Hadoop Lab

Server Credentials:

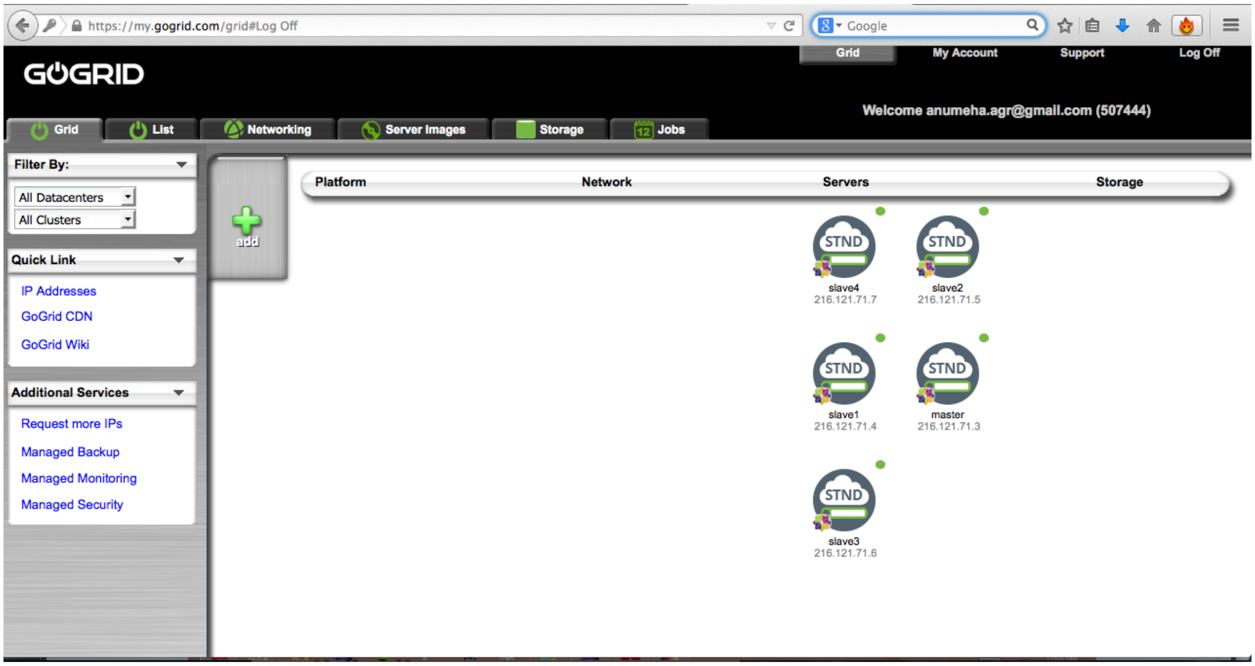
master: Public Ip: 216.121.71.3, root password: Kr1sh@yPr1sh@4ggAnumeh@@*#&%9
slave1: Public Ip: 216.121.71.4, root password: Kr1sh@yPr1sh@4ggAnumeh@@*#&%9
slave2: Public Ip: 216.121.71.5, root password: Kr1sh@yPr1sh@4ggAnumeh@@*#&%9
slave3: Public Ip: 216.121.71.6, root password: Kr1sh@yPr1sh@4ggAnumeh@@*#&%9
slave4: Public Ip: 216.121.71.7, root password: Kr1sh@yPr1sh@4ggAnumeh@@*#&%9

Cloudera manager wep app credentials

admin, S0n1Gu1t@@r4gg0907

Steps for Setting up Cluster

Step1: Created 1 Extra Large 8GB, 8 cores server and 4 4 GB 4 cores server.
Step2: Changed the root password of all the servers.



Steps for Setting up hostname configuration.

Step1: Changed the hostname for each server using hostname <hostname>,

```
ex. hostname master.example.com
```

Step2: Change the value of HOSTNAME in /etc/sysconfig/network to the <hostname>

```
ex. HOSTNAME=master.example.com
```

Step3: Restart the service network using the command

```
# service network restart.
```

Step4: Open /etc/sysconfig/network-scripts/ifcfg-eth1 and make change BOOTPROTO to static and add two more values IPADDR and NETMASK. IPADDR is the private ip of the server.

```
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
IPADDR=10.102.110.3
NETMASK=255.255.255.0
```

Step5: edit etc/hosts file and add the hostnames of all the servers. Delete the second line and add the below lines. here IP address is the private ip of the hosts.

```
10.102.110.3 master.example.com master
10.102.110.4 slave1.example.com slave1
10.102.110.5 slave2.example.com slave2
10.102.110.6 slave3.example.com slave3
10.102.110.7 slave4.example.com slave4
```

Step6: Now run the below command

```
ifup eth1
```

Step7: Next run the below command

```
service iptables stop
```

Step8: Now run the command below command

```
chkconfig iptables off
```

Step9: Make the above hostname configuration changes for all the servers

Steps For Running the cloudera manager installer.

Step1: download the latest cloudera manager installer version on the local machine.

Step2: copy the installer to the master server using SCP

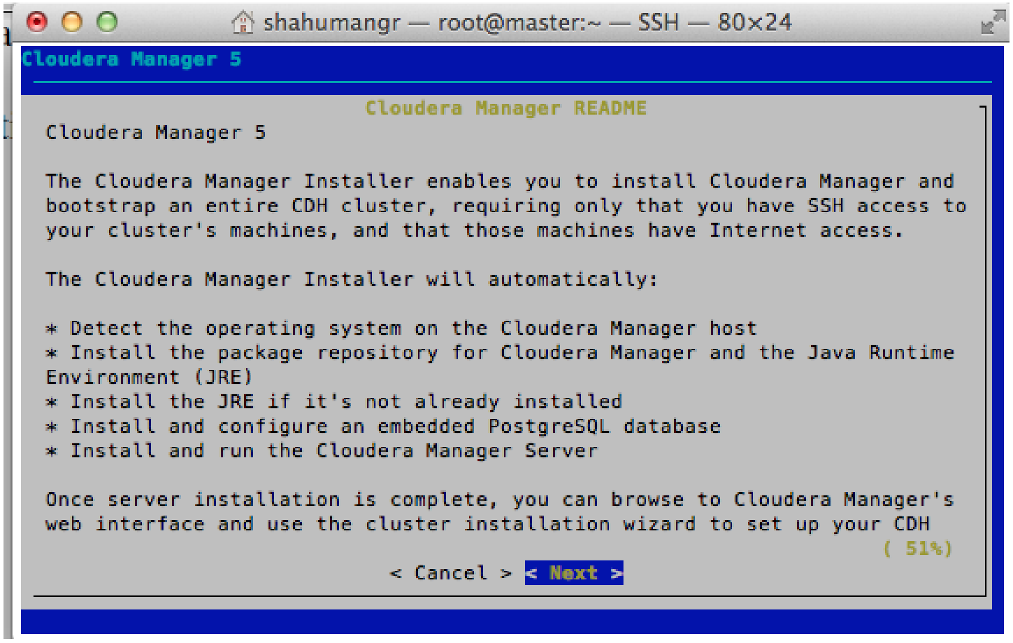
```
scp cloudera-manager-installer.bin root@216.121.71.3:/root
```

Step3: Change the mode to executable of the cloudera manager installer

```
chmod +x cloudera-manager-installer.bin
```

