

Apache Hadoop 2.6.0 Pseudo distributed mode set up on CentOS

A guide to install and setup Single - Node Apache Hadoop 2.6.0 Cluster

edureka!

edureka!

Software Requirements

- ✓ VMware Player or Oracle Virtual Box
- ✓ CentOS Virtual Machine

Hardware Requirements

- ✓ Intel Core i3 processor or higher
- ✓ **8 GB RAM Recommended**
- ✓ **300 GB for VM Recommended (By default 40 GB is taken)**

edureka!

Introduction

This setup and configuration document is a guide to setup a single-Node Apache Hadoop 2.6 cluster on a CentOS virtual machine on your PC.

In a Pseudo distributed cluster single machine acts as Master and slave. So in this cluster all the daemons are running single machine.

Note: The configuration described here is intended for learning purposes only.

edureka!

Section-1: Setting up the CentOS Virtual Machine.

1.1: Download the CentOS from the below link.

https://edureka.wistia.com/medias/n8s4sh3tek/download?media_file_id=44348215

Extract the CentOS using WinRAR. You will get the CentOS virtual machine Image.

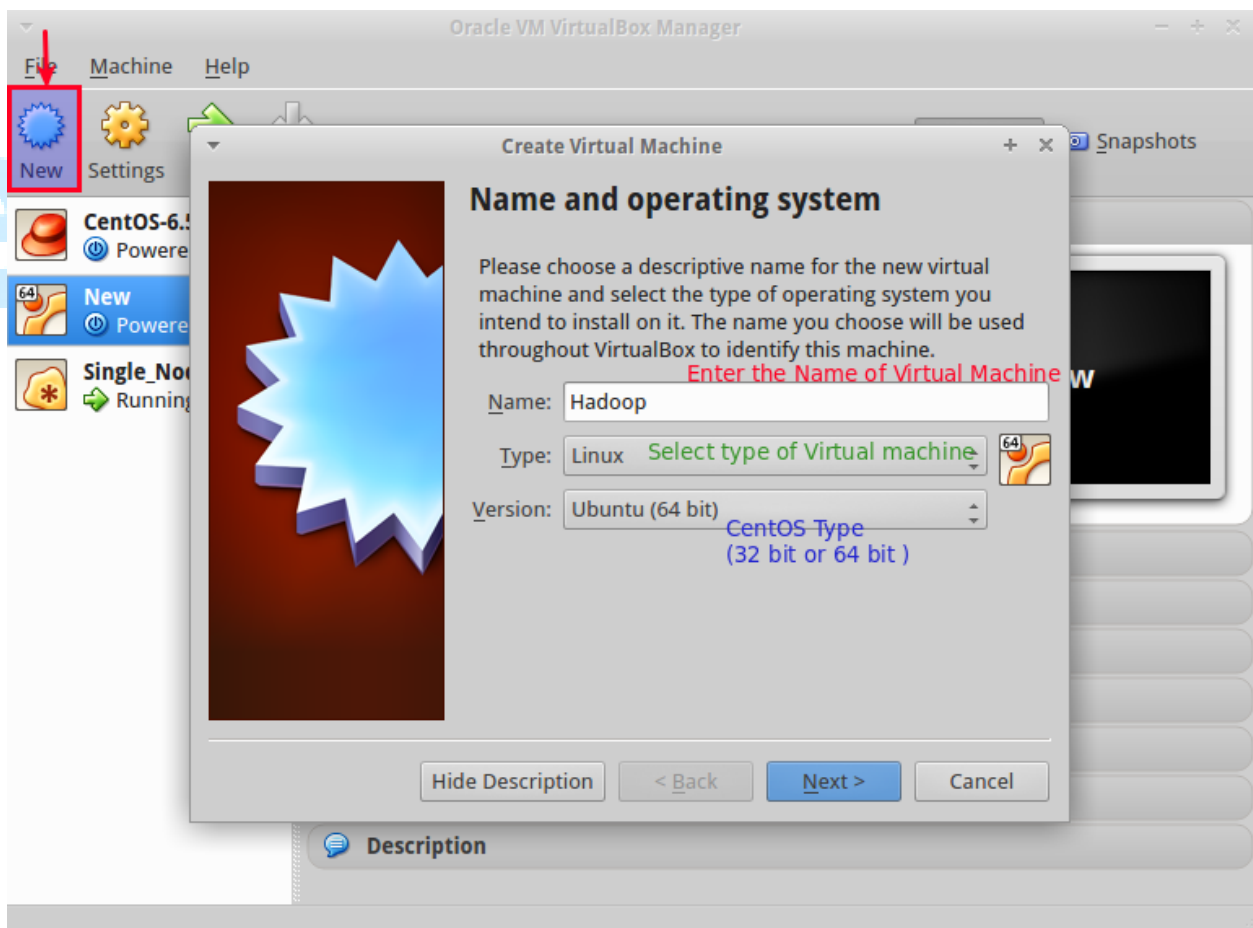
1.2: Download the install the Oracle virtual box or VMware player to open the CentOS Virtual machine.

Oracle Virtual box: <http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html>

Or

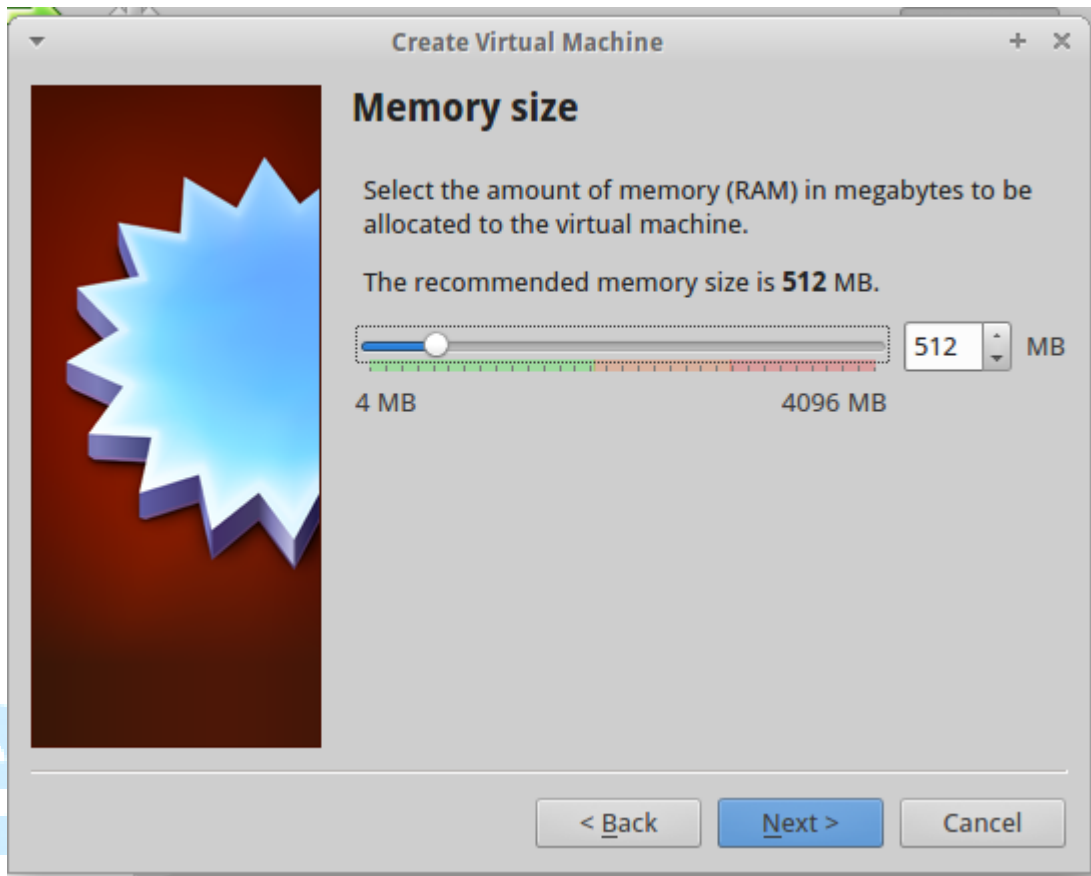
VMware Player: <https://www.vmware.com/tryvmware/?p=player>

