Sumit Chawla's Blog

HADOOP

Installing Hadoop on Ubuntu 14.04

JUNE 15, 2014MARCH 9, 2015 SUMITCHAWLA 28 COMMENTS

In this article, I wanted to document my first hand experience of installing Hadoop on Ubuntu 14.04. I am using the Hadoop Stable version 2.2.0 for this article. This article covers a single node installation of Hadoop. If you want to do a multi-node installation, follow my other article here – Install a Multi Node Hadoop Cluster on Ubuntu 14.04 (https://chawlasumit.wordpress.com/2015/03/09/install-a-multi-node-hadoop-cluster-on-ubuntu-14-04/)

Installing Java

```
$ sudo add-apt-repository ppa:webupd8team/java
$ sudo apt-get update
$ sudo apt-get install oracle-java7-installer
# Updata Java runtime
$ sudo update-java-alternatives -s java-7-oracle
```

Disable IPv6

As of now Hadoop does not support IPv6, and is tested to work only on IPv4 networks. If you are using IPv6, you need to switch Hadoop host machines to use IPv4. The Hadoop Wiki (http://wiki.apache.org/hadoop/HadoopIPv6) link provides a one liner command to disable the IPv6. If you are not using IPv6, skip this step:

```
sudo sed -i 's/net.ipv6.bindv6only\ =\ 1/net.ipv6.bindv6only\ =\ 0/' \
/etc/sysctl.d/bindv6only.conf && sudo invoke-rc.d procps restart
```

Setting up a Hadoop User

Hadoop talks to other nodes in the cluster using no-password ssh. By having Hadoop run under a specific user context, it will be easy to distribute the ssh keys around in the hadoop cluster

```
# Create hadoopgroup
$ sudo addgroup hadoopgroup
# Create hadoopuser user
$ sudo adduser –ingroup hadoopgroup hadoopuser
# Login as hadoopuser
$ su - hadoopuser
#Generate a ssh key for the user
$ ssh-keygen -t rsa -P ""
#Authorize the key to enable password less ssh
$ cat /home/hadoopuser/.ssh/id_rsa.pub >> /home/hadoopuser/.ssh/authorized_key
$ chmod 600 authorized_keys
```

Download and Install Hadoop

Pick the best mirror site to download the binaries from Apache Hadoop (http://www.apache.org/dyn/closer.cgi/hadoop/core/), and download the stable/hadoop-2.2.0.tar.gz for your installation.

```
$ cd /home/hadoopuser
$ wget http://www.webhostingjams.com/mirror/apache/hadoop/core/stable/hadoop-2
$ tar xvf hadoop-2.2.0.tar.gz
$ mv hadoop-2.2.0 hadoop
```

Setup Hadoop Environment

Copy and paste following lines into your .bashrc file under /home/hadoopuser.

```
# Set HADOOP_HOME
export HADOOP_HOME=/home/hduser/hadoop
# Set JAVA_HOME
export JAVA_HOME=/usr/lib/jvm/java-7-oracle
# Add Hadoop bin and sbin directory to PATH
export PATH=$PATH:$HADOOP_HOME/bin;$HADOOP_HOME/sbin
```

Update hadoop-env.sh

Update JAVA_HOME in /home/hadoopuser/hadoop/etc/hadoop/hadoop_env.sh to following

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle
```

Common Terminologies

Before we start getting into configuration details, lets discuss some of the basic terminologies used in Hadoop.

- **Hadoop Distributed File System**: A distributed file system that provides high-throughput access to application data. A HDFS cluster primarily consists of a NameNode that manages the file system metadata and DataNodes that store the actual data. If you compare HDFS to a traditional storage structures (e.g. FAT, NTFS), then NameNode is analogous to a Directory Node structure, and DataNode is analogous to actual file storage blocks.
- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

Update Configuration Files
Hadoop Wiki (http://wiki.apache.org/hadoop/GettingStartedWithHadoop) provides with set of configurations that are needed to start a single node cluster.   The documentation is outdated, and file structure has changed since that document was written. Add following setting to respective files under <configuration> section to do the settings in new file scheme. Make sure to replace **machine-name** with the name of your machine.

/home/hadoopuser/hadoop/etc/hadoop/core-site.xml (Other Options) (http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/core-default.xml)

```
<property>
  <name>hadoop.tmp.dir</name>
  <value>/home/hadoopuser/tmp</value>
  <description>Temporary Directory.</description>
</property>

