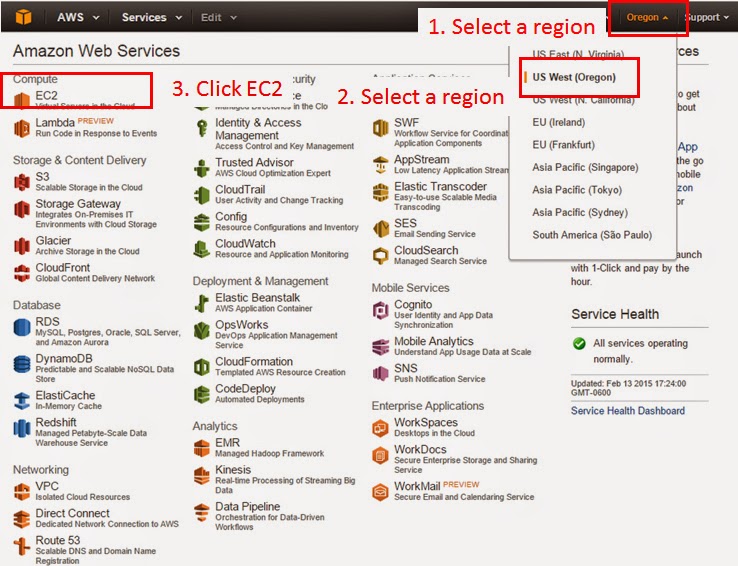
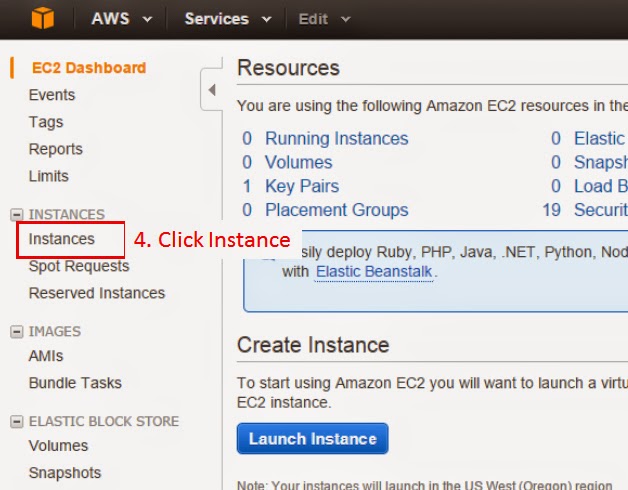
**Basic instructions:**

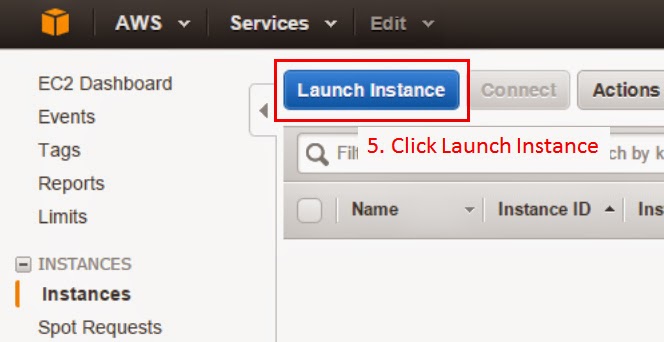
You want to follow basic instructions. This instructions describe how to launch Ubuntu instances on AWS, how to connect to the instances, how to install/configure Cassandra with three nodes, and how to install/benchmark them using YCSB.

