Hadoop Deployment

Monday 21st February, 2022

Contents

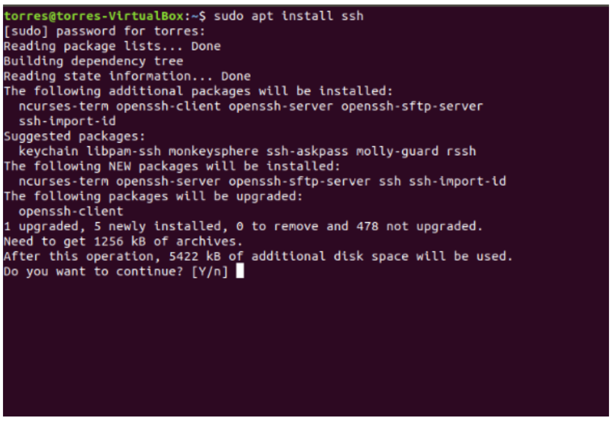
1 Preinstall

1.1 ssh connection

pdsh can run multiple remote commands in parallel.

Listing 1: Install SSH and PDSH

```
sudo apt install ssh
```

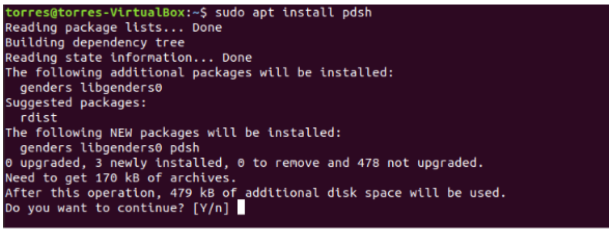


```
torres@torres-VirtualBox:~$ sudo apt install ssh
[sudo] password for torres:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-client openssh-server openssh-sftp-server
  ssh-import-id
Suggested packages:
  keychain libpam-ssh monkeysphere ssh-askpass molly-guard rssh
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh ssh-import-id
The following packages will be upgraded:
  openssh-client
1 upgraded, 5 newly installed, 0 to remove and 478 not upgraded.
Need to get 1256 kB of archives.
After this operation, 5422 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Figure 1

Listing 2: Install SSH and PDSH

```
sudo apt install pdsh
```



```
torres@torres-VirtualBox:~$ sudo apt install pdsh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  genders libgenders0
Suggested packages:
  rdist
The following NEW packages will be installed:
  genders libgenders0 pdsh
0 upgraded, 3 newly installed, 0 to remove and 478 not upgraded.
Need to get 170 kB of archives.
After this operation, 479 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Figure 2

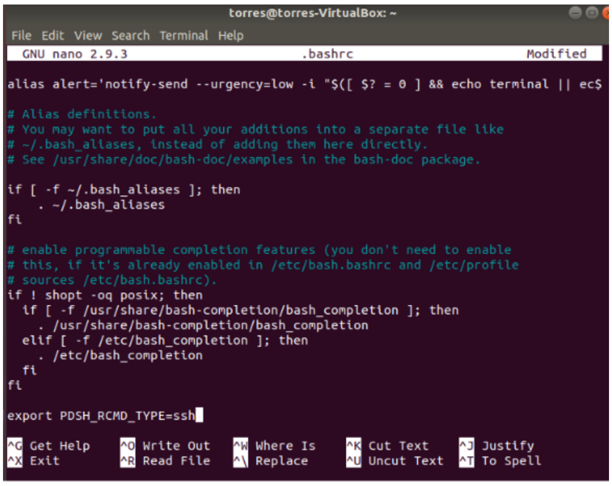
Listing 3: Install PDSH

```
nano .bashrc
```

At the end of the file just write the following line:

```
export PDSH_RCMD_TYPE=ssh
```

Now let's configure SSH. Let's create a new key using the following command:



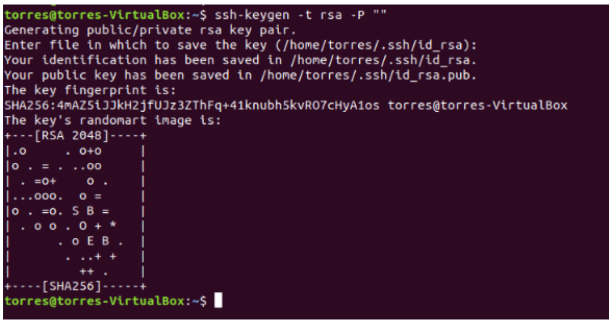
torres@torres-VirtualBox: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                     .bashrc                         Modified

alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || ec$

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export PDSH_RCMD_TYPE=ssh

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell
```

Figure 3

Listing 4: ssh-keygen

```
ssh-keygen -t rsa -P ""
```

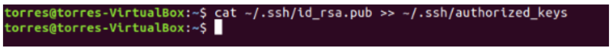


```
torres@torres-VirtualBox:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/torres/.ssh/id_rsa):
Your identification has been saved in /home/torres/.ssh/id_rsa.
Your public key has been saved in /home/torres/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:4mAZ5iJJkH2jfUJz3ZThFq+41knubh5kvRO7cHyA1os torres@torres-VirtualBox
The key's randomart image is:
+---[RSA 2048]----+
|.o      . o+o    |
|o . = . ..oo     |
| . =o+    o .    |
|...ooo.  o =     |
|o . =o. S B =    |
| . o o . O + *   |
|      . o E B .  |
|       . ..+ +   |
|         ++ .    |
+----[SHA256]-----+
torres@torres-VirtualBox:~$
```

Figure 4

Listing 5: ssh-keygen

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



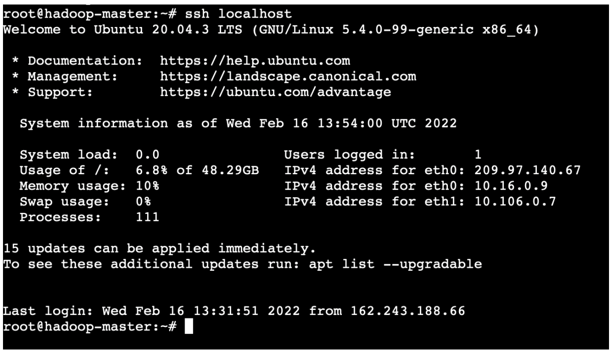
```
torres@torres-VirtualBox:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
torres@torres-VirtualBox:~$
```

Figure 5

Now we can verify the SSH configuration by connecting to the localhost:

Listing 6: ssh local host

