# Hadoop Configuration for Pseudo-Distribution Mode Installation

## Prerequisites

### Hardware

* 64 bit machine
* 4 GB RAM

### Software

* Linux Operating System
* Java

## Installation

### Linux Operating System

Below steps uses *Oracle VM VirtualBox* and *Linux Mint 17.2* (that is based on *Ubuntu 14.04*).

#### Download

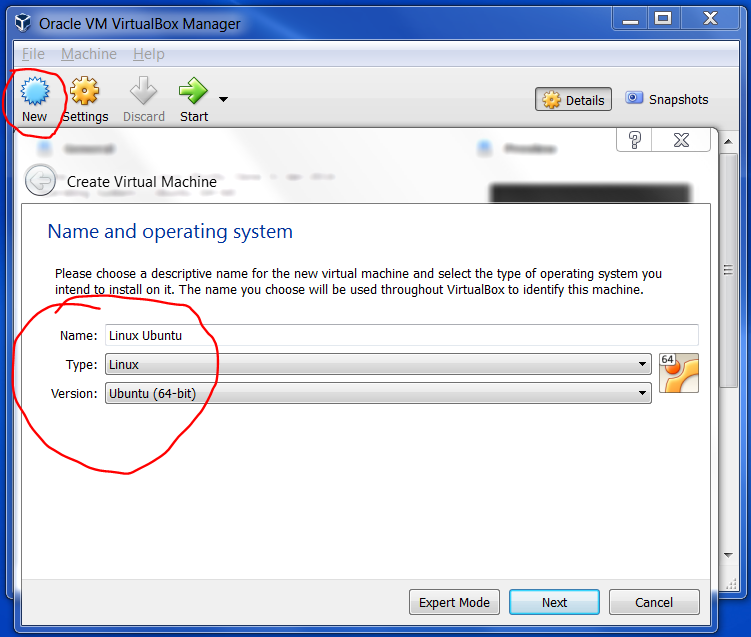
* Oracle VM VirtualBox: <https://www.virtualbox.org/wiki/Downloads>
  + you can download *VirtualBox 5.0.14 for Windows hosts*
* Linux Mint 17.2: <http://www.linuxmint.com/download.php>
  + you can download *Cinnamon 64-bit*

#### Installation Steps for Oracle VM VirtualBox

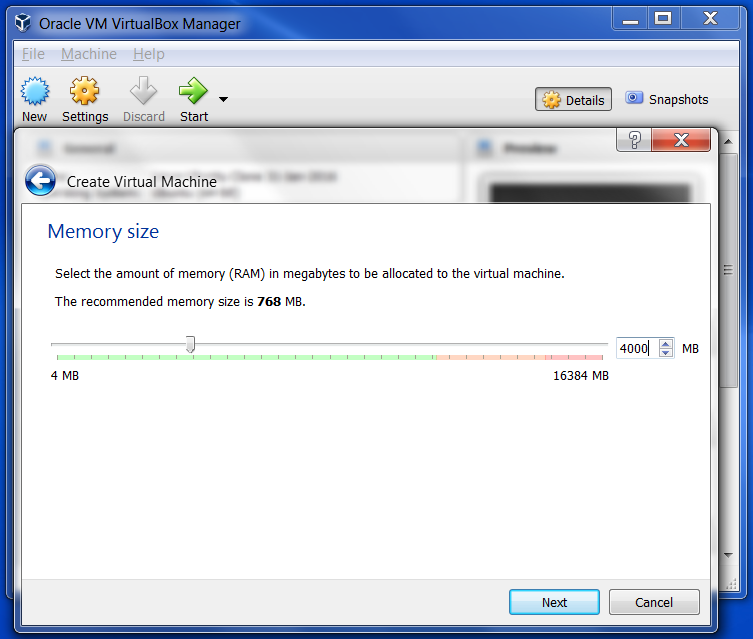
* Please install *Oracle VM VirtualBox* on your *Windows 64-bit* machine

#### Installation Steps for Linux Mint 17.2 on Oracle VM VirtualBox

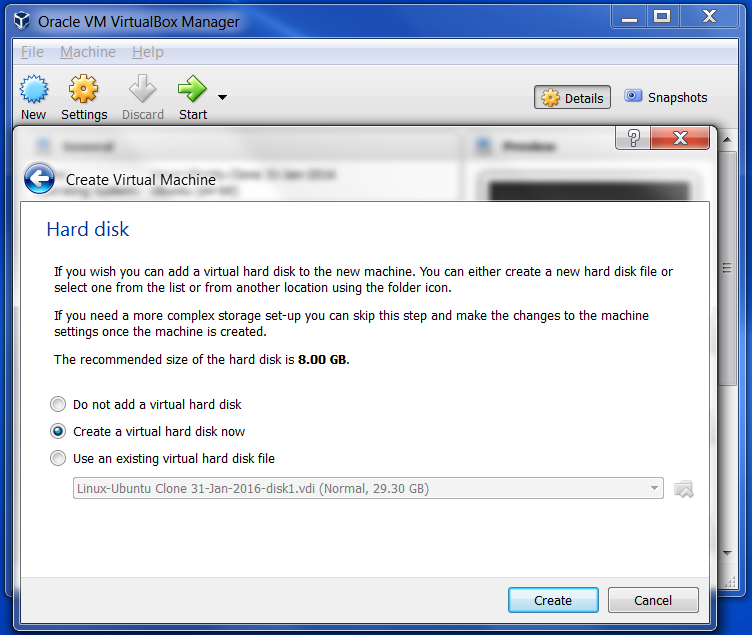
* Open *Oracle VM VirtualBox Manager* and click *New*
  + Give *Name*
  + Select *Linux* for *Type*
  + Select *Ubuntu (64-bit)* for *Version*