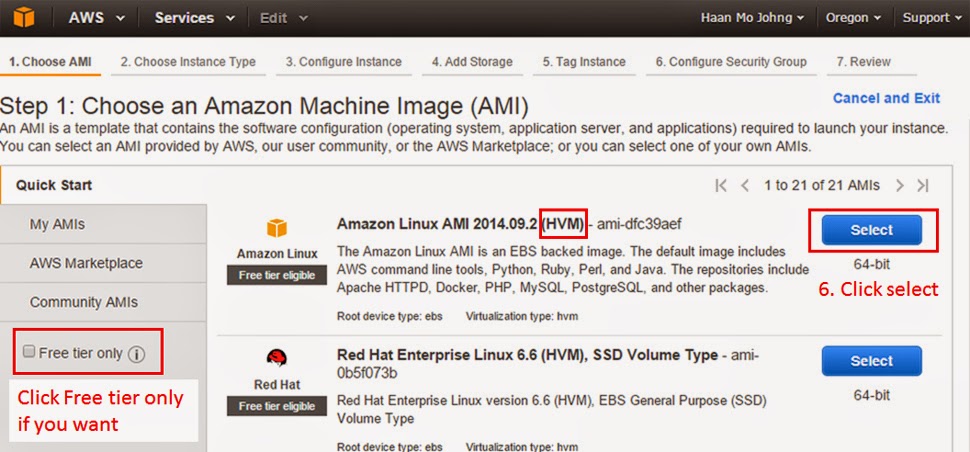
1. **Sign up for free trial**

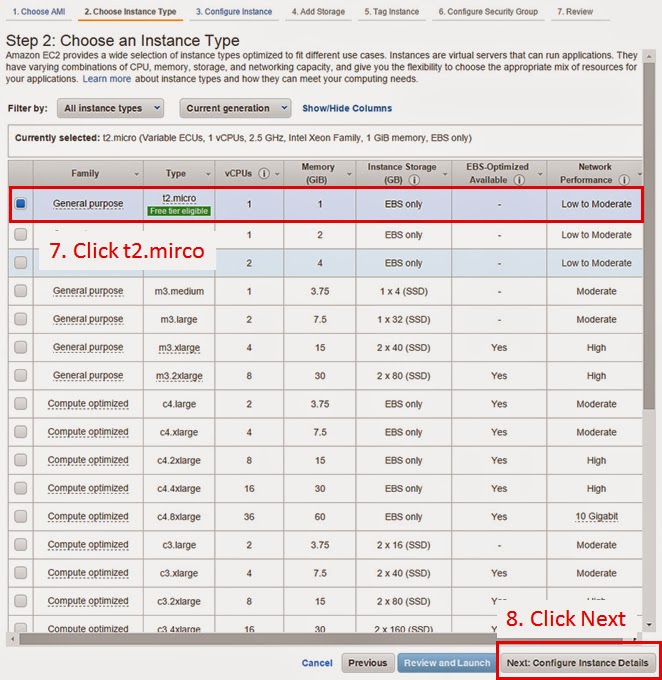
[**https://aws.amazon.com/free/**](https://aws.amazon.com/free/)

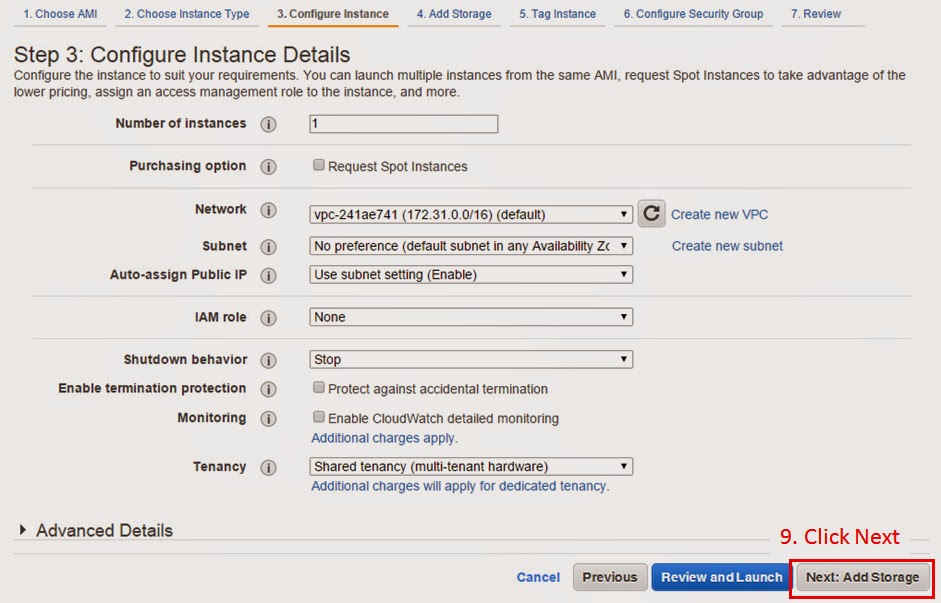
1. **Create an AWS EC2 Instance**
2. **[](http://2.bp.blogspot.com/-FH4-3SR-Rhs/VN6JG0d5jdI/AAAAAAAAADQ/JxmS2fK-4gQ/s1600/ec2_01_.jpg)**