```
ssh localhost
```



```
root@hadoop-master:~# ssh localhost
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-99-generic x86_64)

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

  System information as of Wed Feb 16 13:54:00 UTC 2022

  System load:  0.0               Users logged in:       1
  Usage of /:   6.8% of 48.29GB   IPv4 address for eth0: 209.97.140.67
  Memory usage: 10%               IPv4 address for eth0: 10.16.0.9
  Swap usage:   0%                IPv4 address for eth1: 10.106.0.7
  Processes:    111

15 updates can be applied immediately.
To see these additional updates run: apt list --upgradable


Last login: Wed Feb 16 13:31:51 2022 from 162.243.188.66
root@hadoop-master:~# █
```

Figure 6

1.2 java installation

Install OpenJDK, the default Java Development Kit

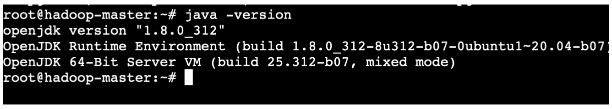
Listing 7: jdk

```
sudo apt install default-jdk
```

This step isn't really a step, it's just to check if Java is now correctly installed:

Listing 8: version check

```
java -version
```



```
root@hadoop-master:~# java -version
openjdk version "1.8.0_312"
OpenJDK Runtime Environment (build 1.8.0_312-8u312-b07-0ubuntu1~20.04-b07
OpenJDK 64-Bit Server VM (build 25.312-b07, mixed mode)
root@hadoop-master:~# █
```

Figure 7

2 Installing Hadoop

2.1 hadoop version

In example I used the hadoop-3.0.3 version but we can use any version that we want to change the version in this link:

https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/

2.2 download hadoop

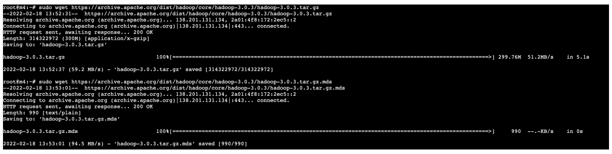
Download Hadoop using the following command:

Listing 9: version check

```
sudo wget https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz
```

Listing 10: version check

```
sudo wget
    https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz.mds
```



```
root@m4:~# sudo wget https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz
--2022-02-18 13:52:31--  https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz
Resolving archive.apache.org (archive.apache.org)... 138.201.131.134, 2a01:4f8:172:2ec5::2
Connecting to archive.apache.org (archive.apache.org)|138.201.131.134|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 314322972 (300M) [application/x-gzip]
Saving to: ‘hadoop-3.0.3.tar.gz’

hadoop-3.0.3.tar.gz                  100%[==========================================================================================>] 299.76M  51.2MB/s    in 5.1s

2022-02-18 13:52:37 (59.2 MB/s) - ‘hadoop-3.0.3.tar.gz’ saved [314322972/314322972]

root@m4:~# sudo wget https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz.mds
--2022-02-18 13:53:01--  https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/hadoop-3.0.3.tar.gz.mds
Resolving archive.apache.org (archive.apache.org)... 138.201.131.134, 2a01:4f8:172:2ec5::2
Connecting to archive.apache.org (archive.apache.org)|138.201.131.134|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 990 [text/plain]
Saving to: ‘hadoop-3.0.3.tar.gz.mds’

hadoop-3.0.3.tar.gz.mds              100%[==========================================================================================>]     990  --.-KB/s    in 0s

2022-02-18 13:53:01 (94.5 MB/s) - ‘hadoop-3.0.3.tar.gz.mds’ saved [990/990]
```

Figure 8

Then run the verification:

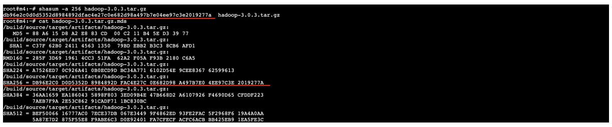
Listing 11: version check

```
shasum -a 256 hadoop-3.0.3.tar.gz
```

Compare this value with the SHA-256 value in the .mds file:

Listing 12: version check

