CCA 131 preparation

Ismac :

Iot :internet of things

S :social

M:mobility

A:analytics

C:cloud plat form

Lambda architecture:

3 components:

1.batch layer -> sqoop flump

2.servicing layer -> hbase, mongo cassendra

3.streaming layer -> spark

AWS:

1.billing dashboards

2.simple monthly calculator

3. Availability zones and regions

One region associated with more than one availability zone by providing high availability and scalability and high performance.

15 regions across the world. Head quarter is Virginia

Ec2 instance :

Elastic computing cloud. Its like creating cloud virtual environment on AWS cloud and we can use those resources and develop the applications on top of operating system

By default it gets created with on demand ec2 instance. If we want to go with the learning and practice purpose, we need to go with the spot requests of AWS ec2 instances.

S3 – Simple storage service

Its one of the service to store the data. It will be created like bucket – objects like that… bucket name should be unique.

How to login into ec2 instance ?

There will be multiple ways to connect to ec2 linux ami

1. Like using putty
2. Like using git bash

With putty -> we need to take that .pem file convert into ppk file and connect to it.

With git bash -> we can directly connect it.

AMI :

Amazon machine image:

Best example is : this is like template. Rather than creating eveytime ec2 instance and use it.

What we can do is.

You launch 5 ec2 instances.. set up all the software prerequisites and install and configure it.

Then we can do like.. we can create image on top of this ec2 –instance and terminate ec2 –instance and next time you login with AMI you can use directly.

Its like providing re-usability.

It will create snapshot for that EBS if we create AMI.

Practicals:



AWS RDS : Relational data service:

This service is for database service. Its mainly used to install different types of databases.

Amazon aurora

Mysql

Postgresql

SQL server

<https://www.heidisql.com/download.php>

heidisql will provide interface to access database.

HDFS:

Hadoop distributed file system

* Storage layer – high availability – scalabity – replication – fault tolerance

HDFS architecture -> master and slave

64 MB

128 MB

256 MB

YARN:

MYSQL installation:

Connect to cento 6.5 or 6.9 or 7 AMI.

Then connect that instance using putty.

First run

yum update -y

to run yum commands we require sudo access.

sudo su -

yum update –y

attached mysql database script.



Centos 7 version mysql installation:

yum update -y

yum install –y wget

wget <http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm>

rpm –ivh mysql-community-release-el7-5.noarch.rpm

yum –y install mysql-server mysql-connector-java

ls -ltr /usr/share/java/mysql\*

/usr/bin/mysql\_secure\_installation

mysql -u root -h localhost –p

password:password

mysql>show databases;

mysql> create user ‘temp’@’%’ identified by ‘password123’;

mysql> grant all privileges on \*.\* to ‘temp’@’%’ with grant option;

mysql>flush privileges;

mysql>quit

mysql -u temp -h localhost -p

password:password123

mysql> create database reportmanager;

mysql> use reportmanager;

mysql>create user ‘reportmanager’@’%’ identified by ‘password123’;

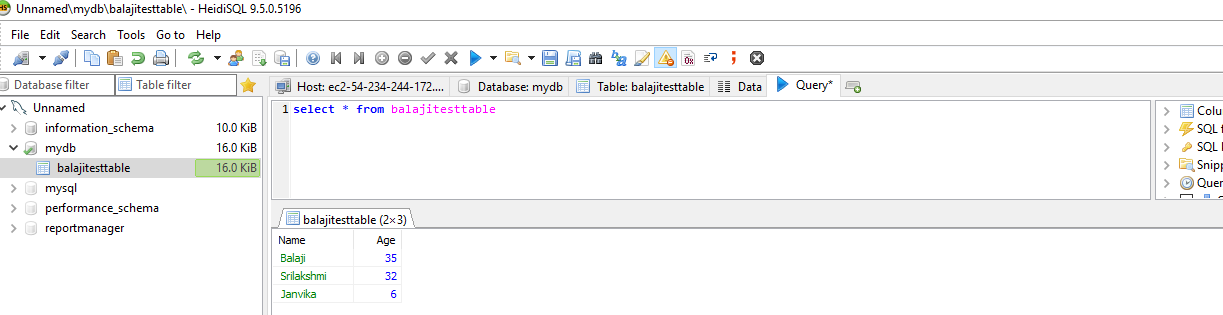
mysql>grant all privileges on \*.\* to ‘reportmanager’@’%’ with grant option;

mysql>flush privileges;

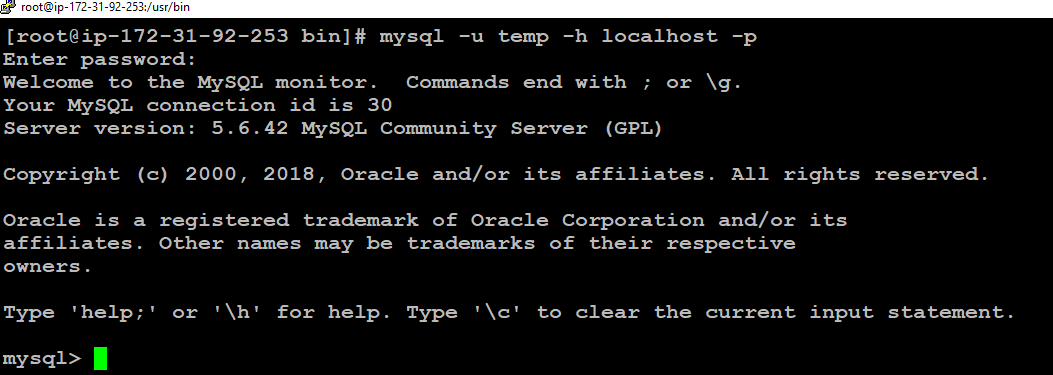
open session manager in heidisql:

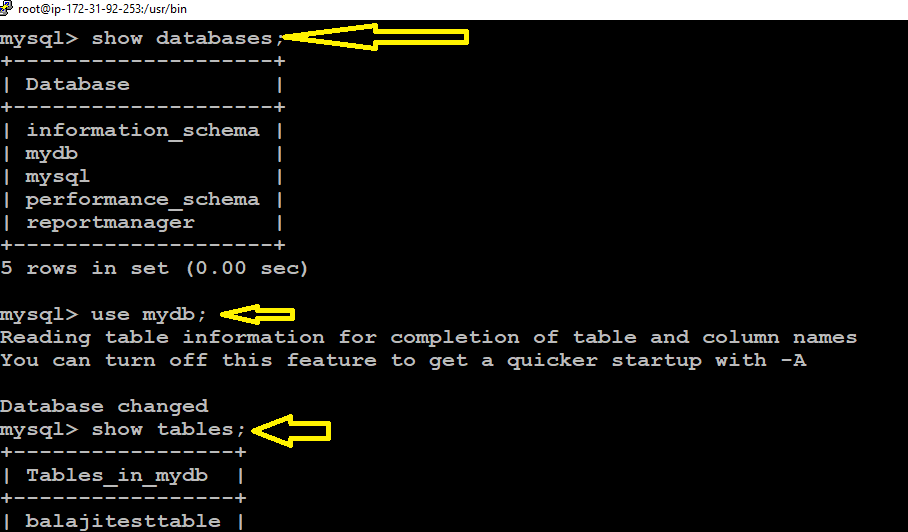
you can connect of AWS public DNS, username suppose temp password give password123

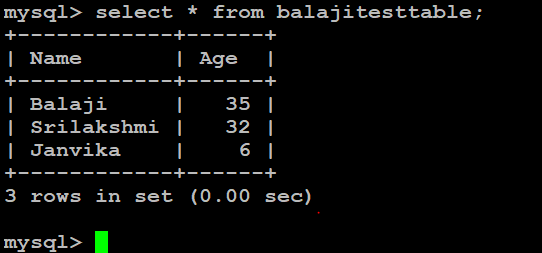
you can see in Heidi sql user interface see this tables and databases and users.



Same thing I can see mysql also from putty.







Prepare AWS AMI for Cloudera Installation:



AMI ID for centos 7 is ami-4bf3d731

1.First create ec2-instance with centos 7.

2. Prepare AWS AMI for Cloudera Installation

T2.large -> 2 CPU and 8 GB RAM.