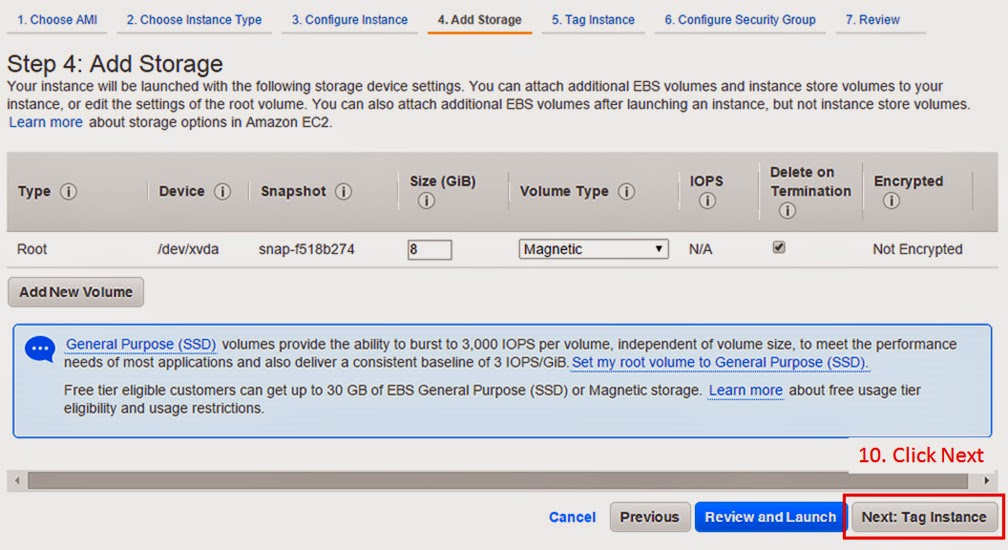
[](http://2.bp.blogspot.com/-jhb3faHmXsQ/VN6VtBuNsoI/AAAAAAAAADg/PFZTmv2gu1Y/s1600/ec2_02_.jpg)

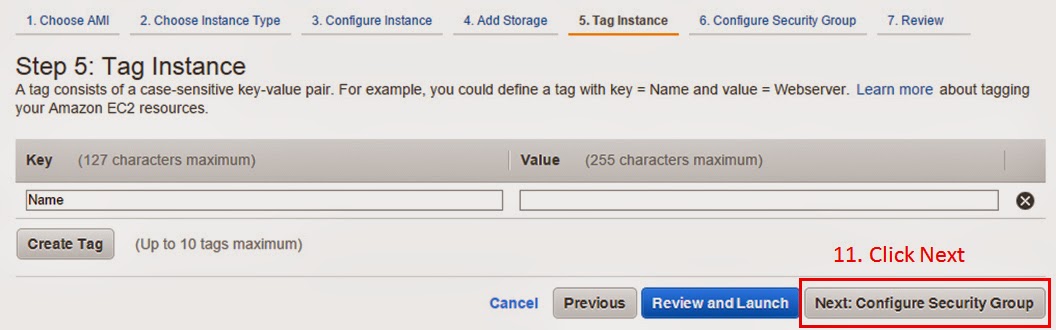
[](http://4.bp.blogspot.com/-DIjuyR_OfCo/VN6VtPXFymI/AAAAAAAAADo/EfOSihRQkFA/s1600/ec2_03_.jpg)

