Setup Hadoop 2.9 Multi-node Cluster on Ubuntu 16.04 Virtual Machine

Hadoop multi-node cluster has 5 machines. One of them is the master and the other four are the slaves.

* Master 192.168.40.129
* Slave1 192.168.40.130
* Slave2 192.168.40.131
* Slave3 192.168.40.132
* Slave4 192.168.40.133

Requirements:

|  |  |
| --- | --- |
| Master | Slaves |
| 2 GB RAM | 2 GB RAM |
| Hard Disk 50 GB | Hard Disk 50 GB |
| Network Adapter Bridge | Network Adapter Bridge |
| Static IP address | Static IP address |
| Java 8 | Java 8 |
| Hadoop 2.9 | Hadoop 2.9 |
| Spark 2.3.0 | Spark 2.3.0 |

**1. Setup in all machines (Master and Slaves)**

**1.1 Create virtual machines**

Download and install VMware® Workstation 15 Player (available at <https://www.vmware.com/products/workstation-pro/workstation-pro-evaluation.html>).

**1.2. Install Operating System**

Download and install Ubuntu 16.0.4 (available at <http://releases.ubuntu.com/16.04/>). The virtual machine will be creating with the following settings:

* Hard Disk: 50 GB
* Memory: 2 GB

**1.3. Update and Install on Ubuntu**

Launch virtual machine and login. Open a terminal (CTRL + ALT + T) and type the following sudo command to install the latest updates.

$ sudo apt-get update

$ sudo apt-get install ssh

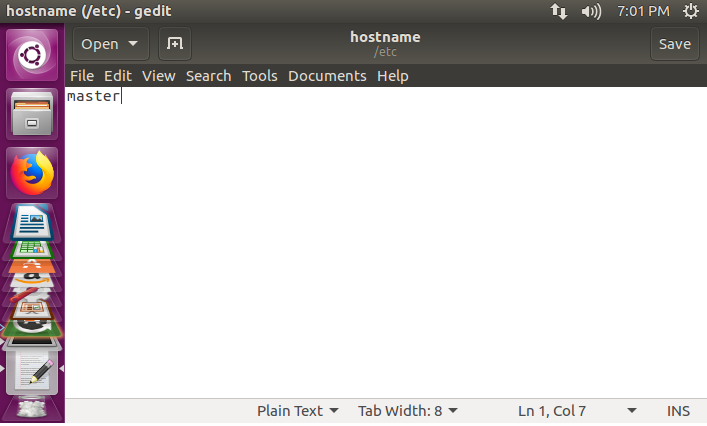
**1.4. Hostname Identification**

Change the name of the hostname for the master and all slaves.

* Type the following sudo command to change the name of the hostname for the master.

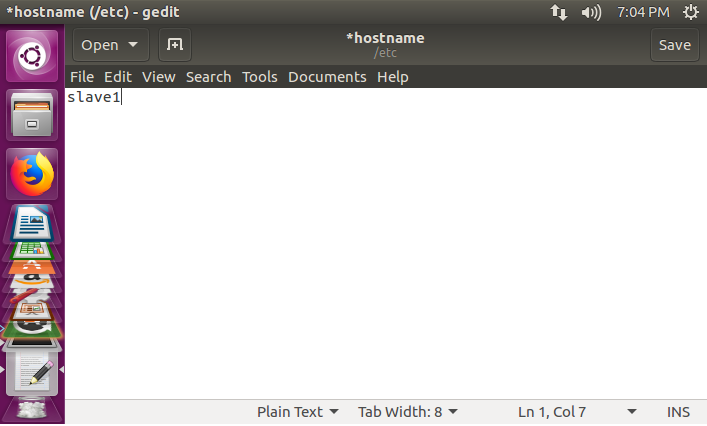
$ sudo gedit /etc/hostname

Type master and save for the Master node.



* Type the following sudo command to change the name of the hostname for all the slaves.

$ sudo gedit /etc/hostname



* Reboot all the machines.

$ sudo reboot

**1.5. Creating User Account**

Create a user account for Hadoop. This user account must be the same for the master and slaves.

