

# PROGRAMMING ASSIGNMENT 2

## Terasort

**Ganesh Mahesh**

**A20391878**

**Sudhindra Manohar Bellur Vasudeva**

**A20377851**

### **Description**

The purpose of this document is to describe steps to install, configure and capture the results after running Sorting implementation on files greater than memory size in Hadoop, Java and also, by comparing the performance of Shared Memory Sort Java, Hadoop and Spark and determine which one is better.

### **Abstract**

Implementation of external sort algorithm for Shared Memory and Hadoop configurations on amazon ec2 instances. Custom external sort algorithm has been implemented in shared memory. Whereas Hadoop inbuild services are used to achieve tera sort. Here ec2 configurations considered are single node and eight node clusters of i3.large and i3.4xlarge (single node) and 8 node i3large instances.

### **Problem statement**

- Due to shortage of memory size to load the entire data set in memory sort isn't possible
- as a result of which external sort is employed to sort the huge dataset
- external merge sort is employed to perform sort on shared memory configuration
- multiple threads are used to have concurrent read and write from and to the disk (useful in case of a SSD based storage)

### **Runtime environment settings:**

Operating System: Ubuntu Linux Operating System.

Java version: 1.8

Hadoop version:2.9.0

Config 1: Instance type i3.large SSD size 475GB

Config 1: Instance type i3.4xlarge SSD size 2\*1800Gb

Linux distribution kernel: 3.19.0-25-generic

## **Gensort and Valsort**

**Gensort** is used to generate some content with key/value pairs where a 10 digit key is followed by a value separated by blank spaces. Dataset generated by Gensort is given as input while running the code.

**Valsort** is way to check the correctness of the sorting algorithm that has been applied on the dataset generated by Gensort. The result would be a success only if all the records in the file is completely and correctly sorted. If not we would see records are unordered in the output screen.

## **SHARED MEMORY**

Proposed solution for Share Memory

### **Over view:**

- Read certain chunk of data in to main memory from disk and sort by merge sort
- Write sorted data to disk
- Repeat the above steps until all the data is traversed and stored into chunked intermediate files(this completes initial map phase)
- Merge phase begins where based on number of threads divide the available intermediate files in such a way that a thread handles certain number of files.
  - Each thread reads certain amount of data from the files it is handling and sorts
  - Writes the sorted data to another intermediate file
  - This results in the completion of one phase of merge
  - Further phases are carried out in order to sort and combine all the files
  - Last we have a single file consisting of sorted data

### **Implementation details**

#### **Prerequisites:**

- Gensort data generator
- Latest JAVA environment
- Amazon EC2 instance as per configurations

### **Configuration 1**

#### **Shared Memory Terasort on amazon ec2 i3.large single node instance**

I3.large	Description
Compute	2 vCPU
Memory	15.25 GiB
Disk	0.475 TB SSD

## Amazon EC2 instance used

```
ubuntu@ip-172-31-9-127: ~/cloud/src
```

## Running instance

```
ubuntu@ip-172-31-9-127: ~/cloud/src
File Edit View Search Terminal Help

Last login: Mon Dec 4 03:34:09 2017 from 98.227.60.255
ubuntu@ip-172-31-9-127:~$ cd cloud/src/
ubuntu@ip-172-31-9-127:~/cloud/src$ vi
com/ out
ubuntu@ip-172-31-9-127:~/cloud/src$ vi com/iit/cs553/sharedmem/
Chunk.java Record.java SortData.java
FilesMerge.java SharedMemory.java Utility.java
ubuntu@ip-172-31-9-127:~/cloud/src$ vi com/iit/cs553/sharedmem/SharedMemory.java

ubuntu@ip-172-31-9-127:~/cloud/src$ javac com/iit/cs553/sharedmem/
Chunk.java Record.java SortData.java
FilesMerge.java SharedMemory.java Utility.java
ubuntu@ip-172-31-9-127:~/cloud/src$ javac com/iit/cs553/sharedmem/*.java
com/iit/cs553/sharedmem/Utility.java:47: warning: unreachable catch clause
    } catch (IOException e) {
        ^
    thrown type FileNotFoundException has already been caught
1 warning
ubuntu@ip-172-31-9-127:~/cloud/src$ java -Xmx14703m -classpath . com.iit.cs553.s
haredmem.SharedMemory
```

```
+ warning
ubuntu@ip-172-31-9-127:~/cloud/src$ java -Xmx14703m -classpath . com.iit.cs553.
sharedmem.SharedMemory
/nFile size in bytes 1.374389534E11
/nFile size in 100 byte records 1.374389534E9
Ram Size in bytes 1.6033144832E10
/n availableRAMSizeInRecords 1.60331448E8
/n numberOfRamFits 9.0
/n chunkSize 4.0082862E7
/n numOfChunksAvailable 35.0
/n chunksPerThread 9.0
Thread 0 started
Thread 1 started
Thread 2 started
Thread 3 started
```

## Output format

```

ubuntu@ip-172-31-9-127:~/cloud/src$ javac com/iit/cs553/sharedmem/*.java
com/iit/cs553/sharedmem/Utility.java:47: warning: unreachable catch clause
    } catch (IOException e) {
        ^
    thrown type FileNotFoundException has already been caught
1 warning
ubuntu@ip-172-31-9-127:~/cloud/src$ java -Xmx14703m -classpath . com.iit.cs553.s
haredmem.SharedMemory
/nFile size in bytes 1.374389534E11
/nFile size in 100 byte records 1.374389534E9
Ram Size in bytes 1.6033144832E10
/n availableRAMSizeInRecords 1.60331448E8
/n numberofRamFits 9.0
/n chunkSize 4.0082862E7
/n numOfChunksAvailable 35.0
/n chunksPerThread 9.0
Time taken in minutes : 168
Initial Map Phase number of times read :288written :248
Number of intermediate files created during map phase :248
Available Ram size 1.6033144832E10
File size under consideration in bytes :1.374389534E11
Number of times merge happened :248
Number of times read :256written :248 during merge phase

```

### Val sort output

```

0_0_30.out.tmp  com  out
ubuntu@ip-172-31-9-127:~/cloud/src$ ./../64/valsort -o out 0_0_30.out.tmp
First unordered record is record 2
sump pump fatal error: pfunc_get_rec: partial record of 56 bytes found at end of input
ubuntu@ip-172-31-9-127:~/cloud/src$ 

```

### Configuration 2

#### Shared Memory Terasort on amazon ec2 i3.4xlarge single node instance

I3.large	Description
Compute	16 vCPU
Memory	122 GiB
Disk	3.8 TB SSD(2 disks)