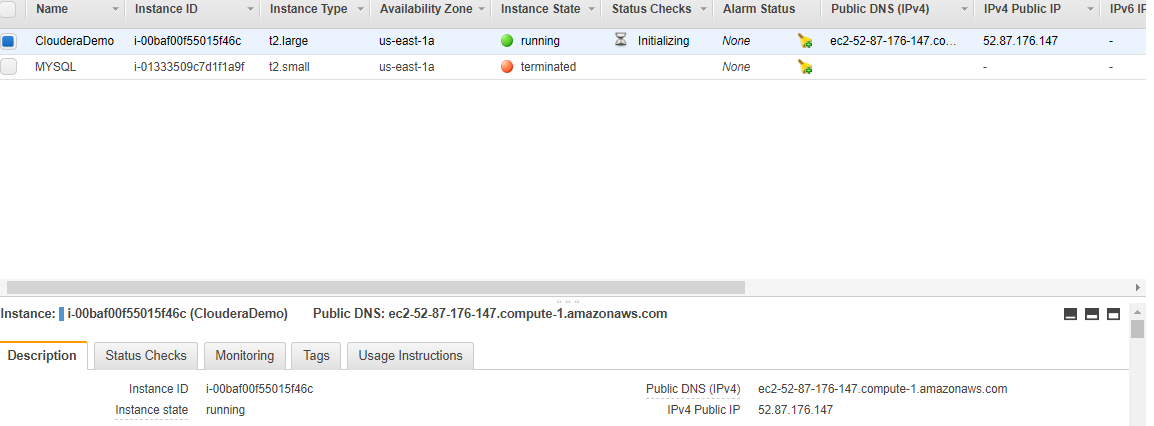
Number of instances 1

By default EBS size is 8 GB . change this value to 30 GB.

Security Group;ClouderaDemo

Keypair:ClouderaTest

Ec2 instance is ready now.



Next is launch with putty .

Once connected to putty.

Prerequisites:

1.disable SELINUX

2.Turn off the firewalls

3. resize the ebs size

3.change the swappiness value in /etc/sysctl.conf

4.change THP to never and add entry in /etc/rc.local

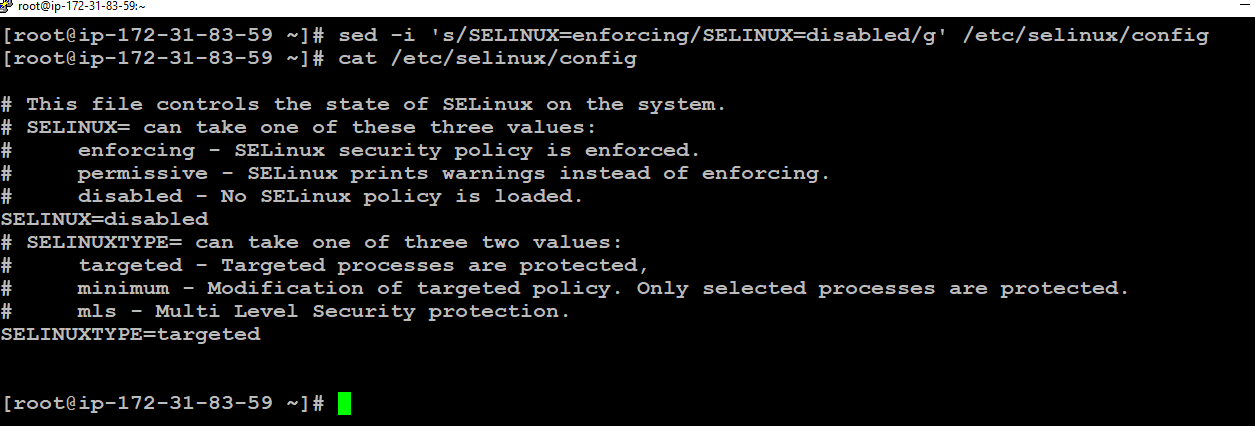
5. Some people change hostname as well.

6. Install supporting RPM”S

Let’s do it.

Command is

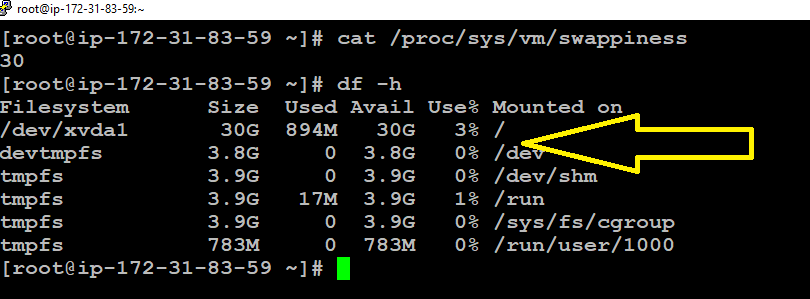
sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config



In centos 7 using aws ec2 , no need of turning off firewalld, its automatically disabled.

Resize the EBS size: if not resized.

resize2fs /dev/xvda1



Swappiness:

cat /proc/sys/vm/swappiness

30

Change this to 1 using like below

echo "vm.swappiness=1" >> /etc/sysctl.conf

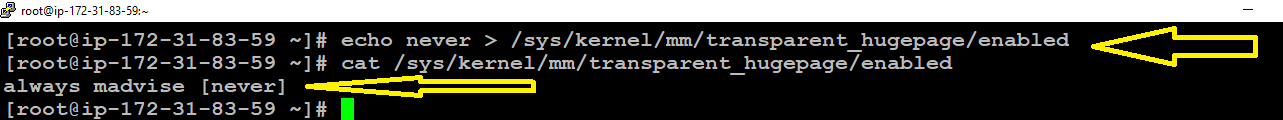
install wget package

yum install –y wget

change the THP(transparent huge page compaction) page to never.

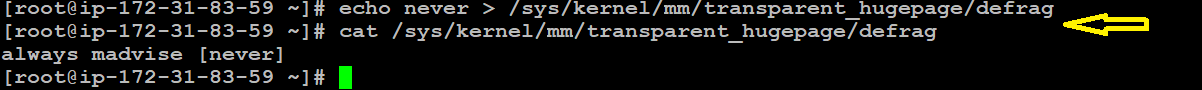
echo never > /sys/kernel/mm/transparent\_hugepage/enabled

cat /sys/kernel/mm/transparent\_hugepage/enabled



echo never > /sys/kernel/mm/transparent\_hugepage/defrag

cat /sys/kernel/mm/transparent\_hugepage/defrag



Nexttime system or instance rebooted, it should be these values to be reservered, then add entry in /etc/rc.local file.

echo "echo never > /sys/kernel/mm/transparent\_hugepage/enabled" >> /etc/rc.local

echo "echo never > sys/kernel/mm/transparent\_hugepage/defrag" >> /etc/rc.local

cat /etc/rc.local

reboot the ec2 instance

Reconnected the same instance using putty. I can see changes are persist.

Creating image (AMI) to my ec2 instance so that I can use the same AMI for number of instances.

Image name is : Balaji\_Cloudera

I have verified All above configurations are persist.