Step4: Execute the installer.

```
# ./cloudera-manager-installer.bin
```



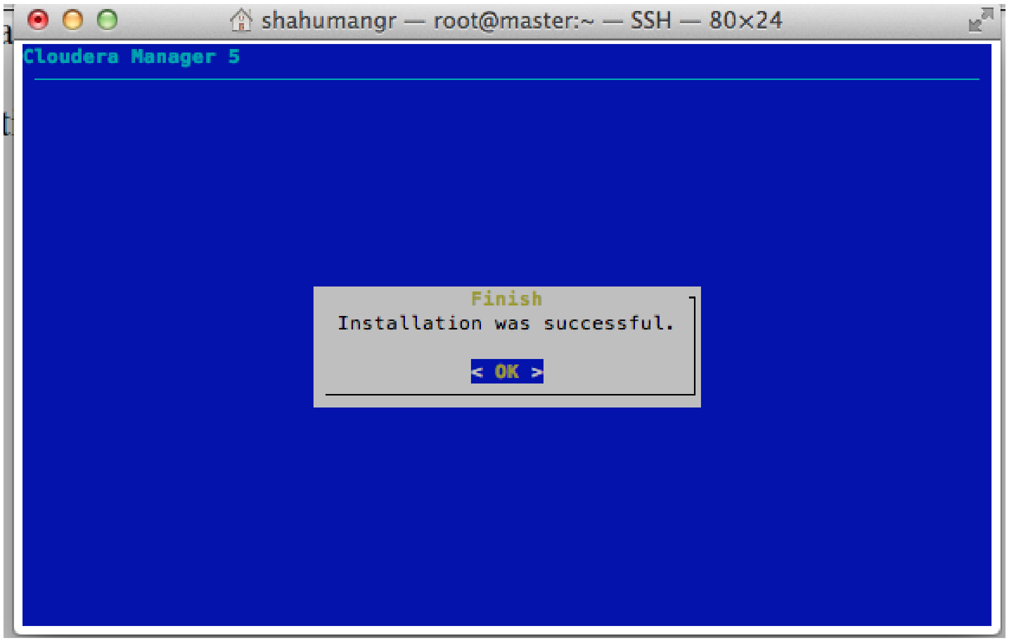
```
Cloudera Manager 5
                          Cloudera Manager README
 Cloudera Manager 5

 The Cloudera Manager Installer enables you to install Cloudera Manager and
 bootstrap an entire CDH cluster, requiring only that you have SSH access to
 your cluster's machines, and that those machines have Internet access.

 The Cloudera Manager Installer will automatically:

 * Detect the operating system on the Cloudera Manager host
 * Install the package repository for Cloudera Manager and the Java Runtime
 Environment (JRE)
 * Install the JRE if it's not already installed
 * Install and configure an embedded PostgreSQL database
 * Install and run the Cloudera Manager Server

 Once server installation is complete, you can browse to Cloudera Manager's
 web interface and use the cluster installation wizard to set up your CDH
                                                                      ( 51%)
                          < Cancel >  < Next >
```



```
Cloudera Manager 5

                         Finish
              Installation was successful.

                         < OK >
```

Steps For Installing CDH using cloudera manager web interface.

Step1: Login to cloudera manager web interface using admin. admin

Step2: Select the cloudera manager Data Hub Edition Trial.

Step3: Next specify hosts for your CDH cluster. Specify the private ips of the servers seperated by comma. Press continue

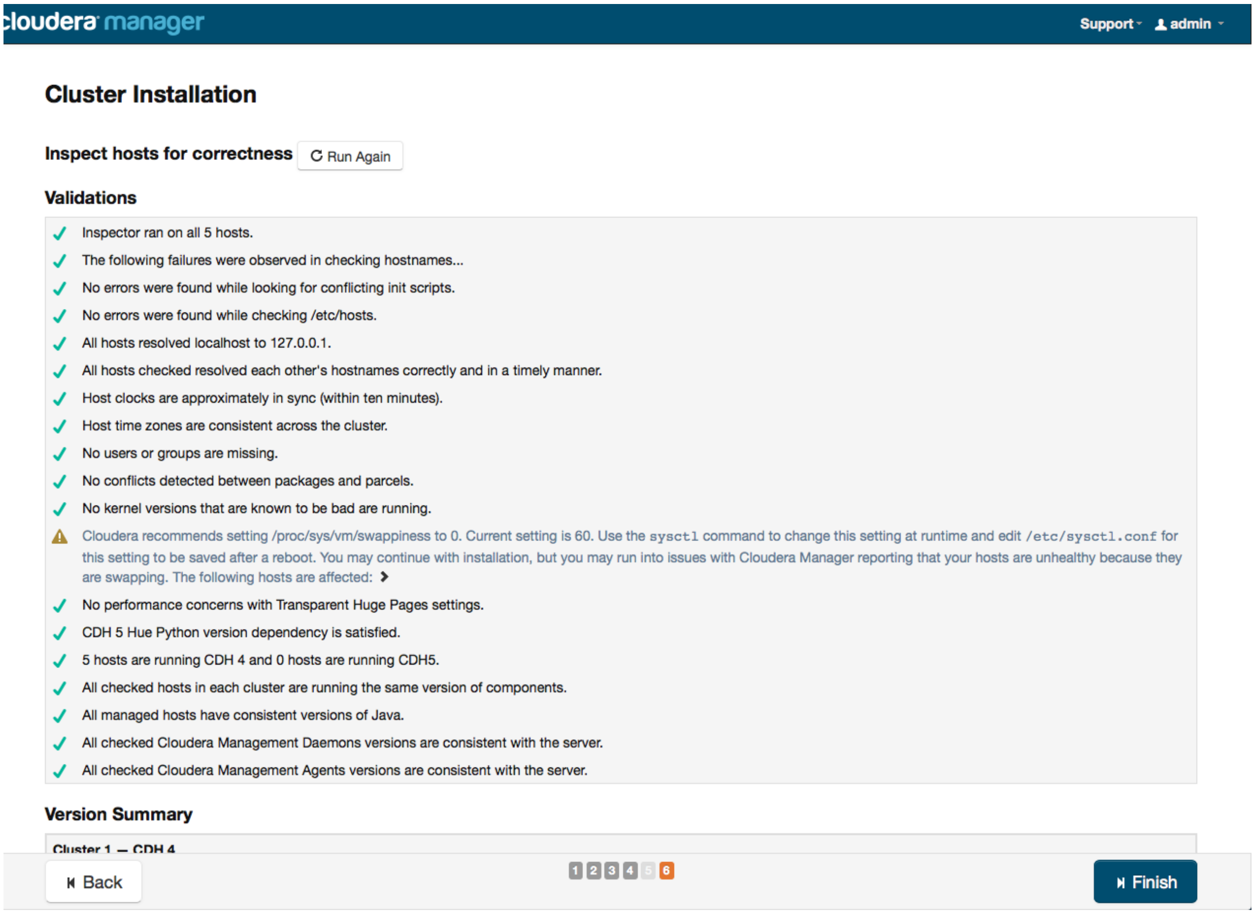
10.102.110.3, 10.102.110.4, 10.102.110.5, 10.102.110.6, 10.102.110.7

Step4: Next choose CDH version. I selected latest CDH 4 as it comes by default with mapreduce version 1 and CDH5 comes with mapreduce version 2 by default and mapreduce version 1 is deprecated in CDH5 but has been added for backward compatibility. Press continue

Step5: Next provide cluster information and select Login To All Hosts As: root and provide the root password for the cluster and the SSH Port which is 22 by default until and unless you changed the default SSH port. Press Continue

Once the cluster installation is finished press Continue.

Step 6: On Next page you can inspect the hosts for correctness.



Cluster Installation

Inspect hosts for correctness

Validations

- ✓ Inspector ran on all 5 hosts.
- ✓ The following failures were observed in checking hostnames...
- ✓ No errors were found while looking for conflicting init scripts.
- ✓ No errors were found while checking /etc/hosts.
- ✓ All hosts resolved localhost to 127.0.0.1.
- ✓ All hosts checked resolved each other's hostnames correctly and in a timely manner.
- ✓ Host clocks are approximately in sync (within ten minutes).
- ✓ Host time zones are consistent across the cluster.
- ✓ No users or groups are missing.
- ✓ No conflicts detected between packages and parcels.
- ✓ No kernel versions that are known to be bad are running.
- ⚠ Cloudera recommends setting /proc/sys/vm/swappiness to 0. Current setting is 60. Use the `sysctl` command to change this setting at runtime and edit `/etc/sysctl.conf` for this setting to be saved after a reboot. You may continue with installation, but you may run into issues with Cloudera Manager reporting that your hosts are unhealthy because they are swapping. The following hosts are affected: ›
- ✓ No performance concerns with Transparent Huge Pages settings.
- ✓ CDH 5 Hue Python version dependency is satisfied.
- ✓ 5 hosts are running CDH 4 and 0 hosts are running CDH5.
- ✓ All checked hosts in each cluster are running the same version of components.
- ✓ All managed hosts have consistent versions of Java.
- ✓ All checked Cloudera Management Daemons versions are consistent with the server.
- ✓ All checked Cloudera Management Agents versions are consistent with the server.