1.3: In a oracle virtual box Click on New and Add the CentOS properties.



Click on Next button.

1.4: Add the RAM to your Virtual Machine. You can increase the Virtual machine RAM by dragging forward and backward.

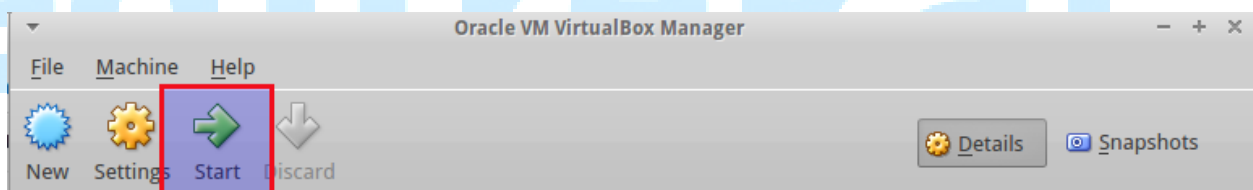


Click on next.

1.5 : Select 3rd Option (Use an existing virtual hard drive file)and click on the folder icon, and go to the path where you have extracted the CentOS virtual machine in 1.1 step, Select centos-6.2-x64-virtual-machine-org.vmdk file. Click on Create button.



Click on Start Button.



1.6: It open the CentOS virtual Machine with the user tom.

User name: tom

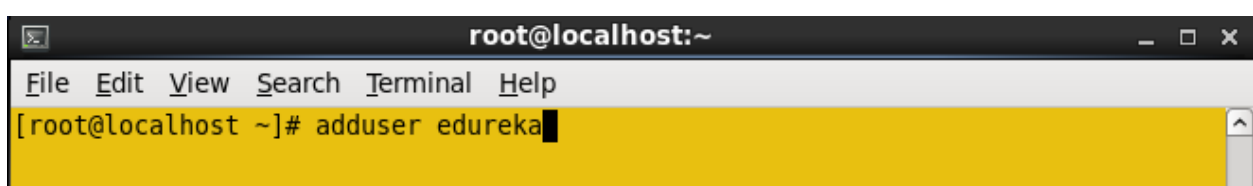
Password: tomtom

Open the terminal and login to root user.

Command: su - root

Password: tomtom

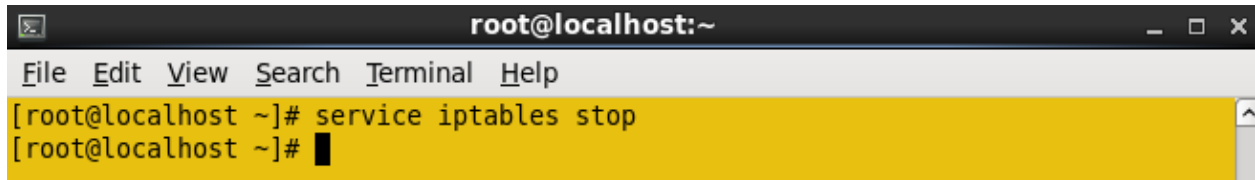
1.7 : Create the any user (in my case user is edureka) and using below commands.



Edit the password.

Command: `passwd edureka`

1.8 : Run the below commands to stop the **iptables**. Run this command from root user.

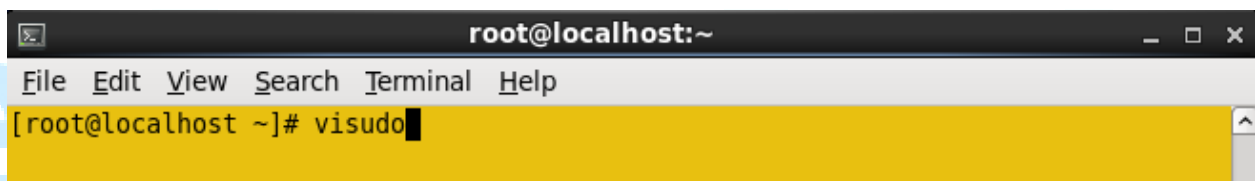
A terminal window titled 'root@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command '[root@localhost ~]# service iptables stop' has been entered and executed, resulting in a new prompt '[root@localhost ~]#'.

Stop file walls permanatly.

Command: `chkconfig iptables off`

Add the **edureka** user to **sudoers** file.

Command to open the sudoers file is : `visudo`

A terminal window titled 'root@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command '[root@localhost ~]# visudo' has been entered and executed, resulting in a new prompt '[root@localhost ~]#'.

Add below command after the link **# allow root to run any command in any where.**

To edit the vi editor you have to press i button.

edureka ALL=(ALL) ALL

```

root@localhost:~
File Edit View Search Terminal Help
## Syntax:
##
##      user      MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)      ALL
edureka ALL=(ALL) ALL
## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOC
ATE, DRIVERS

## Allows people in group wheel to run all commands
# %wheel      ALL=(ALL)      ALL

## Same thing without a password
# %wheel      ALL=(ALL)      NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

```

Save the file by press Esc button and wq buttons and press enter.

Reboot the Virtual machine and log in to Edureka user.

Step-2 : To install the hadoop required software.