**Now lets statrt cloudera manager installation using your AMI:**

****

1.connect your AMI Ec2 instance

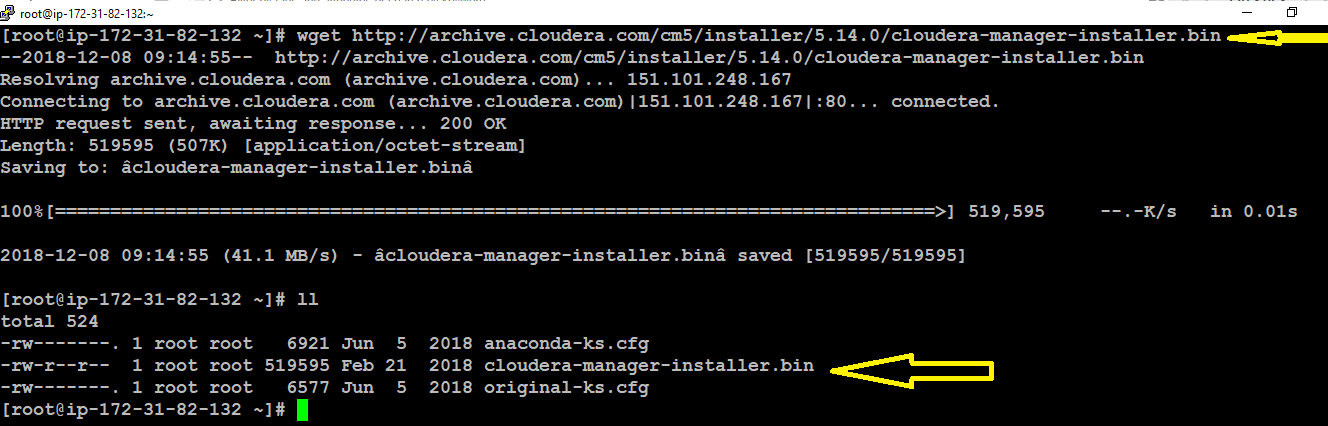
2. then install wget package

yum -y install wget

1. After that open cloudera website like below

<http://archive.cloudera.com/cm5/installer/5.14.0/>

wget <http://archive.cloudera.com/cm5/installer/5.14.0/>

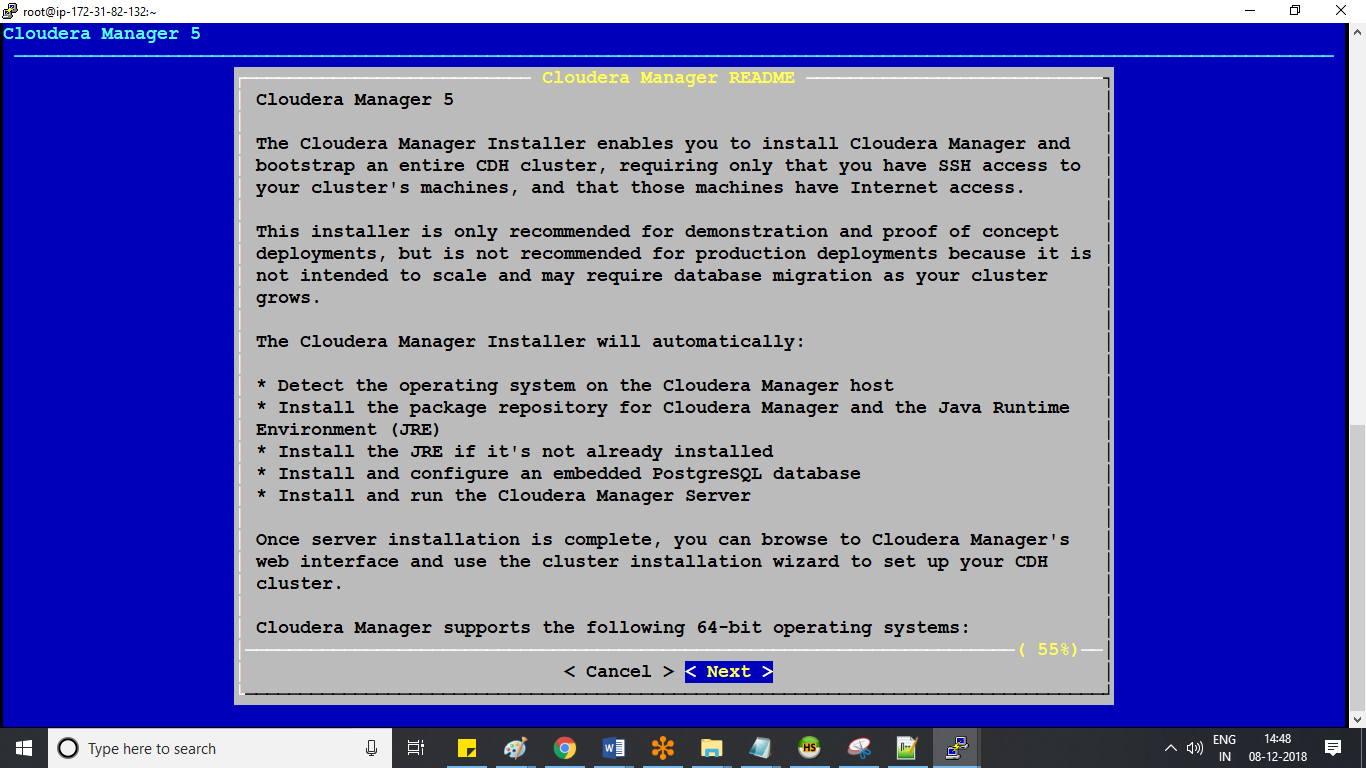


Run the below command to get install JDK, Cloudera manager server , agent and CMS services and embedded database

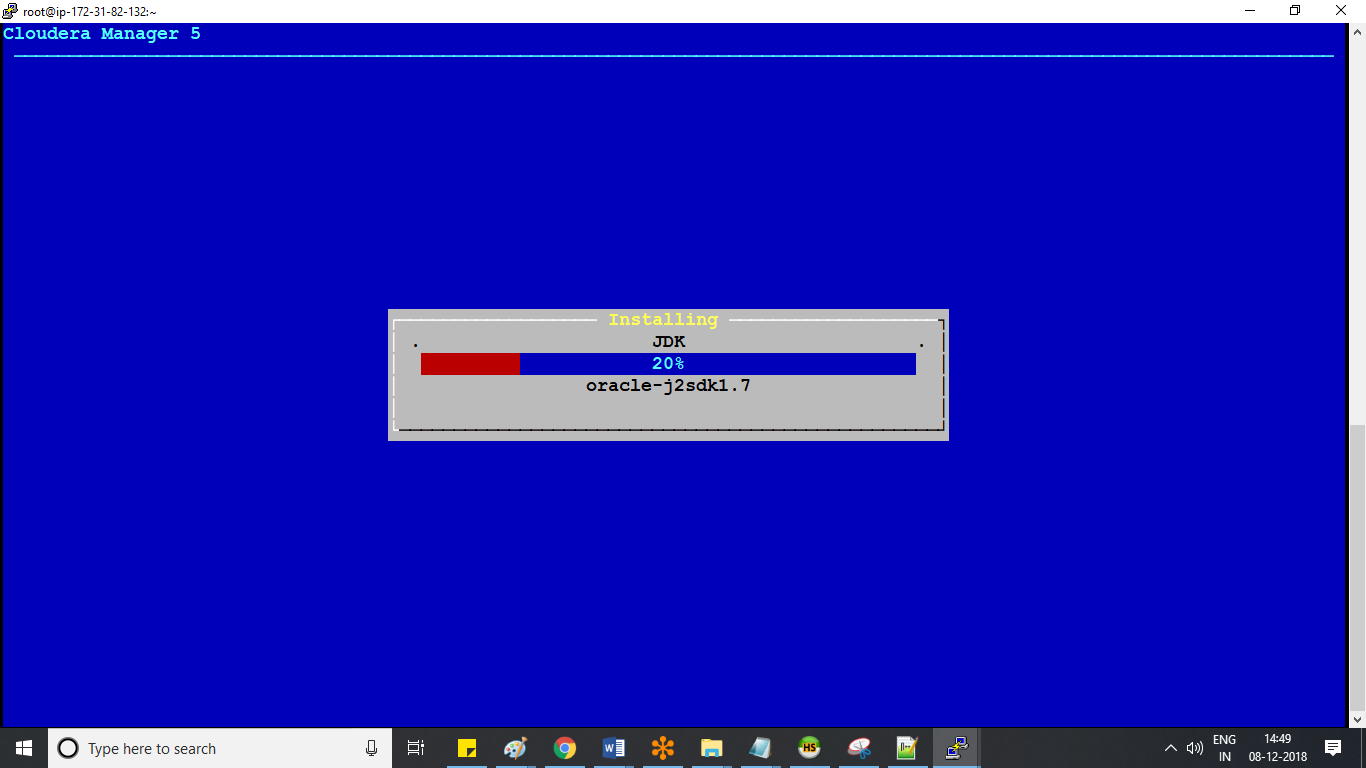
chmod u+x cloudera-manager-installer.bin

