**HDP (2.6.2) setup using Ambari and Online repository**

Setup atleast 3 linux machines (considering ubuntu 16.04 machines)

-Machines should have different IP Addresses

-They should be able to ping each other, ssh each other for a dedicated user and for root, ping external network

Install the following for easy working

apt-get install wget

apt-get install vim

apt-get install ntp

(service ntpd start)

apt-get install openssh-server

apt-get update

-Disable firewall

-Correctly update /etc/hosts

Example:

Ipaddress-c1 ipadress.example.com c1

Ipaddress-c2 ipadress.example.com c2

Ipaddress-c3 ipadress.example.com c3

-Correctly update /etc/hostname to reflect hostname as mentioned above

-Machines should have minimum 5.5gb ram + 30 gb Hard disk + 2 cpu cores

(if possible assign more resources)

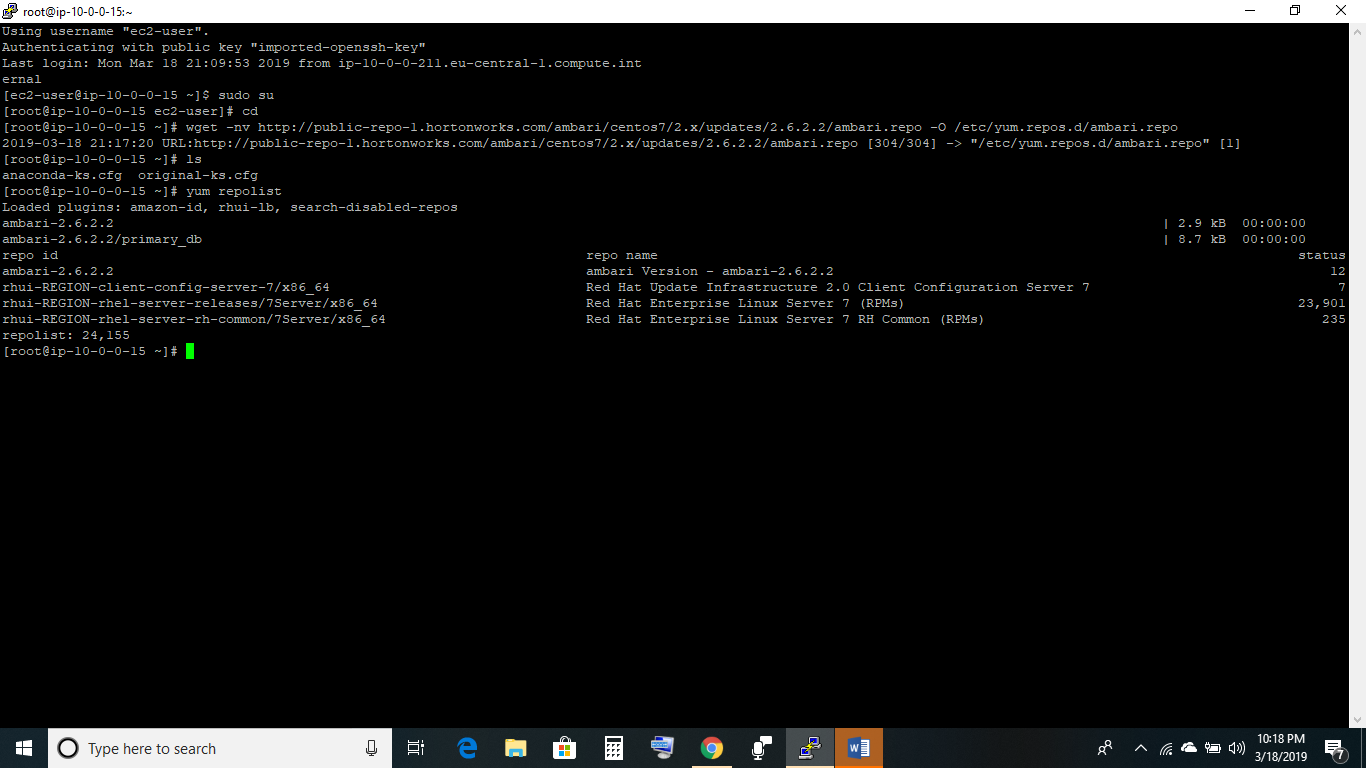
Check if any ports are blocked and open them.

