1.tar.gz  jdk-8u71-linux-x64.tar.gz
hduser@alakouche-VB:~/Documents$ mv hadoop-3.2.1 hadoop
hduser@alakouche-VB:~/Documents$ sudo mv hadoop /usr/local/hadoop/
[sudo] password for hduser:
```



```
hduser@alakouche-VB:~/Documents$ sudo chown -R hduser /usr/local/hadoop
hduser@alakouche-VB:~/Documents$
```

On affecte les droits à notre utilisateur **hduse.**

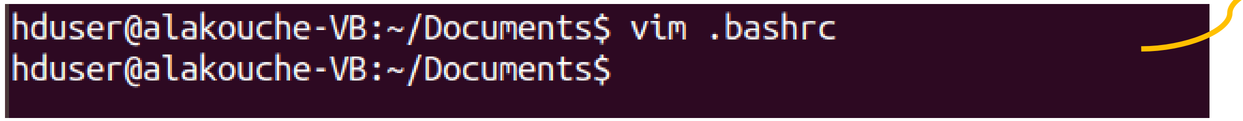


```
hduser@alakouche-VB:~/Documents$ sudo mkdir -p /usr/local/hadoop_store/hdfs/namenode
hduser@alakouche-VB:~/Documents$ sudo mkdir -p /usr/local/hadoop_store/hdfs/datanode
hduser@alakouche-VB:~/Documents$ sudo chown -R hduser /usr/local/hadoop_store
hduser@alakouche-VB:~/Documents$
```

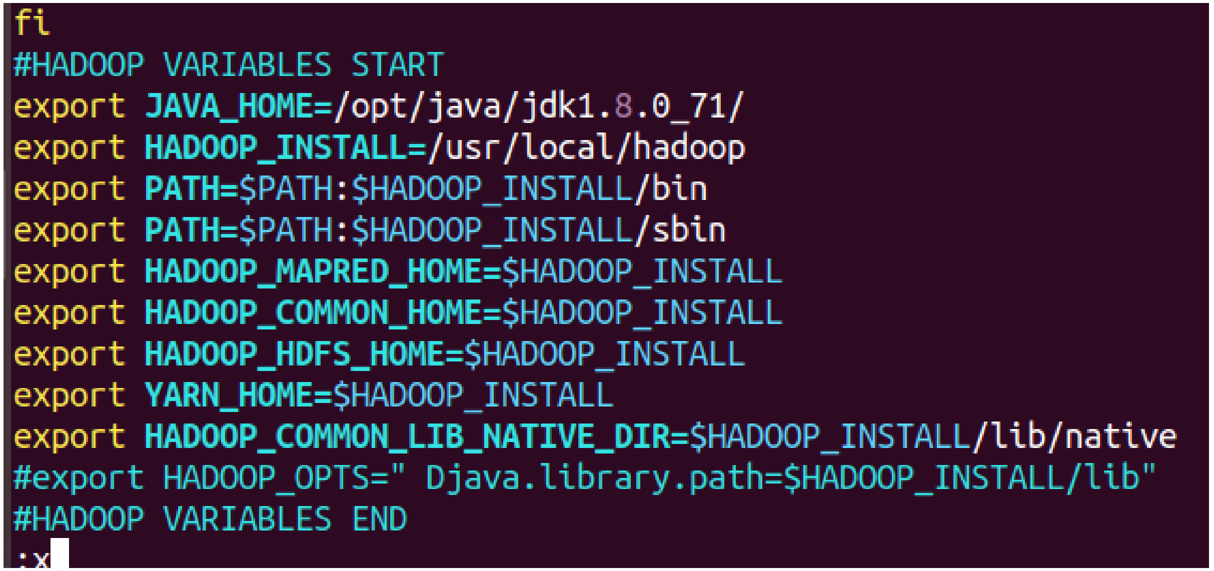
Création de datanode et namenode. On affecte ensuite les droits à notre utilisateur **hduse**.

Etape 5 : Configuration d'Apache Hadoop 3.2.1

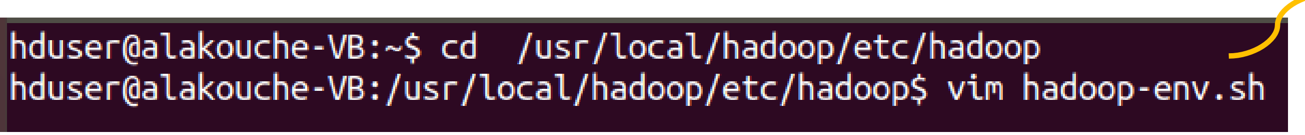
- Il faut maintenant définir la configuration de Hadoop et pour cela plusieurs fichiers de configurations doivent être modifiés



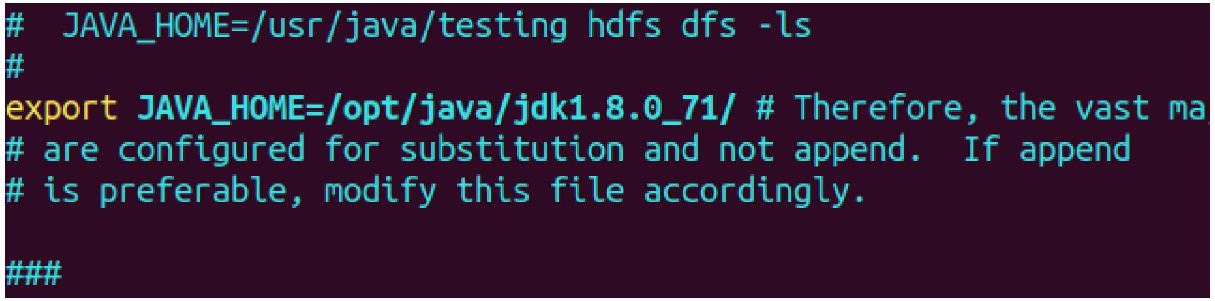
```
hduser@alakouche-VB:~/Documents$ vim .bashrc
hduser@alakouche-VB:~/Documents$
```



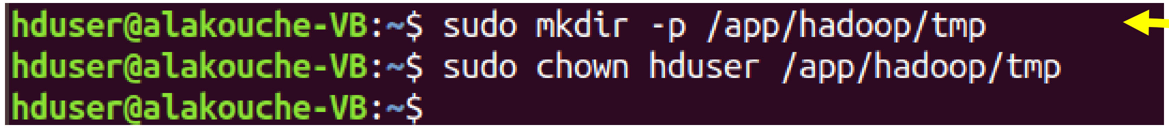
```
fi
#HADOOP VARIABLES START
export JAVA_HOME=/opt/java/jdk1.8.0_71/
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
#export HADOOP_OPTS=" Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END
:x
```



```
hduser@alakouche-VB:~$ cd  /usr/local/hadoop/etc/hadoop
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ vim hadoop-env.sh
```



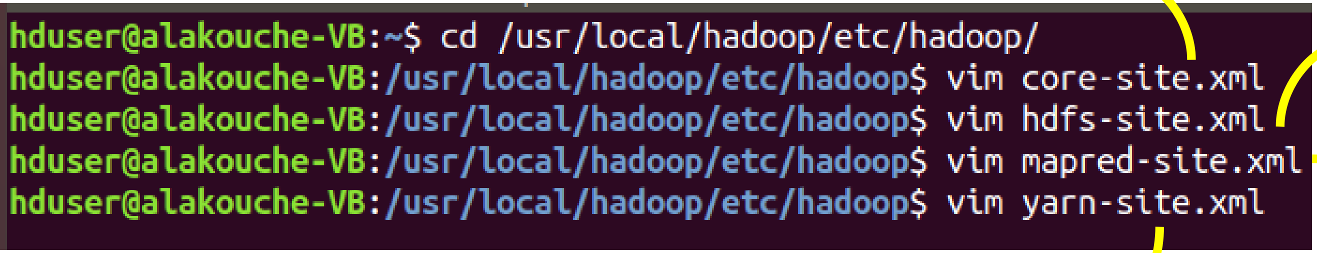
```
#  JAVA_HOME=/usr/java/testing hdfs dfs -ls
#
export JAVA_HOME=/opt/java/jdk1.8.0_71/ # Therefore, the vast ma
# are configured for substitution and not append.  If append
# is preferable, modify this file accordingly.

###
```



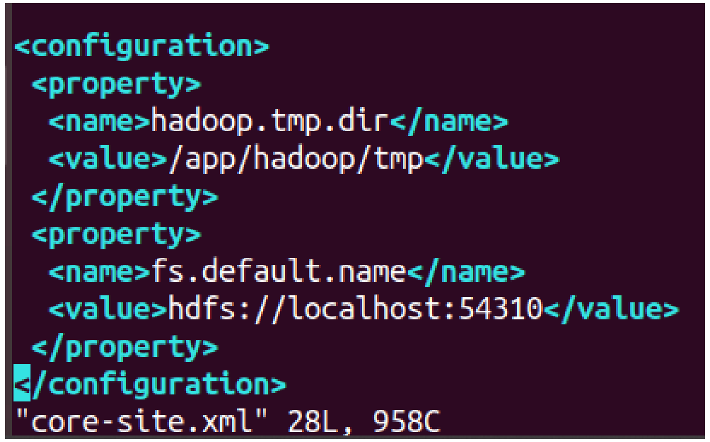
```
hduser@alakouche-VB:~$ sudo mkdir -p /app/hadoop/tmp
hduser@alakouche-VB:~$ sudo chown hduser /app/hadoop/tmp
hduser@alakouche-VB:~$
```

Création de répertoire des fichiers temporaires de Hadoop

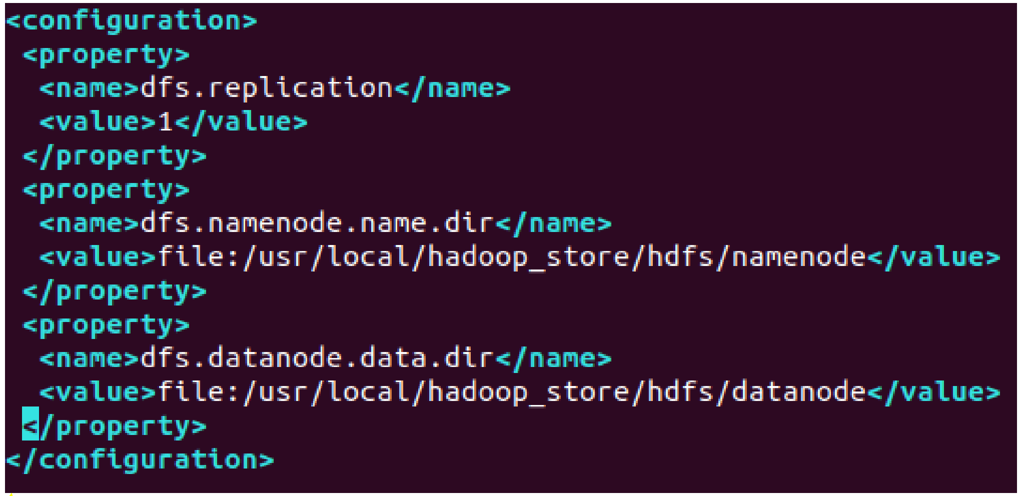
- Modification des autres fichiers de configurations : on ajoute les lignes suivantes entre les balises de configurations.



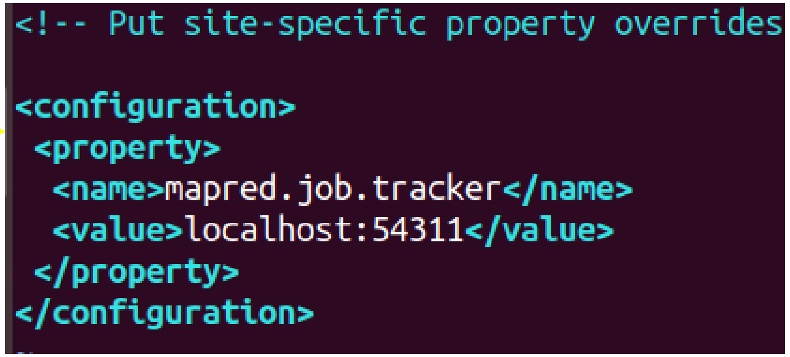
```
hduser@alakouche-VB:~$ cd /usr/local/hadoop/etc/hadoop/
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ vim core-site.xml
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ vim hdfs-site.xml
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ vim mapred-site.xml
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ vim yarn-site.xml
```



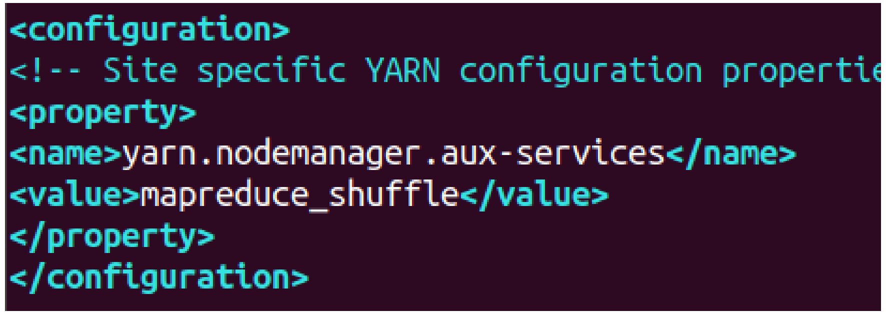
```
<configuration>
 <property>
  <name>hadoop.tmp.dir</name>
  <value>/app/hadoop/tmp</value>
 </property>
 <property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:54310</value>
 </property>
</configuration>
"core-site.xml" 28L, 958C
```



```
<configuration>
 <property>
  <name>dfs.replication</name>
  <value>1</value>
 </property>
 <property>
  <name>dfs.namenode.name.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
 </property>
 <property>
  <name>dfs.datanode.data.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
 </property>
</configuration>
```

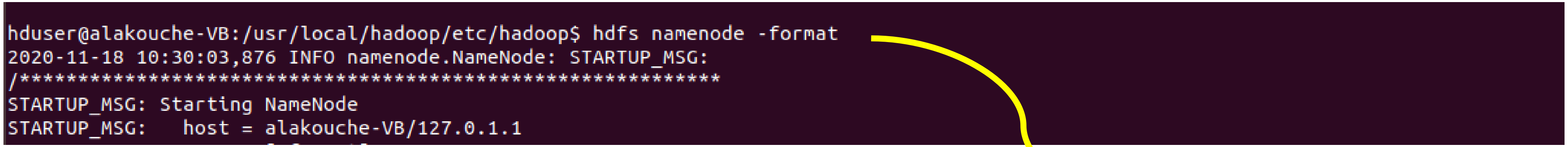


```
<!-- Put site-specific property overrides

<configuration>
 <property>
  <name>mapred.job.tracker</name>
  <value>localhost:54311</value>
 </property>
</configuration>
```

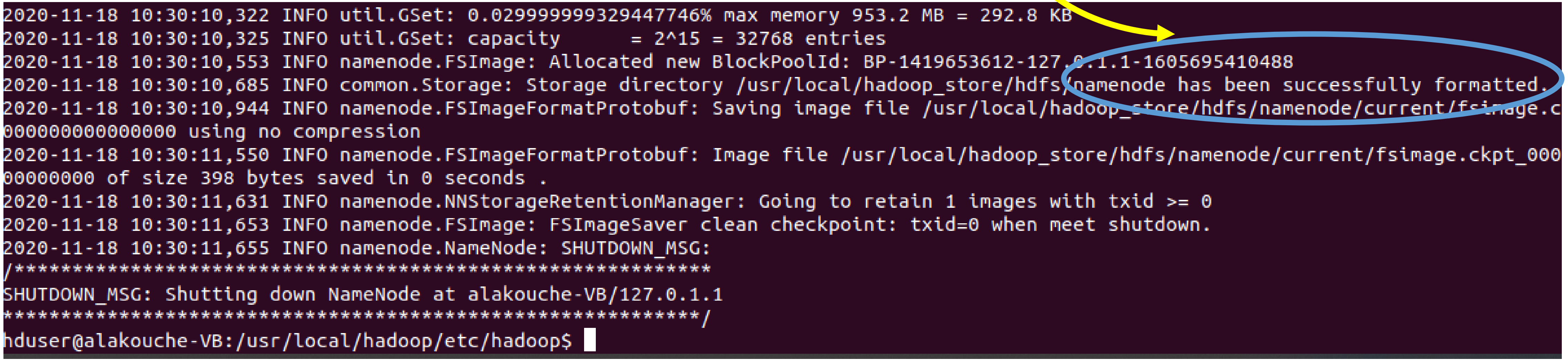


```
<configuration>
<!-- Site specific YARN configuration propertie
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
</configuration>
```

- Avant de démarrer le serveur Hadoop, il faut formater le système de fichiers HDFS.

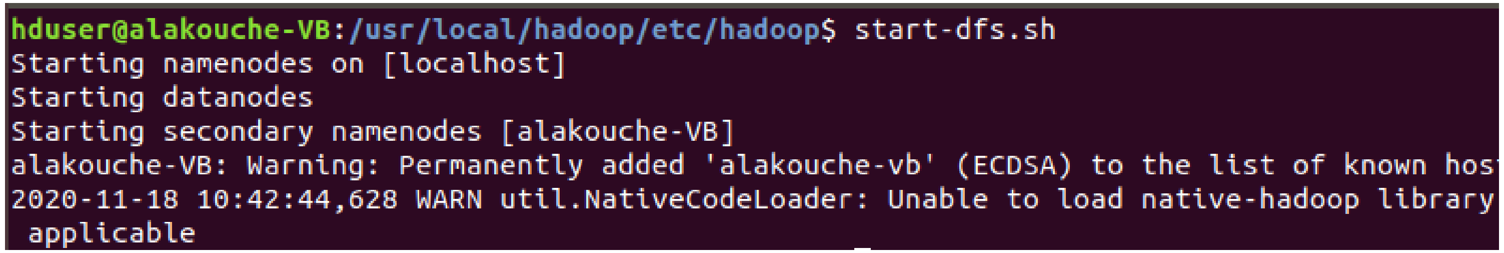


```
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ hdfs namenode -format
2020-11-18 10:30:03,876 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = alakouche-VB/127.0.1.1
```

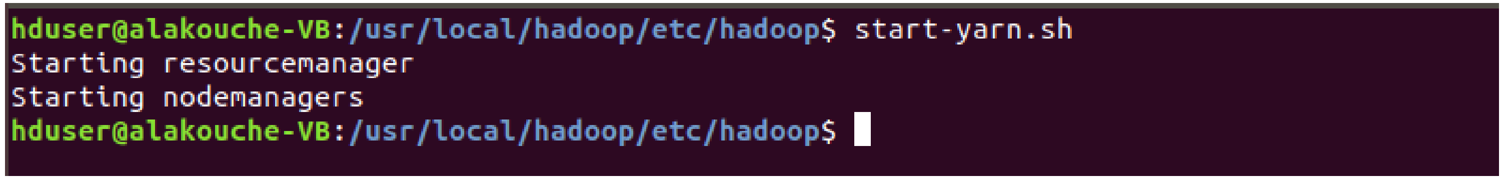


```
2020-11-18 10:30:10,322 INFO util.GSet: 0.029999999329447746% max memory 953.2 MB = 292.8 KB
2020-11-18 10:30:10,325 INFO util.GSet: capacity      = 2^15 = 32768 entries
2020-11-18 10:30:10,553 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1419653612-127.0.1.1-1605695410488
2020-11-18 10:30:10,685 INFO common.Storage: Storage directory /usr/local/hadoop_store/hdfs/namenode has been successfully formatted.
2020-11-18 10:30:10,944 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.c
000000000000000 using no compression
2020-11-18 10:30:11,550 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.ckpt_000
00000000 of size 398 bytes saved in 0 seconds .
2020-11-18 10:30:11,631 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2020-11-18 10:30:11,653 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2020-11-18 10:30:11,655 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at alakouche-VB/127.0.1.1
************************************************************/
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$
```

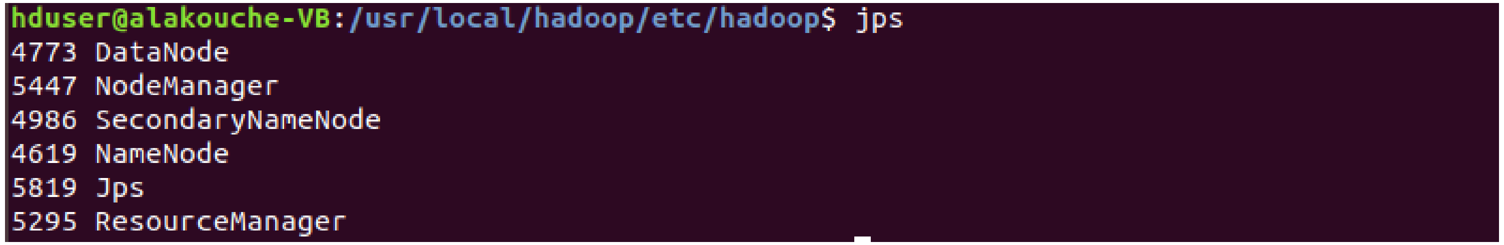
- Démarrage de Hadoop



```
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [alakouche-VB]
alakouche-VB: Warning: Permanently added 'alakouche-vb' (ECDSA) to the list of known hos
2020-11-18 10:42:44,628 WARN util.NativeCodeLoader: Unable to load native-hadoop library
 applicable
```



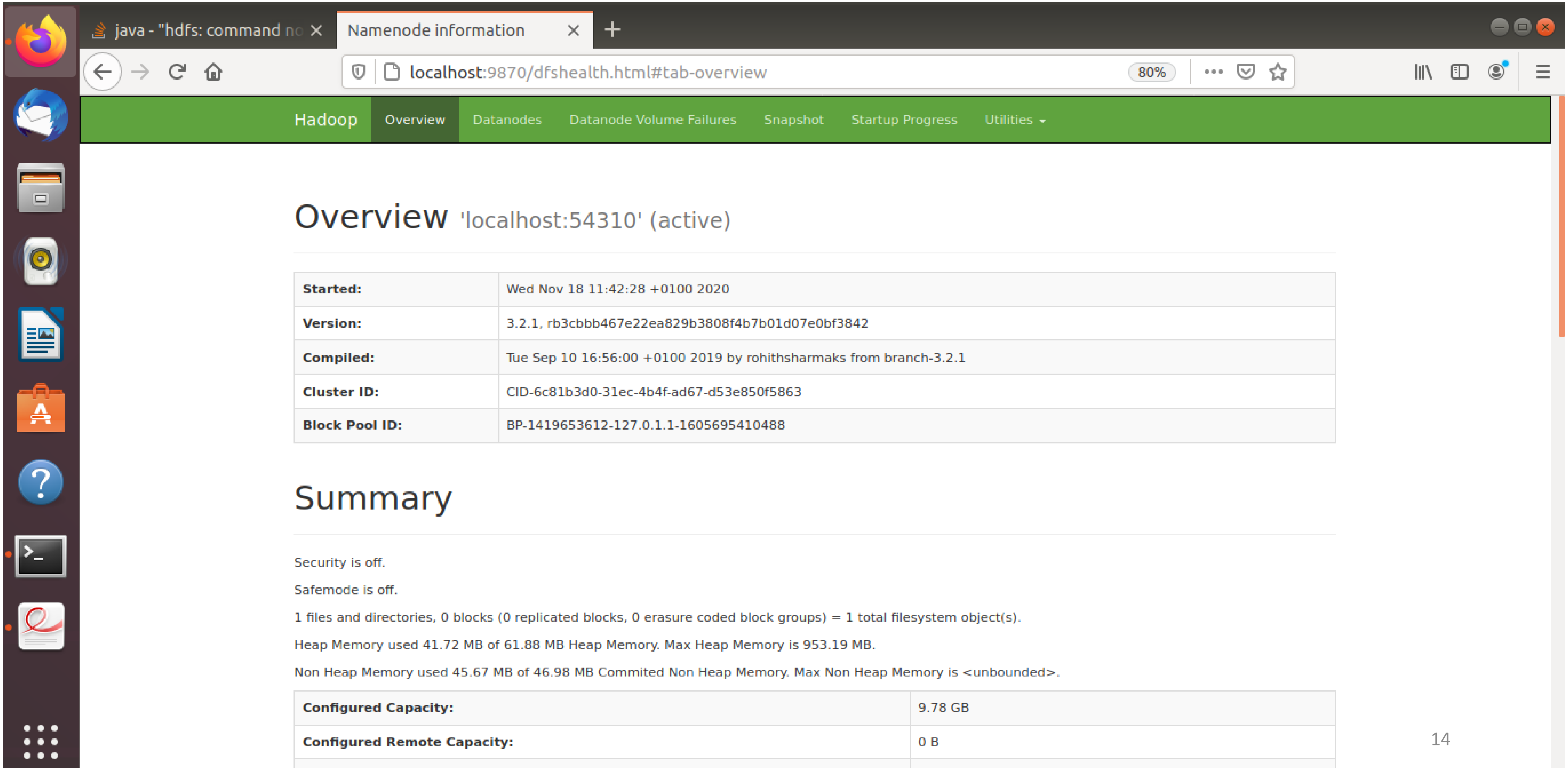
```
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$
```