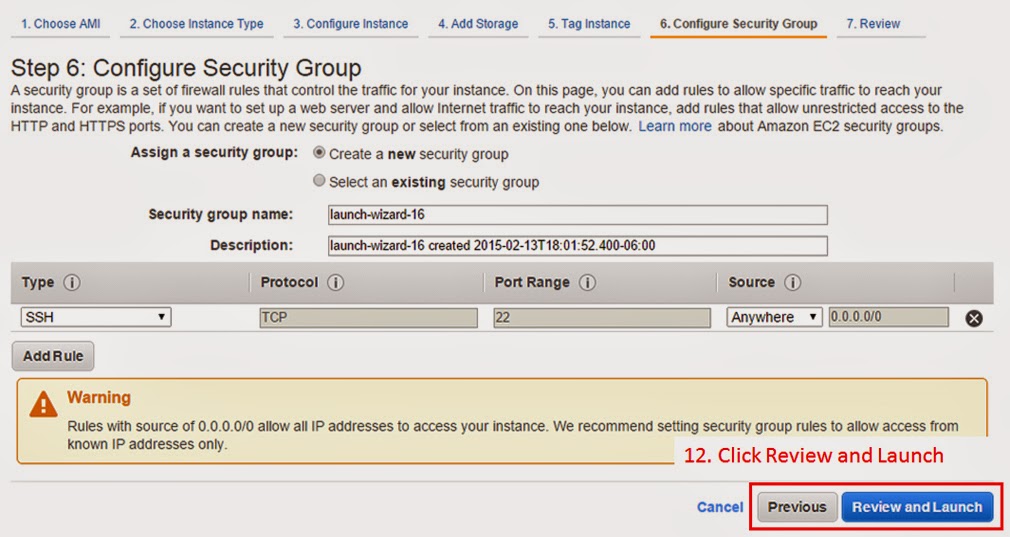
[](http://1.bp.blogspot.com/-mfsKRyOvD6g/VN6VtmvU5JI/AAAAAAAAADs/eZHmeZGalTQ/s1600/ec2_04_.jpg)

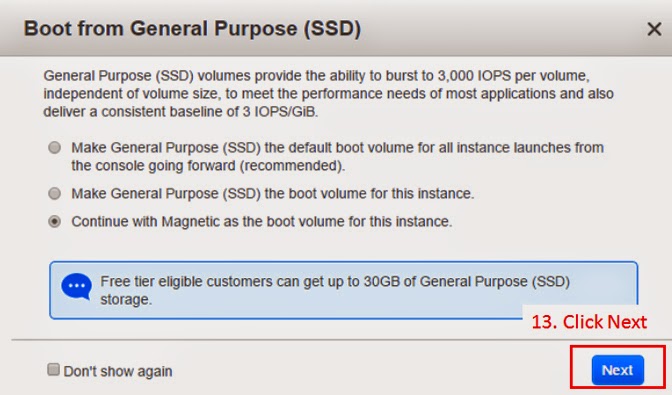
[](http://3.bp.blogspot.com/-4_kdzlPWrBA/VN6Vt6k-T5I/AAAAAAAAADw/nlyqv8-6OZ8/s1600/ec2_05_.jpg)

