How to Install Hadoop on Ubuntu 13.10

PUBLISHED: MAR 20, 2014 • UPDATED: MAY 30, 2014
SUBSCRIBE TAGGED IN: MISCELLANEOUS, JAVA • DIFFICULTY: ADVANCED

Prerequisites

The only prerequisite for this tutorial is a VPS with **Ubuntu 13.10 x64** installed.

You will need to execute commands from the command line which you can do in one of the two ways:

1. Use SSH to access the droplet.
2. Use the 'Console Access' from the Digital Ocean Droplet Management Panel

What is Hadoop?

Hadoop is a framework (consisting of software libraries) which simplifies the processing of data sets distributed across clusters of servers. Two of the main components of Hadoop are **HDFS** and **MapReduce**.

HDFS is the filesystem that is used by Hadoop to store all the data on. This file system spans across all the nodes that are being used by Hadoop. These nodes could be on a single VPS or they can be spread across a large number of virtual servers.

MapReduce is the framework that orchestrates all of Hadoop's activities. It handles the assignment of work to different nodes in the cluster.

Benefits of using Hadoop

The architecture of Hadoop allows you to scale your hardware as and when you need to. New nodes can be added incrementally without having to worry about the change in data

formats or the handling of applications that sit on the file system.

One of the most important features of Hadoop is that it allows you to save enormous amounts of money by substituting cheap commodity servers for expensive ones. This is possible because Hadoop transfers the responsibility of fault tolerance from the hardware layer to the application layer.

Installing Hadoop

Installing and getting Hadoop up and running is quite straightforward. However, since this process requires editing multiple configuration and setup files, make sure that each step is properly followed.

1. Install Java

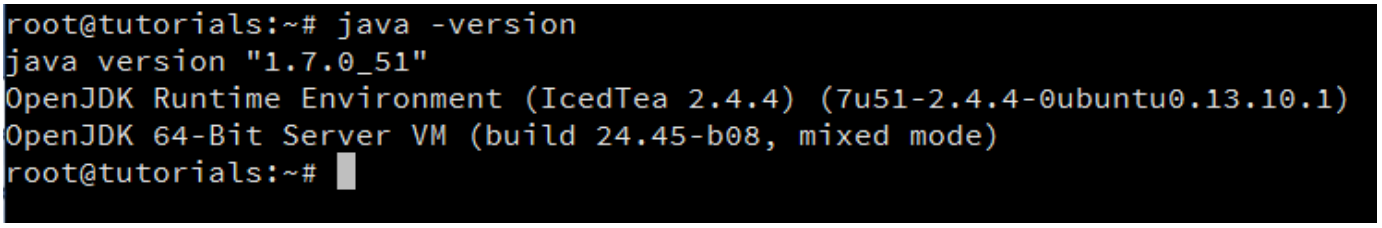
Hadoop requires Java to be installed, so let's begin by installing Java:

```
apt-get update
apt-get install default-jdk
```

These commands will update the package information on your VPS 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:



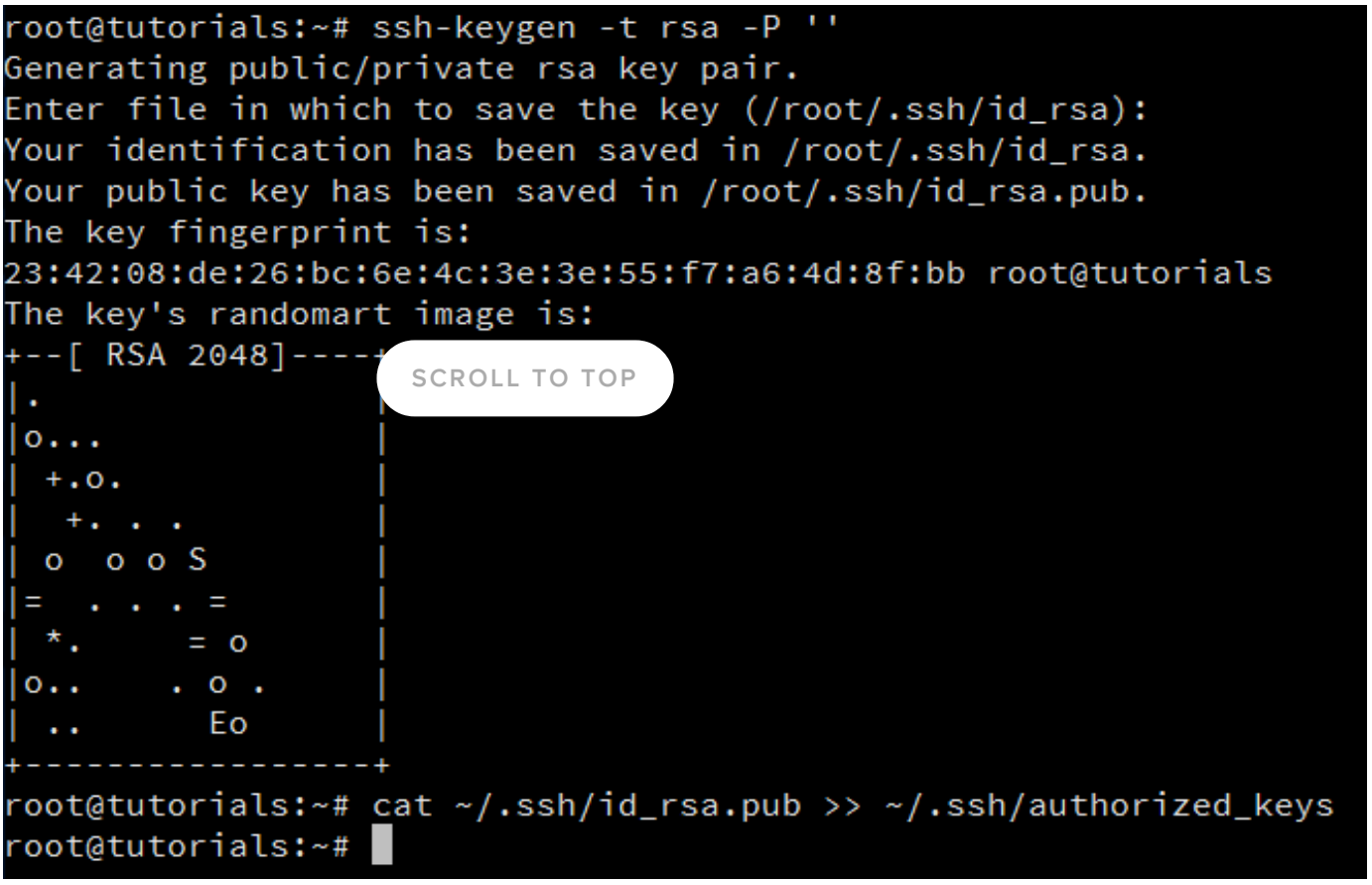
2. Create and Setup SSH Certificates

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.



3. Fetch and Install Hadoop

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

```
wget http://www.motorlogy.com/apache/hadoop/common/current/hadoop-2.3.0.tar.gz
```

Note: *This command uses a download a link on one of the mirrors listed on the Hadoop website. The list of mirrors can be found on this link. You can choose any other mirror if you*

want to. To download the latest stable version, choose the *hadoop-X.Y.Z.tar.gz** file from the **current** or the **current2** directory on your chosen mirror.*

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

```
tar xfz hadoop-2.3.0.tar.gz
```

This command will extract all the files in this package in a directory named `hadoop-2.3.0`. For this tutorial, the Hadoop installation will be moved to the `/usr/local/hadoop` directory using the following command:

```
mv hadoop-2.3.0 /usr/local/hadoop
```

Note: *The name of the extracted folder depends on the Hadoop version your have downloaded and extracted. If your version differs from the one used in this tutorial, change the above command accordingly.*

4. Edit and Setup Configuration Files

To complete the setup of Hadoop, the following files will have to be modified:

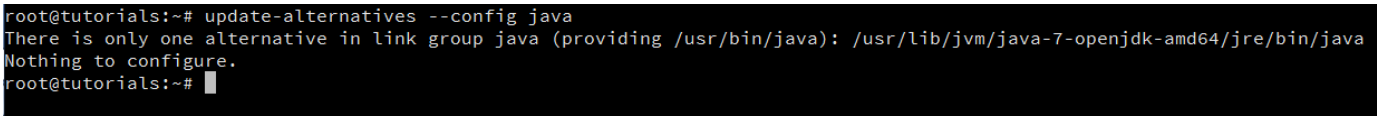
- ~/.bashrc
- /usr/local/hadoop/etc/hadoop/hadoop-env.sh
- /usr/local/hadoop/etc/hadoop/core-site.xml
- /usr/local/hadoop/etc/hadoop/yarn-site.xml
- /usr/local/hadoop/etc/hadoop/mapred-site.xml.template
- /usr/local/hadoop/etc/hadoop/hdfs-site.xml

i. 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
```

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` . Make a note of this as we'll be using this value in this step and in one other step.

Now use `nano` (or your favored editor) to edit ~/.bashrc using the following command:

```
nano ~/.bashrc
```

This will open the `.bashrc` file in a text editor. Go to the end of the file and paste/type the following content in it:

```
#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END
```

Note 1: *If the value of `JAVA_HOME` is different on your VPS, make sure to alter the first `export` statement in the above content accordingly.*

Note 2: *File*

Note 2: *Files opened and edited using nano can be saved using `Ctrl + X`. Upon the prompt to save changes, type `Y`. If you are asked for a filename, just press the enter key.*

The end of the `.bashrc` file should look something like this:

```
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
#if [ -f /etc/bash_completion ] && ! shopt -oq posix; then
#    . /etc/bash_completion
#fi

#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

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 VPS starts up.

ii. Editing /usr/local/hadoop/etc/hadoop/hadoop-env.sh

Open the `/usr/local/hadoop/etc/hadoop/hadoop-env.sh` file with nano using the following command:

```
nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
```

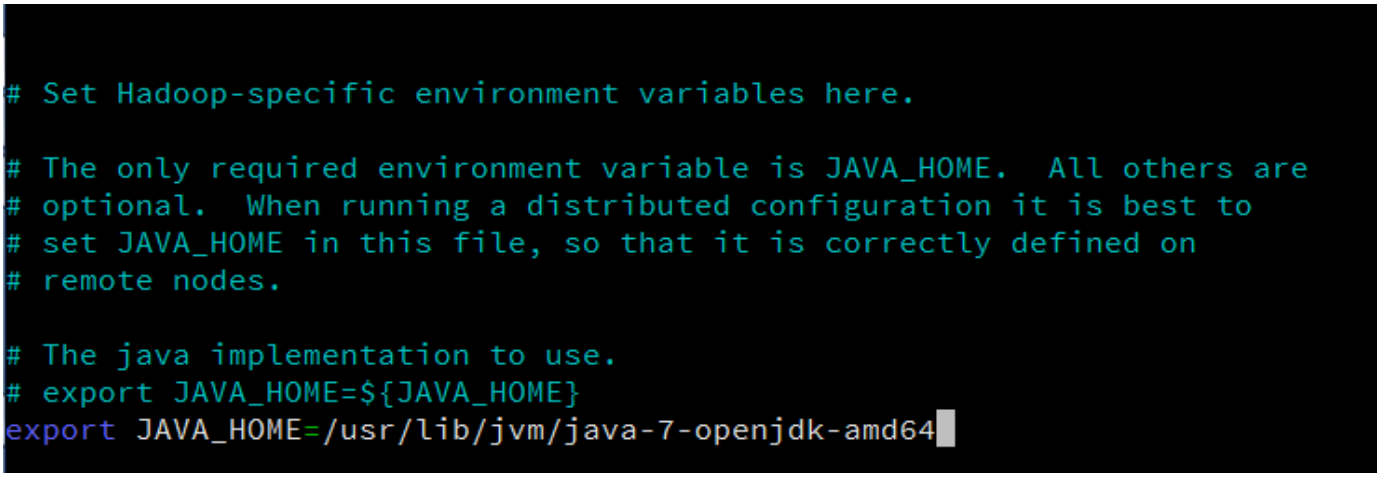
In this file, locate the line that exports the `JAVA_HOME` variable. Change this line to the

following:

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
```

Note: *If the value of `JAVA_HOME` is different on your VPS, make sure to alter this line accordingly.*

The `hadoop-env.sh` file should look something like this:



Save and close this file. Adding the above statement in the `hadoop-env.sh` file ensures that the value of `JAVA_HOME` variable will be available to Hadoop whenever it is started up.

iii. Editing /usr/local/hadoop/etc/hadoop/core-site.xml

The `/usr/local/hadoop/etc/hadoop/core-site.xml` file contains configuration properties that Hadoop uses when starting up. This file can be used to override the default settings that Hadoop starts with.

Open this file with nano using the following command:

```
nano /usr/local/hadoop/etc/hadoop/core-site.xml
```

In this file, enter the following content in between the `<configuration></configuration>` tag:

```
<property>
```

```
    <name>fs.default.name</name>
    <value>hdfs://localhost:9000</value>
  </property>
```

The `core-site.xml` file should look something like this:

```
<?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://localhost:9000</value>
        </property>
</configuration>
```

Save and close this file.

iv. Editing /usr/local/hadoop/etc/hadoop/yarn-site.xml

The `/usr/local/hadoop/etc/hadoop/yarn-site.xml` file contains configuration properties that MapReduce uses when starting up. This file can be used to override the default settings that MapReduce starts with.

Open this file with nano using the following command:

```
nano /usr/local/hadoop/etc/hadoop/yarn-site.xml
```

In this file, enter the following content in between the `<configuration></configuration>` tag:

tag.

