

# BÀI TẬP TRÊN LỚP

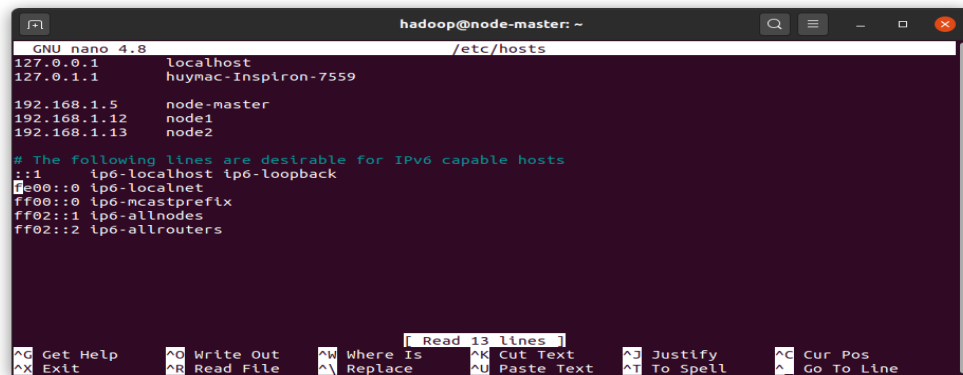
## MÔN HỌC: LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU LỚN

### LAB 1: HDFS

NHÓM: GIỮA CHÚNG TA

#### 1. Kết quả cài đặt

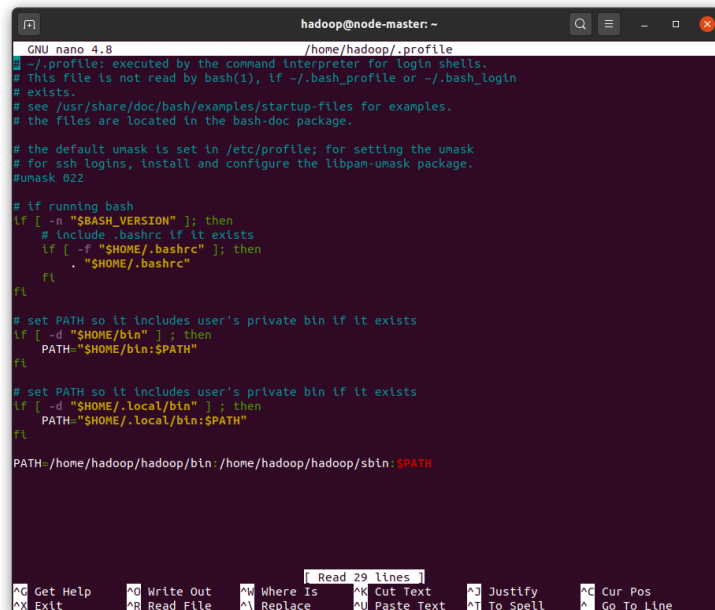
##### 1.1. Configure the System



```
hadoop@node-master: ~  
GNU nano 4.8 /etc/hosts  
127.0.0.1 localhost  
127.0.1.1 huymac-Inspiron-7559  
  
192.168.1.5 node-master  
192.168.1.12 node1  
192.168.1.13 node2  
  
# 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  
  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ^R Read File ^M Replace ^U Paste Text ^T To Spell ^_ Go To Line
```

Figure 1: Hosts file

##### 1.2. Set Environment Variables



```
hadoop@node-master: ~  
GNU nano 4.8 /home/hadoop/.profile  
# ~/.profile: executed by the command interpreter for login shells.  
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login  
# exists.  
# see /usr/share/doc/bash/examples/startup-files for examples.  
# the files are located in the bash-doc package.  
  
# the default umask is set in /etc/profile; for setting the umask  
# for ssh logins, install and configure the libpan-umask package.  
umask 022  
  
# if running bash  
if [ -n "$BASH_VERSION" ]; then  
    # include .bashrc if it exists  
    if [ -f "$HOME/.bashrc" ]; then  
        . "$HOME/.bashrc"  
    fi  
fi  
  
# set PATH so it includes user's private bin if it exists  
if [ -d "$HOME/bin" ] ; then  
    PATH="$HOME/bin:$PATH"  
fi  
  
# set PATH so it includes user's private bin if it exists  
if [ -d "$HOME/.local/bin" ] ; then  
    PATH="$HOME/.local/bin:$PATH"  
fi  
  
PATH=/home/hadoop/hadoop/bin:/home/hadoop/hadoop/sbin:$PATH
```

