

**Hadoop Installation Document**

**Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Version** | **Release Date** | **Author** | **Reviewer** | **Remarks** |
| 1 | 1.0 | 22-04-15 | Dikshit Rajkhowa |  | First Version |
| 2 | 2.0 | 24-05-2015 | Dikshit Rajkhowa |  | Second Version |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

[Prerequisites and Assumptions 4](#_Toc417487448)

[Installation Steps 5](#_Toc417487449)

[1. Install Java 5](#_Toc417487450)

[2. Create and Setup SSH Certificates 5](#_Toc417487451)

[3. Install SSH Server and Client 6](#_Toc417487452)

[4. Fetch and Install Hadoop 6](#_Toc417487453)

[5. Edit and Setup Configuration Files 6](#_Toc417487454)

[i. Editing ~/.bashrc 7](#_Toc417487455)

[ii. Edit files under /home/bootcamp/software/hadoop-1.2.1/conf/ 7](#_Toc417487456)

1. [hadoop-env.sh 8](#_Toc417487457)
2. [core-site.xml 8](#_Toc417487458)
3. [hdfs-site.xml 8](#_Toc417487459)
4. [mapred-site.xml 9](#_Toc417487460)

[6. Setting Datanode directory file permissions 9](#_Toc417487461)

[7. Format the name node: 10](#_Toc417487462)

[8. Start Hadoop daemons 10](#_Toc417487463)

The steps given below are for installing Hadoop 1.2.1 in a pseudo distributed mode in which the Hadoop daemons run on a local machine, thus simulating a cluster on a small scale. Different Hadoop daemons run in different JVM instances, but on a single machine. HDFS is used instead of local FS.

# Prerequisites and Assumptions

1. Linux OS - For this document we are assuming Ubuntu 14.04 version running independently or on top of Windows through VMPlayer
2. User – It is assumed that there is a user created with following credentials

User Name – hadoop

Password – hadoop

1. Home Directory - It is assumed that the user’s home directory will be

“/home/hadoop”. If you are using different username then change your home directory accordingly.

# Installation Steps

## Install Java

Hadoop requires Java to be installed, so let's begin by installing Java. Execute the following commands from the terminal

sudo apt‐get update

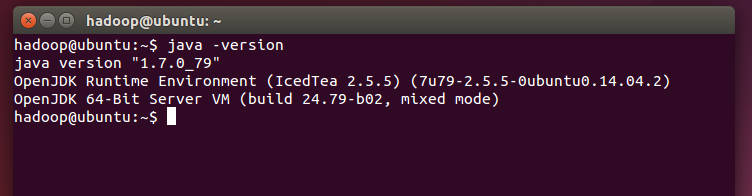
sudo apt‐get install default‐jdk

These commands will update the package information on your system and then install Java.

After executing these commands, execute the following command to verify that Java has been installed:

java ‐version

If Java has been installed, this should display the version details as illustrated in the following image:



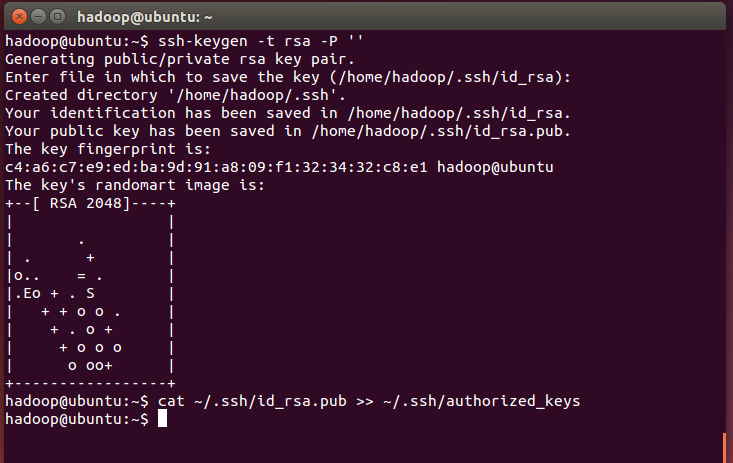
## Create and Setup SSH Certificates

Hadoop uses SSH (to access its nodes) which would normally require the user to enter a password. However, this requirement can be eliminated by creating and setting up SSH certificates using the following commands:

ssh‐keygen ‐t rsa ‐P ''

cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

After executing the first of these two commands, you might be asked for a filename. Just leave it blank and press the enter key to continue. The second command adds the newly created key to the list of authorized keys so that Hadoop can use SSH without prompting for a password.



## Install SSH Server and Client

As mentioned above Hadoop uses SSH to access its nodes and since we are installing it in pseudo distributed mode ( Name node and data nodes in the same machine ) we need to install both SSH Server and SSH Client.

Execute the below commands for installing SSH Server and Client

sudo apt-get install openssh-server

sudo apt-get install openssh-client

## Fetch and Install Hadoop

Make present working directory as home directory “/home/hadoop”.

