Hadoop

Multi Node Cluster Installation

Prepared by: M Fazli A Jalaluddin (RESEARCH/UTP)

Guideline

- Guidelines:
  - (Master, Slave or Both) - The commands need to be executed in which node
  - # command - means to execute commands in terminal
  - **Line to append** - means lines that need to be added in a file
  - [comments] - means comments
  - <value> - means input the value

Installation Overall Steps

Step 1: Set hostname (Master and Slave)

Step 2: Install Java (Master and Slave)

Step 3: Create group and user (Master and Slave)

Step 4: Configure paswordless ssh (Master and Slave)

Step 5: Install Hadoop (Master and Slave)

Step 6: Hadoop Configuration (Master)

Step 7: Start Hadoop Services (Master)

Step 8: Start Hadoop History Server (Slave)

Step 9: Run sample job (Master)

Step 10: Check Hadoop Health and Jobs using Web UI (Master and Slave)

Installation

Step 1: Set hostname (Master and Slave)

```
#  sudo apt-get update;
#  sudo apt-get install -y vim openssh-server
#  ifconfig [To check both IP Address]
#  sudo vi /etc/hosts [Delete 127.0.1.1 address and input these new address]
        <IP Address> hadoop-master
        <IP Address> hadoop-slave
```

Installation

Step 1: Cont. (Master and Slave)

sudo vi /etc/hostname

-> In Master [Replace default hostname]

hadoop-master

-> In Slave [Replace default hostname]

hadoop-slave

sudo reboot

ping hadoop-master [After reboot]

ping hadoop-slave

Installation

Step 2: Install Java (Master and Slave)

```
#  sudo apt-add-repository ppa:webupd8team/java

#  sudo apt-get update

#  sudo apt-get install oracle-java7-installer
```

Installation

Step 3: Create group and user (Master and Slave)

```
#  sudo addgroup hdgroup
#  sudo adduser -ingroup hdgroup hduser
#  sudo usermod -a -G sudo hduser
```

Installation

Step 4: Configure paswordless ssh (Master and Slave)

```
#  su - hduser
#  ssh-keygen -t rsa -P ""
#  ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop-master
#  ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop-slave
#  ssh localhost [If no password asked, success. Exit afterward]
#  ssh hadoop-master [If no password asked, success. Exit afterward]
#  ssh hadoop-slave [If no password asked, success. Exit afterward]
```

Installation

Step 5: Install Hadoop (Master and Slave)

[Download hadoop packages using web browser in hadoop-master]

[Locate the downloaded package and copy to hduser home folder]

```
#  su - hduser
#  cp /home/user/packages/hadoop-2.7.1.tar.gz .
```

[Copy the downloaded package from hadoop-master to hadoop-slave]

```
#  scp ~/hadoop-2.7.1.tar.gz hduser@hadoop-slave:~/
```

Installation

Step 5: Cont. (Master and Slave)

```
# tar xzfv hadoop-2.7.1.tar.gz [Extract the tar file]
# sudo mkdir -p /usr/local/hadoop
# sudo mkdir -p /usr/local/hadoop_tmp/hdfs/namenode
# sudo mkdir -p /usr/local/hadoop_tmp/hdfs/datanode
# sudo mv ~/hadoop-2.7.1/* /usr/local/hadoop/
# ls - la /usr/local/hadoop
# sudo chown -R hduser:hdgroup /usr/local/hadoop
# sudo chown -R hduser:hdgroup /usr/local/hadoop_tmp
```

Installation

Step 5: Cont. (Master and Slave)

vi ~/.bashrc

```
export HADOOP_HOME=/usr/local/hadoop
export JAVA_HOME=/usr/lib/jvm/java-7-oracle
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

Installation

Step 5: Cont. (Master and Slave)

source ~/.bashrc

hadoop version [The output should shows the version]

Installation

Step 6: Hadoop Configuration (Master)

[The configuration file in **/usr/local/hadoop/etc/hadoop** that must be edited]:

1. hadoop-env.sh
2. core-site.xml
3. mapred-site.xml
4. yarn-site.xml
5. hdfs-site.xml
6. masters
7. slaves

Installation

Step 6: Cont. (Master)

vi /usr/local/hadoop/etc/hadoop/**hadoop-env.sh** [1]

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

Installation

Step 6: Cont. (Master)

\# vi /usr/local/hadoop/etc/hadoop/**core-site.xml** [2]