Figure 2: Add Hadoop binaries to PATH

```

GNU nano 4.8 /home/hadoop/.bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64

export HADOOP_HOME=/home/hadoop/hadoop
export PATH=${PATH}:${HADOOP_HOME}/bin:${HADOOP_HOME}/sbin
# If not running interactively, don't do anything
case $- in
  *i*) ;;
  *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

# If set, the pattern "**" used in a pathname expansion context will
# match all files and zero or more directories and subdirectories.
#shopt -s globstar

# make less more friendly for non-text input files, see lesspipe(1)
[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^I To Spell     ^_ Go To Line

```

Figure 3: Add Hadoop to your PATH for the shell

```

GNU nano 4.8 /home/hadoop/hadoop/etc/hadoop/core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  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. -->

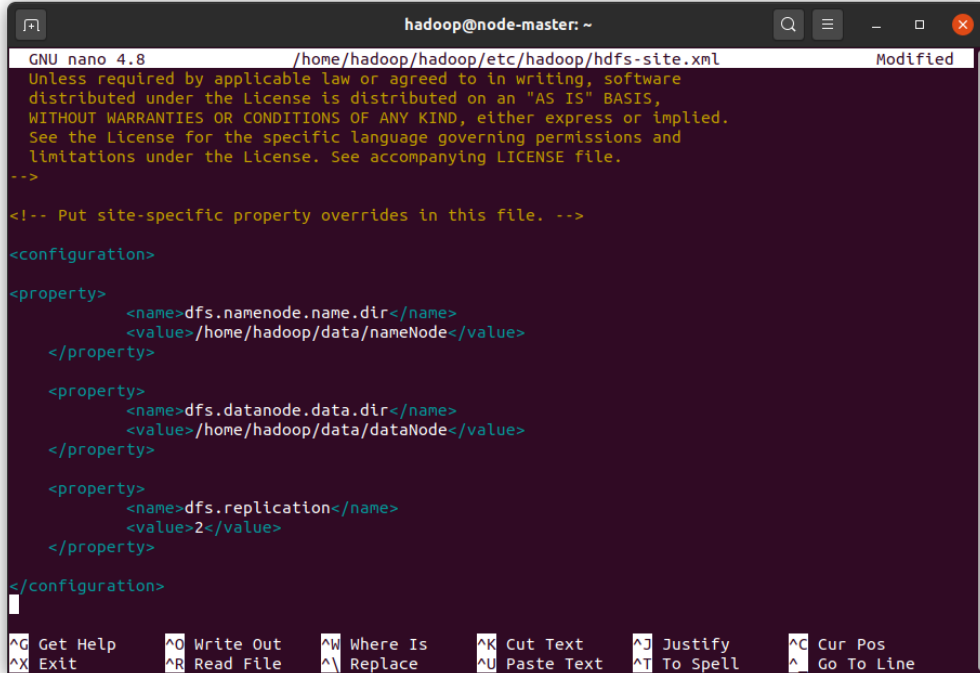
<configuration>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://node-master:9000</value>
  </property>
</configuration>

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^I To Spell     ^_ Go To Line