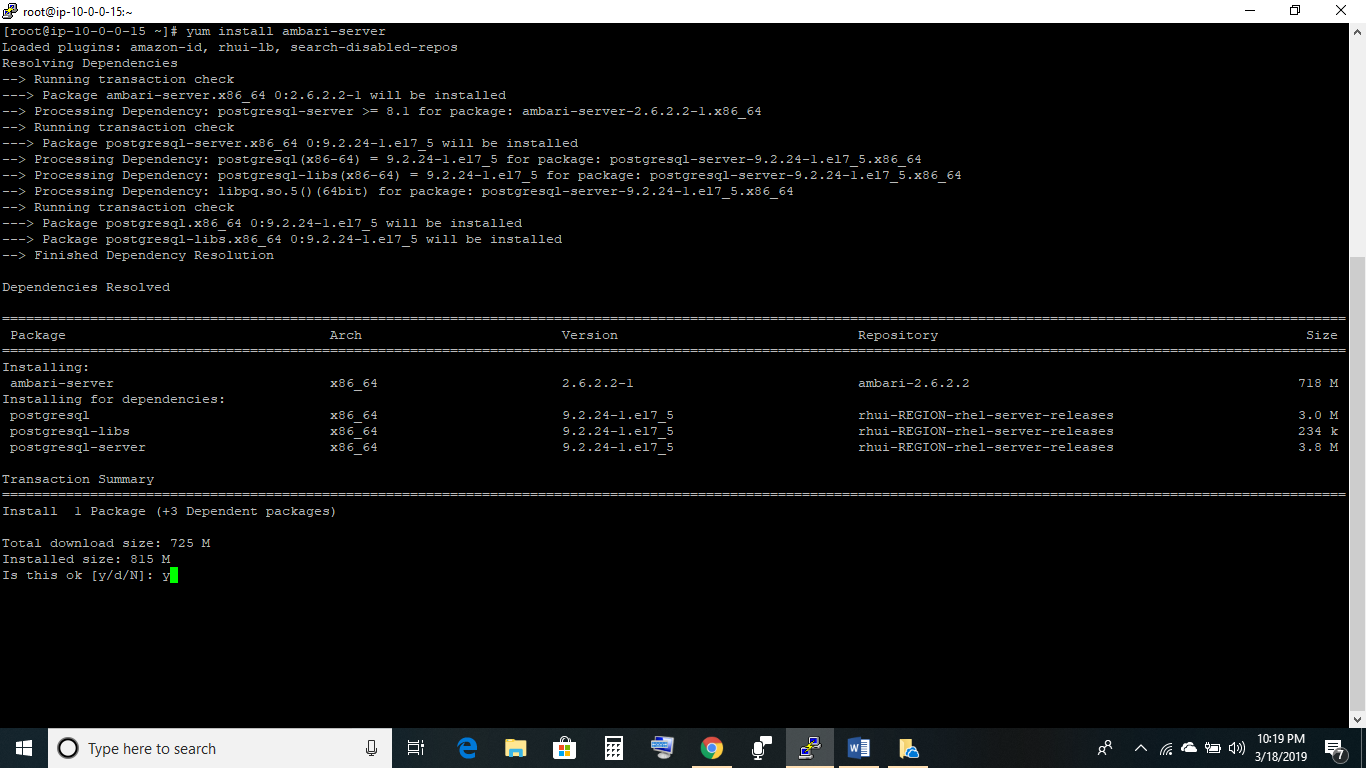
Check if all the above is done correctly.

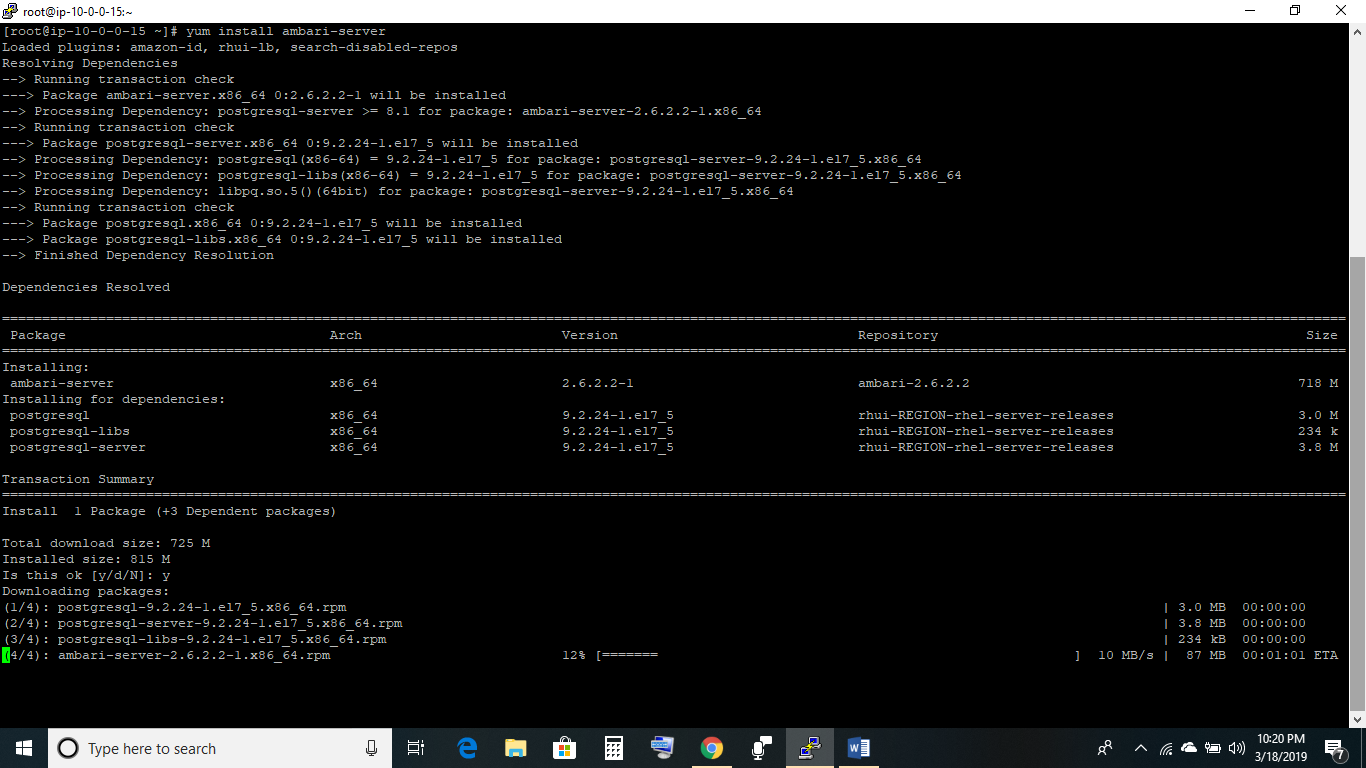
Note\*\* If you have limited RAM machines, you can even setup 2 machines for cluster setup or use AWS/Azure/Google cloud instances with higher configurations.