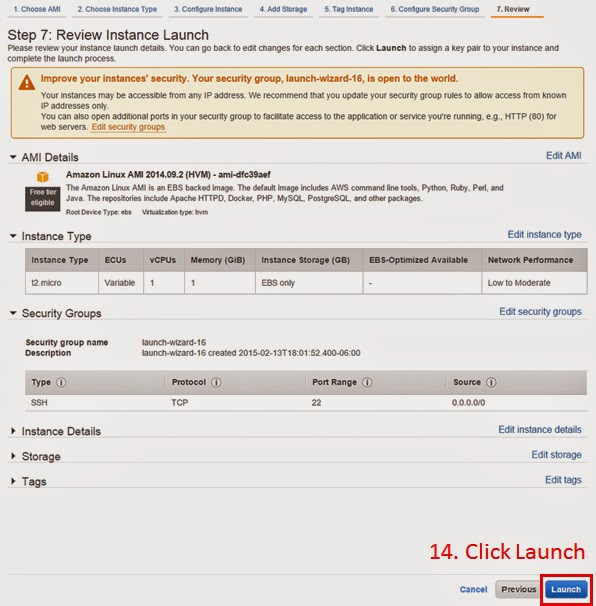
[](http://3.bp.blogspot.com/-M54fXYBND4Y/VN6VuLsvPnI/AAAAAAAAAD0/xrQpgy9tbZQ/s1600/ec2_06_.jpg)

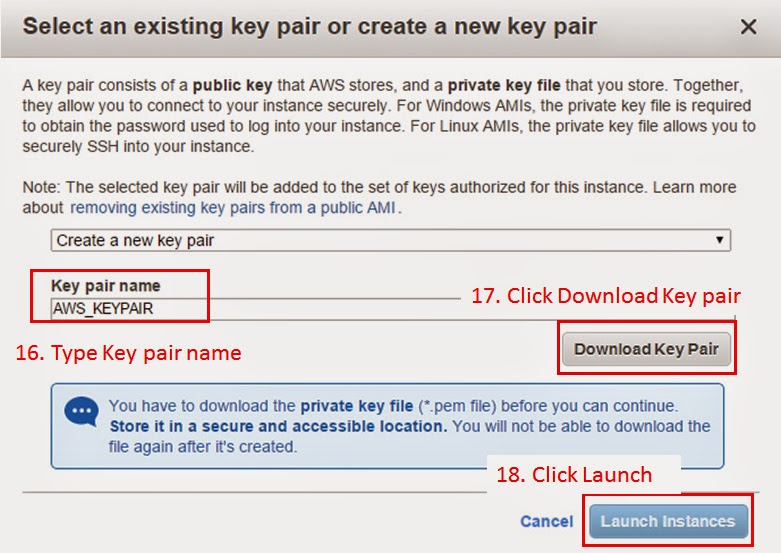
[](http://4.bp.blogspot.com/-OJLrhnMlRh4/VN6VuZh_cDI/AAAAAAAAAD4/dGo21O5nERw/s1600/ec2_07_.jpg)

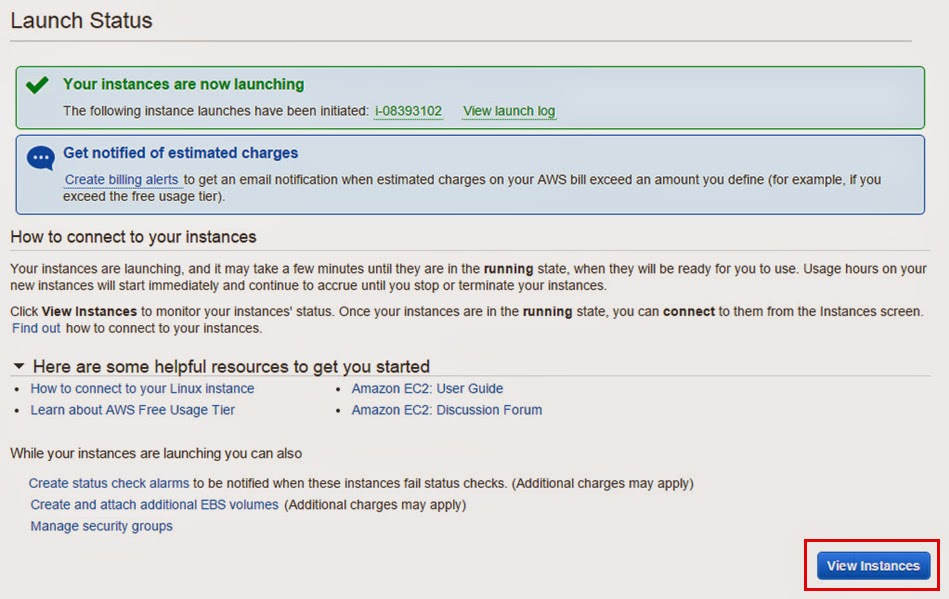
[](http://2.bp.blogspot.com/-bh36Qu7JS5I/VN6VuXQgOTI/AAAAAAAAAEI/YA3aZ_-C-FY/s1600/ec2_08_.jpg)