#### Amazon EC2 instance used

```
ubuntu@ip-172-31-8-55: ~/cloud/src
File Edit View Search Terminal Help
ubuntu@ip-172-31-8-55:~/cloud$ cd src/
ubuntu@ip-172-31-8-55:~/cloud/src$ java -Xmx130996m -classpath . com.iit.cs553.s
haredmem.SharedMemory
File Edit View Search Terminal Help
1 warning
ubuntu@ip-172-31-9-127:~/cloud/src$ java -Xmx14703m -classpath .
com.iit.cs553.s
haredmem.SharedMemory
/nFile size in bytes 1.374389534E11
/nFile size in 100 byte records 1.374389534E9
Ram Size in bytes 1.6033144832E10
/n availableRAMSizeInRecords 1.60331448E8
/n numberOfRamFits 9.0
/n chunkSize 4.0082862E7
/n numofChunksAvailable 35.0
/n chunksPerThread 9.0
Time taken in minutes : 168
Initial Map Phase number of times read :288written :248
Number of intermediate files created during map phase :248
Available Ram size 1.6033144832E10
File size under consideration in bytes :1.374389534E11
Number of times merge happened :248
Number of times read :256written :248 during merge phase
ubuntu@ip-172-31-9-127:~/cloud/src$ vi com/iit/cs553/sharedmem
d
ubuntu@ip-172-31-9-127:~/cloud/src$ vi com/iit/cs553/sharedmem
```

```
ubuntu@ip-172-31-8-55: ~/cloud/src
File Edit View Search Terminal Help
ubuntu@ip-172-31-8-55:~/cloud$ cd src/
ubuntu@ip-172-31-8-55:~/cloud/src$ java -Xmx130996m -classpath . com.iit.cs553.s
haredmem.SharedMemory
/nFile size in bytes 1.0995116278E12
/nFile size in 100 byte records 1.0995116278E10
Ram Size in bytes 1.28847745024E11
/n availableRAMSizeInRecords 1.28847745E9
/n numberofRamFits 9.0
/n chunkSize 3.22119363E8 size in bytes 1.374389534E11
/n numofChunksAvailable 35.0 size in 100 byte records 1.374389534E9
/n chunksPerThread 9.0 Size in bytes 1.6033144832E10
```

## Output format

```
0_0_30_out.tmp com out
ubuntu@ip-172-31-9-127:~/cloud/src$ ./../64/valsort -o out 0_0_30_out.tmp
First unordered record is record 2
sump pump fatal error: pfunc_get_rec: partial record of 56 bytes found at end of input
ubuntu@ip-172-31-9-127:~/cloud/src$
```

## HADOOP:

### Configuration File changed:

1. `hadoop-env.sh`- This file should be updated with environment variables settings such as `JAVA_HOME` which affects the Hadoop daemon behavior while storing files and increasing heap size.
2. `Slaves`- This file consists the public DNS of Master and Slave where the Hadoop slave daemons will run. This configuration required for multimode setup. In case of single node it consists of localhost.
3. `Core-site.xml`- This file consists of I/O configuration setting for Hadoop Core common to HDFS and MapReduce.
4. `Hdfs-site.xml`- This file contains the configuration setting for HDFS daemon, the namenode, the secondary namenode and the datanode nodes.
5. `Mapred-site.xml`- This file contains the configuration setting for Hadoop daemon job tracker and task tracker
6. `Yarn-site.xml`- This file contains the configuration setting for YARN daemon the ResourceManager, NodeManager, and WebAppProxy.

### Config1 - Hadoop Single-node cluster setup

1.Create an i3.large EC2 instance .

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
config 2	i-0741bcc112f3d268	i3.4xlarge	us-west-2c	running	2/2 checks ...	None	ec2-52-34-
config 1	i-03ea618e587c116fc	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-35-166-132-238.us-west-2.compute.amazonaws.com
	i-00f14967b7ac96902	i3.large	us-west-2c	terminated		None	

1. Use the below commands to setup the environment for java. Using default JDK we will get the latest version of java from the repository which is JDK 1.8

```
cd ~
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install ssh
sudo apt-get install default-jdk
```

2.Install ssh and add the public key to the authorized keys

```
sudo apt-get install ssh
ssh-keygen -t rsa
```

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

```
Selecting previously unselected package binutils.
Preparing to unpack .../binutils_2.24-5ubuntu14_amd64.deb ...
Unpacking binutils (2.24-5ubuntu14) ...
Selecting previously unselected package libgcc-4.8-dev:amd64.
Preparing to unpack .../libgcc-4.8-dev_4.8.4-2ubuntu1-14.04.1_amd64.deb ...
Unpacking libgcc-4.8-dev:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Selecting previously unselected package gcc-4.8.
Preparing to unpack .../gcc-4.8_4.8.4-2ubuntu1-14.04.1_amd64.deb ...
Unpacking gcc-4.8 (4.8.4-2ubuntu1-14.04.1) ...
Selecting previously unselected package gcc.
Preparing to unpack .../gcc_4%3a4.8.2-1ubuntu6_amd64.deb ...
Unpacking gcc (4:4.8.2-1ubuntu6) ...
Selecting previously unselected package libc-dev-bin.
Preparing to unpack .../libc-dev-bin_2.19-0ubuntu6.7_amd64.deb ...
Unpacking libc-dev-bin (2.19-0ubuntu6.7) ...
Selecting previously unselected package linux-libc-dev:amd64.
Preparing to unpack .../linux-libc-dev_3.13.0-83.127_amd64.deb ...
Unpacking linux-libc-dev:amd64 (3.13.0-83.127) ...
Selecting previously unselected package libc6-dev:amd64.
Preparing to unpack .../libc6-dev_2.19-0ubuntu6.7_amd64.deb ...
Unpacking libc6-dev:amd64 (2.19-0ubuntu6.7) ...
Selecting previously unselected package manpages-dev.
Preparing to unpack .../manpages-dev_3.54-1ubuntu1_all.deb ...
Unpacking manpages-dev (3.54-1ubuntu1) ...
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Setting up libasan0:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up libatomic1:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up libomp1:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up libitm1:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up libquadmath0:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up libtsan0:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up binutils (2.24-Subuntu14) ...
Setting up libgcc-4.8-dev:amd64 (4.8.4-2ubuntu1-14.04.1) ...
Setting up gcc-4.8 (4.8.4-2ubuntu1-14.04.1) ...
Setting up gcc (4:4.8.2-1ubuntu6) ...
Setting up libc-dev-bin (2.19-0ubuntu6.7) ...
Setting up linux-libc-dev:amd64 (3.13.0-83.127) ...
Setting up libc6-dev:amd64 (2.19-0ubuntu6.7) ...
Setting up manpages-dev (3.54-1ubuntu1) ...
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
ubuntu@ip-172-31-5-30:~$ sudo apt-get install ssh
```

### 3. Set up SSH certificates to avoid entering password each time:

```
Preparing to unpack .../openSSH-server_1%3a6.6p1-2ubuntu2.6_amd64.deb ...
Unpacking openSSH-server (1:6.6p1-2ubuntu2.6) over (1:6.6p1-2ubuntu2.4) ...
Preparing to unpack .../openSSH-client_1%3a6.6p1-2ubuntu2.6_amd64.deb ...
Unpacking openSSH-client (1:6.6p1-2ubuntu2.6) over (1:6.6p1-2ubuntu2.4) ...
Selecting previously unselected package ssh.
Preparing to unpack .../ssh_1%3a6.6p1-2ubuntu2.6_all.deb ...
Unpacking ssh (1:6.6p1-2ubuntu2.6) ...
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34-rc-0ubuntu2) ...
Setting up openSSH-client (1:6.6p1-2ubuntu2.6) ...
Setting up openSSH-server (1:6.6p1-2ubuntu2.6) ...
ssh stop/waiting
ssh start/running, process 15734
Setting up ssh (1:6.6p1-2ubuntu2.6) ...
Ubuntu@ip-172-31-5-30:~$ eval `ssh-agent -s`
Agent pid 15748
Ubuntu@ip-172-31-5-30:~$ chmod 600 'hadoop.pem'
Ubuntu@ip-172-31-5-30:~$ ssh-add 'hadoop.pem'
Identity added: hadoop.pem (hadoop.pem)
Ubuntu@ip-172-31-5-30:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Your identification has been saved in /home/ubuntu/.ssh/id_rsa.
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub.
The key fingerprint is:
e8:66:fd:b6:26:b4:0e:f5:00:bc:0f:2d:85:f9:bc:65 ubuntu@ip-172-31-5-30
The key's randomart image is:
----[ RSA 2048]----+
 . o
 = .
 0
 = S E
 . *.*
 =+..
 0 ..o.
 ..+o.
```