* Create group

$ sudo addgroup hadoop\_group

* Create user for the group

$ sudo adduser -–ingroup hadoop\_group hadoop\_user

* Add hadoop\_user to group sudo

$ sudo adduser hadoop\_user sudo

* Change user

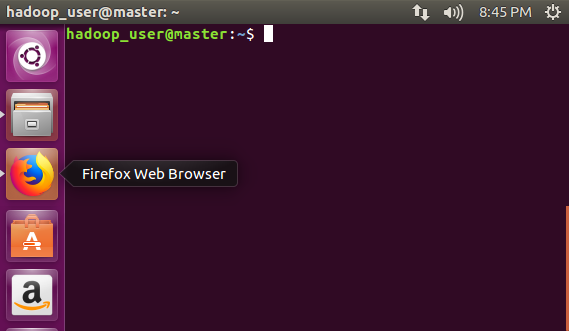
$ sudo hadoop\_user

hadoop\_user@master:~$

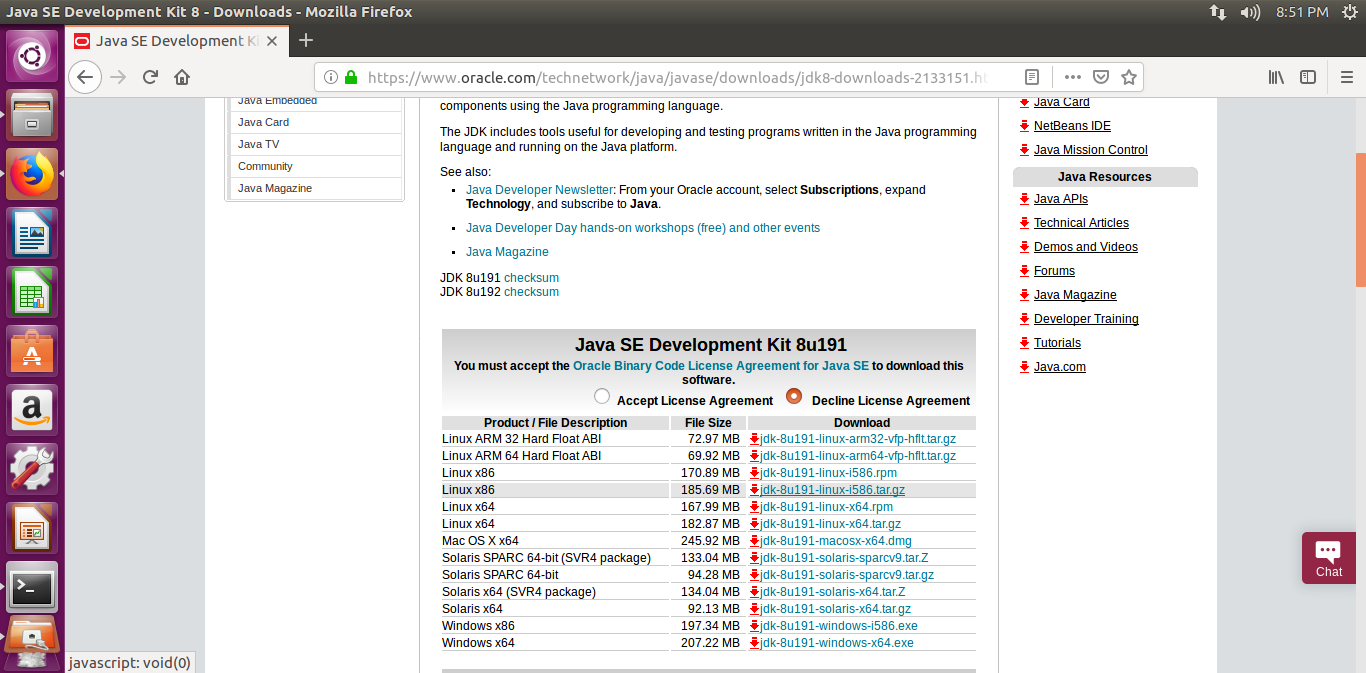
**2. Setup the Master**

**2.1. Download Java, Hadoop, and Spark**

* Go to the Firefox Web Browser

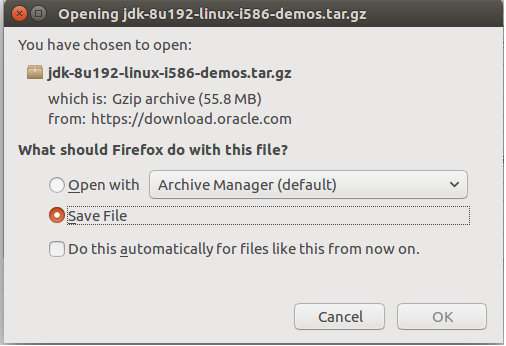


* Download Java JDK 8

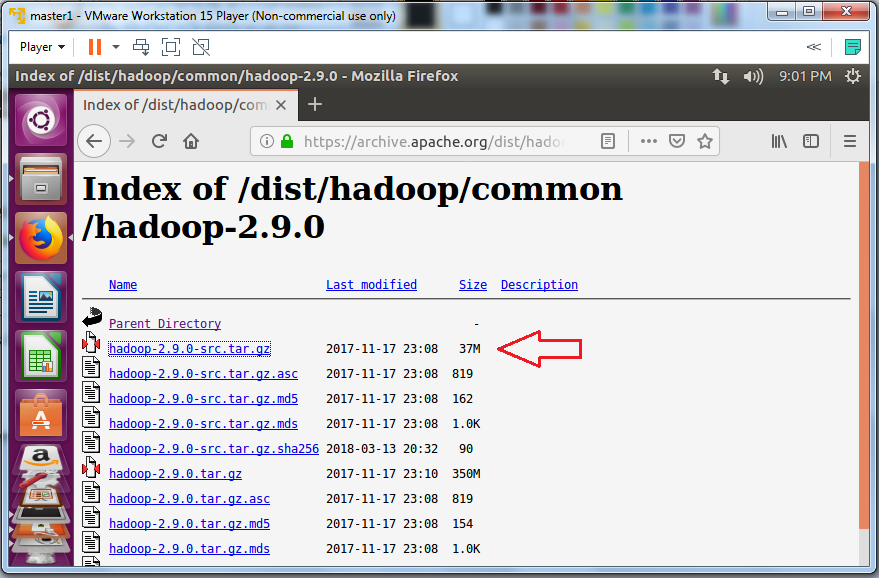


* Save File

The file is going to be saved in /home/hadoop\_user/Downloads

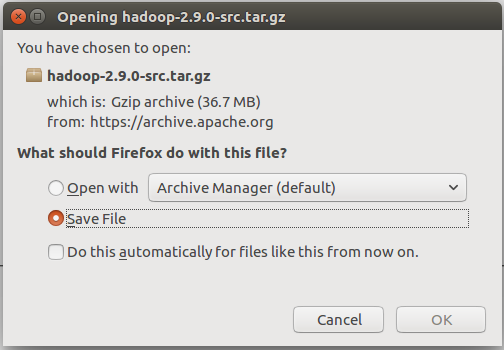


* Download Hadoop

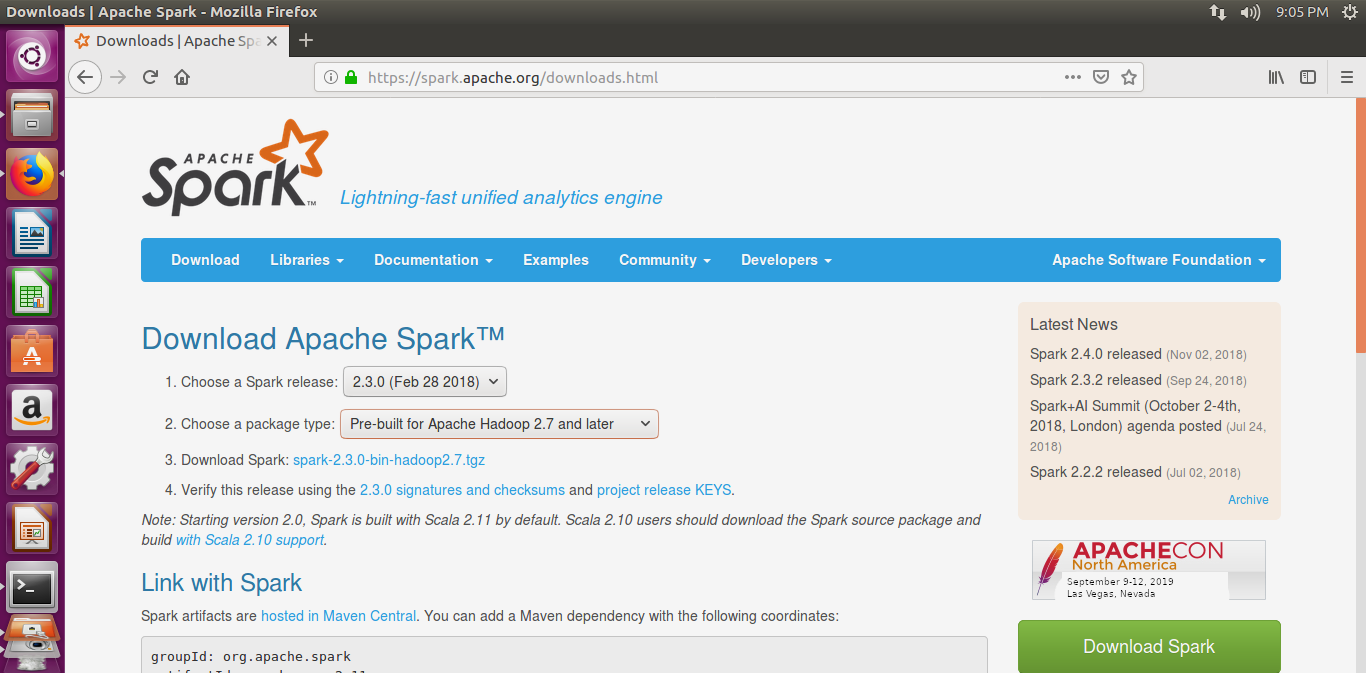


* Save File

The file is going to be saved in /home/hadoop\_user/Downloads

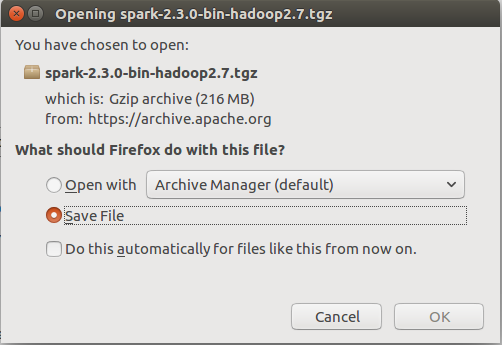


* Download Spark



* Save File

The file is going to be saved in /home/hadoop\_user/Downloads



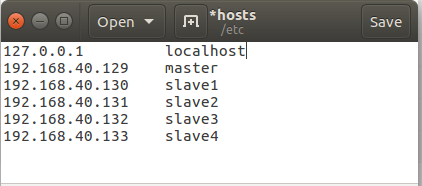
**2.2. SSH**

**SSH** ("Secure Shell") is a protocol for securely accessing one computer from another. All the machines must communicate with one another without any prompt for password. Setup ssh in all the machines master and slaves so they can communicate.

* Update /etc/host and add the IP address and hostname of each machine.

$ hadoop\_user@master:~$ sudo gedit /etc/hosts

* Save and close



* Ping from one machine to another machine to be sure that all the machines can communicate. This is an important step.

hadoop\_user@master:~$ ping 192.168.40.130

* Create the key that will replace the use of the password

hadoop\_user@master:~$ ssh-keygen -t rsa =P “”

* Copy the key to the authorized\_key

hadoop\_user@master:~$ cat ~/.ssh/i\_rsa.pub >> ~/.ssh/authorized\_keys

* Reboot the master node to be sure all changes are take.

hadoop\_user@master:~$ reboot

* Setup ssh in every node such that they can communicate with one another without any prompt for password.

hadoop\_user@master:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave1

hadoop\_user@master:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave2

hadoop\_user@master:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave3

hadoop\_user@master:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave4