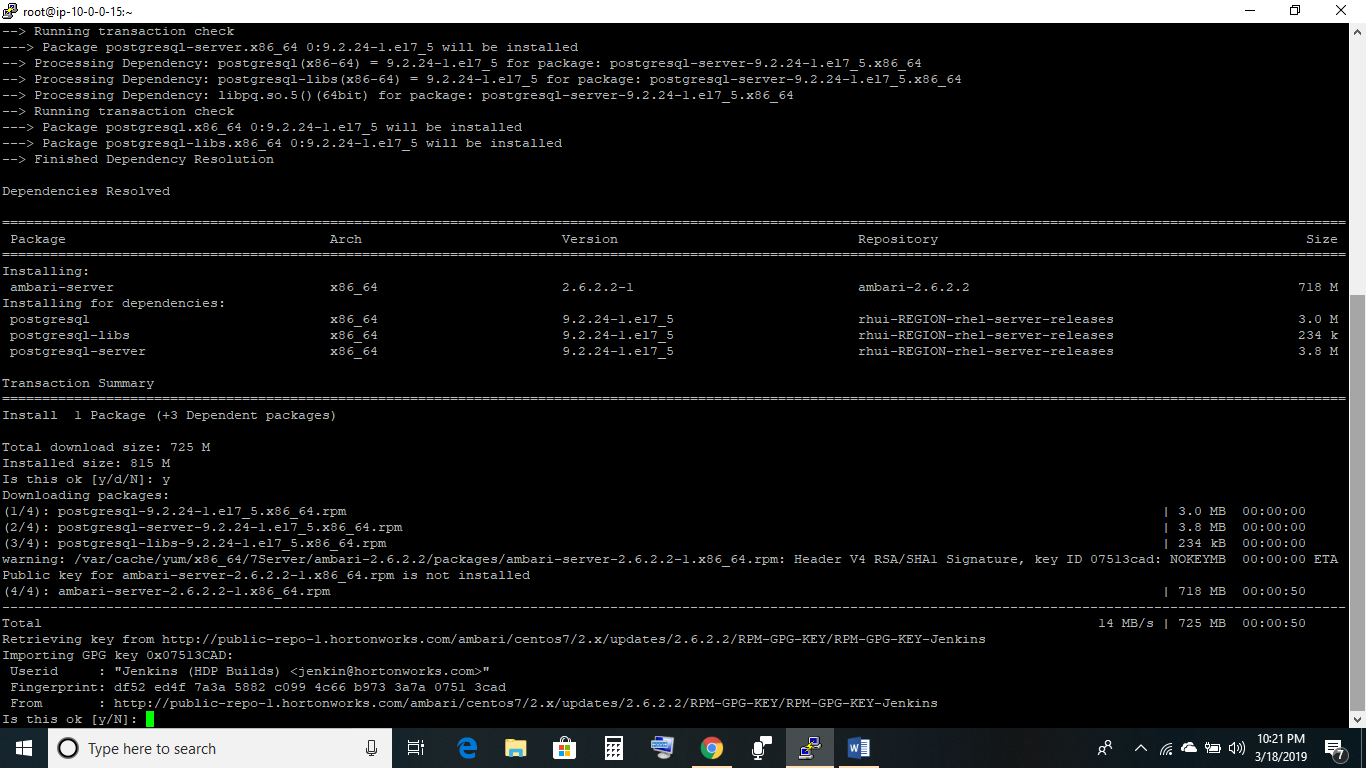
Note\* If you intend to use Mysql as backend database, scroll to end of document to understand the steps that will be required.

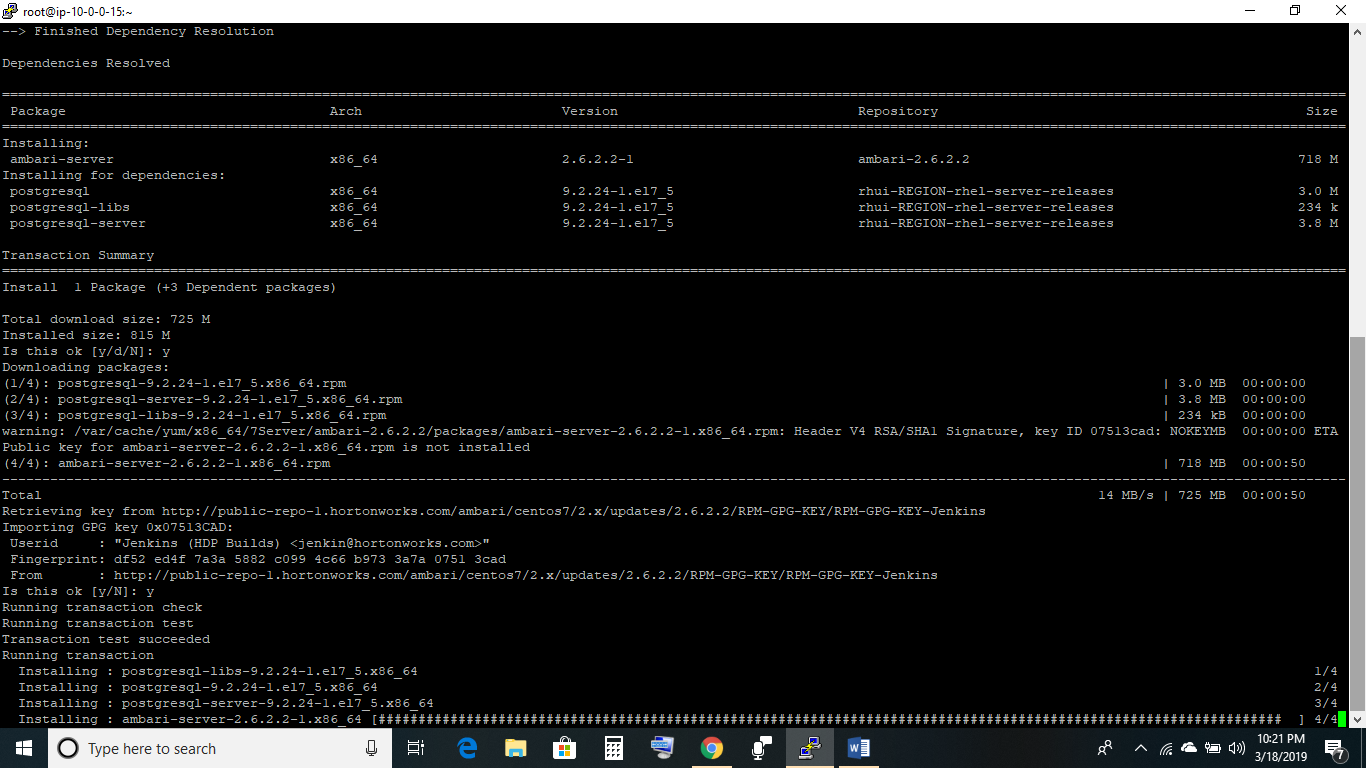
Here I am using EC2 instances (redhat 7.5) from AWS and have done the above accordingly for the instances to be ready for HDP setup using ambari.