```

Figure 4: Set NameNode Location

- `Dfs.replication = 2`



The screenshot shows a terminal window with the title 'hadoop@node-master: ~'. The nano editor is open to the file '/home/hadoop/hadoop/etc/hadoop/hdfs-site.xml'. The file content includes a license notice, a comment to put site-specific property overrides in this file, and an XML configuration block. The configuration block contains three property elements: 'dfs.namenode.name.dir' with value '/home/hadoop/data/nameNode', 'dfs.datanode.data.dir' with value '/home/hadoop/data/dataNode', and 'dfs.replication' with value '2'. The nano editor's status bar at the bottom shows various keyboard shortcuts like '^G Get Help', '^O Write Out', '^W Where Is', '^K Cut Text', '^J Justify', '^C Cur Pos', '^X Exit', '^R Read File', '^M Replace', '^U Paste Text', '^T To Spell', and '^\_ Go To Line'.

```
GNU nano 4.8 /home/hadoop/hadoop/etc/hadoop/hdfs-site.xml Modified
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
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>/home/hadoop/data/nameNode</value>
</property>

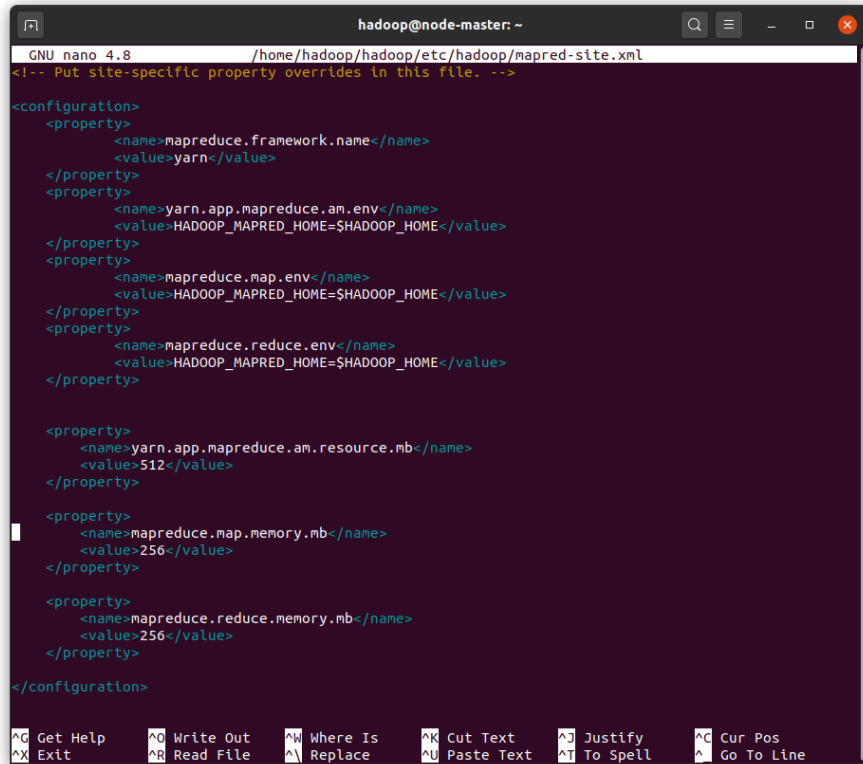
<property>
  <name>dfs.datanode.data.dir</name>
  <value>/home/hadoop/data/dataNode</value>
</property>

<property>
  <name>dfs.replication</name>
  <value>2</value>
</property>

</configuration>

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^M Replace   ^U Paste Text ^T To Spell   ^_ Go To Line
```

