Setup Hadoop On Ubuntu 14.04 Linux

---Multi-Node Cluster

Note: this doc describes steps to install the Hadoop system.
i) You should already set up a cluster of computing nodes, using virtual machines or physical machines.
ii) The OS of the computing nodes should be Ubuntu 14.04. The Hadoop to be installed is 2.6.0.

Highlights of the steps:
Part I: Configure your cluster for Hadoop
i）Create a user “Hadoop”
ii）Install and configure Java
iii）Configure remote access (SSH)
Part II: Install and run Hadoop
i）Install and configure Hadoop at the master node
ii）Install and configure slave nodes

1. Create a user for Hadoop [at each node]

1）Log in your cluster nodes, and then create a user named “hadoop”.

```
$ sudo useradd -m hadoop -s /bin/bash
$ sudo passwd hadoop   # input the password ‘hadoop’ twice
$ sudo adduser hadoop sudo
```

(The words behind “#” are comments and notes, NOT part of the commands!)

2）使用hadoop用户登录继续后面的操作。

(如果是使用SecureCRT远程登录的话必须新建一个会话).

2. Install and configure Java [at each node]

1）Download jdk-8u60-linux-x64.tar.gz using wget and unpack it:

(download address: http://10.1.220.23:8888/test/jdk-8u60-linux-x64.tar.gz)注意空格！

```
$ wget 10.1.220.23:8888/test/jdk-8u60-linux-x64.tar.gz
$ sudo mkdir /usr/lib/jvm
$ sudo chown -R hadoop:hadoop /usr/lib/jvm
$ sudo tar –zxvf jdk-8u60-linux-x64.tar.gz –C /usr/lib/jvm
```

2） Configure the file “~/.bashrc”

```
$     sudo vi ~/.bashrc
```

To append the following statements:

```
export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_60
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

```
$     source    ~/.bashrc  # make it works
```

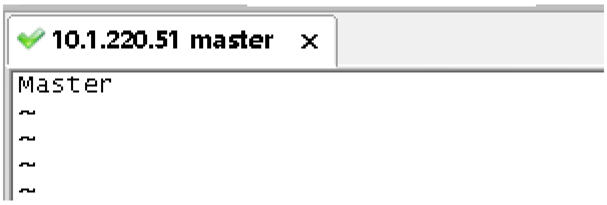
If you haven’t used vi yet, look at next few tips:
-i, insert
-Esc, escape insert
-o, add a new line and enter insert mode
-x, delete a character
-dd, delete a line
-:wq, save and quit
(P.S. if you want to learn more usage of vi, search online or look at this text:
http://10.1.220.23:8888/test/vi-manul.txt.)

3. Configure passphraseless SSH access [at each node]

To run Hadoop, the cluster nodes should be able to access/login to each other automatically. This is realized by passphraseless SSH access.

1) Set hostname

i）使用 vi 命令打开文件/etc/hostname，将里面的内容改为当前节点对应的名字，比如 Master 或者 Slave1. Hostname 自己设定，每个节点的名字应该不同。
’$ sudo vi /etc/hostname’



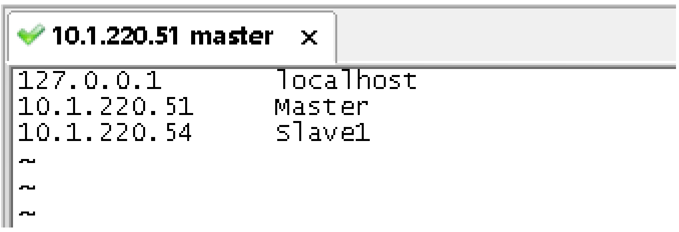
```
Master
~
~
~
~
```

ii）Edit /etc/hosts（使用命令'$ sudo vi /etc/hosts'）：
编辑 hosts 文件，使得各节点相互知道其他节点的 IP-hostname 对应关系。

（注意，Master 和 Slave1 对应的 IP 可以在云平台的实例那里看到）

```
Master     192.168.216.1     # change to your own IP
Slave1    192.168.216.2
```

完成后如下：



```
127.0.0.1       localhost
10.1.220.51     Master
10.1.220.54     Slave1
~
~
~
```

iii）Check reachability using ping
重启每个节点，并通过 ping 主机名测试主机名和 IP 是否正确设置。

2）Configure passphraseless SSH

```
$ ssh localhost        # if ssh cannot run, use command above
$ exit
$cd ~/.ssh      # if there no such file name .ssh, use '$ ssh localhost' and exit first
$ ssh-keygen –t rsa # if wait here, just push enter
$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys       # then can use '$ ssh Master' to test
$ scp ~/.ssh/id_rsa.pub hadoop@Slave1:/home/hadoop/    # copy to Slave1
```

Then on node Slave1, run:

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

Master can passphraseless SSH to Slave1.
Note: every node should '$ ssh localhost' and '$ exit' first.

4. Install Hadoop at master

1） Download hadoop-2.6.0.tar.gz and unpack it:

(https://10.1.220.23:8888/test/hadoop-2.6.0.tar.gz) 同样使用 wget 命令去下载

