Programming Assignment 2 Report

Team members:

Vaishnavi Gudur(A20384900)

Mekala Satya Suveen Reddy(A20379568)

Problem :

Programming Assignment 2 aims to sort two datasets, one of 128GB size and the other one at 1TB size using 4 methodologies, Shared Memory, Hadoop,Spark and MPI.

Methodologies used for Terasort application :

1. Shared Memory
2. Hadoop
3. Spark
4. MPI

Server Used :
AWS EC2 instances : i3.large, i3.4xlarge

Configuration of EC2 instance:
AMI : **Ubuntu Server 16.04 LTS (HVM), SSD Volume Type** - ami-aa2ea6d0
No of vCores : 2
No of threads per core : 2
Instance type : i3.large, i3.xlarge

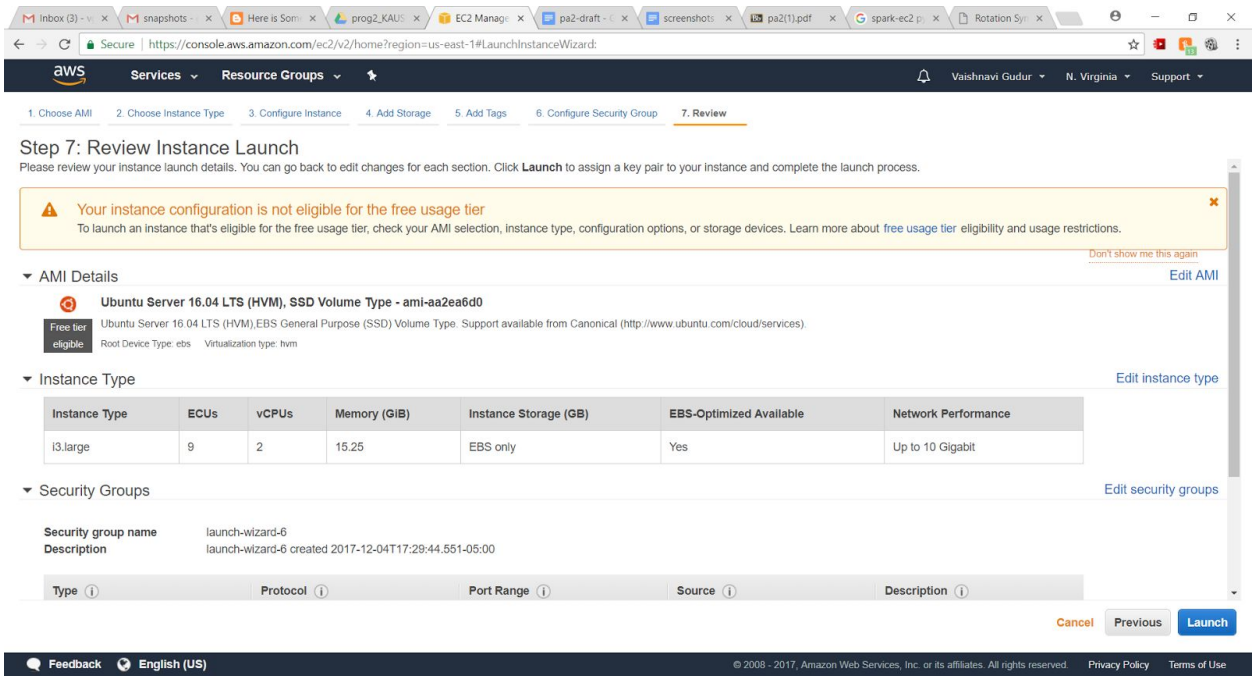

Fig: i3.large instance conf.

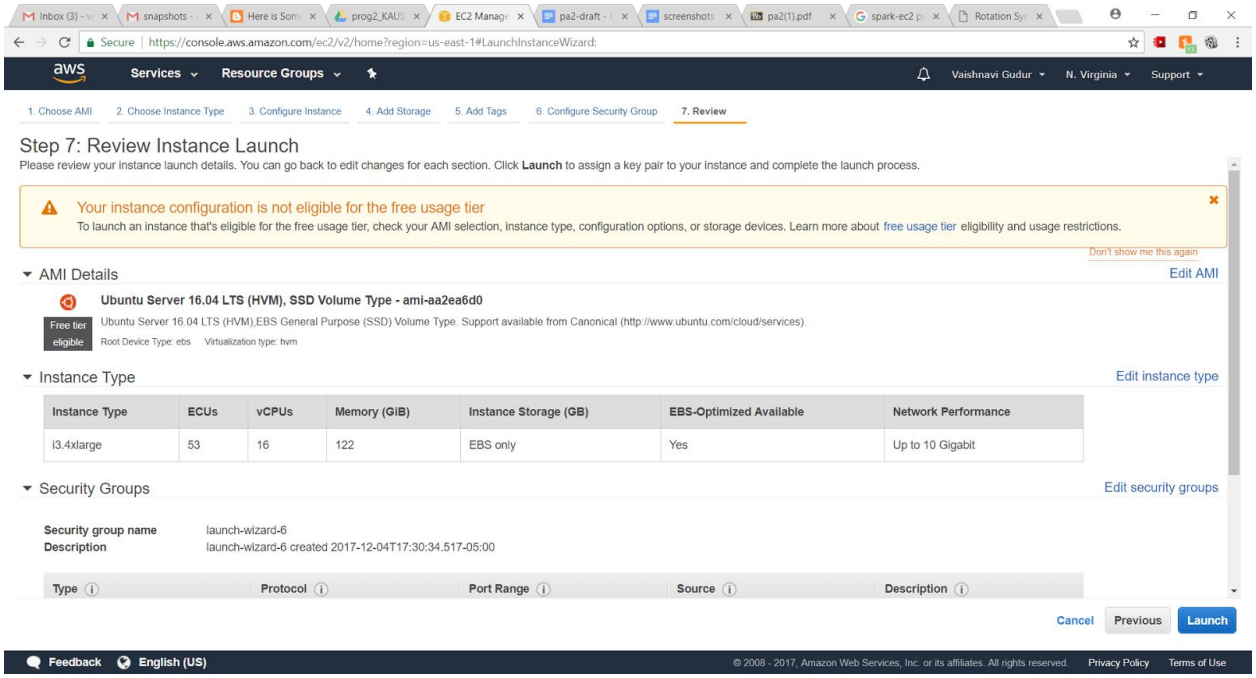

Fig : i3.4xlarge instance conf.

The following steps were followed during the configuration of every instance:

1. Make ‘pa1_gudur_reddy.pem’ private using the following command
   sudo chmod 400 pa1_gudur_reddy
   Since we worked on separate components over the whole assignment, we generated separate keys to connect to our respective instances.
2. Sudo apt-get update
   Sudo apt-get install java-8-openjdk-amd64
3. Now we need to mount SSD to our instance. It can be done in the following way:

   sudo mdadm --create --verbose --force /dev/md0 --level=0 --name=raidLocal
   --raid-devices=1 /dev/nvme0n1
   sudo mdadm --detail /dev/md0
   sudo mkfs.ext4 -L raidLocal /dev/md0
   sudo mkdir -p /mnt/raid
   sudo mount LABEL=raidLocal /mnt/raid
   cd /mnt/raid

Data Generation :

Data is generated using gensort, directly on /mnt/raid .
Gensort generated data in terms of records of 100 bytes each. Hence, for 128 GB data generation, following command can be used:

./gensort -a 137438953472 data.txt

For 1 TB data file generation,

./gensort -a 1099511627776 data.txt

Shared Memory

Java version- jdk8

Shared memory Terasort is a technique applied when the size of the dataset is larger than memory size, as a result of which sorting of the whole dataset cannot be done at once. The idea of the implementation is as follows:

Sort:

The large dataset is divided into chunks of block size, and read the block size of data in memory and sort them in-place, now write that temporary sorted data into the temporary file which will then be merged and sorted to a single sorted file called ‘external-sorted.txt’.