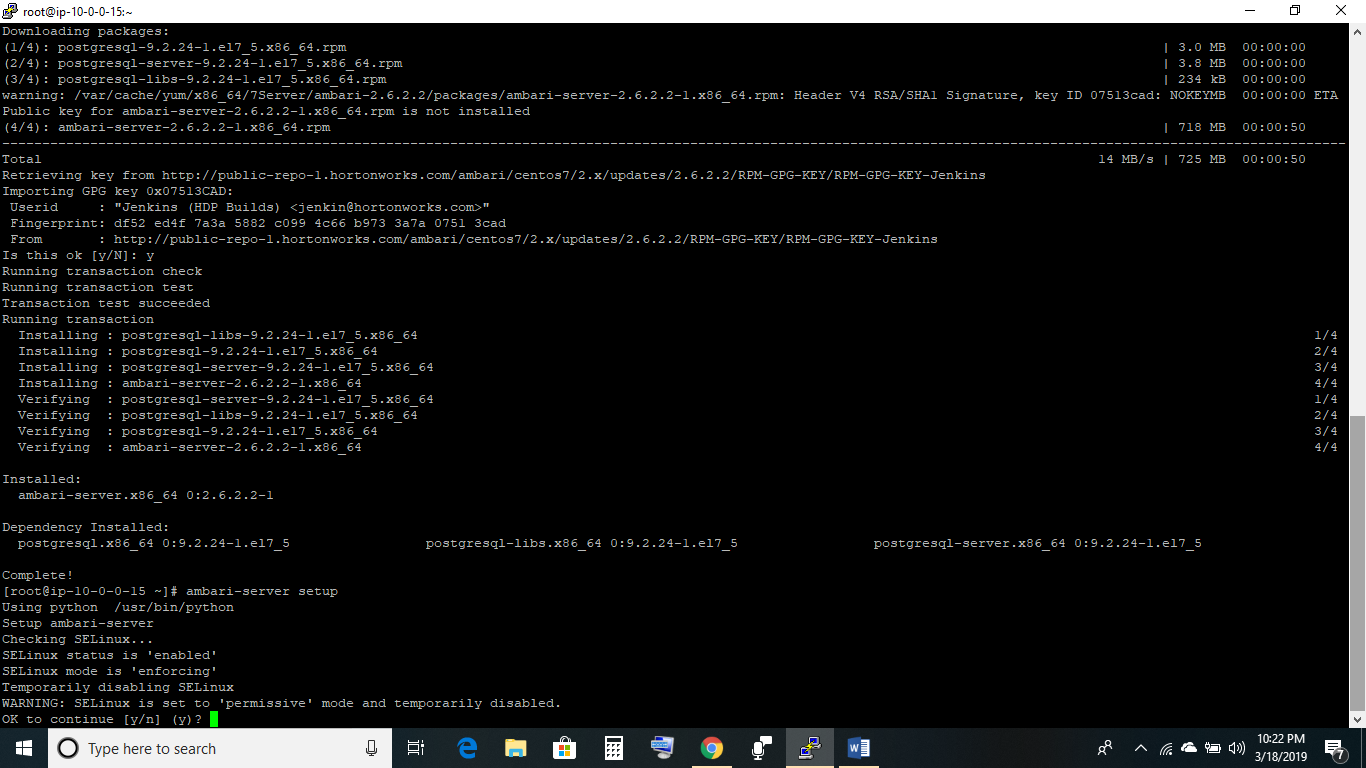


