Cloud Computing Homework #4a

1.

• HDFS NameNode formatting

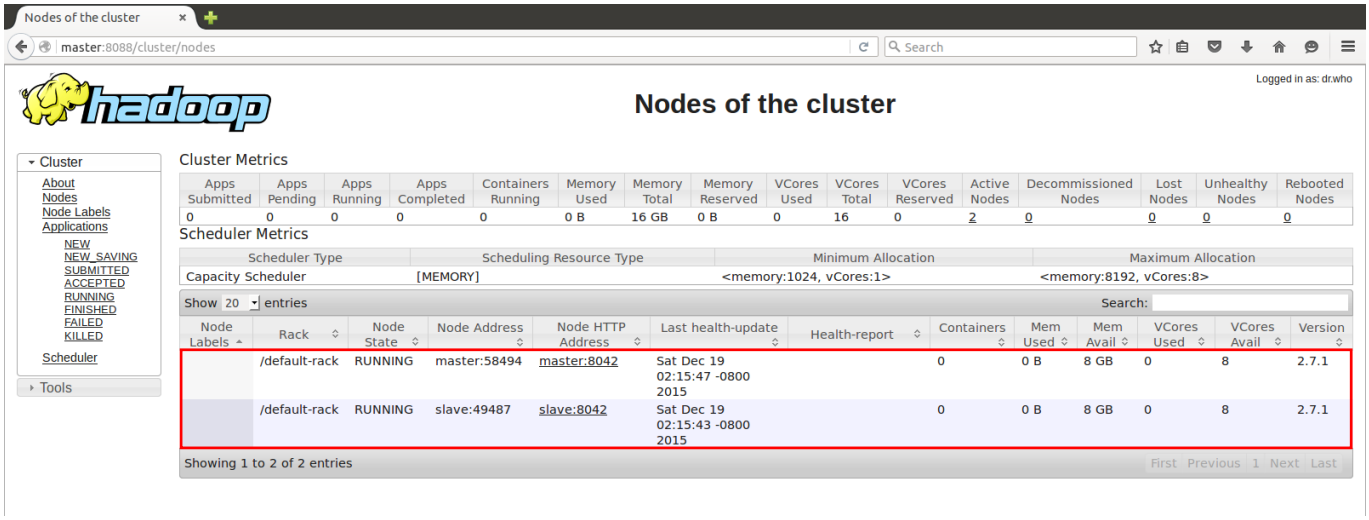
```
15/12/19 02:08:51 INFO util.ExitUtil: Exiting with status 0
15/12/19 02:08:51 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at master/172.16.114.131
************************************************************/
u0456024@master:~/hadoop/hadoop-2.7.1$
```

• jps

```
u0456024@master:~/hadoop/hadoop-2.7.1$ jps
2779 DataNode
3166 ResourceManager
3293 NodeManager
3584 Jps
2992 SecondaryNameNode
2655 NameNode
u0456024@master:~/hadoop/hadoop-2.7.1$
```

```
u0456024@slave:~$ jps
2673 NodeManager
2517 DataNode
2773 Jps
u0456024@slave:~$
```