Version Summary

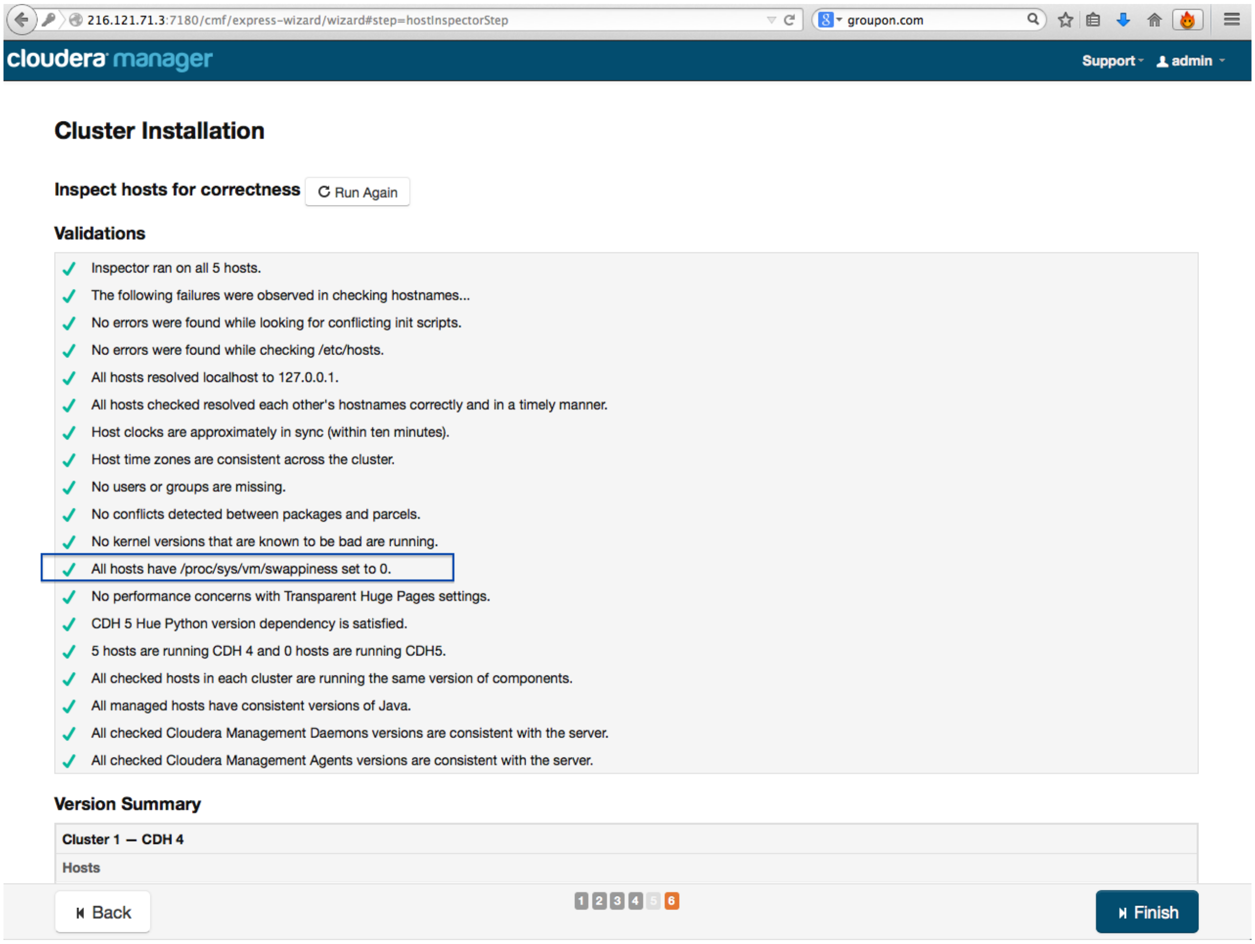
Cluster 1 — CDH 4

If you get a warning related to /proc/sys/vm/swappiness then set the value of swappiness accordingly using the below command

```
sysctl -w vm.swappiness=0.
```

change the value of swappiness on all the hosts.

After changing the swappiness value Click on Run Again and if you see no warning then you are good.


216.121.71.3:7180/cmf/express-wizard/wizard#step=hostInspectorStep
groupon.com
cloudera manager
Support
admin

Cluster Installation

Inspect hosts for correctness

Validations

- ✓ Inspector ran on all 5 hosts.
- ✓ The following failures were observed in checking hostnames...
- ✓ No errors were found while looking for conflicting init scripts.
- ✓ No errors were found while checking /etc/hosts.
- ✓ All hosts resolved localhost to 127.0.0.1.
- ✓ All hosts checked resolved each other's hostnames correctly and in a timely manner.
- ✓ Host clocks are approximately in sync (within ten minutes).
- ✓ Host time zones are consistent across the cluster.
- ✓ No users or groups are missing.
- ✓ No conflicts detected between packages and parcels.
- ✓ No kernel versions that are known to be bad are running.
- ✓ All hosts have /proc/sys/vm/swappiness set to 0.
- ✓ No performance concerns with Transparent Huge Pages settings.
- ✓ CDH 5 Hue Python version dependency is satisfied.
- ✓ 5 hosts are running CDH 4 and 0 hosts are running CDH5.
- ✓ All checked hosts in each cluster are running the same version of components.
- ✓ All managed hosts have consistent versions of Java.
- ✓ All checked Cloudera Management Daemons versions are consistent with the server.
- ✓ All checked Cloudera Management Agents versions are consistent with the server.

Version Summary

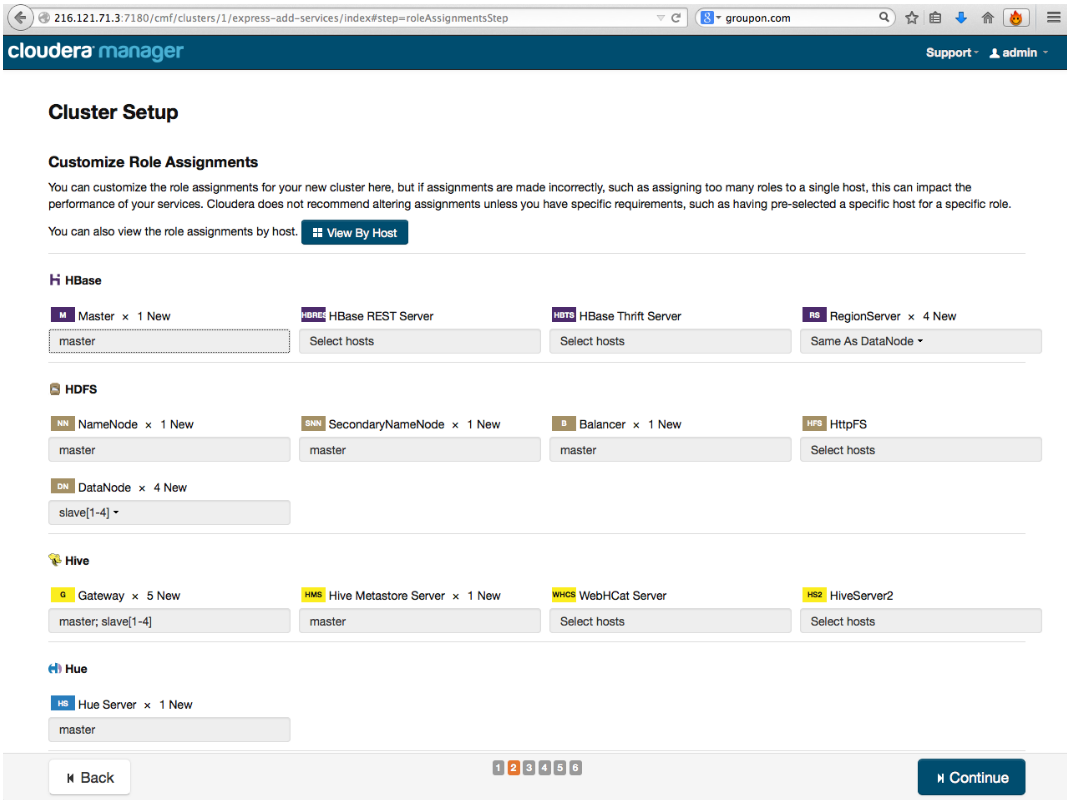
Cluster 1 — CDH 4

Hosts

Step7: Choose the CDH4 services that you want to install. I selected All Services. On CDH5 if you want to select mapreduce instead of mapreduce 2 then select the Custom Services Option and then from the Custom Services select mapreduce.

Press Continue.

Step8: Inspect the Cluster Setup.



216.121.71.3:7180/cmf/clusters/1/express-add-services/index#step=roleAssignmentsStep
groupon.com
cloudera manager
Support
admin

Cluster Setup

Customize Role Assignments

You can customize the role assignments for your new cluster here, but if assignments are made incorrectly, such as assigning too many roles to a single host, this can impact the performance of your services. Cloudera does not recommend altering assignments unless you have specific requirements, such as having pre-selected a specific host for a specific role.