./cloudera-manager-installer.bin

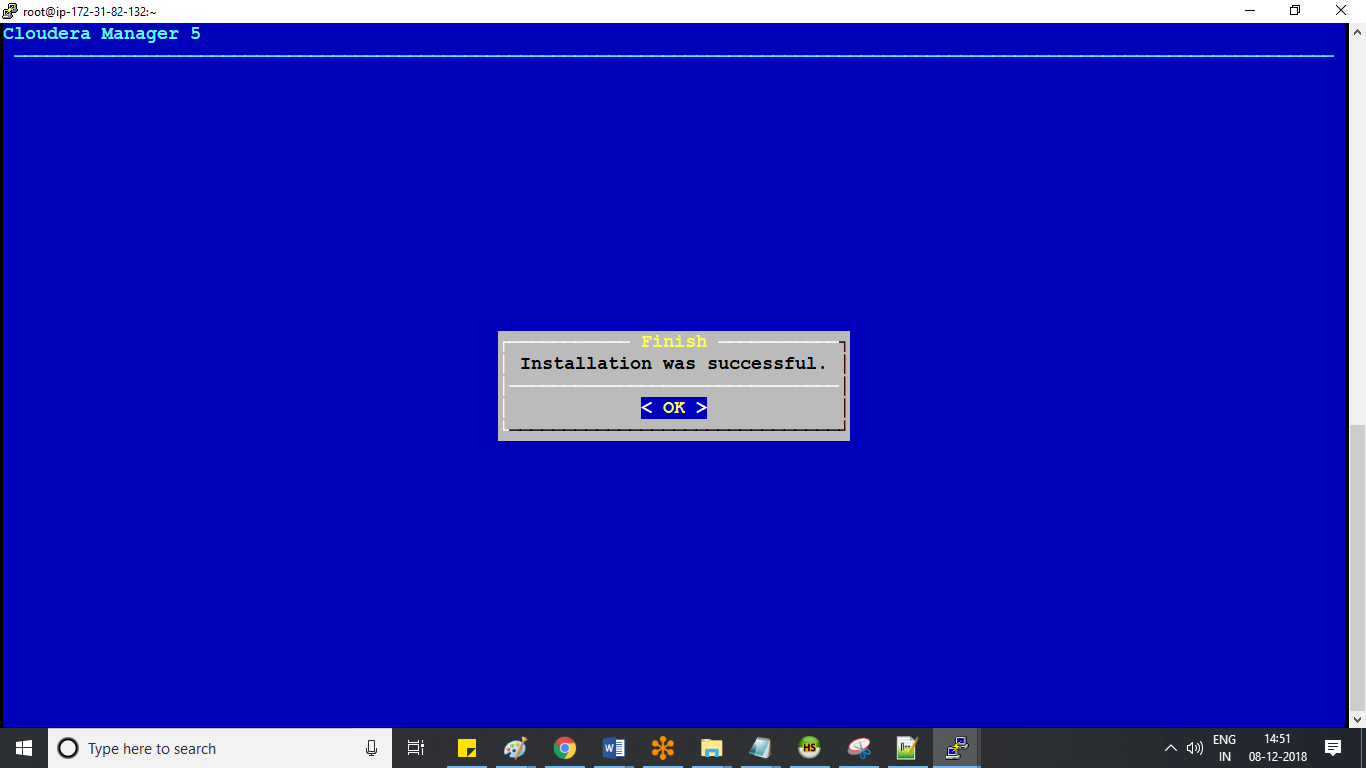
After that it will open below window.



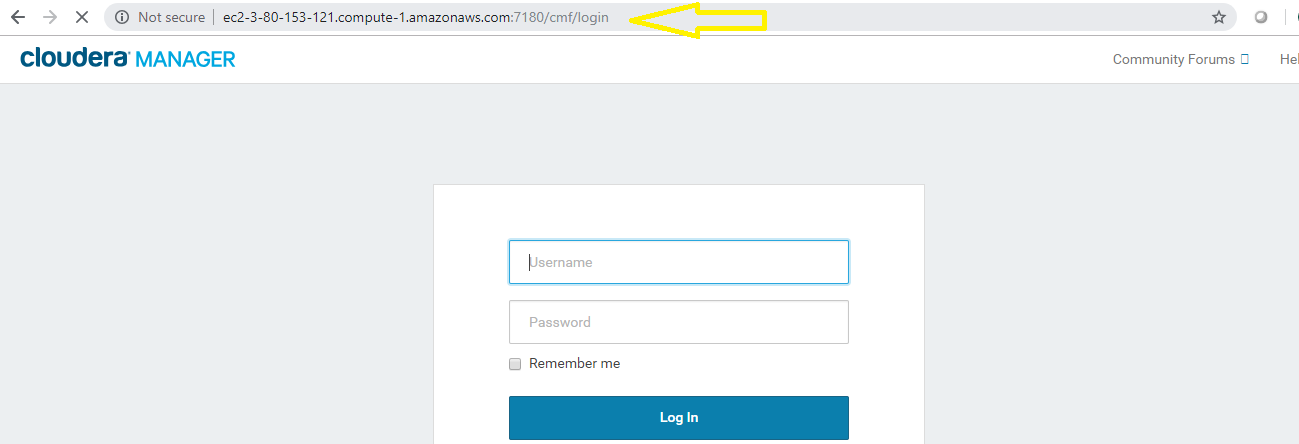
First install JDK



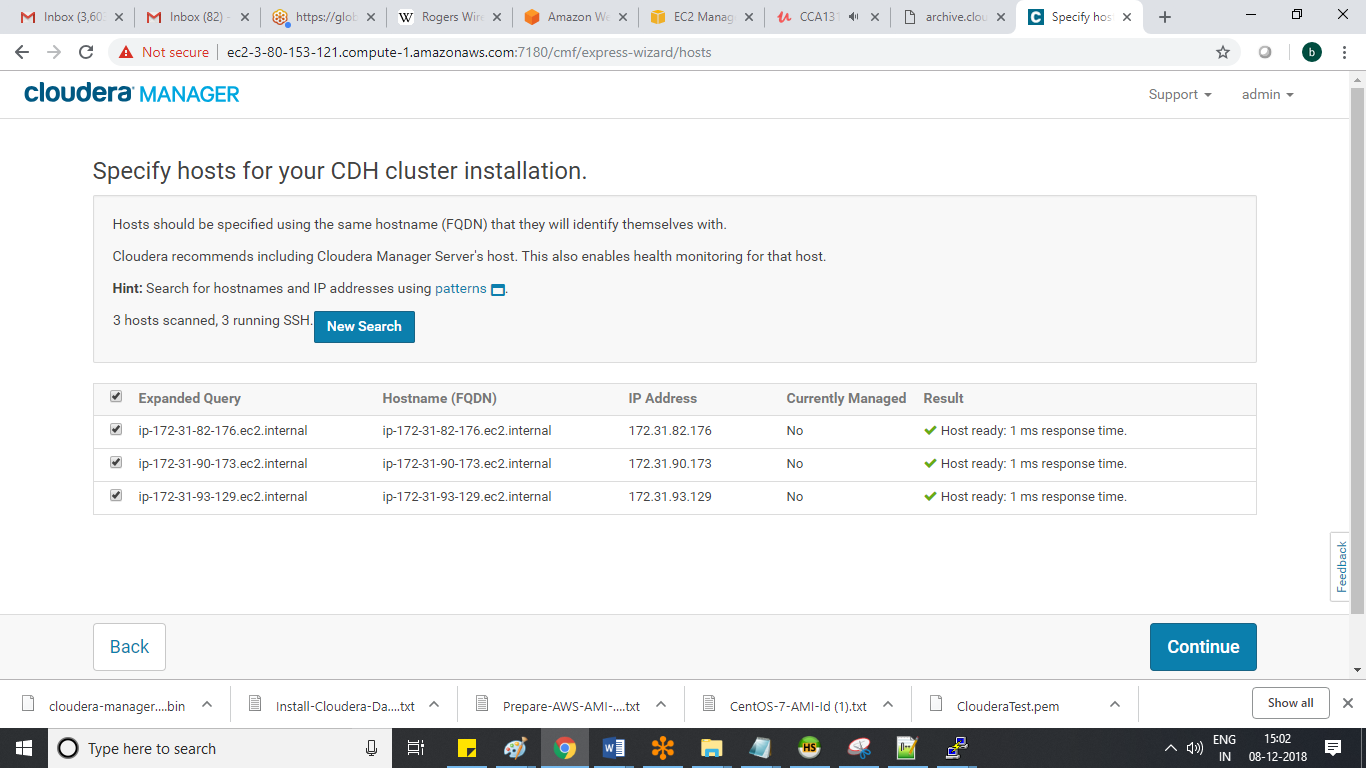
Installing cloudera manager and embedded and it will show successful.