Merge:

Open all the temp files in the list and read the first line of data from each of the file and sort it based on ASCII value, write it to the final sorted file. Now compare the second value of the temp file from which the minimum sorted value we got before was written to final sorted file, likewise repeat the steps.

Here the dataset is such that out of 100 bytes of each line, first 10 characters are considered as key and rest of them as value. So sorting and merge is done based on key, giving sorted file as output. The code is implemented in Java programming language with multithreading functionality.

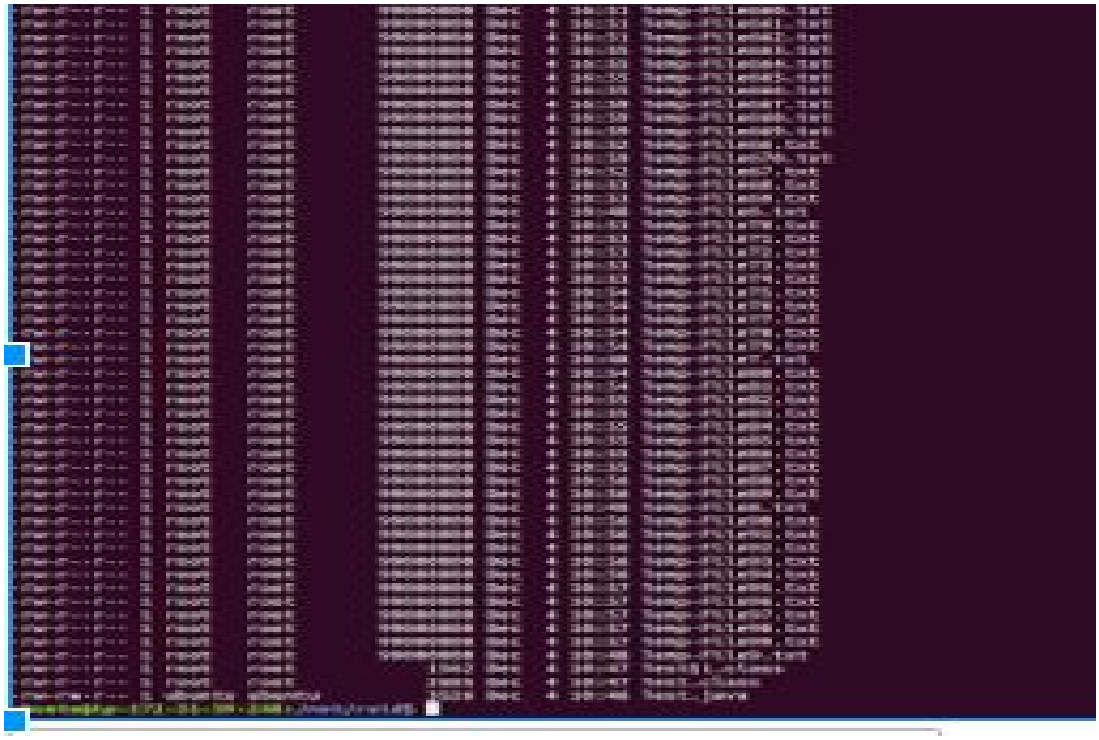

Fig : temp files created

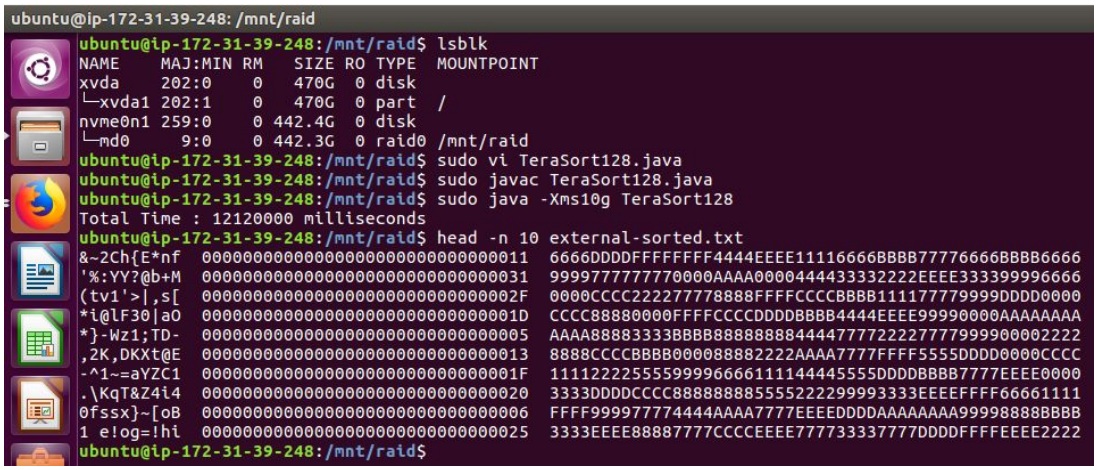

Fig : First 10 lines of sorted file (128 GB)

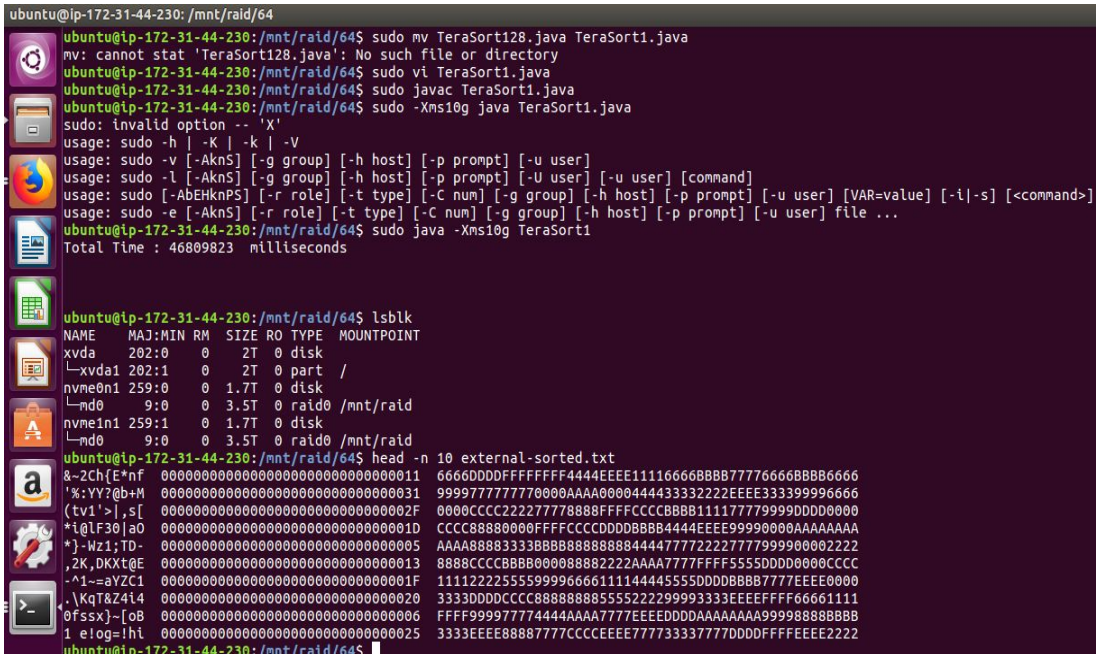

Fig : First 10 lines of sorted file (1 TB)

Observations:

The program runs for 1 thread to 8 threads. From the time and reading it's been observed that as we increase the number of threads, the time required for sorting a dataset is nearly constant.