- 2.1 : Jdk 1.6 and above
- 2.2 : Hadoop-2.6.0 and ssh

2.1.1 : Download the JDK-1.7 to set up the Java path in your hadoop environment.

Create Java directory to store the Java files.

Command: mkdir java

```

edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ mkdir java

```


Change the directory to java

```
edureka@localhost:~/java
File Edit View Search Terminal Help
[edureka@localhost ~]$ cd java
[edureka@localhost java]$ pwd
/home/edureka/java
[edureka@localhost java]$
```

2.1.2 : Download the JDK-1.7 tar ball from below command.

wget <http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-x64.tar.gz>

```
edureka@localhost:~/java
File Edit View Search Terminal Help
[edureka@localhost java]$ wget http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-x64.tar.gz
```

2.1.3 : After download the java file, Extract the JDK tar ball.

Command: tar -xvf jdk-7u79-linux-x64.tar.gz

```
edureka@localhost:~/java
File Edit View Search Terminal Help
[edureka@localhost java]$ tar -xvf jdk-7u79-linux-x64.tar.gz
```

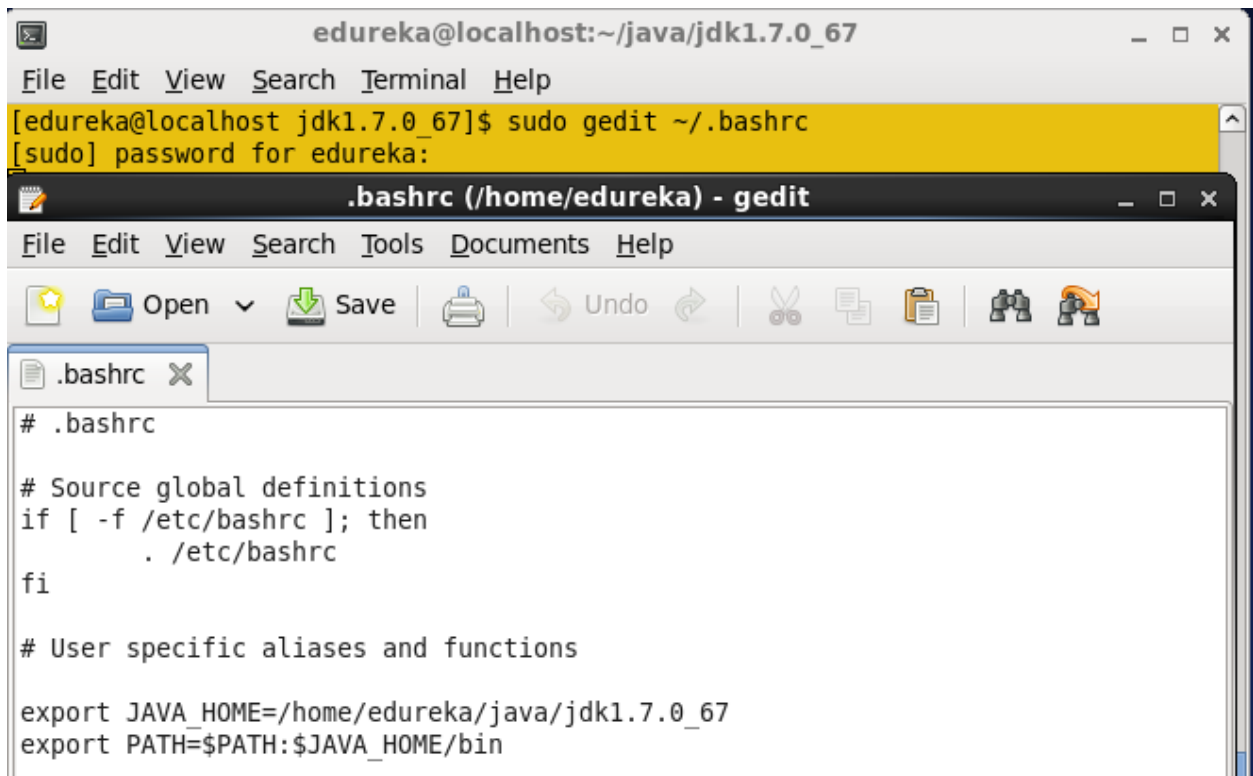
Tarball extraction creates Java directory.

```
edureka@localhost:~/java/jdk1.7.0_67
File Edit View Search Terminal Help
[edureka@localhost java]$ ls
jdk1.7.0_67  jdk-7u67-linux-x64.gz
[edureka@localhost java]$ cd jdk1.7.0_67/
[edureka@localhost jdk1.7.0_67]$ pwd
/home/edureka/java/jdk1.7.0_67
[edureka@localhost jdk1.7.0_67]$
```

2.1.4 : Add the Java path to bashrc file.

Command: sudo gedit ~/.bashrc

Add JAVA_HOME and PATH environment variables to .bashrc file.



```

edureka@localhost:~/java/jdk1.7.0_67
File Edit View Search Terminal Help
[edureka@localhost jdk1.7.0_67]$ sudo gedit ~/.bashrc
[sudo] password for edureka:

.bashrc (/home/edureka) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions

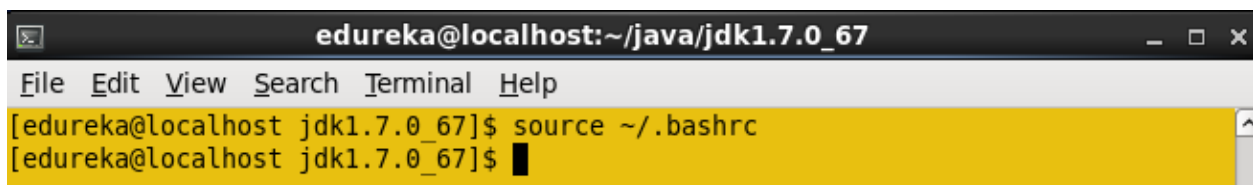
export JAVA_HOME=/home/edureka/java/jdk1.7.0_67
export PATH=$PATH:$JAVA_HOME/bin

```

Save and close the bashrc file.

2.1.5 : Run the below command to apply the changes to current running terminal.