```
$ sudo tar -zxvf ./hadoop-2.6.0.tar.gz -C /usr/local    # decompress to /usr/local
$ cd /usr/local/
$ sudo mv ./hadoop-2.6.0/ ./hadoop                     # change file name to hadoop
$ sudo chown -R hadoop:hadoop ./hadoop                  # change file owner
$ cd ./hadoop
$ ./bin/hadoop # test hadoop
```

2) Configuration setups

Under /usr/local/hadoop/etc/hadoop/

i) Edit the file “slaves” (‘$ sudo vi filename’) to add the list of slave nodes:

```
Slave1
```

ii) Edit the file “core-site.xml” to set list of Hadoop nodes

Put these content below between ‘<configuration>’ and ‘</configuration>’.

```
//File: core-site.xml
<property>
        <name>fs.defaultFS</name>
        <value>hdfs://Master:9000</value>
</property>
<property>
        <name>hadoop.tmp.dir</name>
        <value>file:/usr/local/hadoop/tmp</value>
        <description>Abase for other temporary directories.</description>
</property>
```

```
//File: hdfs-site.xml# dfs.replication value is number of Slave
<property>
        <name>dfs.namenode.secondary.http-address</name>
        <value>Master:50090</value>
</property>
<property>
        <name>dfs.namenode.name.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/name</value>
</property>
<property>
        <name>dfs.datanode.data.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/data</value>
</property>
<property>
        <name>dfs.replication</name>
        <value>1</value>
</property>
```

# this file does not exist, you should copy from template like this:

```
$ cp mapred-site.xml.template mapred-site.xml
```

```
//File: mapred-site.xml
<property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
</property>
```

```
//File: yarn-site.xml
<property>
        <name>yarn.resourcemanager.hostname</name>
        <value>Master</value>
</property>
<property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
</property>
```

iii) Edit file hadoop-env.sh

find ‘`export JAVA_HOME=${JAVA_HOME}`’ and change it to ‘`export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_60`’.

5. Install and configure Hadoop at slaves

1） Pack Hadoop files at Master node, in /usr/local/, and copy the tar package to slaves.

```
$ sudo tar –zcvf hadoop.tar.gz hadoop
$ scp hadoop.tar.gz hadoop@Slave1:/home/hadoop
```

2) Release Hadoop files at slaves:

```
$ sudo tar -zxvf ~/hadoop.tar.gz -C /usr/local
$ sudo chown -R hadoop:hadoop /usr/local/hadoop
```

6. Run your Hadoop system

Note: run, start, stop operation just do on Master.

To start the system :

```
$ bin/hdfs namenode –format       #just run once
```

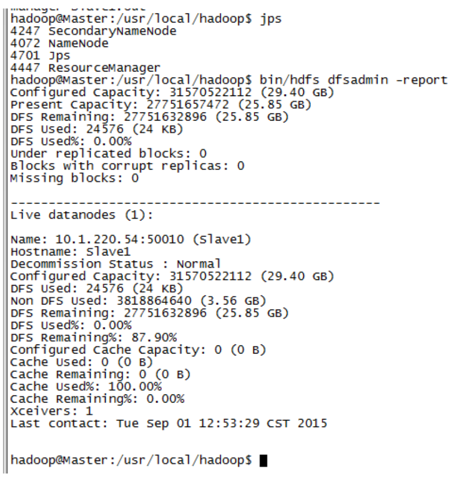
```
$ sbin/start-dfs.sh
$ sbin/start-yarn.sh
```

To check its status: (the information reported is show below)

```
$ jps
$ bin/hdfs dfsadmin -report
```

To stop the cluster:

```
$ sbin/stop-dfs.sh
$ sbin/stop-yarn.sh
```



```
hadoop@Master:/usr/local/hadoop$ jps
4247 SecondaryNameNode
4072 NameNode
4701 Jps
4447 ResourceManager
hadoop@Master:/usr/local/hadoop$ bin/hdfs dfsadmin -report
Configured Capacity: 31570522112 (29.40 GB)
Present Capacity: 27751657472 (25.85 GB)
DFS Remaining: 27751632896 (25.85 GB)
DFS Used: 24576 (24 KB)
DFS Used%: 0.00%
Under replicated blocks: 0
Blocks with corrupt replicas: 0
Missing blocks: 0

-------------------------------------------------
Live datanodes (1):

Name: 10.1.220.54:50010 (Slave1)
Hostname: Slave1
Decommission Status : Normal
Configured Capacity: 31570522112 (29.40 GB)
DFS Used: 24576 (24 KB)
Non DFS Used: 3818864640 (3.56 GB)
DFS Remaining: 27751632896 (25.85 GB)
DFS Used%: 0.00%
DFS Remaining%: 87.90%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Tue Sep 01 12:53:29 CST 2015


hadoop@Master:/usr/local/hadoop$ █
```