Shared Memory Results:

Dataset : 128 GB

Dataset : 1 TB

Data Size	Time (milliseconds)
128 GB	**12120s**
1 TB	**46809s**

Thus, throughput achieved can be calculated as,
Throughput for 128 GB = (Data size/time elapsed)
= (128000/12120)
= **10.56** MB/s

Throughput (1 TB data) = (1000000/46809.823)

= 21.36 MB/s

Conclusion:

Thus, the throughput increases when the data size is more and the processing power of the instance is increased because of increased number of cores and memory. Performance achieved is higher when the program is multithreaded as compared to it running on a single thread. On the contrary, if the number of threads are not properly selected then there might be deadlocks and the OS takes more time to process the data.

Apache Hadoop

java version- jdk9, Hadoop- 2.3.0

Apache Hadoop is an open-source software framework used for distributed storage and processing of dataset of big data using the MapReduce programming model.There are two major phases in Hadoop : Map and Reduce.

Map Phase:
During this phase, input data stored in Hadoop FIle System (HDFS) is divided into input splits for analysis by map tasks running in parallel across the Hadoop cluster.Mapper maps input key/value pairs to a set of intermediate key/value pairs.

Reduce Phase:
This phase takes key/value pairs generated in the Map phase, as input and reduces a set of intermediate values which share a key to a smaller set of values.

Hadoop follows a Master-Slave architecture.

Master :
NameNode is the master node in Hadoop. It is responsible for managing the operations of file system namespace like opening,closing,renaming files and determining the mapping of blocks to DataNodes.

Slave :
DataNodes are the slaves in Hadoop architecture. They are responsible for serving read and write requests from the file system's clients along with performing block creation,deletion and replication upon instruction from Master.

The following are the configuration files set to run the code on a single node virtual cluster:

1. **Core-site.xml**

```
<property>
<name>fs.default.name</name>
<value>hdfs://ec2-54-205-252-21.compute-1.amazonaws.com:9000</value>
</property>

<property>
<name>dfs.data.dir</name>
<value>/mnt/raid/data</value>
</property>
```

```
<property>
<name>dfs.name.dir</name>
<value>/mnt/raid/name</value>
</property>

<property>
<name>hadoop.tmp.dir</name>
<value>/mnt/raid/data/tmp</value>
</property>
```

2. Hdfs-site.xml

```
<configuration>
<property>
  <name>dfs.data.dir</name>
  <value>/mnt/raid/data</value>
</property>
<property>
<name>dfs.name.dir</name>
<value>/mnt/raid/name</value>
</property>

<property>
<name>dfs.replication</name>
<value>1</value>
</property>

<property>
<name>dfs.permission</name>
<value>false</value>
</property>

</configuration>
```

3. Mapred-site.xml

```
<configuration>

<property>
<name>mapreduce.jobtracker.address</name>
<value>hdfs://ec2-54-205-252-21.compute-1.amazonaws.com:9001</value>
<description>The host and the port that the map reduce job tracker runs at. If "local",
then jobs are run in process as a single map and reduce task.</description>
```

```
</property>

<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
<description>The framework for running map reduce jobs.</description>
</property>

<property>
   <name>mapreduce.cluster.local.dir</name>
   <value>/mnt/raid/local</value>
</property>
<property>
   <name>mapreduce.jobtracker.system.dir</name>
   <value>/mnt/raid/local</value>
</property>
<property>
   <name>mapreduce.jobtracker.staging.root.dir</name>
   <value>/mnt/raid/local</value>
</property>
<property>
  <name>mapreduce.cluster.temp.dir</name>
  <value>/mnt/raid/local</value>
</property>
```

4. Yarn-site.xml

```
<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>

<property>
```

```
<name>yarn.resourcemanager.scheduler.address</name>
<value>ec2-54-205-252-21.compute-1.amazonaws.com:8030</value>
</property>

<property>
<name>yarn.resourcemanager.address</name>
<value>ec2-54-205-252-21.compute-1.amazonaws.com:8032</value>
</property>

<property>
<name>yarn.resourcemanager.webapp.address</name>
<value>ec2-54-205-252-21.compute-1.amazonaws.com:8088</value>
</property>

<property>
<name>yarn.resourcemanager.resource-tracker.address</name>
<value>ec2-54-205-252-21.compute-1.amazonaws.com:8031</value>
</property>

<property>
<name>yarn.resourcemanager-admin.address</name>
<value>ec2-54-205-252-21.compute-1.amazonaws.com:8033</value>
</property>

<property>
<name>yarn.nodemanager.resource.memory</name>
<value>3072</value>
</property>

</configuration>
```

5. Slaves

Here, since it is a single node the master acts as its own slave. Hence it is localhost.