Command : source ~/.bashrc



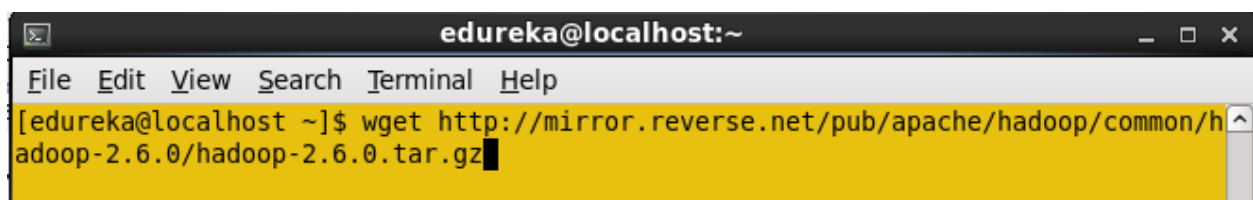
```

edureka@localhost:~/java/jdk1.7.0_67
File Edit View Search Terminal Help
[edureka@localhost jdk1.7.0_67]$ source ~/.bashrc
[edureka@localhost jdk1.7.0_67]$

```

2.2.1 : Download the stable hadoop-2.6.0 version tarball from the apache mirrors.

wget <http://mirrors.sonic.net/apache/hadoop/common/stable/hadoop-2.6.0.tar.gz>



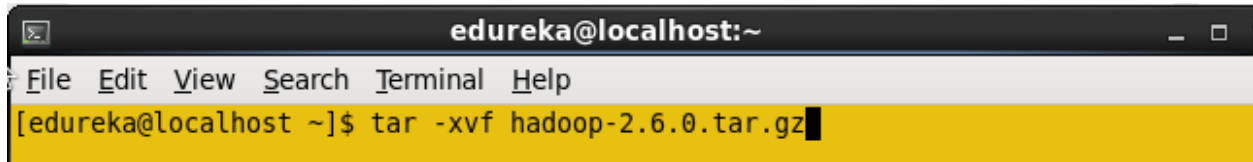
```

edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ wget http://mirror.reverse.net/pub/apache/hadoop/common/h
adoop-2.6.0/hadoop-2.6.0.tar.gz

```

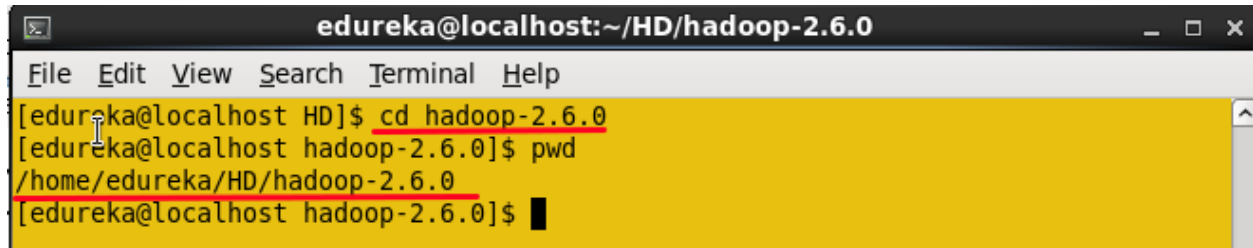
2.2.2 : extract the hadoop-2.6.0.tar.gz

Command: tar -xvf hadoop-2.6.0.tar.gz



```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ tar -xvf hadoop-2.6.0.tar.gz
```

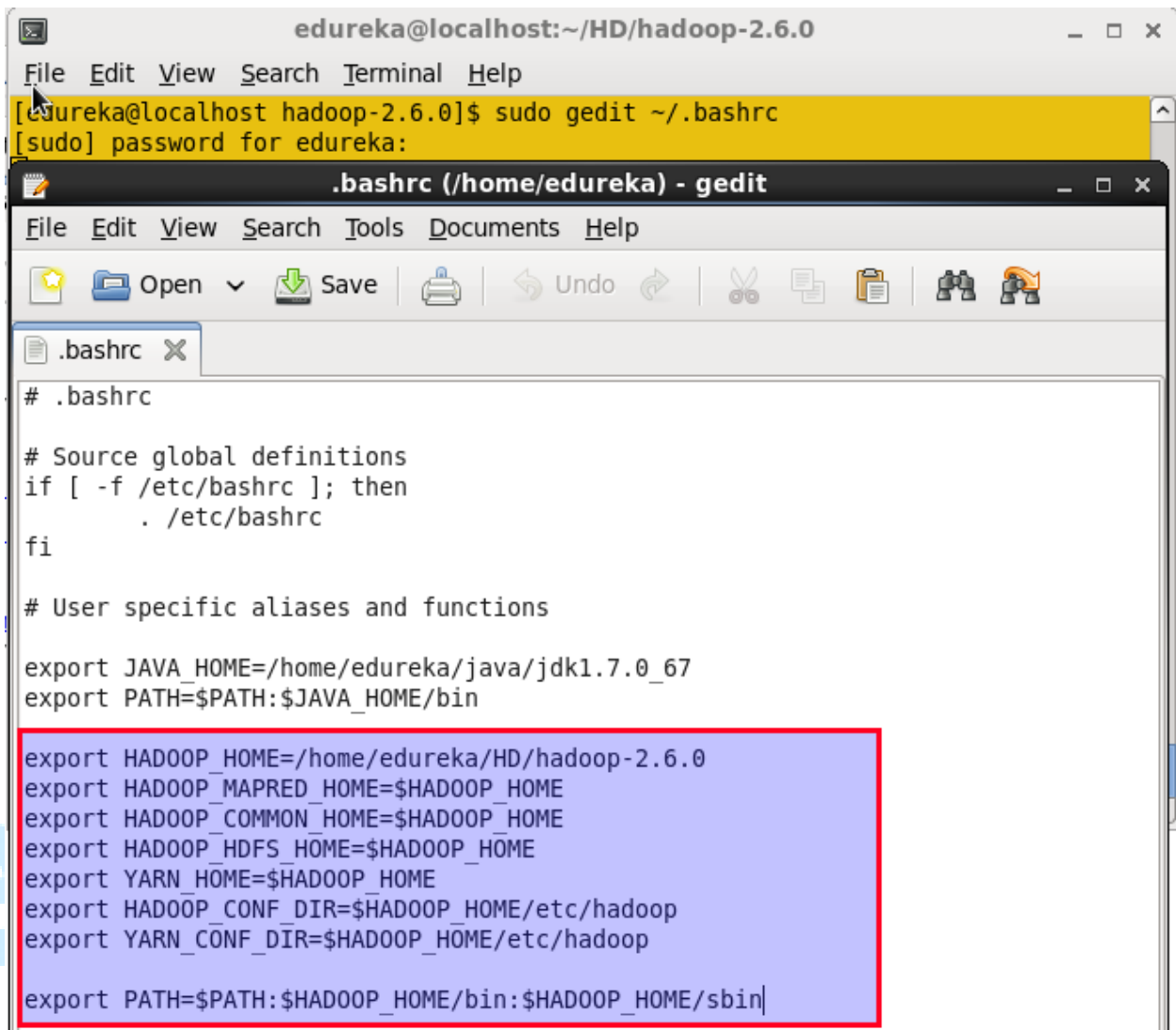
2.2.3 : change the directory to hadoop-2.6.0



```
edureka@localhost:~/HD/hadoop-2.6.0
File Edit View Search Terminal Help
[edureka@localhost HD]$ cd hadoop-2.6.0
[edureka@localhost hadoop-2.6.0]$ pwd
/home/edureka/HD/hadoop-2.6.0
[edureka@localhost hadoop-2.6.0]$
```