```
hduser@alakouche-VB:/usr/local/hadoop/etc/hadoop$ jps
4773 DataNode
5447 NodeManager
4986 SecondaryNameNode
4619 NameNode
5819 Jps
5295 ResourceManager
```

- Vérification de l’installation
  On peut maintenant accéder à l’interface web Hadoop .

- interface graphique accessible par notre navigateur



Overview 'localhost:54310' (active)

Started:	Wed Nov 18 11:42:28 +0100 2020
Version:	3.2.1, rb3cbbb467e22ea829b3808f4b7b01d07e0bf3842
Compiled:	Tue Sep 10 16:56:00 +0100 2019 by rohithsharmaks from branch-3.2.1
Cluster ID:	CID-6c81b3d0-31ec-4b4f-ad67-d53e850f5863
Block Pool ID:	BP-1419653612-127.0.1.1-1605695410488

Summary

Security is off.

Safemode is off.

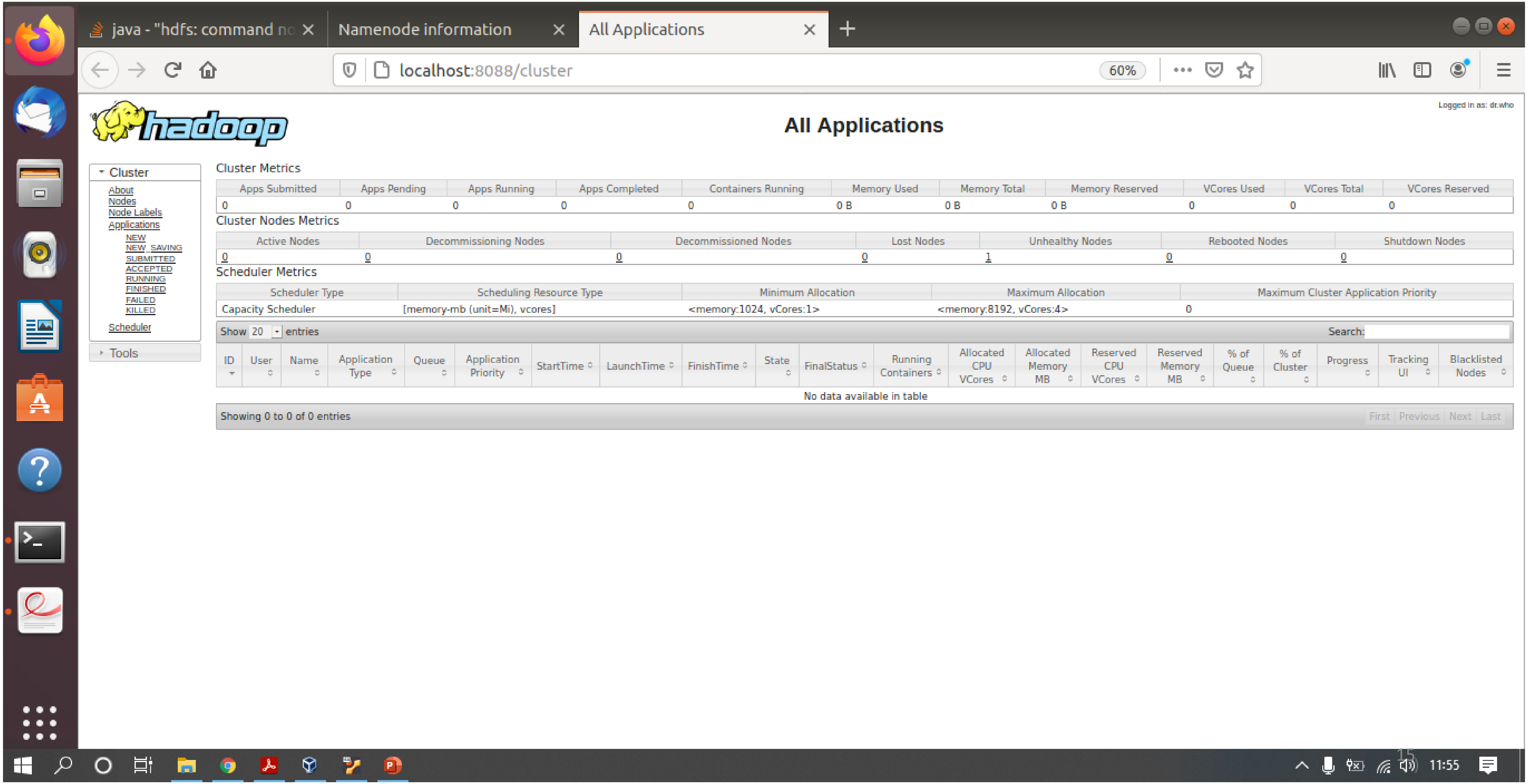
1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).

Heap Memory used 41.72 MB of 61.88 MB Heap Memory. Max Heap Memory is 953.19 MB.

Non Heap Memory used 45.67 MB of 46.98 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	9.78 GB
Configured Remote Capacity:	0 B

- également on peut visualiser l'avancement et les résultats de notre Jobs



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
0	0	0	0	0	0 B	0 B	0 B	0	0	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
0	0	0	0	1	0	0

Scheduler Metrics

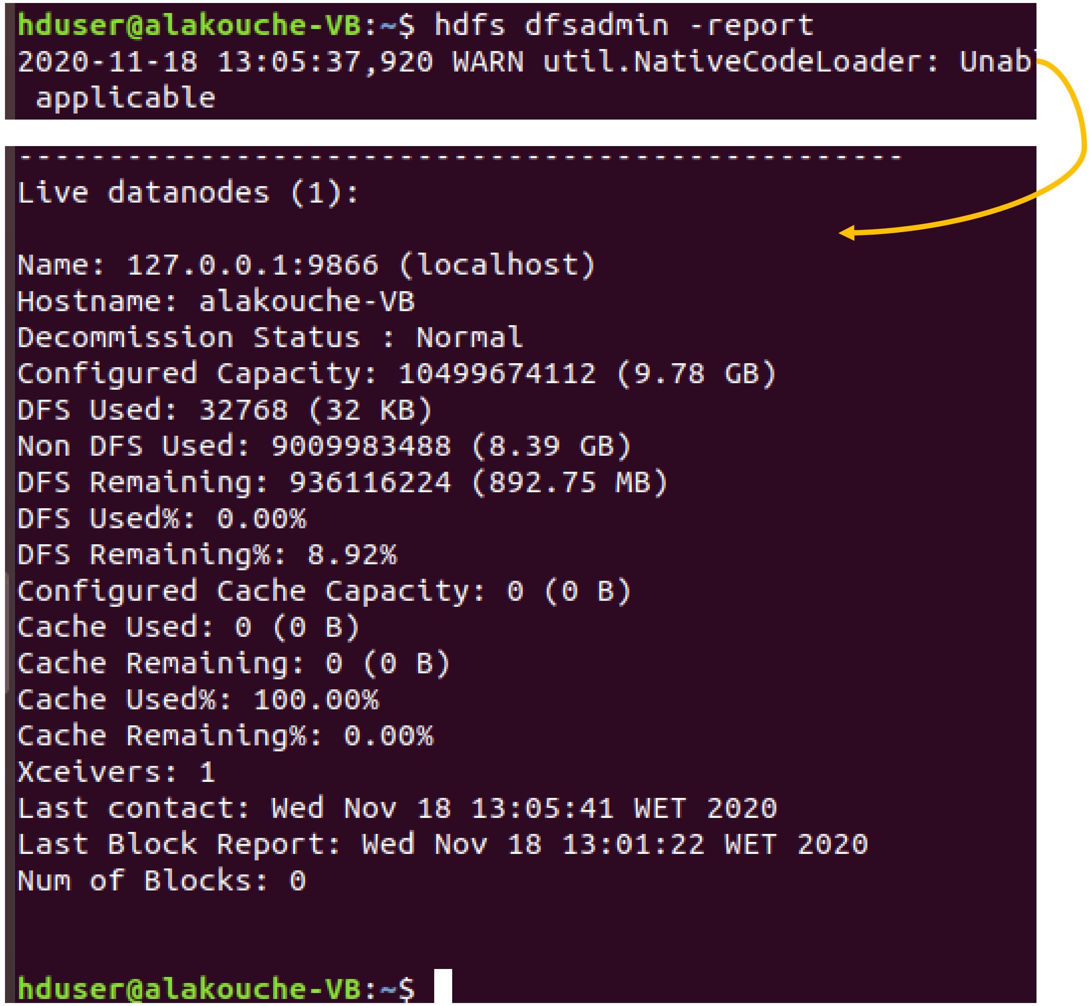
Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0

Show 20 entries Search:

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB	% of Queue	% of Cluster	Progress	Tracking UI	Blacklisted Nodes
No data available in table																				

Showing 0 to 0 of 0 entries First Previous Next Last

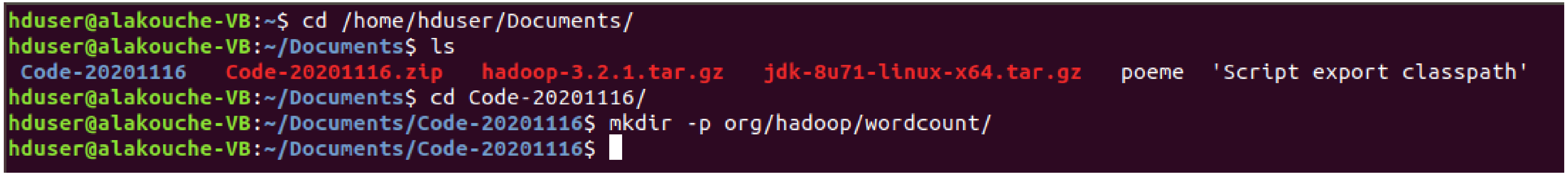
II Exécution du programme MapReduce « Word Count » dans le cluster à nœud unique



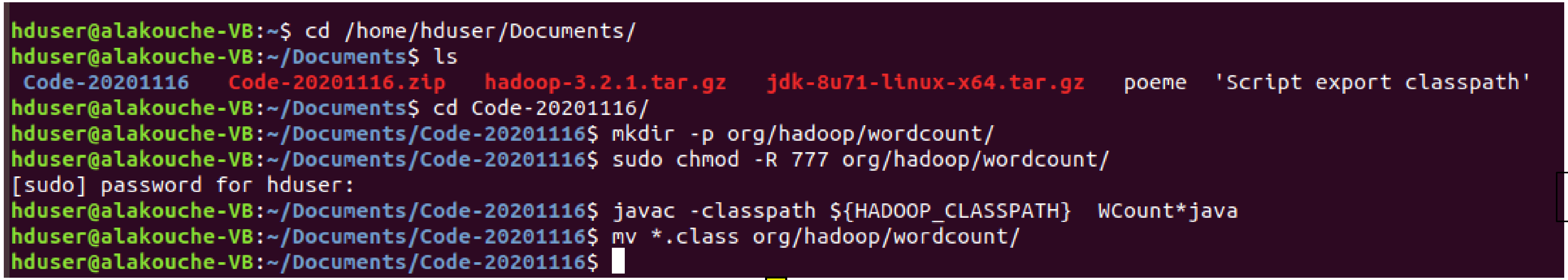
```
hduser@alakouche-VB:~$ hdfs dfsadmin -report
2020-11-18 13:05:37,920 WARN util.NativeCodeLoader: Unab
 applicable
```

```
-------------------------------------------------
Live datanodes (1):

Name: 127.0.0.1:9866 (localhost)
Hostname: alakouche-VB
Decommission Status : Normal
Configured Capacity: 10499674112 (9.78 GB)
DFS Used: 32768 (32 KB)
Non DFS Used: 9009983488 (8.39 GB)
DFS Remaining: 936116224 (892.75 MB)
DFS Used%: 0.00%
DFS Remaining%: 8.92%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Wed Nov 18 13:05:41 WET 2020
Last Block Report: Wed Nov 18 13:01:22 WET 2020
Num of Blocks: 0


hduser@alakouche-VB:~$
```

On lance cette commande afin de vérifier le bon fonctionnement de notre noeud



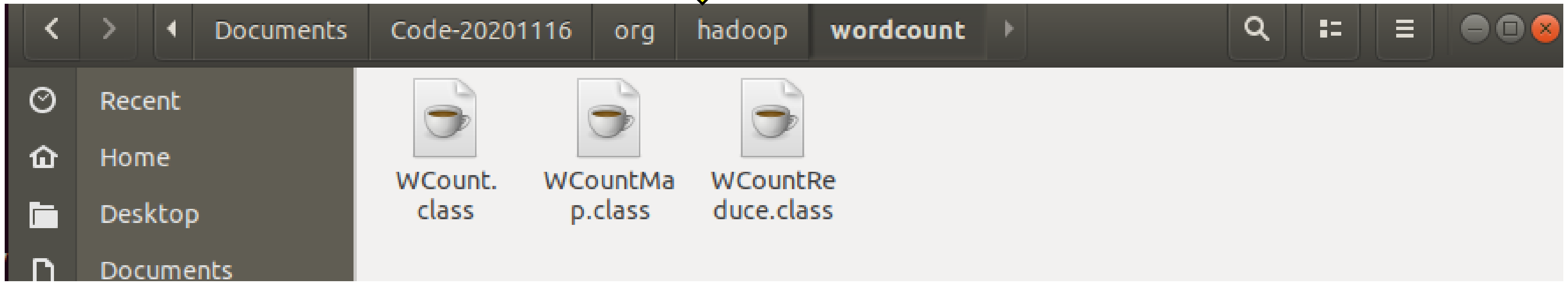
```
hduser@alakouche-VB:~$ cd /home/hduser/Documents/
hduser@alakouche-VB:~/Documents$ ls
 Code-20201116   Code-20201116.zip   hadoop-3.2.1.tar.gz   jdk-8u71-linux-x64.tar.gz   poeme  'Script export classpath'
hduser@alakouche-VB:~/Documents$ cd Code-20201116/
hduser@alakouche-VB:~/Documents/Code-20201116$ mkdir -p org/hadoop/wordcount/
hduser@alakouche-VB:~/Documents/Code-20201116$ 
```



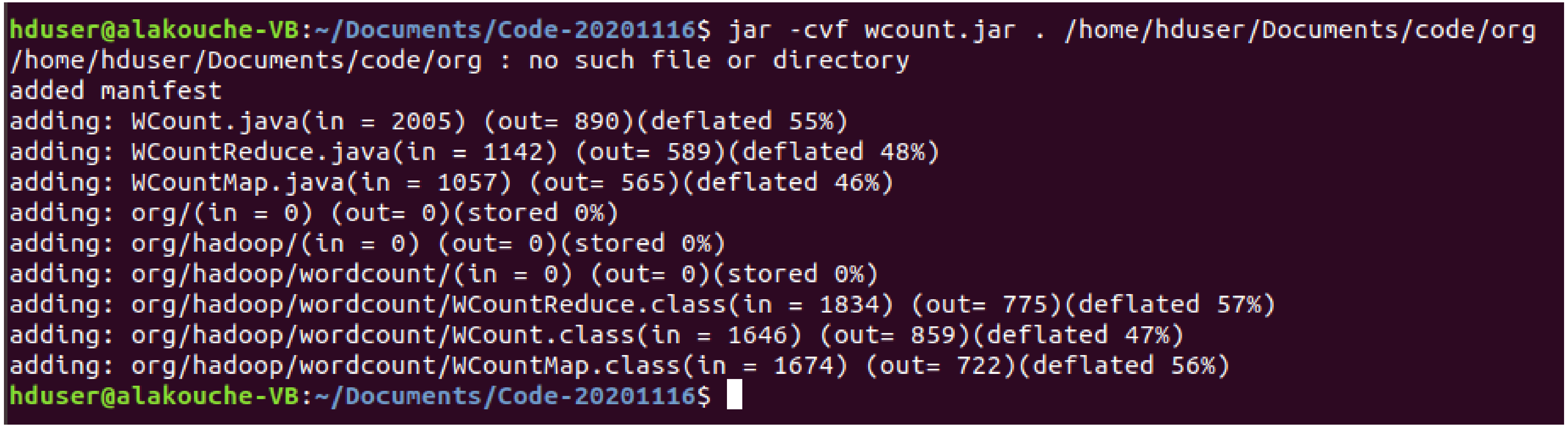
```
hduser@alakouche-VB:~$ cd /home/hduser/Documents/
hduser@alakouche-VB:~/Documents$ ls
 Code-20201116   Code-20201116.zip   hadoop-3.2.1.tar.gz   jdk-8u71-linux-x64.tar.gz   poeme  'Script export classpath'
hduser@alakouche-VB:~/Documents$ cd Code-20201116/
hduser@alakouche-VB:~/Documents/Code-20201116$ mkdir -p org/hadoop/wordcount/
hduser@alakouche-VB:~/Documents/Code-20201116$ sudo chmod -R 777 org/hadoop/wordcount/
[sudo] password for hduser:
hduser@alakouche-VB:~/Documents/Code-20201116$ javac -classpath ${HADOOP_CLASSPATH}  WCount*java
hduser@alakouche-VB:~/Documents/Code-20201116$ mv *.class org/hadoop/wordcount/
hduser@alakouche-VB:~/Documents/Code-20201116$ 
```

Compilation de programme

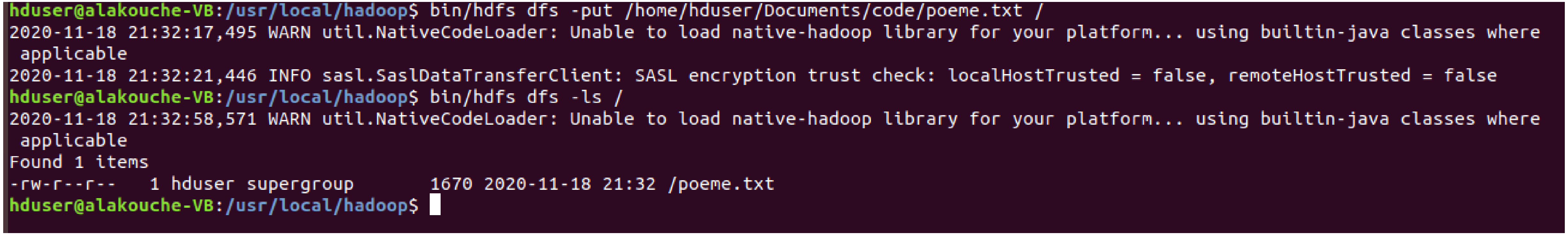
La compilation de programme a généré trois fichiers **.class**



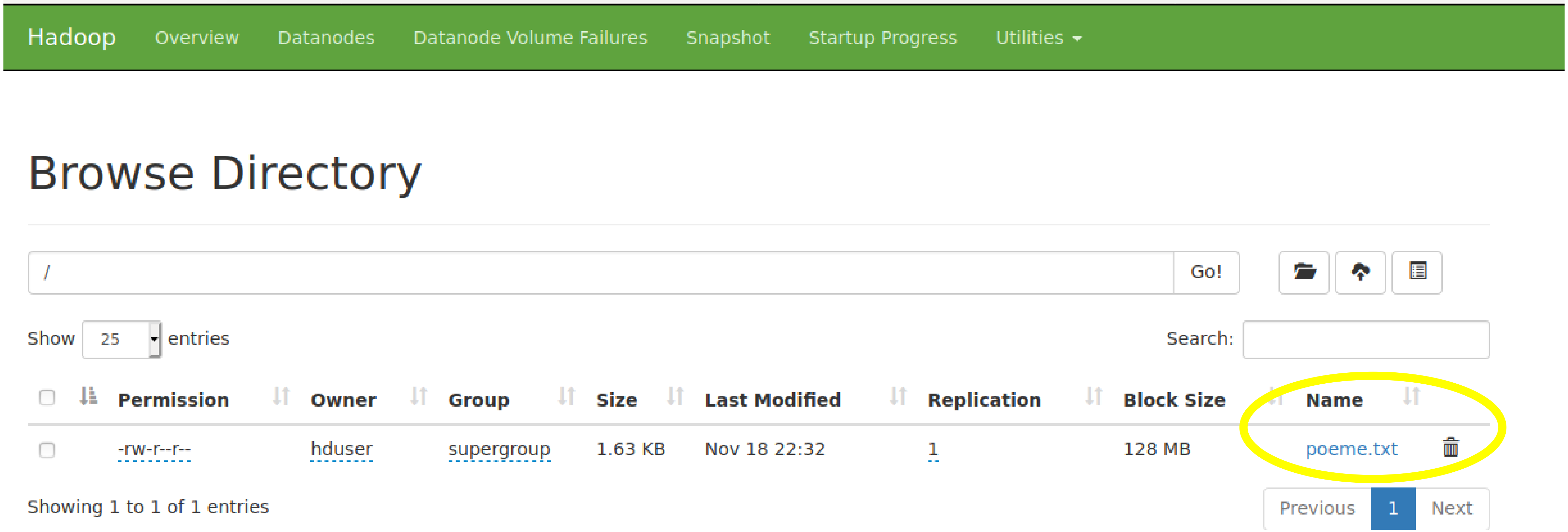
On génère le jar file



```
hduser@alakouche-VB:~/Documents/Code-20201116$ jar -cvf wcount.jar . /home/hduser/Documents/code/org
/home/hduser/Documents/code/org : no such file or directory
added manifest
adding: WCount.java(in = 2005) (out= 890)(deflated 55%)
adding: WCountReduce.java(in = 1142) (out= 589)(deflated 48%)
adding: WCountMap.java(in = 1057) (out= 565)(deflated 46%)
adding: org/(in = 0) (out= 0)(stored 0%)
adding: org/hadoop/(in = 0) (out= 0)(stored 0%)
adding: org/hadoop/wordcount/(in = 0) (out= 0)(stored 0%)
adding: org/hadoop/wordcount/WCountReduce.class(in = 1834) (out= 775)(deflated 57%)
adding: org/hadoop/wordcount/WCount.class(in = 1646) (out= 859)(deflated 47%)
adding: org/hadoop/wordcount/WCountMap.class(in = 1674) (out= 722)(deflated 56%)
hduser@alakouche-VB:~/Documents/Code-20201116$ 
```