The following is the process of configuring the Hadoop cluster.

```
ubuntu@ip-172-31-45-62:~$ sudo su
root@ip-172-31-45-62:/home/ubuntu# sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/main Sources [868 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/restricted Sources [4,808 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/universe Sources [7,728 kB]
Get:7 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/multiverse Sources [179 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [7,532 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/universe Translation-en [4,354 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [144 kB]
Get:12 http://security.ubuntu.com/ubuntu xenial-security/main Sources [103 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial/multiverse Translation-en [106 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main Sources [285 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/restricted Sources [3,404 B]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/universe Sources [184 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/multiverse Sources [7,600 B]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [670 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main Translation-en [280 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [561 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/universe Translation-en [227 kB]
Get:22 http://security.ubuntu.com/ubuntu xenial-security/restricted Sources [2,600 B]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [16.2 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/multiverse Translation-en [7,996 B]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/main Sources [3,428 B]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe Sources [4,948 B]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [4,860 B]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/main Translation-en [3,220 B]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [6,624 B]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [3,768 B]
Get:31 http://security.ubuntu.com/ubuntu xenial-security/universe Sources [46.0 kB]
Get:32 http://security.ubuntu.com/ubuntu xenial-security/multiverse Sources [1,140 B]
Get:33 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [397 kB]
Get:34 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [175 kB]
Get:35 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [183 kB]
Get:36 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [96.5 kB]
Get:37 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [3,212 B]
Get:38 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [1,336 B]
Fetched 24.5 MB in 4s (5,943 kB/s)
Reading package lists... Done
root@ip-172-31-45-62:/home/ubuntu# lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
xvda    202:0    0     8G  0 disk
└─xvda1 202:1    0     8G  0 part /
nvme0n1 259:0    0 442.4G  0 disk
root@ip-172-31-45-62:/home/ubuntu# mdadm --create --verbose --force /dev/md0 --level=0 --name=MY_RAID --raid-devices=1 /dev/nvme0n1
mdadm: chunk size defaults to 512K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
root@ip-172-31-45-62:/home/ubuntu# mkfs.ext4 -L MY_RAID /dev/md0
mke2fs 1.42.13 (17-May-2015)
Discarding device blocks: ^[[C^[[D 70258688/115933952

done
Creating filesystem with 115933952 4k blocks and 28991488 inodes
Filesystem UUID: 8b5fb02c-5a2d-4174-ac57-69dc229e4cbc
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
        102400000



Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

root@ip-172-31-45-62:/home/ubuntu#
root@ip-172-31-45-62:/home/ubuntu#
root@ip-172-31-45-62:/home/ubuntu# mkdir -p /mnt/raid
root@ip-172-31-45-62:/home/ubuntu# mount LABEL=MY_RAID /mnt/raid
root@ip-172-31-45-62:/home/ubuntu# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            7.5G     0  7.5G   0% /dev
tmpfs           1.5G  8.6M  1.5G   1% /run
/dev/xvda1      7.7G 1001M  6.7G  13% /
tmpfs           7.5G     0  7.5G   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           7.5G     0  7.5G   0% /sys/fs/cgroup
tmpfs           1.5G     0  1.5G   0% /run/user/1000
/dev/md0        436G   71M  413G   1% /mnt/raid
root@ip-172-31-45-62:/home/ubuntu# lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
xvda    202:0    0     8G  0 disk
└─xvda1 202:1    0     8G  0 part  /
nvme0n1 259:0    0 442.4G  0 disk
└─md0     9:0    0 442.3G  0 raid0 /mnt/raid
root@ip-172-31-45-62:/home/ubuntu# cd /mnt/raid/
root@ip-172-31-45-62:/mnt/raid# ls
lost+found
root@ip-172-31-45-62:/mnt/raid# sudo apt-get install oracle-java9-installer
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package oracle-java9-installer is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'oracle-java9-installer' has no installation candidate
root@ip-172-31-45-62:/mnt/raid# sudo add-apt-repository ppa:webupd8team/java
 Oracle Java (JDK) Installer (automatically downloads and installs Oracle JDK7 / JDK8 / JDK9). There are no actual Java files in this PPA.

Important -> Why Oracle Java 7 And 6 Installers No Longer Work: http://www.webupd8.org/2017/06/why-oracle-java-7-and-6-installers-no.html

Ubuntu 16.10 Yakkety Yak is no longer supported by Canonical (and thus, Launchpad and this PPA). The PPA supports Ubuntu 17.10, 17.04, 16.04, 14.04 and 12.04.

More info (and Ubuntu installation instructions):
- for Oracle Java 7: http://www.webupd8.org/2012/01/install-oracle-java-jdk-7-in-ubuntu-via.html
- for Oracle Java 8: http://www.webupd8.org/2012/09/install-oracle-java-8-in-ubuntu-via-ppa.html

Debian installation instructions:
- Oracle Java 7: http://www.webupd8.org/2012/06/how-to-install-oracle-java-7-in-debian.html
- Oracle Java 8: http://www.webupd8.org/2014/03/how-to-install-oracle-java-8-in-debian.html

Oracle Java 9 (for both Ubuntu and Debian): http://www.webupd8.org/2015/02/install-oracle-java-9-in-ubuntu-linux.html

Oracle JDK 9 is now considered stable. There are currently only 64bit builds (no other builds are available for download: http://www.oracle.com/technetwork/java/javase/downloads/index.html )
 More info: https://launchpad.net/~webupd8team/+archive/ubuntu/java
Press [ENTER] to continue or ctrl-c to cancel adding it

gpg: keyring `/tmp/tmp54isxarz/secring.gpg' created
gpg: keyring `/tmp/tmp54isxarz/pubring.gpg' created
gpg: requesting key EEA14886 from hkp server keyserver.ubuntu.com
gpg: /tmp/tmp54isxarz/trustdb.gpg: trustdb created
```


