**Hadoop**

**For 1 node configuration**

**To mount EBS volume disk, after creating instance, do following steps, take i3.large instance as example**  
1. lsblk  
2. sudo mkfs -t ext4 /dev/nvme0n1  
3. sudo mkdir /data  
4. sudo mount /dev/nvme0n1 /data  
5. Chmod 777 /data  
  
  
**Set up the environment**

./hadoop\_script.sh

**Change Hadoop configuration**

1. hadoop/core-site.xml

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/home/ubuntu/tmp</value>

</property>

</configuration>

2. hadoop/hdfs-site.xml

<configuration>

<property>

<name>dfs.blocksize</name>

<value>1g</value>

</property>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

<value>file:///home/ubuntu/dfs/namenode</value>

</property>

<property>

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

<value>file:///home/ubuntu/dfs/datanode</value>

</property>

</configuration>

3. hadoop/mapred-site.xml

<configuration>

<property>

<name>mapreduce.map.memory.mb</name>

<value>-Xmx4g</value>

</property>

<property>

<name>mapreduce.reduce.memory.mb</name>

<value>-Xmx4g</value>

</property>

<property>

<name>mapreduce.map.java.opts</name>

<value>-Xmx4g</value>

</property>

<property>

<name>mapreduce.reduce.java.opts</name>

<value>-Xmx4g</value>

</property>

<property>

<name>mapreduce.job.tracker</name>

<value>ec2-52-14-11-189.us-east-2.compute.amazonaws.com</value>

</property>

</configuration>

4. hadoop/yarn-site.xml

<configuration>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.auz-services.mapreduce.shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

</configuration>

5. ~/.bashrc

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre

export HADOOP\_HOME=/home/ubuntu/hadoop

export HADOOP\_CLASSPATH=/usr/lib/jvm/java-8-openjdk-amd64/lib/tools.jar

export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export YARN\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

PATH=$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin

source ~/.bashrc

**Setup password-less login between namenode and datanode**

ssh-keygen -t rsa -P '' -f ~/.ssh/id\_rsa

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

chmod 0600 ~/.ssh/authorized\_keys

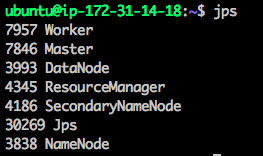
**Starting Hadoop**

hdfs namenode -format

./hadoop/sbin/start-dfs.sh

./hadoop/sbin/start-yarn.sh

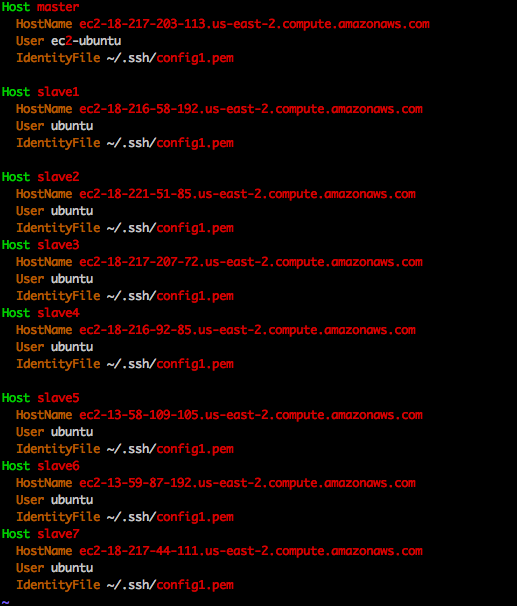
jps



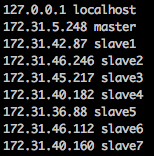
**For 8 node configuration**

**Passwordless SSH between datanode and namenode**

~/.ssh/config



/etc/hosts



cat ~/.ssh/id\_rsa.pub | ssh <host\_name> 'cat >> ~/.ssh/authorized\_keys'

**Set hadoop configuration**

1. hadoop/core-site.xml

<property>

<name>fs.defaultFS</name>

<value>hdfs://ec2-18-217-94-14.us-east-2.compute.amazonaws.com:9000</value>

</property>

2. hadoop/yarn-site.xml

<property>

<name>yarn.resourcemanager.hostname</name>

<value>ec2-18-217-94-14.us-east-2.compute.amazonaws.com</value>

</property>

<property>

<name>yarn.resourcemanager.address</name>

<value>ec2-18-217-94-14.us-east-2.compute.amazonaws.com:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>ec2-18-217-94-14.us-east-2.compute.amazonaws.com:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>ec2-18-217-94-14.us-east-2.compute.amazonaws.com:8031</value>

</property>

3. hadoop/mapred-site.xml

<property>

<name>mapreduce.jobtracker.address</name>

<value>ec2-18-217-94-14.us-east-2.compute.amazonaws.com:54311</value>

</property>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

4.scp ~/hadoop/etc/hadoop/core-site.xml ~/hadoop/etc/hadoop/hdfs-site.xml ~/hadoop/etc/hadoop/yarn-site.xml~/hadoop/etc/hadoop/mapred-site.xml <host\_name>:/home/ubuntu/hadoop/etc/hadoop

-------------------------------------------------------------------------

**Generating 128GB/1TB**

hadoop jar ~/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.4.jar teragen <file\_size> /inputfile

**Compile and run file**

hadoop com.sun.tools.javac.Main ~/hadoop/SortHadoop.java

jar cf wc.jar SortHadoop\*.class

time hadoop jar wc.jar SortHadoop /inputfile/part-m-00000 /outputfile

**Validate sorting file**

hadoop jar ~/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.4.jar teravalidate /outputfile /outputfile