```
hduser@alakouche-VB:/usr/local/hadoop$ bin/hdfs dfs -put /home/hduser/Documents/code/poeme.txt /
2020-11-18 21:32:17,495 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
2020-11-18 21:32:21,446 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
hduser@alakouche-VB:/usr/local/hadoop$ bin/hdfs dfs -ls /
2020-11-18 21:32:58,571 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
Found 1 items
-rw-r--r--   1 hduser supergroup       1670 2020-11-18 21:32 /poeme.txt
hduser@alakouche-VB:/usr/local/hadoop$ 
```



Browse Directory

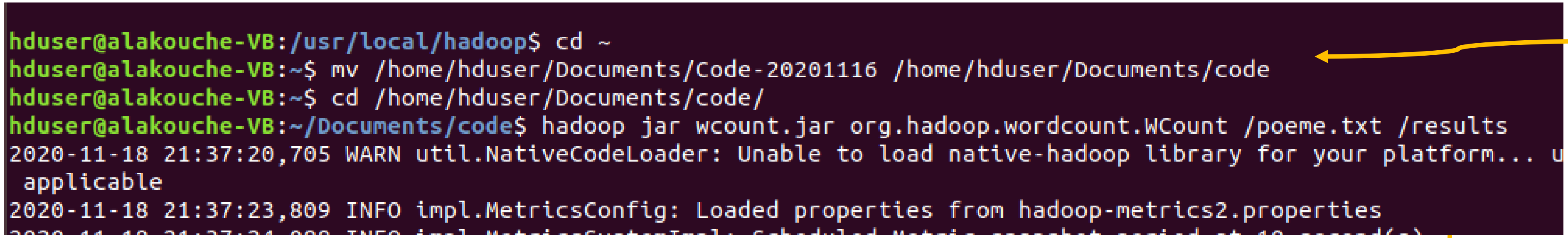
/

Show 25 entries

Search:

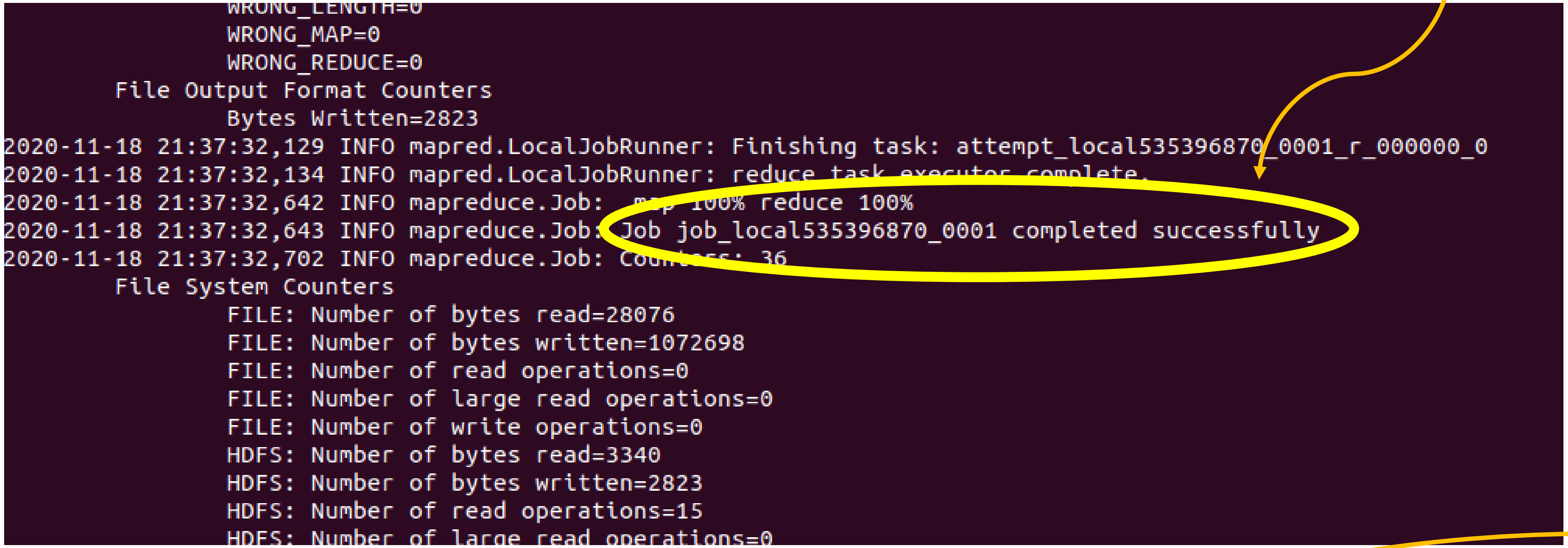
□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	-rw-r--r--	hduser	supergroup	1.63 KB	Nov 18 22:32	1	128 MB	poeme.txt

Showing 1 to 1 of 1 entries



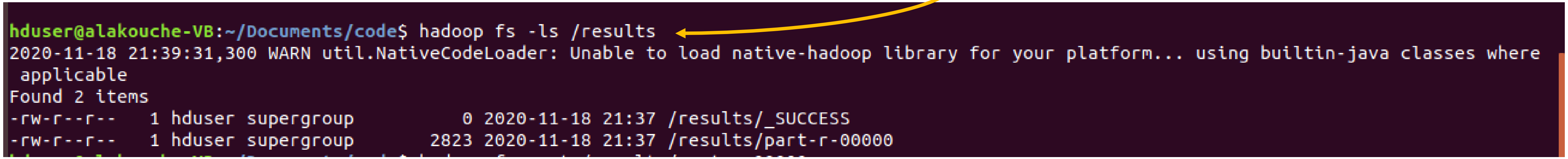
```
hduser@alakouche-VB:/usr/local/hadoop$ cd ~
hduser@alakouche-VB:~$ mv /home/hduser/Documents/Code-20201116 /home/hduser/Documents/code
hduser@alakouche-VB:~$ cd /home/hduser/Documents/code/
hduser@alakouche-VB:~/Documents/code$ hadoop jar wcount.jar org.hadoop.wordcount.WCount /poeme.txt /results
2020-11-18 21:37:20,705 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... u
 applicable
2020-11-18 21:37:23,809 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2020-11-18 21:37:24,888 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
```

On renomme le répertoire pour plus de simplicité



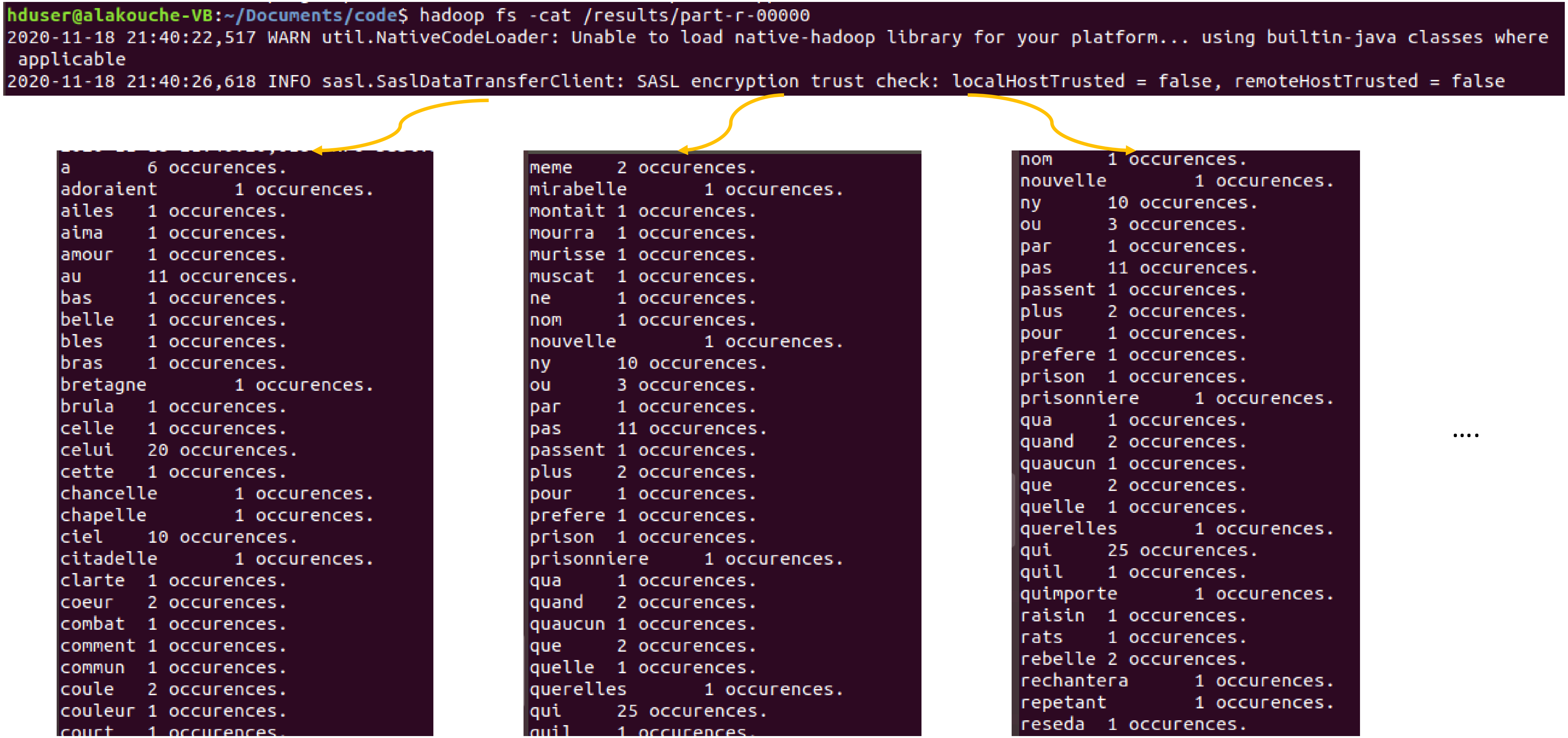
```
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Output Format Counters
                Bytes Written=2823
2020-11-18 21:37:32,129 INFO mapred.LocalJobRunner: Finishing task: attempt_local535396870_0001_r_000000_0
2020-11-18 21:37:32,134 INFO mapred.LocalJobRunner: reduce task executor complete.
2020-11-18 21:37:32,642 INFO mapreduce.Job:  map 100% reduce 100%
2020-11-18 21:37:32,643 INFO mapreduce.Job: Job job_local535396870_0001 completed successfully
2020-11-18 21:37:32,702 INFO mapreduce.Job: Counters: 36
        File System Counters
                FILE: Number of bytes read=28076
                FILE: Number of bytes written=1072698
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=3340
                HDFS: Number of bytes written=2823
                HDFS: Number of read operations=15
                HDFS: Number of large read operations=0
```

On Vérifie la préséance de fichier de résultats



```
hduser@alakouche-VB:~/Documents/code$ hadoop fs -ls /results
2020-11-18 21:39:31,300 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2020-11-18 21:37 /results/_SUCCESS
-rw-r--r--   1 hduser supergroup       2823 2020-11-18 21:37 /results/part-r-00000
hduser@alakouche-VB:~/Documents/code$ hadoop fs -cat /results/part-r-00000
```

- Affichage de résultats



```
hduser@alakouche-VB:~/Documents/code$ hadoop fs -cat /results/part-r-00000
2020-11-18 21:40:22,517 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
2020-11-18 21:40:26,618 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
```

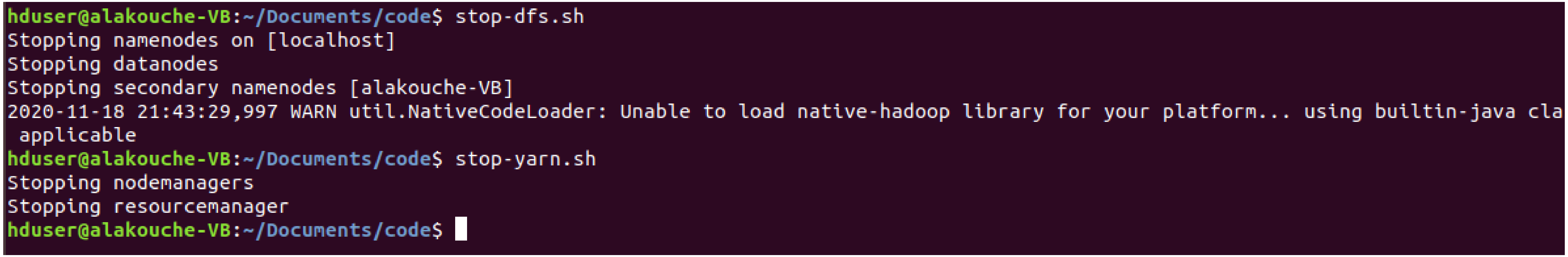
```
a       6 occurences.
adoraient       1 occurences.
ailes   1 occurences.
aima    1 occurences.
amour   1 occurences.
au      11 occurences.
bas     1 occurences.
belle   1 occurences.
bles    1 occurences.
bras    1 occurences.
bretagne        1 occurences.
brula   1 occurences.
celle   1 occurences.
celui   20 occurences.
cette   1 occurences.
chancelle       1 occurences.
chapelle        1 occurences.
ciel    10 occurences.
citadelle       1 occurences.
clarte  1 occurences.
coeur   2 occurences.
combat  1 occurences.
comment 1 occurences.
commun  1 occurences.
coule   2 occurences.
couleur 1 occurences.
court   1 occurences.
```

```
meme    2 occurences.
mirabelle       1 occurences.
montait 1 occurences.
mourra  1 occurences.
murisse 1 occurences.
muscat  1 occurences.
ne      1 occurences.
nom     1 occurences.
nouvelle        1 occurences.
ny      10 occurences.
ou      3 occurences.
par     1 occurences.
pas     11 occurences.
passent 1 occurences.
plus    2 occurences.
pour    1 occurences.
prefere 1 occurences.
prison  1 occurences.
prisonniere     1 occurences.
qua     1 occurences.
quand   2 occurences.
quaucun 1 occurences.
que     2 occurences.
quelle  1 occurences.
querelles       1 occurences.
qui     25 occurences.
quil    1 occurences.
```

```
nom     1 occurences.
nouvelle        1 occurences.
ny      10 occurences.
ou      3 occurences.
par     1 occurences.
pas     11 occurences.
passent 1 occurences.
plus    2 occurences.
pour    1 occurences.
prefere 1 occurences.
prison  1 occurences.
prisonniere     1 occurences.
qua     1 occurences.
quand   2 occurences.
quaucun 1 occurences.
que     2 occurences.
quelle  1 occurences.
querelles       1 occurences.
qui     25 occurences.
quil    1 occurences.
quimporte       1 occurences.
raisin  1 occurences.
rats    1 occurences.
rebelle 2 occurences.
rechantera      1 occurences.
repetant        1 occurences.
reseda  1 occurences.
```

....

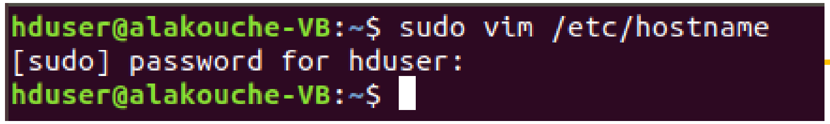
- On arrête tous les daemons en Cours d'exécution sur notre machine virtuelle.



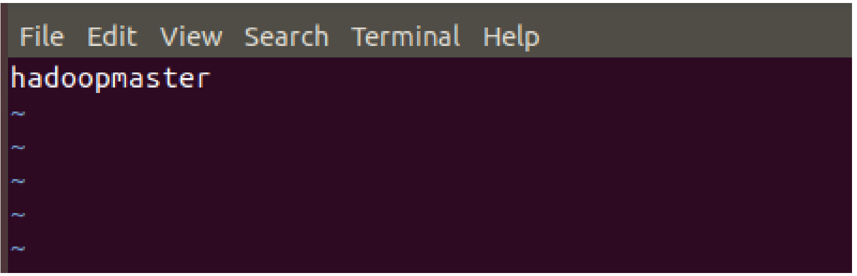
```
hduser@alakouche-VB:~/Documents/code$ stop-dfs.sh
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [alakouche-VB]
2020-11-18 21:43:29,997 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cla
 applicable
hduser@alakouche-VB:~/Documents/code$ stop-yarn.sh
Stopping nodemanagers
Stopping resourcemanager
hduser@alakouche-VB:~/Documents/code$
```

III Installation et configuration d’un cluster multi-nœude

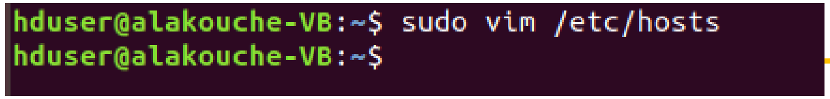
Nous allons travaillé avec la machine virtuelle précédemment configuré en node unique dans la section précédente. Et pour cela plusieurs fichiers doivent être modifiés



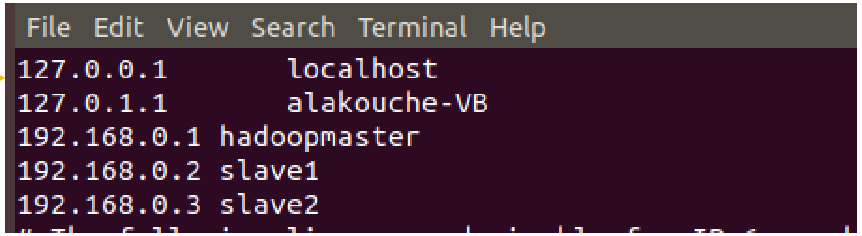
```
hduser@alakouche-VB:~$ sudo vim /etc/hostname
[sudo] password for hduser:
hduser@alakouche-VB:~$
```



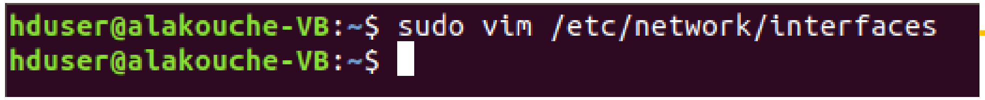
```
hadoopmaster
~
~
~
~
~
```



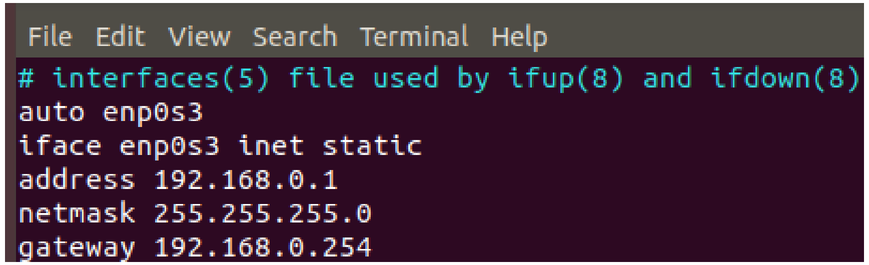
```
hduser@alakouche-VB:~$ sudo vim /etc/hosts
hduser@alakouche-VB:~$
```



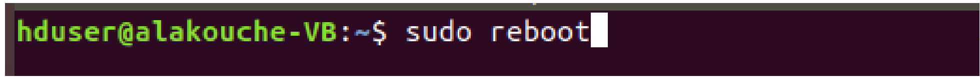
```
127.0.0.1       localhost
127.0.1.1       alakouche-VB
192.168.0.1 hadoopmaster
192.168.0.2 slave1
192.168.0.3 slave2
```



```
hduser@alakouche-VB:~$ sudo vim /etc/network/interfaces
hduser@alakouche-VB:~$
```



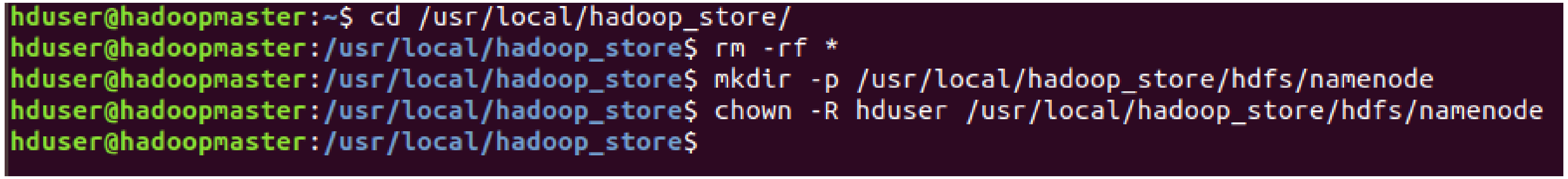
```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto enp0s3
iface enp0s3 inet static
address 192.168.0.1
netmask 255.255.255.0
gateway 192.168.0.254
```



```
hduser@alakouche-VB:~$ sudo reboot
```

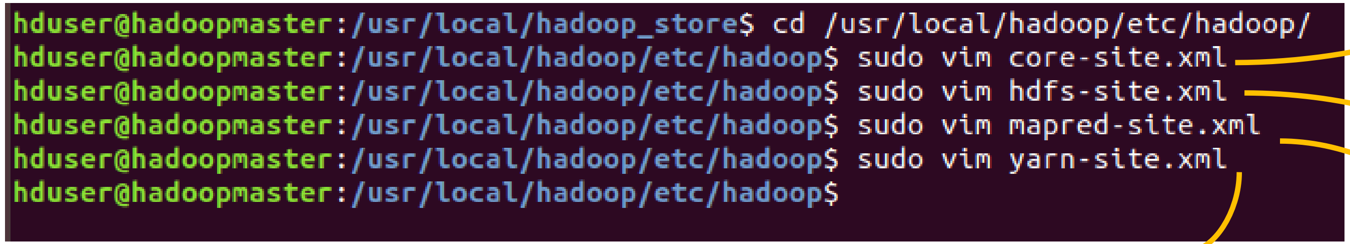
On Redémarre la machine pour prendre en compte les configurations

- Suppression des fichiers du répertoire de stockage de données créer par l'installation single node de Hadoop

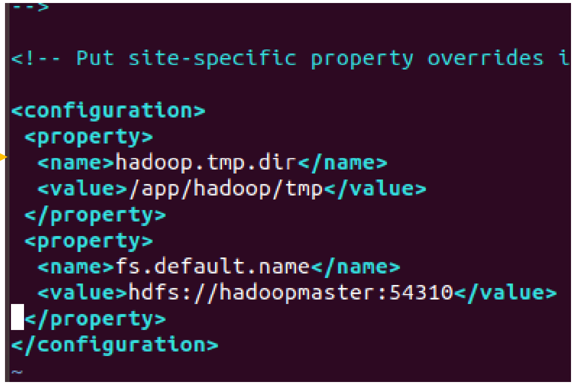


```
hduser@hadoopmaster:~$ cd /usr/local/hadoop_store/
hduser@hadoopmaster:/usr/local/hadoop_store$ rm -rf *
hduser@hadoopmaster:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/namenode
hduser@hadoopmaster:/usr/local/hadoop_store$ chown -R hduser /usr/local/hadoop_store/hdfs/namenode
hduser@hadoopmaster:/usr/local/hadoop_store$
```

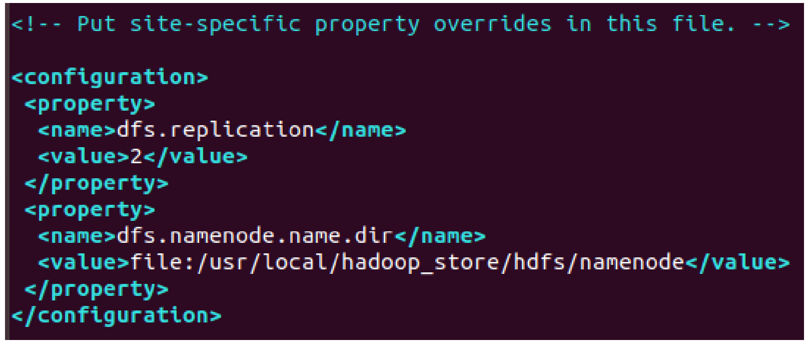
- Modification des fichiers de configuration de hadoop



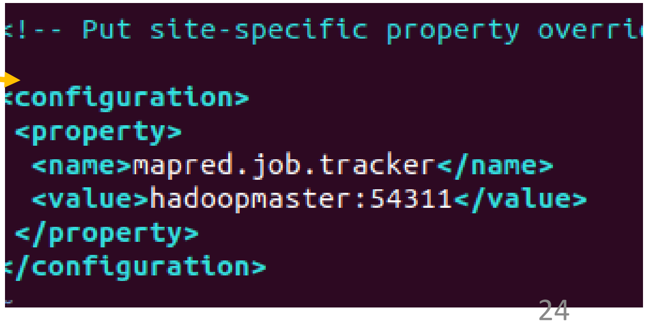
```
hduser@hadoopmaster:/usr/local/hadoop_store$ cd /usr/local/hadoop/etc/hadoop/
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ sudo vim core-site.xml
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ sudo vim hdfs-site.xml
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ sudo vim mapred-site.xml
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ sudo vim yarn-site.xml
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```