Terminal Shell Edit View Window Help
87% Mon Dec 4 8:06 PM Suveen
Desktop — root@ip-172-31-45-62: /mnt/raid/64 — ssh -i HadoopSetup.pem ubuntu@ec2-34-229-208-83.compute-1.amazonaws.com — 204×63
root@ip-172-31-45-62:/mnt/raid# apt-get install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
ssh
0 upgraded, 1 newly installed, 0 to remove and 13 not upgraded.
Need to get 7,076 B of archives.
After this operation, 99.3 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 ssh all 1:7.2p2-4ubuntu2.2 [7,076 B]
Fetched 7,076 B in 0s (287 kB/s)
Selecting previously unselected package ssh.
(Reading database ... 51669 files and directories currently installed.)
Preparing to unpack .../ssh_1%3a7.2p2-4ubuntu2.2_all.deb ...
Unpacking ssh (1:7.2p2-4ubuntu2.2) ...
Setting up ssh (1:7.2p2-4ubuntu2.2) ...
root@ip-172-31-45-62:/mnt/raid# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:sHMDu7cQwxq/khK1ZZqnJtqZzj5VpqXDnOWmnDIt+XI root@ip-172-31-45-62
The key's randomart image is:
+---[RSA 2048]----+
|
|
|
|
|
|
|
|
|
|
+----[SHA256]-----+
root@ip-172-31-45-62:/mnt/raid# cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
root@ip-172-31-45-62:/mnt/raid# chmod 0600 authorized_keys
chmod: cannot access 'authorized_keys': No such file or directory
root@ip-172-31-45-62:/mnt/raid# ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:PMIDkrKo1rM+Z3y3KNq62uOlStfuS1kRH75YnTHMYm8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-1041-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud
13 packages can be updated.
8 updates are security updates.
The programs included with the Ubuntu system are free software;
root@ip-172-31-45-62:~# source ~/.bashrc
root@ip-172-31-45-62:~# vi /usr/local/hadoop/etc/hadoop/core-site.xml
root@ip-172-31-45-62:~# cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml
root@ip-172-31-45-62:~# vi /usr/local/hadoop/etc/hadoop/mapred-site.xml
root@ip-172-31-45-62:~# ls
root@ip-172-31-45-62:~# cd /mnt/raid
root@ip-172-31-45-62:/mnt/raid# ls
hadoop-2.3.0.tar.gz lost+found
root@ip-172-31-45-62:/mnt/raid# cd /usr/local/
root@ip-172-31-45-62:/usr/local# ls
bin etc games hadoop include lib man sbin share src
root@ip-172-31-45-62:/usr/local# cd hadoop
root@ip-172-31-45-62:/usr/local/hadoop# ls
bin include libexec NOTICE.txt sbin
etc lib LICENSE.txt README.txt share
root@ip-172-31-45-62:/usr/local/hadoop# cd ..
root@ip-172-31-45-62:/usr/local# mv hadoop /mnt/raid
root@ip-172-31-45-62:/usr/local#
root@ip-172-31-45-62:/usr/local# ls
bin etc games include lib man sbin share src
root@ip-172-31-45-62:/usr/local# cd ..
root@ip-172-31-45-62:/usr# cd /mnt/raid
root@ip-172-31-45-62:/mnt/raid# ls
hadoop hadoop-2.3.0.tar.gz lost+found
root@ip-172-31-45-62:/mnt/raid# cd hadoop
root@ip-172-31-45-62:/mnt/raid/hadoop# ls
bin include libexec NOTICE.txt sbin
etc lib LICENSE.txt README.txt share
root@ip-172-31-45-62:/mnt/raid/hadoop# cd etc
root@ip-172-31-45-62:/mnt/raid/hadoop/etc# cd hadoop
root@ip-172-31-45-62:/mnt/raid/hadoop/etc/hadoop# ls
capacity-scheduler.xml httpfs-site.xml
configuration.xsl log4j.properties
container-executor.cfg mapred-env.cmd
core-site.xml mapred-env.sh
hadoop-env.cmd mapred-queues.xml.template
hadoop-env.sh mapred-site.xml
hadoop-metrics2.properties mapred-site.xml.template
hadoop-metrics.properties slaves
hadoop-policy.xml ssl-client.xml.example
hdfs-site.xml ssl-server.xml.example
httpfs-env.sh yarn-env.cmd
httpfs-log4j.properties yarn-env.sh
httpfs-signature.secret yarn-site.xml
root@ip-172-31-45-62:/mnt/raid/hadoop/etc/hadoop# vi core-site.xml
root@ip-172-31-45-62:/mnt/raid/hadoop/etc/hadoop# cd
root@ip-172-31-45-62:~# vi ./bashrc
root@ip-172-31-45-62:~# vi ~./bashrc
root@ip-172-31-45-62:~# vi ~/.bashrc
root@ip-172-31-45-62:~# source ~/.bashrc
root@ip-172-31-45-62:~# hadoop
Usage: hadoop [--config confdir] COMMAND
where COMMAND is one of:
fs run a generic filesystem user client
version print the version
jar <jar> run a jar file
checknative [-a|-h] check native hadoop and compression libraries availability
distcp <srcurl> <desturl> copy file or directories recursively
archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
classpath prints the class path needed to get the
Hadoop jar and the required libraries
daemonlog get/set the log level for each daemon

```
etInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
17/12/05 01:18:55 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Formatting using clusterid: CID-158dc90b-5429-49be-b78c-4591575ecadb
17/12/05 01:18:56 INFO namenode.FSNamesystem: fsLock is fair:true
17/12/05 01:18:56 INFO namenode.HostFileManager: read includes:
HostSet(
)
17/12/05 01:18:56 INFO namenode.HostFileManager: read excludes:
HostSet(
)
17/12/05 01:18:56 INFO blockmanagement.DatanodeManager: dfs.block.invalidate.limit=1000
17/12/05 01:18:56 INFO blockmanagement.DatanodeManager: dfs.namenode.datanode.registration.ip-hostname-check=true
17/12/05 01:18:56 INFO util.GSet: Computing capacity for map BlocksMap
17/12/05 01:18:56 INFO util.GSet: VM type       = 64-bit
17/12/05 01:18:56 INFO util.GSet: 2.0% max memory 1000 MB = 20 MB
17/12/05 01:18:56 INFO util.GSet: capacity      = 2^21 = 2097152 entries
17/12/05 01:18:56 INFO blockmanagement.BlockManager: dfs.block.access.token.enable=false
17/12/05 01:18:56 INFO blockmanagement.BlockManager: defaultReplication         = 1
17/12/05 01:18:56 INFO blockmanagement.BlockManager: maxReplication             = 512
17/12/05 01:18:56 INFO blockmanagement.BlockManager: minReplication             = 1
17/12/05 01:18:56 INFO blockmanagement.BlockManager: maxReplicationStreams      = 2
17/12/05 01:18:56 INFO blockmanagement.BlockManager: shouldCheckForEnoughRacks  = false
17/12/05 01:18:56 INFO blockmanagement.BlockManager: replicationRecheckInterval = 3000
17/12/05 01:18:56 INFO blockmanagement.BlockManager: encryptDataTransfer        = false
17/12/05 01:18:56 INFO blockmanagement.BlockManager: maxNumBlocksToLog          = 1000
17/12/05 01:18:56 INFO namenode.FSNamesystem: fsOwner             = root (auth:SIMPLE)
17/12/05 01:18:56 INFO namenode.FSNamesystem: supergroup          = supergroup
17/12/05 01:18:56 INFO namenode.FSNamesystem: isPermissionEnabled = true
17/12/05 01:18:56 INFO namenode.FSNamesystem: HA Enabled: false
17/12/05 01:18:56 INFO namenode.FSNamesystem: Append Enabled: true
17/12/05 01:18:56 INFO util.GSet: Computing capacity for map INodeMap
17/12/05 01:18:56 INFO util.GSet: VM type       = 64-bit
17/12/05 01:18:56 INFO util.GSet: 1.0% max memory 1000 MB = 10 MB
17/12/05 01:18:56 INFO util.GSet: capacity      = 2^20 = 1048576 entries
17/12/05 01:18:56 INFO namenode.NameNode: Caching file names occuring more than 10 times
17/12/05 01:18:56 INFO util.GSet: Computing capacity for map cachedBlocks
17/12/05 01:18:56 INFO util.GSet: VM type       = 64-bit
17/12/05 01:18:56 INFO util.GSet: 0.25% max memory 1000 MB = 2.5 MB
17/12/05 01:18:56 INFO util.GSet: capacity      = 2^18 = 262144 entries
17/12/05 01:18:56 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.9990000128746033
17/12/05 01:18:56 INFO namenode.FSNamesystem: dfs.namenode.safemode.min.datanodes = 0
17/12/05 01:18:56 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension     = 30000
17/12/05 01:18:56 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
17/12/05 01:18:56 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
17/12/05 01:18:56 INFO util.GSet: Computing capacity for map Namenode Retry Cache
17/12/05 01:18:56 INFO util.GSet: VM type       = 64-bit
17/12/05 01:18:56 INFO util.GSet: 0.029999999329447746% max memory 1000 MB = 307.2 KB
17/12/05 01:18:56 INFO util.GSet: capacity      = 2^15 = 32768 entries
17/12/05 01:18:56 INFO common.Storage: Storage directory /mnt/raid/hadoop_store/hdfs/namenode has been successfully formatted.
17/12/05 01:18:56 INFO namenode.FSImage: Saving image file /mnt/raid/hadoop_store/hdfs/namenode/current/fsimage.ckpt_0000000000000000000 using no compression
17/12/05 01:18:56 INFO namenode.FSImage: Image file /mnt/raid/hadoop_store/hdfs/namenode/current/fsimage.ckpt_0000000000000000000 of size 216 bytes saved in 0 seconds.
17/12/05 01:18:56 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
17/12/05 01:18:56 INFO util.ExitUtil: Exiting with status 0
17/12/05 01:18:56 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at ip-172-31-45-62.ec2.internal/172.31.45.62
************************************************************/
root@ip-172-31-45-62:/mnt/raid/hadoop/etc/hadoop# start-dfs.sh
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
```

```
tar: Child returned status 2
tar: Error is not recoverable: exiting now
root@ip-172-31-45-62:/mnt/raid# cd 64
root@ip-172-31-45-62:/mnt/raid/64# ./gensort -a 1000 inputfile
root@ip-172-31-45-62:/mnt/raid/64# ls
gensort  inputfile  valsort
root@ip-172-31-45-62:/mnt/raid/64# hadoop fs -mkdir /input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
etInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
17/12/05 01:22:40 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
root@ip-172-31-45-62:/mnt/raid/64# hadoop fs -put inputfile /input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
etInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
17/12/05 01:23:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
root@ip-172-31-45-62:/mnt/raid/64# hadoop fs -ls /input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
etInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
17/12/05 01:23:12 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r--   1 root supergroup     100000 2017-12-05 01:23 /input/inputfile
root@ip-172-31-45-62:/mnt/raid/64# cd ~
root@ip-172-31-45-62:~# cd /mnt/raid/hadoop
root@ip-172-31-45-62:/mnt/raid/hadoop# ls
bin  include  libexec      logs       README.txt  share
etc  lib      LICENSE.txt  NOTICE.txt  sbin
root@ip-172-31-45-62:/mnt/raid/hadoop# cd bin
root@ip-172-31-45-62:/mnt/raid/hadoop/bin# cd ..
root@ip-172-31-45-62:/mnt/raid/hadoop# cd
root@ip-172-31-45-62:~# ls
root@ip-172-31-45-62:~# cd ~
root@ip-172-31-45-62:~# cd /mnt/raid/64
root@ip-172-31-45-62:/mnt/raid/64# ls
gensort  inputfile  valsort
root@ip-172-31-45-62:/mnt/raid/64# cp /home/ubuntu/HadoopTerasortDriver.jar .
root@ip-172-31-45-62:/mnt/raid/64# ls
gensort  HadoopTerasortDriver.jar  inputfile  valsort
root@ip-172-31-45-62:/mnt/raid/64# hadoop fs -mkdir /input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
etInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
17/12/05 01:31:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
mkdir: `/input': File exists
root@ip-172-31-45-62:/mnt/raid/64# ls
gensort  HadoopTerasortDriver.jar  inputfile  valsort
root@ip-172-31-45-62:/mnt/raid/64# hadoop fs -ls /input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/mnt/raid/hadoop/share/hadoop/common/lib/hadoop-auth-2.3.0.jar) to method sun.security.krb5.Config.g
etInstance()
```

```
17/12/05 02:01:30 INFO mapreduce.Job: The url to track the job: http://ip-172-31-45-62:8088/proxy/application_1512436773081_0002/
17/12/05 02:01:30 INFO mapreduce.Job: Running job: job_1512436773081_0002
17/12/05 02:01:37 INFO mapreduce.Job: Job job_1512436773081_0002 running in uber mode : false
17/12/05 02:01:37 INFO mapreduce.Job:  map 0% reduce 0%
17/12/05 02:01:46 INFO mapreduce.Job:  map 100% reduce 0%
17/12/05 02:01:52 INFO mapreduce.Job:  map 100% reduce 100%
17/12/05 02:01:53 INFO mapreduce.Job: Job job_1512436773081_0002 completed successfully
17/12/05 02:01:53 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=102006
                FILE: Number of bytes written=460481
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=103426
                HDFS: Number of bytes written=100000
                HDFS: Number of read operations=9
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=14569
                Total time spent by all reduces in occupied slots (ms)=3708
                Total time spent by all map tasks (ms)=14569
                Total time spent by all reduce tasks (ms)=3708
                Total vcore-seconds taken by all map tasks=14569
                Total vcore-seconds taken by all reduce tasks=3708
                Total megabyte-seconds taken by all map tasks=14918656
                Total megabyte-seconds taken by all reduce tasks=3796992
        Map-Reduce Framework
                Map input records=1000
                Map output records=1000
                Map output bytes=100000
                Map output materialized bytes=102012
                Input split bytes=178
                Combine input records=1000
                Combine output records=1000
                Reduce input groups=1000
                Reduce shuffle bytes=102012
                Reduce input records=1000
                Reduce output records=1000
                Spilled Records=2000
                Shuffled Maps =2
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=88
                CPU time spent (ms)=2760
                Physical memory (bytes) snapshot=868040704
                Virtual memory (bytes) snapshot=6360055808
                Total committed heap usage (bytes)=629145600
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=103248
        File Output Format Counters
                Bytes Written=100000
```