```
cat hadoop-3.0.3.tar.gz.mds
```



```
root@m4:~# shasum -a 256 hadoop-3.0.3.tar.gz
db96e2c0d0d5352d8984892dfac4e27c0e682d98a497b7e04ee97c3e2019277a  hadoop-3.0.3.tar.gz
root@m4:~# cat hadoop-3.0.3.tar.gz.mds
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
   MD5 = 88 A6 15 D8 A2 E8 83 CD  00 C2 11 B4 5E D3 39 77
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
  SHA1 = C37F 62B0 2411 4563 1350  79BD EBB2 B3C3 8CB6 AFD1
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
RMD160 = 285F 3D69 1961 4CC3 51FA  62A2 F05A F93B 2180 C6A5
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
SHA224 = A7526ED7 0C926A41 0B0ECD9D BC34A771 6102D54E 9CEE8367 62599613
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
SHA256 = DB96E2C0 D0D5352D 8984892D FAC4E27C 0E682D98 A497B7E0 4EE97C3E 2019277A
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
SHA384 = 36AA1659 EA186043 5898F803 3ED09B4E 47B668D2 A6107926 F4690D65 CFDDF223
         7AEB7F9A 2E53C862 91CADF71 1BC830BC
/build/source/target/artifacts/hadoop-3.0.3.tar.gz:
SHA512 = BEF50066 16777AC0 7ECE37DB 067E3449 9F4862ED 93FE2FAC 5F2968F6 19A4A0AA
         5A87E7D2 875F55E8 F9ABE6C3 D0E92401 FA7CFECF ACFC6ACB BB425EB9 1EA5FE3C
```

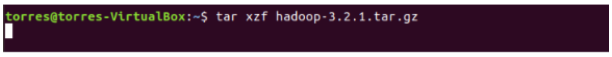
Figure 9

2.3 extract hadoop tar

Now that we've verified that the file wasn't corrupted or changed, we'll use the tar command with the **-x** flag to extract, **-z** to uncompress, **-v** for verbose output, and **-f** to specify that we're extracting from a file. Use tab-completion or substitute the correct version number in the command below:

Listing 13: Output

```
tar -xzvf hadoop-3.0.3.tar.gz
```



```
torres@torres-VirtualBox:~$ tar xzf hadoop-3.2.1.tar.gz
```

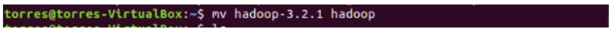
Figure 10

Finally, we'll move the extracted files into /usr/local, the appropriate place for locally installed software. Change the version number, if needed, to match the version you downloaded.

2.4 move extracted file

Listing 14: Move

```
sudo mv hadoop-3.0.3 /usr/local/hadoop
```



```
torres@torres-VirtualBox:~$ mv hadoop-3.2.1 hadoop
```

Figure 11

3 Configuring Hadoop's Java Home

The path to Java, **/usr/bin/java** is a symlink to **/etc/alternatives/java**, which is in turn a symlink to default Java binary. We will use **readlink** with the -f flag to follow every symlink in every part of the path, recursively. Then, we'll use sed to trim bin/java from the output to give us the correct value for JAVA HOME.

To find the default Java path:

Listing 15: java path

```
readlink -f /usr/bin/java | sed "s:bin/java::"
```

We can copy this output to set Hadoop's Java home to this specific version, which ensures that if the default Java changes, this value will not.

To begin, open **hadoop-env.sh**:

Listing 16: java path

```
sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
```

3.1 option 1: set a static value

Listing 17: /usr/local/hadoop/etc/hadoop/hadoop-env.sh

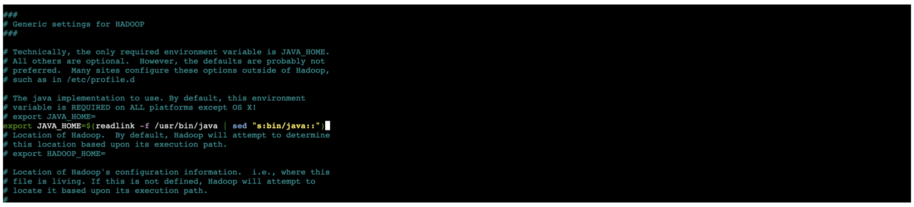
```
 . . .
#export JAVA_HOME=${JAVA_HOME}
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64/
 . . .
```

3.2 option 2: use readlink to set the value dynamically

It is prefered setting dynamically.

Listing 18: /usr/local/hadoop/etc/hadoop/hadoop-env.sh

```
 . . .
#export JAVA_HOME=${JAVA_HOME}
export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java::")
 . . .
```



```
###
# Generic settings for HADOOP
###

# Technically, the only required environment variable is JAVA_HOME.
# All others are optional.  However, the defaults are probably not
# preferred.  Many sites configure these options outside of Hadoop,
# such as in /etc/profile.d

# The java implementation to use. By default, this environment
# variable is REQUIRED on ALL platforms except OS X!
# export JAVA_HOME=
export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java::")
# Location of Hadoop.  By default, Hadoop will attempt to determine
# this location based upon its execution path.
# export HADOOP_HOME=

# Location of Hadoop's configuration information.  i.e., where this
# file is living. If this is not defined, Hadoop will attempt to
# locate it based upon its execution path.
#
```

Figure 12

3.3 set the value on environment

Open the environment file on nano with this command:

Listing 19: s

```
sudo nano /etc/environment
```

Then, add the following configurations: (**Path is in one line**)

Listing 20: sudo nano /etc/environment

```
PATH=
"/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/
local/games:/usr/local/hadoop/bin:/usr/local/hadoop/sbin"
JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java::")
```

4 Create Hadoop User

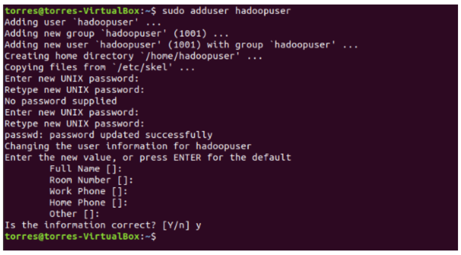
we will add a user called **hadoopuser**, and we will set up it's configurations:

Listing 21: sudo nano /etc/environment

```
sudo adduser hadoopuser
```

Provide the password and you can leave the rest blank, just press Enter.

.



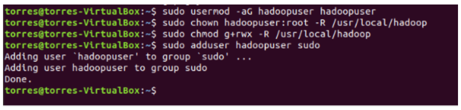
```
torres@torres-VirtualBox:~$ sudo adduser hadoopuser
Adding user `hadoopuser' ...
Adding new group `hadoopuser' (1001) ...
Adding new user `hadoopuser' (1001) with group `hadoopuser' ...
Creating home directory `/home/hadoopuser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
No password supplied
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hadoopuser
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
torres@torres-VirtualBox:~$
```

Figure 13

Now type these commands :

Listing 22: a