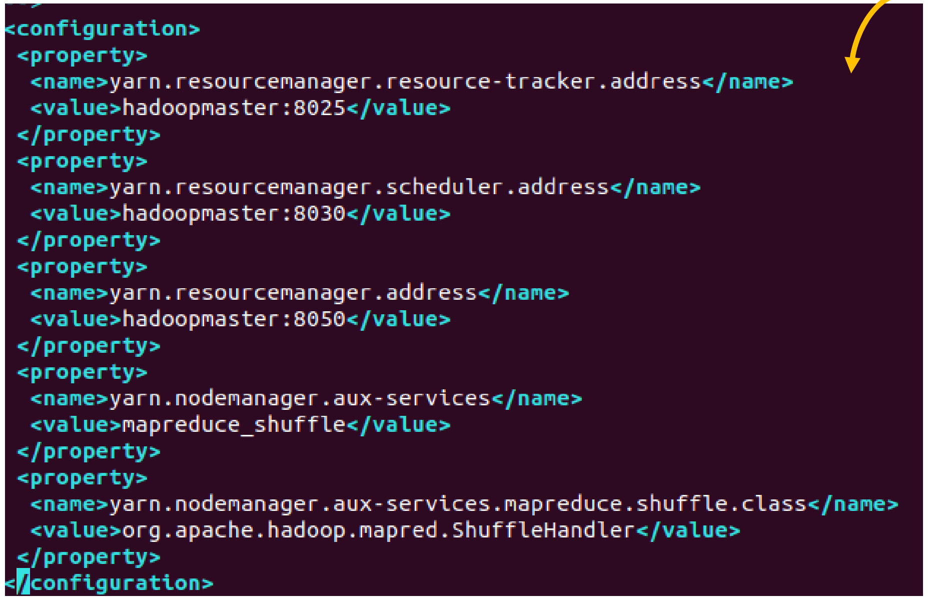
```
-->

<!-- Put site-specific property overrides i

<configuration>
 <property>
  <name>hadoop.tmp.dir</name>
  <value>/app/hadoop/tmp</value>
 </property>
 <property>
  <name>fs.default.name</name>
  <value>hdfs://hadoopmaster:54310</value>
 </property>
</configuration>
```



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

<configuration>
 <property>
  <name>dfs.replication</name>
  <value>2</value>
 </property>
 <property>
  <name>dfs.namenode.name.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
 </property>
</configuration>
```

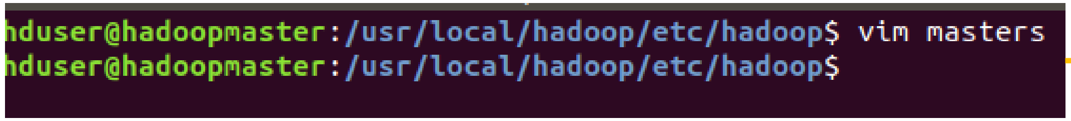


```
<!-- Put site-specific property overri

<configuration>
 <property>
  <name>mapred.job.tracker</name>
  <value>hadoopmaster:54311</value>
 </property>
</configuration>
```

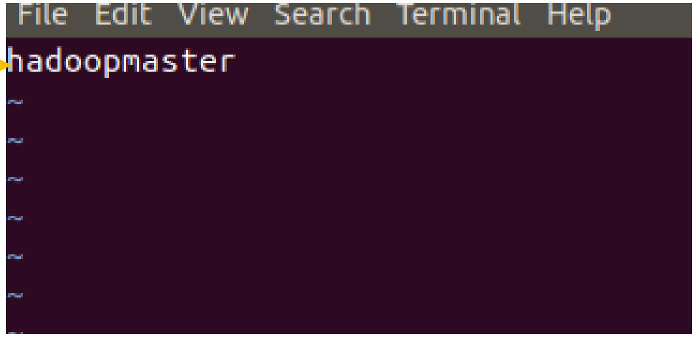


```
<configuration>
 <property>
  <name>yarn.resourcemanager.resource-tracker.address</name>
  <value>hadoopmaster:8025</value>
 </property>
 <property>
  <name>yarn.resourcemanager.scheduler.address</name>
  <value>hadoopmaster:8030</value>
 </property>
 <property>
  <name>yarn.resourcemanager.address</name>
  <value>hadoopmaster:8050</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>
</configuration>
```

- On crée le fichier masters qui contient le hostname de la machine master

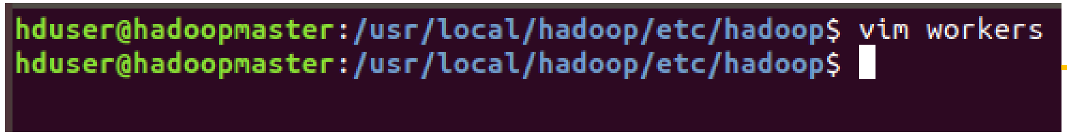


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ vim masters
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```

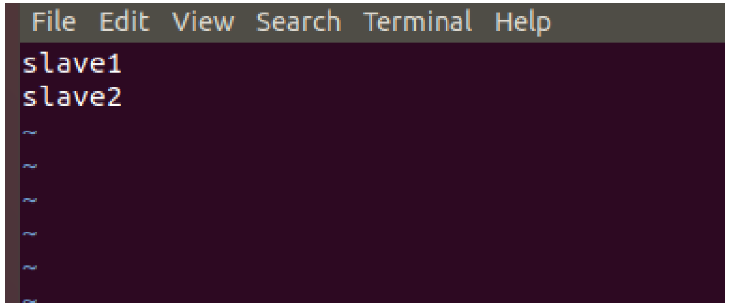


```
hadoopmaster
~
~
~
~
~
~
```

- Modification de fichier workers qui contient le hostname de chaque machine slave dans le répertoire

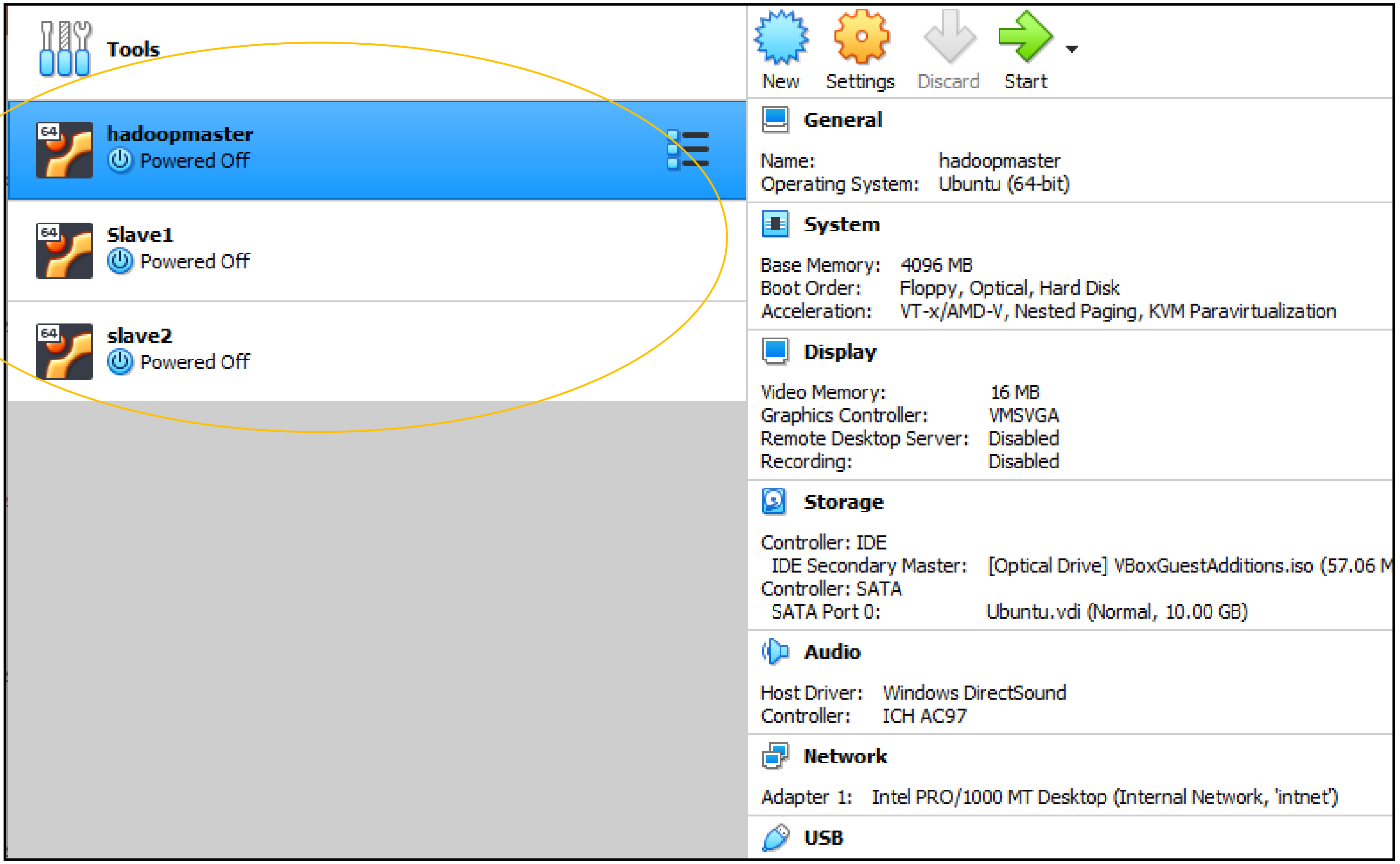


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ vim workers
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```



```
slave1
slave2
~
~
~
~
~
```

Clonage de la machine hadoopmaster



hadoopmaster
Powered Off

Slave1
Powered Off

slave2
Powered Off

General

Name: hadoopmaster
Operating System: Ubuntu (64-bit)

System

Base Memory: 4096 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Secondary Master: [Optical Drive] VBoxGuestAdditions.iso (57.06 M
Controller: SATA
SATA Port 0: Ubuntu.vdi (Normal, 10.00 GB)

Audio

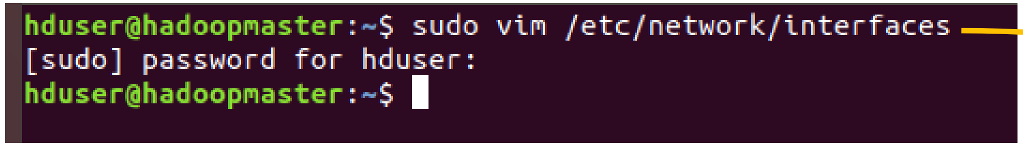
Host Driver: Windows DirectSound
Controller: ICH AC97

Network

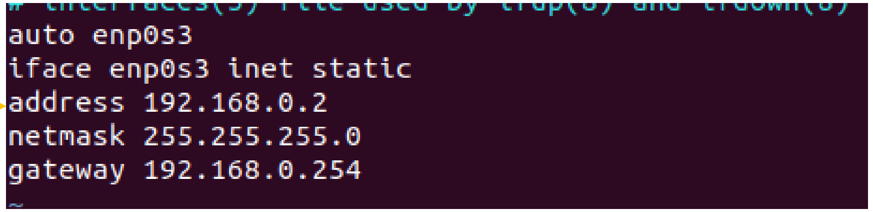
Adapter 1: Intel PRO/1000 MT Desktop (Internal Network, 'intnet')

USB

- Configuration de l'adresse IP fixe de la machine **slave1**

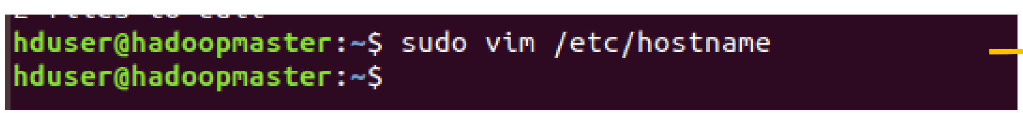


```
hduser@hadoopmaster:~$ sudo vim /etc/network/interfaces
[sudo] password for hduser:
hduser@hadoopmaster:~$ 
```

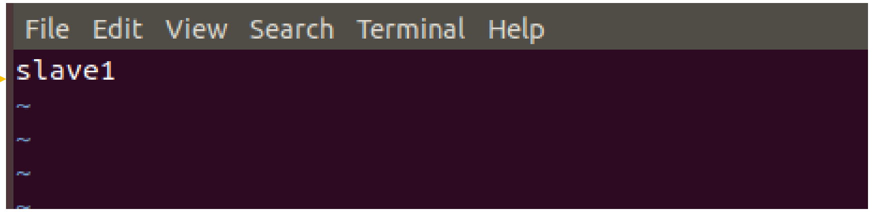


```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto enp0s3
iface enp0s3 inet static
address 192.168.0.2
netmask 255.255.255.0
gateway 192.168.0.254
~
```

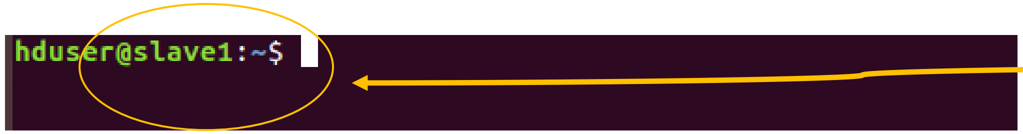
- Modification de fichier /etc/hostname de la machine **slave1**



```
2 files to edit
hduser@hadoopmaster:~$ sudo vim /etc/hostname
hduser@hadoopmaster:~$
```



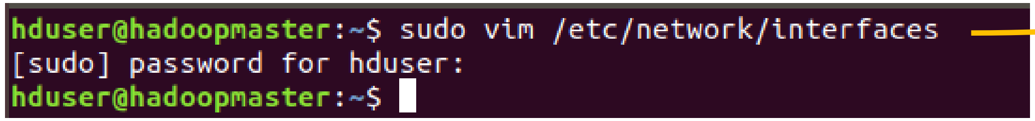
```
slave1
~
~
~
~
```



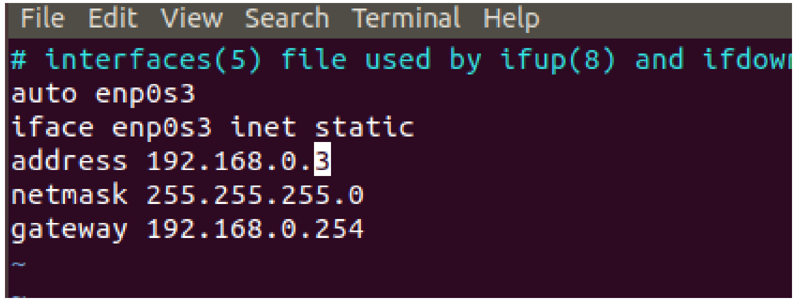
```
hduser@slave1:~$ 
```

On Redémarre la machine pour prendre en compte les configurations.

- Configuration de l'adresse IP fixe de la machine **slave2**

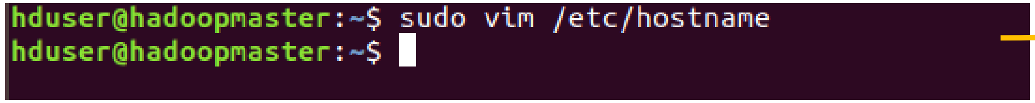


```
hduser@hadoopmaster:~$ sudo vim /etc/network/interfaces
[sudo] password for hduser:
hduser@hadoopmaster:~$ 
```

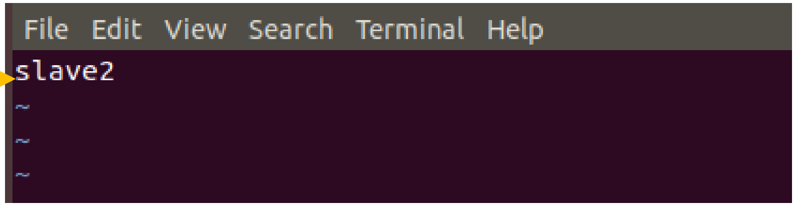


```
# interfaces(5) file used by ifup(8) and ifdow
auto enp0s3
iface enp0s3 inet static
address 192.168.0.3
netmask 255.255.255.0
gateway 192.168.0.254
~
```

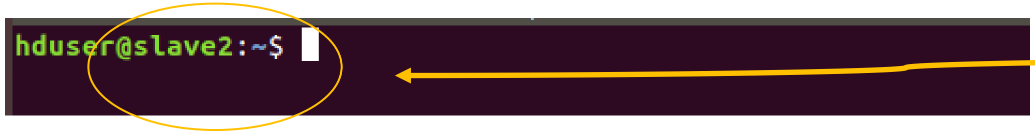
- Modification de fichier /etc/hostname de la machine **slave2**



```
hduser@hadoopmaster:~$ sudo vim /etc/hostname
hduser@hadoopmaster:~$ 
```



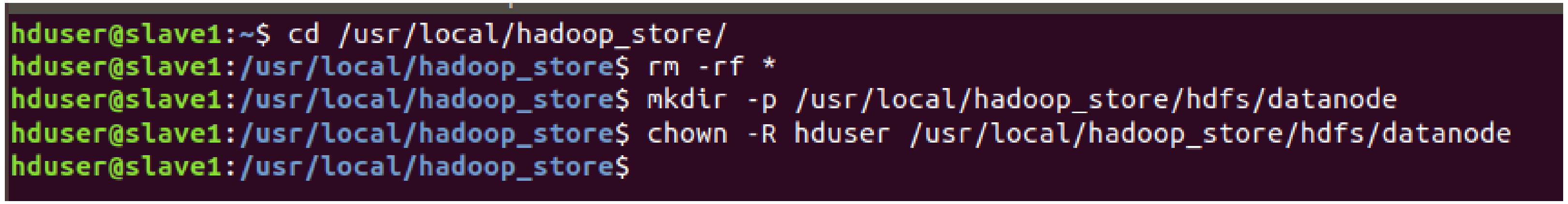
```
slave2
~
~
~
```



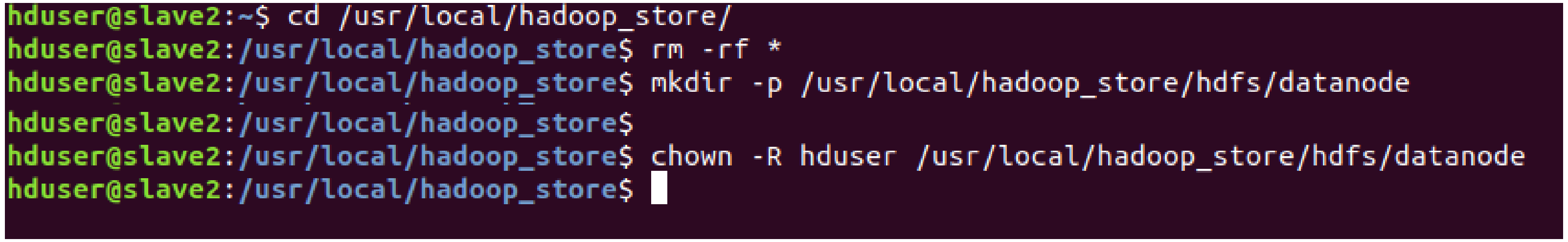
```
hduser@slave2:~$ 
```

On Redémarre la machine pour prendre en compte les configurations.

- On Supprime les fichiers du répertoire de stockage de données créer par l'installation single node de Hadoop pour slave1 et slave2:

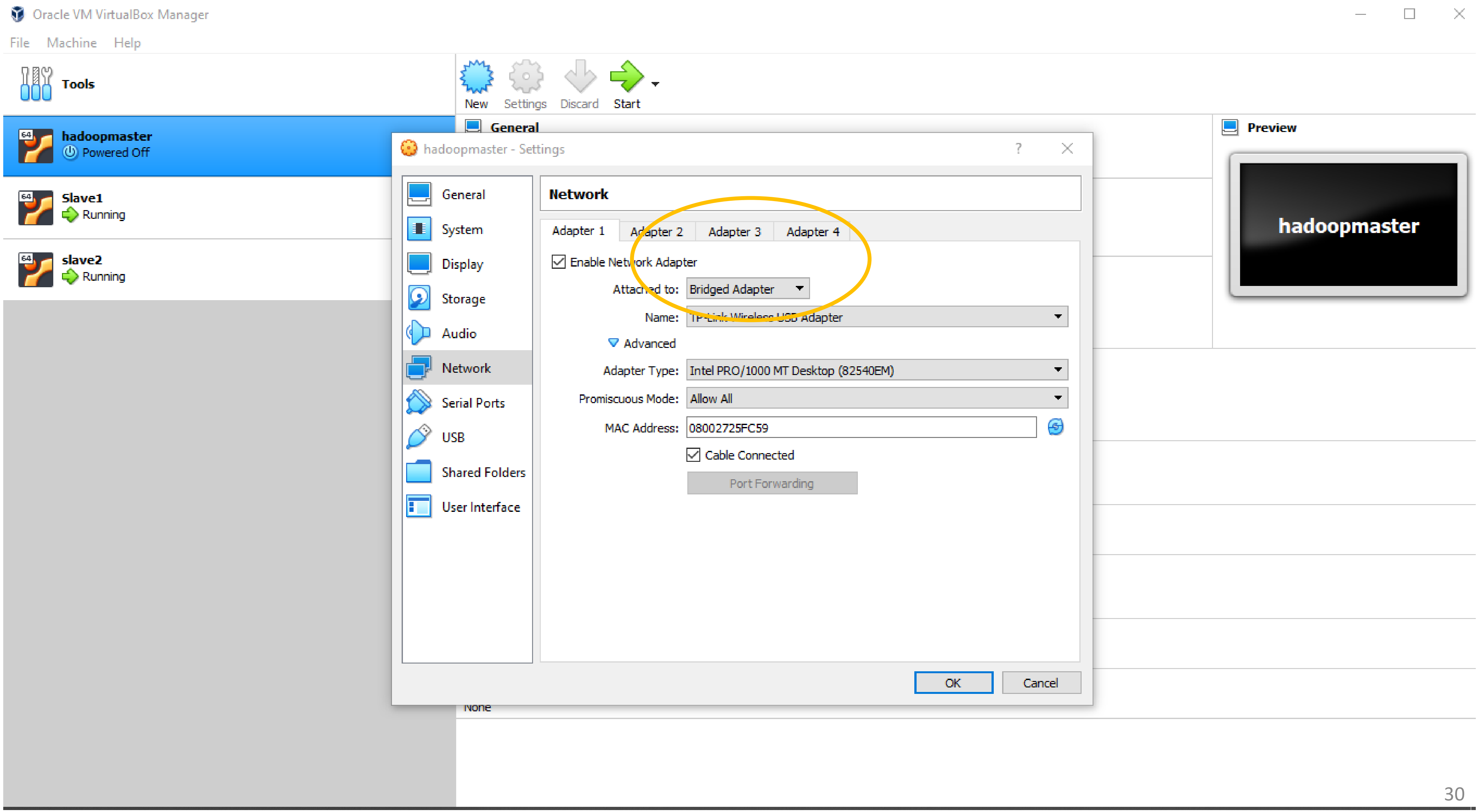