1TB DATA on single node

Data Generation

```
root@ip-172-31-45-62:/mnt/raid/64# ./gensort -a 1000000000000 inputfileTB
^Z
[3]+  Stopped                 ./gensort -a 1000000000000 inputfileTB
root@ip-172-31-45-62:/mnt/raid/64# bg
[3]+ ./gensort -a 1000000000000 inputfileTB &
root@ip-172-31-45-62:/mnt/raid/64#
```

```
root@ip-172-31-45-62:/mnt/raid/64# jobs
[3]+  Running                 ./gensort -a 1000000000000 inputfileTB &
root@ip-172-31-45-62:/mnt/raid/64# ls -la
total 247501728
drwxrwxr-x 2  500  500         4096 Dec  5 02:19 .
drwxr-xr-x 7 root root        4096 Dec  5 01:20 ..
-rwxrwxr-x 1  500  500      141045 Mar 17  2013 gensort
-rw-r--r-- 1 root root        2567 Dec  5 01:36 HadoopTerasortDriver.jar
-rw-r--r-- 1 root root        2637 Dec  5 02:00 HTerasort.jar
-rwxr-xr-x 1 root root      100000 Dec  5 01:21 inputfile
-rwxr-xr-x 1 root root 253440819200 Dec  5 02:41 inputfileTB
-rwxrwxr-x 1  500  500      134558 Mar 17  2013 valsort
root@ip-172-31-45-62:/mnt/raid/64# sp_write_input: inputfileTB: aio_write() return failure: Success, offset: 466808209408, returned size 401408 != requested size: 524288


[3]+  Exit 1                  ./gensort -a 1000000000000 inputfileTB
```

Map Reduce Phase:

Performance:

Data Size	Time (seconds)
128 GB	**13800**
1 TB	**35057**

Throughput(128 GB) = (Data Size/Time)

= (128000/13800)
= **8.28 MB/s**

Throughput (1 TB) = (Data Size/Time)
= 1000000/35057
= **21.36 MB/s**

Apache Spark

Spark Version- 2.2.0, scala version- 2.11.8, java version- 1.8.0

Apache Spark is a open-source, fast and general cluster-computing framework for large-scale data processing. Spark provides an interface for programming entire clusters with implicit data parallelism and fault tolerance.It uses and extends Hadoop's MapReduce to introduce more types of which includes Interactive Queries and Stream Processing.

Apache Spark has as its architectural foundation the **Resilient Distributed Dataset (RDD)**, a read-only multiset of data items distributed over a cluster of machines, that is maintained in a fault-tolerant way. Since Spark has its own cluster management computation, it uses Hadoop for storage purposes only.

Start the Scala shell using
cd $SPARK_HOME
$SPARK_HOME/bin/spark-shell
This starts the interactive spark shell.