### 4. Download and extract Hadoop

```
wget http://mirror.olnevhost.net/pub/apache/hadoop/common/hadoop-2.9.0/hadoop-2.9.0.tar.gz
```

```
tar -xvf hadoop-2.9.0.tar.gz
```

```
mv hadoop-2.9.0 hadoop
```

5.create directory to run name node and data node

```
mkdir ~/hdfs
```

```
mkdir ~/hdfs/namenode
```

```
mkdir ~/hdfs/datanode
```

6.Give write and execute permission to that directoy

```
chmod 755 ~/hadoop
```

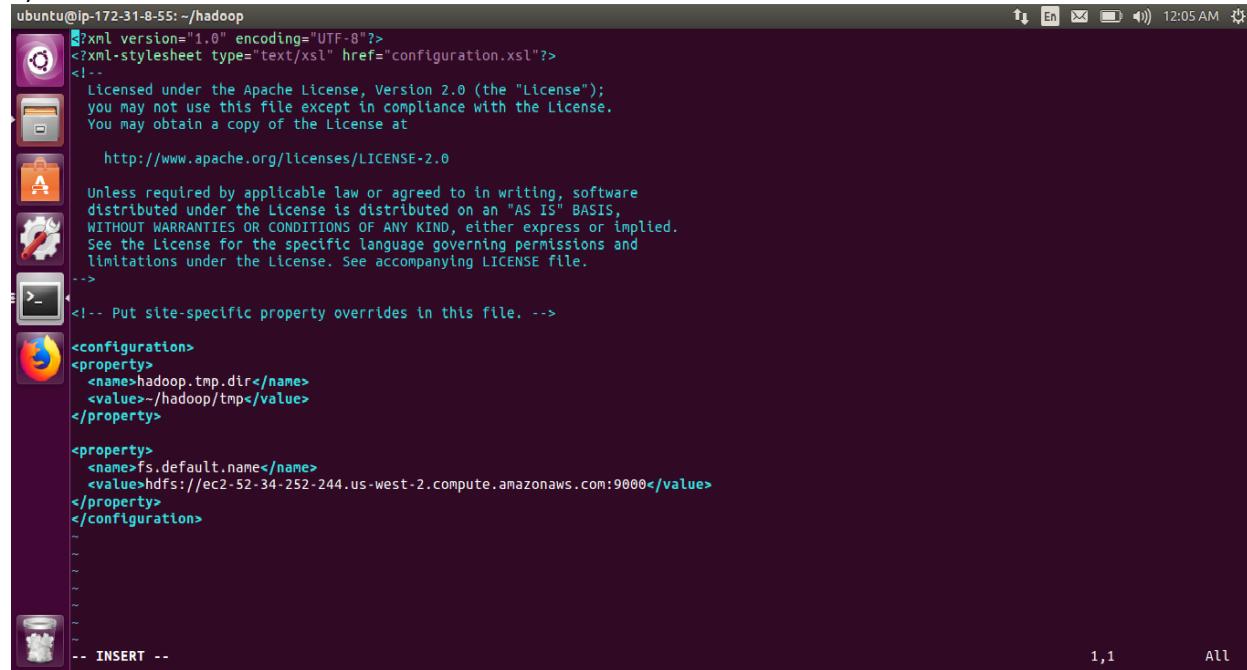
```
chmod 755 ~/hdfs
```

```
chmod 755 ~/hdfs/namenode
```

```
chmod 755 ~/hdfs/datanode/
```

7.Now start edit the config files present in /Hadoop/etc/Hadoop/ folder.

a) Edit core-site.xml



```
ubuntu@ip-172-31-8-55:~/hadoop
: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>hadoop.tmp.dir</name>
<value>~/hadoop/tmp</value>
</property>
<property>
<name>fs.default.name</name>
<value>hdfs://ec2-52-34-252-244.us-west-2.compute.amazonaws.com:9000</value>
</property>
</configuration>
~
~
~
~
~
~
-- INSERT --
1,1      All
```

b) Edit hdfs-site.xml

```
ubuntu@ip-172-31-8-55:~/hadoop
  <?xml version="1.0" encoding="UTF-8"?>
  <xsl-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.name.dir</name>
  <value>/home/ubuntu/hdfs/namenode</value>
</property>
<property>
  <name>dfs.data.dir</name>
  <value>/home/ubuntu/hdfs/datanode</value>
</property>
</configuration>
~
~
~
"etc/hadoop/hdfs-site.xml" 32L, 1059C
1,1          All
```

### c)Edit yarn-site.xml

```
ubuntu@ip-172-31-8-55:~/hadoop
  <?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>
<!-- Site specific YARN configuration properties -->
<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>
~
~
~
~
~
~
"etc/hadoop/yarn-site.xml" 26L, 947C
1,1          All
```

### d)Edit mapred-site.xml

```
ubuntu@ip-172-31-8-55:~/hadoop
  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>mapred.job.tracker</name>
  <value>localhost:54311</value>
</property>
</configuration>
~
```

e) Edit hadoop-env.sh

```
ubuntu@ip-172-31-8-55:~/hadoop
# Licensed to the Apache Software Foundation (ASF) under one
# or more contributor license agreements. See the NOTICE file
# distributed with this work for additional information
# regarding copyright ownership. The ASF licenses this file
# to you 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.

# Set Hadoop-specific environment variables here.

# The only required environment variable is JAVA_HOME. All others are
# optional. When running a distributed configuration it is best to
# set JAVA_HOME in this file, so that it is correctly defined on
# remote nodes.

# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/

# The jsvc implementation to use. Jsvc is required to run secure datanodes
# that bind to privileged ports to provide authentication of data transfer
# protocol. Jsvc is not required if SASL is configured for authentication of
# data transfer protocol using non-privileged ports.
#export JSVC_HOME=${JSVC_HOME}

export HADOOP_CONF_DIR=${HADOOP_CONF_DIR:-"/etc/hadoop"}

# Extra Java CLASSPATH elements. Automatically insert capacity-scheduler.
for f in SHADOOB_HOME/contrib/capacity-scheduler/*.jar; do
"etc/hadoop/hadoop-env.sh" 98L, 4246C
1,1
Top
```

## 8.Format the Namenode and datanode

hdfs namenode -format

## hdfs datanode -format

```

ubuntu@ip-172-31-14-138:~                               ubuntu@ip-172-31-14-138:~/hadoop
oop/mapreduce/lib/jersey-core-1.9.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/lib/junit-4.11.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-core-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-hs-plugins-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-hs-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-app-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-shuffle-2.9.0.jar:/home/ubuntu/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client-common-2.9.0.jar:/usr/lib/jvm/java-8-openjdk-amd64/lib/tools.jar:/contrib/capacity-scheduler*.jar
STARTUP_MSG: build = https://git-wip-us.apache.org/repos/asf/hadoop.git -r 756ebc8394e473ac25feac05fa493f6d612e6c50; compiled by 'arsuresh' on 2017-11-13T23:15Z
STARTUP_MSG: java = 1.8.0_151
*****
17/12/05 03:21:05 INFO datanode.DataNode: registered UNIX signal handlers for [TERM, HUP, INT]
Usage: hdfs datanode [-regular | -rollback | -rollingupgrade rollback ]
  -regular           : Normal DataNode startup (default).
  -rollback          : Rollback a standard or rolling upgrade.
  -rollingupgrade rollback : Rollback a rolling upgrade operation.
  Refer to HDFS documentation for the difference between standard and rolling upgrades.

17/12/05 03:21:05 WARN datanode.DataNode: Exiting Datanode
17/12/05 03:21:05 INFO util.ExitUtil: Exiting with status 1: ExitException
17/12/05 03:21:05 INFO datanode.DataNode: SHUTDOWN_MSG:
*****
SHUTDOWN_MSG: Shutting down DataNode at ip-172-31-14-138.us-west-2.compute.internal/172.31.14.138
*****
ubuntu@ip-172-31-14-138:~$ jps
1980 FSShell
2454 Jps
ubuntu@ip-172-31-14-138:~$ start-dfs.
start-dfs: command not found
ubuntu@ip-172-31-14-138:~$ start-dfs.sh
17/12/05 03:21:25 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [ec2-35-166-132-238.us-west-2.compute.amazonaws.com]