<property>
  <name>fs.defaultFS</name>
  <value>hdfs://machine-name:54310</value>
  <description>Use HDFS as file storage engine</description>
</property>
```

/home/hadoopuser/hadoop/etc/hadoop/mapred-site.xml (Other Options) (http://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/mapred-default.xml)

```
<property>
 <name>mapreduce.jobtracker.address</name>
 <value>machine-name:54311</value>
 <description>The host and port that the MapReduce job tracker runs
  at. If “local”, then jobs are run in-process as a single map
  and reduce task.
</description>
</property>
```

/home/hadoopuser/hadoop/etc/hadoop/hdfs-site.xml (Other Options) (http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/hdfs-default.xml)

```
<property>
 <name>dfs.replication</name>
 <value>1</value>
 <description>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 create time.
 </description>
</property>
```

Format the Namenode
Before starting the cluster, we need to format the Namenode. Use the following command:

```
$ hdfs namenode -format
```

Start the Distributed Format System

Run the following command to start the DFS.

```
$ ./home/hadoopuser/hadoop/sbin/start-dfs.sh
```

After this command is successfully run, you can run command *jps*, and see that you have *NameNode, SecondaryNameNode, DataNode* running now.

Start the Yarn MapReduce Job tracker

Run the following command to start the DFS.

```
$ ./home/hadoopuser/hadoop/sbin/start-yarn.sh
```

After this command is successfully run, you can run command *jps*, and see that you have *NodeManager, ResourceManager* running now.

Lets's execute a MapReduce example now

You should be all set to run a MapReduce example now. Run the following command

```
$ hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.2.0.jar pi 3 10
```

Feedback and Questions?

if you have any feedback, or questions do leave a comment

Troubleshooting
Hadoop uses $HADOOP_HOME/logs directory. In case you get into any issues with your installation, that should be the first point to look at. In case, you need help with anything else, do leave me a comment.

Related Articles

Installing a Multi Node Hadoop Cluster on Ubuntu 14.04 (https://chawlasumit.wordpress.com/2015/03/09/install-a-multi-node-hadoop-cluster-on-ubuntu-14-04/)

Hadoop Java HotSpot execstack warning (https://chawlasumit.wordpress.com/2014/06/17/hadoop-java-hotspottm-execstack-warning/)

References
http://wiki.apache.org/hadoop/GettingStartedWithHadoop (http://wiki.apache.org/hadoop/GettingStartedWithHadoop)
http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster/ (http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster/)

ĭ CLUSTER, HADOOP, HADOOP DISTRIBUTED FILE SYSTEM, HADOOP ON UBUNTU 14.04, HDFS, INSTALLING HADOOP ON UBUNTU 14.04, SINGLE NODE HADOOP INSTALLATION, SUMIT CHAWLA, UBUNTU

28 thoughts on “Installing Hadoop on Ubuntu 14.04”

dineshramitc says:
JUNE 16, 2014 AT 6:20 PM
Reblogged this on Dinesh Ram Kali..

REPLY
Matt says:
AUGUST 31, 2014 AT 6:46 PM
A quick question: how did your prompt become
$hdfs
?

REPLY

sumitchawla says:
SEPTEMBER 6, 2014 AT 8:19 PM
Sorry Matt hdfs is the command here.

REPLY

dnyaneshwar11.patil@gmail.com says:
FEBRUARY 8, 2015 AT 9:50 AM
can i install hadoop on ubuntu 14.04(32-bit) ?

REPLY

sumitchawla says:
FEBRUARY 10, 2015 AT 3:12 AM
hi. yes you should be able to install it on 32bit version. hadoop binaries are built for 32bit by default

REPLY

Monica Gajbhe says:
FEBRUARY 10, 2015 AT 9:26 AM