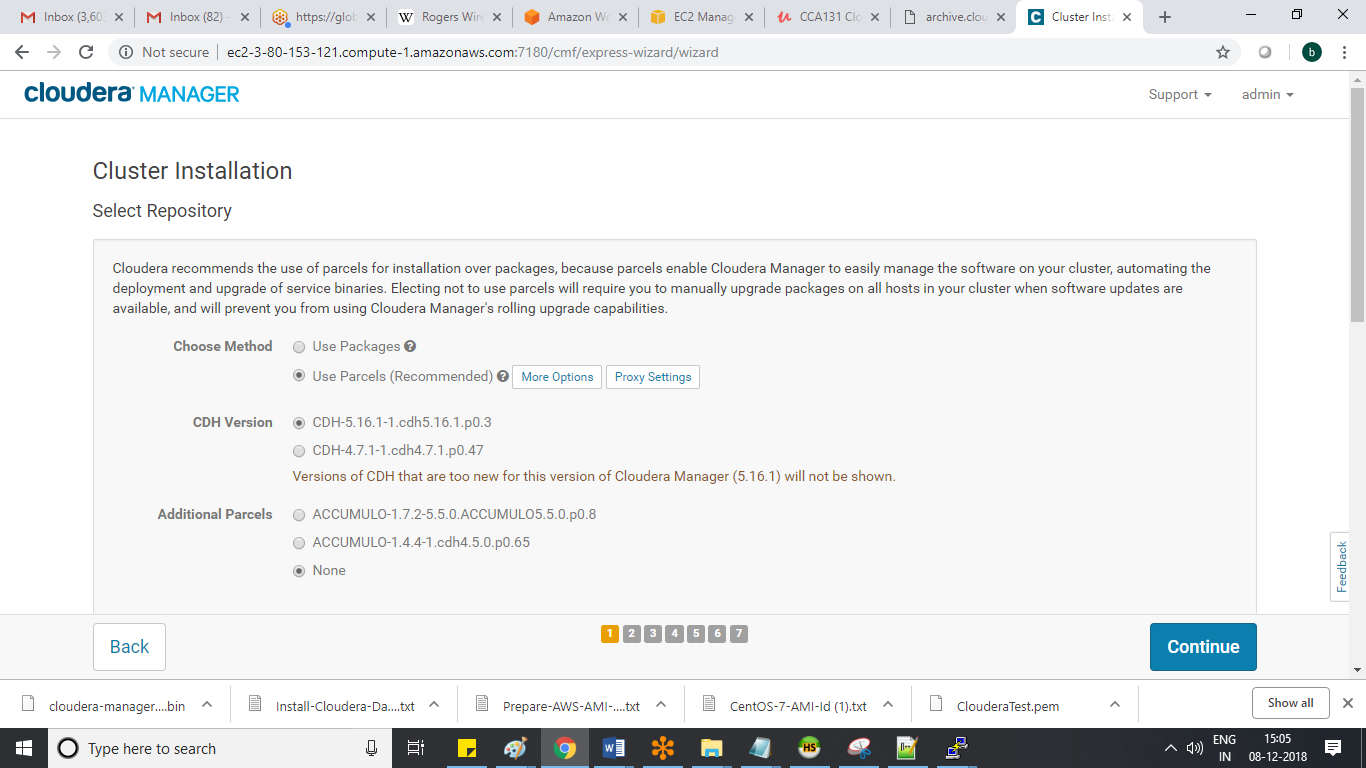


After few min, open public DNS ip and with 7180 and try to access cloudera manager web UI.