**2.3 Installing Java**

* Create directory /usr/lib/java/java/ jdk1.8.0

hadoop\_user@master:~$ cd sudo mkdir -p /usr/lib/java/jdk1.8.0

* Go to /home/hadoop\_user/Downloads

hadoop\_user@master:~$ cd /home/Hadoop\_user/Downloads

* Unpack jdk-8u171-linux-x64.tar.gz

hadoop\_user@master:~/Downloads$ tar -xvzf jdk-8u171-linux-x64.tar.gz

* Move jdk1.8.0\_171 to /home/hadoop\_user/Downloads

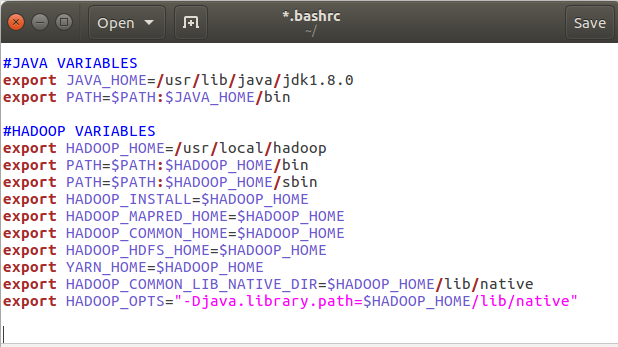
hadoop\_user@master:~/Downloads$ move /home/hadoop\_user/Downloads/ jdk1.8.0\_171/\* /usr/lib/java/jdk1.8.0

**2.4 Setup Environment Variables**

* Update file ~/.bashrc and add the following lines at the end of the file.

hadoop\_user@master:~$sudo gedit ~/.bashrc

* After update the file, save and close the file.



* Source the environment variables.

hadoop\_user@master:~$ source ~/.bashrc

**2.5. Installing Hadoop**

* Create directory to copy hadoop files and give access to hadoop\_user

hadoop\_user@master:~$sudo mkdir -p /usr/local/hadoop

hadoop\_user@master:~$sudo chown -R hadoop\_group:hadoop\_user /usr/local/hadoop

* Create directory /usr/local/hadoop\_store/hdfs/datanode file to store all datanode information and give access to hadoop\_user

hadoop\_user@master:~$sudo mkdir -p /usr/local/hadoop\_store/hdfs/namenode

hadoop\_user@master:~$sudo chown -R hadoop\_group:hadoop\_user /usr/local/hadoop\_store/hdfs/namenode

* Extract the contest from the hadoop package

hadoop\_user@master:~$sudo /home/hadoop\_user/Downloads

hadoop@user@master:~/Downloads$ sudo tar -xvzf hadoop-2.9.0.tar.gz

* Move all the files from hadoop-2.9.0 directory to /usr/local/hadoop

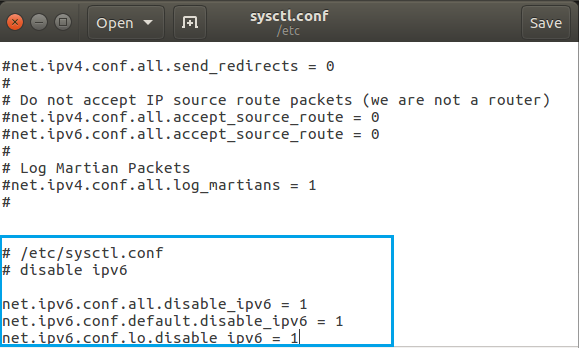
hadoop\_user@master:~$sudo move /home/Hadoop\_user/Downloads/ hadoop-2.9.0/\* /usr/local/hadoop

* Disable IPv6.

Hadoop does not support IPv6. If you are using IPv6, you need to switch Hadoop host machines to use IPv4.

Update file /etc/sysctl.conf and add the following lines at the end of the file.

hadoop\_user@master:~$sudo gedit /etc/sysctl.conf



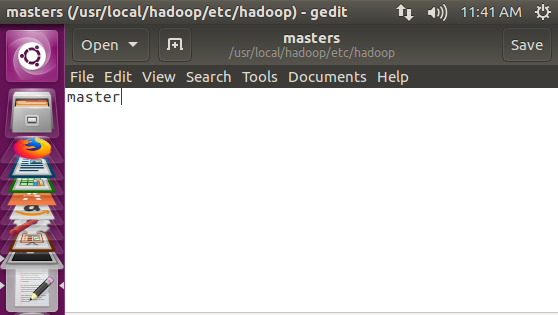
* Update masters file

hadoop\_user@master:~cd $HADOOP\_HOME/etc/hadoop

hadoop\_user@master:/usr/local/hadoop/etc/hadoop

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit masters

* Add master to the file. Save and close.

