Since you're creating a larger cluster now, the NameNode needs more memory to keep track of the metadata than what you used in the previous lesson.

From your EC2 dashboard, launch an instance. Again, choose Ubuntu server. The t2.micro instance won't work anymore, instead, you should use m3.large since it has 7.5 GB of memory.

Just one instance is fine for now. Next, choose 30 GB of storage for the instance.

TAG it "Ambari server." ami-5e63d13e

Then you're going to configure the security group. You'll need to add a rule to allow access to the Ambari web client. Add a Custom TCP Rule, set the port to 8080, and leave the source as all these zeros. Here you'll be leaving it open to anyone. Typically you'd restrict the source address so that only your organization could access the client.

Finally, launch your instance! Again, you'll need a private key, you'll most likely want to use the one you created in the previous lesson. You should be able to select it from the drop down menu. If not, create a new key and download it.

```
ssh -i /path/to/key_file.pem ubuntu@DNS_hostname
```

ssh -i "jingamasti.pem" ubuntu@ec2-54-201-242-111.us-west-2.compute.amazonaws.com

```
sudo apt-get update && sudo apt-get dist-upgrade -y

sudo apt-get install ntp -y

sudo service ntp status

sudo service ntp start

sudo nano /etc/rc.local

#Add these lines:

if test -f /sys/kernel/mm/transparent_hugepage/enabled; then
    echo never > /sys/kernel/mm/transparent_hugepage/enabled
fi

if test -f /sys/kernel/mm/transparent_hugepage/defrag; then
    echo never > /sys/kernel/mm/transparent_hugepage/defrag
fi

#Create SnapShot save the instance to an image, call it something like “Ambari
#node.” Make sure to check “No reboot,” you'll be using this instance as the
#Ambari server.

Save the instance to an image, call it “Ambari” Make sure to check “No reboot,”
you'll be using this instance as the Ambari server.

#You can easily create any number of nodes for our cluster by launching
#instances from the image you just created.

#Launch 6 nodes or more for the cluster

#Install and start Ambari server Ambari version 2.2.0

scp -i key_file.pem key_file.pem ubuntu@server_public_hostname:~/.ssh/

https://cwiki.apache.org/confluence/display/AMBARI/Ambari+User+Guides

cd /etc/apt/sources.list.d

sudo wget http://public-repo-
1.hortonworks.com/ambari/ubuntu14/2.x/updates/2.2.0.0/ambari.list
```

```
#Add the key to authenticate Ambari package

sudo apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD

sudo apt-get update && sudo apt-get dist-upgrade -y

sudo apt-get install ambari-server -y

sudo ambari-server setup

sudo ambari-server start

sudo ambari-server status
```

52.66.119.133:8080

ADMIN

ADMIN

ip-172-31-8-174.ap-south-1.compute.internal
ip-172-31-8-175.ap-south-1.compute.internal
ip-172-31-8-176.ap-south-1.compute.internal
ip-172-31-8-177.ap-south-1.compute.internal
ip-172-31-8-178.ap-south-1.compute.internal
ip-172-31-8-179.ap-south-1.compute.internal

Transparent Huge Pages Issues
JDK Issues
Disk Issues
Repository Issues
Firewall Issues
Process Issues
Package Issues
File and Folder Issues
Service Issues
User Issues
Misc Issues
Alternatives Issues
Reverse Lookup Issues
Hostname Resolution Issues

Slider
SmartSense
ZooKeeper
YARN + MapReduce2
HDFS
Ambari Metrics

sudo su – hdfs

hadoop jar /usr/hdp/2.3.6.0-3796/hadoop-mapreduce/hadoop-mapreduce-examples-*.jar teragen 500000 random-data

hadoop jar /usr/hdp/2.3.6.0-3796/hadoop-mapreduce/hadoop-mapreduce-examples-*.jar terasort random-data sorted-data

hadoop jar /usr/hdp/2.3.6.0-3796/hadoop-mapreduce/hadoop-mapreduce-client-jobclient-tests.jar TestDFSIO -write -nrFiles 10 -fileSize 50MB

hadoop jar /usr/hdp/2.3.6.0-3796/hadoop-mapreduce/hadoop-mapreduce-client-jobclient-tests.jar TestDFSIO -read -nrFiles 10 -fileSize 50MB