```
sudo usermod -aG hadoopuser hadoopuser
sudo chown hadoopuser:root -R /usr/local/hadoop/
sudo chmod g+rwx -R /usr/local/hadoop/
sudo adduser hadoopuser sudo
```



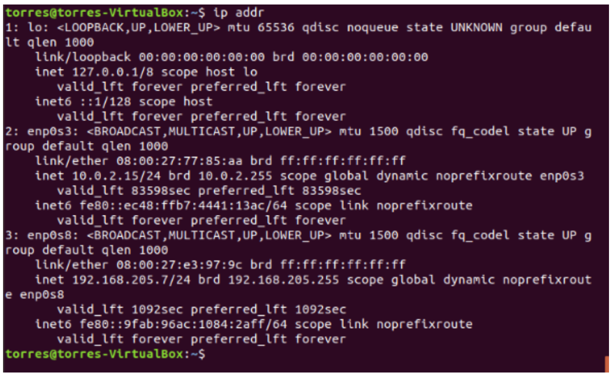
```
torres@torres-VirtualBox:~$ sudo usermod -aG hadoopuser hadoopuser
torres@torres-VirtualBox:~$ sudo chown hadoopuser:root -R /usr/local/hadoop
torres@torres-VirtualBox:~$ sudo chmod g+rwx -R /usr/local/hadoop
torres@torres-VirtualBox:~$ sudo adduser hadoopuser sudo
Adding user `hadoopuser' to group `sudo' ...
Adding user hadoopuser to group sudo
Done.
torres@torres-VirtualBox:~$
```

Figure 14

4.1 check your ip address

Listing 23: a

```
ip addr
```



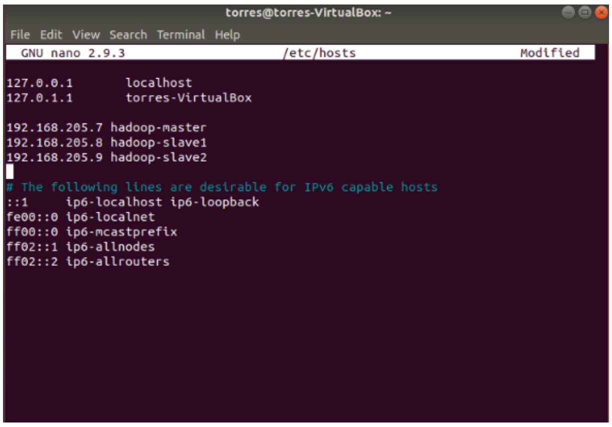
```
torres@torres-VirtualBox:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defau
lt qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
roup default qlen 1000
    link/ether 08:00:27:77:85:aa brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 83598sec preferred_lft 83598sec
    inet6 fe80::ec48:ffb7:4441:13ac/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
roup default qlen 1000
    link/ether 08:00:27:e3:97:9c brd ff:ff:ff:ff:ff:ff
    inet 192.168.205.7/24 brd 192.168.205.255 scope global dynamic noprefixrout
e enp0s8
       valid_lft 1092sec preferred_lft 1092sec
    inet6 fe80::9fab:96ac:1084:2aff/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
torres@torres-VirtualBox:~$
```

Figure 15

Open the hosts file and insert your Network configurations:

Listing 24: a

```
sudo nano /etc/hosts
```



torres@torres-VirtualBox: ~
File Edit View Search Terminal Help

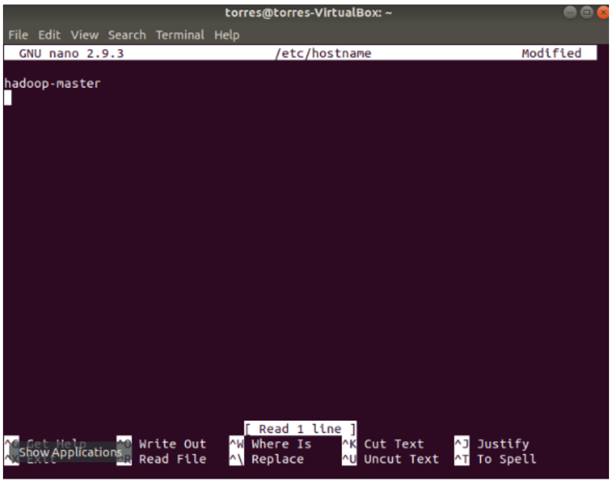
```
  GNU nano 2.9.3                  /etc/hosts                         Modified

127.0.0.1       localhost
127.0.1.1       torres-VirtualBox

192.168.205.7 hadoop-master
192.168.205.8 hadoop-slave1
192.168.205.9 hadoop-slave2

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Figure 16

On the master , open the hostname file on nano:

Listing 25: a

```
sudo nano /etc/hostname
```

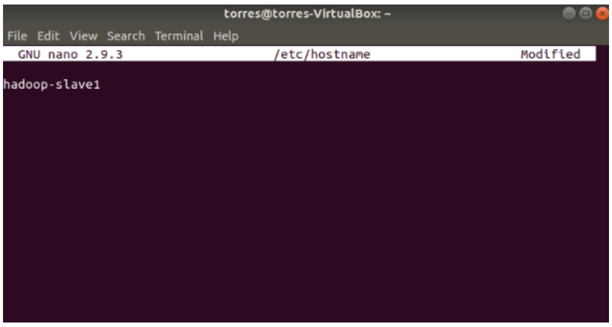


torres@torres-VirtualBox: ~
File Edit View Search Terminal Help
Show Applications

```
  GNU nano 2.9.3                  /etc/hostname                      Modified

hadoop-master

                              [ Read 1 line ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell
```