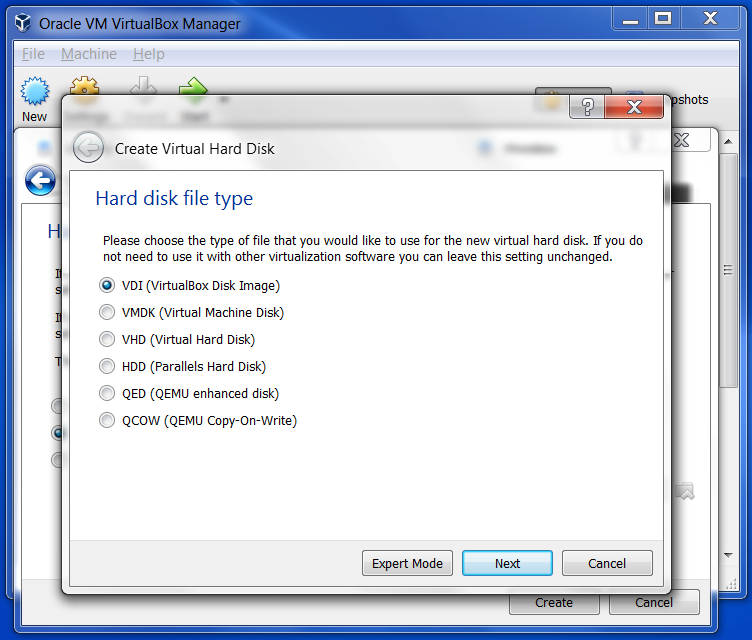
* Select *Memory size* as 4000MB



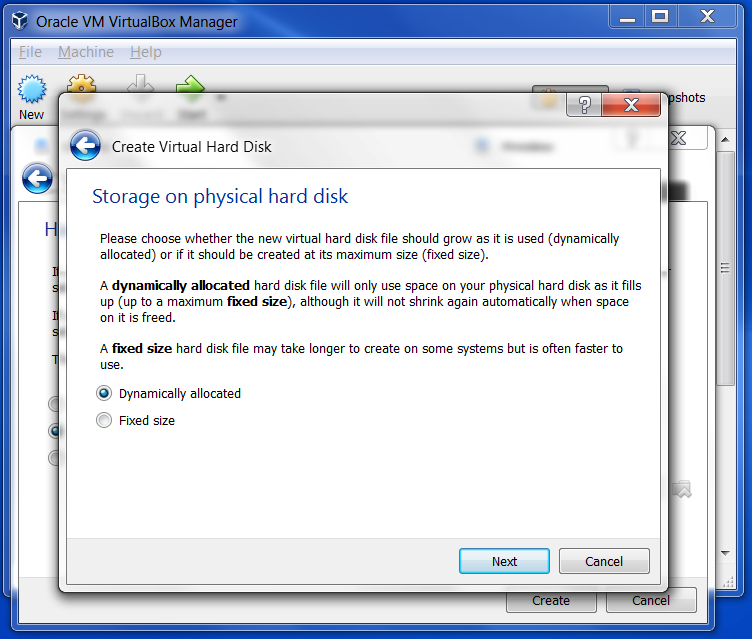
* Select *Create a virtual hard disk now*



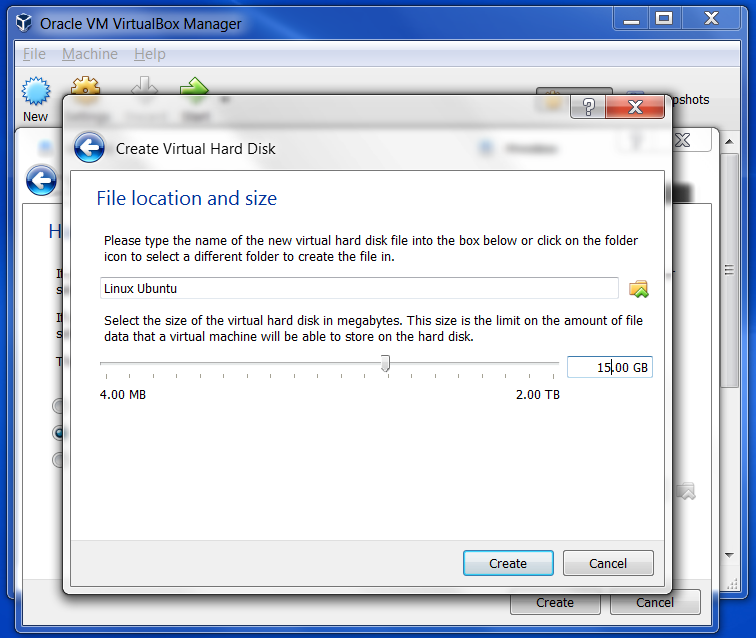
* Select *VDI (VirtualBox Disk Image)*



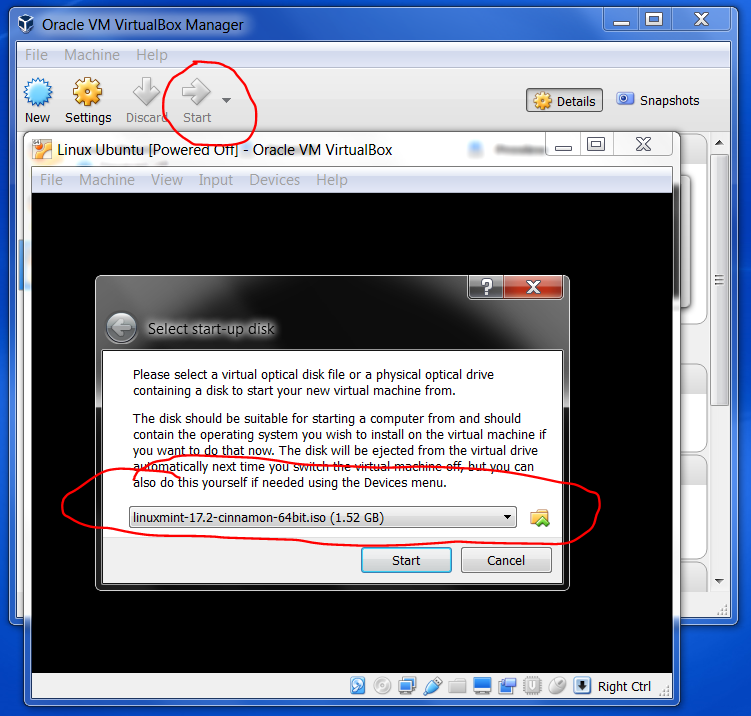
* Select *Dynamically allocated*



* Select the amount of hard disk size



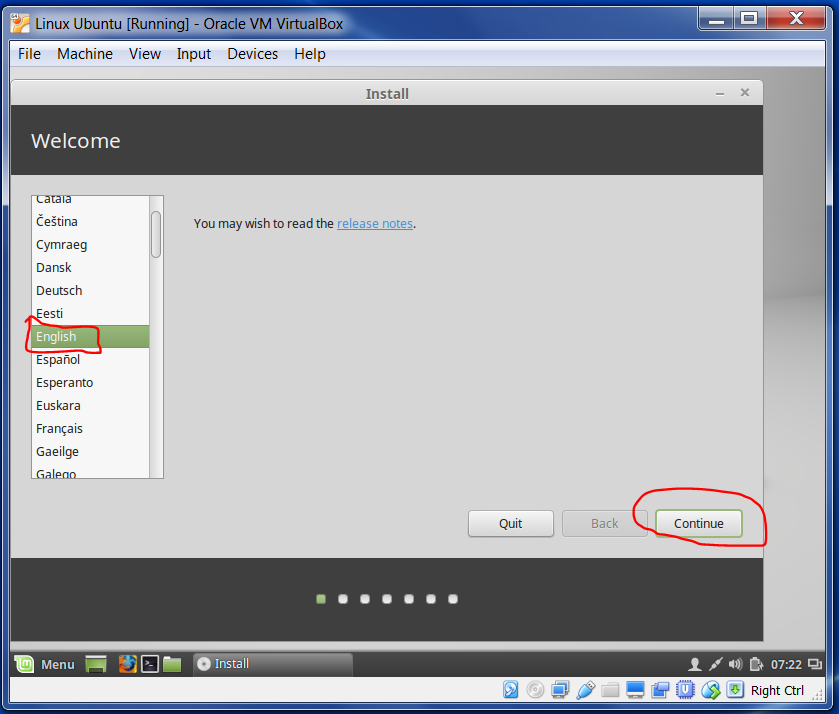
* Select the *Linux Ubuntu* Virtual Machine and click Start
* Select the *linuxmint-17.2-cinnamon-64bit* ISO image and click Start