• Hadoop administration website

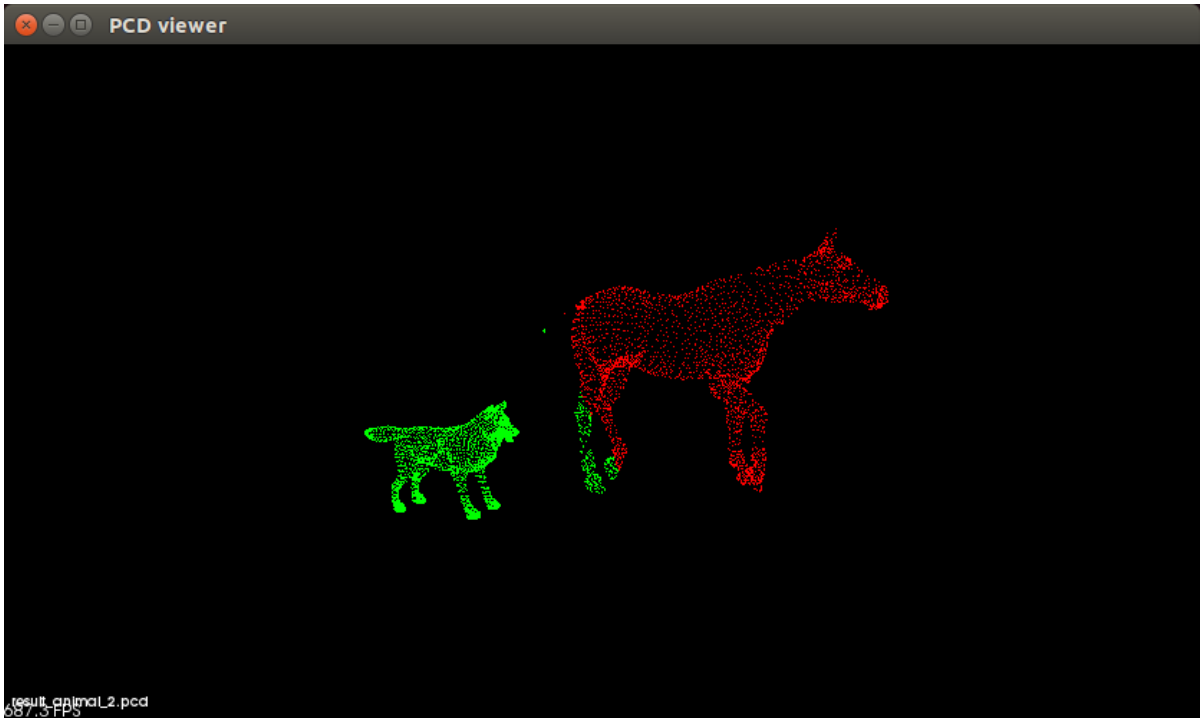


2.

• k-means for animal.pcd

```
Iteration 8: 2 converged.
    7.7430286 -10.121379 44.609634 65280
    99.9346 -198.43648 110.534935 16711680

Execution time: 14.502000 sec.
```



• k-means for object3_cut.pcd

```
Iteration 27: 3 converged.
    -0.11498297 0.07552916 -0.82701665 16711680
    -0.158693 -0.10661176 -0.6262176 65280
    0.20826593 -0.06735779 -0.6762534 16776960

Execution time: 99.859001 sec.
```

3.

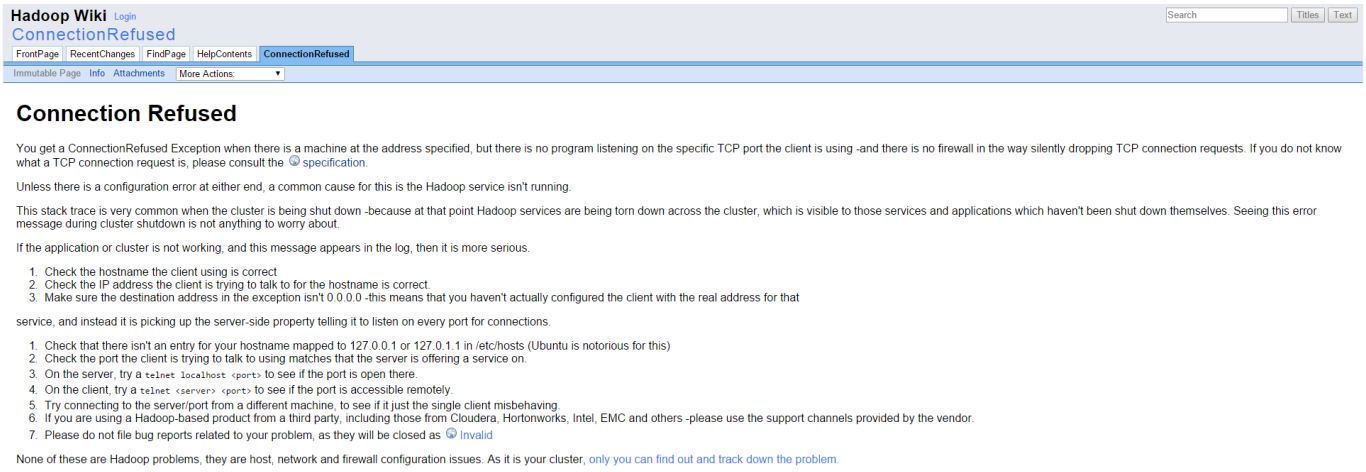
Hadoop is an open source software framework for distributed storage and distributed processing on clusters of commodity hardware. The base framework is composed of four modules: Hadoop Common – libraries and utilities needed by other modules, Hadoop Distributed File System (DFS) – a distributed file system, Hadoop Yet Another Resource Negotiator (YARN) – a resource management platform, Hadoop MapReduce – an implementation of the MapReduce programming model for big data processing.