```
hduser@slave1:~$ cd /usr/local/hadoop_store/
hduser@slave1:/usr/local/hadoop_store$ rm -rf *
hduser@slave1:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/datanode
hduser@slave1:/usr/local/hadoop_store$ chown -R hduser /usr/local/hadoop_store/hdfs/datanode
hduser@slave1:/usr/local/hadoop_store$
```



```
hduser@slave2:~$ cd /usr/local/hadoop_store/
hduser@slave2:/usr/local/hadoop_store$ rm -rf *
hduser@slave2:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/datanode
hduser@slave2:/usr/local/hadoop_store$
hduser@slave2:/usr/local/hadoop_store$ chown -R hduser /usr/local/hadoop_store/hdfs/datanode
hduser@slave2:/usr/local/hadoop_store$ 
```

- Connexion entre les machines du cluster



Network

☑ Enable Network Adapter

Attached to: Bridged Adapter

Name: TP-Link Wireless USB Adapter

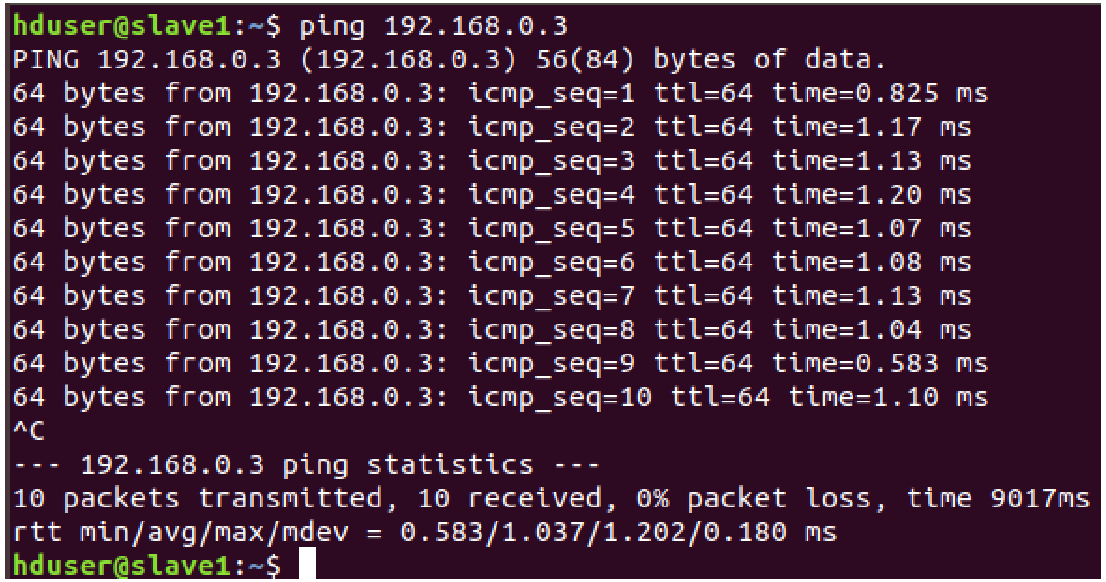
Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Allow All

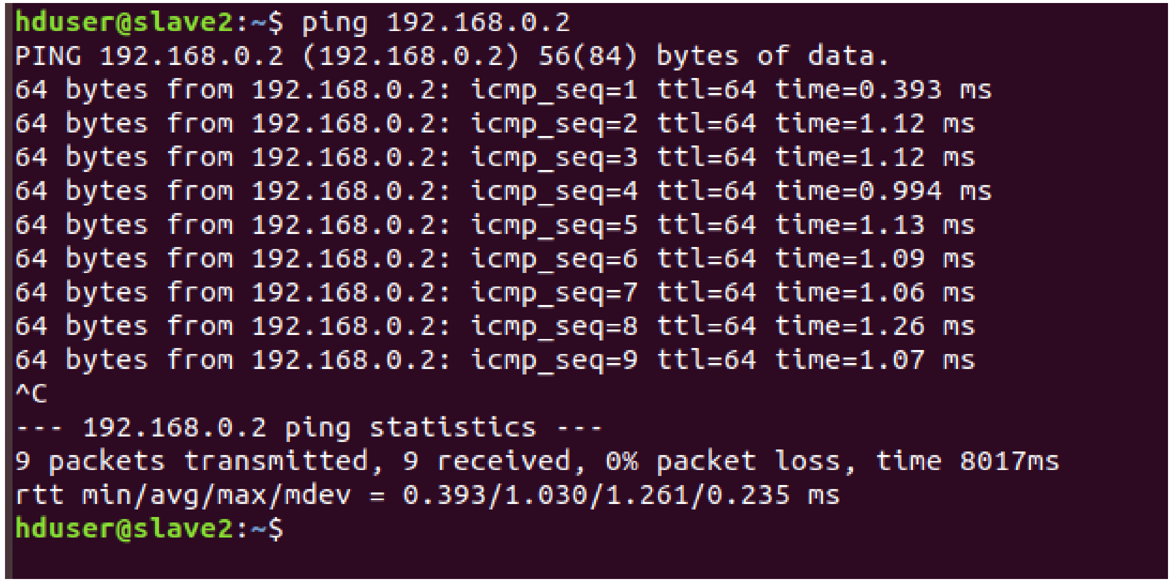
MAC Address: 08002725FC59

☑ Cable Connected

Test de connexion entre slave1 et slave2

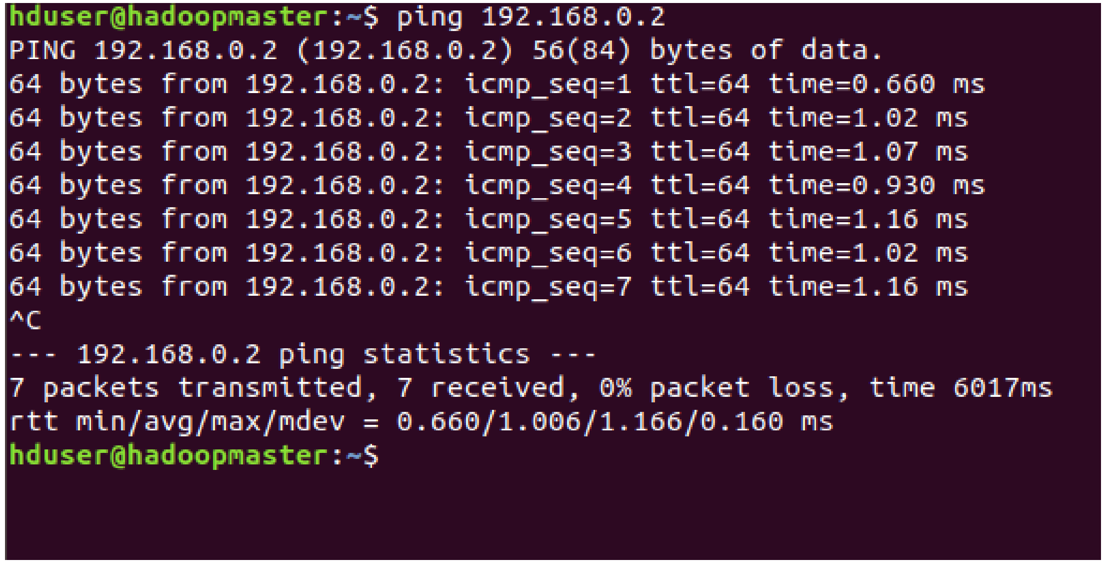


```
hduser@slave1:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.825 ms
64 bytes from 192.168.0.3: icmp_seq=2 ttl=64 time=1.17 ms
64 bytes from 192.168.0.3: icmp_seq=3 ttl=64 time=1.13 ms
64 bytes from 192.168.0.3: icmp_seq=4 ttl=64 time=1.20 ms
64 bytes from 192.168.0.3: icmp_seq=5 ttl=64 time=1.07 ms
64 bytes from 192.168.0.3: icmp_seq=6 ttl=64 time=1.08 ms
64 bytes from 192.168.0.3: icmp_seq=7 ttl=64 time=1.13 ms
64 bytes from 192.168.0.3: icmp_seq=8 ttl=64 time=1.04 ms
64 bytes from 192.168.0.3: icmp_seq=9 ttl=64 time=0.583 ms
64 bytes from 192.168.0.3: icmp_seq=10 ttl=64 time=1.10 ms
^C
--- 192.168.0.3 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9017ms
rtt min/avg/max/mdev = 0.583/1.037/1.202/0.180 ms
hduser@slave1:~$
```

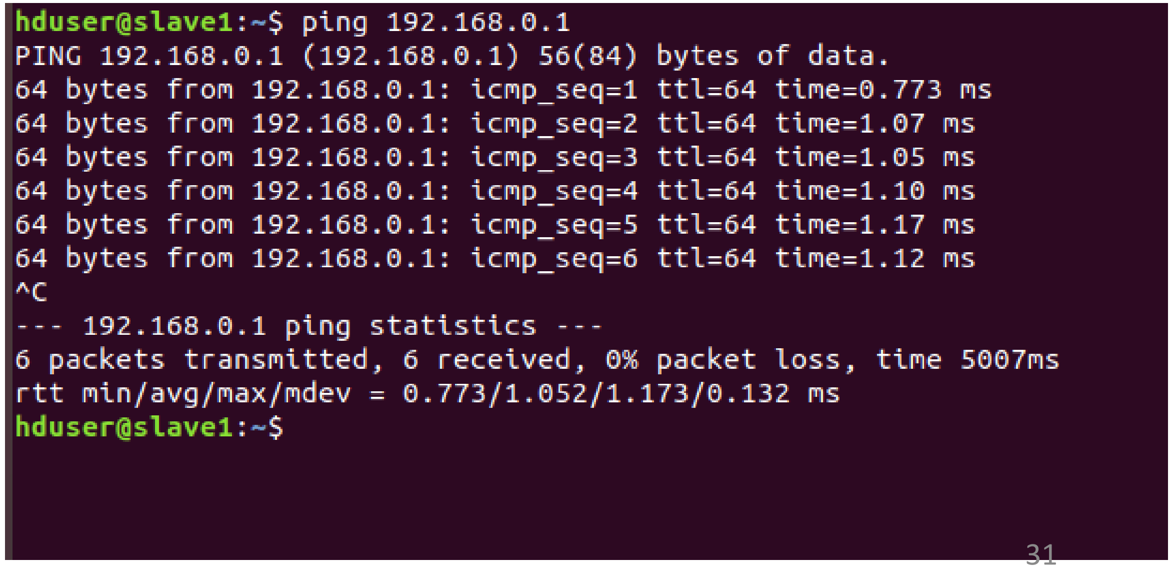


```
hduser@slave2:~$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.393 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=1.12 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=64 time=1.12 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=64 time=0.994 ms
64 bytes from 192.168.0.2: icmp_seq=5 ttl=64 time=1.13 ms
64 bytes from 192.168.0.2: icmp_seq=6 ttl=64 time=1.09 ms
64 bytes from 192.168.0.2: icmp_seq=7 ttl=64 time=1.06 ms
64 bytes from 192.168.0.2: icmp_seq=8 ttl=64 time=1.26 ms
64 bytes from 192.168.0.2: icmp_seq=9 ttl=64 time=1.07 ms
^C
--- 192.168.0.2 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8017ms
rtt min/avg/max/mdev = 0.393/1.030/1.261/0.235 ms
hduser@slave2:~$
```

Test de connexion entre hadoopmaster et slave1

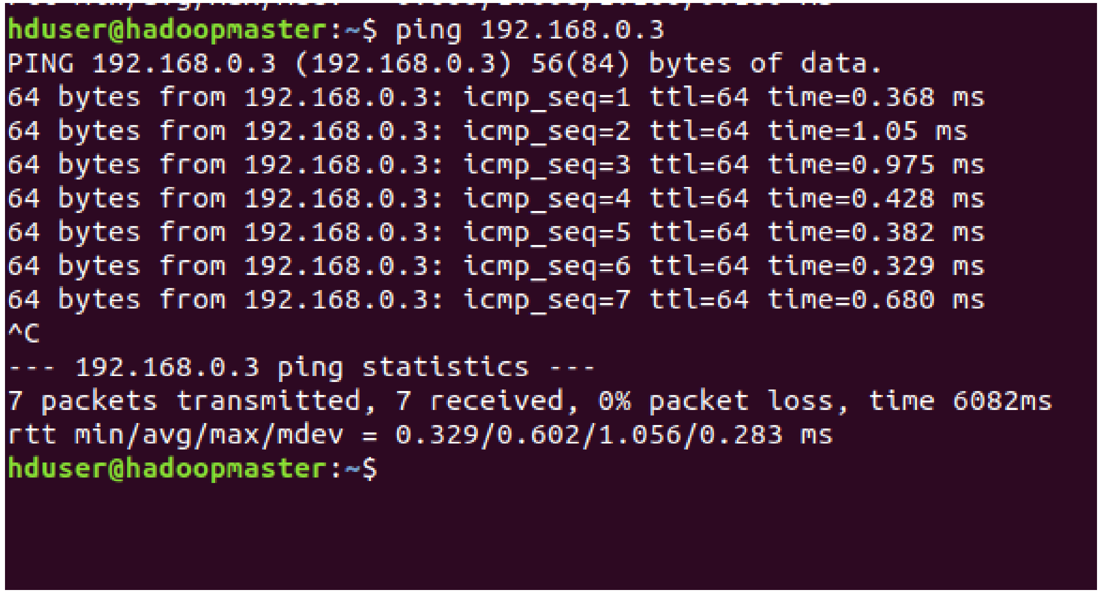


```
hduser@hadoopmaster:~$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.660 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=1.02 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=64 time=1.07 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=64 time=0.930 ms
64 bytes from 192.168.0.2: icmp_seq=5 ttl=64 time=1.16 ms
64 bytes from 192.168.0.2: icmp_seq=6 ttl=64 time=1.02 ms
64 bytes from 192.168.0.2: icmp_seq=7 ttl=64 time=1.16 ms
^C
--- 192.168.0.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6017ms
rtt min/avg/max/mdev = 0.660/1.006/1.166/0.160 ms
hduser@hadoopmaster:~$
```

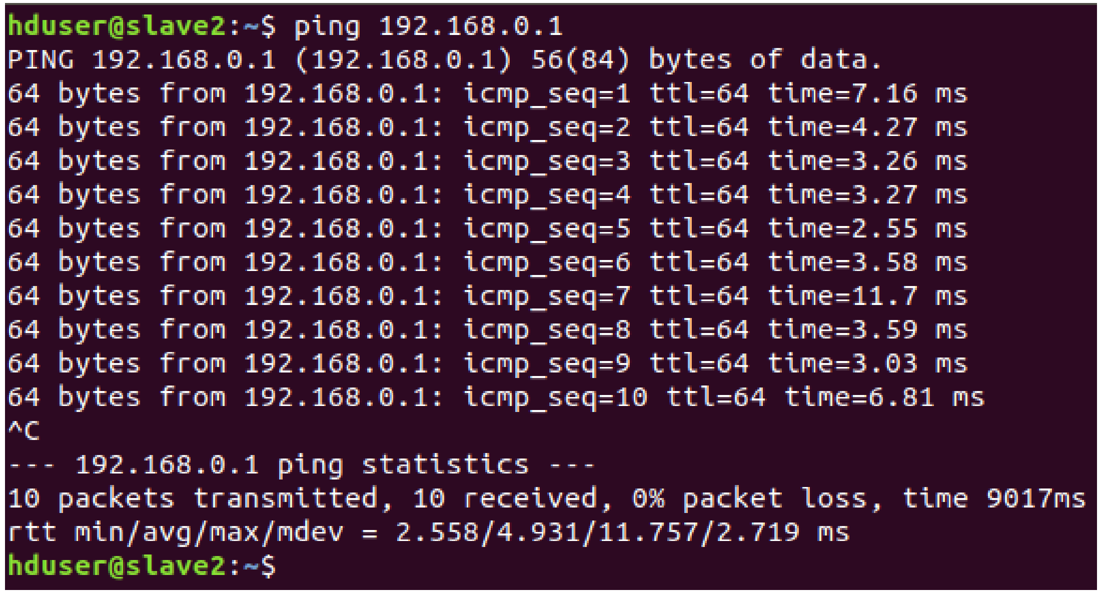


```
hduser@slave1:~$ ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.773 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=1.07 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=64 time=1.05 ms
64 bytes from 192.168.0.1: icmp_seq=4 ttl=64 time=1.10 ms
64 bytes from 192.168.0.1: icmp_seq=5 ttl=64 time=1.17 ms
64 bytes from 192.168.0.1: icmp_seq=6 ttl=64 time=1.12 ms
^C
--- 192.168.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5007ms
rtt min/avg/max/mdev = 0.773/1.052/1.173/0.132 ms
hduser@slave1:~$
```

Test de connexion entre hadoopmaster et slave2

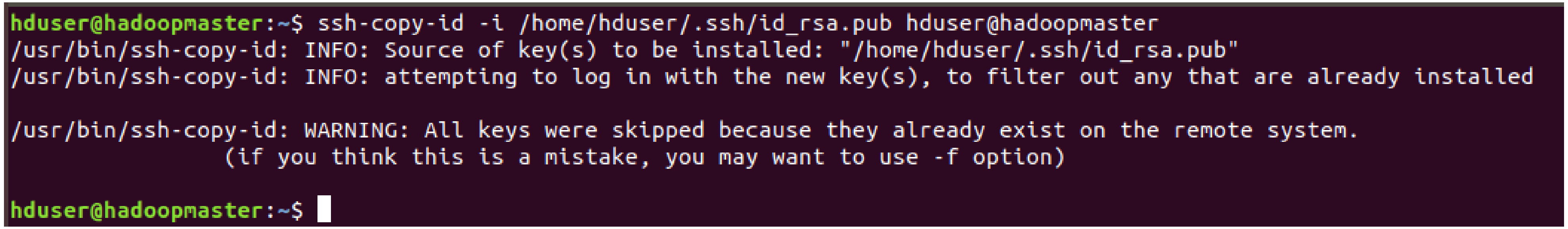


```
hduser@hadoopmaster:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.368 ms
64 bytes from 192.168.0.3: icmp_seq=2 ttl=64 time=1.05 ms
64 bytes from 192.168.0.3: icmp_seq=3 ttl=64 time=0.975 ms
64 bytes from 192.168.0.3: icmp_seq=4 ttl=64 time=0.428 ms
64 bytes from 192.168.0.3: icmp_seq=5 ttl=64 time=0.382 ms
64 bytes from 192.168.0.3: icmp_seq=6 ttl=64 time=0.329 ms
64 bytes from 192.168.0.3: icmp_seq=7 ttl=64 time=0.680 ms
^C
--- 192.168.0.3 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6082ms
rtt min/avg/max/mdev = 0.329/0.602/1.056/0.283 ms
hduser@hadoopmaster:~$
```

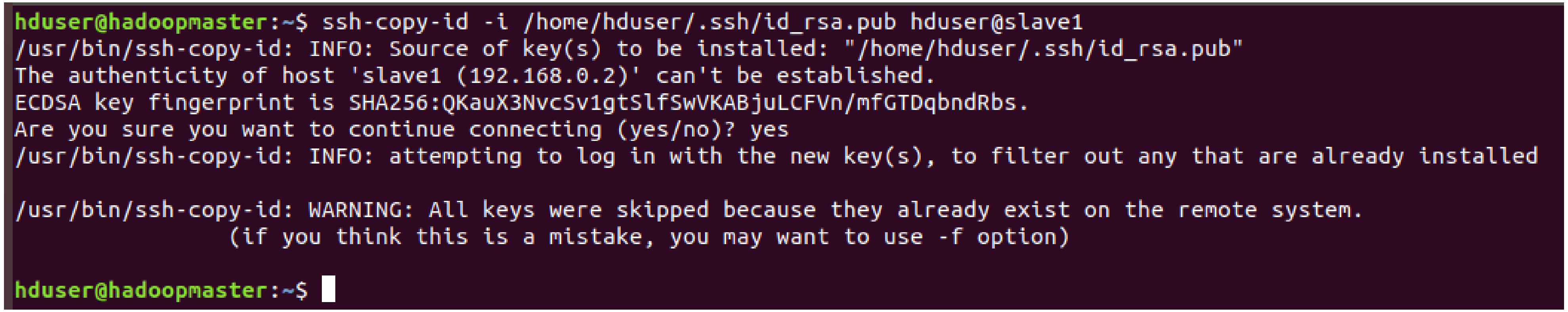


```
hduser@slave2:~$ ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=7.16 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=4.27 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=64 time=3.26 ms
64 bytes from 192.168.0.1: icmp_seq=4 ttl=64 time=3.27 ms
64 bytes from 192.168.0.1: icmp_seq=5 ttl=64 time=2.55 ms
64 bytes from 192.168.0.1: icmp_seq=6 ttl=64 time=3.58 ms
64 bytes from 192.168.0.1: icmp_seq=7 ttl=64 time=11.7 ms
64 bytes from 192.168.0.1: icmp_seq=8 ttl=64 time=3.59 ms
64 bytes from 192.168.0.1: icmp_seq=9 ttl=64 time=3.03 ms
64 bytes from 192.168.0.1: icmp_seq=10 ttl=64 time=6.81 ms
^C
--- 192.168.0.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9017ms
rtt min/avg/max/mdev = 2.558/4.931/11.757/2.719 ms
hduser@slave2:~$
```

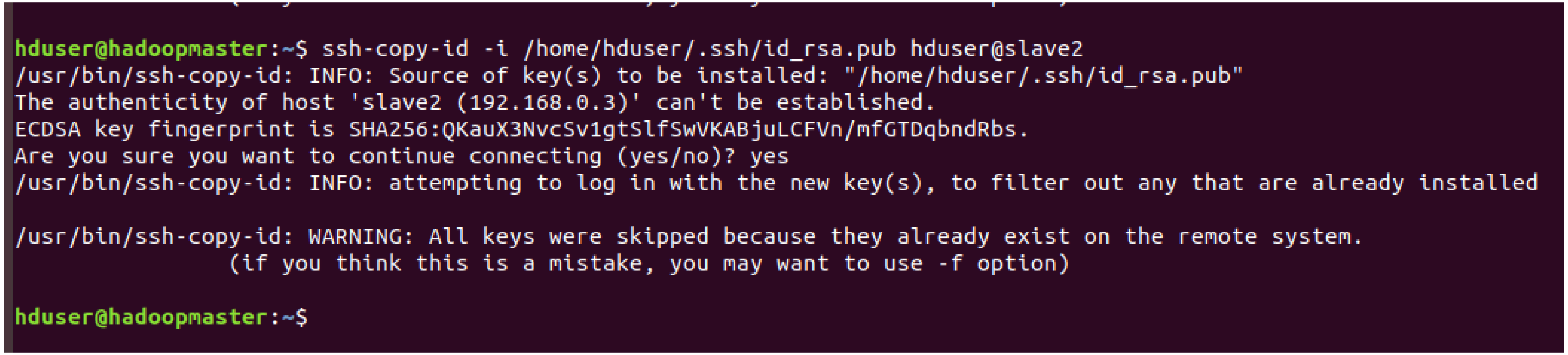
- On copie la clé ssh pour configurer un accès ssh sans mot de passe entre les machines du cluster.