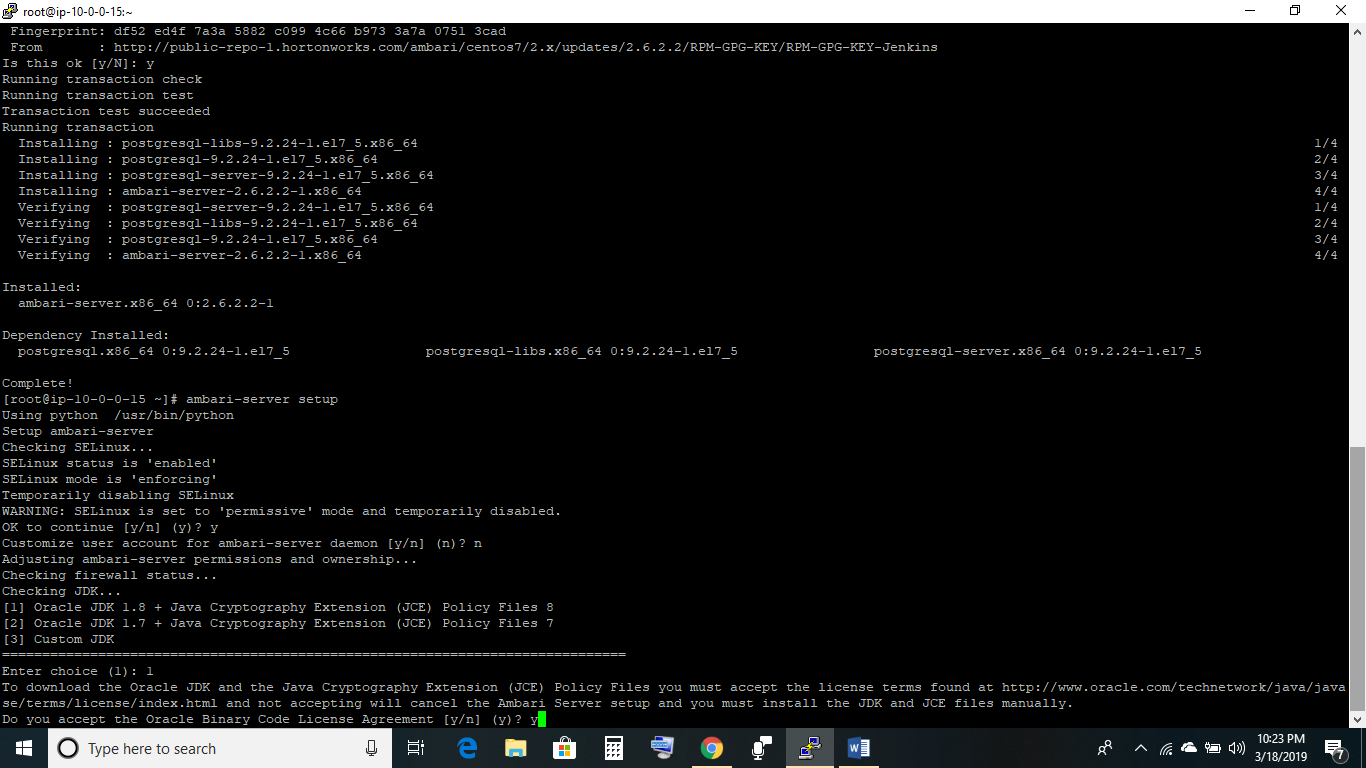




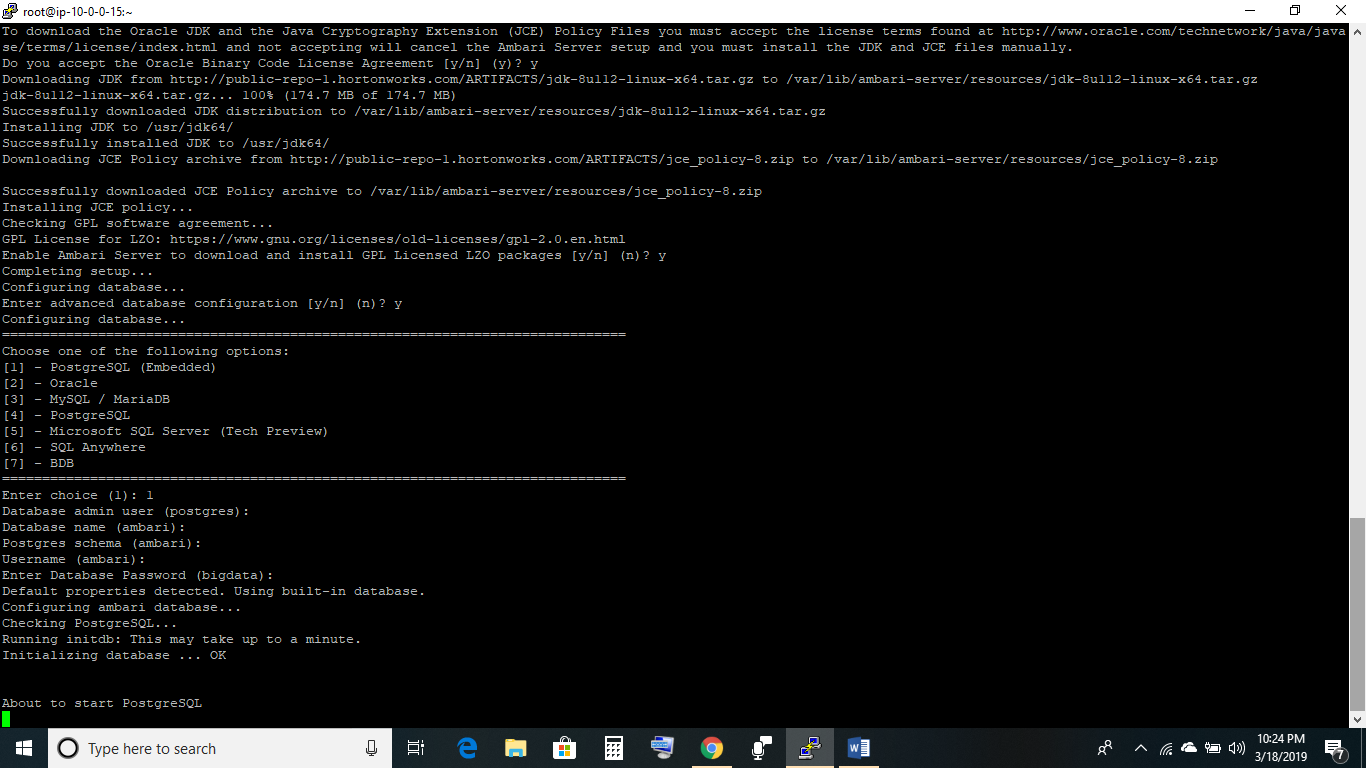