Figure 5: Set path for HDFSPermalink



The screenshot shows a terminal window with the title 'hadoop@node-master: ~'. The nano editor is open to the file '/home/hadoop/hadoop/etc/hadoop/mapred-site.xml'. The file content includes a comment to put site-specific property overrides in this file, and an XML configuration block. The configuration block contains several property elements: 'mapreduce.framework.name' with value 'yarn', 'yarn.app.mapreduce.am.env' with value 'HADOOP\_MAPRED\_HOME=\$HADOOP\_HOME', 'mapreduce.map.env' with value 'HADOOP\_MAPRED\_HOME=\$HADOOP\_HOME', 'mapreduce.reduce.env' with value 'HADOOP\_MAPRED\_HOME=\$HADOOP\_HOME', 'yarn.app.mapreduce.am.resource.mb' with value '512', 'mapreduce.map.memory.mb' with value '256', and 'mapreduce.reduce.memory.mb' with value '256'. The nano editor's status bar at the bottom shows various keyboard shortcuts like '^G Get Help', '^O Write Out', '^W Where Is', '^K Cut Text', '^J Justify', '^C Cur Pos', '^X Exit', '^R Read File', '^M Replace', '^U Paste Text', '^T To Spell', and '^\_ Go To Line'.

```
GNU nano 4.8 /home/hadoop/hadoop/etc/hadoop/mapred-site.xml
<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>yarn.app.mapreduce.am.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME</value>
  </property>

  <property>
    <name>yarn.app.mapreduce.am.resource.mb</name>
    <value>512</value>
  </property>

  <property>
    <name>mapreduce.map.memory.mb</name>
    <value>256</value>
  </property>

  <property>
    <name>mapreduce.reduce.memory.mb</name>
    <value>256</value>
  </property>

</configuration>

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^M Replace   ^U Paste Text ^T To Spell   ^_ Go To Line
```

Figure 6: Set YARN as Job Scheduler

```

hadoop@node-master: ~
GNU nano 4.8 /home/hadoop/hadoop/etc/hadoop/yarn-site.xml
-->
<configuration>
  <property>
    <name>yarn.acl.enable</name>
    <value>0</value>
  </property>

  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>113.190.16.206</value>
  </property>

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

  <property>
    <name>yarn.nodemanager.resource.memory-mb</name>
    <value>1536</value>
  </property>

  <property>
    <name>yarn.scheduler.maximum-allocation-mb</name>
    <value>1536</value>
  </property>

  <property>
    <name>yarn.scheduler.minimum-allocation-mb</name>
    <value>128</value>
  </property>

  <property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
  </property>
</configuration>
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^M Replace   ^U Paste Text ^T To Spell   ^_ Go To Line

```

Figure 7: Configure YARN

### 1.3. Configure Memory Allocation

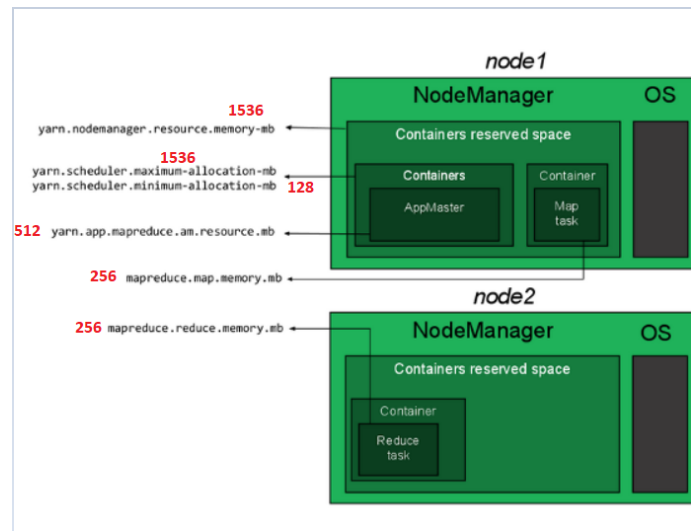


Figure 8: Sample config for 2GB Nodes (will change in next lab)

### 1.4. Duplicate Config Files On Each Node

- First, install ssh
- Generate an SSH key
- Make a new file master.pub in the /home/hadoop/.ssh. Paste your public key into
- Copy key:

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

- **Note:**

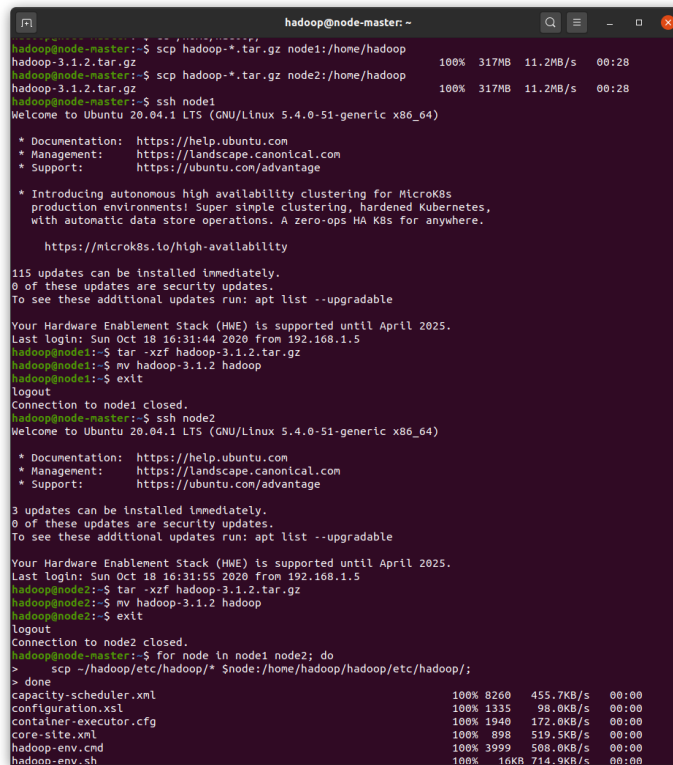
- We use this command for ssh node1, node2 without req passwd

```
Ssh-copy-id -I ~/.ssh/id_rsa.pub hadoop@node-master
```

```
Ssh-copy-id -I ~/.ssh/id_rsa.pub hadoop@node1
```

```
Ssh-copy-id -I ~/.ssh/id_rsa.pub hadoop@node2
```

- Importance: Config username of node-master, node1, node2 is hadoop



```
hadoop@node-master:~$ scp hadoop-*.tar.gz node1:/home/hadoop
hadoop@node-master:~$ scp hadoop-*.tar.gz node2:/home/hadoop
hadoop@node-master:~$ ssh node1
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-51-generic x86_64)

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

 * Introducing autonomous high availability clustering for MicroK8s
   production environments! Super simple clustering, hardened Kubernetes,
   with automatic data store operations. A zero-ops HA K8s for anywhere.
   https://microk8s.io/high-availability

115 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Sun Oct 18 16:31:44 2020 from 192.168.1.5
hadoop@node1:~$ tar -xzf hadoop-3.1.2.tar.gz
hadoop@node1:~$ mv hadoop-3.1.2 hadoop
hadoop@node1:~$ exit
logout
Connection to node1 closed.
hadoop@node-master:~$ ssh node2
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-51-generic x86_64)

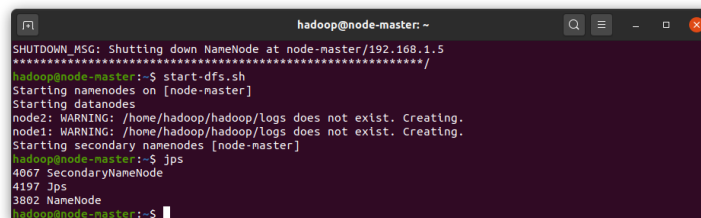
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

3 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Sun Oct 18 16:31:55 2020 from 192.168.1.5
hadoop@node2:~$ tar -xzf hadoop-3.1.2.tar.gz
hadoop@node2:~$ mv hadoop-3.1.2 hadoop
hadoop@node2:~$ exit
logout
Connection to node2 closed.
hadoop@node-master:~$ for node in node1 node2; do
>   scp ~/.ssh/master.pub $node:/home/hadoop/hadoop/etc/hadoop/;
> done
capacity-scheduler.xml      100% 8260   455.7KB/s   00:00
configuration.xml          100% 1335    98.0KB/s   00:00
container-executor.cfg     100% 1940   172.0KB/s   00:00
core-site.xml              100% 890    519.5KB/s   00:00
hadoop-env.cmd             100% 3999   508.0KB/s   00:00
hadoop-env.sh              100% 16KB   714.9KB/s   00:00
```