You can also view the role assignments by host. View By Host

HBase

M Master × 1 New: master

HBREST HBase REST Server: Select hosts

HBTS HBase Thrift Server: Select hosts

RS RegionServer × 4 New: Same As DataNode

HDFS

NN NameNode × 1 New: master

SNN SecondaryNameNode × 1 New: master

B Balancer × 1 New: master

HFS HttpFS: Select hosts

DN DataNode × 4 New: slave[1-4]

Hive

G Gateway × 5 New: master; slave[1-4]

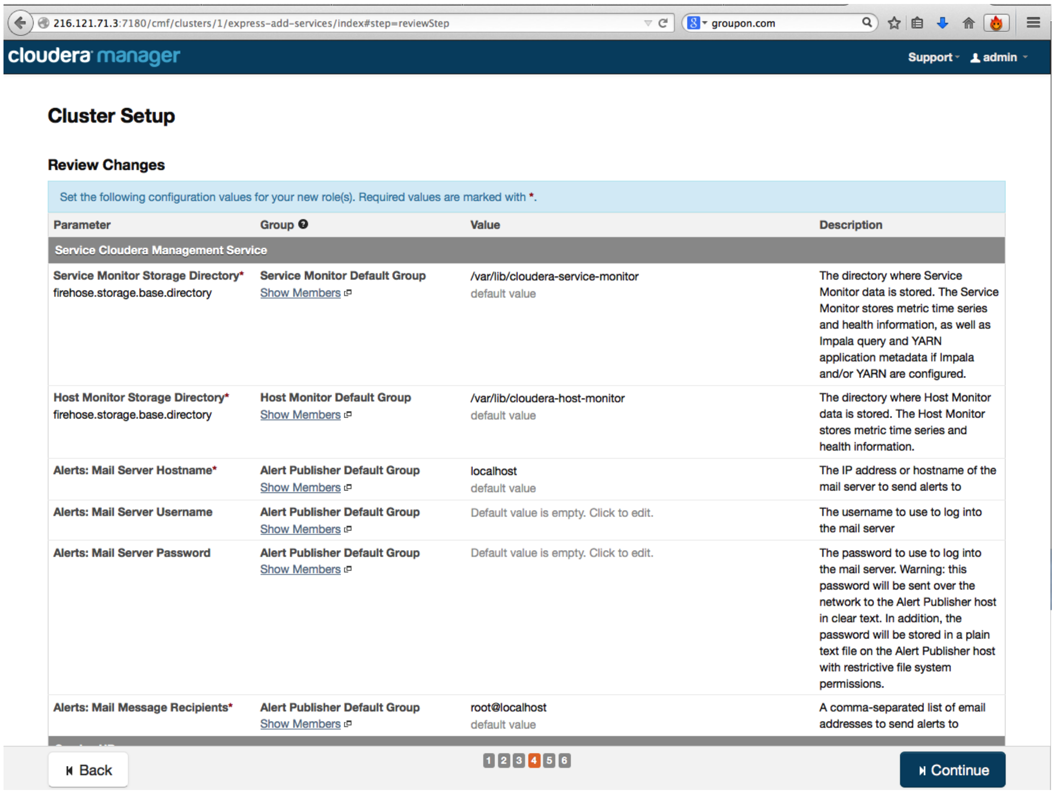
HMS Hive Metastore Server × 1 New: master

WHCS WebHCat Server: Select hosts

HS2 HiveServer2: Select hosts

Hue

HS Hue Server × 1 New: master



216.121.71.3:7180/cmf/clusters/1/express-add-services/index#step=reviewStep
groupon.com
cloudera manager
Support
admin

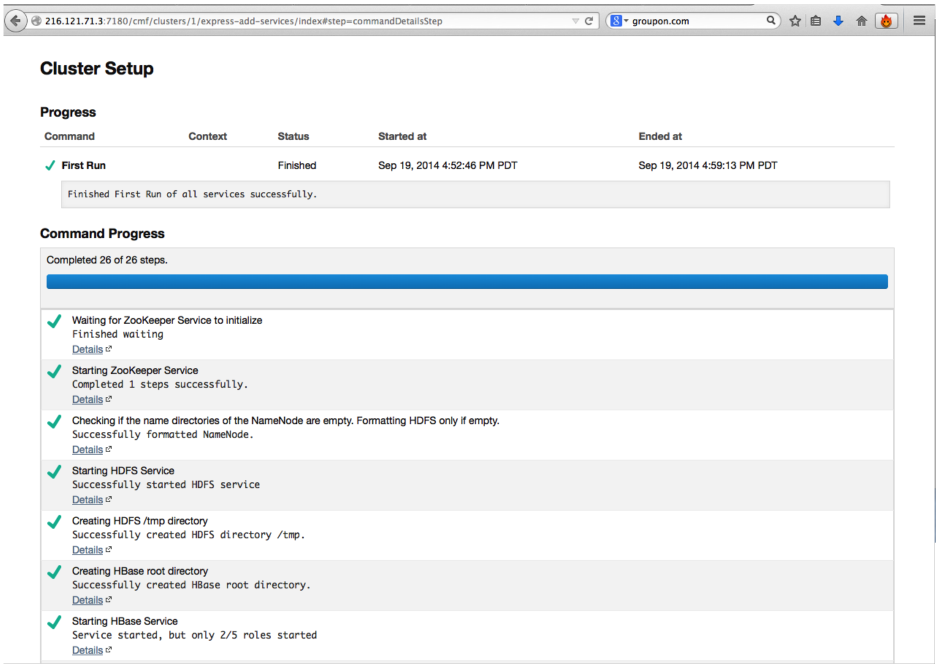
Cluster Setup

Review Changes

Set the following configuration values for your new role(s). Required values are marked with *.

Parameter	Group	Value	Description
Service Cloudera Management Service			
Service Monitor Storage Directory* firehose.storage.base.directory	**Service Monitor Default Group** Show Members	/var/lib/cloudera-service-monitor default value	The directory where Service Monitor data is stored. The Service Monitor stores metric time series and health information, as well as Impala query and YARN application metadata if Impala and/or YARN are configured.
Host Monitor Storage Directory* firehose.storage.base.directory	**Host Monitor Default Group** Show Members	/var/lib/cloudera-host-monitor default value	The directory where Host Monitor data is stored. The Host Monitor stores metric time series and health information.
Alerts: Mail Server Hostname*	**Alert Publisher Default Group** Show Members	localhost default value	The IP address or hostname of the mail server to send alerts to
Alerts: Mail Server Username	**Alert Publisher Default Group** Show Members	Default value is empty. Click to edit.	The username to use to log into the mail server
Alerts: Mail Server Password	**Alert Publisher Default Group** Show Members	Default value is empty. Click to edit.	The password to use to log into the mail server. Warning: this password will be sent over the network to the Alert Publisher host in clear text. In addition, the password will be stored in a plain text file on the Alert Publisher host with restrictive file system permissions.
Alerts: Mail Message Recipients*	**Alert Publisher Default Group** Show Members	root@localhost default value	A comma-separated list of email addresses to send alerts to

Step 9: Verify the first run of all the services. Once all the services are running Then Click on Continue.



216.121.71.3:7180/cmf/clusters/1/express-add-services/index#step=commandDetailsStep
groupon.com

Cluster Setup

Progress

Command	Context	Status	Started at	Ended at
✓ **First Run**		Finished	Sep 19, 2014 4:52:46 PM PDT	Sep 19, 2014 4:59:13 PM PDT

```
Finished First Run of all services successfully.
```

Command Progress

Completed 26 of 26 steps.