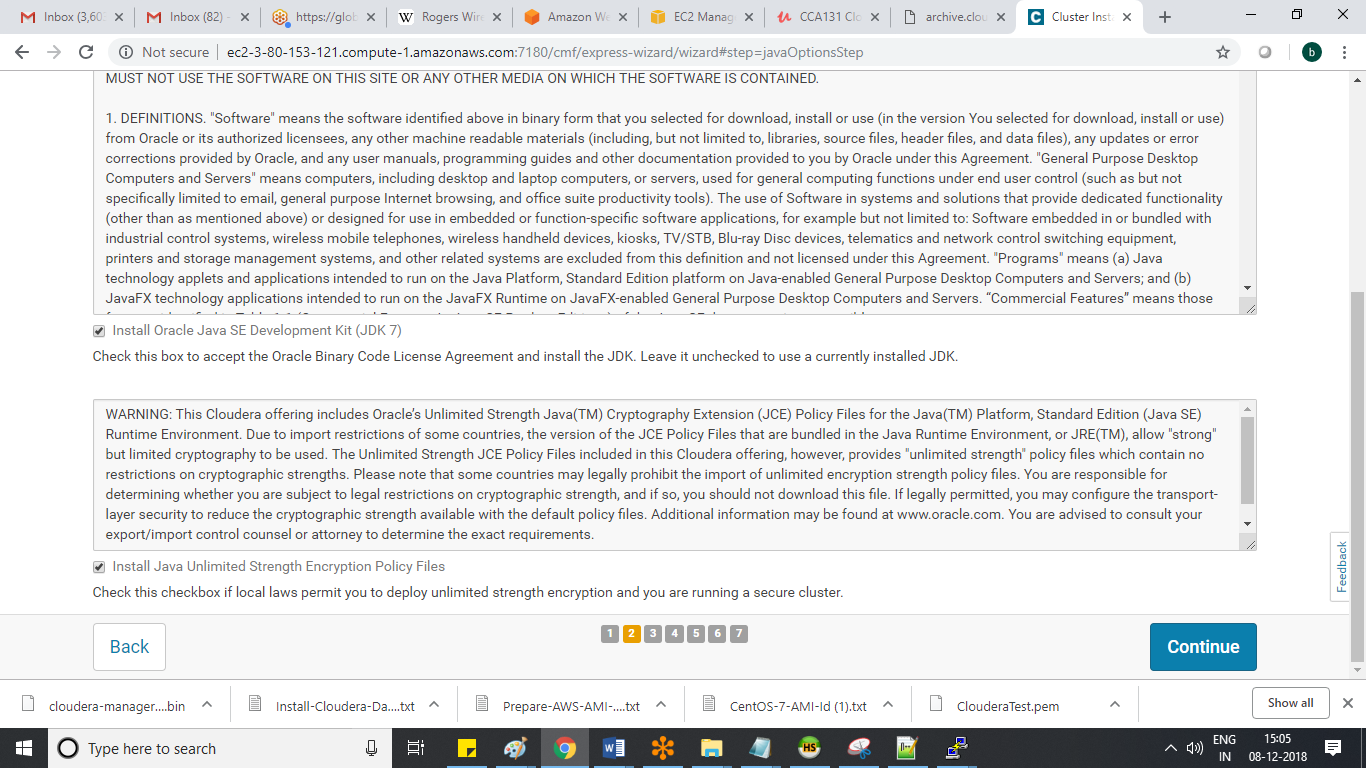


See its opened , now give admin and admin

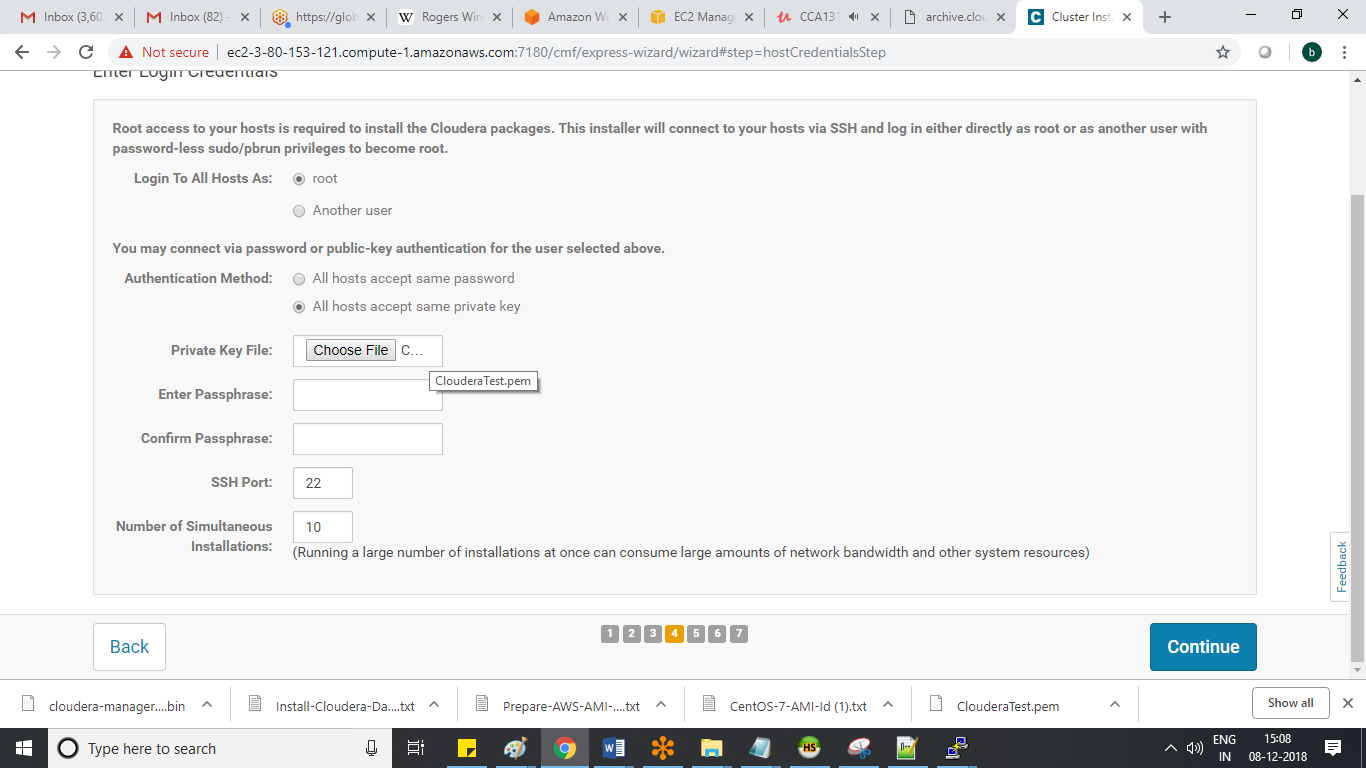


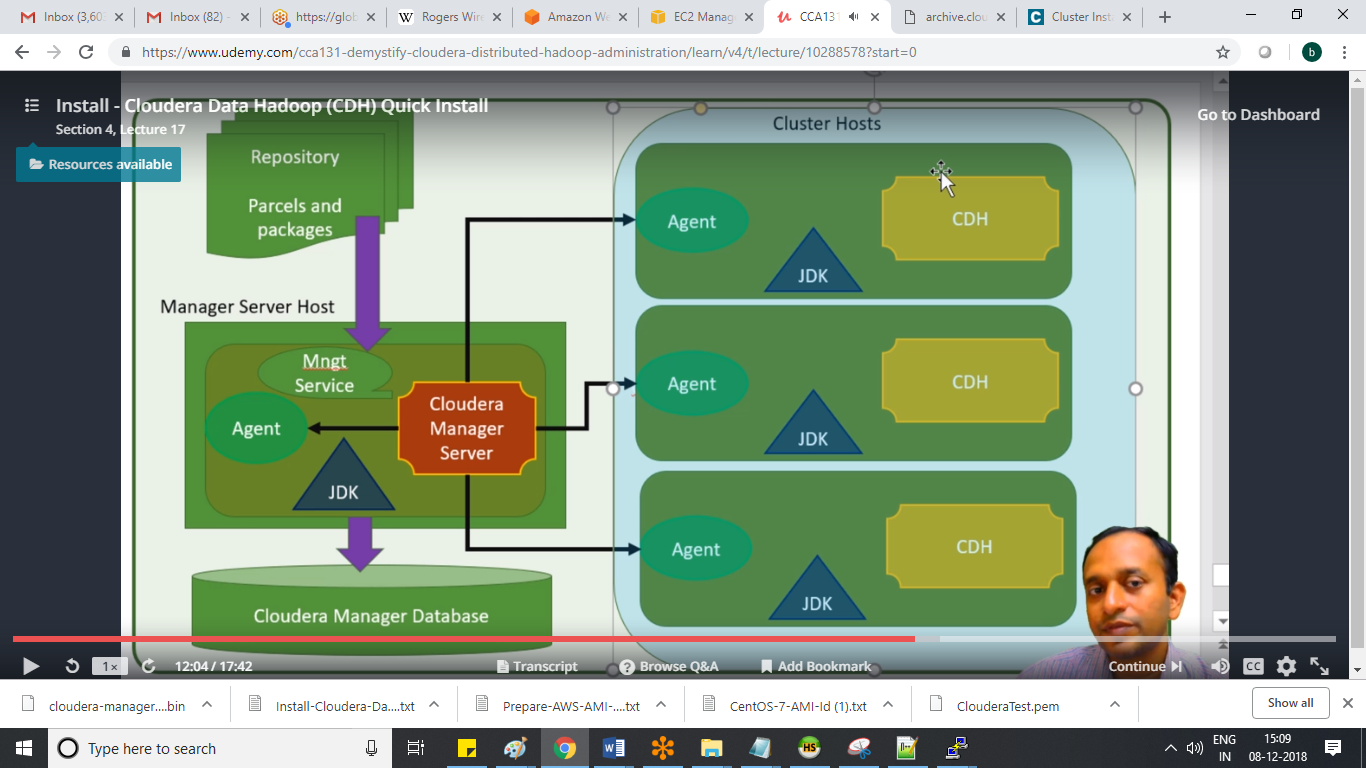


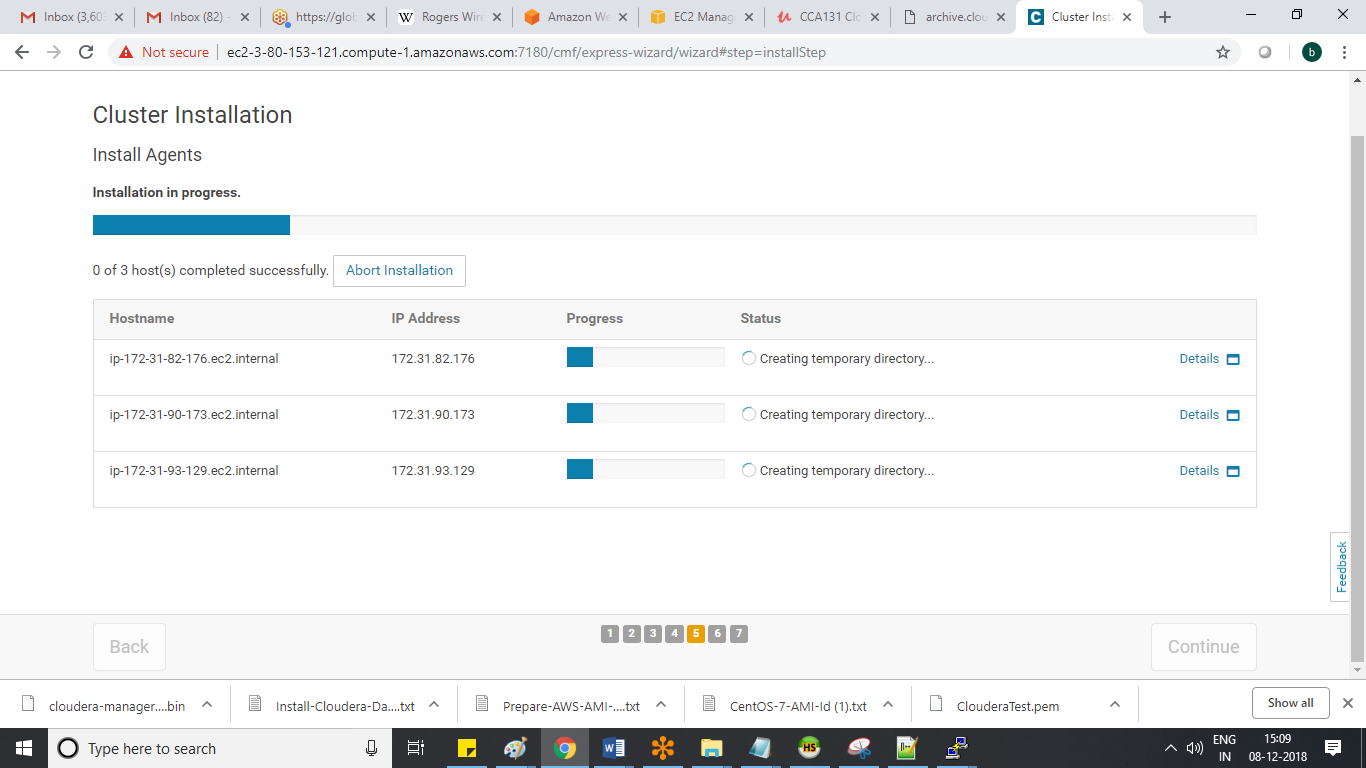
Use parcel method and select java and JCE polciies.