```

```

ubuntu@ip-172-31-14-138:~                               ubuntu@ip-172-31-14-138:~/hadoop
17/12/05 03:09:24 INFO util.GSet: VM type      = 64-bit
17/12/05 03:09:24 INFO util.GSet: 1.0% max memory 889 MB = 8.9 MB
17/12/05 03:09:24 INFO util.GSet: capacity     = 2^20 = 1048576 entries
17/12/05 03:09:24 INFO namenode.FSDirectory: ACLs enabled? false
17/12/05 03:09:24 INFO namenode.FSDirectory: XAttr enabled? true
17/12/05 03:09:24 INFO namenode.NameNode: Caching file names occurring more than 10 times
17/12/05 03:09:24 INFO snapshot.SnapshotManager: Loaded config captureOpenFiles: false skipCaptureAccessTimeOnlyChange: false
17/12/05 03:09:24 INFO util.GSet: Computing capacity for map cachedBlocks
17/12/05 03:09:24 INFO util.GSet: VM type      = 64-bit
17/12/05 03:09:24 INFO util.GSet: 0.25% max memory 889 MB = 2.2 MB
17/12/05 03:09:24 INFO util.GSet: capacity     = 2^18 = 262144 entries
17/12/05 03:09:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
17/12/05 03:09:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
17/12/05 03:09:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
17/12/05 03:09:24 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
17/12/05 03:09:24 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
17/12/05 03:09:24 INFO util.GSet: Computing capacity for map NameNodeRetryCache
17/12/05 03:09:24 INFO util.GSet: VM type      = 64-bit
17/12/05 03:09:24 INFO util.GSet: 0.02999999329447746% max memory 889 MB = 273.1 KB
17/12/05 03:09:24 INFO util.GSet: capacity     = 2^15 = 32768 entries
Re-format filesystem in Storage Directory /home/ubuntu/hdfs/namenode ? (Y or N) y
17/12/05 03:09:26 INFO namenode.FSImage: Allocated new BlockPoolId: BP-330988773-172.31.14.138-1512443366970
17/12/05 03:09:26 INFO common.Storage: Will remove files: [/home/ubuntu/hdfs/namenode/current/fsimage_00000000000000000000, /home/ubuntu/hdfs/namenode/current/seen_txid, /home/ubuntu/hdfs/namenode/current/fsimage_00000000000000000000, /home/ubuntu/hdfs/namenode/current/VERSION]
17/12/05 03:09:26 INFO common.Storage: Storage directory /home/ubuntu/hdfs/namenode has been successfully formatted.
17/12/05 03:09:27 INFO namenode.FSImageFormatProtobuf: Saving image file /home/ubuntu/hdfs/namenode/current/fsimage.ckpt_00000000000000000000 using no compression
17/12/05 03:09:27 INFO namenode.FSImageFormatProtobuf: Image file /home/ubuntu/hdfs/namenode/current/fsimage.ckpt_00000000000000000000 of size 323 bytes saved in 0 seconds.
17/12/05 03:09:27 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
17/12/05 03:09:27 INFO namenode.NameNode: SHUTDOWN_MSG:
*****
SHUTDOWN_MSG: Shutting down NameNode at ip-172-31-14-138.us-west-2.compute.internal/172.31.14.138
*****
ubuntu@ip-172-31-14-138:~$ 

```

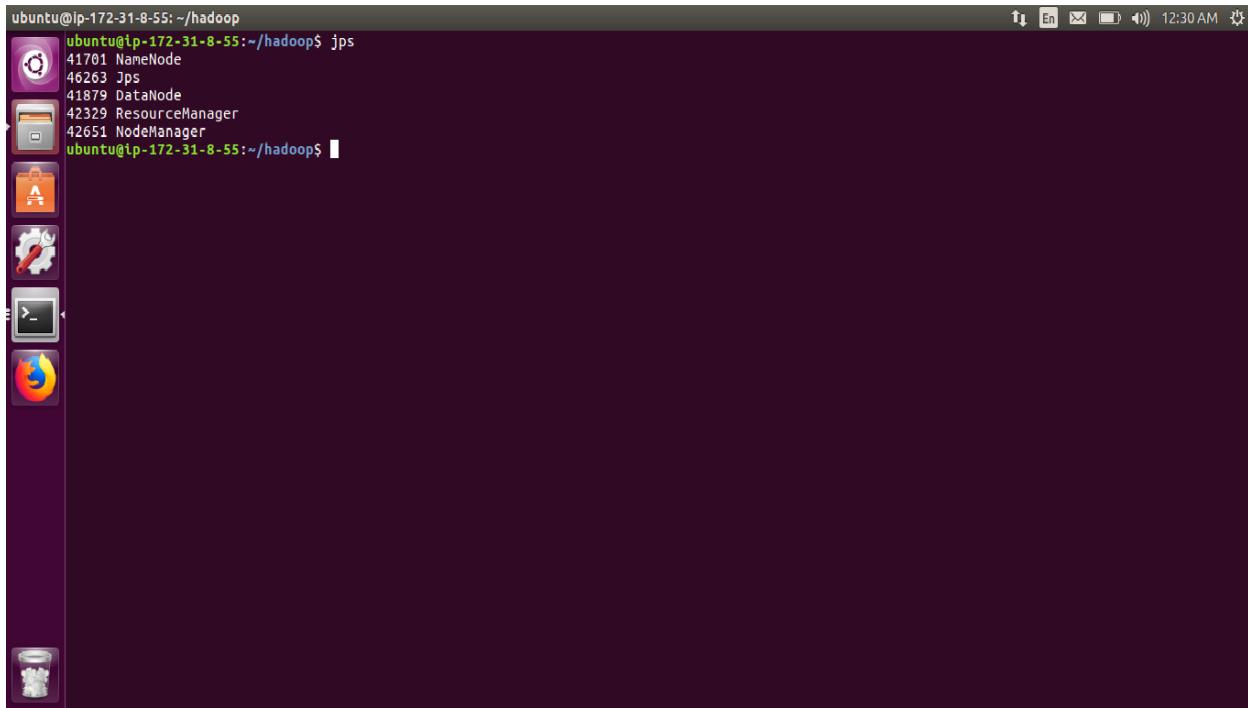
## 9. Start dfs and yarn and generate data

./start-dfs.sh  
 ./start-yarn.sh

```
ubuntu@ip-172-31-14-138:~          ubuntu@ip-172-31-14-138:~          ubuntu@ip-172-31-14-138:~/hadoop
ubuntu@ip-172-31-14-138:~$ start-dfs.sh
17/12/05 03:22:29 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [ec2-35-166-132-238.us-west-2.compute.amazonaws.com]
```