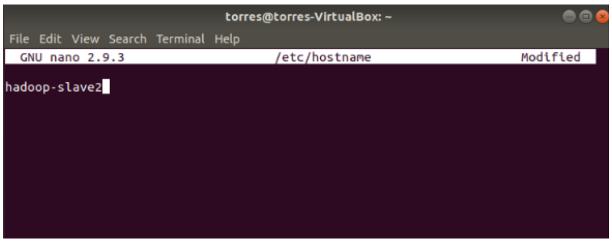
Figure 17

You should do the :same for slave nodes:
:


torres@torres-VirtualBox: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                /etc/hostname                       Modified

hadoop-slave1
```

Figure 18


torres@torres-VirtualBox: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                /etc/hostname                       Modified

hadoop-slave2
```

Figure 19

Also, you should reboot all of them so this configuration taked effect:

Listing 26: a

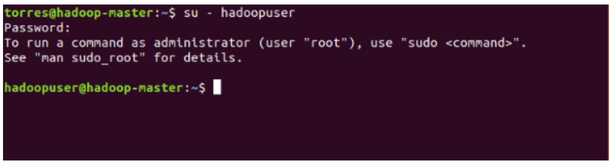
```
sudo reboot
```

5 On Hadoop User

Configure the SSH on hadoop-master, with the hadoopuser. This is the command:

Listing 27: a

```
su - hadoopuser
```



```
torres@hadoop-master:~$ su - hadoopuser
Password:
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

hadoopuser@hadoop-master:~$ 
```

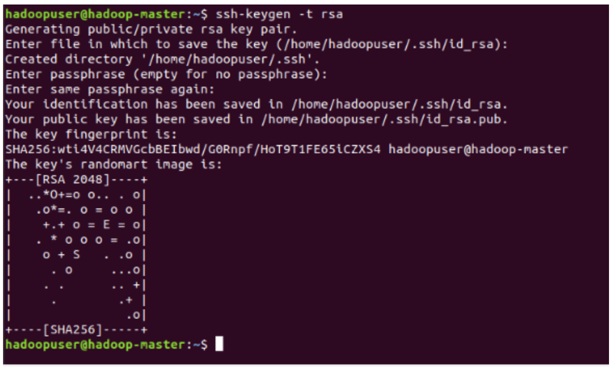
Figure 20

5.1 ssh

Create an SSH key:

Listing 28: a

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



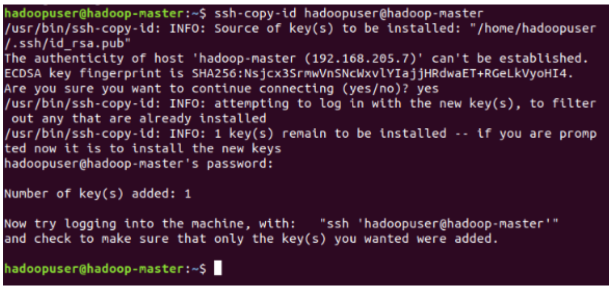
```
hadoopuser@hadoop-master:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoopuser/.ssh/id_rsa):
Created directory '/home/hadoopuser/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoopuser/.ssh/id_rsa.
Your public key has been saved in /home/hadoopuser/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:wti4V4CRMVGcbBEIbwd/G0Rnpf/HoT9T1FE65iCZXS4 hadoopuser@hadoop-master
The key's randomart image is:
+---[RSA 2048]----+
|  ..*O+=o o.. . o|
|   .o*=. o = o o |
|    +.+ o = E = o|
|   . * o o o = .o|
|    o + S   . .o |
|     . o     ...o|
|    . .      .. +|
|     .        .+ |
|               .o|
+----[SHA256]-----+
hadoopuser@hadoop-master:~$
```

Figure 21

Now we need to copy the SSH key to all the users. Use this command:

Listing 29: a

```
ssh-copy-id hadoopuser@hadoop-master
```



```
hadoopuser@hadoop-master:~$ ssh-copy-id hadoopuser@hadoop-master
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hadoopuser
/.ssh/id_rsa.pub"
The authenticity of host 'hadoop-master (192.168.205.7)' can't be established.
ECDSA key fingerprint is SHA256:NsjcxSrmwVnSNcWxvlYIajjHRdwaET+RGeLkVyoHI4.
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: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
hadoopuser@hadoop-master's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'hadoopuser@hadoop-master'"
and check to make sure that only the key(s) you wanted were added.

hadoopuser@hadoop-master:~$
```

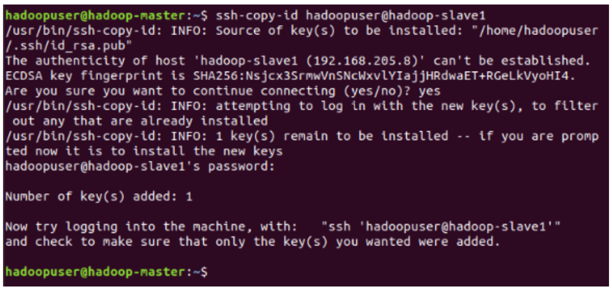
Figure 22

Listing 30: a

```
ssh-copy-id hadoopuser@hadoop-slave1
```

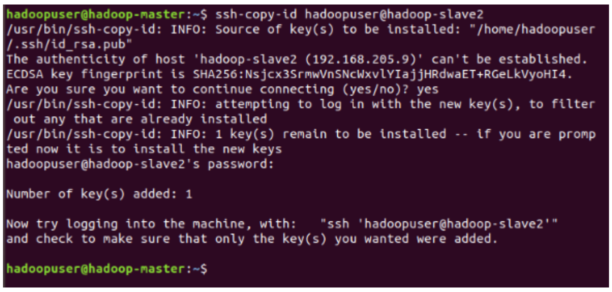
Listing 31: a

```
ssh-copy-id hadoopuser@hadoop-slave2
```



```
hadoopuser@hadoop-master:~$ ssh-copy-id hadoopuser@hadoop-slave1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hadoopuser
/.ssh/id_rsa.pub"
The authenticity of host 'hadoop-slave1 (192.168.205.8)' can't be established.
ECDSA key fingerprint is SHA256:Nsjcx3SrmwVnSNcWxvlYIajjHRdwaET+RGeLkVyoHI4.
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: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
hadoopuser@hadoop-slave1's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'hadoopuser@hadoop-slave1'"
and check to make sure that only the key(s) you wanted were added.

hadoopuser@hadoop-master:~$
```