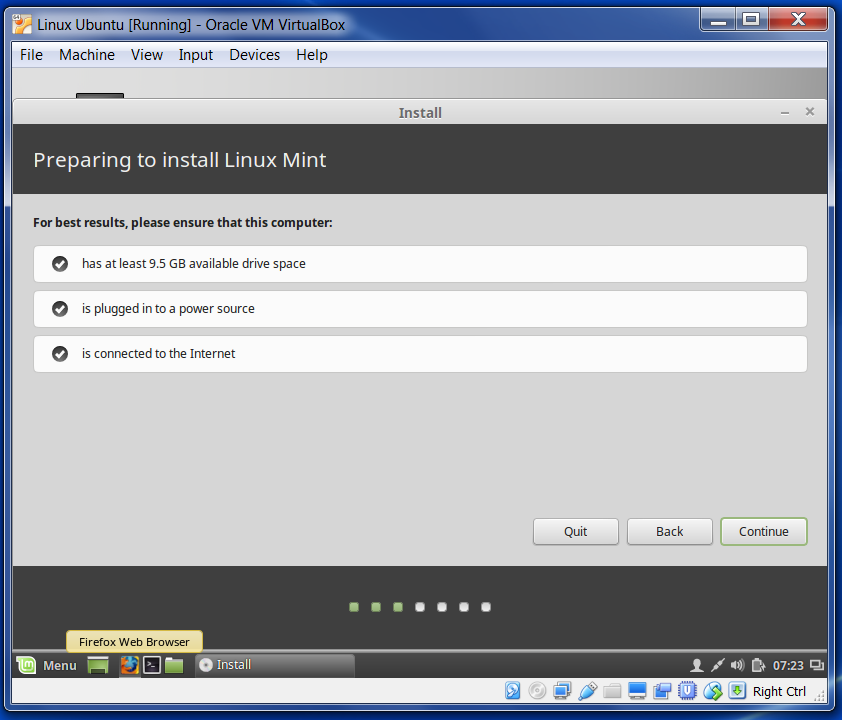
* We will get below interface.



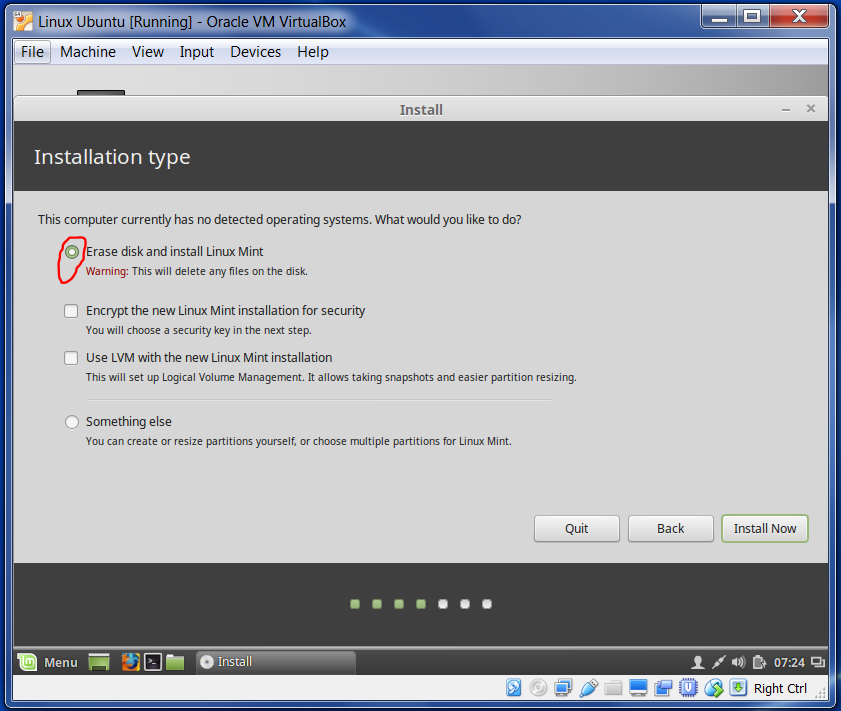
* Double click *Install Linux Mint*
* Select *English* and *Continue*