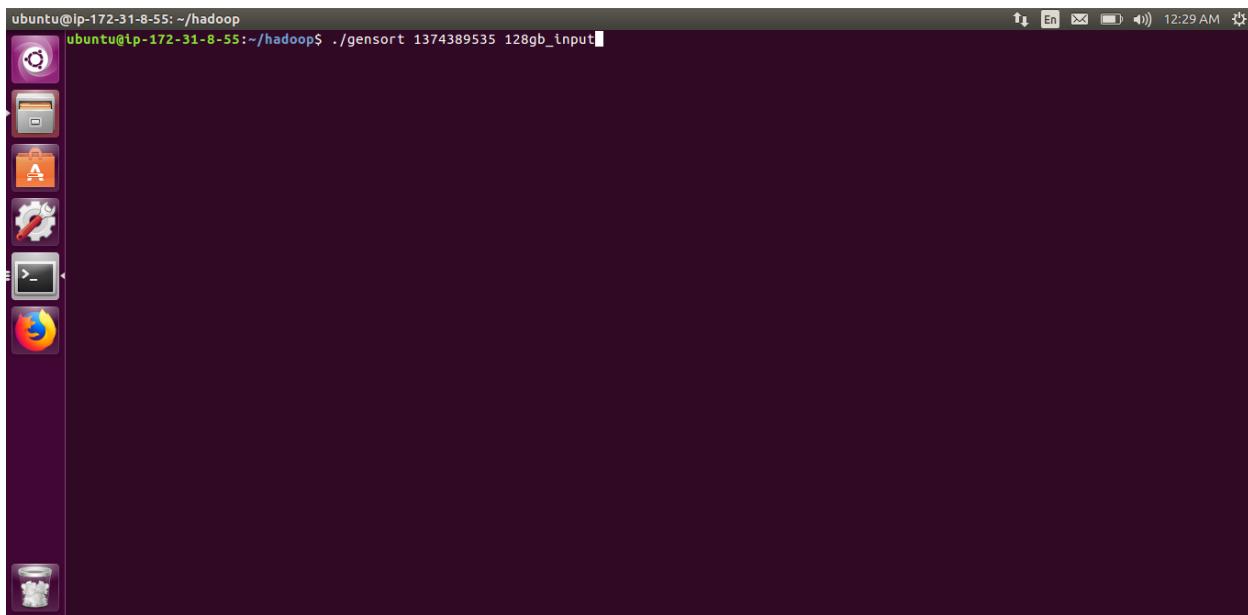
```
ubuntu@ip-172-31-14-138:~          ubuntu@ip-172-31-14-138:~          ubuntu@ip-172-31-14-138:~/hadoop
ubuntu@ip-172-31-14-138:~$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-resourcemanager-ip-172-31-14-138.out
The authenticity of host 'ec2-34-215-253-242.us-west-2.compute.amazonaws.com (172.31.15.55)' can't be established.
ECDSA key fingerprint is SHA256:u8ZGnQ06e2M55ZJa00zXExyC/Nwy/HTRQeh0Ut4ghE.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-215-129-105.us-west-2.compute.amazonaws.com (172.31.7.102)' can't be established.
ECDSA key fingerprint is SHA256:15dG2Gq07sQBv0CqMKYqCBmrV8+ZrRCNZAoqNQ/YDzY.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-161-52.us-west-2.compute.amazonaws.com (172.31.9.5)' can't be established.
ECDSA key fingerprint is SHA256:cJfwQlDRWcznIF/oqHbBHTqXMK5xasjrl7UIZwRUKpU.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-101-97.us-west-2.compute.amazonaws.com (172.31.2.54)' can't be established.
ECDSA key fingerprint is SHA256:VUUvLdSPq5lUcxAO0JooYT3/Tsp4tTQ1XXB+Z6EJLY.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-173-56.us-west-2.compute.amazonaws.com (172.31.13.125)' can't be established.
ECDSA key fingerprint is SHA256:SiX/edz0OqXMDPCK0t5bPcAgrS0sDJlip8c0Kox0wHA.
Are you sure you want to continue connecting (yes/no)? ec2-52-43-58-166.us-west-2.compute.amazonaws.com: Permission denied (publickey).
ec2-35-164-65-19.us-west-2.compute.amazonaws.com: Permission denied (publickey).
localhost: starting nodemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-nodemanager-ip-172-31-14-138.out
```

10. Check to see if all the services are running using  
Jps



ubuntu@ip-172-31-8-55: ~/hadoop  
ubuntu@ip-172-31-8-55:~/hadoop\$ jps  
41701 NameNode  
46263 Jps  
41879 DataNode  
42329 ResourceManager  
42651 NodeManager  
ubuntu@ip-172-31-8-55:~/hadoop\$

11. Generate 128gb data using below command  
. ./gensort -a 1374389535 128gb\_input



ubuntu@ip-172-31-8-55: ~/hadoop  
ubuntu@ip-172-31-8-55:~/hadoop\$ ./gensort 1374389535 128gb\_input

12. now make a directory for input file in hdfs and put the 128gb\_input file into hdfs  
hdfs dfs -mkdir /input  
hdfs dfs -put 128\_Gb\_in /input/

12.Run the terasort.jar to start the mapreduce task.

```
hadoop jar terasort.jar CustomTerasort /input/128_gb_input /output
```

### Output

```
ubuntu@ip-172-31-5-30:~/hadoop-2.2.2
localhost: starting nodemanager, logging to /home/ubuntu/hadoop-2.7.2/logs/yarn-ubuntu-nodemanager-ip-172-31-5-30.out
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2/sbin$ jps
18979 ResourceManager
19417 Jps
18823 SecondaryNameNode
18615 DataNode
19126 NodeManager
18443 NameNode
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2/sbin$ cd /home/ubuntu/hadoop-2.7.2/
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ bin/hadoop fs -mkdirr -p /user/ubuntu/gutenberg
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ ./gensort -a 100000000 /data/input1
bash: ./gensort: No such file or directory
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ cd ..
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ ./gensort -a 100000000 /data/input1
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ cd hadoop-2.7.2/
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ bin/hadoop dfs -copyFromLocal /home/ubuntu/input1 /user/ubuntu/gutenberg
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

copyFromLocal: '/home/ubuntu/input1': No such file or directory
ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ bin/hadoop dfs -copyFromLocal /data/input1 /user/ubuntu/gutenberg
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

ubuntu@ip-172-31-5-30:~/hadoop-2.7.2$ bin/hadoop jar /home/ubuntu/HadoopTerasort.jar terasort /user/ubuntu/gutenberg /user/ubuntu/gutenberg-output
16/03/25 04:43:49 INFO client.RMProxy: Connecting to ResourceManager at ec2-52-23-225-126.compute-1.amazonaws.com/172.31.5.30:9050
16/03/25 04:43:50 INFO client.RMProxy: Connecting to ResourceManager at ec2-52-23-225-126.compute-1.amazonaws.com/172.31.5.30:9050
16/03/25 04:43:50 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
16/03/25 04:43:50 INFO mapred.FileInputFormat: Total input paths to process : 1
16/03/25 04:43:50 INFO mapreduce.JobSubmitter: number of splits:75
16/03/25 04:43:51 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1458880468001_0001
16/03/25 04:43:51 INFO impl.YarnClientImpl: Submitted application application_1458880468001_0001
16/03/25 04:43:51 INFO mapreduce.Job: The url to track the job: http://ec2-52-23-225-126.compute-1.amazonaws.com:9006/proxy/application_1458880468001_0001/
16/03/25 04:43:51 INFO mapreduce.Job: Running job: job_1458880468001_0001
16/03/25 04:44:00 INFO mapreduce.Job: Job job_1458880468001_0001 running in uber mode : false
16/03/25 04:44:00 INFO mapreduce.Job: map 0% reduce 0%
16/03/25 04:44:29 INFO mapreduce.Job: map 1% reduce 0%
```