```
cat .ssh/id_rsa.pub >> .ssh/authorized_keys
-su: .ssh/authorized_keys: No such file or directory
hduser@monica-VirtualBox:~$
```

Sir I am getting error when i m executing this command.As per your suggestions i have done all the steps, but i am stuck here.

please tell me solution for this problem…

REPLY

sumitchawla says:
FEBRUARY 10, 2015 AT 6:46 PM
Hi Monica

Did you create a ssh key using ssh-keygen -t rsa -P “”. If yes, it should create a key under .ssh directory. .ssh is a hidden directory under the home directory of the hadoop user. Please make sure ssh key is created by ssh-keygen command

REPLY

Hadoop Java HotSpot execstack warning | Sumit Chawla's Blog says:
MARCH 4, 2015 AT 5:16 AM
[…] Installing Hadoop on Ubuntu 14.04 […]

REPLY

Install a Multi Node Hadoop Cluster on Ubuntu 14.04 | Sumit Chawla's Blog says:
MARCH 9, 2015 AT 12:43 AM
[…] This article is about multi-node installation of Hadoop cluster. You would need minimum of 2 ubuntu machine/vm to complete a multi-node installation. If you want to just try out a single node cluster, follow this article on Installing Hadoop on Ubuntu 14.04. […]

REPLY

Dj says:
JUNE 14, 2015 AT 3:54 AM
Did the example run of hadoop give you a value of 3.6000000000 for PI?

REPLY

sumitchawla says:
JUNE 15, 2015 AT 6:07 PM
Yes i got following result:

Estimated value of Pi is 3.60000000000000000000

REPLY

Dhruv says:
JUNE 17, 2015 AT 5:59 AM
Awesome! Thank you!

Dj says:
JUNE 15, 2015 AT 12:08 AM
For the above question, could you please reply here so that I get an email notification?

REPLY

Garima says:
JULY 4, 2015 AT 7:09 AM
Hello Sir
When Iam tryin to run mapreduce example pi as in $ hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.2.0.jar pi 3 10 , I'am getting the error Connection Refused

```
Number of Maps = 16
Samples per Map = 1000
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Wrote input for Map #3
Wrote input for Map #4
Wrote input for Map #5
Wrote input for Map #6
Wrote input for Map #7
Wrote input for Map #8
Wrote input for Map #9
Wrote input for Map #10
Wrote input for Map #11
Wrote input for Map #12
Wrote input for Map #13
Wrote input for Map #14
Wrote input for Map #15
Starting Job
15/07/04 11:41:40 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:18032
15/07/04 11:41:41 INFO mapreduce.JobSubmitter: Cleaning up the staging area /tmp/hadoop-yarn/staging/hdfs/.staging/job_1435987646575_0001
java.net.ConnectException: Call From garima-HP-Compaq-8100-Elite-CMT-PC/127.0.1.1 to localhost:9000 failed on connection exception: java.net.ConnectException: Connection refused; For more details see: http://wiki.apache.org/hadoop/ConnectionRefused
at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
```

```
at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
at org.apache.hadoop.net.NetUtils.wrapWithMessage(NetUtils.java:791)
at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:731)
at org.apache.hadoop.ipc.Client.call(Client.java:1472)
at org.apache.hadoop.ipc.Client.call(Client.java:1399) and lot more…
```

Please help me with this…..

REPLY

sumitchawla says:
JULY 7, 2015 AT 5:07 AM
You need to use fully qualified hostname when doing any configuration. Are you using localhost anywhere? How many nodes are there in your cluster?

REPLY

Garima says:
JULY 7, 2015 AT 5:33 AM
Sir I'am using it on localhost.

sumitchawla says:
JULY 8, 2015 AT 3:34 AM
please check your config files and make sure you are not using localhost anywhere. Use fully qualified names in config files.

Harshita says:
JULY 31, 2015 AT 4:48 AM
Hello sir,
I am installing hadoop 2.6.0 on CentOS-6.6-x86_64.I have followed all the steps and all are working fine without error but start-dfs.sh is unable to start namenode datanode and secondarynamenode.and gives a warning as:
WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform… using builtin-java classes where applicable
Please help me out

REPLY

sumitchawla says:
JULY 31, 2015 AT 5:02 AM
This is just a warning. Are you getting any errors? Did you check log files under $HADOOP_HOME/logs

REPLY

Harshita says:
JULY 31, 2015 AT 5:32 AM
no other errors..it is simply not running the three daemons.
Everytime it is asking for password.
and now it is showing problem with jps too.

```
[madam@localhost Desktop]$ start-dfs.sh
15/07/30 22:27:41 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform… using builtin-java classes where applicable
Starting namenodes on [localhost]
madam@localhost's password:
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-madam-namenode-lo
```

```
localhost.localdomain.out
training
madam@localhost’s password:
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-madam-datanode-localhost.localdomain.out
training
Starting secondary namenodes [0.0.0.0]
madam@0.0.0.0’s password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-madam-secondarynamenode-localhost.localdomain.out
15/07/30 22:28:39 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform… using builtin-java classes where applicable
[madam@localhost Desktop]$ jps
bash: jps: command not found
```

sumitchawla says:
JULY 31, 2015 AT 5:39 AM
Did you setup password less SSH correctly? Its asking for password because SSH is not correctly setup. When logged in as hadoopuser , you should be able to do a password less ssh to your localhost also .

REPLY
Harshita says:
JULY 31, 2015 AT 5:43 AM
Thanks for such a quick response..:)

REPLY
Harshita says:
JULY 31, 2015 AT 5:44 AM
yep…i havent given any password

REPLY
Harshita says:
JULY 31, 2015 AT 6:22 AM
This is error is coming now.Please help me out what to do next.