2.2.4 : Add the HADOOP_HOME and Hadoop environment variables to bashrc file.

```
export HADOOP_HOME=<Hadoop directory path>
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export YARN_CONF_DIR=$HADOOP_HOME/etc/hadoop
export PATH=$HADOOP_HOME/bin/:$HADOOP_HOME/sbin/:$PATH
```



```
edureka@localhost:~/HD/hadoop-2.6.0
File Edit View Search Terminal Help
[edureka@localhost hadoop-2.6.0]$ sudo gedit ~/.bashrc
[sudo] password for edureka:

.bashrc (/home/edureka) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
.bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

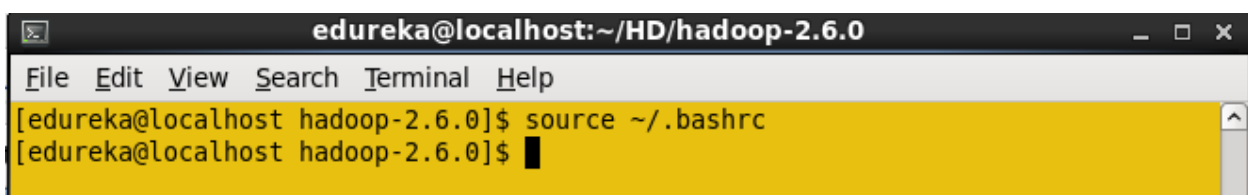
# User specific aliases and functions

export JAVA_HOME=/home/edureka/java/jdk1.7.0_67
export PATH=$PATH:$JAVA_HOME/bin

export HADOOP_HOME=/home/edureka/HD/hadoop-2.6.0
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export YARN_CONF_DIR=$HADOOP_HOME/etc/hadoop
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

save the file and close the bashrc file.

Run the source command to apply the bashrc changes to current running terminal.



```
edureka@localhost:~/HD/hadoop-2.6.0
File Edit View Search Terminal Help
[edureka@localhost hadoop-2.6.0]$ source ~/.bashrc
[edureka@localhost hadoop-2.6.0]$
```

2.2.5 : Create two directory any location to store Namenode metadata and Datanode HDFS blocks.

```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ mkdir namenode
[edureka@localhost ~]$ mkdir datanode
[edureka@localhost ~]$
```

2.2.6 : Change the permission to both the directories.

```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ chmod 755 namenode/
[edureka@localhost ~]$ chmod 755 datanode/
[edureka@localhost ~]$
```

2.2.7 : Change the hadoop configuration files.

To configure the Hadoop cluster you will need to configure the environment in which the Hadoop daemons execute as well as the configuration parameters for the Hadoop daemons.

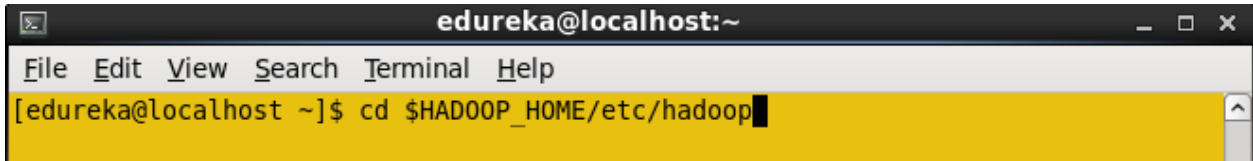
The configuration files that need to change is :

Configuration Filenames	Description
hadoop-env.sh	Environment variables that are used in the scripts to run Hadoop.
core-site.xml	Configuration settings for Hadoop Core such as I/O settings that are common to all Hadoop applications and MapReduce.
hdfs-site.xml	Configuration settings for HDFS daemons, the namenode, the secondary namenode, and the data nodes.
mapred-site.xml	Configuration settings for MapReduce Applications.
yarn-site.xml	Configuration settings for ResourceManager and NodeManager.

Change the directory to hadoop configuration location.

All the hadoop configurations files are located in hadoop directory present in the path \$HADOOP_HOME/etc/hadoop

In a each and every configuration file you have to add the configuration property and it's value using property and value tag.



```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ cd $HADOOP_HOME/etc/hadoop
```

2.2.8 : Edit the core-site.xml configuration file and save it and close the file.

Property	Description
fs.defaultFS	The name of the default file system. A URI whose scheme and authority determine the FileSystem implementation. The uri's scheme determines the config property (fs.SCHEME.impl) naming the FileSystem implementation. The uri's authority is used to determine the host, port, etc. for a file

edureka!

```

edureka@localhost:~/HD/hadoop-2.6.0/etc/hadoop
File Edit View Search Terminal Help
[edureka@localhost hadoop]$ sudo gedit core-site.xml

core-site.xml (/home/edureka/HD/hadoop-2.6.0/etc/hadoop) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
core-site.xml
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.defaultFS</name>
    <value>hdfs://localhost.localdomain:9000</value>
  </property>
</configuration>
XML Tab Width: 8 Ln 1, Col 1 INS

```

2.2.9 : Edit the hdfs-site.xml file.

Give the namenode and datanode directory paths for `dfs.namenode.name.dir` and `dfs.datanode.data.dir` values which is created in step:2.2.5

Property	Description
dfs.namenode.name.dir	Determines where on the local filesystem the DFS name node should store the name table (fsimage). If this is a comma-delimited list of directories the table is replicated in all of the directories, for redundancy.
dfs.datanode.data.dir	Determines where on the local filesystem an DFS data node should store its data. If this is a comma-delimited list of directories, then data will be stored in all of the directories.

	directories, typically on different devices. Directories that do not exist
dfs.replication	Default block replication. The actual number of replications can be specified when the file is created. The default is used if replication is not specified in
dfs.permissions.enabled	If "true", enable permission checking in HDFS. If "false", permissions are turned off, but all other behavior is unchanged. Switching from on to off or vice versa value to the other does not change the mode, owner or group of files or

```
*hdfs-site.xml X
I 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>dfs.namenode.name.dir</name>
<value>/home/edureka/HD/hadoop/data/namenode</value>
</property>