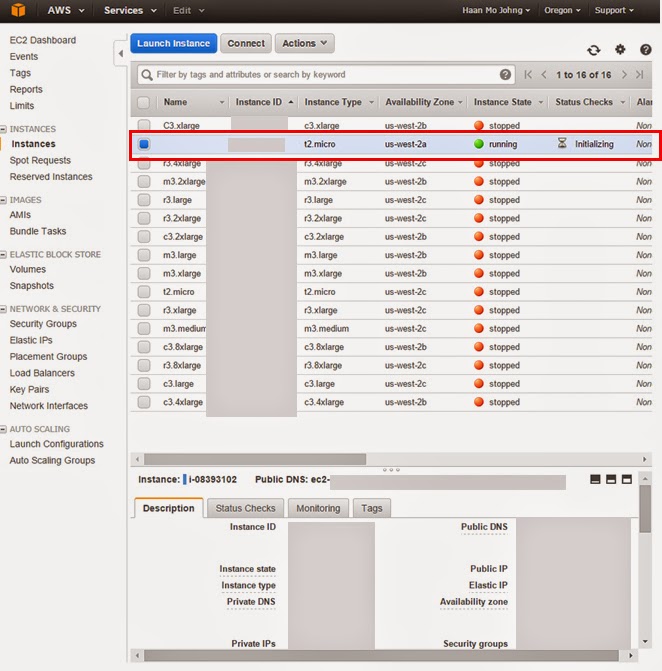
[](http://1.bp.blogspot.com/-oiyr-k-n-QU/VN6VugxZj5I/AAAAAAAAAD8/7JqttD2QHDw/s1600/ec2_09_.jpg)

[](http://1.bp.blogspot.com/-D5EFipA7rZM/VN6Vu4VogqI/AAAAAAAAAEA/JhkjVXgIF4c/s1600/ec2_10_.jpg)

[](http://1.bp.blogspot.com/-BXBm-ANQ5X0/VN6VvC-DDlI/AAAAAAAAAEs/4QQeGDROhpM/s1600/ec2_11_.jpg)

[](http://3.bp.blogspot.com/-OqCdVDApuxI/VN6VvfZ34sI/AAAAAAAAAEQ/72G0n5jbll4/s1600/ec2_13.jpg)

[](http://1.bp.blogspot.com/-iCHyCiRnN0g/VN6VvqhrB1I/AAAAAAAAAEU/42XHSLMmkKM/s1600/ec2_14_.jpg)

[](http://4.bp.blogspot.com/--PnOqBCUk5g/VN6VwKUv7DI/AAAAAAAAAEY/E9uW6-ifryA/s1600/ec2_15_.jpg)

*Note:*

*The links describe how to create Linux instance.*

*You need to create Ubuntu instance and create keys for Linux and Ubuntu separately.*