Figure 23



```
hadoopuser@hadoop-master:~$ ssh-copy-id hadoopuser@hadoop-slave2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hadoopuser
/.ssh/id_rsa.pub"
The authenticity of host 'hadoop-slave2 (192.168.205.9)' can't be established.
ECDSA key fingerprint is SHA256:Nsjcx3SrmwVnSNcWxvlYIajjHRdwaET+RGeLkVyoHI4.
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: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
hadoopuser@hadoop-slave2's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'hadoopuser@hadoop-slave2'"
and check to make sure that only the key(s) you wanted were added.

hadoopuser@hadoop-master:~$
```

Figure 24

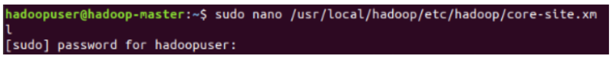
6 Add Hadoop Services

6.1 core-site.xml

On hadoop-master, open core-site.xml file on nano:

Listing 32: a

```
sudo nano /usr/local/hadoop/etc/hadoop/core-site.xml
```



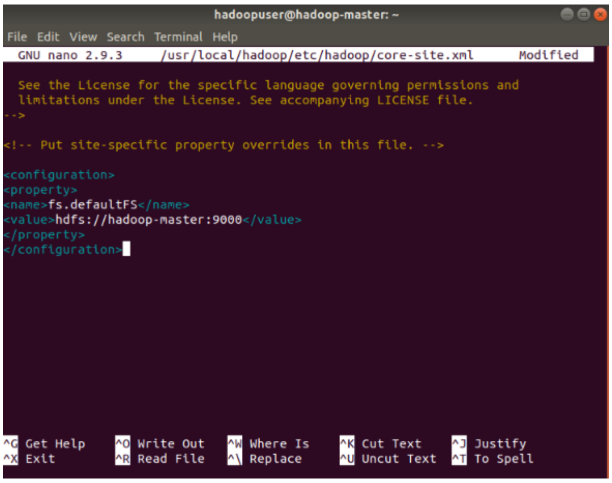
```
hadoopuser@hadoop-master:~$ sudo nano /usr/local/hadoop/etc/hadoop/core-site.xm
l
[sudo] password for hadoopuser:
```

Figure 25

Then add the following configurations:

Listing 33: a

```
<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://hadoop-master:9000</value>
</property>
</configuration>
```



hadoopuser@hadoop-master: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3        /usr/local/hadoop/etc/hadoop/core-site.xml        Modified

  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://hadoop-master:9000</value>
</property>
</configuration>

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell
```

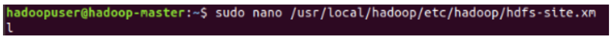
Figure 26

6.2 hdfs-site.xml

Still on hadoop-master, open the hdfs-site.xml file.

Listing 34: a

```
sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml
```



```
hadoopuser@hadoop-master:~$ sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xm
l
```

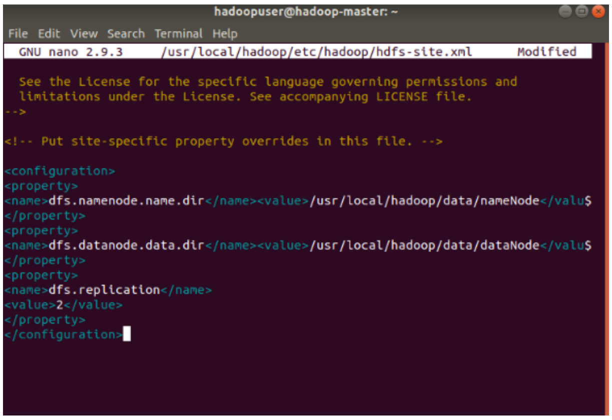
Figure 27

Add the following configurations:

Listing 35: a

```
<configuration>
<property>
<name>dfs.namenode.name.dir</name><value>/usr/local/hadoop/data/nameNode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name><value>/usr/local/hadoop/data/dataNode</value>
</property>
<property>
<name>dfs.replication</name>
<value>2</value>
</property>
</configuration>
```

.



hadoopuser@hadoop-master: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3      /usr/local/hadoop/etc/hadoop/hdfs-site.xml       Modified

  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

<configuration>
<property>
<name>dfs.namenode.name.dir</name><value>/usr/local/hadoop/data/nameNode</valu$
</property>
<property>
<name>dfs.datanode.data.dir</name><value>/usr/local/hadoop/data/dataNode</valu$
</property>
<property>
<name>dfs.replication</name>
<value>2</value>
</property>
</configuration>
```

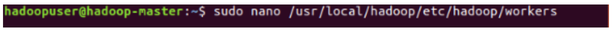
Figure 28

6.3 copy to workers

We're still on hadoop-master, let's open the workers file:

Listing 36: a

```
sudo nano /usr/local/hadoop/etc/hadoop/workers
```



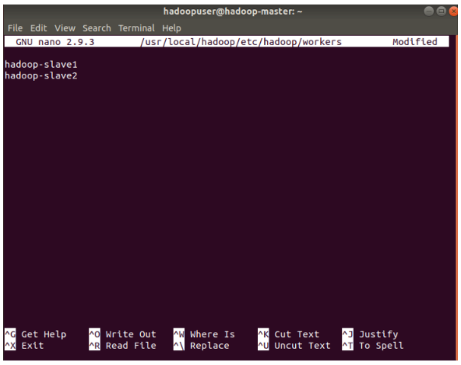
```
hadoopuser@hadoop-master:~$ sudo nano /usr/local/hadoop/etc/hadoop/workers
```

Figure 29

Add these two lines: (the slave names, remember the hosts file?)