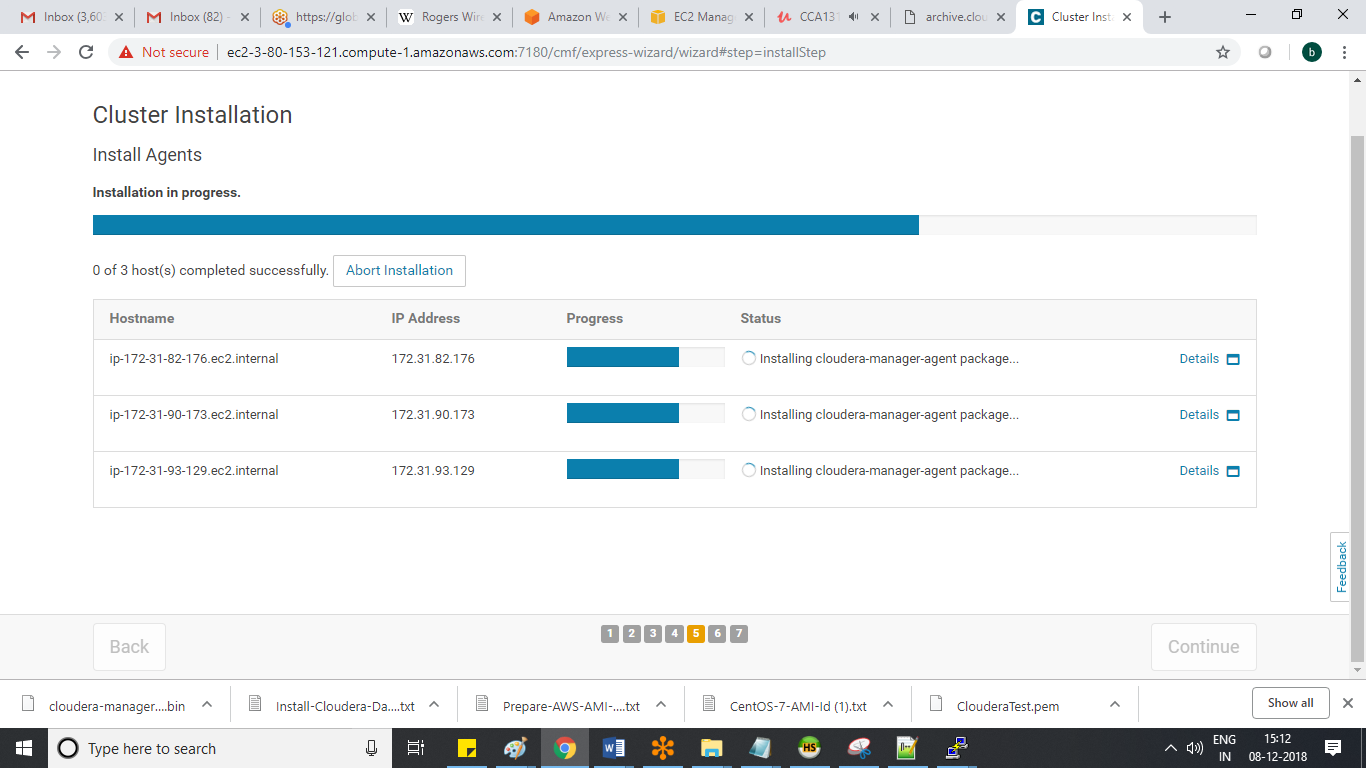


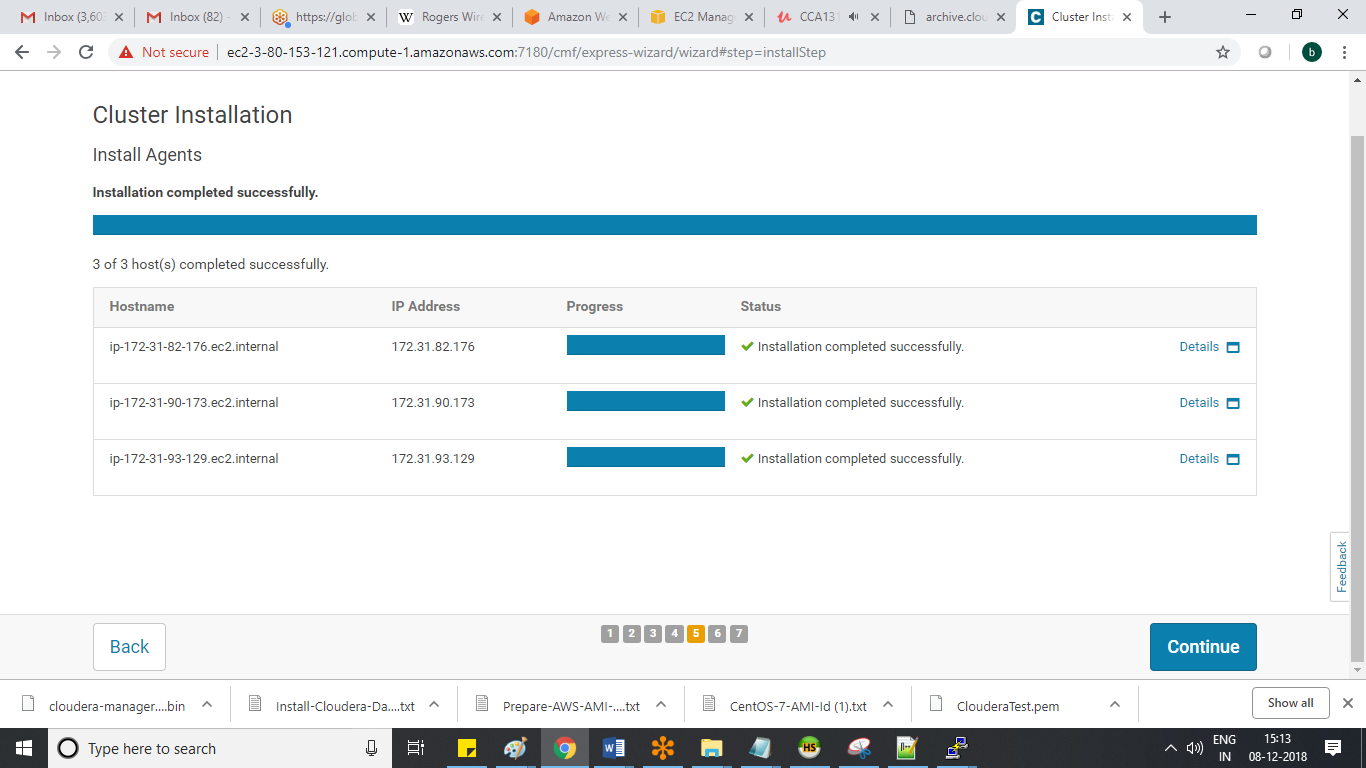
Select the respective .pem file











It will get above success message.