13. Validate the sorted data using valsor

```
./valsor output
```

```
ubuntu@ip-172-31-8-55:~/hadoop
ubuntu@ip-172-31-8-55:~/hadoop$ ls
CustomTerasort.java          gensort    LICENSE.txt  output      terasort.jar
128gb_input                  CustomTerasort$TerasortMapper.class  include    logs        README.txt
bin                           CustomTerasort$TerasortReducer.class lib        NOTICE.txt  sbin       valsor
CustomTerasort.class          etc                    libexec   onetb_in   share
etc                           part-r-00000             SUCCESS
ubuntu@ip-172-31-8-55:~/hadoop$ ls output/
part-r-00000      _SUCCESS
ubuntu@ip-172-31-8-55:~/hadoop$ ./valsor output/part-r-00000
Records: 1374389535 |
Checksum: 308423c998
Duplicate keys: 0
SUCCESS - all records are in order
ubuntu@ip-172-31-8-55:~/hadoop$
```

### Config 3 (Hadoop Multi-node cluster setup):

For each Slave Node:

1. Update the Hosts file with the private IP and public DNS of master and slave.
2. Update the Slaves file with the public DNS of master and slave.

For the Master Node

1. Update the Hosts file with the private IP and public DNS of master and all the slave nodes.
2. Update the Slaves file with the public DNS of master and all the slave nodes

Steps:

- 1.Create an Ec2 instance in Amazon for Master node and slaves

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (which is selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Images (AMIs and Bundle Tasks), and Elastic Block Store. The main content area has a table titled 'Instances' with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS. There are 9 instances listed, all labeled 'Slave' except for one labeled 'Master Node'. All instances are in the 'running' state. The Public DNS column shows various names like ec2-34-215-123-45. At the bottom of the table, it says 'Select an instance above'.

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
	Slave	i-0005f2b8b1684b849	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-34-215-123-45
	Slave	i-001d9d6e122f5aa6f	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-34-215-123-45
	Slave	i-04c8a592d926de0d8	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-54-187-123-45
	Slave	i-0d72118f52787d668	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-52-24-10-45
	Slave	i-0dc69a59b046c6dec	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-34-214-123-45
	Slave	i-0e86ba905fb97f0f3	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-34-215-123-45
	Slave	i-0fdc93b8a1363aef	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-34-215-123-45
	Master Node	i-03ea618e587c116fc	i3.large	us-west-2c	running	2/2 checks ...	None	ec2-35-16-123-45

- 2.Repeat the above steps from step 1 to step 8. From single node setup to both master node and slave node,

4.On the master node add slaves IP in Hadoop/etc/Hadoop/slavefile

ubuntu@ip-172-31-14-138: ~/hadoop

```
sudhindra@sudhindra-VirtualBox: ~
```

```
localhost
ec2-52-43-58-166.us-west-2.compute.amazonaws.com
ec2-35-104-65-19.us-west-2.compute.amazonaws.com
ec2-34-216-161-52.us-west-2.compute.amazonaws.com
ec2-34-216-173-56.us-west-2.compute.amazonaws.com
ec2-34-215-129-105.us-west-2.compute.amazonaws.com
ec2-34-216-101-97.us-west-2.compute.amazonaws.com
ec2-34-215-253-242.us-west-2.compute.amazonaws.com
```

-- INSERT --

8,51 All

5.In the master's node, add Masters ip in Hadoop/etc/Hadoop/masters file

ubuntu@ip-172-31-14-138: ~/hadoop

```
sudhindra@sudhindra-VirtualBox: ~
```

```
ec2-34-216-91-87.us-west-2.compute.amazonaws.com
```

"etc/hadoop/masters" 1L, 49C

1,1 All

## 6. For each slave now we need to generate RSA key

```
ubuntu@ip-172-31-31-253:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
/home/ubuntu/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa.
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub.
The key fingerprint is:
ds:f8:af:b1:9b:79:cb:03:dd:47:cf:61:d7:89:74:04 ubuntu@ip-172-31-31-253
<The key's randomart image is:
+-[ RSA 2048]----+
|          Eo. |
|         o . . |
|        o o + o |
|       . . . *+|
|      S . . =o=|
|     + ==+|
|      * o|
|     .o .|
|      o.|
+-----+
ubuntu@ip-172-31-31-253:~$ eval "$(ssh-agent)"
Agent pid 1390
ubuntu@ip-172-31-31-253:~$ ssh-add hadoop.pem
Identity added: hadoop.pem (hadoop.pem)
ubuntu@ip-172-31-31-253:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub ubuntu@ec2-52-90-68-182.compute-1.amazonaws.com
The authenticity of host 'ec2-52-90-68-182.compute-1.amazonaws.com (172.31.31.1)' can't be established.
ECDSA key fingerprint is cec6:11:be:cdf:51:4bb7:e0:b0:cf:13:a6:61:5a.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'ubuntu@ec2-52-90-68-182.compute-1.amazonaws.com'"
and check to make sure that only the key(s) you wanted were added.
ubuntu@ip-172-31-31-253:~$ chmod 0600 ~/.ssh/authorized_keys
ubuntu@ip-172-31-31-253:~$
```

## 8. Now generate the RSA in Masters and copy the generated key to all the slaves

Ssh copy-id -l ~/.ssh/id\_rsa.pub ec2-52-43-58-166.us-west-2.compute.amazonaws.com