```
  <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>
```

The `yarn-site.xml` file should look something like this:

```
<?xml version="1.0"?>
<!--
  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.
-->
<configuration>

        <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>

</configuration>
```

Save and close this file.

v. Creating and Editing /usr/local/hadoop/etc/hadoop/mapred-site.xml

By default, the `/usr/local/hadoop/etc/hadoop/` folder contains the `/usr/local/hadoop/etc/hadoop/mapred-site.xml.template` file which has to be

renamed/copied with the name `mapred-site.xml` . This file is used to specify which framework is being used for MapReduce.

This can be done using the following command:

```
cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hac
```

Once this is done, open the newly created file with nano using the following command:

```
nano /usr/local/hadoop/etc/hadoop/mapred-site.xml
```

In this file, enter the following content in between the `<configuration></configuration>` tag:

```
<property>
   <name>mapreduce.framework.name</name>
   <value>yarn</value>
</property>
```

The `mapred-site.xml` file should look something like this:

```
<?xml version="1.0"?>
<?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>mapreduce.framework.name</name>
           <value>yarn</value>
        </property>
```

```
</configuration>
```

Save and close this file.

vi. Editing /usr/local/hadoop/etc/hadoop/hdfs-site.xml

The `/usr/local/hadoop/etc/hadoop/hdfs-site.xml` has to be configured for each host in the cluster that is being used. It is used to specify the directories which will be used as the **namenode** and the **datanode** on that host.

Before editing this file, we need to create two directories which will contain the **namenode** and the **datanode** for this Hadoop installation. This can be done using the following commands:

```
mkdir -p /usr/local/hadoop_store/hdfs/namenode
mkdir -p /usr/local/hadoop_store/hdfs/datanode
```

Note: *You can create these directories in different locations, but make sure to modify the contents of `hdfs-site.xml` accordingly.*

Once this is done, open the `/usr/local/hadoop/etc/hadoop/hdfs-site.xml` file with nano using the following command:

```
nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml
```

In this file, enter the following content in between the `<configuration></configuration>` tag:

```
<property>
   <name>dfs.replication</name>
   <value>1</value>
 </property>
 <property>
   <name>dfs.namenode.name.dir</name>
   <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
 </property>
 <property>
```

```
    <name>dfs.datanode.data.dir</name>
    <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
  </property>
```

The `hdfs-site.xml` file should look something like this:

```
<?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>dfs.replication</name>
          <value>1</value>
         </property>
         <property>
          <name>dfs.namenode.name.dir</name>
          <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
         </property>
         <property>
          <name>dfs.datanode.data.dir</name>
          <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
         </property>
</configuration>
```

Save and close this file.

Format the New Hadoop Filesystem

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

```
hdfs 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

All that remains to be done is starting the newly installed single node cluster:

```
start-dfs.sh
```

While executing this command, you'll be prompted twice with a message similar to the following:

> Are you sure you want to continue connecting (yes/no)?

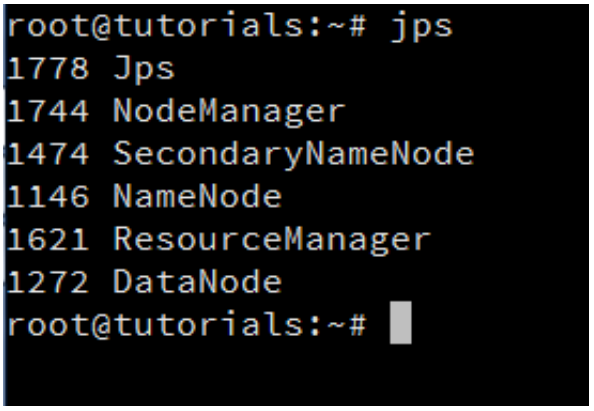
Type in `yes` for both these prompts and press the enter key. Once this is done, execute the following command:

```
start-yarn.sh
```

Executing the above two commands will get Hadoop up and running. You can verify this by typing in the following command:

```
jps
```

Executing this command should show you something similar to the following:



If you can see a result similar to the depicted in the screenshot above, it means that you now have a functional instance of Hadoop running on your VPS.