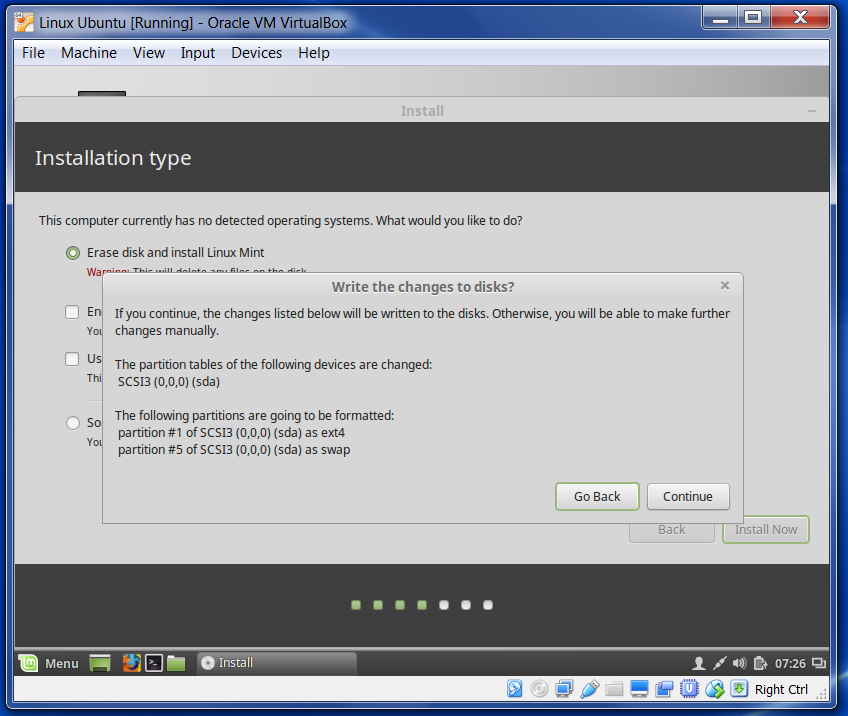
* Select *Continue*



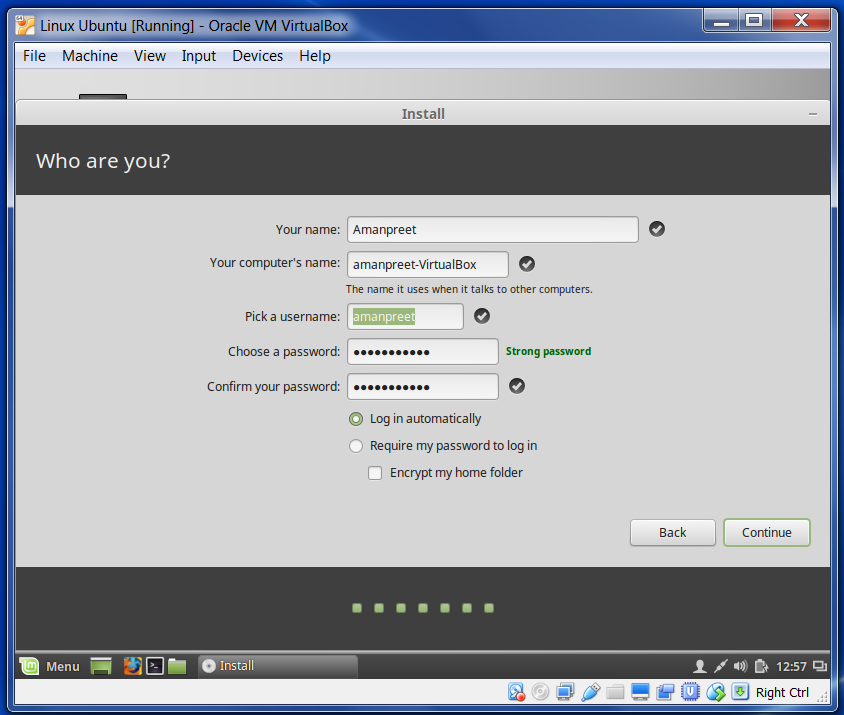
* Select *Erase disk and install Linux Mint* and click *Install Now*



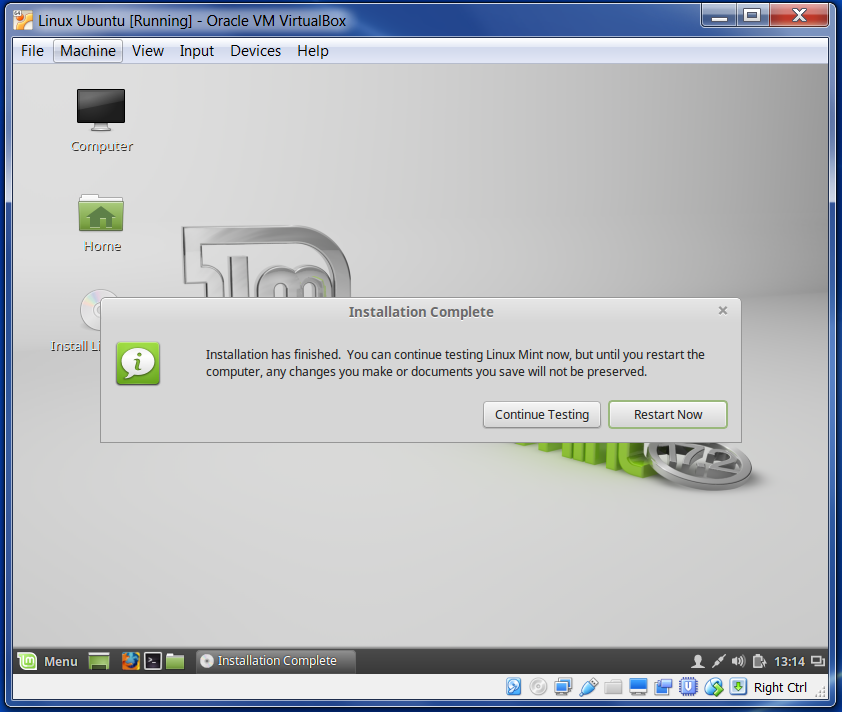
* Click *Continue*



* Create Account and click *Continue*



* After installation finishes, we get below screen. Click on *Restart Now*



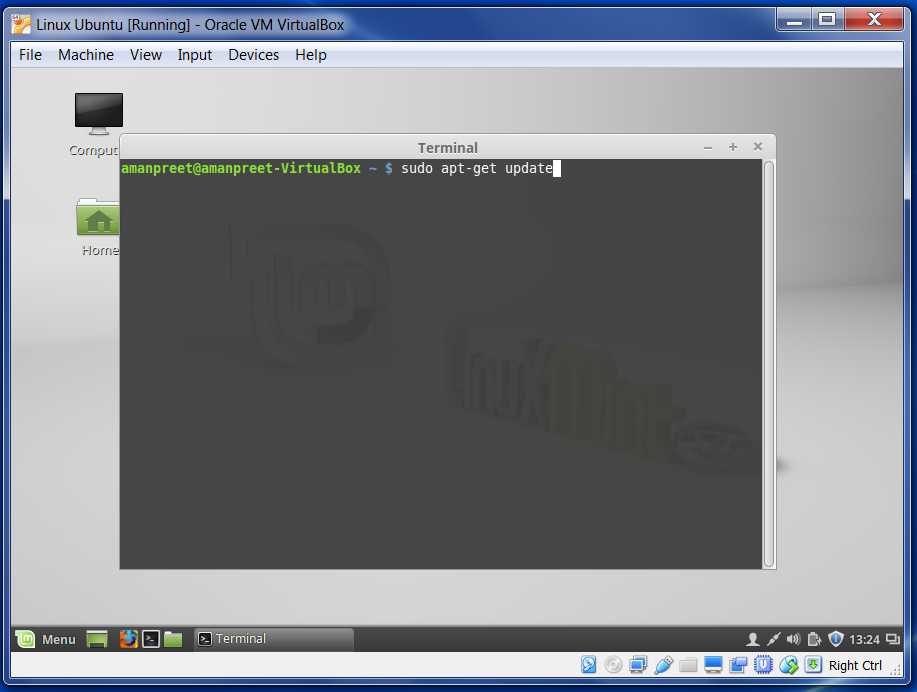
* *Linux Mint 17.2 (based on Ubuntu 14.04)* is installed on *Oracle VM VirtualBox* running in *Windows OS*