## 9. Start dfs and yarn

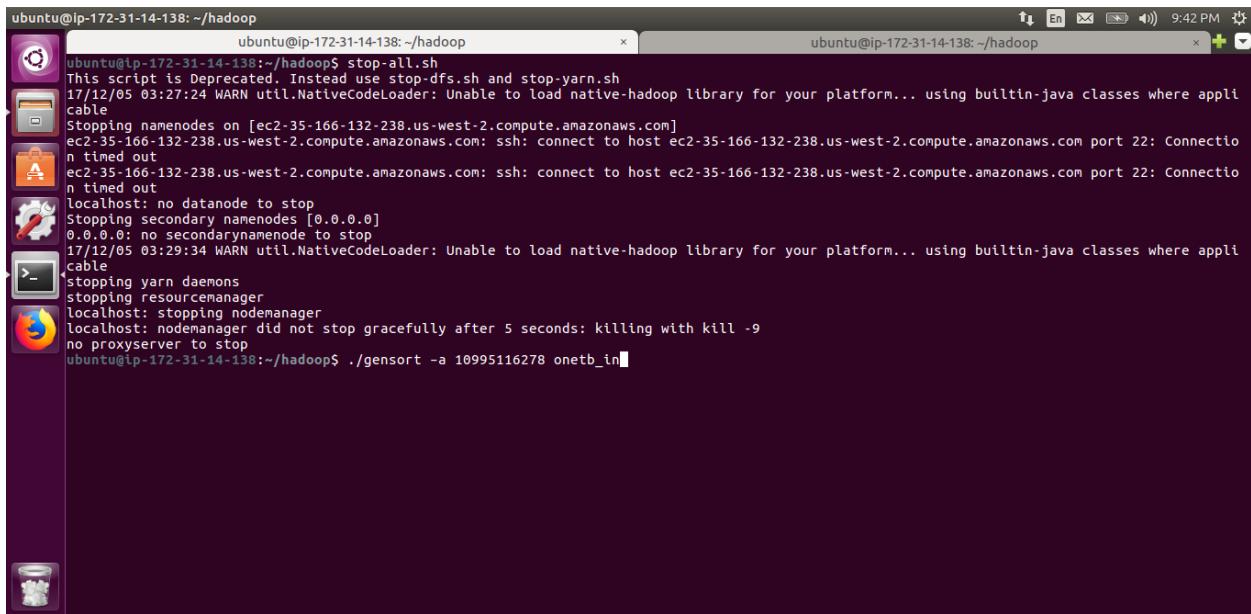
./start-dfs.sh

./start-yarn.sh

```
ubuntu@ip-172-31-14-138:~$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-resourcemanager-ip-172-31-14-138.out
The authenticity of host 'ec2-34-215-253-242.us-west-2.compute.amazonaws.com (172.31.15.55)' can't be established.
ECDSA key fingerprint is SHA256:u8ZGnQ06E2M5Z3a02xExyc/Nwy/HTRQeh0Uf4ghE.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-215-129-105.us-west-2.compute.amazonaws.com (172.31.7.102)' can't be established.
ECDSA key fingerprint is SHA256:15dg2Gq07SQBv0CqMKYqCBmrV8+ZrRCN2AoqNQ/YDzY.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-161-52.us-west-2.compute.amazonaws.com (172.31.9.5)' can't be established.
ECDSA key fingerprint is SHA256:cJfwQlDRWcznIF/oMbBHTqXMK5xasjrl7UIZwRUKpU.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-101-97.us-west-2.compute.amazonaws.com (172.31.2.4)' can't be established.
ECDSA key fingerprint is SHA256:VUVVlLdpqSlucxA00JooYT3/Tsp4TQ1XXB+Z6EJly.
Are you sure you want to continue connecting (yes/no)? The authenticity of host 'ec2-34-216-173-56.us-west-2.compute.amazonaws.com (172.31.13.125)' can't be established.
ECDSA key fingerprint is SHA256:SiX/edz00qqXMDPCK0ts5bPcAgrS0sDJilp8c0KoxOwHA.
Are you sure you want to continue connecting (yes/no)? ec2-52-43-58-166.us-west-2.compute.amazonaws.com: Permission denied (publickey).
ec2-35-164-65-19.us-west-2.compute.amazonaws.com: Permission denied (publickey).
localhost: starting nodemanager, logging to /home/ubuntu/hadoop/logs/yarn-ubuntu-nodemanager-ip-172-31-14-138.out
```

## 10. Generate data using gensort

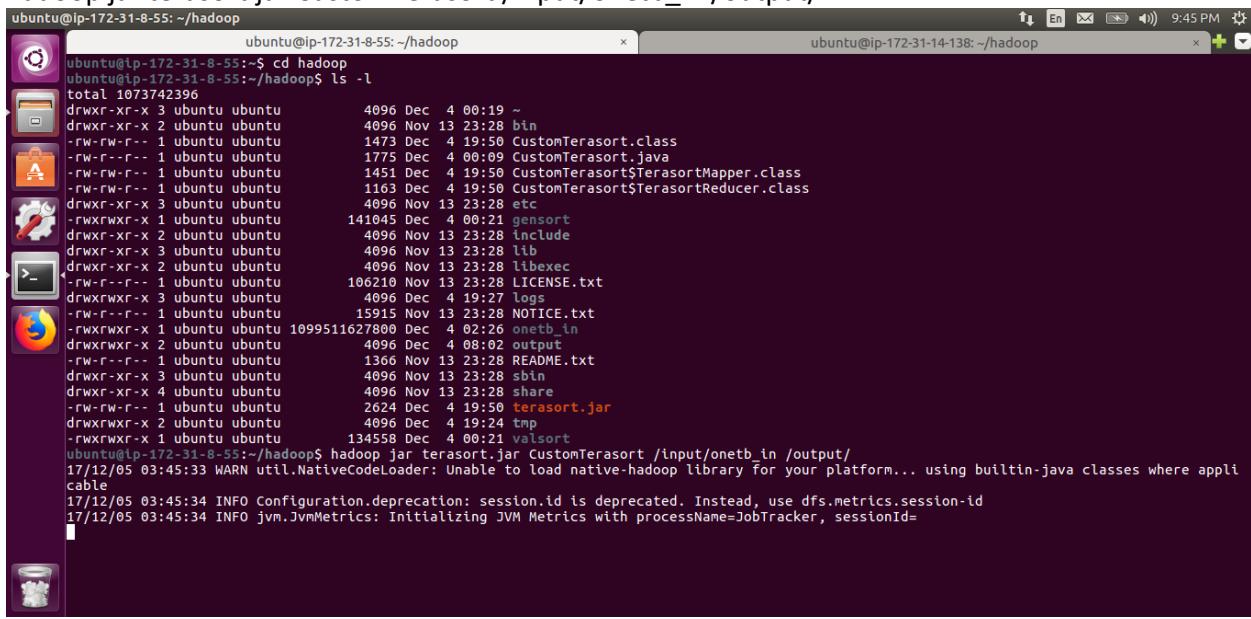
```
./gensort -a 10995116278 onetb_in
```



```
ubuntu@ip-172-31-14-138: ~/hadoop
ubuntu@ip-172-31-14-138:~/hadoop$ stop-all.sh
This script is Deprecated. Instead use stop-dfs.sh and stop-yarn.sh
17/12/05 03:27:24 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Stopping namenodes on [ec2-35-166-132-238.us-west-2.compute.amazonaws.com]
ec2-35-166-132-238.us-west-2.compute.amazonaws.com: ssh: connect to host ec2-35-166-132-238.us-west-2.compute.amazonaws.com port 22: Connection timed out
ec2-35-166-132-238.us-west-2.compute.amazonaws.com: ssh: connect to host ec2-35-166-132-238.us-west-2.compute.amazonaws.com port 22: Connection timed out
localhost: no datanode to stop
Stopping secondary namenodes [0.0.0.0]
0.0.0.0: no secondarynamenode to stop
17/12/05 03:29:34 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
stopping yarn daemons
stopping resourcemanager
localhost: stopping nodemanager
localhost: nodemanager did not stop gracefully after 5 seconds: killing with kill -9
no proxyserver to stop
ubuntu@ip-172-31-14-138:~/hadoop$ ./gensort -a 10995116278 onetb_in
```