1. **Connect to EC2 instance with PuTTY**

1. Execute PUTTYGEN

Putty and PuttyGen Download

2. Click Conversions -> click Import key

3. Click an "AWS KEY PAIR.per" file that you made while creating EC2-instance.

-> click Open

4. Click Save private key -> click Y -> type file name (AWS\_KEYPAIR.ppk)

-> Click Save

5. Go to AWS console -> click EC2 instance that you created -> check Public IP address

6. Execute Putty.

7. Type the IP address on Host Name (or IP address) box.

8. Click SSH(on left layer)->click Auth(on left layer) -> click Browser -> click the AWS\_KEYPAIR.ppk file -> click open.

9. Click Connection(on left layer) -> click Data(on left layer) -> type ec2-user on Auto-login username box.

10. Click Open

1. **Install Cassandra 2.2.2 on AWS Ubuntu**

**1. Check Auto-login username should be "ubuntu"**

**2. Cassandra version 2.2.2 (Higher versions can be used)**

**3. Change root password**  
sudo passwd root  
yourpw  
  
su  
yourpw  
  
**4. Install Oracle Java 7 in Ubuntu or Linux Mint via PPA**  
sudo add-apt-repository ppa:webupd8team/java  
sudo apt-get update  
sudo apt-get install oracle-java7-installer

**5. Set Java environment variables**

sudo apt-get install oracle-java7-set-default

sudo apt-get update

**6. Install cassandra**

mkdir /usr/local/cassandra

cd /usr/local/cassandra

wget http://apache.cs.utah.edu/cassandra/2.2.2/apache-cassandra-2.2.2-bin.tar.gz

tar xvfz apache-cassandra-2.2.2-bin.tar.gz

**7. check IP addresse of instances**

ifconfig

//example

172.31.36.171

172.31.42.247

172.31.42.151

**8. Make a backup for all nodes**

cp apache-cassandra-2.2.2/conf/cassandra.yaml apache-cassandra-2.2.2/conf/cassandra.yaml\_bak

**9. Modify configuration for all nodes.**

nano apache-cassandra-2.2.2/conf/cassandra.yaml

*Note:*

*Find listen\_address: and seeds: (Ctrl+W) and modify it*

*listen\_address should be own IP of each instance.*

listen\_address: 172.31.36.171

seeds:  "172.31.36.171,172.31.42.247,172.31.42.151"  
rpc\_address: 0.0.0.0  
broadcast\_rpc\_address: 1.2.3.4

**10. Excute cassandra**

/usr/local/cassandra/apache-cassandra-2.2.2/bin/cassandra

/usr/local/cassandra/apache-cassandra-2.2.2/bin/nodetool -h localhost -p 7199 status

*Note: All 3 nodes should be ready*

/usr/local/cassandra/apache-cassandra-2.2.2/bin/cqlsh

**11. Create keyspace and table for YCSB benchmark**

CREATE KEYSPACE usertable WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication\_factor' : 3};

use usertable;

CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);

describe table data;

1. **Install YCSB and benchmark Cassandra with 3 nodes**

**Install YCSB**

curl -O --location https://github.com/brianfrankcooper/YCSB/releases/download/0.3.0/ycsb-0.3.0.tar.gz  
tar xfvz ycsb-0.3.0.tar.gz  
cd ycsb-0.3.0

**Create workload files in workload folder (do not include ------ )**  
nano workload\_test10  
  
-------------------------------------------------------------  
recordcount=214748  
operationcount=100000  
workload=com.yahoo.ycsb.workloads.CoreWorkload  
readallfields=true  
readproportion=0.5435  
updateproportion=0.4565  
scanproportion=0  
insertproportion=0  
requestdistribution=zipfian

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

nano workload\_test40  
  
-------------------------------------------------------------  
recordcount=536871  
operationcount=400000  
workload=com.yahoo.ycsb.workloads.CoreWorkload  
readallfields=true  
readproportion=0.5435  
updateproportion=0.4565  
scanproportion=0  
insertproportion=0  
requestdistribution=zipfian