Figure 9: Copy files to node1, node2

## 1.5. Start HDFS



```
hadoop@node-master:~$ start-dfs.sh
SHUTDOWN_MSG: Shutting down NameNode at node-master/192.168.1.5
*****
hadoop@node-master:~$ start-dfs.sh
Starting namenodes on [node-master]
Starting datanodes
node2: WARNING: /home/hadoop/hadoop/logs does not exist. Creating.
node1: WARNING: /home/hadoop/hadoop/logs does not exist. Creating.
Starting secondary namenodes [node-master]
hadoop@node-master:~$ jps
4067 SecondaryNameNode
4197 Jps
3802 NameNode
hadoop@node-master:~$
```

Figure 10: Start the HDFS by running the following script from node-master

- Or can: \$ start-all.sh

## 2. Monitor HDFS Cluter

### 2.1. Put Data to HDFS

```
hdfs dfs -mkdir books
```

- use wget to download file, see here ☺: <https://speed.hetzner.de/> for what ever file size you want

```
hdfs dfs -put 500MB.bin books
hdfs dfs -put 500MB.bin books
```

### 2.2. View web user interface

- <http://192.168.1.5:9870> (192.168.1.5 is my master node id)

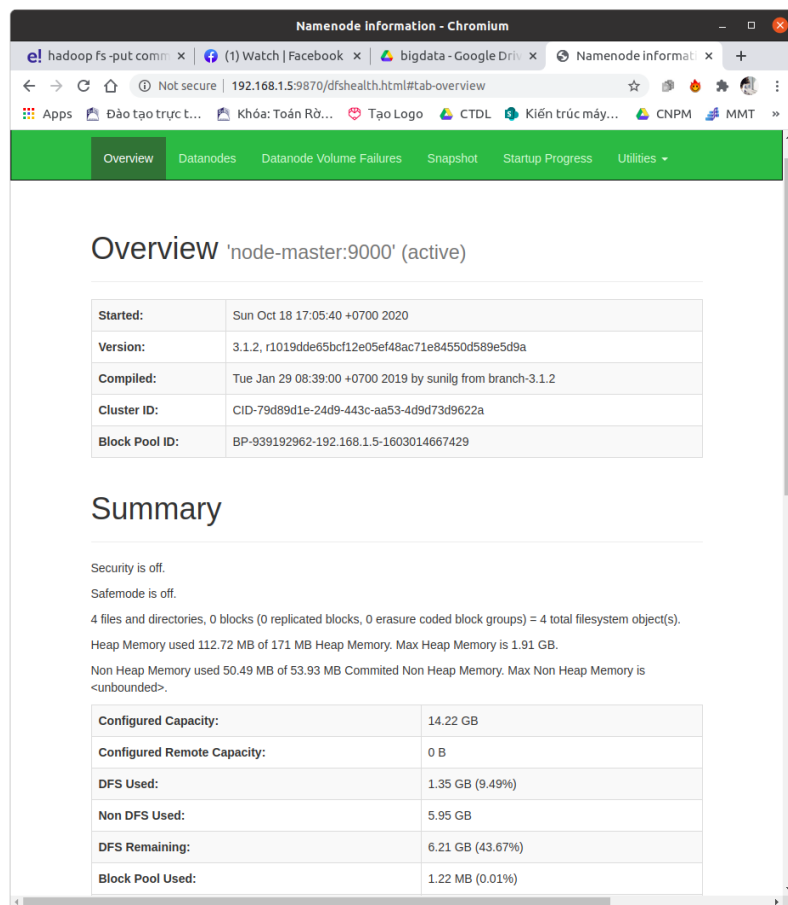


Figure 11: Web user interface

どうもありがとう、先生