Next Steps

If you have an application that is set up to use Hadoop, you can fire that up and start using it with the new installation. On the other hand, if you're just playing around and exploring Hadoop, you can start by adding/manipulating data or files on the new filesystem to get a feel for it.

Tagged In: Miscellaneous, Java

Submitted by: Jay

Written By:
Jay Martinez

Do you find this post helpful?
Show the author by Hearting it!

Related Tutorials

How To Set Up an OSRM Server on Ubuntu 14.04

How To Install and Configure Sphinx on Ubuntu 14.04

Как использовать SFTP для безопасного обмена файлами с удаленным сервером

How To Sync Calendars and Contacts Using CardDAV and CalDAV Standards with Baïkal on Ubuntu ...

How To Recover from File System Corruption Using FreeBSD's Single-User Mode

80 Comments

You must be logged in to comment. Log In

Leave a comment...

SUBMIT COMMENT >

Notify me of replies to my comment

cream_craker *March 25, 2014*

♡ Thx man, this tutorial saved my life

byoigres *March 29, 2014*

♡ I will try it, thanks.

ed.rabelo *March 31, 2014*

♡ This was the best tutorial of hadoop installation i've ever saw, you teach very well!! thanks!

joaoluizgg *April 2, 2014*

♡ I had some trouble when running 'start-dfs.sh'. All the other steps I made exactly as they appear in the tutorial.

xxx@ubuntu:/usr/local/hadoop/sbin$ start-dfs.sh
/usr/local/hadoop/sbin/start-dfs.sh: line 55: /home/xxx/hadoop/bin/hdfs: No such file or directory
Starting namenodes on []
/usr/local/hadoop/sbin/start-dfs.sh: line 60: /home/xxx/hadoop/sbin/hadoop-daemons.sh: No such file or directory

daemons.sh: No such file or directory
/usr/local/hadoop/sbin/start-dfs.sh: line 73: /home/xxx/hadoop/sbin/hadoop-daemons.sh: No such file or directory
/usr/local/hadoop/sbin/start-dfs.sh: line 108: /home/xxx/hadoop/bin/hdfs: No such file or directory

Any ideas?

a.starr.b *April 3, 2014*

♡ @joaoluizgg

Your HADOOP_INSTALL environmental variable seems to be pointing to you home directory not /usr/local/

Double check the variables that you added to your ~/.basrc file in step 4. Also make sure that you ran "source ~/.bashrc" after adding them.

jaymartinez1180 *April 12, 2014*

♡ @cream_craker, @ed.rabelo:
glad that you liked this tutorial :)

@joaoluizgg:
@a.starr.b is spot on about where the problem might be.

veach.emily *April 15, 2014*

♡ Hi,
Thanks for the tutorials. Small typo here:
One of the most important features of Hadoop is that it allows you to **use save** enormous amounts of money by substituting cheap commodity servers for expensive ones.

asb MOD *April 15, 2014*

♡ @veach.emily: Thanks for catching that. Fixed!

no.andrey *April 24, 2014*

♡ You guys have the best tutorials on the net! Tried 3 others and got annoying erros -

and no errors at all using this one! Perfect! Keep going!

r600041 *April 30, 2014*

Thank you very much for this very useful tutorial.
I have followed your instructions and installed hadoop in a debian 7.5 in virtualbox setup. with Java sdk 1.7.
When I startup hadoop i get the following warning.
14/04/30 08:36:50 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

Also I get the following warning for the secondary namenode

The authenticity of host '0.0.0.0 (0.0.0.0)' can't be established.
0.0.0.0: Warning: Permanently added '0.0.0.0' (ECDSA) to the list of known hosts.
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-xxxxx-secondarynamenode-debian7.out

r600041 *April 30, 2014*

java -version
java version "1.7.0_51"
OpenJDK Runtime Environment (IcedTea 2.4.6) (7u51-2.4.6-1)
OpenJDK Client VM (build 24.51-b03, mixed mode, sharing)