```
[madam@localhost hadoop]$ start-dfs.sh
15/07/30 23:19:46 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform… using builtin-java classes where applicable
15/07/30 23:19:57 WARN hdfs.DFSUtil: Namenode for null remains unresolved for ID null. Check your hdfs-site.xml file to ensure namenodes are configured properly.
Starting namenodes on [machine-name]
machine-name: ssh: Could not resolve hostname machine-name: Temporary failure in name resolution
madam@localhost’s password:
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-madam-datanode-localhost.localdomain.out
Starting secondary namenodes [0.0.0.0]
madam@0.0.0.0’s password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-madam-secondarynamenode-localhost.localdomain.out
15/07/30 23:20:49 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform… using builtin-java classes where applicable
```

REPLY

Harshita says:
JULY 31, 2015 AT 3:24 PM
Now There are no log files in the hadoop directory..i tried it again

REPLY

rajeshgeek says:
SEPTEMBER 24, 2015 AT 2:45 PM
Hello,

Any suggestion on system configuration for 3 node Hadoop cluster, generally for POC use cases

REPLY

sumitchawla says:
SEPTEMBER 25, 2015 AT 1:55 AM
Hi Rajesh

The Answer depends on your use case. We have generally experimented in machines with atleast 8G RAM, and SSD as hard drive for improved I/O performance. Another dependency factor is the distribution you choose for final install. Here are my observations:

1. Apache Hadoop Binaries –
Pros – You get to play with all the latest code and features. You can upgrade your installation whenever you want. Lot of community support. You are at liberty to change the code and customize it to your needs.

Cons- Lack of commercial support in case you run into any issues, and you don't have anyway to solve it.

2. MapR –
Pros – Great improvement in I/O performance. If your hadoop jobs are going to do a lot of I/O operations, then this distribution performs much better than native hadoop hdfs support.
Cons – Cost factor if you are using a paid enterprise version. A relatively closed system. You will be dependent on MapR for code updates.

3. Hortonworks HDP –

Pros – Great Ambari integration. Installation is much easier and management is very easy.
Cons – Cost factor if you are using the support.

We ended up using HDP 2.2 in our production environment.

REPLY

Sethu Raam says:
OCTOBER 21, 2015 AT 2:14 PM
Getting this exception on running the example jar :

```
Number of Maps = 3
Samples per Map = 10
java.lang.IllegalArgumentException: java.net.UnknownHostException: machine-name
at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:377)
at org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310)
at org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176)
at org.apache.hadoop.hdfs.DFSClient.(DFSClient.java:678)
at org.apache.hadoop.hdfs.DFSClient.(DFSClient.java:619)
at org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149)
at org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653)
at org.apache.hadoop.fs.FileSystem.access$200(FileSystem.java:92)
at org.apache.hadoop.fs.FileSystem$Cache.getInternal(FileSystem.java:2687)
at org.apache.hadoop.fs.FileSystem$Cache.get(FileSystem.java:2669)
at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371)
at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:170)
at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:355)
at org.apache.hadoop.fs.Path.getFileSystem(Path.java:295)
at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.setInputPaths(FileInputFormat.java:500)
at org.apache.hadoop.examples.QuasiMonteCarlo.estimatePi(QuasiMonteCarlo.java:274)
at org.apache.hadoop.examples.QuasiMonteCarlo.run(QuasiMonteCarlo.java:354)
at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)
at org.apache.hadoop.examples.QuasiMonteCarlo.main(QuasiMonteCarlo.java:363)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.ProgramDriver$ProgramDescription.invoke(ProgramDriver.java:71)
at org.apache.hadoop.util.ProgramDriver.run(ProgramDriver.java:144)
at org.apache.hadoop.examples.ExampleDriver.main(ExampleDriver.java:74)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
Caused by: java.net.UnknownHostException: machine-name
```

REPLY

BLOG AT WORDPRESS.COM. | THE NUCLEARE THEME.