✓ Waiting for ZooKeeper Service to initialize
```
Finished waiting
```
Details

✓ Starting ZooKeeper Service
```
Completed 1 steps successfully.
```
Details

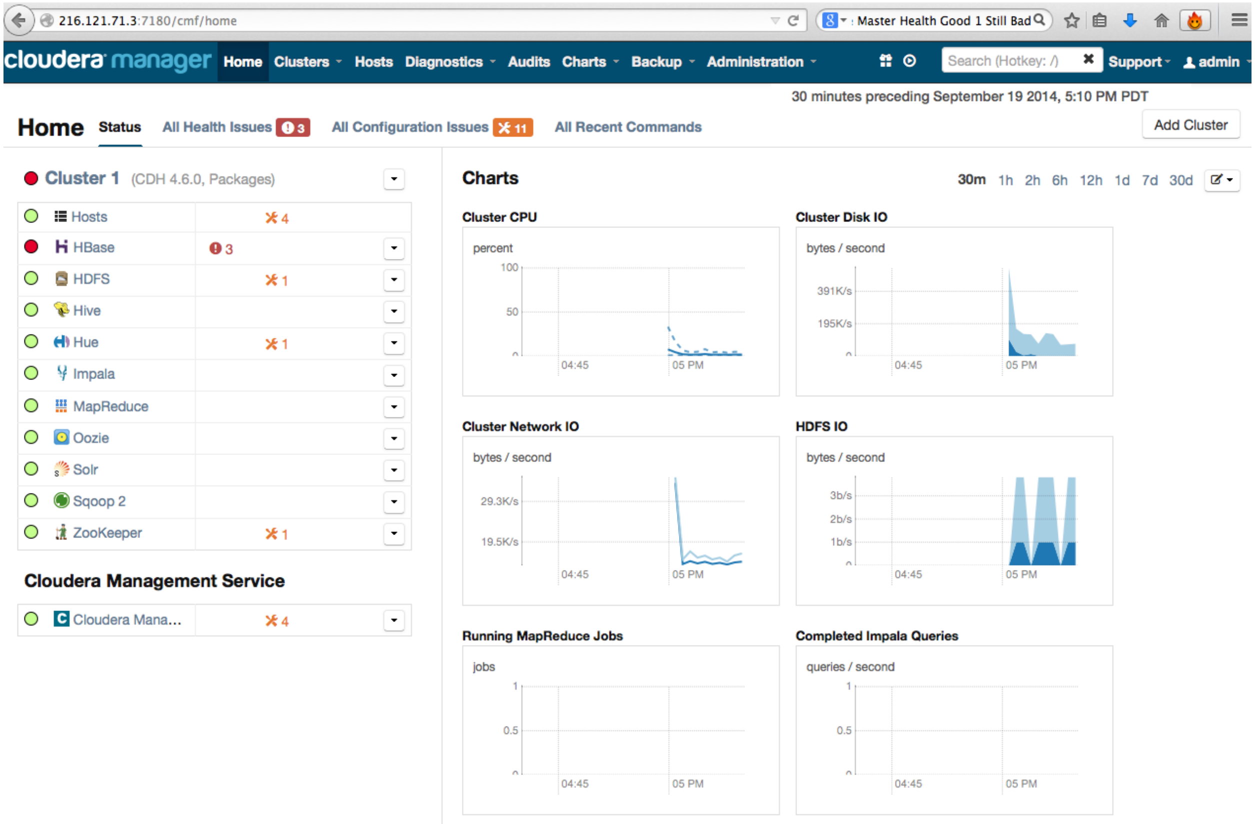
✓ Checking if the name directories of the NameNode are empty. Formatting HDFS only if empty.
```
Successfully formatted NameNode.
```
Details

✓ Starting HDFS Service
```
Successfully started HDFS service
```
Details

✓ Creating HDFS /tmp directory
```
Successfully created HDFS directory /tmp.
```
Details

✓ Creating HBase root directory
```
Successfully created HBase root directory.
```
Details

✓ Starting HBase Service
```
Service started, but only 2/5 roles started
```
Details



Cluster 1 (CDH 4.6.0, Packages)

○	Hosts	4
●	HBase	3
○	HDFS	1
○	Hive	
○	Hue	1
○	Impala	
○	MapReduce	
○	Oozie	
○	Solr	
○	Sqoop 2	
○	ZooKeeper	1

Cloudera Management Service

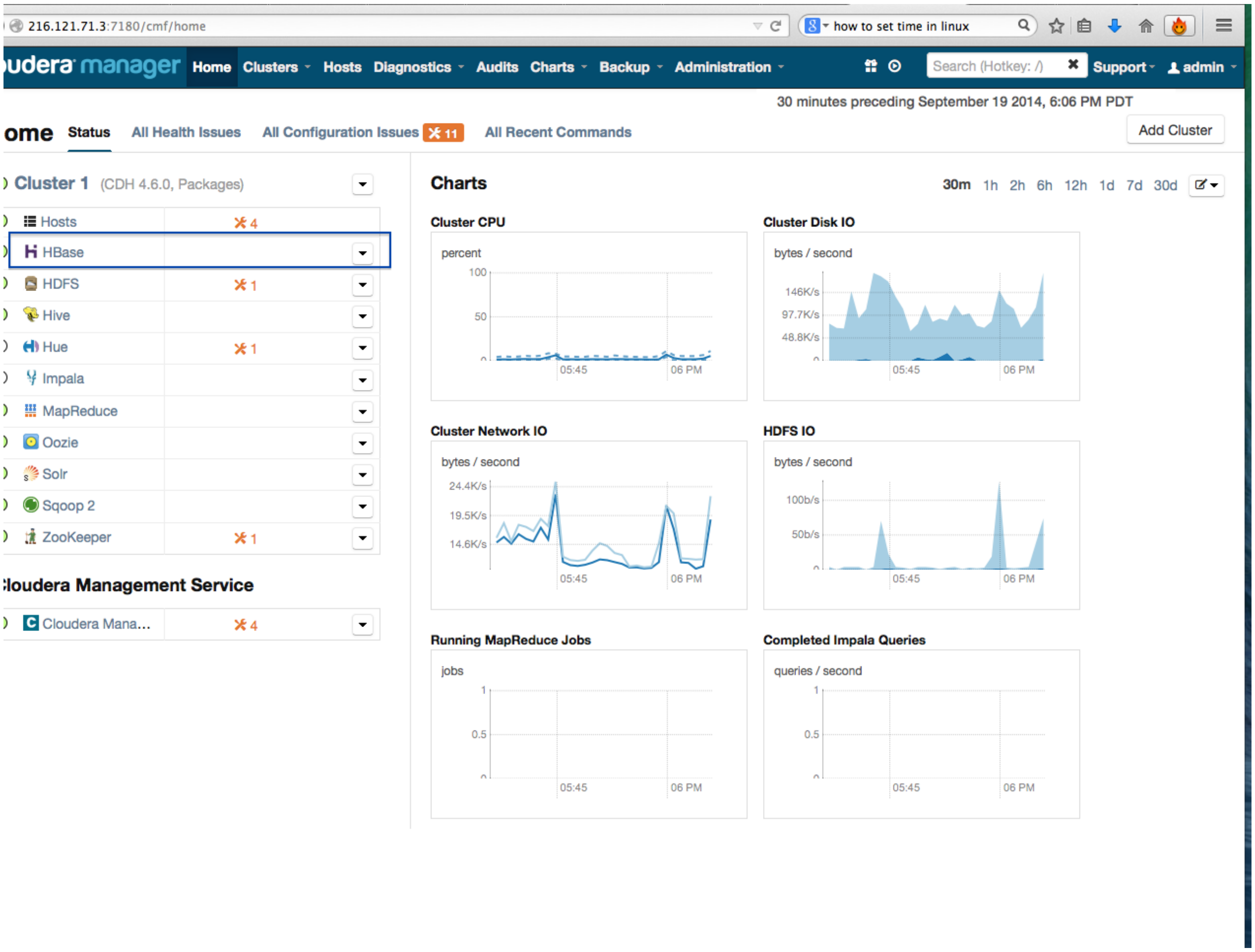
○	Cloudera Mana...	4

Step 10: Resolving HBase critical health issue
You might see some critical health issue for HBase.
The reason is that the slave servers are out of sync with the master servers.
We need to set the slave servers date according to master server and try to keep the difference less than 10-15 seconds.
set date using below command for all the slaves

```
date --set="Fri Sep 19 18:03:55"
```

Now the HBse issue is gone.

Output Screen shot



Steps For Running the Word Count Program

Step 1: Create the input directory /user/cloudera/wordcount/input in HDFS by running following commands

```
sudo su hdfs
hadoop fs -mkdir /user/cloudera
hadoop fs -chown cloudera /user/cloudera
exit
hadoop fs -mkdir /user/cloudera/wordcount /user/cloudera/wordcount/input
```

The above command will create a directory named input directory with in hdfs under /user/cloudera/wordcount.