Listing 37: a

```
hadoop-slave1
hadoop-slave2
```



hadoopuser@hadoop-master: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3          /usr/local/hadoop/etc/hadoop/workers           Modified

hadoop-slave1
hadoop-slave2

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell
```

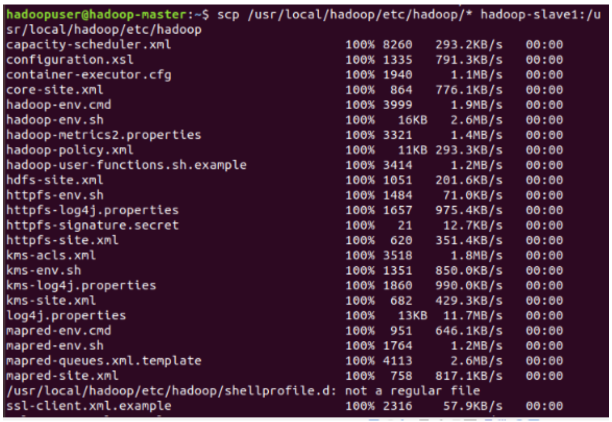
Figure 30

.

We need to copy the Hadoop Master configurations to the slaves, to do that we use these commands:

Listing 38: a

```
scp /usr/local/hadoop/etc/hadoop/* hadoop-slave1:/usr/local/hadoop/etc/hadoop//
```

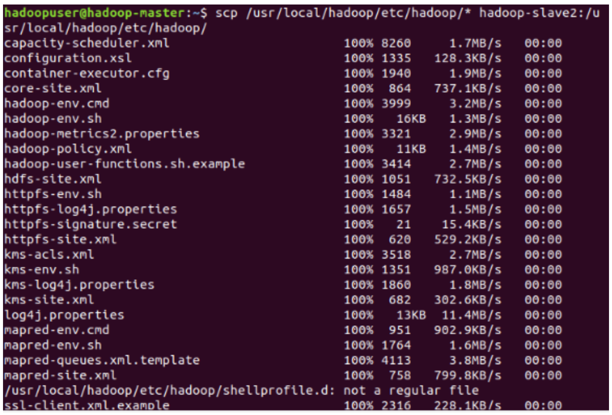


```
hadoopuser@hadoop-master:~$ scp /usr/local/hadoop/etc/hadoop/* hadoop-slave1:/u
sr/local/hadoop/etc/hadoop
capacity-scheduler.xml                        100% 8260   293.2KB/s   00:00
configuration.xsl                             100% 1335   791.3KB/s   00:00
container-executor.cfg                        100% 1940     1.1MB/s   00:00
core-site.xml                                 100%  864   776.1KB/s   00:00
hadoop-env.cmd                                100% 3999     1.9MB/s   00:00
hadoop-env.sh                                 100%   16KB   2.6MB/s   00:00
hadoop-metrics2.properties                    100% 3321     1.4MB/s   00:00
hadoop-policy.xml                             100%   11KB 293.3KB/s   00:00
hadoop-user-functions.sh.example              100% 3414     1.2MB/s   00:00
hdfs-site.xml                                 100% 1051   201.6KB/s   00:00
httpfs-env.sh                                 100% 1484    71.0KB/s   00:00
httpfs-log4j.properties                       100% 1657   975.4KB/s   00:00
httpfs-signature.secret                       100%   21    12.7KB/s   00:00
httpfs-site.xml                               100%  620   351.4KB/s   00:00
kms-acls.xml                                  100% 3518     1.8MB/s   00:00
kms-env.sh                                    100% 1351   850.0KB/s   00:00
kms-log4j.properties                          100% 1860   990.0KB/s   00:00
kms-site.xml                                  100%  682   429.3KB/s   00:00
log4j.properties                              100%   13KB  11.7MB/s   00:00
mapred-env.cmd                                100%  951   646.1KB/s   00:00
mapred-env.sh                                 100% 1764     1.2MB/s   00:00
mapred-queues.xml.template                    100% 4113     2.6MB/s   00:00
mapred-site.xml                               100%  758   817.1KB/s   00:00
/usr/local/hadoop/etc/hadoop/shellprofile.d: not a regular file
ssl-client.xml.example                        100% 2316    57.9KB/s   00:00
```

Figure 31

Listing 39: a

```
scp /usr/local/hadoop/etc/hadoop/* hadoop-slave2:/usr/local/hadoop/etc/hadoop/
```