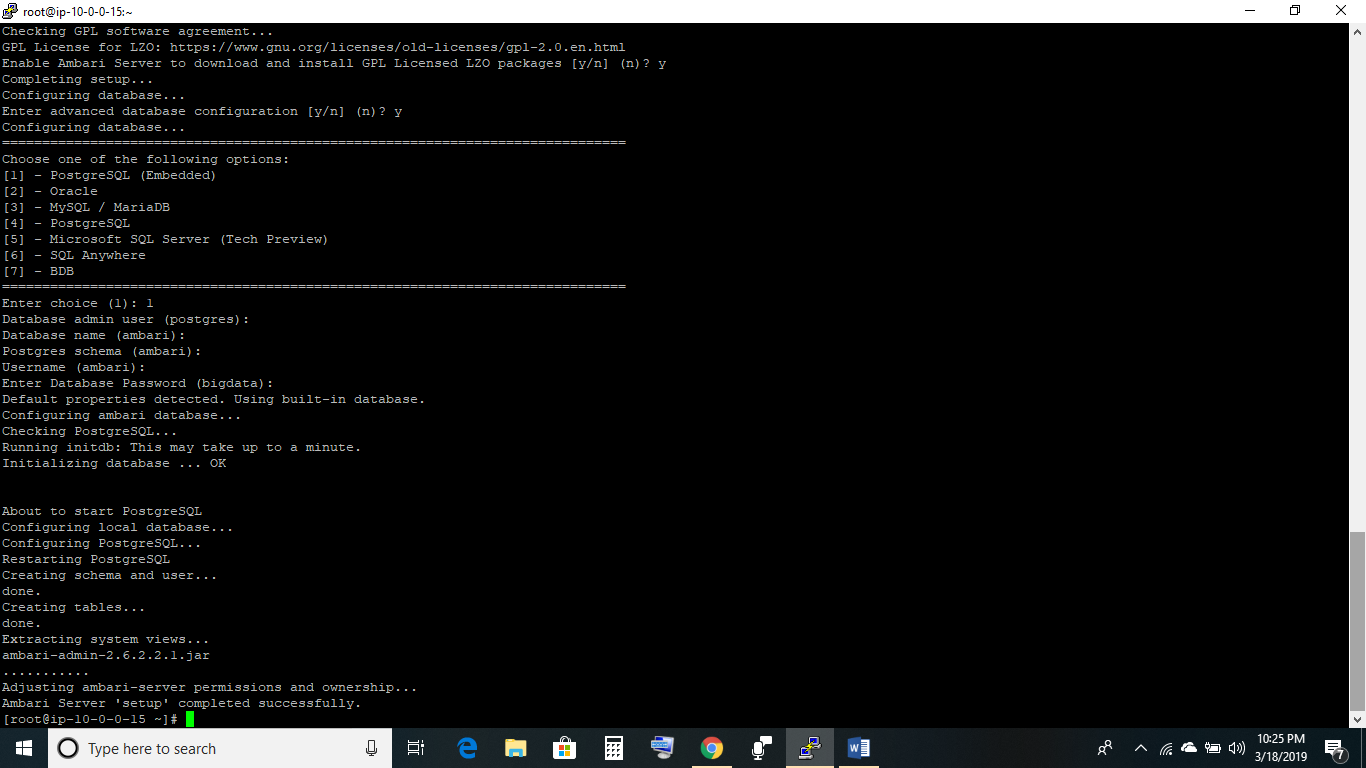
3.