### Installing Software

* First step is to update the packages. This should be the first step since the software we are going to install may have some unmet dependencies. An update will resolve any probable issue. Run below command in the Terminal:

*sudo apt-get update*

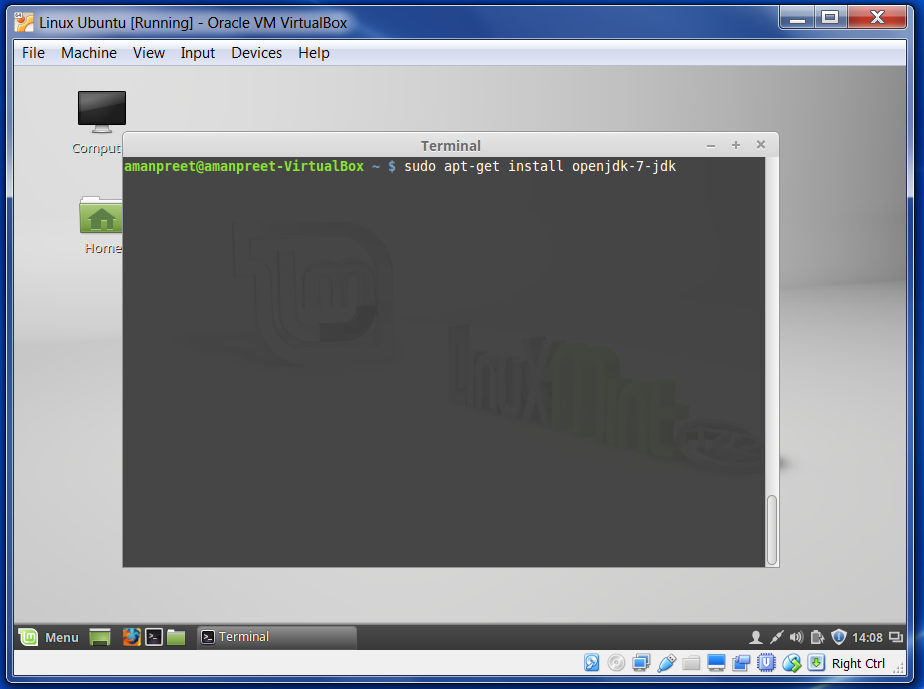


* Install JDK

JDK is mandatory to run Hadoop since both masters and slaves utilize it for processing. Run below command in the Terminal:

*sudo apt-get install openjdk-7-jdk*

After installation, you can check this location */usr/lib/jvm/java-7-openjdk-amd64*

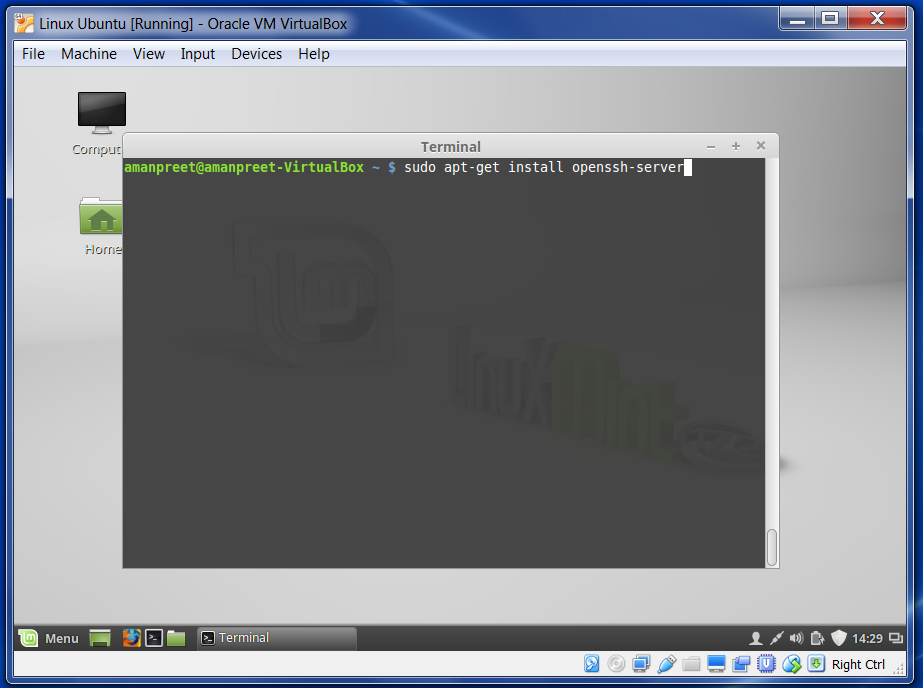


* Install & configure OpenSSH

OpenSSH is used for remotely controlling and transferring files between computers. OpenSSH is secure and encrypted. So in our case OpenSSH is required for secure communication between *Name Node, Data Node, Resource Manager, Node Manager*.

We need to install OpenSSH using below command in Terminal:

*sudo apt-get install openssh-server*

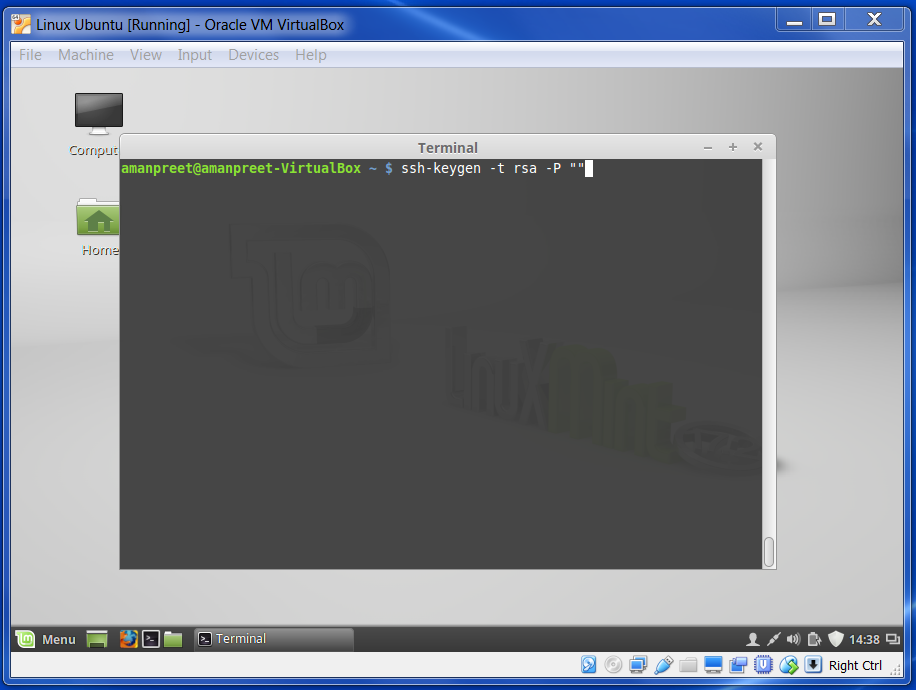


* + SSH Keys

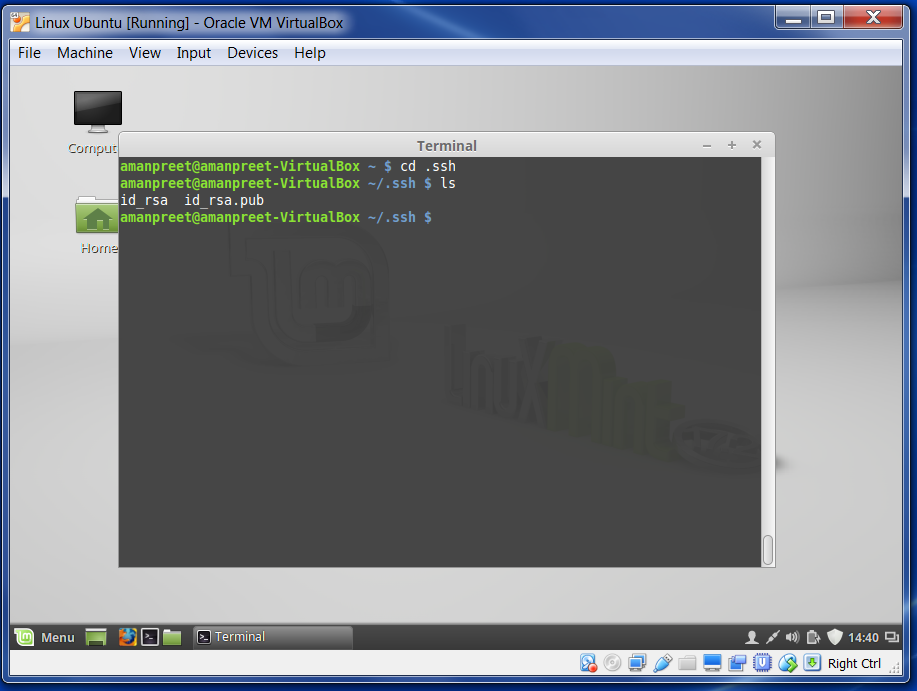
SSH keys allow authentication between two hosts without the need of a password. SSH key authentication uses two keys, a private key and a public key.

* + - Please generate the keys (using RSA Algorithm) from the Terminal using below command:

*ssh-keygen -t rsa -P ““*



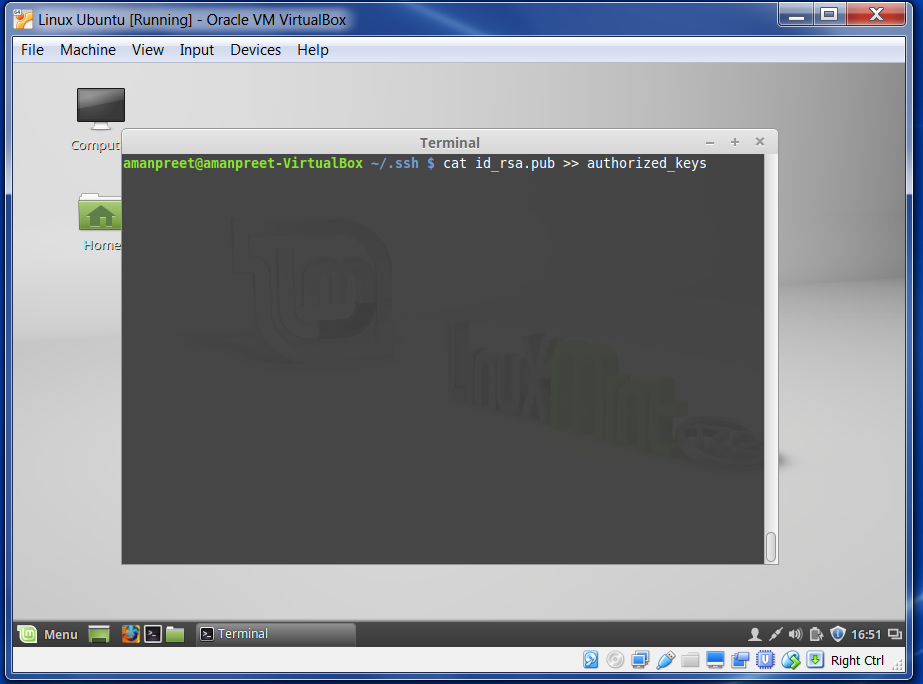
Public key is saved in the file *~/.ssh/id\_rsa.pub* and private key in the file *~/.ssh/id\_rsa*. You can check the generated ssh key here:



Once the key is generated, we need to configure the keys for a password-less secured connection.

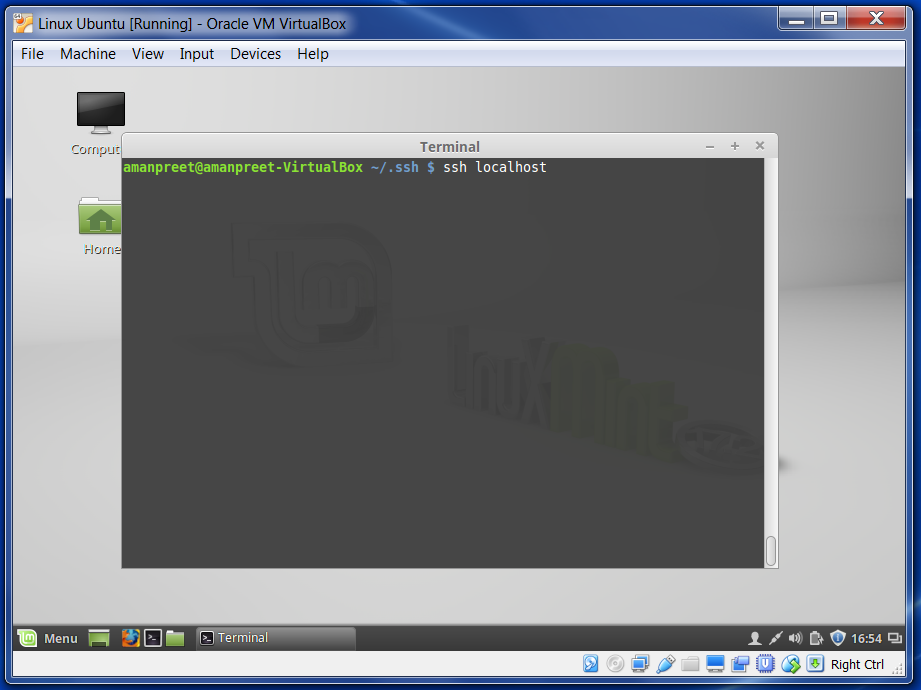
* + Next is to move the trusted public key to *authorized \_keys* under *.ssh* directory. Please enter below command in Terminal:

*cat id\_rsa.pub >> authorized\_keys*

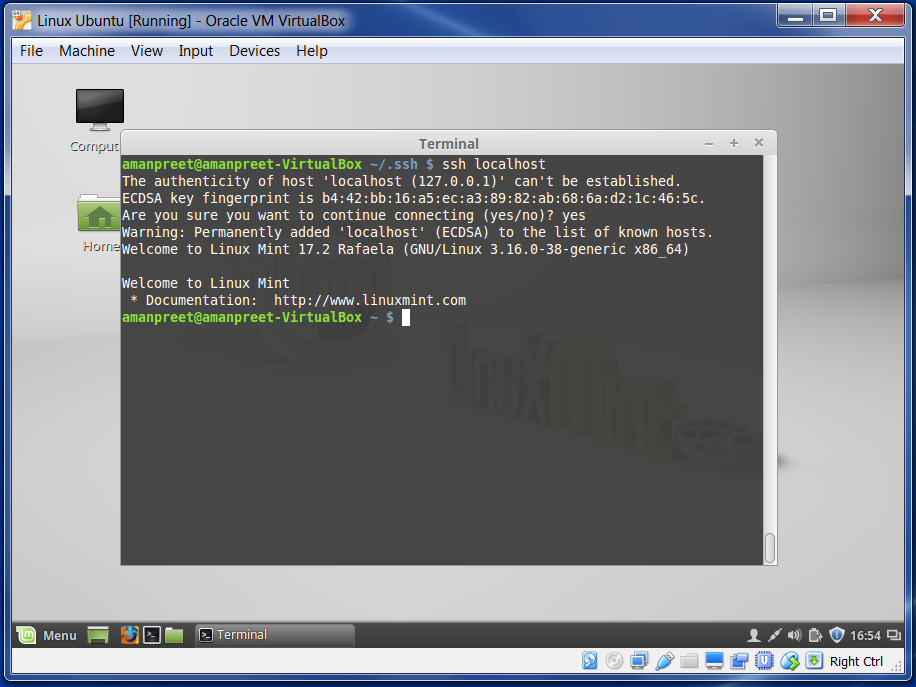


* + Below command will add *localhost* to the list of trusted known hosts. Right now we have only one machine (*localhost*), so run below command to put *localhost* as the known host:

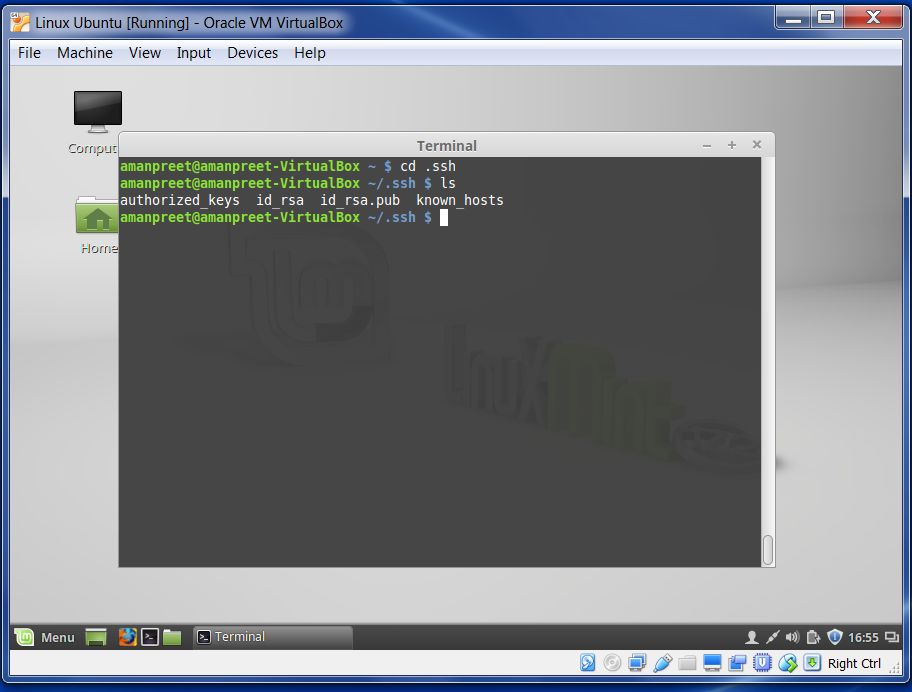
*ssh localhost*



We get below output after running the command.



If we check the contents of *~/.ssh*, we get *known\_hosts* that lists all the trusted known hosts. See below:

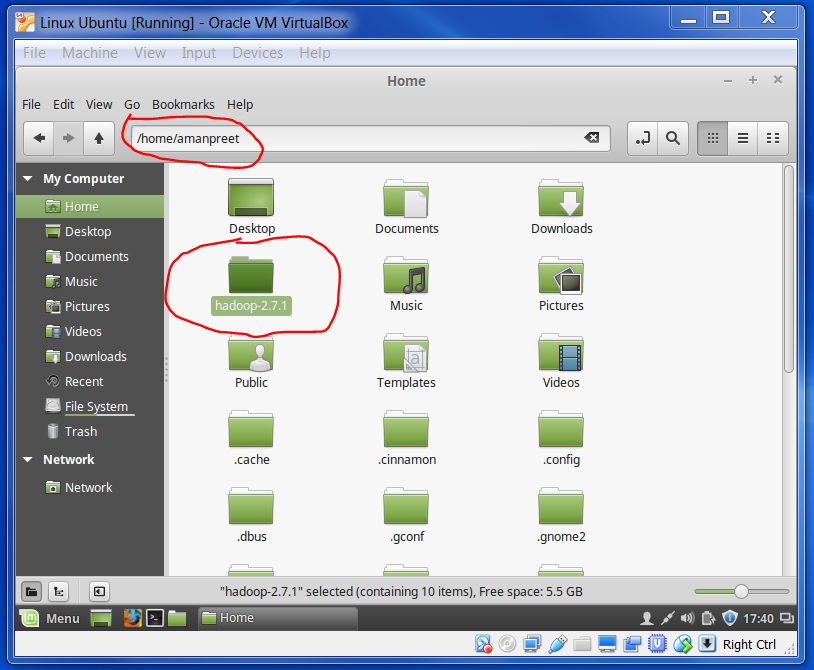


* + Now ssh is successfully configured.