```
hduser@hadoopmaster:~$ ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub hduser@hadoopmaster
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hduser/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
                (if you think this is a mistake, you may want to use -f option)

hduser@hadoopmaster:~$ 
```

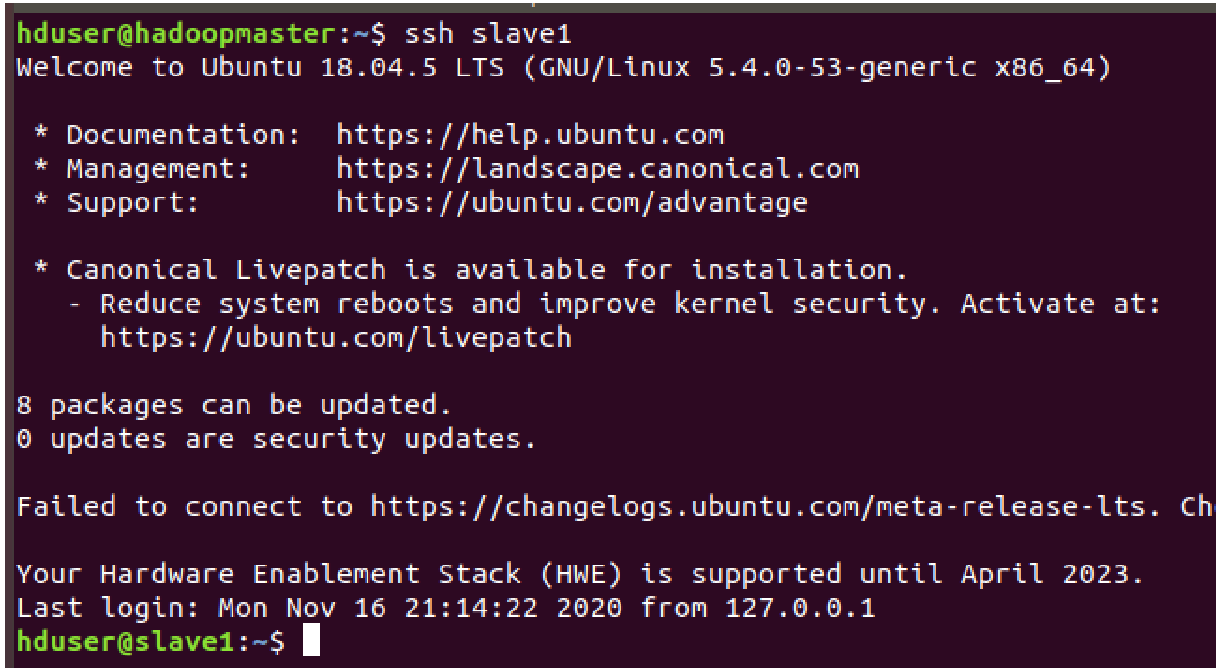


```
hduser@hadoopmaster:~$ ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub hduser@slave1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hduser/.ssh/id_rsa.pub"
The authenticity of host 'slave1 (192.168.0.2)' can't be established.
ECDSA key fingerprint is SHA256:QKauX3NvcSv1gtSlfSwVKABjuLCFVn/mfGTDqbndRbs.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
                (if you think this is a mistake, you may want to use -f option)

hduser@hadoopmaster:~$ 
```

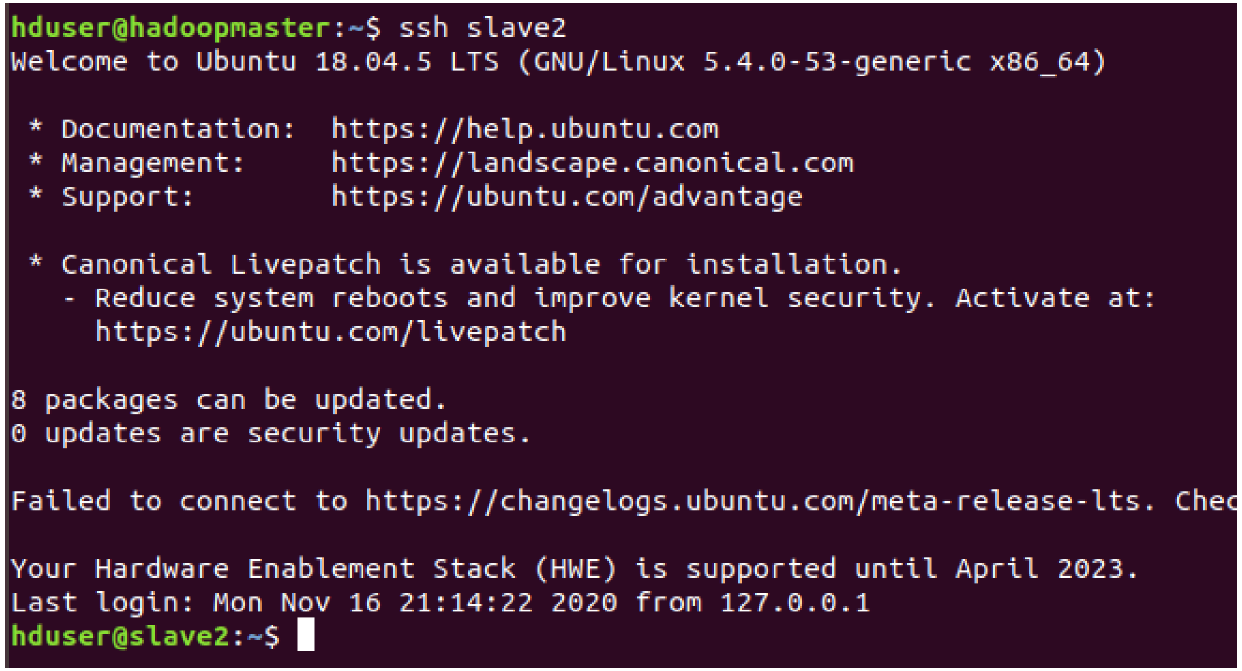


```
hduser@hadoopmaster:~$ ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub hduser@slave2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hduser/.ssh/id_rsa.pub"
The authenticity of host 'slave2 (192.168.0.3)' can't be established.
ECDSA key fingerprint is SHA256:QKauX3NvcSv1gtSlfSwVKABjuLCFVn/mfGTDqbndRbs.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
                (if you think this is a mistake, you may want to use -f option)

hduser@hadoopmaster:~$
```

- On Teste la connexion **ssh** entre les machines du cluster.

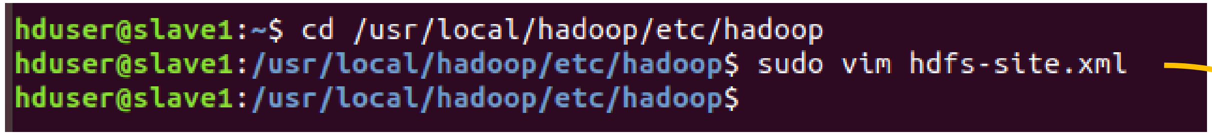


```
hduser@hadoopmaster:~$ ssh slave1
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-53-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

8 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Ch

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Mon Nov 16 21:14:22 2020 from 127.0.0.1
hduser@slave1:~$ 
```

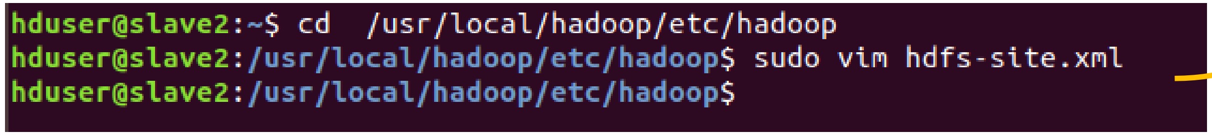


```
hduser@hadoopmaster:~$ ssh slave2
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-53-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

8 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Chec

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Mon Nov 16 21:14:22 2020 from 127.0.0.1
hduser@slave2:~$ 
```

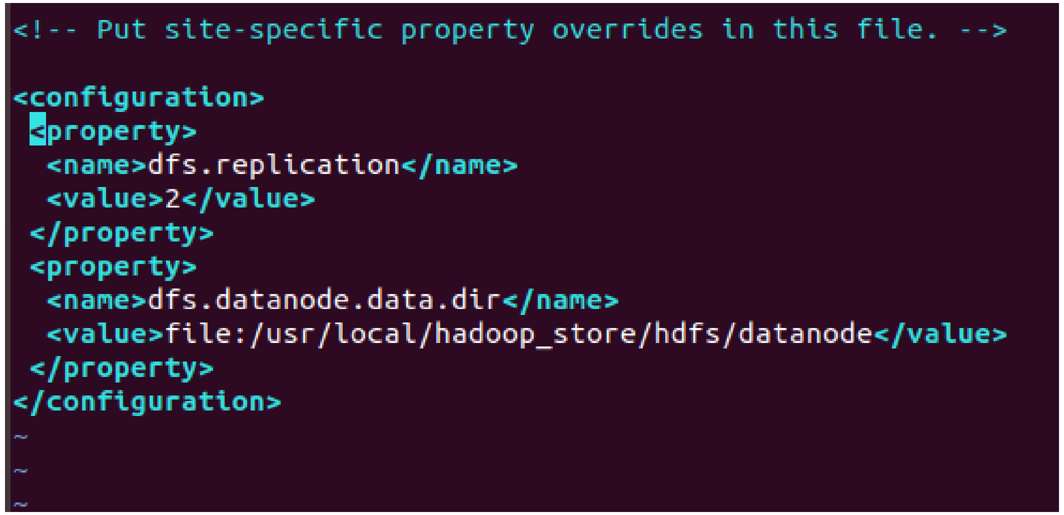
- Modification de fichier **hdfs-site.xml** de la machine virtuelle slave1 et slave2



```
hduser@slave1:~$ cd /usr/local/hadoop/etc/hadoop
hduser@slave1:/usr/local/hadoop/etc/hadoop$ sudo vim hdfs-site.xml
hduser@slave1:/usr/local/hadoop/etc/hadoop$
```

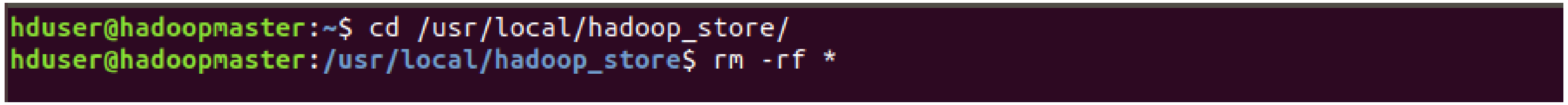


```
hduser@slave2:~$ cd  /usr/local/hadoop/etc/hadoop
hduser@slave2:/usr/local/hadoop/etc/hadoop$ sudo vim hdfs-site.xml
hduser@slave2:/usr/local/hadoop/etc/hadoop$
```

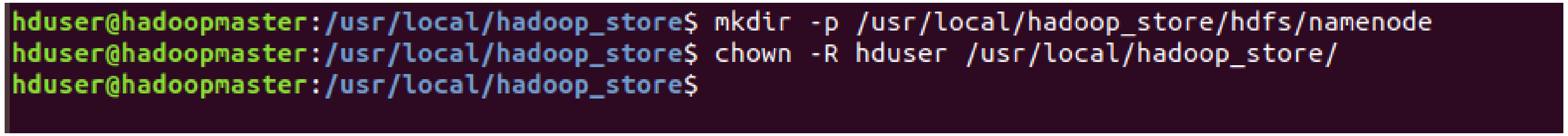


```
<!-- Put site-specific property overrides in this file. -->

<configuration>
 <property>
  <name>dfs.replication</name>
  <value>2</value>
 </property>
 <property>
  <name>dfs.datanode.data.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
 </property>
</configuration>
~
~
~
```

- Avant de démarrer le cluster, il faut vider aussi le répertoire de stockage du nœud hadoopmaster

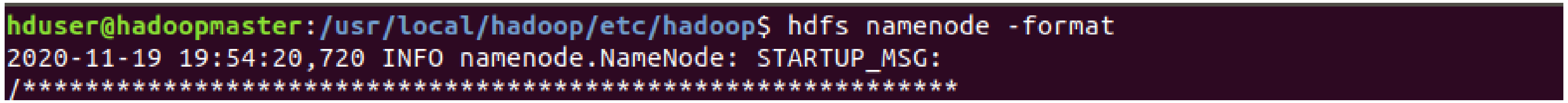


```
hduser@hadoopmaster:~$ cd /usr/local/hadoop_store/
hduser@hadoopmaster:/usr/local/hadoop_store$ rm -rf *
```

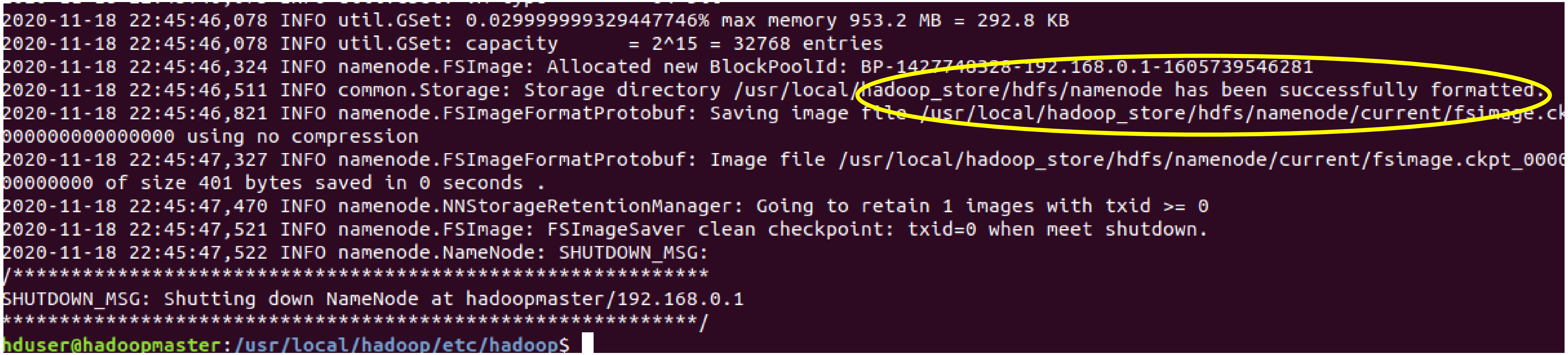


```
hduser@hadoopmaster:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/namenode
hduser@hadoopmaster:/usr/local/hadoop_store$ chown -R hduser /usr/local/hadoop_store/
hduser@hadoopmaster:/usr/local/hadoop_store$
```

- Avant de démarrer le serveur Hadoop, il faut formater le système de fichiers HDFS.

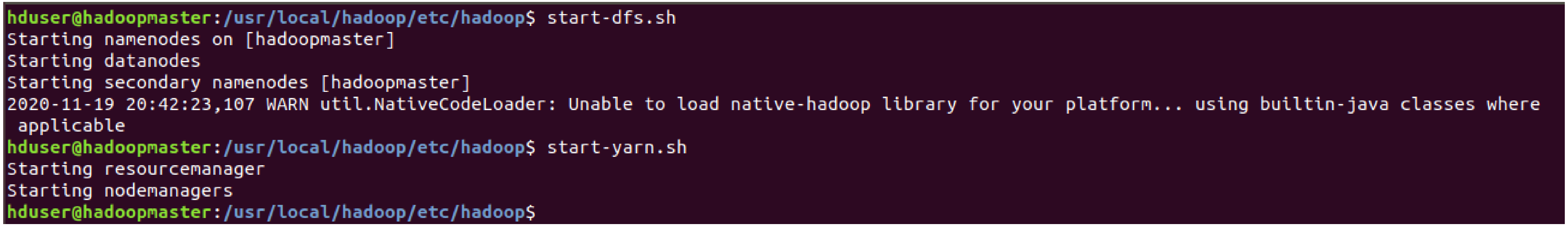


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ hdfs namenode -format
2020-11-19 19:54:20,720 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
```

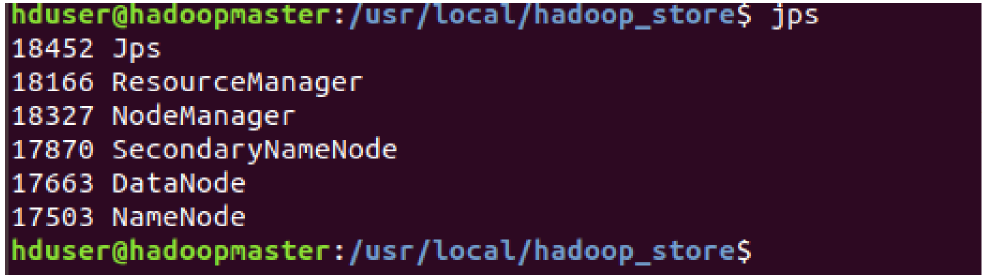


```
2020-11-18 22:45:46,078 INFO util.GSet: 0.029999999329447746% max memory 953.2 MB = 292.8 KB
2020-11-18 22:45:46,078 INFO util.GSet: capacity      = 2^15 = 32768 entries
2020-11-18 22:45:46,324 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1427748328-192.168.0.1-1605739546281
2020-11-18 22:45:46,511 INFO common.Storage: Storage directory /usr/local/hadoop_store/hdfs/namenode has been successfully formatted.
2020-11-18 22:45:46,821 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.ck
000000000000000 using no compression
2020-11-18 22:45:47,327 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.ckpt_0000
00000000 of size 401 bytes saved in 0 seconds .
2020-11-18 22:45:47,470 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2020-11-18 22:45:47,521 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2020-11-18 22:45:47,522 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at hadoopmaster/192.168.0.1
************************************************************/
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```

- Démarrage de Hadoop

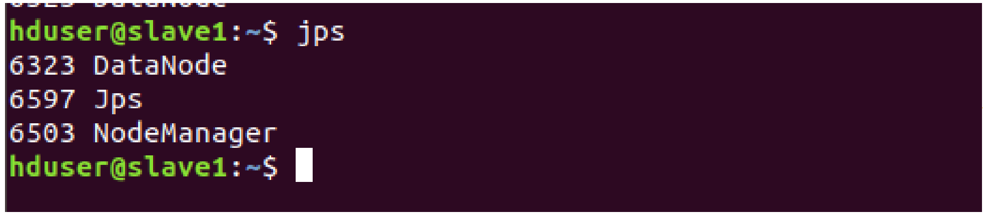


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ start-dfs.sh
Starting namenodes on [hadoopmaster]
Starting datanodes
Starting secondary namenodes [hadoopmaster]
2020-11-19 20:42:23,107 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```

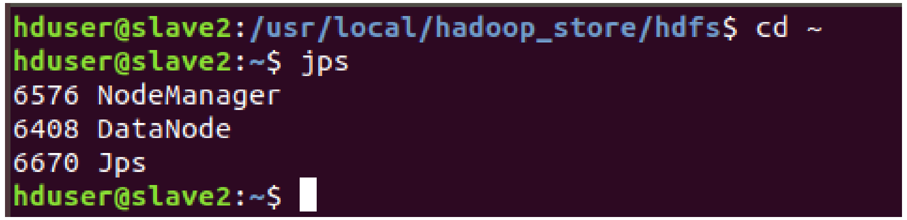


```
hduser@hadoopmaster:/usr/local/hadoop_store$ jps
18452 Jps
18166 ResourceManager
18327 NodeManager
17870 SecondaryNameNode
17663 DataNode
17503 NameNode
hduser@hadoopmaster:/usr/local/hadoop_store$
```

On lance la commande jps dans chaque cluster

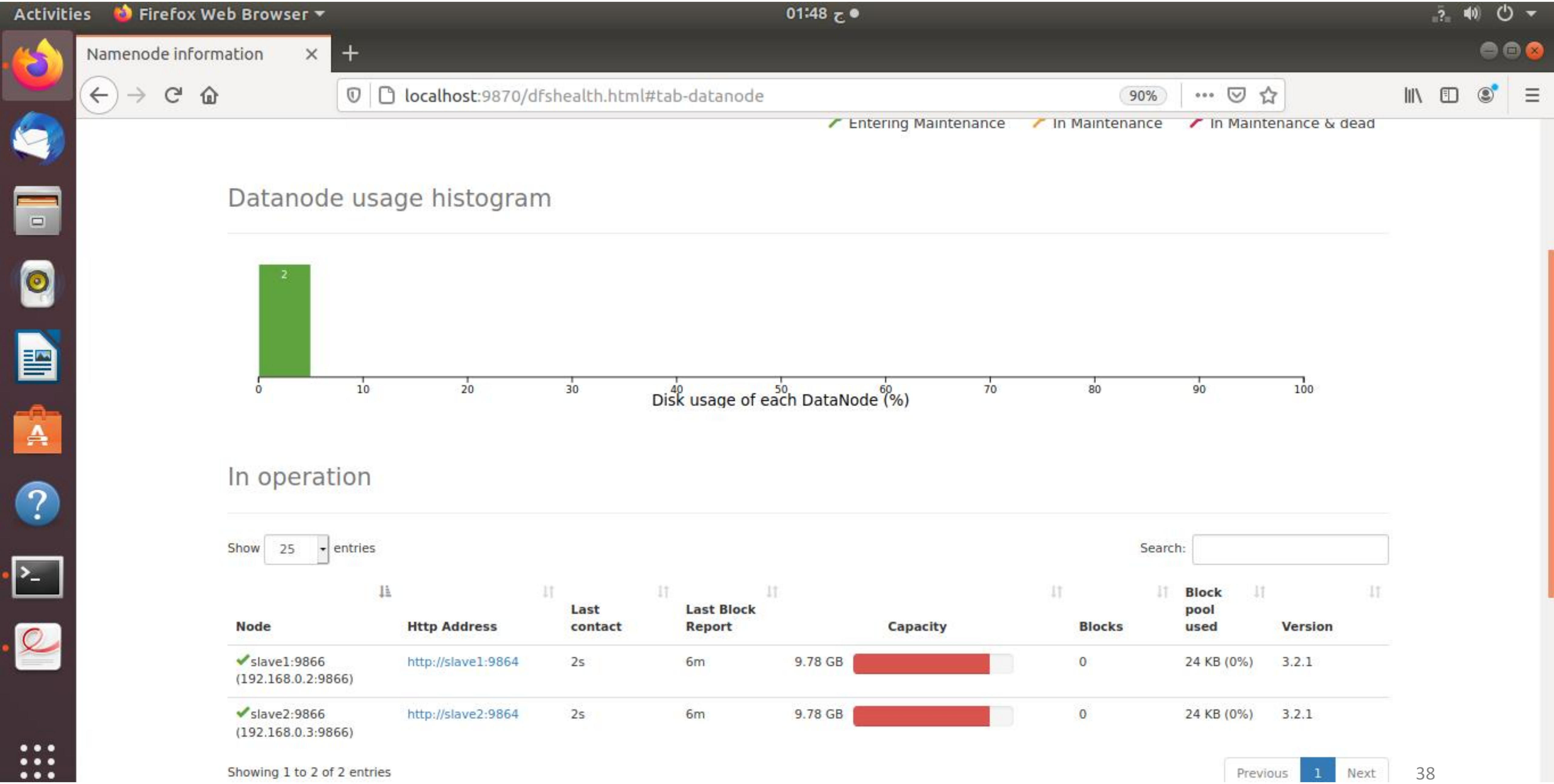


```
hduser@slave1:~$ jps
6323 DataNode
6597 Jps
6503 NodeManager
hduser@slave1:~$ 
```