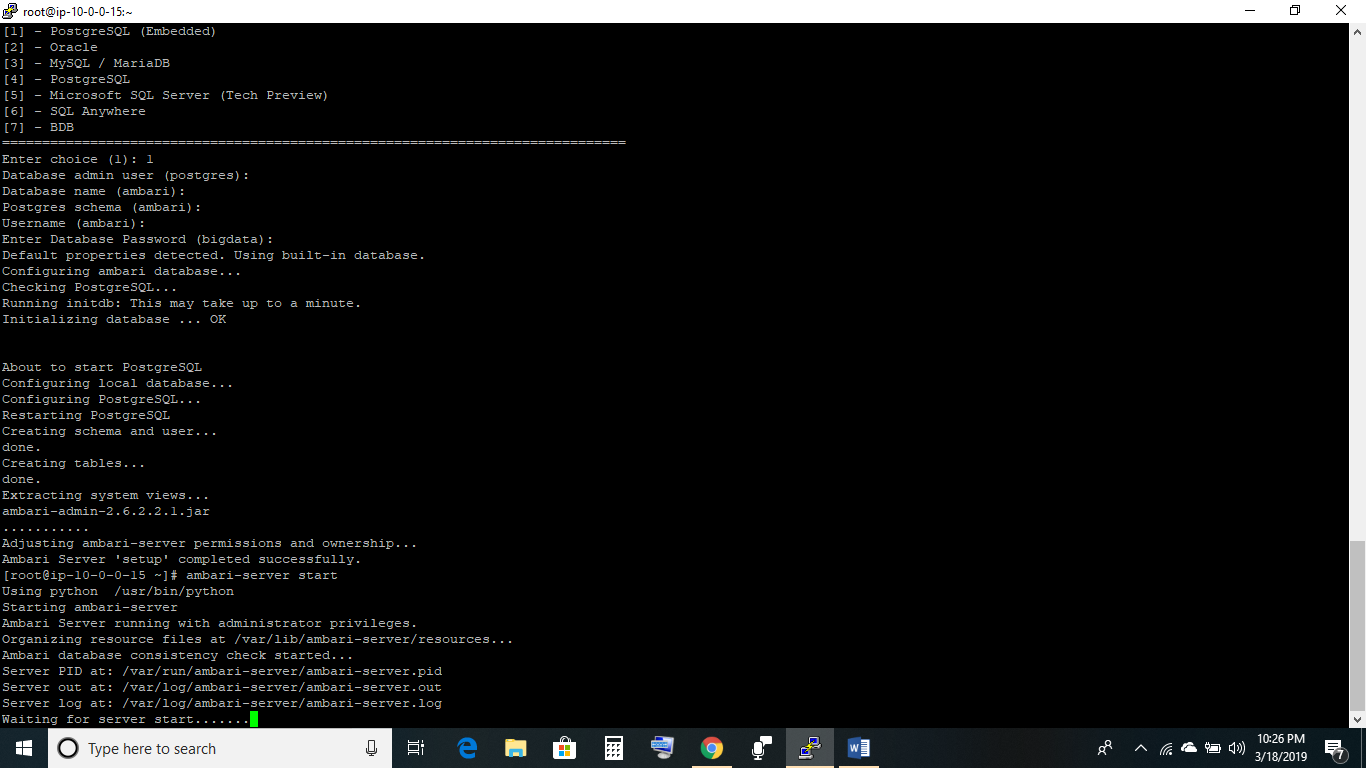


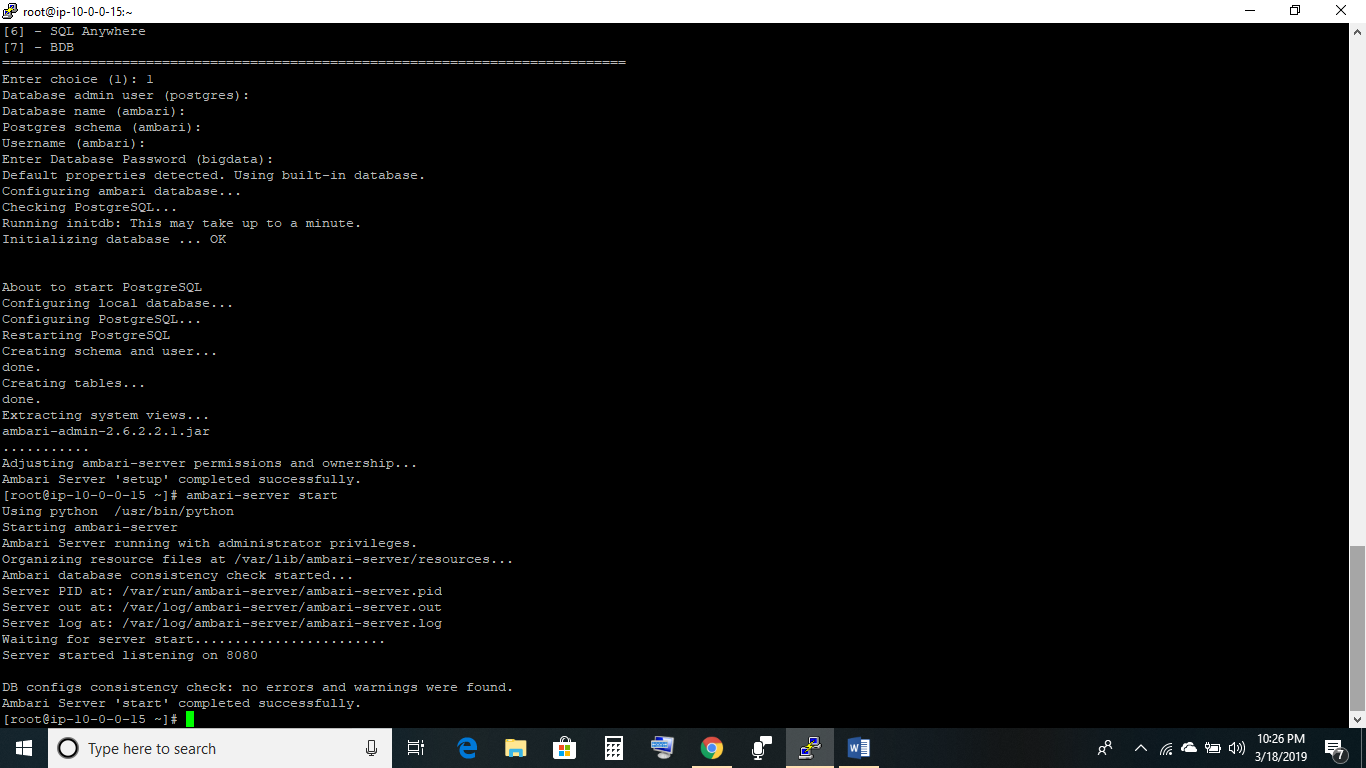