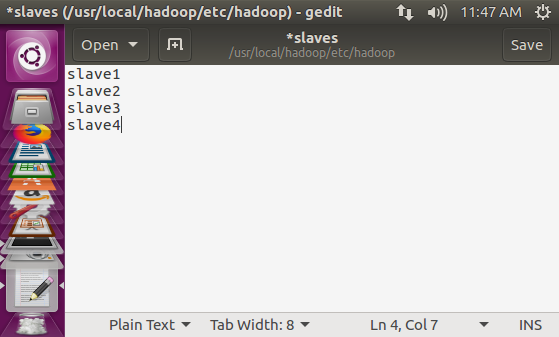


Save and close

* Update slaves file

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit slave

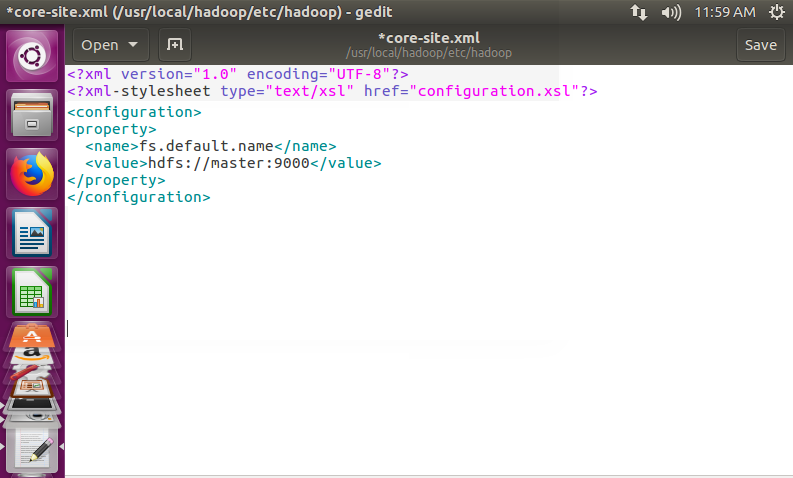
* Add all the slaves: slave1, slave2, slave3 and slave4 to the file. Save and close.



* Update core-site.xml file

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit core-sire.xml

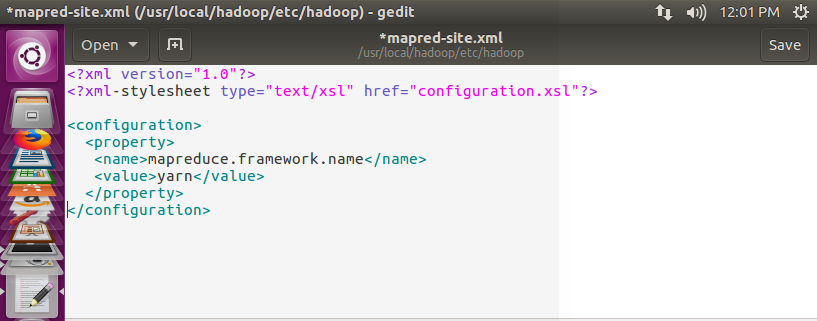
* Add the following lines:



* Update mapred-site.xml file

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit mapred-site.xml

* Add the following lines:



* Update hdfs-site.xml file

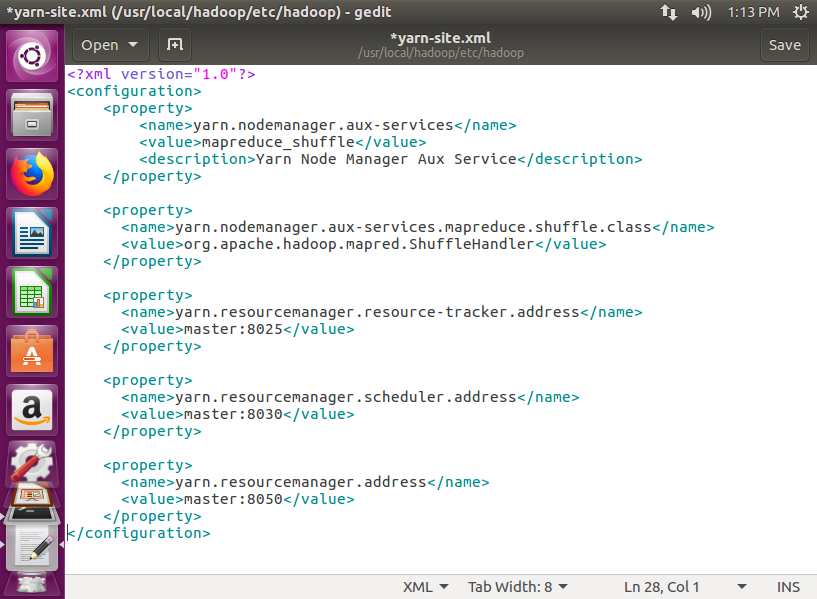
hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit hdfs-site.xml

* Add the following lines:

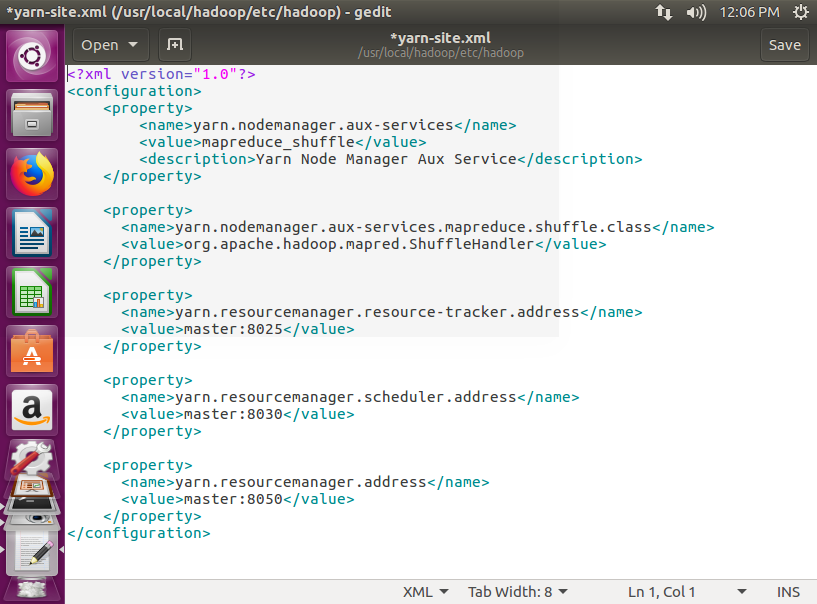


* Update yarn-site.xml file

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit yarn-site.xml

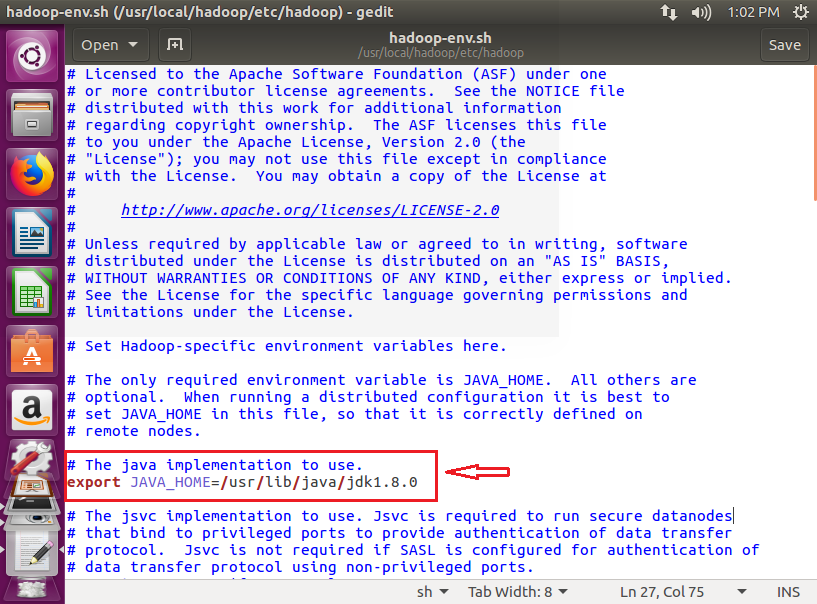


* Add all the following lines:



* Update hadoop-env.sh file

hadoop\_user@master:/usr/local/hadoop/etc/hadoop/sudo gedit hadoop-env.sh



**3. Setup the Slaves**

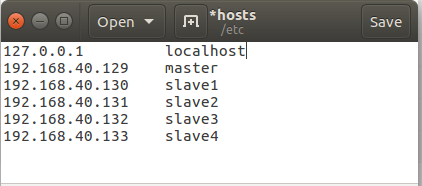
**3.1. SSH**

**SSH** ("Secure Shell") is a protocol for securely accessing one computer from another. All the machines must communicate with one another without any prompt for password. Setup ssh in all the machines master and slaves so they can communicate.

* Update /etc/host and add the IP address and hostname of each machine.

$ hadoop\_user@slave1:~$ sudo gedit /etc/hosts

* Save and close



* Ping from one machine to another machine to be sure that all the machines can communicate. This is an important step.

hadoop\_user@slave1:~$ ping 192.168.40.130

* Create the key that will replace the use of the password

hadoop\_user@slave1:~$ ssh-keygen -t rsa =P “”

* Copy the key to the authorized\_key

hadoop\_user@slave1:~$ cat ~/.ssh/i\_rsa.pub >> ~/.ssh/authorized\_keys

* Reboot the master node to be sure all changes are take.

hadoop\_user@slave1:~$ reboot

* Repeat this step in each node: slave1, slave2, slave3, slave4

hadoop\_user@slave1:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@master

hadoop\_user@slave1:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave2

hadoop\_user@slave1:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave3

hadoop\_user@slave1:~$ ssh-copy-id -i ~/.ssh/id\_rsa.pub hadoop\_user@slave4

**3.2 Copy Java, Hadoop, Spark from the Master node to the all the slaves**

* Login into the master node and copy the following files to all the slaves

hadoop\_user@master:~$ scp -r /home/hadoop\_user/Downloads/hadoop-2.9.0.tar.gz hadoop\_user@master:/home/hadoop\_user/Downloads/

hadoop\_user@master:~$ scp -r /home/hadoop\_user/Downloads/ jdk-8u171-linux-x64.tar.gz hadoop\_user@slave1:/home/hadoop\_user/Downloads/

hadoop\_user@master:~$ scp -r /home/hadoop\_user/Downloads/ spark-2.3.0-bin-hadoop2.7.gz gz hadoop\_user@slave1:/home/hadoop\_user/Downloads/

* Login into slave1

hadoop\_user@slave1:~$

* Create directory /usr/lib/java/java/ jdk1.8.0

hadoop\_user@slave1:~$ cd sudo mkdir -p /usr/lib/java/jdk1.8.0

* Go to /home/hadoop\_user/Downloads

hadoop\_user@slave1:~$ cd /home/Hadoop\_user/Downloads

* Unpack jdk-8u171-linux-x64.tar.gz

hadoop\_user@slave1:~/Downloads$ tar -xvzf jdk-8u171-linux-x64.tar.gz

* Move jdk1.8.0\_171 to /usr/lib/java/jdk1.8.0

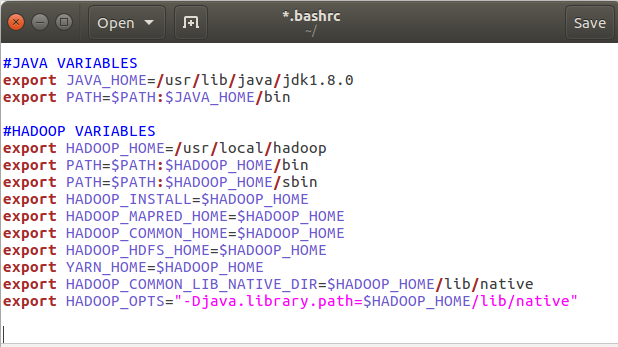
hadoop\_user@slave1:~/Downloads$ move /home/hadoop\_user/Downloads/ jdk1.8.0\_171/\* /usr/lib/java/jdk1.8.0

**3.3 Setup Environment Variables**

* Update file ~/.bashrc and add the following lines at the end of the file.

hadoop\_user@slave1:~$sudo gedit ~/.bashrc

* After update the file, save and close the file.



* Source the environment variables.

hadoop\_user@slave1:~$ source ~/.bashrc

**3.4 Installing Hadoop**

* Create directory to copy hadoop files and give access to hadoop\_user

hadoop\_user@slave1:~$sudo mkdir -p /usr/local/hadoop

hadoop\_user@slave1:~$sudo chown -R hadoop\_group:hadoop\_user /usr/local/hadoop

* Create directory /usr/local/hadoop\_store/hdfs/datanode file to store all datanode information and give access to hadoop\_user

hadoop\_user@slave1:~$sudo mkdir -p /usr/local/hadoop\_store/hdfs/datanode

hadoop\_user@slave1:~$sudo chown -R hadoop\_group:hadoop\_user /usr/local/hadoop\_store/hdfs/datanode

* Extract the contest from the hadoop package

hadoop\_user@slave1:~$sudo /home/hadoop\_user/Downloads

hadoop@user@slave1:~/Downloads$ sudo tar -xvzf hadoop-2.9.0.tar.gz

* Move all the files from hadoop-2.9.0 directory to /usr/local/hadoop

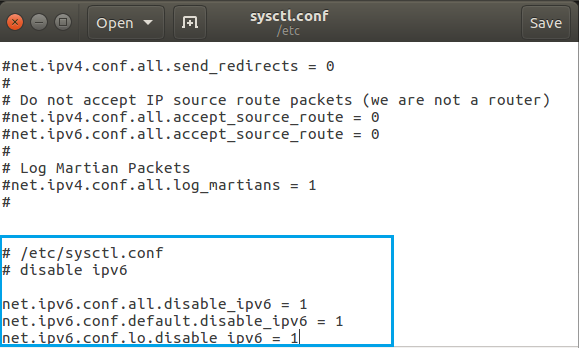
hadoop\_user@slave1:~$sudo move /home/Hadoop\_user/Downloads/ hadoop-2.9.0/\* /usr/local/hadoop

* Disable IPv6.

Hadoop does not support IPv6. If you are using IPv6, you need to switch Hadoop host machines to use IPv4.

Update file /etc/sysctl.conf and add the following lines at the end of the file.