<property>
<name>dfs.replication</name>
<value>1</value>
</property>

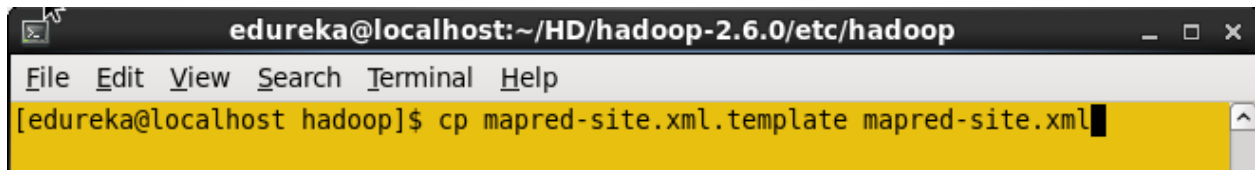
<property>
<name>dfs.datanode.data.dir</name>
<value>/home/edureka/HD/hadoop/data/datanode</value>
</property>

<property>
<name>dfs.permissions.enabled</name>
<value>>false</value>
</property>

</configuration>
```


2.2.10 : Edit the mapred-site.xml file.

Sometimes the mapred-site.xml file will not be available, but we can create the mapred-site.xml file using mapred-site.xml.template file.

A terminal window titled 'edureka@localhost:~/HD/hadoop-2.6.0/etc/hadoop' with a menu bar (File, Edit, View, Search, Terminal, Help). The command '[edureka@localhost hadoop]\$ cp mapred-site.xml.template mapred-site.xml' is entered and highlighted in yellow.

```
edureka@localhost:~/HD/hadoop-2.6.0/etc/hadoop
File Edit View Search Terminal Help
[edureka@localhost hadoop]$ cp mapred-site.xml.template mapred-site.xml
```

2.2.11 : Now edit the mapred-site.xml file.

Property	Description
mapreduce.framework.name	Execution framework set to Hadoop YARN.

edureka!

```

edureka@localhost:~/HD/hadoop-2.6.0/etc/hadoop
File Edit View Search Terminal Help
[edureka@localhost hadoop]$ gedit mapred-site.xml

*mapred-site.xml (~/.HD/hadoop-2.6.0/etc/hadoop) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
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>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
</configuration>

```

2.2.12 : Edit the yarn-site.xml file.

Command : `sudo gedit yarn-site.xml`

Property	Description
yarn.nodemanager.aux-services	<p>Selects a shuffle service that needs to be set for MapReduce to run.</p> <p>This property, in conjunction with other properties, sets "direct shuffle" shuffle for MapReduce.</p> <p>Default value: mapreduce_shuffle, mapr_direct_shuffle</p>
yarn.nodemanager.aux-services.mapreduce_shuffle.class	<p>This property, in conjunction with other properties, sets "direct shuffle" shuffle for MapReduce.</p> <p>Default value: org.apache.hadoop.mapred.ShuffleHandler</p>

yarn.resourcemanager.resource-tracker.address	Provide the resource tracker details to Yarn services.
yarn.resourcemanager.scheduler.address	Applications in the cluster talk to the ResourceManager.
yarn.resourcemanager.address	The hostname of the ResourceManager and the port on which the client connects to the Resource Manager. Example value: \${yarn.resourcemanager.hostname}:{Port number}

```

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.
-->
<configuration>
<!-- Site specific YARN configuration properties -->
  <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>

  <property>
    <name>yarn.resourcemanager.resource-tracker.address</name>
    <value>localhost.localdomain:9001</value>
  </property>

  <property>
    <name>yarn.resourcemanager.scheduler.address</name>
    <value>localhost.localdomain:9002</value>
  </property>

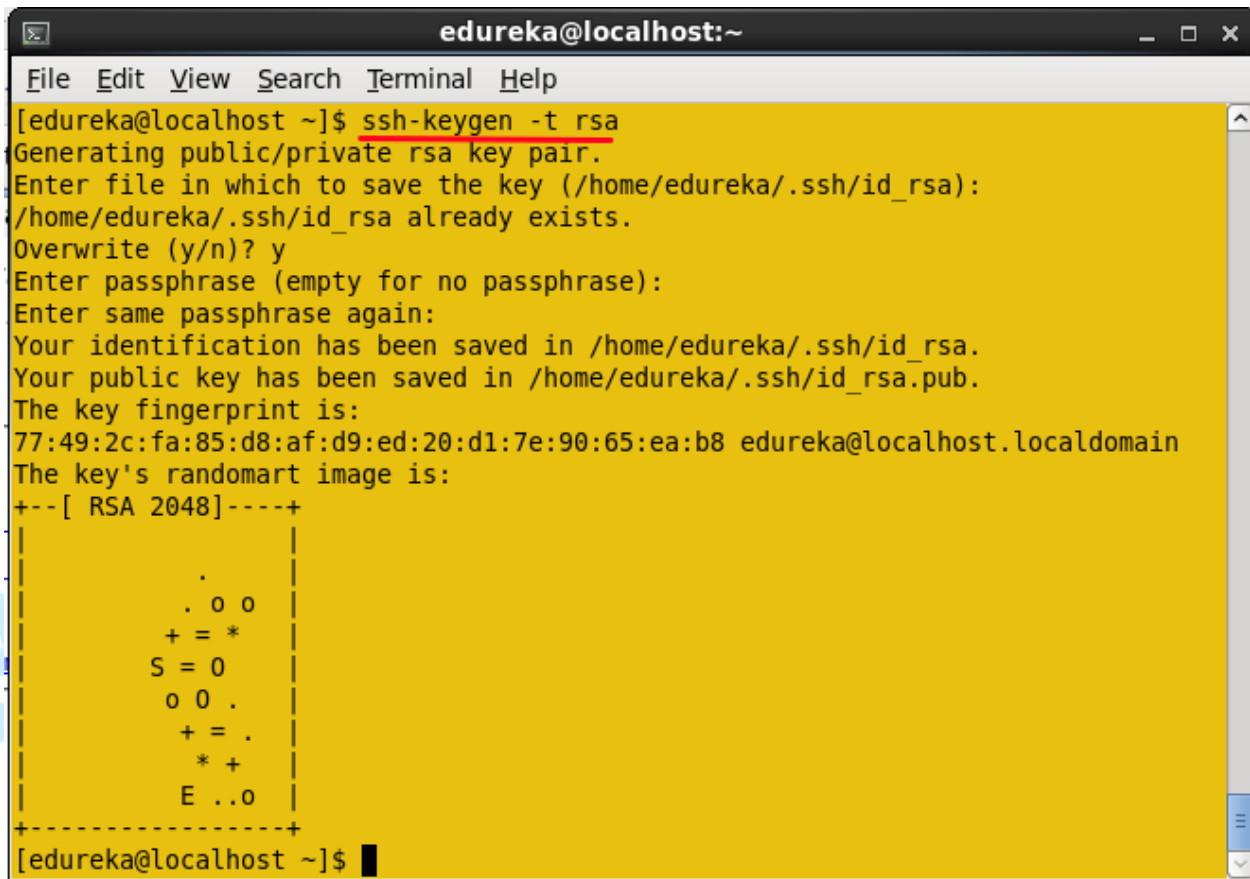
  <property>
    <name>yarn.resourcemanager.address</name>
    <value>localhost.localdomain:9003</value>
  </property>
</configuration>

```

2.2.13 : Create the ssh key to start all the hadoop daemons without asking user password.

Command to create the ssh key is : `ssh-keygen -t rsa`

Note : Once run this command it asks to the location where key should be stored and passphrase, You no need to use any custom location and passphrase just click on enter button.