## 11. Run the terasort jar

```
hadoop jar terasort.jar CustomTerasort /input/onetb_in /output/
```



```
ubuntu@ip-172-31-8-55: ~/hadoop
ubuntu@ip-172-31-8-55:~/hadoop$ cd hadoop
ubuntu@ip-172-31-8-55:~/hadoop$ ls -l
total 1073742396
drwxr-xr-x 3 ubuntu ubuntu      4096 Dec  4 00:19 ~
drwxr-xr-x 2 ubuntu ubuntu      4096 Nov 13 23:28 bin
-rw-r--r-- 1 ubuntu ubuntu     1473 Dec  4 19:50 CustomTerasort.class
-rw-r--r-- 1 ubuntu ubuntu    1775 Dec  4 00:09 CustomTerasort.java
-rw-r--r-- 1 ubuntu ubuntu    1451 Dec  4 19:50 CustomTerasort$TerasortMapper.class
-rw-r--r-- 1 ubuntu ubuntu    1163 Dec  4 19:50 CustomTerasort$TerasortReducer.class
drwxr-xr-x 3 ubuntu ubuntu      4096 Nov 13 23:28 etc
-rwxrwxr-x 1 ubuntu ubuntu   141045 Dec  4 00:21 gensort
drwxrwxr-x 2 ubuntu ubuntu      4096 Nov 13 23:28 include
drwxr-xr-x 3 ubuntu ubuntu      4096 Nov 13 23:28 lib
drwxr-xr-x 2 ubuntu ubuntu      4096 Nov 13 23:28 libexec
-rw-r--r-- 1 ubuntu ubuntu   106210 Nov 13 23:28 LICENSE.txt
drwxrwxr-x 3 ubuntu ubuntu      4096 Dec  4 19:27 logs
-rw-r--r-- 1 ubuntu ubuntu   15915 Nov 13 23:28 NOTICE.txt
-rwxrwxr-x 1 ubuntu ubuntu 1099511627800 Dec  4 02:26 onetb_in
drwxrwxr-x 2 ubuntu ubuntu      4096 Dec  4 00:02 output
-rw-r--r-- 1 ubuntu ubuntu    1366 Nov 13 23:28 README.txt
drwxr-xr-x 3 ubuntu ubuntu      4096 Nov 13 23:28 sbin
drwxr-xr-x 4 ubuntu ubuntu      4096 Nov 13 23:28 share
-rw-r--r-- 1 ubuntu ubuntu    2624 Dec  4 19:50 terasort.jar
drwxrwxr-x 2 ubuntu ubuntu      4096 Dec  4 19:24 tmp
-rwxrwxr-x 1 ubuntu ubuntu   134558 Dec  4 00:21 valsort
ubuntu@ip-172-31-8-55:~/hadoop$ hadoop jar terasort.jar CustomTerasort /input/onetb_in /output/
17/12/05 03:45:33 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
17/12/05 03:45:34 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
17/12/05 03:45:34 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
```

## 12.validate the sorted output

```

ubuntu@ip-172-31-8-55:~/hadoop
ubuntu@ip-172-31-8-55:~/hadoop$ ls
CustomTerasort.java          gemsort    LICENSE.txt  output   terasort.jar
CustomTerasort$TerasortMapper.class  include    logs        README.txt  valsrt
CustomTerasort.class          lib        NOTICE.txt  sbin
CustomTerasort$TerasortReducer.class  ltbexec  ometb_in  share
Ubuntu@ip-172-31-8-55:~/hadoop$ ls output/
part-r-00000  SUCCESS
Ubuntu@ip-172-31-8-55:~/hadoop$ ./valsrt output/part-r-00000
Records: 10995116278
Checksum: 308423c998
Duplicate keys: 0
SUCCESS - all records are in order
Ubuntu@ip-172-31-8-55:~/hadoop$ 

```

### Difficulties faced during the process

- When we do not see namenode in the processes running even after we start the processes – We would need to stop all the running processes and restart all the processes again.
- When we do not see the Datanode in the processes running even after we start the processes – We would need to remove a folder which gets created under the Datanode directory. Now stop all the running processes and start the processes.
- Connection resets every few hours

### Formula:

Throughput = (data read +write)/ Compute time

Efficiency = (Time taken to run Hadoop on 8 node / time taken to run Hadoop on 1 node)\*100

Speedup = time taken to run i3.4x large / time taken to run on i3.large

**Table 1: Performance evaluation of TeraSort**

Experiment (instance/dataset)	Shared Memory TeraSort	Hadoop TeraSort	Spark TeraSort	MPI TeraSort
Compute Time (sec) [1xi3.large 128GB]	10080	14372		
Data Read (GB) [1xi3.large 128GB]	171.52	384		
Data Write (GB) [1xi3.large 128GB]	161.28	384		
I/O Throughput (MB/sec) [1xi3.large 128GB]	33.80	54.71		

<b>Compute Time (sec) [1xi3.4xlarge 1TB]</b>	161280	56243		
<b>Data Read (GB) [1xi3.4xlarge 1TB]</b>	8576	3072		
<b>Data Write (GB) [1xi3.4xlarge 1TB]</b>	8064	3072		
<b>I/O Throughput (MB/sec) [1xi3.4xlarge 1TB]</b>	105.65	111.86		
<b>Compute Time (sec) [8xi3.large 1TB]</b>	N/A	42330		
<b>Data Read (GB) [8xi3.large 1TB]</b>	N/A	3072		
<b>Data Write (GB) [8xi3.large 1TB]</b>	N/A	3072		
<b>I/O Throughput (MB/sec) [8xi3.large 1TB]</b>	N/A	13.77		
<b>Speedup (weak scale)</b>	16	3.91		
<b>Efficiency (weak scale)</b>	-	75.26		

We have implemented just Hadoop and Shred memory. Based on the available data we have we can summarize the following

**1. Compare the performance of the three versions of TeraSort (Shared-Memory, Hadoop, Spark, and MPI [EC]) on 1 node scale on two different types of instances as well as on a virtual cluster of 8 nodes, and explain your observations**

**-Compute time evaluation:**

From the above table it clear that Shared memory is really good for when compared with Hadoop single node as it is taking just 2hrs:50 minutes when compared to Hadoop which takes almost 3 hours 55 minutes.

**-Data Read/Data Write Evaluation:**

From the above table it is clear that Shared memory takes less bytes to read and less bytes to write as Hadoop takes almost 3 times of what the input data was given to sort. Since, the shuffle sort takes more data to read and write, we can conclude that Shared memory is better.

**- I/O Throughput**

From the above table it is clear that the throughput of shared memory is slightly less than Hadoop which is better for a data having 128 GB.

**2. compare the Shared-Memory performance of 1 node to Hadoop and Spark TeraSort at 8 node scales and explain your observations**

At one node from the above data Shared Memory performs really well and the execution time for 128 GB data is really good in i3.large ec2 instance. Data read, and write is also less as it takes almost half of what Hadoop single node took.

**4. What conclusions can you draw? Which seems to be best at 1 node scale? How about 8 nodes? Can you predict which would be best at 100 node scale? How about 1000 node scales?**

Observation:

1. It is observed that the speed up is greater than 1 for Hadoop 8 node with respect to Hadoop 1-node.
2. It follows the expected trend.

At 1 node we can see that Shared memory is better. At 8 node Hadoop is better as the computing is shared between datanodes from namenodes. So, considering the current trend, its clear that even at 1000 node Hadoop is better.

**5.Compare your results with those from the Sort Benchmark [9], specifically the winners in 2013 and 2014 who used Hadoop and Spark**

Past winner Hadoop Performance: 102.5 TB in 4,328 seconds 2100 nodes

Our performance: 1 TB in 42330 seconds 8 nodes.

**References:**

<http://www.novixys.com/blog/setup-apache-hadoop-cluster-aws-ec2/>

<http://www.ordinal.com/gensort.html>

<http://michael-noll.com/blog/2011/04/09/benchmarking-and-stress-testing-an-hadoop-cluster-with-terasort-testdfsio-nnbench-mrbench/>

<https://aws.amazon.com/blogs/aws/now-available-i3-instances-for-demanding-io-intensive-applications/>

<https://doctuts.readthedocs.io/en/latest/hadoop.html>

[https://en.wikipedia.org/wiki/External\\_sorting](https://en.wikipedia.org/wiki/External_sorting)

[https://en.wikipedia.org/wiki/Merge\\_sort](https://en.wikipedia.org/wiki/Merge_sort)

<http://sortbenchmark.org/>