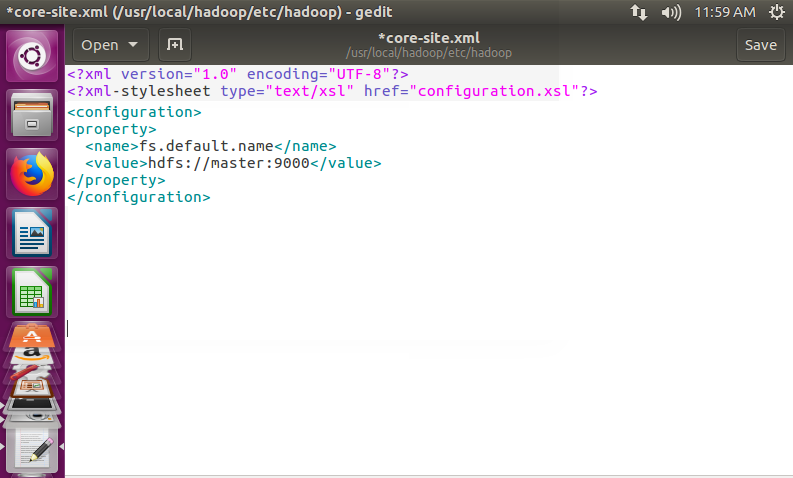
hadoop\_user@slave1:~$sudo gedit /etc/sysctl.conf



* Update core-site.xml file

hadoop\_user@slave1:/usr/local/hadoop/etc/hadoop/sudo gedit core-sire.xml

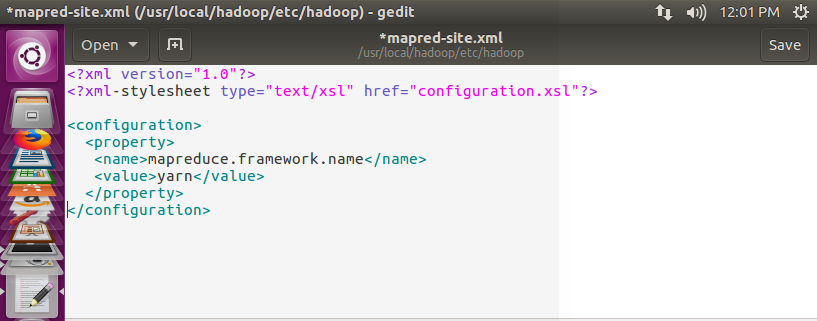
* Add the following lines:



* Update mapred-site.xml file

hadoop\_user@slave1:/usr/local/hadoop/etc/hadoop/sudo gedit mapred-site.xml

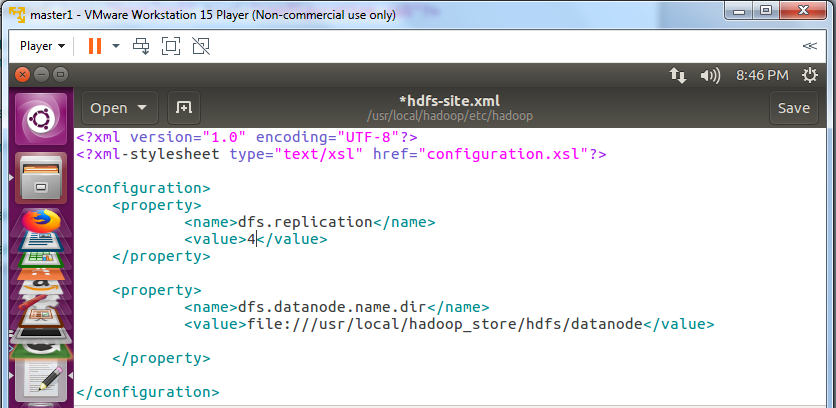
* Add the following lines:



* Update hdfs-site.xml file

hadoop\_user@slave1:/usr/local/hadoop/etc/hadoop/sudo gedit hdfs-site.xml

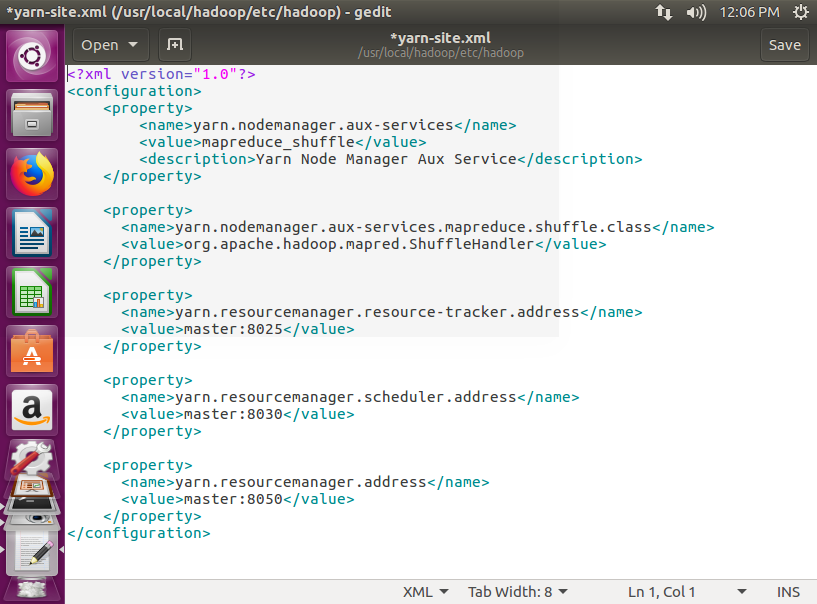
* Add the following lines:



* Update yarn-site.xml file

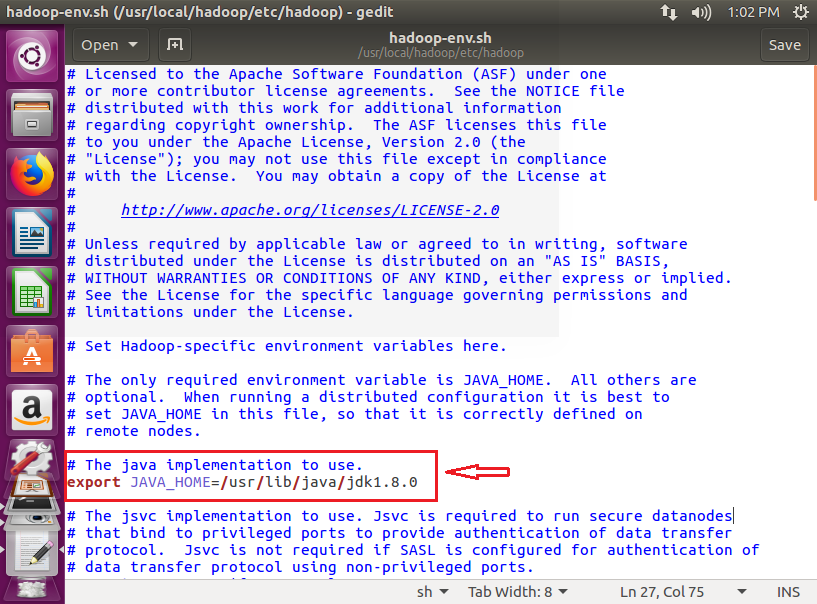
hadoop\_user@slave1:/usr/local/hadoop/etc/hadoop/sudo gedit yarn-site.xml

* Add all the following lines:



* Update hadoop-env.sh file

hadoop\_user@slave1:/usr/local/hadoop/etc/hadoop/sudo gedit hadoop-env.sh



**4. Format HDFS**

* Format HDFS

hadoop\_user@master:~$hdfs namenode -format

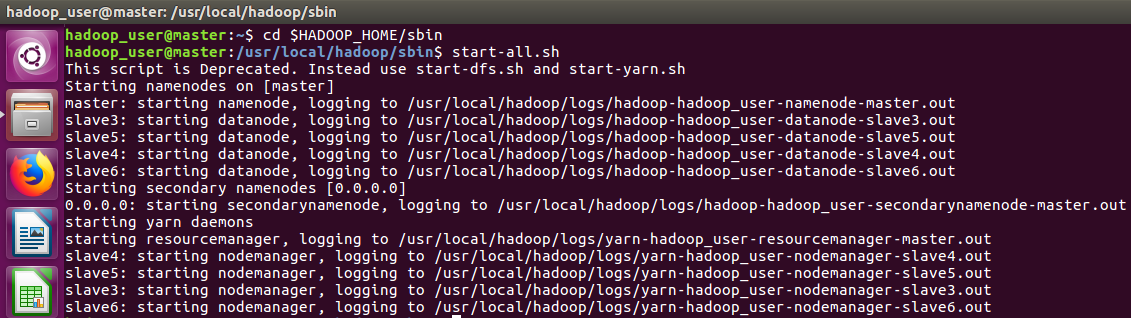
**5. Start and Stop HDFS**

* Start Namenode daemon, DataNode daemon and Yarn daemon

hadoop\_user@master:~$cd $HADOOP\_HOME/sbin

[hadoop\_user@master:/usr/local/hadoop/sbin/start-dfs.sh](mailto:hadoop_user@master:/usr/local/hadoop/sbin/start-dfs.sh)

hadoop\_user@master:/usr/local/hadoop/sbin/start-yarn.sh



* Check if every process is running with the jps command on each node

hadoop\_user@master:~$cd $HADOOP\_HOME/sbin$jps

* jps Master

3185 SecondaryNameNode

3656 Jps

2991 NameNode

3375 ResourceManager

* jps Slaves

5497 Jps

5196 DataNode

5310 NodeManager

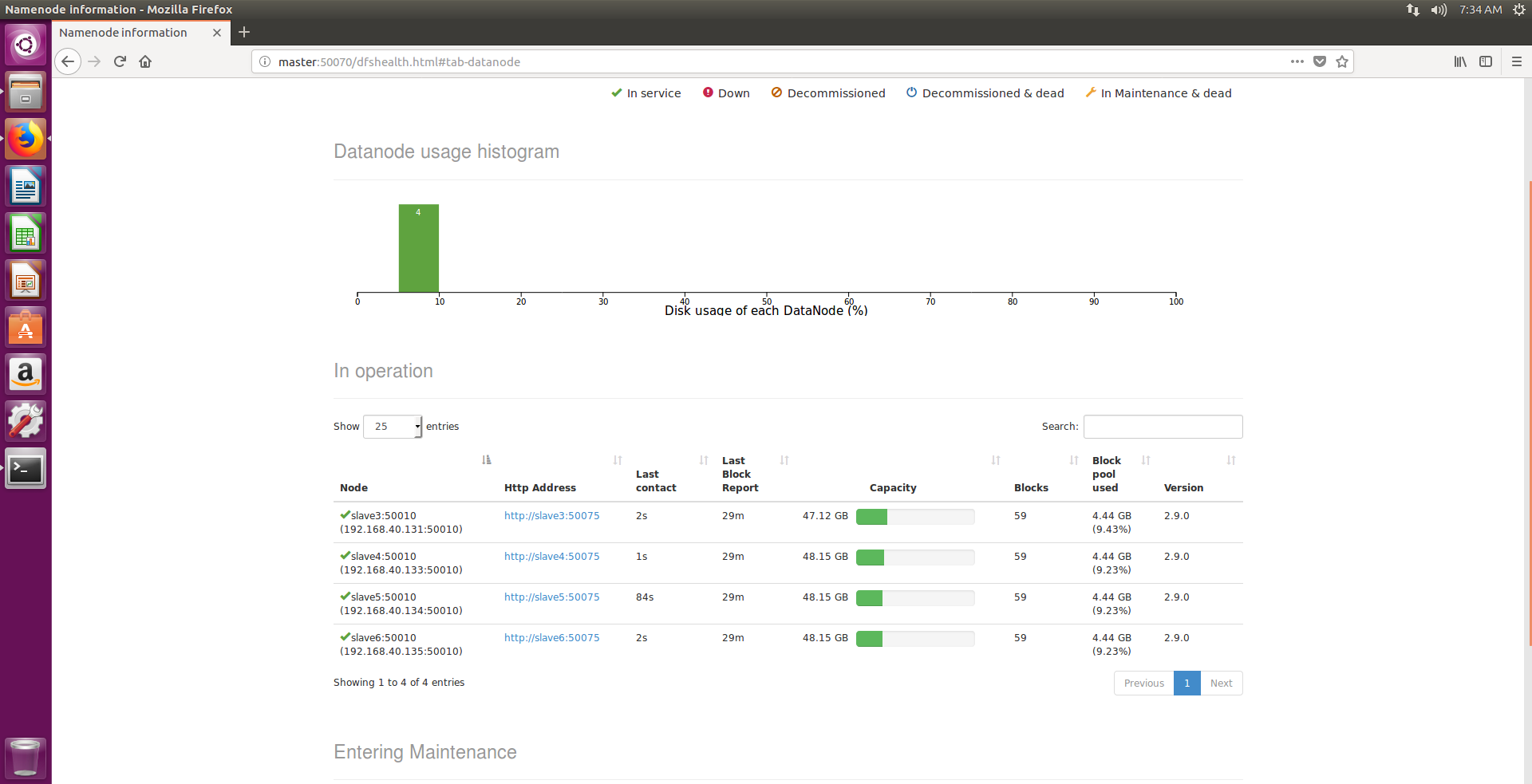
* Stop HDFS

hadoop\_user@master:~$cd $HADOOP\_HOME/sbin$stop-all.sh

* Hadoop Web UI

Go to the Firefox Web Browser and type:

http://master:50070



**6. Setup Spark**

* Create directory to copy Spark files and give access to hadoop\_user

hadoop\_user@master:~$sudo mkdir -p /usr/local/spark

hadoop\_user@master:~$sudo mkdir -p /usr/local/spark/log

hadoop\_user@master:~$sudo mkdir -p /usr/local/spark/work

hadoop\_user@master:~$sudo chown -R hadoop\_group:hadoop\_user /usr/local/spark

* Extract the contest from the spark package

hadoop\_user@master:~$sudo /home/hadoop\_user/Downloads

hadoop@user@master:~/Downloads$ sudo tar -xvzf spark-2.3.0-bin-hadoop2.7.gz

* Move all the files from hadoop-2.9.0 directory to /usr/local/spark

hadoop\_user@master:~$sudo move /home/Hadoop\_user/Downloads/ spark-2.3.0/\* /usr/local/spark

* Update environment variables in .bashrc file

hadoop\_user@master:~$sudo gedit ~/.bashrc

* Add the following lines at the end of the file:

#SPARK VARIABLES

export SPARK\_HOME=/usr/local/spark

export PATH=$PATH:$SPARK\_HOME/bin

export PYSPARK\_PYTHON=python2.7

* Update spark-env.sh

hadoop\_user@master:~$cd $SPARK\_HOME/conf

hadoop\_user@master:/usr/local/spark/conf$

hadoop\_user@master:/usr/local/spark/conf$cp spark-env.sh.template spark-env.sh

hadoop\_user@master:/usr/local/spark/conf$sudo gedit spark\_env.sh

* Add the following lines at the end of the file:

export JAVA\_HOME=/usr/lib/java/jdk1.8.0

export SPARK\_HOME=/usr/local/spark

export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export SPARK\_WORKER\_DIR=/usr/local/spark/work

export SPARK\_LOG\_DIR=/usr/local/spark/log

export SPARK\_MASTER\_IP=master

* Update spark-defaults.conf

hadoop\_user@master:/usr/local/spark/conf$cp spark-defaults.conf.template spark-defaults.conf

hadoop\_user@master:/usr/local/spark/conf$sudo gedit spark-defaults.conf

* Add the following lines at the end of the file

spark.master spark://master:7077

* Update slaves file

hadoop\_user@master:/usr/local/spark/conf$cp slaves.template slaves

hadoop\_user@master:/usr/local/spark/conf$sudo gedit slaves

* Add the following lines at the end of the file

slave1

slave2

slave3

slave4

* Start standalone master server.

hadoop\_user@master:~$cd $SPARK\_HOME/sbin

hadoop\_user@master:/usr/local/spark/sbin~

hadoop\_user@master:/usr/local/spark/sbin$./start-all.sh

starting org.apache.spark.deploy.master.Master, logging to /usr/local/spark/log/spark-hadoop\_user-org.apache.spark.deploy.master.Master-1-master.out

slave1: starting org.apache.spark.deploy.worker.Worker, logging to /usr/local/spark/log/spark-hadoop\_user-org.apache.spark.deploy.worker.Worker-1-slave1.out

slave2: starting org.apache.spark.deploy.worker.Worker, logging to /usr/local/spark/log/spark-hadoop\_user-org.apache.spark.deploy.worker.Worker-1-slave2.out

slave3: starting org.apache.spark.deploy.worker.Worker, logging to /usr/local/spark/log/spark-hadoop\_user-org.apache.spark.deploy.worker.Worker-1-slave3.out

slave4: starting org.apache.spark.deploy.worker.Worker, logging to /usr/local/spark/log/spark-hadoop\_user-org.apache.spark.deploy.worker.Worker-1-slave4.out

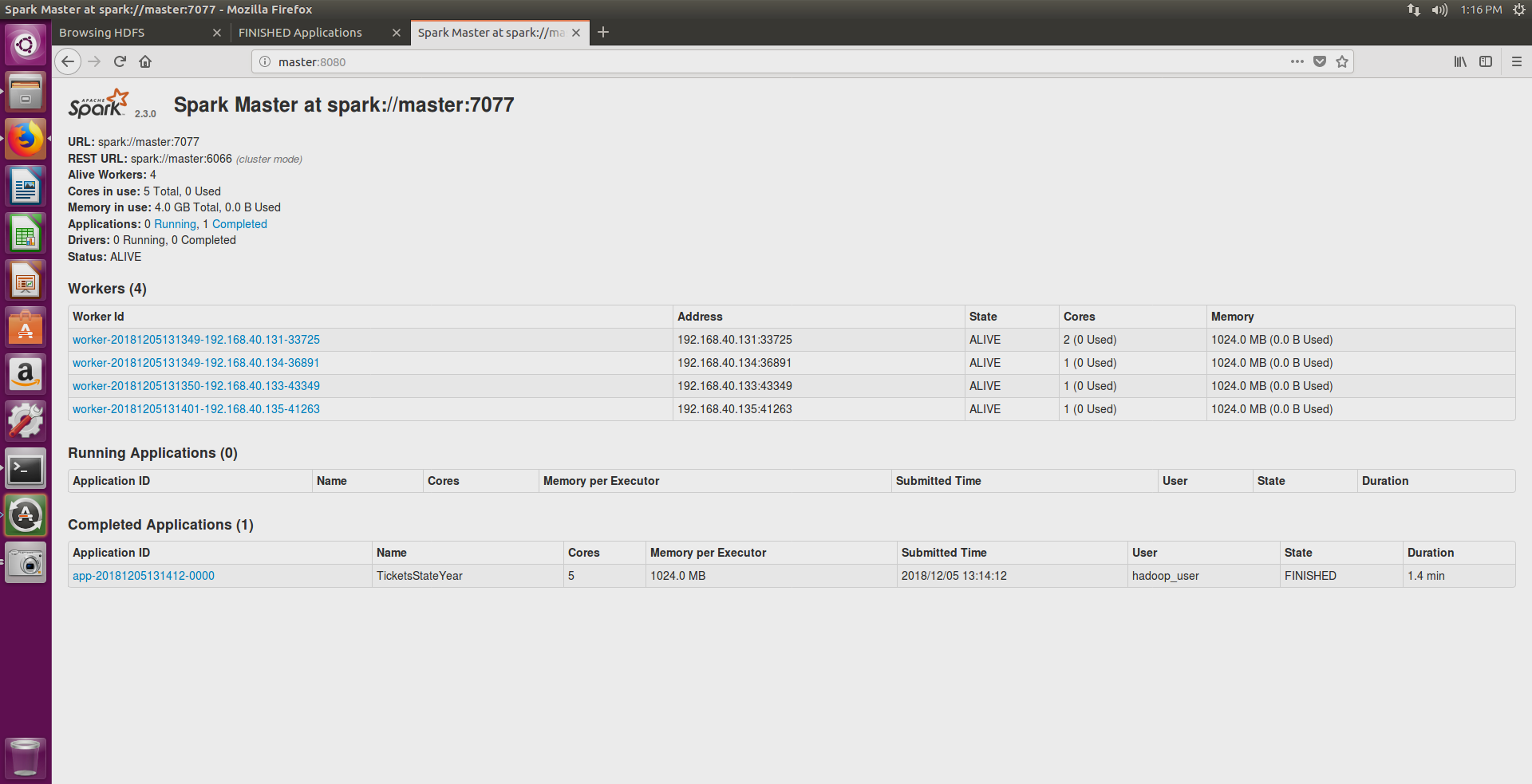
* Execute jps to check if all workers are running

hadoop\_user@master:/usr/local/spark/sbin$jps

* Spark Web UI

Go to the Firefox Web Browser and type:

http://master:8080

****

* Run PySpark MapReduce job

hadoop\_user@master:/usr/local/spark/sbin$ spark-submit --master spark://master:7077 /home/hadoop\_user/Desktop/tickets\_month\_year.py

* Stop Spark

hadoop\_user@master:/usr/local/spark/sbin$./stop-all.sh

slave1: stopping org.apache.spark.deploy.worker.Worker

slave2: stopping org.apache.spark.deploy.worker.Worker

slave3: stopping org.apache.spark.deploy.worker.Worker

slave4: stopping org.apache.spark.deploy.worker.Worker

stopping org.apache.spark.deploy.master.Master