```

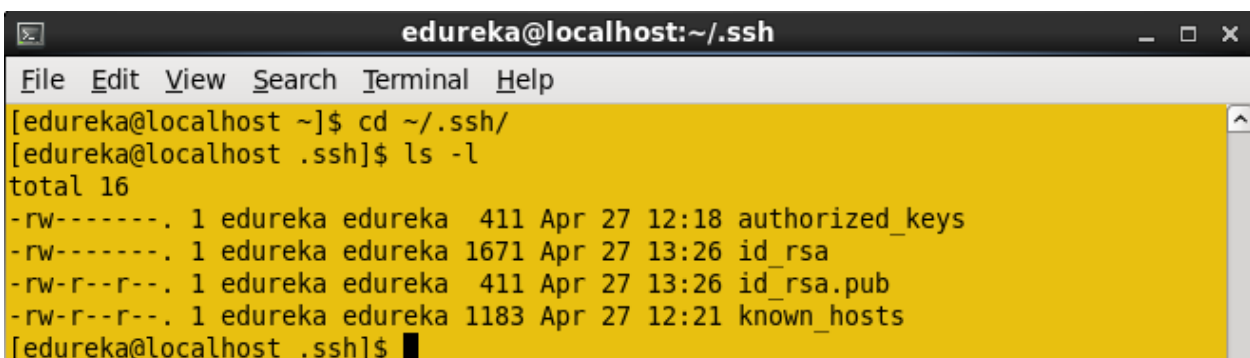
edureka@localhost:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/edureka/.ssh/id_rsa):
/home/edureka/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/edureka/.ssh/id_rsa.
Your public key has been saved in /home/edureka/.ssh/id_rsa.pub.
The key fingerprint is:
77:49:2c:fa:85:d8:af:d9:ed:20:d1:7e:90:65:ea:b8 edureka@localhost.localdomain
The key's randomart image is:
+--[ RSA 2048 ]-----+
|
|      .
|    . o o
|   + = *
|  S = 0
| o 0 .
|   + = .
|    * +
|   E . . o
|
+-----+
edureka@localhost ~]$

```

2.2.14 : Change the directory to ssh location.

Command : `cd ~/.ssh`

Once change to ssh directory you will see the private key and public key.



```

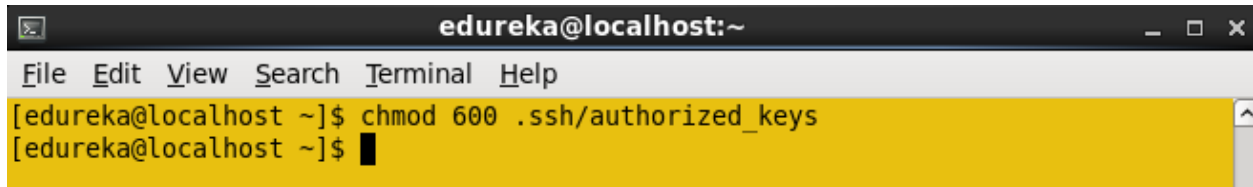
edureka@localhost:~/.ssh$ cd ~/.ssh/
edureka@localhost .ssh$ ls -l
total 16
-rw-----. 1 edureka edureka 411 Apr 27 12:18 authorized_keys
-rw-----. 1 edureka edureka 1671 Apr 27 13:26 id_rsa
-rw-r--r--. 1 edureka edureka 411 Apr 27 13:26 id_rsa.pub
-rw-r--r--. 1 edureka edureka 1183 Apr 27 12:21 known_hosts
edureka@localhost .ssh$

```

2.2.15 : Create the authorized_keys file to login without password.

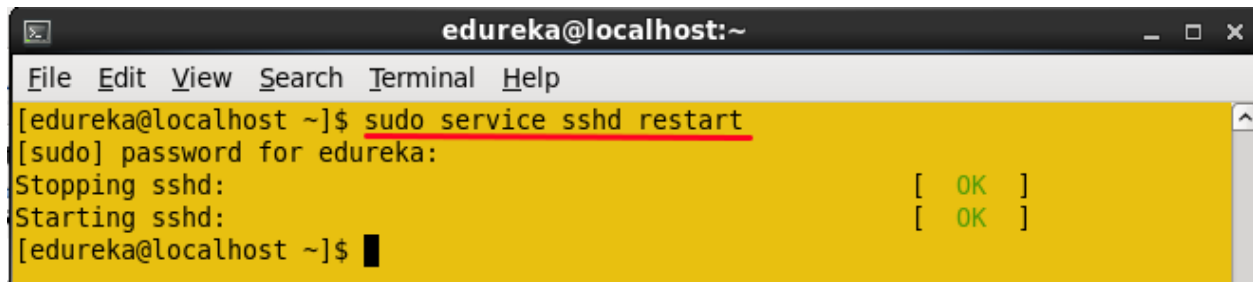
```
cat id_rsa.pub >> ~/.ssh/authorized_keys
```

Change the permission to authorized_keys.



```
edureka@localhost:~  
File Edit View Search Terminal Help  
[edureka@localhost ~]$ chmod 600 .ssh/authorized_keys  
[edureka@localhost ~]$
```

2.2.17 : Restart the ssh service.



```
edureka@localhost:~  
File Edit View Search Terminal Help  
[edureka@localhost ~]$ sudo service sshd restart  
[sudo] password for edureka:  
Stopping sshd: [ OK ]  
Starting sshd: [ OK ]  
[edureka@localhost ~]$
```

edureka!

2.2.18 : Format the namenode.

```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ hadoop namenode -format
```

2.2.19 : start the dfs daemons and yarn daemons using start-dfs.sh script and start-yarn.sh script.