```
hadoopuser@hadoop-master:~$ scp /usr/local/hadoop/etc/hadoop/* hadoop-slave2:/u
sr/local/hadoop/etc/hadoop/
capacity-scheduler.xml                        100% 8260     1.7MB/s   00:00
configuration.xsl                             100% 1335   128.3KB/s   00:00
container-executor.cfg                        100% 1940     1.9MB/s   00:00
core-site.xml                                 100%  864   737.1KB/s   00:00
hadoop-env.cmd                                100% 3999     3.2MB/s   00:00
hadoop-env.sh                                 100%   16KB   1.3MB/s   00:00
hadoop-metrics2.properties                    100% 3321     2.9MB/s   00:00
hadoop-policy.xml                             100%   11KB   1.4MB/s   00:00
hadoop-user-functions.sh.example              100% 3414     2.7MB/s   00:00
hdfs-site.xml                                 100% 1051   732.5KB/s   00:00
httpfs-env.sh                                 100% 1484     1.1MB/s   00:00
httpfs-log4j.properties                       100% 1657     1.5MB/s   00:00
httpfs-signature.secret                       100%   21    15.4KB/s   00:00
httpfs-site.xml                               100%  620   529.2KB/s   00:00
kms-acls.xml                                  100% 3518     2.7MB/s   00:00
kms-env.sh                                    100% 1351   987.0KB/s   00:00
kms-log4j.properties                          100% 1860     1.8MB/s   00:00
kms-site.xml                                  100%  682   302.6KB/s   00:00
log4j.properties                              100%   13KB  11.4MB/s   00:00
mapred-env.cmd                                100%  951   902.9KB/s   00:00
mapred-env.sh                                 100% 1764     1.6MB/s   00:00
mapred-queues.xml.template                    100% 4113     3.8MB/s   00:00
mapred-site.xml                               100%  758   799.8KB/s   00:00
/usr/local/hadoop/etc/hadoop/shellprofile.d: not a regular file
ssl-client.xml.example                        100% 2316   228.1KB/s   00:00
```

Figure 32

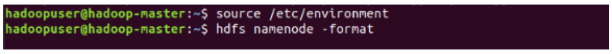
7 HDFS

7.1 format hdfs

Now we need to format the HDFS file system. Run these commands:

Listing 40: a

```
source /etc/environment
hdfs namenode -format
```



```
hadoopuser@hadoop-master:~$ source /etc/environment
hadoopuser@hadoop-master:~$ hdfs namenode -format
```

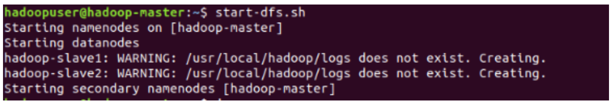
Figure 33

7.2 start hdfs

Start HDFS with this command:

Listing 41: a

```
start-dfs.sh
```



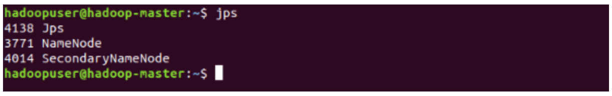
```
hadoopuser@hadoop-master:~$ start-dfs.sh
Starting namenodes on [hadoop-master]
Starting datanodes
hadoop-slave1: WARNING: /usr/local/hadoop/logs does not exist. Creating.
hadoop-slave2: WARNING: /usr/local/hadoop/logs does not exist. Creating.
Starting secondary namenodes [hadoop-master]
```

Figure 34

To check if this worked, run the follwing command. This will tell you what resources have been initialized:

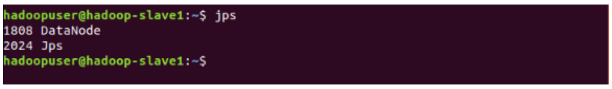
Listing 42: for all nodes

```
jps
```



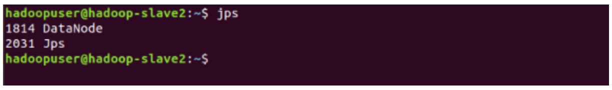
```
hadoopuser@hadoop-master:~$ jps
4138 Jps
3771 NameNode
4014 SecondaryNameNode
hadoopuser@hadoop-master:~$
```

Figure 35



```
hadoopuser@hadoop-slave1:~$ jps
1808 DataNode
2024 Jps
hadoopuser@hadoop-slave1:~$
```

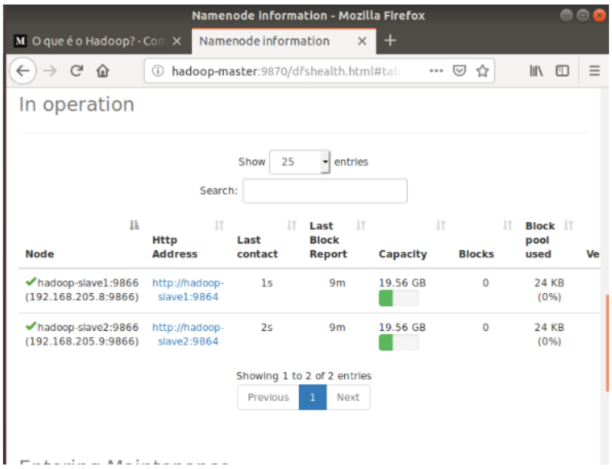
Figure 36



```
hadoopuser@hadoop-slave2:~$ jps
1814 DataNode
2031 Jps
hadoopuser@hadoop-slave2:~$
```

Figure 37

Let's see if this worked: Open your browser and type **hadoop-master:9870** This is what mine shows, hopefully yours is showing the same thing!



Namenode information - Mozilla Firefox
O que é o Hadoop? - Com
Namenode information
hadoop-master:9870/dfshealth.html#tab

In operation

Show 25 entries

Search:

Node	Http Address	Last contact	Last Block Report	Capacity	Blocks	Block pool used	Ve
✓hadoop-slave1:9866 (192.168.205.8:9866)	http://hadoop-slave1:9864	1s	9m	19.56 GB	0	24 KB (0%)	
✓hadoop-slave2:9866 (192.168.205.9:9866)	http://hadoop-slave2:9864	2s	9m	19.56 GB	0	24 KB (0%)	

Showing 1 to 2 of 2 entries

Entering Maintenance

Figure 38

As you can see, both nodes are operational!

8 Resources

https://medium.com/@jootorres_11979/how-to-set-up-a-hadoop-3-2-1-multi-node-cluster-on-ubuntu-18
https://www.digitalocean.com/community/tutorials/how-to-install-hadoop-in-stand-alone-mode-on
https://archive.apache.org/dist/hadoop/core/hadoop-3.0.3/
https://data-flair.training/blogs/install-hadoop-on-ubuntu/