First let's fetch Hadoop from one of the mirrors using the following command:

wget <http://mirrors.advancedhosters.com/apache/hadoop/common/hadoop-1.2.1/hadoop-1.2.1.tar.gz>

Note :- You can use a different mirror to download

Assuming that we executed the command from our home directory “/home/hadoop” , this will download the file hadoop‐1.2.1-bin.tar.gz in the home directory

[In the virtual machine that is shared, this is already downloaded and present in the home directory i.e /home/hadoop, so you can skip the downloading step]

After downloading the Hadoop package, execute the following command to extract it:

tar xfz hadoop‐1.2.1-bin.tar.gz

This command will extract all the files in this package in a directory named hadoop‐1.2.1

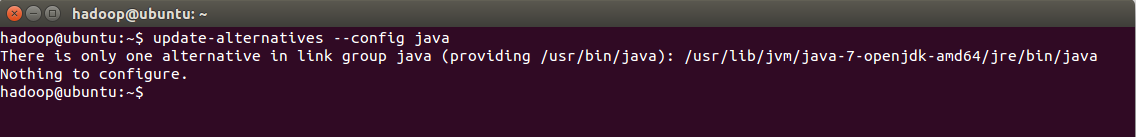
## Edit and Setup Configuration Files

### Editing ~/.bashrc

Before editing the .bashrc file in your home directory, we need to find the path where Java has been installed to set the JAVA\_HOME environment variable. Let's use the following command to do that:

update‐alternatives ‐‐config java

This will display something like the following:



The complete path displayed by this command is:

/usr/lib/jvm/java‐7‐openjdk‐amd64/jre/bin/java

Or you can type following command if “java –version” is giving correct output:

readlink -f $(which java)

The value for JAVA\_HOME is everything before /jre/bin/java in the above path - in this case, /usr/lib/jvm/java‐7‐openjdk‐amd64

Now use gedit (or your favourite editor) to edit ~/.bashrc using the following command:

sudo gedit ~/.bashrc

Edit the .bashrc file to add the following lines.

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

export HADOOP\_INSTALL=/home/hadoop/hadoop-1.2.1

export HADOOP\_HOME=/home/hadoop/hadoop-1.2.1

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

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

**Note :** If the value of JAVA\_HOME or Hadoop installation directory is different on your system, make sure to alter the export statements in the above content accordingly.

After saving and closing the .bashrc file, execute the following command so that your system recognizes the newly created environment variables:

source ~/.bashrc

Putting the above content in the .bashrc file ensures that these variables are always available when your system starts up.

### **Edit files under <Hadoop-installation-path>/hadoop-1.2.1/conf/**

#### **hadoop-env.sh**

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

#### **core-site.xml**

<configuration>

<property>

<name>fs.default.name</name>

<value>hdfs://localhost:8020</value>

<final>true</final>

</property>

</configuration>

#### **hdfs-site.xml**

Before editing the hdfs-site.xml file we need to create some directories in our local file system using the following commands

mkdir -p /home/hadoop/hdfs/namenode

mkdir -p /home/hadoop/hdfs/data

mkdir -p /home/hadoop/hdfs/checkpoint

This will create a directory “hdfs” under home directory and three sub directories namely – namenode, data and checkpoint inside the hdfs directory

Now edit the hdfs-site.xml file and enter the following lines

<configuration>

<property>

<name>dfs.name.dir</name>

<value>/home/hadoop/hdfs/namenode</value>

<final>true</final>

</property>

<property>

<name>fs.checkpoint.dir</name>

<value>/home/hadoop/hdfs/checkpoint</value>

<final>true</final>

</property>

<property>

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

<value>/home/hadoop/hdfs/data</value>

<final>true</final>

</property>

</configuration>

#### **mapred-site.xml**

Before editing the file we need to create a directory in our local file system by executing the following command

mkdir /home/hadoop/mapred

Edit the mapred-site.xml and enter the following details

<configuration>

<property>

<name>mapred.job.tracker</name>

<value>localhost:8021</value>

<final>true</final>

</property>

<property>

<name>mapred.local.dir</name>

<value>/home/hadoop/mapred</value>

<final>true</final>

</property>

<property>

<name>mapred.system.dir</name>

<value>/home/hadoop/mapred</value>

<final>true</final>

</property>

<property>

<name>mapred.tasktracker.map.tasks.Maximum</name>

<value>3</value>

<final>true</final>

</property>

<property>

<name>mapred.tasktracker.reduce.tasks.maximum</name>

<value>3</value>

<final>true</final>

</property>

<property>

<name>mapred.child.java.opts</name>

<value>-Xmx400m</value>

</property>

</configuration>

## [Setting Datanode directory file permissions](http://supunk.blogspot.com/2013/04/starting-hadoop-setting-datanode.html)

Set directory permission of data node directory as follows

sudo chmod 755 /home/hadoop/hdfs/data

## Format the name node:

After completing all the configuration outlined in the above steps, the Hadoop file system needs to be formatted so that it can start being used. This is done by executing the following command:

hadoop namenode ‐format

*Note: This only needs to be done once before you start using Hadoop. If this command is executed again after Hadoop has been used, it'll destroy all the data on the Hadoop file system.*

## Start Hadoop daemons

Start hadoop by executing the following command

start-all.sh

Once it starts successfully the following processes should run in your system.

SecondaryNameNode

DataNode

JobTracker

TaskTracker

NameNode

This can be verified by executing the JPS command as shown below