## Hadoop Configuration for Single-Node Installation

### Hadoop Configuration

* Download Hadoop from <http://hadoop.apache.org/releases.html>. For this guide I am using *hadoop-2.7.1.tar*
* Extract that setup in your home directory. After this is done, below steps are required for configuring Hadoop.



* In the extracted folder *hadoop-2.7.1*, go to *etc > Hadoop*
* Edit *core-site.xml* file and put below configuration. This file contains the *Name Node* configuration like the address and port it runs on.

*<configuration>*

*<property>*

*<name>fs.defaultFS</name>*

*<value>hdfs://localhost:9000</value>*

*</property>*

*</configuration>*

* Edit *hdfs-site.xml* and put below configuration.
  + We mention the Replication value here. We are setting this value to 1 (default is 3).
  + Please create two directories *namenode* and *datanode* under */home/amanpreet/hadoop-2.7.1*
  + We also mention the Name Node and Data Node directories.
    - Name Node directory: stores metadata
    - Data Node directory: stores actual blocks

*<configuration>*

*<property>*

*<name>dfs.replication</name>*

*<value>1</value>*

*</property>*

*<property>*

*<name>dfs.namenode.name.dir</name>*

*<value>/home/amanpreet/hadoop-2.7.1/namenode</value>*

*</property>*

*<property>*

*<name>dfs.datanode.data.dir</name>*

*<value>/home/amanpreet/hadoop-2.7.1/datanode</value>*

*</property>*

*</configuration>*

* Check *slaves.xml* file. Currently it has *localhost* in it. When there are multiple machines, then all machine names are put here.
* Edit *yarn-site.xml* file. Here we put configuration like what auxiliary services are required to be run by Node Manager. This is the shuffle service that needs to be set for MapReduce to run.

*<configuration>*

*<property>*

*<name>yarn.nodemanager.aux-services</name>*

*<value>mapreduce\_shuffle</value>*

*</property>*

*</configuration>*

* Create *mapred-site.xml* from *mapred-site.xml.template*. This contains the name of the MapReduce framework we are using. As we are using Hadoop 2, we will put YARN (Yet Another Resource Negotiator) here.

*<configuration>*

*<property>*

*<name>mapreduce.framework.name</name>*

*<value>yarn</value>*

*</property>*

*</configuration>*

* We will update the global *.bashrc* file to add the environment variables. Go to your home directory from Terminal. Add below lines in the *.bashrc* file.

export HADOOP\_INSTALL=/home/amanpreet/hadoop-2.7.1

export JAVA\_HOME=/usr/lib/jvm/java-7-openjdk-amd64

export PATH=$PATH:$JAVA\_HOME/bin:$HADOOP\_INSTALL/bin

* + export HADOOP\_INSTALL=/home/amanpreet/hadoop-2.7.1
    - this is the Hadoop directory
  + export JAVA\_HOME=/usr/lib/jvm/java-7-openjdk-amd64
    - this is the jdk directory
* Now format the *namenode*. We need to format *namenode* only once using below command.

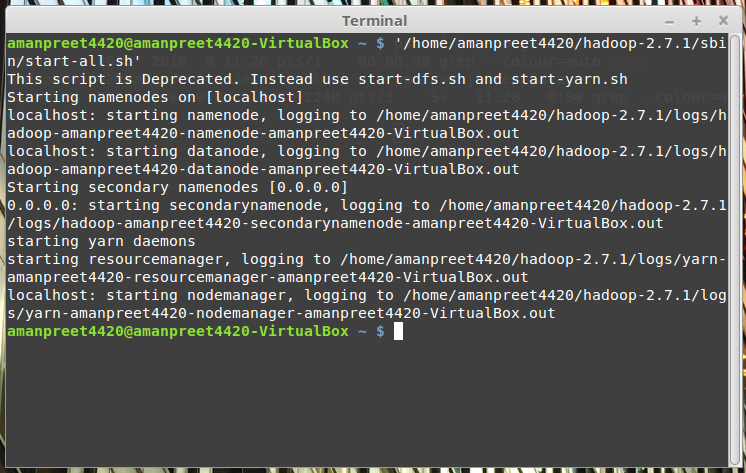
*hdfs namenode –format*

After this command, you can check the directory */home/amanpreet/hadoop-2.7.1/namenode* and you will find files in it. This means *namenode* is formatted using HDFS.

* Hadoop is configured successfully.

### Testing and Storing Data

* Once the *namenode* is formatted, we can start the services from */home/amanpreet/hadoop-2.7.1/sbin/start-all.sh*



* Create a directory in HDFS to store data using below command

*hadoop fs –mkdir /data*

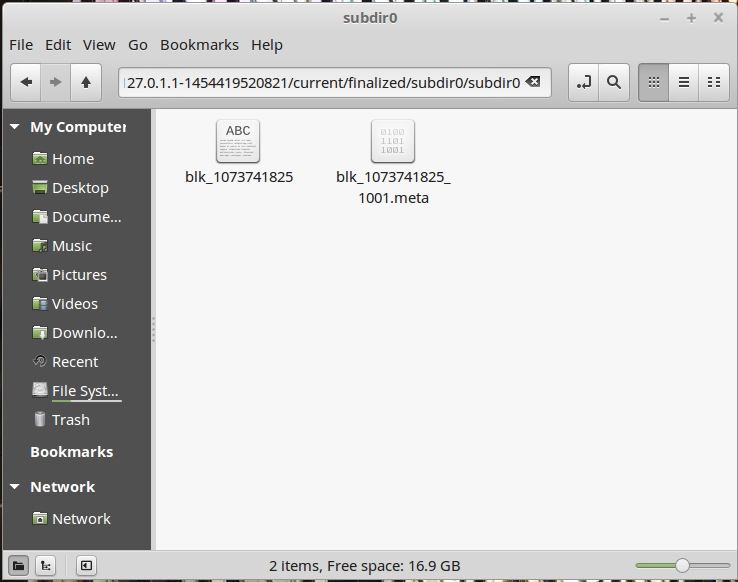
* Create a file (*firstfile*) under */home/amanpreet/data.* We will store this file in HDFS now
* Now put this *firstfile* in HDFS using below command

*hadoop fs -put /home/amanpreet/data/firstfile /data*

You can now check *datanode* directory and it will contain two files. Location on my machine is

*/home/amanpreet/hadoop-2.7.1/datanode/current/BP-2072242420-127.0.1.1-1454419520821/current/finalized/subdir0/subdir0*

* + blk\_1073741825: this is the actual block where data is stored
  + blk\_1073741825\_1001.meta: this contains the checksum information



* Data is successfully written on HDFS
* After your testing, you can stop the services from */home/amanpreet/hadoop-2.7.1/sbin/stop-all.sh*

### Browser View

You can also use the web portal of Hadoop. Open browser and go to <http://localhost:50070> and you will see below interface.