ls -lt /usr/lib/jvm
total 4
drwxr-xr-x 7 root root 4096 Apr 29 19:06 java-7-openjdk-i386
lrwxrwxrwx 1 root root 19 Apr 1 21:09 java-1.7.0-openjdk-i386 -> java-7-openjdk-i386
lrwxrwxrwx 1 root root 23 Feb 1 07:23 default-java -> java-1.7.0-openjdk-i386

echo $JAVA_HOME
/usr/lib/jvm/java-7-openjdk-i386

r600041 *April 30, 2014*

echo $HADOOP_INSTALL
/usr/local/hadoop

/usr/local/hadoop/lib/native
has the following
lrwxrwxrwx 1 xxxxxx xxxxxx 18 Mar 31 05:05 libhadoop.so -> libhadoop.so.1.0.0
lrwxrwxrwx 1 xxxxxx xxxxxx 16 Mar 31 05:05 libhdfs.so -> libhdfs.so.0.0.0
-rw-r--r-- 1 xxxxxx xxxxxx 534024 Mar 31 04:49 libhadooppipes.a
-rw-r--r-- 1 xxxxxx xxxxxx 226360 Mar 31 04:49 libhadooputils.a
-rw-r--r-- 1 xxxxxx xxxxxx 204586 Mar 31 04:49 libhdfs.a
-rwxr-xr-x 1 xxxxxx xxxxxx 167760 Mar 31 04:49 libhdfs.so.0.0.0
-rw-r--r-- 1 xxxxxx xxxxxx 687184 Mar 31 04:49 libhadoop.a
-rwxr-xr-x 1 xxxxxx xxxxxx 488873 Mar 31 04:49 libhadoop.so.1.0.0

Thank you

asb MOD *April 30, 2014*

♡ @r600041: I don't necessarily see anything wrong in your output. That just looks like an informational message. Is hadoop working for you? What's the output of the command "jps"? It shoul look something like:

```
# jps
16261 ResourceManager
16552 Jps
16344 NodeManager
15875 NameNode
16125 SecondaryNameNode
15957 DataNode
```

xmtx1123zm *May 3, 2014*

♡ yeah, the problem r600041 has happened to me too.
14/05/02 22:33:45 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: ssh: connect to host localhost port 22: Connection refused
localhost: ssh: connect to host localhost port 22: Connection refused
Starting secondary namenodes [0.0.0.0]
0.0.0.0: ssh: connect to host 0.0.0.0 port 22: Connection refused
14/05/02 22:33:53 WARN util.NativeCodeLoader: Unable to load native-hadoop

library for your platform... using builtin-java classes where applicable

when I execute the jps command, output like following content...
ubuntu@ubuntu-VirtualBox:~$ jps
4442 Jps

what's the problem, can you tell me how to fix it, I'll do appreciate it.

xmtx1123zm *May 3, 2014*

♡ I have fixed~
Because the ssh server is not installed as only the client is installed by default.~-~

pavankp065 *October 23, 2014*

♡ Hi,

Can you please let me know how did you resolve the above error.

i am facing the same issue with my cluster

Thanks,

anhdo *January 3, 2015*

♡ sudo apt-get install ssh

tsatish14 *May 6, 2014*

♡ How come i don't see ETC directory? Can some one help?

cannot access /usr/local/hadoop/etc: No such file or directory

mkdev *May 17, 2014*

♡ Namenode is not starting for me. Tried changing the port number in core-site.xml. Also, I have the much discussed problem of "WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable".

Any help is appreciated.

mkdev *May 18, 2014*

♡ Formatted namenode and restarted hadoop ..... namenode problem solved but I forgot to mention a problem last time - I do not see task tracker and job tracker in jps :(

piyushramani8 *May 21, 2014*

♡ Hi I am beginner in Hadoop at time of setup I am getting this problem

I am getting following warnning at time of starting of start-dfs.sh

14/05/21 08:36:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

In jps command is not showing the NameNode and DataNode.

In execution of the start-yarn.sh command showing following error:
starting yarn daemons
resourcemanager running as process 15734. Stop it first.
localhost: nodemanager running as process 15871. Stop it first.

gauba.himanshu *May 29, 2014*

♡ when i am giving this terminal command
"mkdir -p /usr/local/hadoop_store/hdfs/namenode"
it is showing me an error by saying that
mkdir : cannot create directory '/usr/local/hadoop_store' :permission denied

Kindly help me.

anhdo *January 3, 2015*

♡ "permission denied" means you don't have permission to create directory there, please use sudo

```
sudo mkdir -p /usr/local/hadoop_store/hdfs/namenode
```

Show More Comments

Proudly Made in NY