The configuration is as follows:

./gensort -a 12800000000 input128GB
wget http://repo.continuum.io/archive/Anaconda3-4.1.1-Linux-x86_64.sh

bash Anaconda3–4.1.1-Linux-x86_64.sh

source .bashrc
which python
jupyter notebook --generate-config
mkdir certs
cd certs
sudo openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout pa1_gudur_reddy.pem -out pa1_gudur_reddy.pem
cd ~/.jupyter/

vi jupyter_notebook_config.py
c = get_config()
# Notebook config this is where you saved your pem cert
c.NotebookApp.certfile = u'/home/ubuntu/certs/pa1_gudur_reddy.pem'
# Run on all IP addresses of your instance
c.NotebookApp.ip = '*'
# Don't open browser by default
c.NotebookApp.open_browser = False
# Fix port to 8888
c.NotebookApp.port = 8888

jupyter notebook

https://ec2-54-205-252-21.compute-1.amazonaws.com:8888
java -version
sudo apt-get install scala
scala -version

```
sudo dpkg -i scala-2.11.8.deb
export PATH=$PATH:$HOME/anaconda3/bin
conda install pip
which pip
pip install py4j
wget
http://archive.apache.org/dist/spark/spark-2.0.0/spark-2.0.0-bin-hadoop2.7.tgz
sudo tar -zxvf spark-2.0.0-bin-hadoop2.7.tgz
export SPARK_HOME='/home/ubuntu/spark-2.0.0-bin-hadoop2.7'
export PATH=$SPARK_HOME:$PATH
export PYTHONPATH=$SPARK_HOME/python:$PYTHONPATH
jupyter notebook

//Now open a notebook on Jupyter Notebook
python sparksort.py input128GB output
```

Performance evaluation of TeraSort

Experiment (instance/dataset)	Shared Memory TeraSort	Hadoop TeraSort	Spark TeraSort	MPI TeraSort
Compute Time (sec) [1xi3.large 128GB]	12120	13800	18495	
Data Read (GB) [1xi3.large 128GB]	17.49	14.73	8.766	
Data Write (GB) [1xi3.large 128GB]	0.0483	0.0034	0.00194	
I/O Throughput (MB/sec) [1xi3.large 128GB]	10.56	8.98	6.315	
Compute Time (sec) [1xi3.4xlarge 1TB]	46809	35057	54987	
Data Read (GB) [1xi3.4xlarge 1TB]	17.92	14.12	7.978	
Data Write (GB) [1xi3.4xlarge 1TB]	0.233	0.603	0.043	
I/O Throughput (MB/sec) [1xi3.4xlarge 1TB]	21.36	28.52	17.762	
Compute Time (sec) [8xi3.large 1TB]	xxxxxx	13908	18292	
Data Read (GB) [8xi3.large 1TB]	xxxxxx	14.56	12.65	
Data Write (GB) [8xi3.large 1TB]	xxxxxx	1.537	0.8932	
I/O Throughput (MB/sec) [8xi3.large 1TB]	xxxxxx	71.90	58.23	
Speedup (weak scale)	2.03	8.0067	9.22	
Efficiency (weak scale)	49.26%	99.91%	86.75%	