Step 2: Now create some text files as sample files.

```
echo "Hello Hadoop Goodbye Hadoop" > file4
echo "Hello Hadoop Hello Hadoop" > file5
```

Step 3: Now Compile WordCount.java

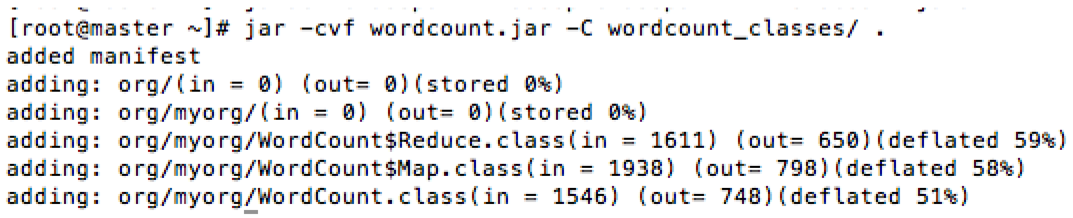
```
mkdir wordcount_classes
javac -classpath `hadoop classpath`:. WordCount.java
```

Here hadoop classpath is the classpath for the package installation which is equivalent to /usr/lib/hadoop/*:/usr/lib/hadoop/client-0.20/*

Step 4: create the JAR of WordCount .java

```
jar -cvf wordcount.jar -C wordcount_classes/ .
```

output screen shot

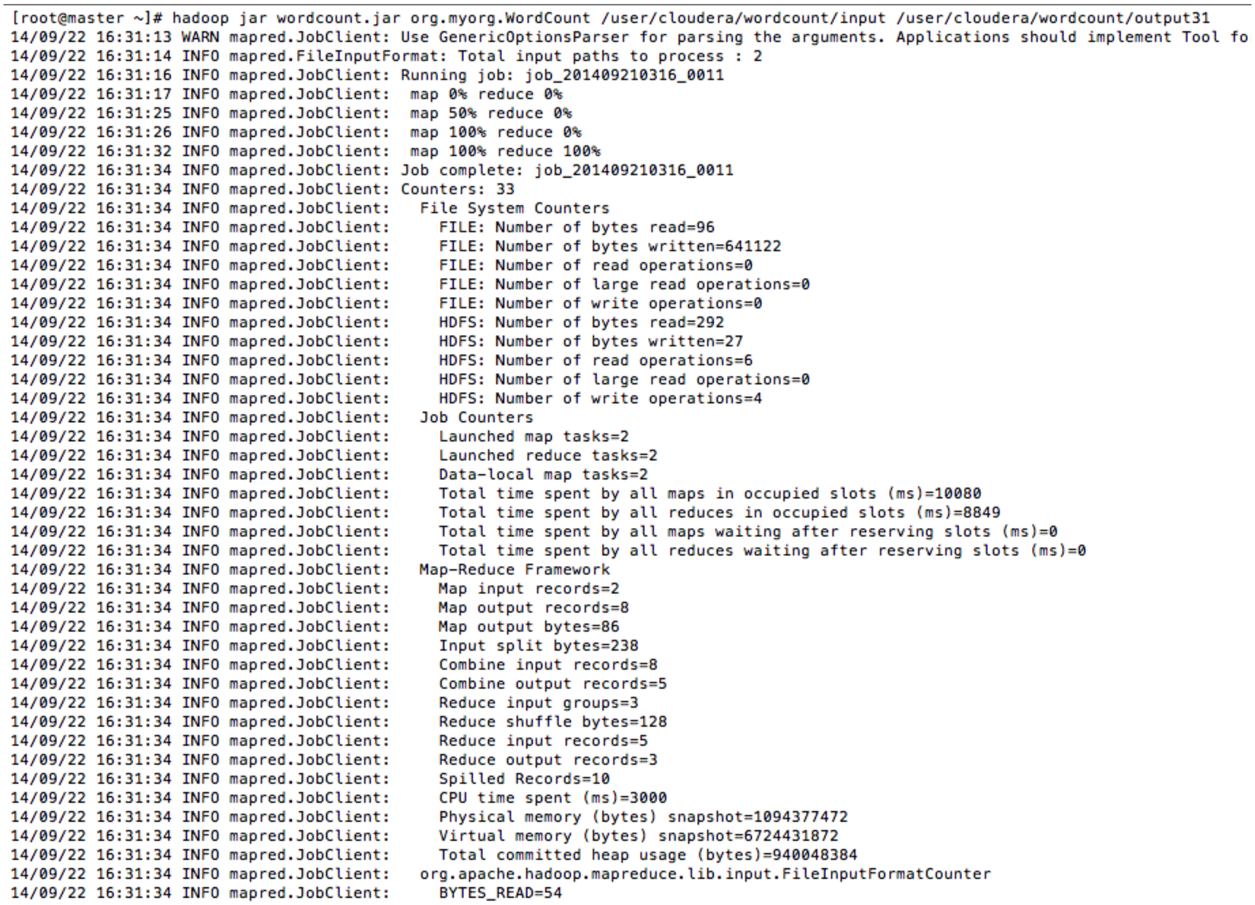


```
[root@master ~]# jar -cvf wordcount.jar -C wordcount_classes/ .
added manifest
adding: org/(in = 0) (out= 0)(stored 0%)
adding: org/myorg/(in = 0) (out= 0)(stored 0%)
adding: org/myorg/WordCount$Reduce.class(in = 1611) (out= 650)(deflated 59%)
adding: org/myorg/WordCount$Map.class(in = 1938) (out= 798)(deflated 58%)
adding: org/myorg/WordCount.class(in = 1546) (out= 748)(deflated 51%)
```

Step 5: Run the WordCount application.

```
hadoop jar wor dcount.jar org.myorg.WordCount /user/cloudera/wordcount/input
/user/cloudera/wordcount/output31
```

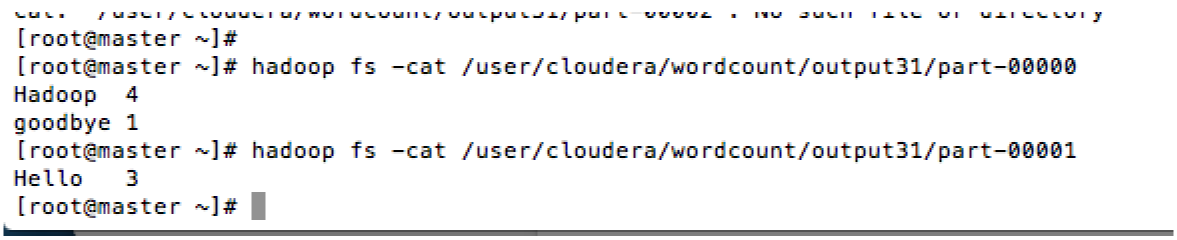
output screen shot



```
[root@master ~]# hadoop jar wordcount.jar org.myorg.WordCount /user/cloudera/wordcount/input /user/cloudera/wordcount/output31
14/09/22 16:31:13 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement Tool fo
14/09/22 16:31:14 INFO mapred.FileInputFormat: Total input paths to process : 2
14/09/22 16:31:16 INFO mapred.JobClient: Running job: job_201409210316_0011
14/09/22 16:31:17 INFO mapred.JobClient:  map 0% reduce 0%
14/09/22 16:31:25 INFO mapred.JobClient:  map 50% reduce 0%
14/09/22 16:31:26 INFO mapred.JobClient:  map 100% reduce 0%
14/09/22 16:31:32 INFO mapred.JobClient:  map 100% reduce 100%
14/09/22 16:31:34 INFO mapred.JobClient: Job complete: job_201409210316_0011
14/09/22 16:31:34 INFO mapred.JobClient: Counters: 33
14/09/22 16:31:34 INFO mapred.JobClient:   File System Counters
14/09/22 16:31:34 INFO mapred.JobClient:     FILE: Number of bytes read=96
14/09/22 16:31:34 INFO mapred.JobClient:     FILE: Number of bytes written=641122
14/09/22 16:31:34 INFO mapred.JobClient:     FILE: Number of read operations=0
14/09/22 16:31:34 INFO mapred.JobClient:     FILE: Number of large read operations=0
14/09/22 16:31:34 INFO mapred.JobClient:     FILE: Number of write operations=0
14/09/22 16:31:34 INFO mapred.JobClient:     HDFS: Number of bytes read=292
14/09/22 16:31:34 INFO mapred.JobClient:     HDFS: Number of bytes written=27
14/09/22 16:31:34 INFO mapred.JobClient:     HDFS: Number of read operations=6
14/09/22 16:31:34 INFO mapred.JobClient:     HDFS: Number of large read operations=0
14/09/22 16:31:34 INFO mapred.JobClient:     HDFS: Number of write operations=4
14/09/22 16:31:34 INFO mapred.JobClient:   Job Counters
14/09/22 16:31:34 INFO mapred.JobClient:     Launched map tasks=2
14/09/22 16:31:34 INFO mapred.JobClient:     Launched reduce tasks=2
14/09/22 16:31:34 INFO mapred.JobClient:     Data-local map tasks=2
14/09/22 16:31:34 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=10080
14/09/22 16:31:34 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=8849
14/09/22 16:31:34 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
14/09/22 16:31:34 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
14/09/22 16:31:34 INFO mapred.JobClient:   Map-Reduce Framework
14/09/22 16:31:34 INFO mapred.JobClient:     Map input records=2
14/09/22 16:31:34 INFO mapred.JobClient:     Map output records=8
14/09/22 16:31:34 INFO mapred.JobClient:     Map output bytes=86
14/09/22 16:31:34 INFO mapred.JobClient:     Input split bytes=238
14/09/22 16:31:34 INFO mapred.JobClient:     Combine input records=8
14/09/22 16:31:34 INFO mapred.JobClient:     Combine output records=5
14/09/22 16:31:34 INFO mapred.JobClient:     Reduce input groups=3
14/09/22 16:31:34 INFO mapred.JobClient:     Reduce shuffle bytes=128
14/09/22 16:31:34 INFO mapred.JobClient:     Reduce input records=5
14/09/22 16:31:34 INFO mapred.JobClient:     Reduce output records=3
14/09/22 16:31:34 INFO mapred.JobClient:     Spilled Records=10
14/09/22 16:31:34 INFO mapred.JobClient:     CPU time spent (ms)=3000
14/09/22 16:31:34 INFO mapred.JobClient:     Physical memory (bytes) snapshot=1094377472
14/09/22 16:31:34 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=6724431872
14/09/22 16:31:34 INFO mapred.JobClient:     Total committed heap usage (bytes)=940048384
14/09/22 16:31:34 INFO mapred.JobClient:   org.apache.hadoop.mapreduce.lib.input.FileInputFormatCounter
14/09/22 16:31:34 INFO mapred.JobClient:     BYTES_READ=54
```