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

nano workload\_test320  
  
-------------------------------------------------------------  
recordcount=3543348  
operationcount=3200000  
workload=com.yahoo.ycsb.workloads.CoreWorkload  
readallfields=true  
readproportion=0.5435  
updateproportion=0.4565  
scanproportion=0  
insertproportion=0  
requestdistribution=zipfian

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

nano workload\_test640

-------------------------------------------------------------  
recordcount=6120328  
operationcount=6400000  
workload=com.yahoo.ycsb.workloads.CoreWorkload  
readallfields=true  
readproportion=0.5435  
updateproportion=0.4565  
scanproportion=0  
insertproportion=0  
requestdistribution=zipfian

------------------------------------------------------------  
  
nano workload\_test1200  
  
------------------------------------------------------------  
recordcount=10630044  
operationcount=12000000  
workload=com.yahoo.ycsb.workloads.CoreWorkload  
  
readallfields=true  
  
readproportion=0.5435  
updateproportion=0.4565  
scanproportion=0  
insertproportion=0  
  
  
requestdistribution=zipfian  
------------------------------------------------------------

**Create a folder for results**

mkdir results\_Cassandra

**Load data**

./bin/ycsb load cassandra-10 -P workloads/workload\_test10 -p hosts=172.31.36.171 -threads 10 -p columnfamily=data -s > results\_Cassandra/load\_10

*Note:*

*Check your IP address and run just one commend for each experiment.*

**Run benchmark**

./bin/ycsb run cassandra-10 -P workloads/workload\_test10 -p hosts=172.31.36.171 -threads 10 -p columnfamily=data -s > results\_Cassandra/run\_10

*Note:*

*Check your IP address*

**Clean loaded data for next benchmark**

drop table data;  
CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);

**Repeat benchmark with different configurations.**

*//Note: benchmark for 40 concurrent users*

./bin/ycsb load cassandra-10 -P workloads/workload\_test40 -p hosts=172.31.36.171 -threads 40 -p columnfamily=data -s > results\_Cassandra/load\_40

./bin/ycsb run cassandra-10 -P workloads/workload\_test40 -p hosts=172.31.36.171 -threads 40 -p columnfamily=data -s > results\_Cassandra/run\_40

drop table data;  
CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);

*//Note: benchmark for 320 concurrent users*

./bin/ycsb load cassandra-10 -P workloads/workload\_test320 -p hosts=172.31.36.171 -threads 320 -p columnfamily=data -s > results\_Cassandra/load\_320

./bin/ycsb run cassandra-10 -P workloads/workload\_test320 -p hosts=172.31.36.171 -threads 320 -p columnfamily=data -s > results\_Cassandra/run\_320

drop table data;  
CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);

*//Note: benchmark for 640 concurrent users*

./bin/ycsb load cassandra-10 -P workloads/workload\_test640 -p hosts=172.31.36.171 -threads 640 -p columnfamily=data -s > results\_Cassandra/load\_640

./bin/ycsb run cassandra-10 -P workloads/workload\_test640 -p hosts=172.31.36.171 -threads 640 -p columnfamily=data -s > results\_Cassandra/run\_640

drop table data;  
CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);

*//Note: benchmark for 640 concurrent users*

./bin/ycsb load cassandra-10 -P workloads/workload\_test1200 -p hosts=172.31.36.171 -threads 1200 -p columnfamily=data -s > results\_Cassandra/load\_1200

./bin/ycsb run cassandra-10 -P workloads/workload\_test1200 -p hosts=172.31.36.171 -threads 1200 -p columnfamily=data -s > results\_Cassandra/run\_1200

drop table data;

CREATE TABLE usertable.data ( key blob, column1 text, value blob, PRIMARY KEY (key, column1)) WITH COMPACT STORAGE AND CLUSTERING ORDER BY (column1 ASC);