```
hduser@slave2:/usr/local/hadoop_store/hdfs$ cd ~
hduser@slave2:~$ jps
6576 NodeManager
6408 DataNode
6670 Jps
hduser@slave2:~$ 
```

- Vérification de l’installation : On peut maintenant accéder à l’interface web Hadoop .



Datanode usage histogram

In operation

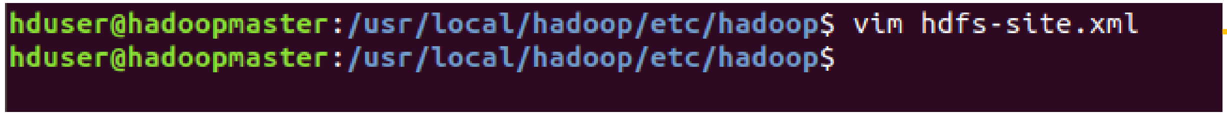
Show 25 entries Search:

Node	Http Address	Last contact	Last Block Report	Capacity		Blocks	Block pool used	Version
✓slave1:9866 (192.168.0.2:9866)	http://slave1:9864	2s	6m	9.78 GB	🟥	0	24 KB (0%)	3.2.1
✓slave2:9866 (192.168.0.3:9866)	http://slave2:9864	2s	6m	9.78 GB	🟥	0	24 KB (0%)	3.2.1

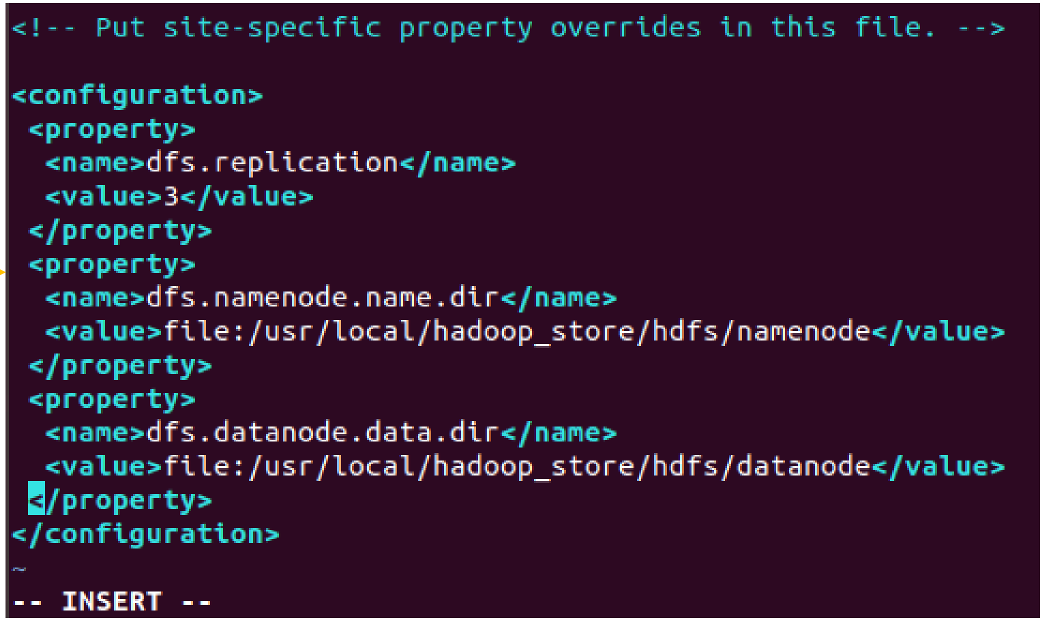
Showing 1 to 2 of 2 entries

Previous 1 Next

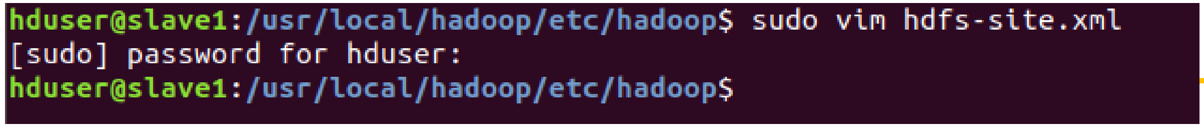
- On peut ajouter le noeud hadoopmaster comme datanode aussi, pour cela les fichiers suivants doivent être modifiés comme suit :



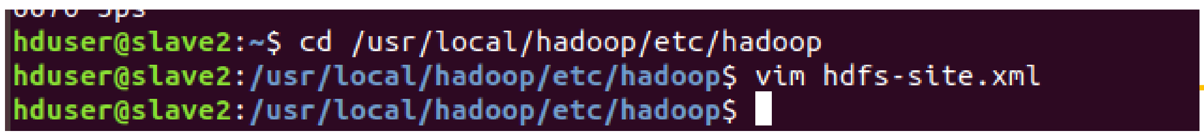
```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ vim hdfs-site.xml
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$
```



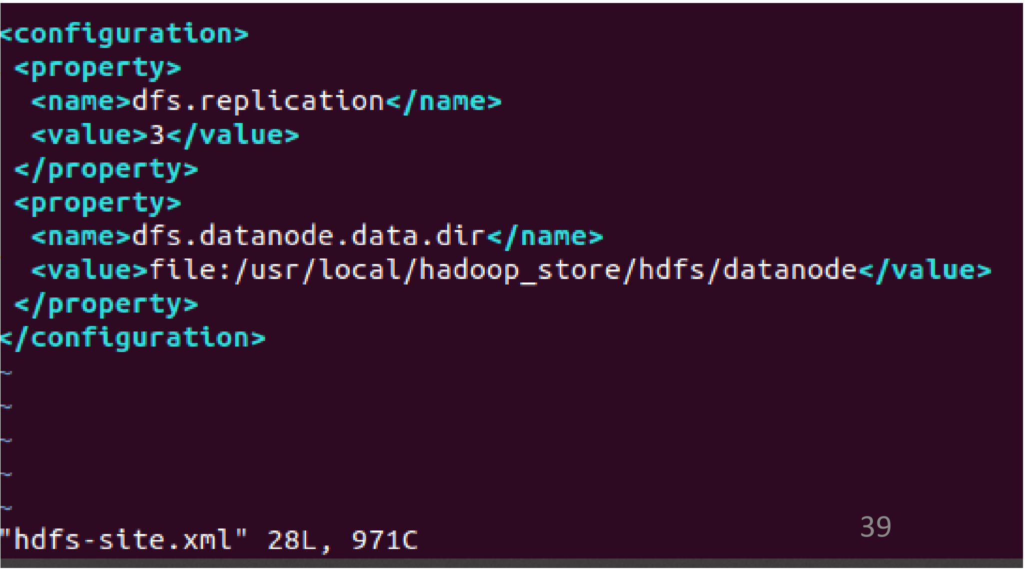
```
<!-- Put site-specific property overrides in this file. -->

<configuration>
 <property>
  <name>dfs.replication</name>
  <value>3</value>
 </property>
 <property>
  <name>dfs.namenode.name.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
 </property>
 <property>
  <name>dfs.datanode.data.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
 </property>
</configuration>
~
-- INSERT --
```



```
hduser@slave1:/usr/local/hadoop/etc/hadoop$ sudo vim hdfs-site.xml
[sudo] password for hduser:
hduser@slave1:/usr/local/hadoop/etc/hadoop$
```

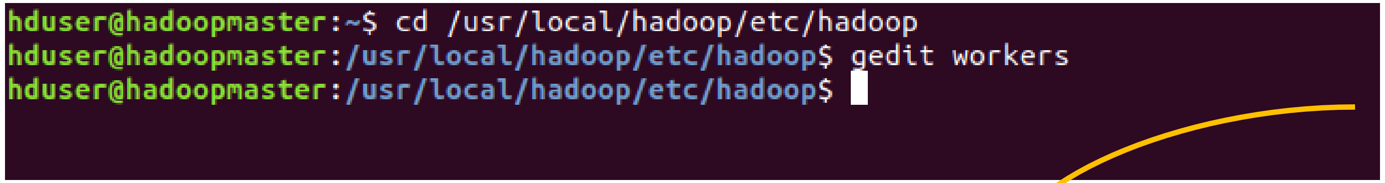


```
hduser@slave2:~$ cd /usr/local/hadoop/etc/hadoop
hduser@slave2:/usr/local/hadoop/etc/hadoop$ vim hdfs-site.xml
hduser@slave2:/usr/local/hadoop/etc/hadoop$ 
```

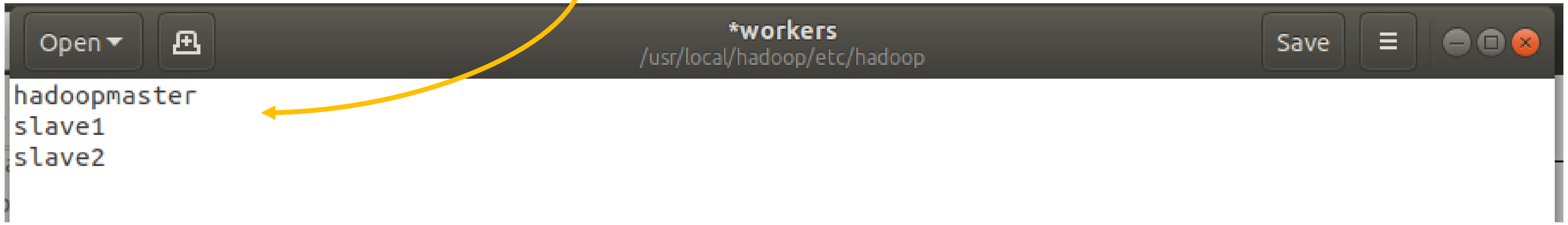


```
<configuration>
 <property>
  <name>dfs.replication</name>
  <value>3</value>
 </property>
 <property>
  <name>dfs.datanode.data.dir</name>
  <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
 </property>
</configuration>
~
~
~
~
~
"hdfs-site.xml" 28L, 971C
```

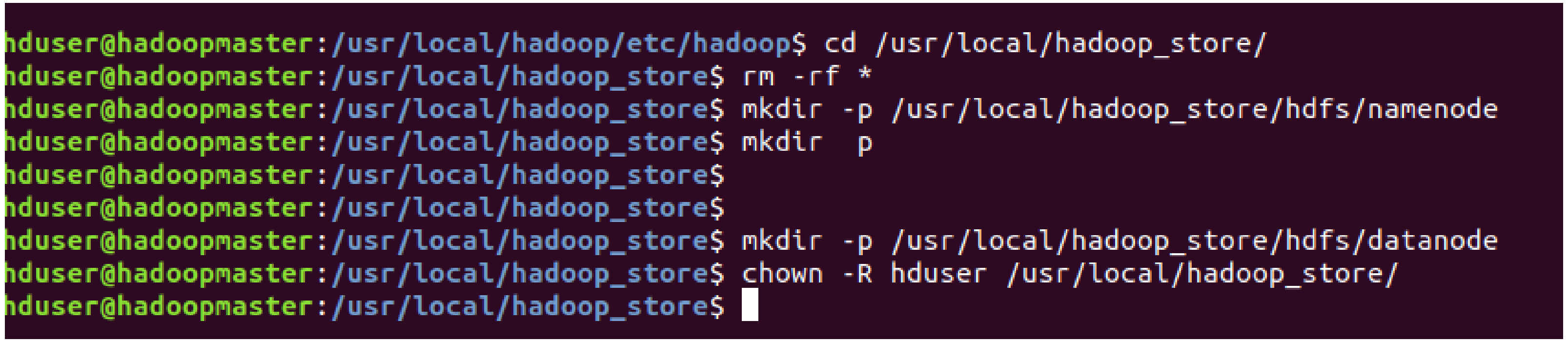
- Modification du fichier workers (slaves) dans hadoopmaster, slave1 et slave2



```
hduser@hadoopmaster:~$ cd /usr/local/hadoop/etc/hadoop
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ gedit workers
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ 
```

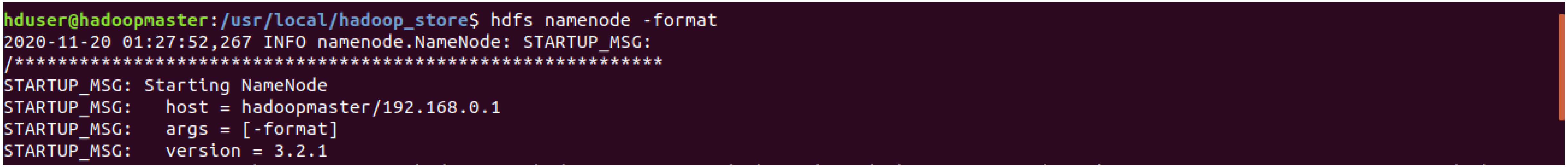


```
hadoopmaster
slave1
slave2
```

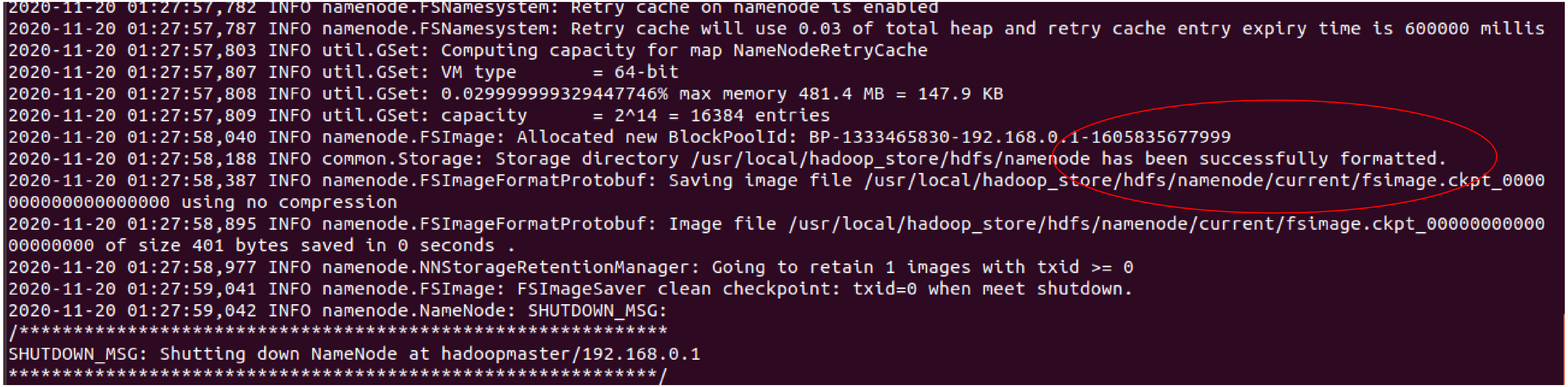


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cd /usr/local/hadoop_store/
hduser@hadoopmaster:/usr/local/hadoop_store$ rm -rf *
hduser@hadoopmaster:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/namenode
hduser@hadoopmaster:/usr/local/hadoop_store$ mkdir  p
hduser@hadoopmaster:/usr/local/hadoop_store$
hduser@hadoopmaster:/usr/local/hadoop_store$
hduser@hadoopmaster:/usr/local/hadoop_store$ mkdir -p /usr/local/hadoop_store/hdfs/datanode
hduser@hadoopmaster:/usr/local/hadoop_store$ chown -R hduser /usr/local/hadoop_store/
hduser@hadoopmaster:/usr/local/hadoop_store$ 
```

- On formate le namenode.

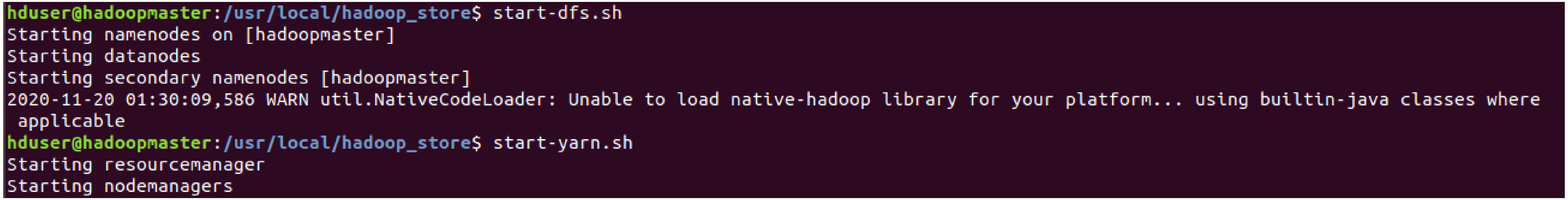


```
hduser@hadoopmaster:/usr/local/hadoop_store$ hdfs namenode -format
2020-11-20 01:27:52,267 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = hadoopmaster/192.168.0.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.2.1
```



```
2020-11-20 01:27:57,782 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2020-11-20 01:27:57,787 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
2020-11-20 01:27:57,803 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2020-11-20 01:27:57,807 INFO util.GSet: VM type       = 64-bit
2020-11-20 01:27:57,808 INFO util.GSet: 0.029999999329447746% max memory 481.4 MB = 147.9 KB
2020-11-20 01:27:57,809 INFO util.GSet: capacity      = 2^14 = 16384 entries
2020-11-20 01:27:58,040 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1333465830-192.168.0.1-1605835677999
2020-11-20 01:27:58,188 INFO common.Storage: Storage directory /usr/local/hadoop_store/hdfs/namenode has been successfully formatted.
2020-11-20 01:27:58,387 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.ckpt_0000000000000000000 using no compression
2020-11-20 01:27:58,895 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop_store/hdfs/namenode/current/fsimage.ckpt_0000000000000000000 of size 401 bytes saved in 0 seconds .
2020-11-20 01:27:58,977 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2020-11-20 01:27:59,041 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2020-11-20 01:27:59,042 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at hadoopmaster/192.168.0.1
************************************************************/
```

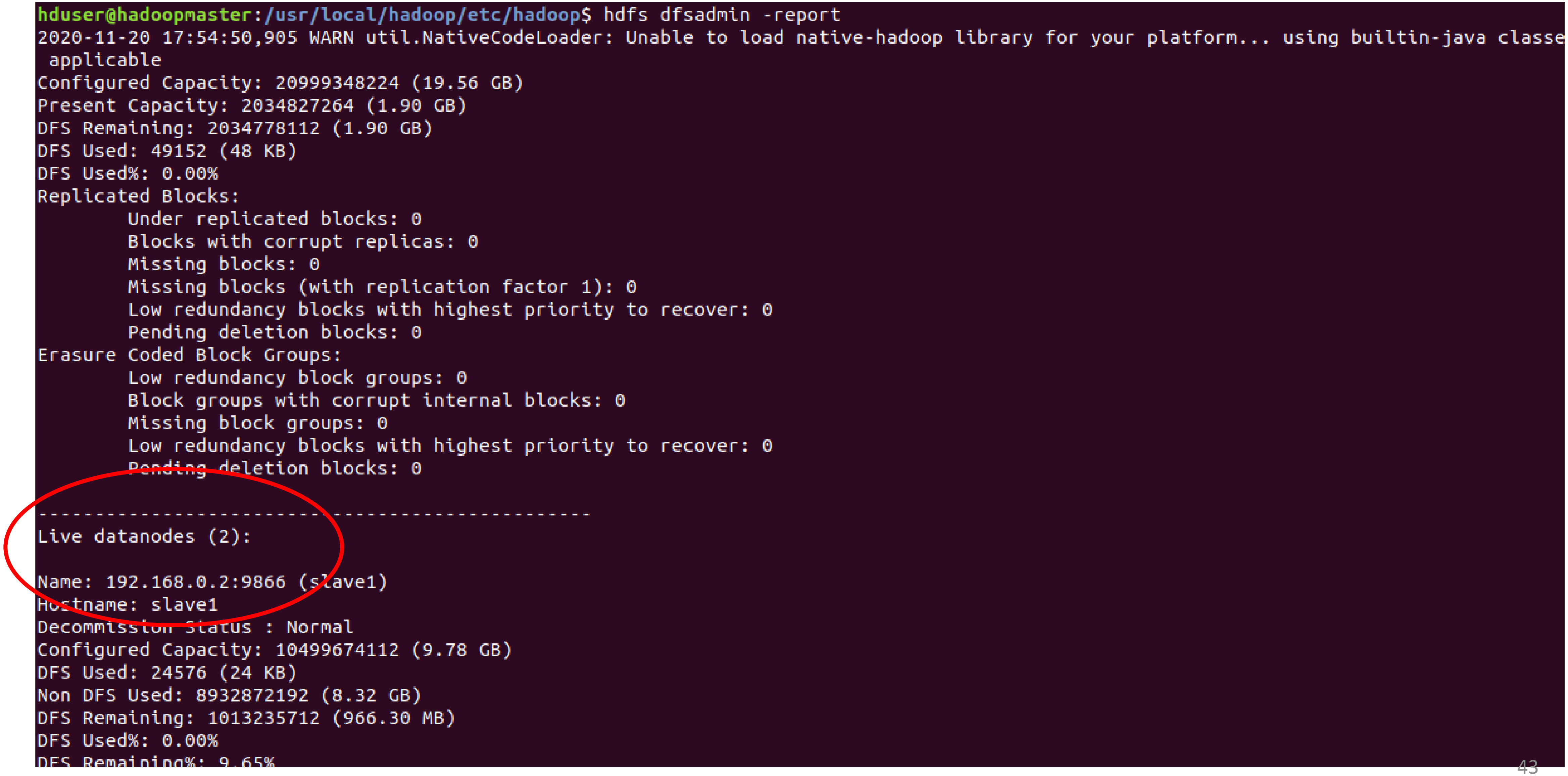
- Démarrage de Hadoop



```
hduser@hadoopmaster:/usr/local/hadoop_store$ start-dfs.sh
Starting namenodes on [hadoopmaster]
Starting datanodes
Starting secondary namenodes [hadoopmaster]
2020-11-20 01:30:09,586 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
hduser@hadoopmaster:/usr/local/hadoop_store$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
```

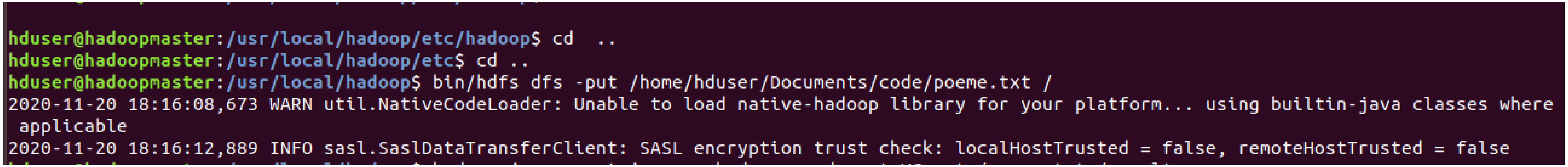
IV Exécution du programme MapReduce « Word Count » dans le cluster multi-nœuds

- On vérifie le bon fonctionnement de tous les noeuds du cluster

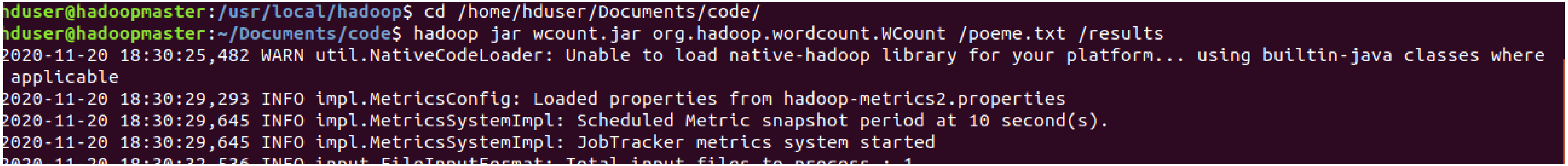


```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ hdfs dfsadmin -report
2020-11-20 17:54:50,905 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classe
 applicable
Configured Capacity: 20999348224 (19.56 GB)
Present Capacity: 2034827264 (1.90 GB)
DFS Remaining: 2034778112 (1.90 GB)
DFS Used: 49152 (48 KB)
DFS Used%: 0.00%
Replicated Blocks:
        Under replicated blocks: 0
        Blocks with corrupt replicas: 0
        Missing blocks: 0
        Missing blocks (with replication factor 1): 0
        Low redundancy blocks with highest priority to recover: 0
        Pending deletion blocks: 0
Erasure Coded Block Groups:
        Low redundancy block groups: 0
        Block groups with corrupt internal blocks: 0
        Missing block groups: 0
        Low redundancy blocks with highest priority to recover: 0
        Pending deletion blocks: 0

-------------------------------------------------
Live datanodes (2):

Name: 192.168.0.2:9866 (slave1)
Hostname: slave1
Decommission Status : Normal
Configured Capacity: 10499674112 (9.78 GB)
DFS Used: 24576 (24 KB)
Non DFS Used: 8932872192 (8.32 GB)
DFS Remaining: 1013235712 (966.30 MB)
DFS Used%: 0.00%
DFS Remaining%: 9.65%
```

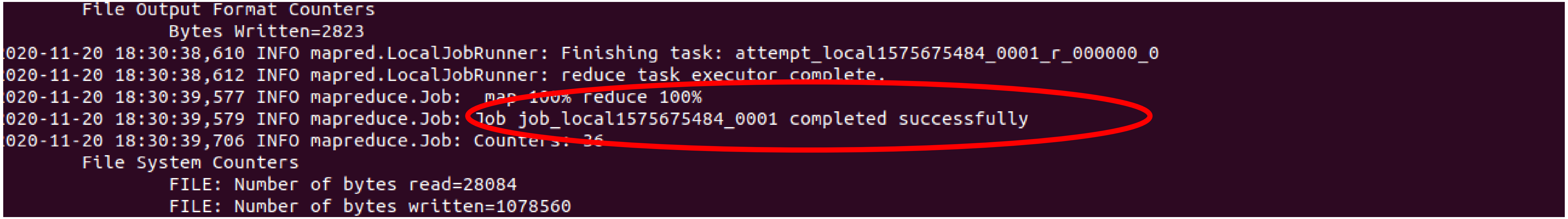
- On répète les mêmes étapes décrites dans la section (II) : Exécution d'un programme Map/Reduce dans un cluster à nœud unique.