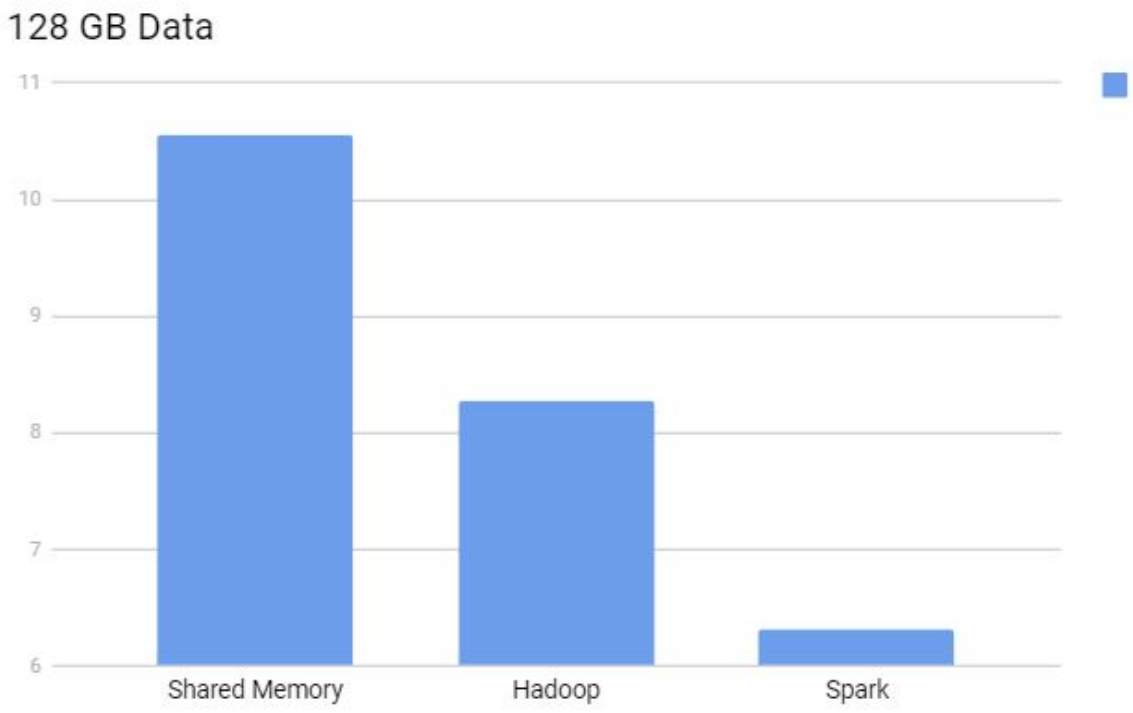

Fig : Throughput

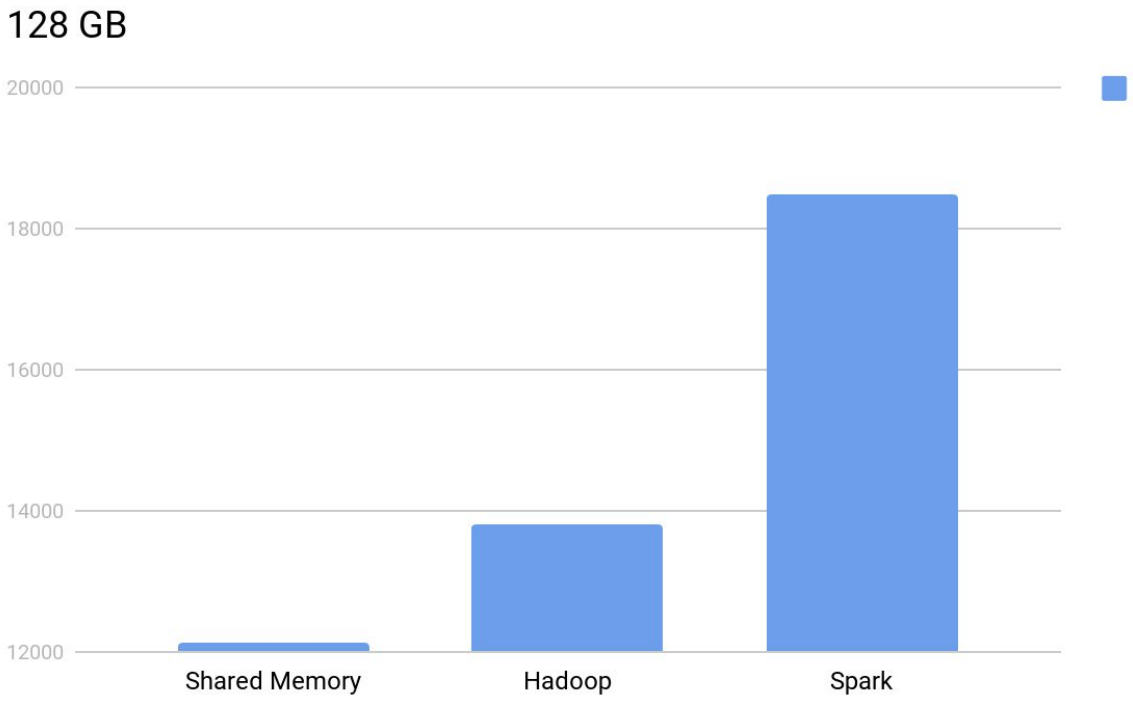

Fig: Time Elapsed

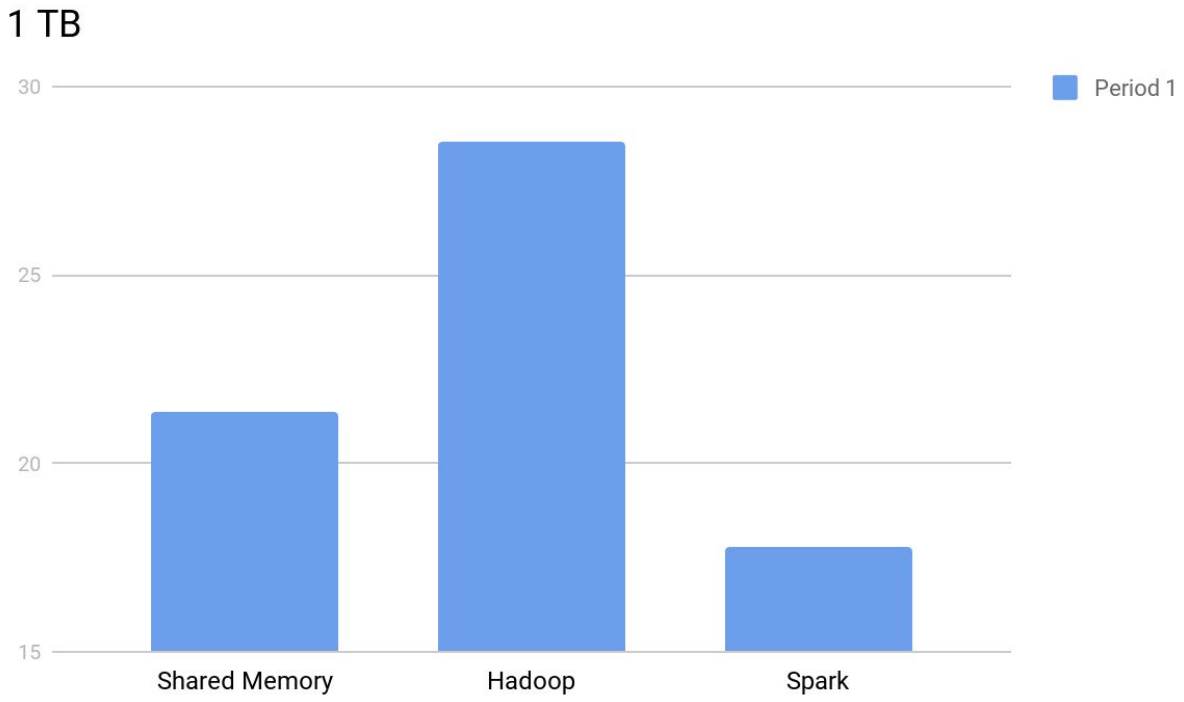

Fig : Throughput

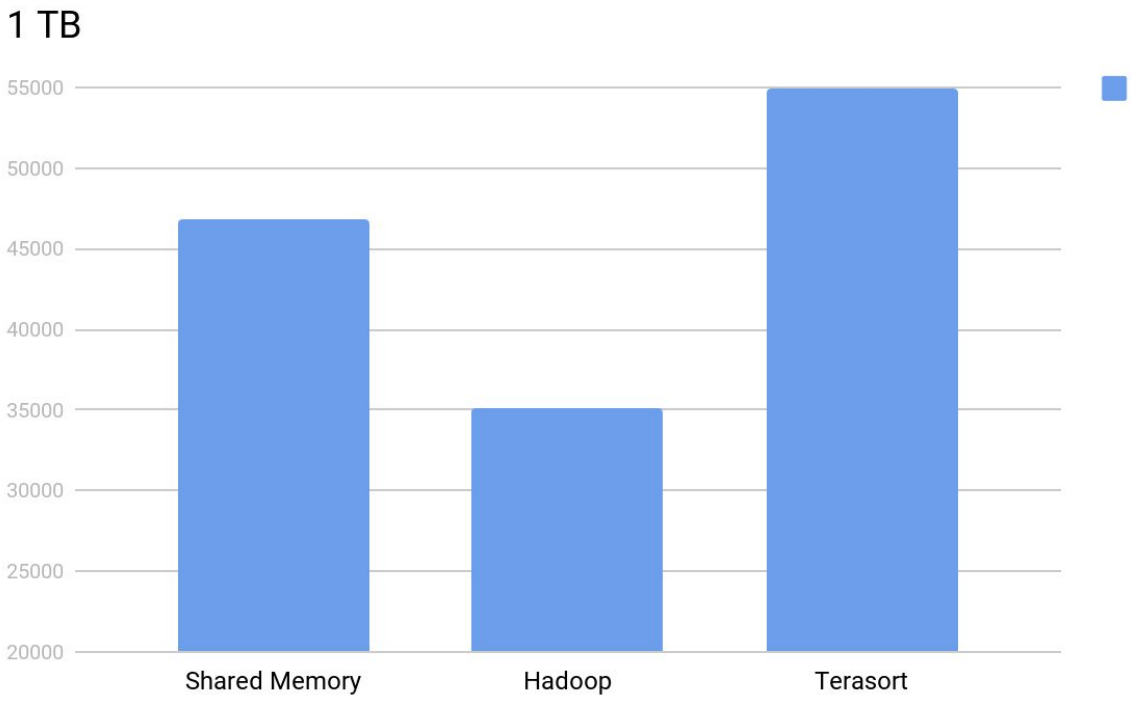

Fig : Time Elapsed

Conclusion:

We can note that the performance for a single node is high on shared memory while the performance is high on 8 node hadoop cluster.

Spark on the other hand performs bad in this context than Hadoop. Spark generally performs better when there are multiple clusters involved. Since we have worked on a single cluster and experimented on it, hadoop performs better.

If the number of nodes increases to 100 and 1000 nodes, the performance will increase too. This is because of the increase in thread count and the combined increase in computing power.

We check the results of the sort benchmark and for the year 2013 and 2014, the winners of the benchmarking are Hadoop and Apache Spark respectively.
Comparing the result of Hadoop system that won the 2013 award for sort, the configuration of the system is much better than the instances that we use for our experiments.

Hence the performance is incomparable as the system specification is better for the Daytona Hadoop. The Hadoop system 2013 winner produced a throughput of 1.42TB/min which is very high as compared to the throughput that we receive with those experiments. Similarly, Apache Spark winner of 2014 produces a throughput of 4.27Tb/min which has system of 32 Virtual cores and a memory of 244GB with 6400 GB SSD. This configuration is pretty less as compared to i3.xlarge instances providing 2 vCPU cores on which this experiments have been tested. Hence we get lesser throughput gradually with respect to the specifications on which both the experiments have been performed.