```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ start-dfs.sh
15/04/27 14:23:09 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost.localdomain]
localhost.localdomain: starting namenode, logging to /home/edureka/HD/hadoop-2.6.0/logs/hadoop-edureka-namenode-localhost.localdomain.out
localhost: starting datanode, logging to /home/edureka/HD/hadoop-2.6.0/logs/hadoop-edureka-datanode-localhost.localdomain.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /home/edureka/HD/hadoop-2.6.0/logs/hadoop-edureka-secondarynamenode-localhost.localdomain.out
15/04/27 14:23:33 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
[edureka@localhost ~]$
```

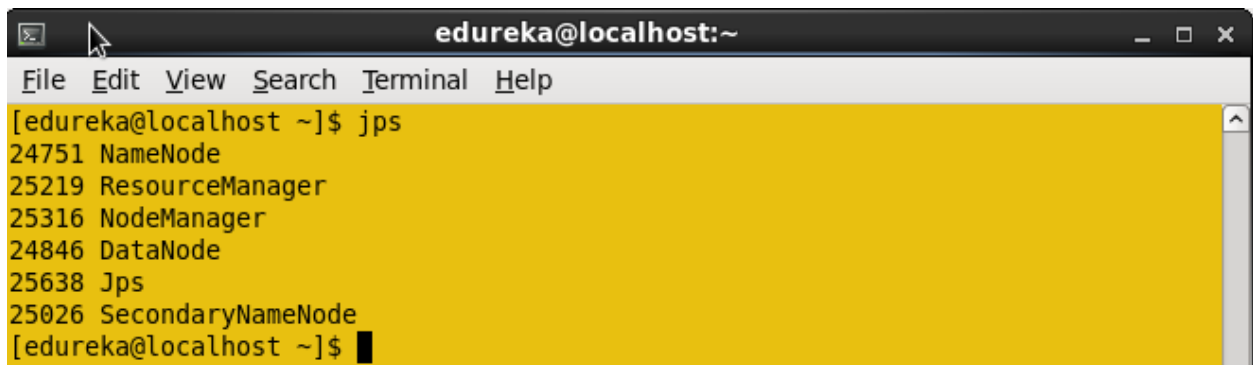
Enter jps (Java Virtual Machine Process Status Tool) command to check the dfs daemons are running or not.

```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ jps
24751 NameNode
25169 Jps
24846 DataNode
25026 SecondaryNameNode
[edureka@localhost ~]$
```

2.2.20 : Start yarn daemons.

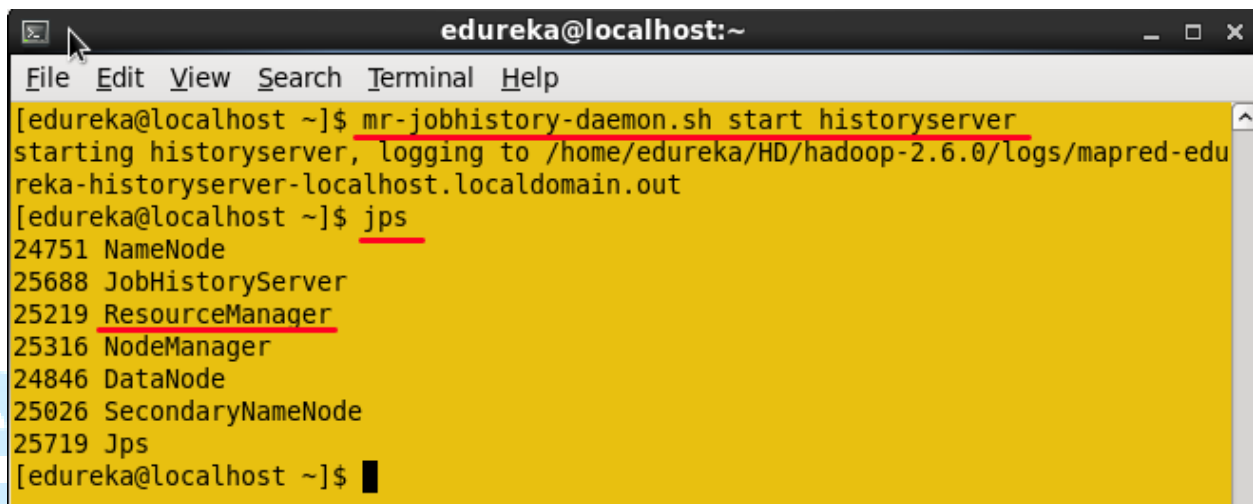
```
edureka@localhost:~
File Edit View Search Terminal Help
[edureka@localhost ~]$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/edureka/HD/hadoop-2.6.0/logs/yarn-edureka-resourcemanager-localhost.localdomain.out
localhost: starting nodemanager, logging to /home/edureka/HD/hadoop-2.6.0/logs/yarn-edureka-nodemanager-localhost.localdomain.out
[edureka@localhost ~]$
```

Enter jps command to check all the daemons.

A terminal window titled 'edureka@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command '[edureka@localhost ~]\$ jps' and its output: '24751 NameNode', '25219 ResourceManager', '25316 NodeManager', '24846 DataNode', '25638 Jps', and '25026 SecondaryNameNode'. The prompt '[edureka@localhost ~]\$' is followed by a cursor.

```
edureka@localhost:~  
File Edit View Search Terminal Help  
[edureka@localhost ~]$ jps  
24751 NameNode  
25219 ResourceManager  
25316 NodeManager  
24846 DataNode  
25638 Jps  
25026 SecondaryNameNode  
[edureka@localhost ~]$
```

2.2.21 : Start the history server.

A terminal window titled 'edureka@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command '[edureka@localhost ~]\$ mr-jobhistory-daemon.sh start historyserver' which outputs 'starting historyserver, logging to /home/edureka/HD/hadoop-2.6.0/logs/mapred-edureka-historyserver-localhost.localdomain.out'. Then, the command '[edureka@localhost ~]\$ jps' is entered, and the output lists the daemons: '24751 NameNode', '25688 JobHistoryServer', '25219 ResourceManager', '25316 NodeManager', '24846 DataNode', '25026 SecondaryNameNode', and '25719 Jps'. The prompt '[edureka@localhost ~]\$' is followed by a cursor.

```
edureka@localhost:~  
File Edit View Search Terminal Help  
[edureka@localhost ~]$ mr-jobhistory-daemon.sh start historyserver  
starting historyserver, logging to /home/edureka/HD/hadoop-2.6.0/logs/mapred-edureka-historyserver-localhost.localdomain.out  
[edureka@localhost ~]$ jps  
24751 NameNode  
25688 JobHistoryServer  
25219 ResourceManager  
25316 NodeManager  
24846 DataNode  
25026 SecondaryNameNode  
25719 Jps  
[edureka@localhost ~]$
```

You can start the each daemon separately using below commands, Instead of using start-dfs.sh and start-yarn.sh

Daemon	Command
Namenode	hadoop-daemon.sh start namenode
Datanode	hadoop-daemon.sh start datanode
ResourceManager	yarn-daemon.sh start resourcemanager
Nodemanager	yarn-daemon.sh start nodemanager
Secondarynamenode	hadoop-daemon.sh start secondarynamenode
Job History Server	mr-jobhistory-daemon.sh start historyserver

edureka!