Step 6: Now check the output in the output directory using the below command

hadoop fs -cat /user/cloudera/wordcount/output31/part-00000



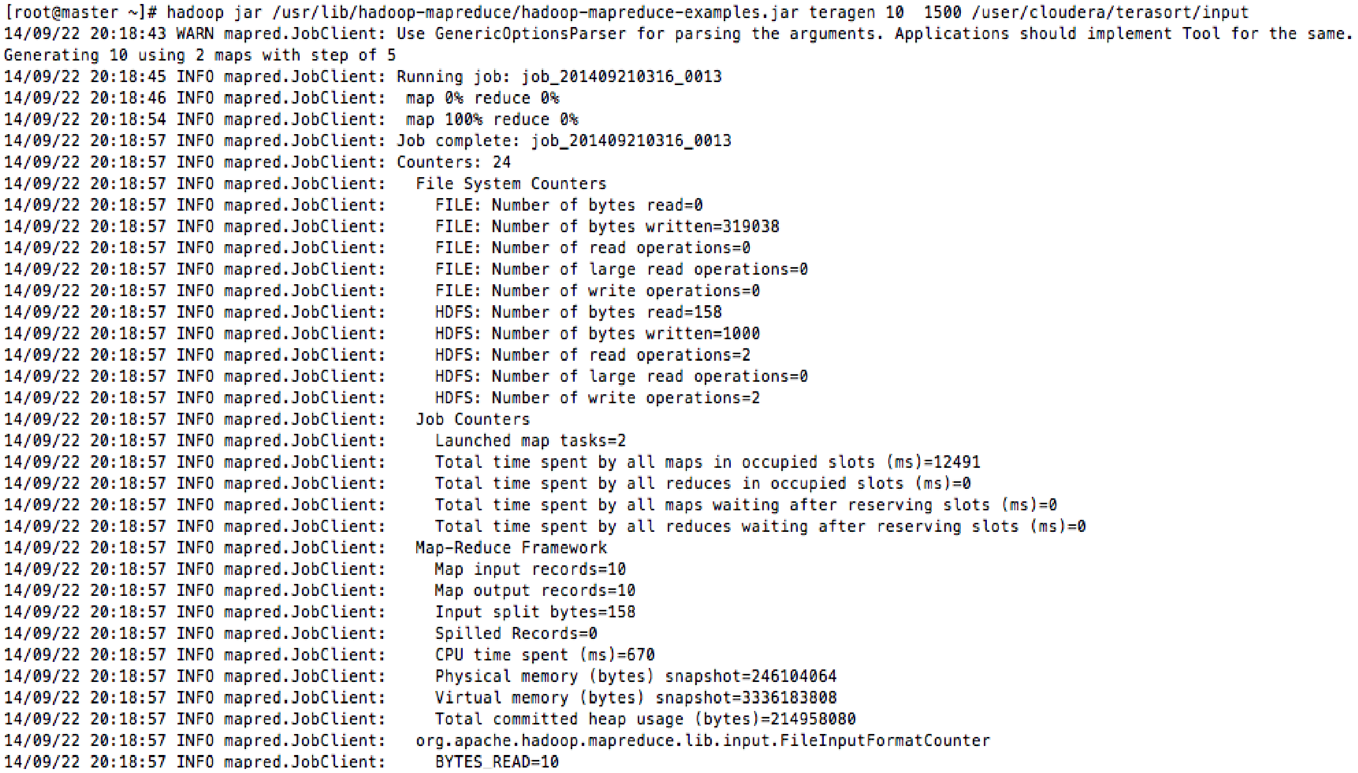
```
cat: /user/cloudera/wordcount/output31/part-00002 : No such file or directory
[root@master ~]#
[root@master ~]# hadoop fs -cat /user/cloudera/wordcount/output31/part-00000
Hadoop  4
goodbye 1
[root@master ~]# hadoop fs -cat /user/cloudera/wordcount/output31/part-00001
Hello   3
[root@master ~]#
```

Steps for running Terasort Program

Step 1: Generate the input using teragen

/usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar teragen 10 1500 /user/cloudera/terasort1/input

Output Screenshot

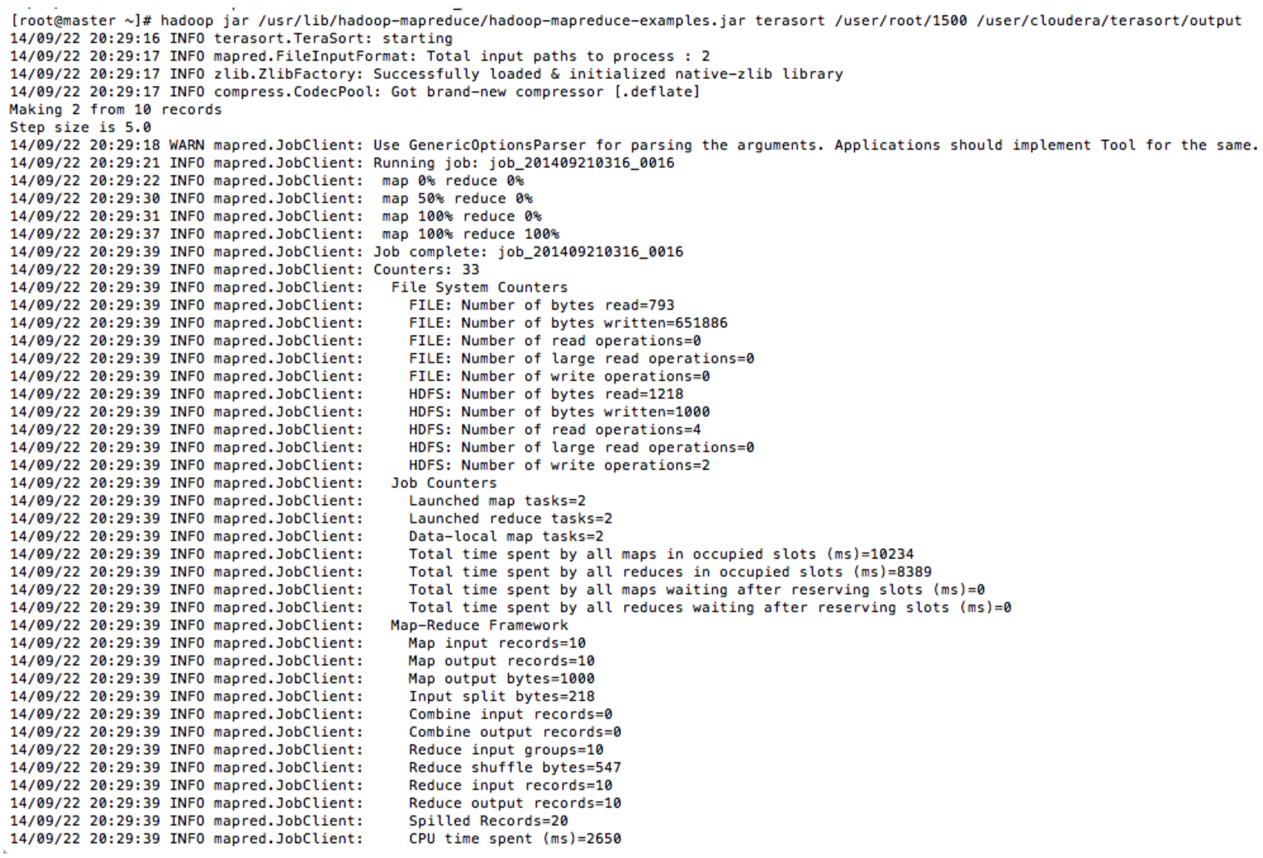


```
[root@master ~]# hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar teragen 10  1500 /user/cloudera/terasort/input
14/09/22 20:18:43 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement Tool for the same.
Generating 10 using 2 maps with step of 5
14/09/22 20:18:45 INFO mapred.JobClient: Running job: job_201409210316_0013
14/09/22 20:18:46 INFO mapred.JobClient:  map 0% reduce 0%
14/09/22 20:18:54 INFO mapred.JobClient:  map 100% reduce 0%
14/09/22 20:18:57 INFO mapred.JobClient: Job complete: job_201409210316_0013
14/09/22 20:18:57 INFO mapred.JobClient: Counters: 24
14/09/22 20:18:57 INFO mapred.JobClient:   File System Counters
14/09/22 20:18:57 INFO mapred.JobClient:     FILE: Number of bytes read=0
14/09/22 20:18:57 INFO mapred.JobClient:     FILE: Number of bytes written=319038
14/09/22 20:18:57 INFO mapred.JobClient:     FILE: Number of read operations=0
14/09/22 20:18:57 INFO mapred.JobClient:     FILE: Number of large read operations=0
14/09/22 20:18:57 INFO mapred.JobClient:     FILE: Number of write operations=0
14/09/22 20:18:57 INFO mapred.JobClient:     HDFS: Number of bytes read=158
14/09/22 20:18:57 INFO mapred.JobClient:     HDFS: Number of bytes written=1000
14/09/22 20:18:57 INFO mapred.JobClient:     HDFS: Number of read operations=2
14/09/22 20:18:57 INFO mapred.JobClient:     HDFS: Number of large read operations=0
14/09/22 20:18:57 INFO mapred.JobClient:     HDFS: Number of write operations=2
14/09/22 20:18:57 INFO mapred.JobClient:   Job Counters
14/09/22 20:18:57 INFO mapred.JobClient:     Launched map tasks=2
14/09/22 20:18:57 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=12491
14/09/22 20:18:57 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=0
14/09/22 20:18:57 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
14/09/22 20:18:57 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
14/09/22 20:18:57 INFO mapred.JobClient:   Map-Reduce Framework
14/09/22 20:18:57 INFO mapred.JobClient:     Map input records=10
14/09/22 20:18:57 INFO mapred.JobClient:     Map output records=10
14/09/22 20:18:57 INFO mapred.JobClient:     Input split bytes=158
14/09/22 20:18:57 INFO mapred.JobClient:     Spilled Records=0
14/09/22 20:18:57 INFO mapred.JobClient:     CPU time spent (ms)=670
14/09/22 20:18:57 INFO mapred.JobClient:     Physical memory (bytes) snapshot=246104064
14/09/22 20:18:57 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=3336183808
14/09/22 20:18:57 INFO mapred.JobClient:     Total committed heap usage (bytes)=214958080
14/09/22 20:18:57 INFO mapred.JobClient:   org.apache.hadoop.mapreduce.lib.input.FileInputFormatCounter
14/09/22 20:18:57 INFO mapred.JobClient:     BYTES_READ=10
```