```
<configuration>
 <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoop-master:54310</value>
 </property>
 <property>
        <name>hadoop.tmp.dir</name>
        <value>/usr/local/hadoop_tmp</value>
 </property>
</configuration>
```

Installation

Step 6: Cont. (Master)

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

vi /usr/local/hadoop/etc/hadoop/**mapred-site.xml** [3]

```
<configuration>
 <property>
        <name>mapreduce.jobtracker.address</name>
        <value>hadoop-master:54311</value>
 </property>
 <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
 </property>
</configuration>
```

Installation

Step 6: Cont. (Master)

vi /usr/local/hadoop/etc/hadoop/**yarn-site.xml** [4]

```
<configuration>
 <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
 </property>
 <property>
        <name>yarn.resourcemanager.scheduler.address</name>
        <value>hadoop-master:8030</value>
 </property>
```

⟶ **Property continues next slide**

Installation

```
<property>
        <name>yarn.resourcemanager.address</name>
        <value>hadoop-master:8032</value>
 </property>
 <property>
        <name>yarn.resourcemanager.webapp.address</name>
        <value>hadoop-master:8088</value>
 </property>
 <property>
        <name>yarn.resourcemanager.resource-tracker.address</name>
        <value>hadoop-master:8031</value>
 </property>
 <property>
        <name>yarn.resourcemanager.admin.address</name>
        <value>hadoop-master:8033</value>
 </property>
```

⟶ Property continues next slide

Installation

```
        <property>
                <name>yarn.nodemanager.vmem-check-enabled</name>
                <value>false</value>
        </property>
        <property>
                <name>yarn.log-aggregation-enable</name>
                <value>true</value>
        </property>
</configuration>
```

Installation

Step 6: Cont. (Master)

\# vi /usr/local/hadoop/etc/hadoop/**hdfs-site.xml** [5]

```
<configuration>
 <property>
        <name>dfs.replication</name>
        <value>2</value>
 </property>
 <property>
        <name>dfs.namenode.name.dir</name>
        <value>/usr/local/hadoop_tmp/hdfs/namenode</value>
 </property>
```

⟶ **Property continues next slide**

Installation

```
<property>
        <name>dfs.datanode.data.dir</name>
        <value>/usr/local/hadoop_tmp/hdfs/datanode</value>
 </property>
 <property>
        <name>dfs.client.socket-timeout</name>
        <value>3000000</value>
 </property>
 <property>
        <name>dfs.datanode.socket.write.timeout</name>
        <value>3000000</value>
 </property>
</configuration>
```

Installation

Step 6: Cont. (Master)

vi /usr/local/hadoop/etc/hadoop/**masters** [6]

hadoop-master

vi /usr/local/hadoop/etc/hadoop/**slaves** [7]

hadoop-slave [Replace localhost]

[IMPORTANT - Execute the command only **one time** on Master]

hdfs namenode -format

Installation

Step 6: Cont. (Master)

[Copy all the edited files from Master to Slave]

```
#  cd /usr/local/hadoop/etc/hadoop

#  scp core-site.xml hadoop-env.sh hdfs-site.xml mapred-site.xml yarn-site.xml
slaves masters hduser@hadoop-slave:/usr/local/hadoop/etc/hadoop
```

Installation

Step 7: Start Hadoop services (Master)

su - hduser [Must start all services as hduser]

start-dfs.sh

start-yarn.sh

jps [To verify installation and should see something like this]

xxxxx SecondaryNameNode
xxxxx Resource Manager
xxxxx NameNode
xxxxx Jps

Installation

Step 8: Start Hadoop history server (Slave)

su - hduser [Must start history server as hduser]

mr-jobhistory-server.sh start historyserver

jps [To verify installation and should see something like this]

xxxxx NodeManager
xxxxx Datanode
xxxxx JobHistoryServer
xxxxx Jps

Installation

Step 9: Run sample job (Master)

```
#  cd /usr/local/hadoop/share/hadoop/mapreduce
#  hadoop jar hadoop-mapreduce-examples-2.7.1.jar pi 5 10
```

[Check job log at slave]

```
# tail -f /usr/local/hadoop/logs/yarn-hduser-nodemanager-hadoop-slave.log
```

Installation

Step 10: Check Hadoop Health and Jobs using Web UI

(Master and Slave)

1. Hadoop Resource Manager -> http://hadoop-master:8088
2. Hadoop Cluster Health -> http://hadoop-master:50070
3. Hadoop Job History -> http://hadoop-master:19888

Applications on Hadoop

R and Octave:

su - hduser

cp -R /home/user/hadoop_test .

[Go into each folder and read README file to get started]

Q&A