When we start up Hadoop, we have to start up DFS and then YARN. The followings are Java virtual machine processes listed by jps command at each step. In DFS, NameNode manages DFS, SecondaryNameNode replicates metadata of NameNode periodically, and DataNode records data in DFS. In YARN, ResourceManager allocates resource and dispatches tasks, and NodeManager monitors resource on the node.

		hadoop/sbin/start-dfs.sh	hadoop/sbin/start-yarn.sh
master	2500 Jps	2779 DataNode 3101 Jps 2992 SecondaryNameNode 2655 NameNode	2779 DataNode 3166 ResourceManager 3293 NodeManager 3584 Jps 2992 SecondaryNameNode 2655 NameNode
slave	2433 Jps	2590 Jps 2517 DataNode	2673 NodeManager 2517 DataNode 2773 Jps

Hadoop is mostly written in Java, and it provides plugins for the Eclipse IDE to develop against the platform. In the MapReduce project, after setting up MapReduce location, we can manipulate the DFS and run applications on Hadoop. We can learn from the k-means source code that the intermediate data of process are access through reading from and writing to a set of temporary files stored in DFS.

When I first time set up MapReduce location, the DFS failed to establish a connection and got a ConnectionRefused Exception. I followed instructions mentioned in Hadoop Wiki to track down the problem and found that it is unable to telnet master since the IP address of virtual machine had changed and was different from the one recorded in /etc/hosts.



Hadoop Wiki Login

ConnectionRefused

Search Titles Text

FrontPage RecentChanges FindPage HelpContents ConnectionRefused

Immutable Page Info Attachments More Actions:

Connection Refused

You get a ConnectionRefused Exception when there is a machine at the address specified, but there is no program listening on the specific TCP port the client is using -and there is no firewall in the way silently dropping TCP connection requests. If you do not know what a TCP connection request is, please consult the specification.

Unless there is a configuration error at either end, a common cause for this is the Hadoop service isn't running.

This stack trace is very common when the cluster is being shut down -because at that point Hadoop services are being torn down across the cluster, which is visible to those services and applications which haven't been shut down themselves. Seeing this error message during cluster shutdown is not anything to worry about.

If the application or cluster is not working, and this message appears in the log, then it is more serious.

1. Check the hostname the client using is correct
2. Check the IP address the client is trying to talk to for the hostname is correct.
3. Make sure the destination address in the exception isn't 0.0.0.0 -this means that you haven't actually configured the client with the real address for that

service, and instead it is picking up the server-side property telling it to listen on every port for connections.

1. Check that there isn't an entry for your hostname mapped to 127.0.0.1 or 127.0.1.1 in /etc/hosts (Ubuntu is notorious for this)
2. Check the port the client is trying to talk to using matches that the server is offering a service on.
3. On the server, try a `telnet localhost <port>` to see if the port is open there.
4. On the client, try a `telnet <server> <port>` to see if the port is accessible remotely.
5. Try connecting to the server/port from a different machine, to see if it just the single client misbehaving.
6. If you are using a Hadoop-based product from a third party, including those from Cloudera, Hortonworks, Intel, EMC and others -please use the support channels provided by the vendor.
7. Please do not file bug reports related to your problem, as they will be closed as Invalid

None of these are Hadoop problems, they are host, network and firewall configuration issues. As it is your cluster, only you can find out and track down the problem.