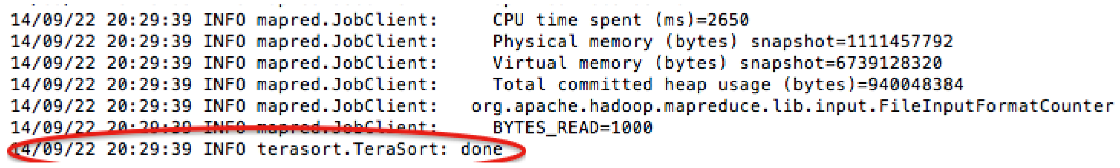
Step 2: Next run Terasort to conduct the sorting

hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar terasort /user/root/1500 /user/cloudera/terasort/output

Output Screen Shot



```
[root@master ~]# hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar terasort /user/root/1500 /user/cloudera/terasort/output
14/09/22 20:29:16 INFO terasort.TeraSort: starting
14/09/22 20:29:17 INFO mapred.FileInputFormat: Total input paths to process : 2
14/09/22 20:29:17 INFO zlib.ZlibFactory: Successfully loaded & initialized native-zlib library
14/09/22 20:29:17 INFO compress.CodecPool: Got brand-new compressor [.deflate]
Making 2 from 10 records
Step size is 5.0
14/09/22 20:29:18 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement Tool for the same.
14/09/22 20:29:21 INFO mapred.JobClient: Running job: job_201409210316_0016
14/09/22 20:29:22 INFO mapred.JobClient:  map 0% reduce 0%
14/09/22 20:29:30 INFO mapred.JobClient:  map 50% reduce 0%
14/09/22 20:29:31 INFO mapred.JobClient:  map 100% reduce 0%
14/09/22 20:29:37 INFO mapred.JobClient:  map 100% reduce 100%
14/09/22 20:29:39 INFO mapred.JobClient: Job complete: job_201409210316_0016
14/09/22 20:29:39 INFO mapred.JobClient: Counters: 33
14/09/22 20:29:39 INFO mapred.JobClient:   File System Counters
14/09/22 20:29:39 INFO mapred.JobClient:     FILE: Number of bytes read=793
14/09/22 20:29:39 INFO mapred.JobClient:     FILE: Number of bytes written=651886
14/09/22 20:29:39 INFO mapred.JobClient:     FILE: Number of read operations=0
14/09/22 20:29:39 INFO mapred.JobClient:     FILE: Number of large read operations=0
14/09/22 20:29:39 INFO mapred.JobClient:     FILE: Number of write operations=0
14/09/22 20:29:39 INFO mapred.JobClient:     HDFS: Number of bytes read=1218
14/09/22 20:29:39 INFO mapred.JobClient:     HDFS: Number of bytes written=1000
14/09/22 20:29:39 INFO mapred.JobClient:     HDFS: Number of read operations=4
14/09/22 20:29:39 INFO mapred.JobClient:     HDFS: Number of large read operations=0
14/09/22 20:29:39 INFO mapred.JobClient:     HDFS: Number of write operations=2
14/09/22 20:29:39 INFO mapred.JobClient:   Job Counters
14/09/22 20:29:39 INFO mapred.JobClient:     Launched map tasks=2
14/09/22 20:29:39 INFO mapred.JobClient:     Launched reduce tasks=2
14/09/22 20:29:39 INFO mapred.JobClient:     Data-local map tasks=2
14/09/22 20:29:39 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=10234
14/09/22 20:29:39 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=8389
14/09/22 20:29:39 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
14/09/22 20:29:39 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
14/09/22 20:29:39 INFO mapred.JobClient:   Map-Reduce Framework
14/09/22 20:29:39 INFO mapred.JobClient:     Map input records=10
14/09/22 20:29:39 INFO mapred.JobClient:     Map output records=10
14/09/22 20:29:39 INFO mapred.JobClient:     Map output bytes=1000
14/09/22 20:29:39 INFO mapred.JobClient:     Input split bytes=218
14/09/22 20:29:39 INFO mapred.JobClient:     Combine input records=0
14/09/22 20:29:39 INFO mapred.JobClient:     Combine output records=0
14/09/22 20:29:39 INFO mapred.JobClient:     Reduce input groups=10
14/09/22 20:29:39 INFO mapred.JobClient:     Reduce shuffle bytes=547
14/09/22 20:29:39 INFO mapred.JobClient:     Reduce input records=10
14/09/22 20:29:39 INFO mapred.JobClient:     Reduce output records=10
14/09/22 20:29:39 INFO mapred.JobClient:     Spilled Records=20
14/09/22 20:29:39 INFO mapred.JobClient:     CPU time spent (ms)=2650
```



```
14/09/22 20:29:39 INFO mapred.JobClient:     CPU time spent (ms)=2650
14/09/22 20:29:39 INFO mapred.JobClient:     Physical memory (bytes) snapshot=1111457792
14/09/22 20:29:39 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=6739128320
14/09/22 20:29:39 INFO mapred.JobClient:     Total committed heap usage (bytes)=940048384
14/09/22 20:29:39 INFO mapred.JobClient:   org.apache.hadoop.mapreduce.lib.input.FileInputFormatCounter
14/09/22 20:29:39 INFO mapred.JobClient:     BYTES_READ=1000
14/09/22 20:29:39 INFO terasort.TeraSort: done
```

Note : You might run into java versioning problem Steps for handling java versioning problem

Step 1: Make sure that your servers are using java 1.7.x
Step2: set JAVA_HOME and point it to java version 1.7.x on all your servers.

Step 3: If you are still getting versioning issue then change the JAVA_HOME values in following two files and point it to java 1.7.x.

Step 4: Run the above process for all the servers
Step 5: Restart the server