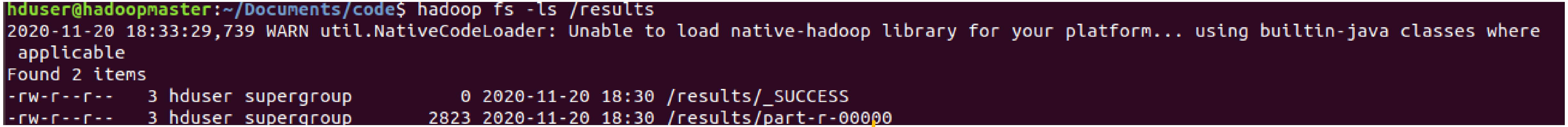
```
hduser@hadoopmaster:/usr/local/hadoop/etc/hadoop$ cd  ..
hduser@hadoopmaster:/usr/local/hadoop/etc$ cd ..
hduser@hadoopmaster:/usr/local/hadoop$ bin/hdfs dfs -put /home/hduser/Documents/code/poeme.txt /
2020-11-20 18:16:08,673 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
2020-11-20 18:16:12,889 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
```



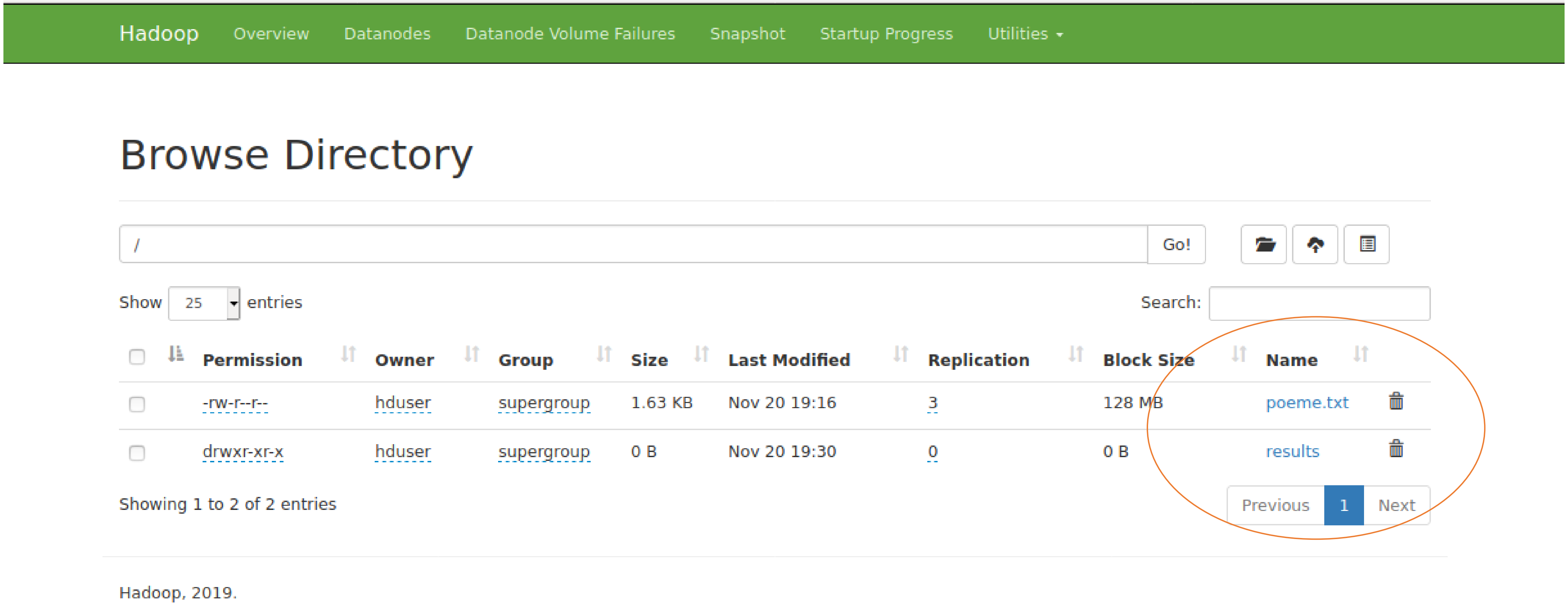
```
hduser@hadoopmaster:/usr/local/hadoop$ cd /home/hduser/Documents/code/
hduser@hadoopmaster:~/Documents/code$ hadoop jar wcount.jar org.hadoop.wordcount.WCount /poeme.txt /results
2020-11-20 18:30:25,482 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
2020-11-20 18:30:29,293 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2020-11-20 18:30:29,645 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-11-20 18:30:29,645 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2020-11-20 18:30:32,536 INFO input.FileInputFormat: Total input files to process : 1
```



```
        File Output Format Counters
                Bytes Written=2823
020-11-20 18:30:38,610 INFO mapred.LocalJobRunner: Finishing task: attempt_local1575675484_0001_r_000000_0
020-11-20 18:30:38,612 INFO mapred.LocalJobRunner: reduce task executor complete.
020-11-20 18:30:39,577 INFO mapreduce.Job:  map 100% reduce 100%
020-11-20 18:30:39,579 INFO mapreduce.Job: Job job_local1575675484_0001 completed successfully
020-11-20 18:30:39,706 INFO mapreduce.Job: Counters: 36
        File System Counters
                FILE: Number of bytes read=28084
                FILE: Number of bytes written=1078560
```



```
hduser@hadoopmaster:~/Documents/code$ hadoop fs -ls /results
2020-11-20 18:33:29,739 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
Found 2 items
-rw-r--r--   3 hduser supergroup          0 2020-11-20 18:30 /results/_SUCCESS
-rw-r--r--   3 hduser supergroup       2823 2020-11-20 18:30 /results/part-r-00000
```



Browse Directory

/ Go!

Show 25 entries Search:

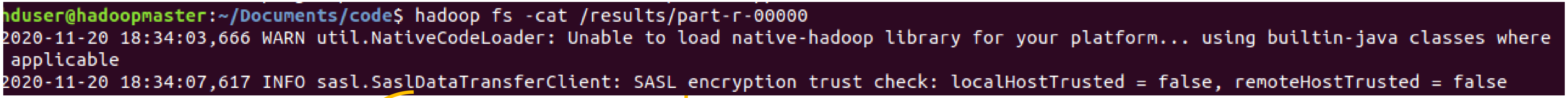
□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	-rw-r--r--	hduser	supergroup	1.63 KB	Nov 20 19:16	3	128 MB	poeme.txt
□	drwxr-xr-x	hduser	supergroup	0 B	Nov 20 19:30	0	0 B	results

Showing 1 to 2 of 2 entries

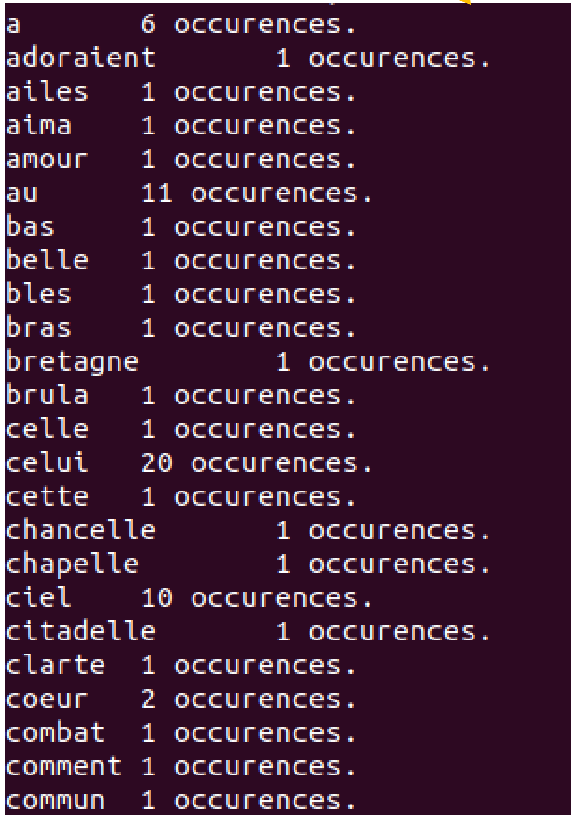
Previous 1 Next

Hadoop, 2019.

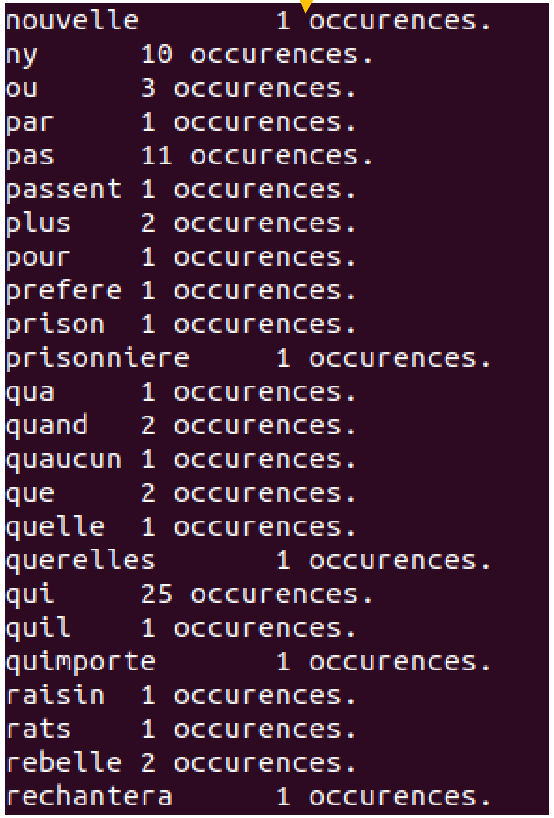
- Affichage de résultats



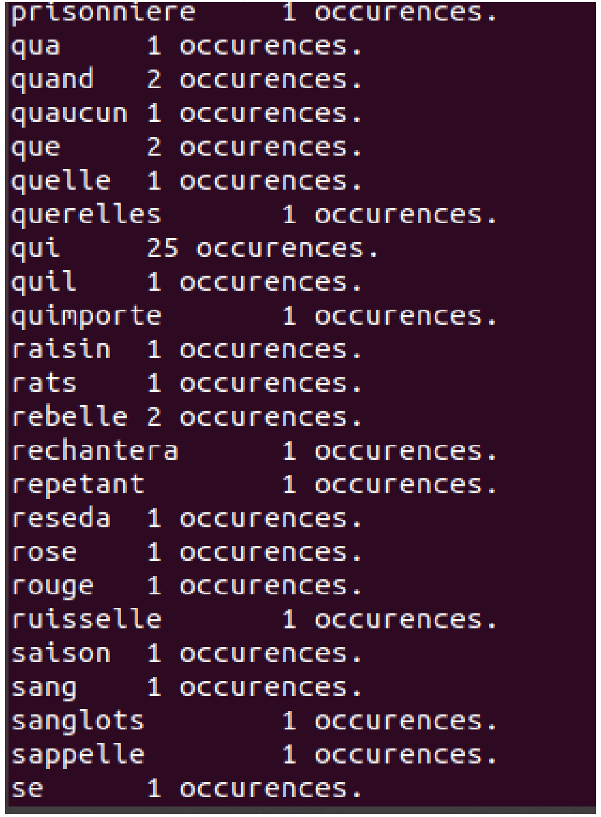
```
hduser@hadoopmaster:~/Documents/code$ hadoop fs -cat /results/part-r-00000
2020-11-20 18:34:03,666 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
 applicable
2020-11-20 18:34:07,617 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
```



```
a       6 occurences.
adoraient       1 occurences.
ailes   1 occurences.
aima    1 occurences.
amour   1 occurences.
au      11 occurences.
bas     1 occurences.
belle   1 occurences.
bles    1 occurences.
bras    1 occurences.
bretagne        1 occurences.
brula   1 occurences.
celle   1 occurences.
celui   20 occurences.
cette   1 occurences.
chancelle       1 occurences.
chapelle        1 occurences.
ciel    10 occurences.
citadelle       1 occurences.
clarte  1 occurences.
coeur   2 occurences.
combat  1 occurences.
comment 1 occurences.
commun  1 occurences.
```



```
nouvelle        1 occurences.
ny      10 occurences.
ou      3 occurences.
par     1 occurences.
pas     11 occurences.
passent 1 occurences.
plus    2 occurences.
pour    1 occurences.
prefere 1 occurences.
prison  1 occurences.
prisonniere     1 occurences.
qua     1 occurences.
quand   2 occurences.
quaucun 1 occurences.
que     2 occurences.
quelle  1 occurences.
querelles       1 occurences.
qui     25 occurences.
quil    1 occurences.
quimporte       1 occurences.
raisin  1 occurences.
rats    1 occurences.
rebelle 2 occurences.
rechantera      1 occurences.
```



```
prisonniere     1 occurences.
qua     1 occurences.
quand   2 occurences.
quaucun 1 occurences.
que     2 occurences.
quelle  1 occurences.
querelles       1 occurences.
qui     25 occurences.
quil    1 occurences.
quimporte       1 occurences.
raisin  1 occurences.
rats    1 occurences.
rebelle 2 occurences.
rechantera      1 occurences.
repetant        1 occurences.
reseda  1 occurences.
rose    1 occurences.
rouge   1 occurences.
ruisselle       1 occurences.
saison  1 occurences.
sang    1 occurences.
sanglots        1 occurences.
sappelle        1 occurences.
se      1 occurences.
```

.....

V

Extension : benchmark détaillé

Hadoop en tant que service (HDaaS) facilite l’approche des projets big data. Voici quelques-uns des principaux fournisseurs de services cloud Hadoop qui rendent cette course encore plus intéressante :

1)Amazon Web Service EMR (AWS EMR)

2)GOOGLE CLOUD

3)Cloudera

4)Microsoft Azure- HDInsight

5) IBM BigInsight

6) MapR

7)CSC

1) Amazon Web Service EMR (AWS EMR):

- Amazon EMR (Amazon Elastic Map Reduce) est actuellement l’un des principaux fournisseurs de services cloud Hadoop. En outre, Amazon EMR n’est pas seulement limité à Hadoop, mais fournit également des services à Spark et d’autres solutions Big Data.

- Amazon EMR facilite la création et la gestion de clusters élastiques et entièrement configurés d'instances Amazon EC2 exécutant Hadoop et d'autres applications dans l'écosystème Hadoop.

- **Avantages d’ EMR :**

Vitesse et souplesse accrues

Vous pouvez initialiser un cluster Hadoop de façon dynamique et rapide, ou ajouter des serveurs à votre cluster Amazon EMR existant, pour mettre les ressources plus rapidement à la disposition de vos utilisateurs et spécialistes des données. En utilisant Hadoop sur la plateforme AWS, votre organisation peut considérablement gagner en souplesse en diminuant le coût et le temps nécessaires à l'allocation des ressources à des fins d'expérimentation et de développement.

Réduction de la complexité administrative

La configuration d'Hadoop, la mise en réseau, l'installation du serveur, la configuration de la sécurité et la maintenance administrative courante peuvent s'avérer des tâches complexes et difficiles. Amazon EMR étant un service géré, la solution répond aux exigences de votre infrastructure Hadoop pour vous permettre de vous concentrer sur votre activité principale.

Vous pouvez exploiter facilement d'autres services AWS.

Vous pouvez facilement intégrer votre environnement Hadoop à d'autres services tels qu'Amazon S3, Amazon Kinesis, Amazon Redshift et Amazon DynamoDB pour permettre le transfert de données, les workflows et les analyses dans les nombreux services de la plate-forme AWS. En outre, vous pouvez utiliser le catalogue de données AWS Glue comme référentiel de métadonnées géré pour Apache Hive et Apache Spark.

Payez pour les clusters uniquement quand vous en avez besoin

De nombreuses tâches Hadoop sont par nature irrégulières. Par exemple, une tâche ETL peut être exécutée une fois par heure, par jour ou par mois, tandis que des tâches de modélisation effectuées pour des sociétés financières ou des séquençages génétiques peuvent avoir lieu seulement quelques fois par an. En utilisant Hadoop sur Amazon EMR, vous pouvez facilement lancer ces clusters de charges de travail, enregistrer les résultats et supprimer vos ressources Hadoop lorsque vous n'en avez plus besoin, pour éviter les coûts d'infrastructure inutiles.

- **<u>Les inconvénients :</u>**

Absence of Hadoop Management Console
AWS ne fournit aucune console de gestion similaire à Ambari ou Cloudera Manager d’Apache, pour EMR. Il est donc difficile de gérer et de surveiller divers services Hadoop sur un cluster en cours d’exécution.

No High Availability for Master Node
Le nœud maître du cluster EMR n’est pas configuré pour la haute disponibilité, ce qui en fait le point unique d’échec.

Automatic Replacement of Unhealthy Nodes
Le service EMR surveille en permanence les nœuds esclaves et remplace de manière transparente tout nœud malsain. Tout en offrant un avantage de maintenance distinct aux administrateurs, cette fonctionnalité risque également la perte de données résidant sur le nœud malsain.

2) GOOGLE CLOUD

Grâce à Dataproc, vous pouvez créer une ou plusieurs instances Compute Engine pouvant se connecter à une instance de Cloud Bigtable et exécuter des tâches Hadoop. On utiliser Dataproc pour automatiser les tâches suivantes :

- Installer Hadoop et le client HBase pour Java
- Configurer Hadoop et Cloud Bigtable
- Définir les champs d'application d'autorisations appropriés pour Cloud Bigtable

Après avoir créé un cluster Dataproc, vous pouvez l'utiliser pour exécuter des tâches Hadoop qui lisent et écrivent des données depuis et vers Cloud Bigtable.

3) Cloudera

CLOUDERA

Cloudera est également l'une des rares entreprises qui fournissent la mise en place complète pour Hadoop. En fait, Cloudera est le plus célèbre de tous.

CDH est un logiciel Apache 100% open source et est la seule solution Hadoop à offrir un traitement par lots unifié, sql interactif, recherche interactive, et des contrôles d'accès basés sur les rôles.

Vous pouvez commencer par Cloudera Free et utiliser les informations d'identification par défaut de Cloudera pour commencer. Si votre système dispose de 10 Go de RAM, vous pouvez également faire l'expérience de Cloudera Manager.

4) Microsoft Azure- HDInsight :

HDInsight est la distribution Hadoop alimentée par le cloud. Il a été conçu pour mettre à l'échelle et traiter les données à partir de téraoctets à pétaoctets.

Il s'agit d'une offre Hadoop cloud entièrement gérée qui fournit des clusters analytiques open source optimisée pour Spark, Hive, MapReduce, HBase, Storm, Kafka et R Server avec 99,9% SLA. Toutes ces technologies big data et applications ISV sont facilement déployables en tant que clusters gérés avec sécurité et surveillance au niveau de l'entreprise.

Les serveurs sont facilement configurables avec de nombreux outils de productivité tels que Datameer, Cask, AtScale et StreamSets.

Le service cloud Hadoop de Microsoft Azure est facile à gérer et à gérer pour les administrateurs. Avec HDInsight, vous pouvez traiter des données non structurées ou semi-structurées à partir de clics web, de médias sociaux, de journaux de serveurs, d'appareils et de capteurs, et plus encore.

5) IBM BigInsight:

IBM BigInsight est un important fournisseur de services cloud Hadoop qui fournit le service cloud sur l'infrastructure cloud mondiale IBM SoftLayer.
IBM Infosphere BigInsight ne nécessite aucune infrastructure locale et il prend en charge Big SQL, Big Sheets, et l'analyse de texte et plus encore, IBM affirme.
Voici quelques-unes des caractéristiques de l'édition standard IBM BigInsight-

- Entièrement intégré, entièrement compatible – Installation intégrée d'Apache Hadoop et des composants open source associés de l'écosystème Apache Hadoop qui est testé et préconfiguré.
- Comprend Jaql, un langage de requête déclaratif, pour faciliter l'analyse des données structurées et non structurées.
- Fournit une console de gestion web pour faciliter l'administration et les vues en temps réel.
- Comprend BigSheets, un outil d'analyse et de visualisation web doté d'une interface familière, semblable à une feuille de calcul, qui permet d'analyser facilement de grandes quantités de données et des tâches de collecte de données à long terme.

- Inclut Big SQL, un moteur de requête SQL natif qui permet l'accès SQL aux données stockées dans BigInsights, en tirant parti de MapReduce pour des jeux de données complexes et un accès direct pour les requêtes plus petites.

6) MapR :

MapR fournit la distribution complète pour Hadoop et est un environnement complet travaillant sur Hadoop 2.0. MapR est également l'un des plus grands fournisseurs d'Apache Hadoop.

La distribution complète de MapR comprend- Apache Hive, Apache Pig, Cascading, Apache HCatalog, Apache HBase™, Apache Oozie, Apache Flume, Apache Sqoop, Apache Mahout et Apache Whirr.

Voici quelques-unes des caractéristiques de MapR:

- Rentabilité
- tolérant aux défauts
- flexible
- évolutif

7) CSC :

CSC est également l’un des principaux fournisseurs de services cloud Big Data Hadoop dans le monde. Ils fournissent un programme intégré entièrement géré.

CSC Big Data Platform as a Service (BDPaaS) aide les entreprises à franchir ces obstacles et à tirer de la valeur de leurs données beaucoup plus rapidement. Avec BDPaaS, les entreprises peuvent rapidement développer, sécuriser et déployer des applications big data et analytiques de nouvelle génération avec une plate-forme centralisée basée sur l’abonnement qui utilise des outils d’analyse, une infrastructure et des logiciels de pointe.

Le logiciel comprend une prise en charge intégrée de Cloudera, Hortonworks, DataStax, Spark, Pentaho, Qlik, Tableau, R, Python, et plus encore.

- Voici quelques-unes des caractéristiques de BDPaaS-
- Augmenter les taux de réussite
- Accélérer l’heure de la valeur
- Réduire les coûts grâce à des applications open-source
- Protéger les données grâce à la sécurité d’entreprise multicouches
- Activer le développement rapide des applications big data et bien d’autres

Fin

Merci