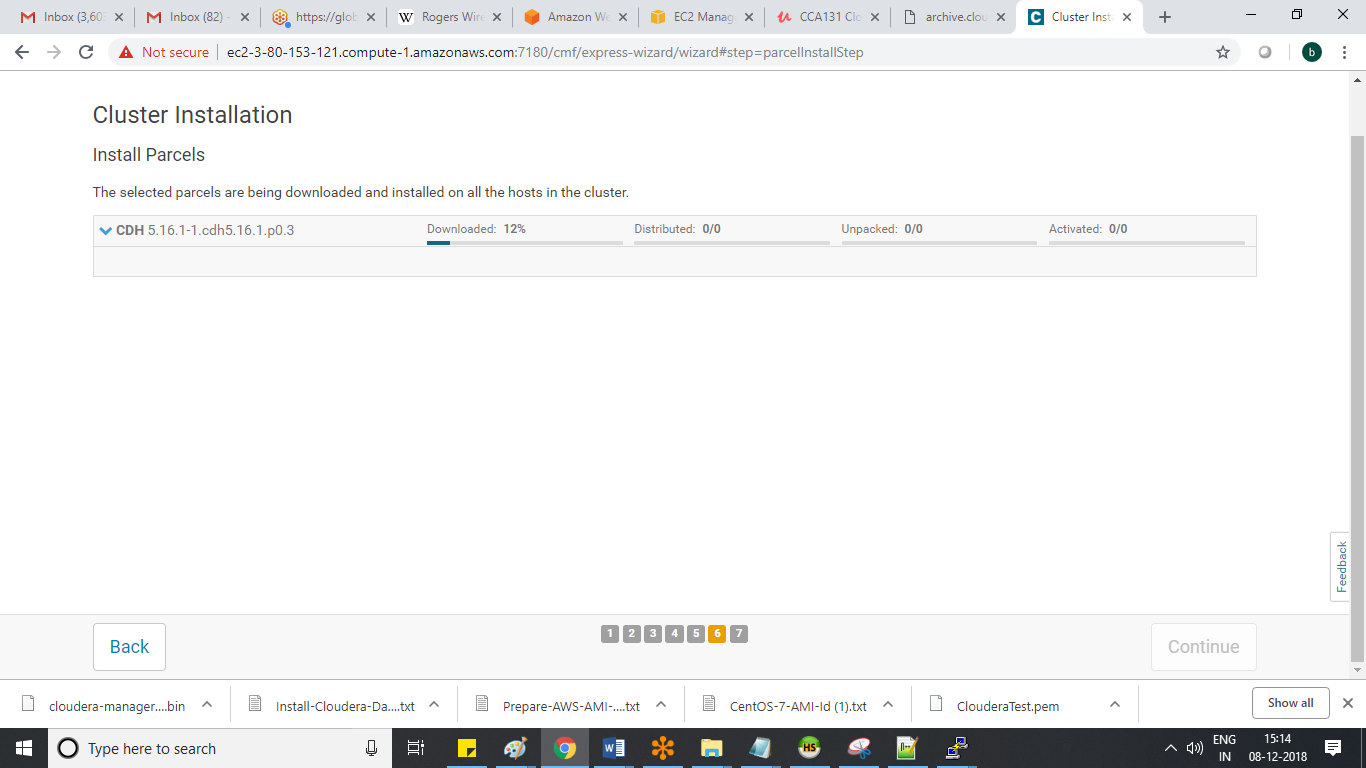
Now click continue

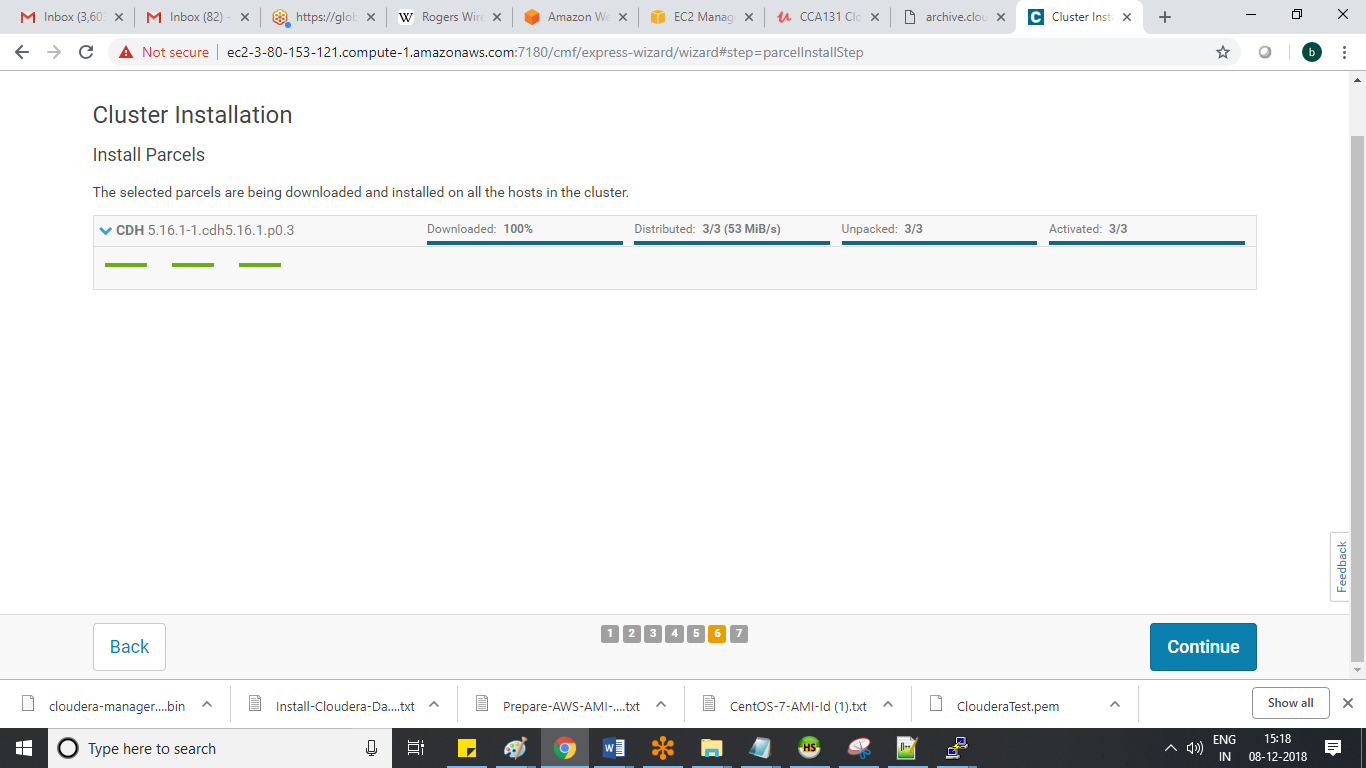
You can get below one.

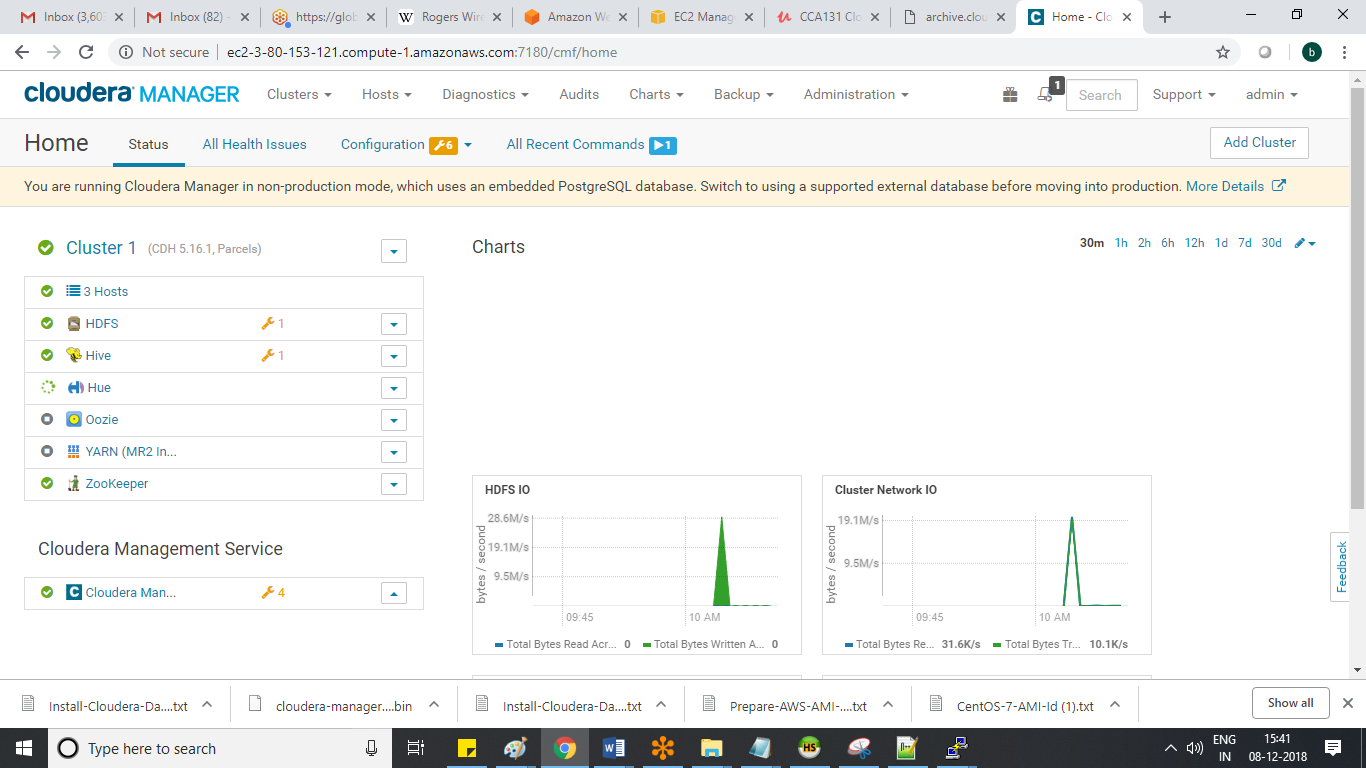
Download the parcels, distribute the parcels across all agents and unzip and it gets activated.

Backend we can see the progress like below

/var/run/cloudera-scm-agent/







**Cloudera repository set up with httpd:**

****

**Cloudera complete installation with local repository and also creating AMI:**

****

****

**Agent installation:**

****

**HDFS commands**

****

**HDFS trash**

****

**HDFS HA.**

**Disable NAMENODE HA**

**Balancer:**

**HDFS maintenance mode : we can do it for cluster, service and individual role**

**HDFS Quota:**

****

**RACK awareness**

****

**HDFS metadata:**

****

**Rolledits**

****

**HDFS savenamespace**

****

**Snapshots:**

****

**Webhdfs:**

****

**Httpfs:**

****

**HDFS FSCK**

****

**HDFS recovery**

****

**HDFS federation and high availability**

****

**HDFS home directory**

****

**OPENLDAP and PHPLDAPDMIN**

****