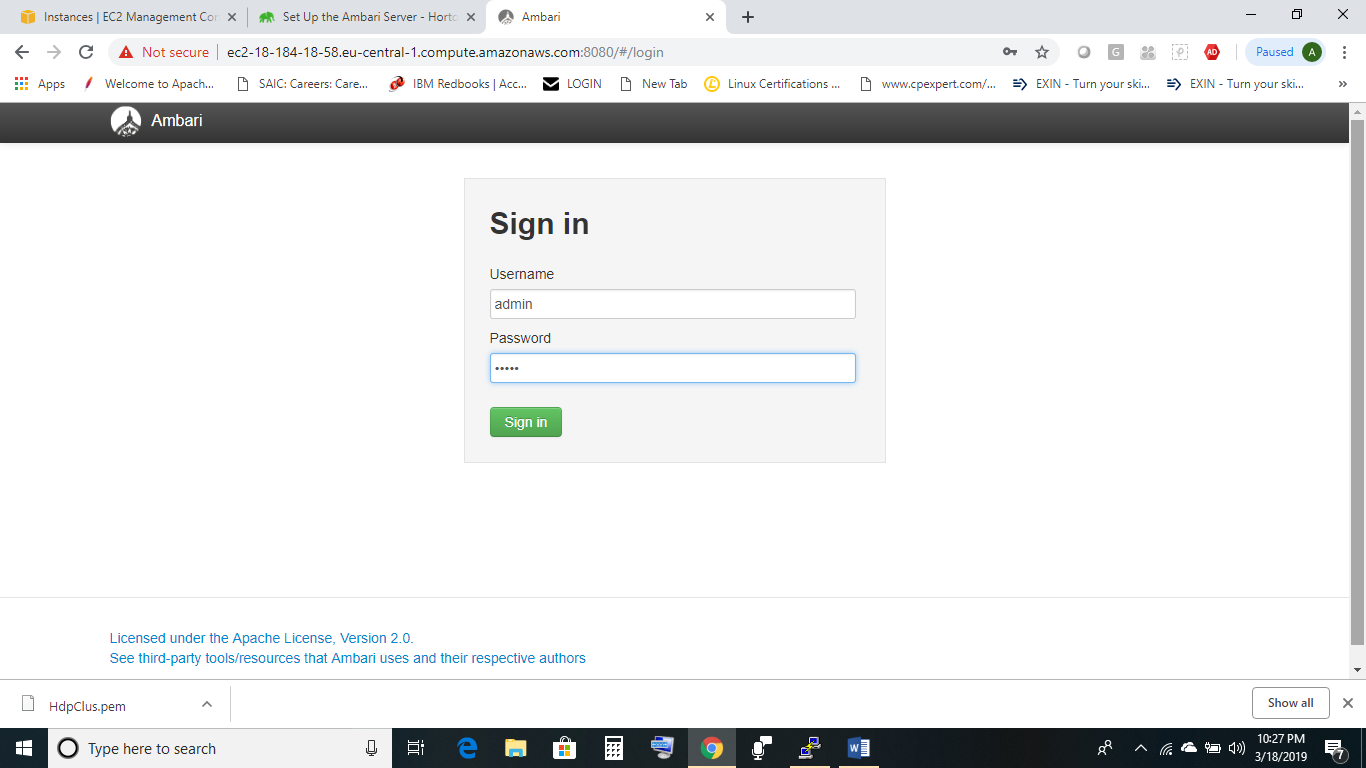


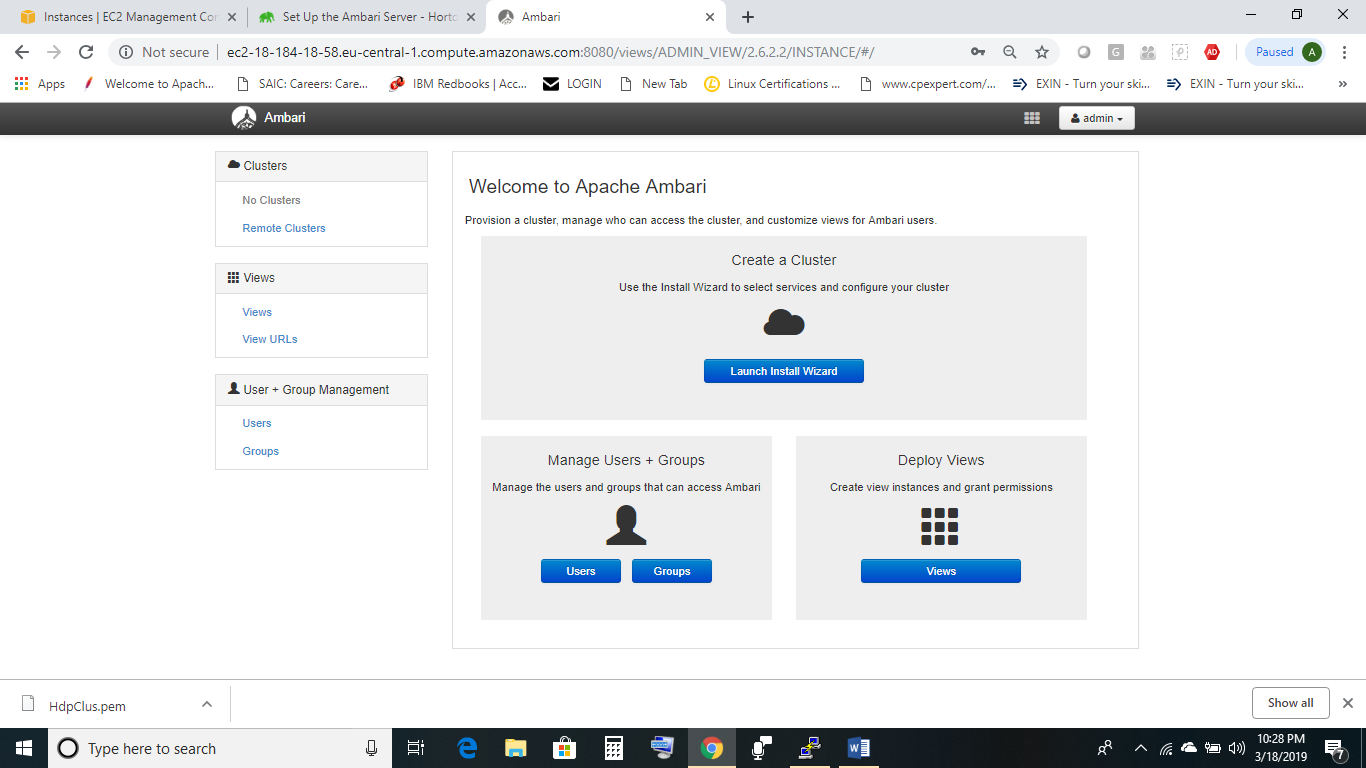


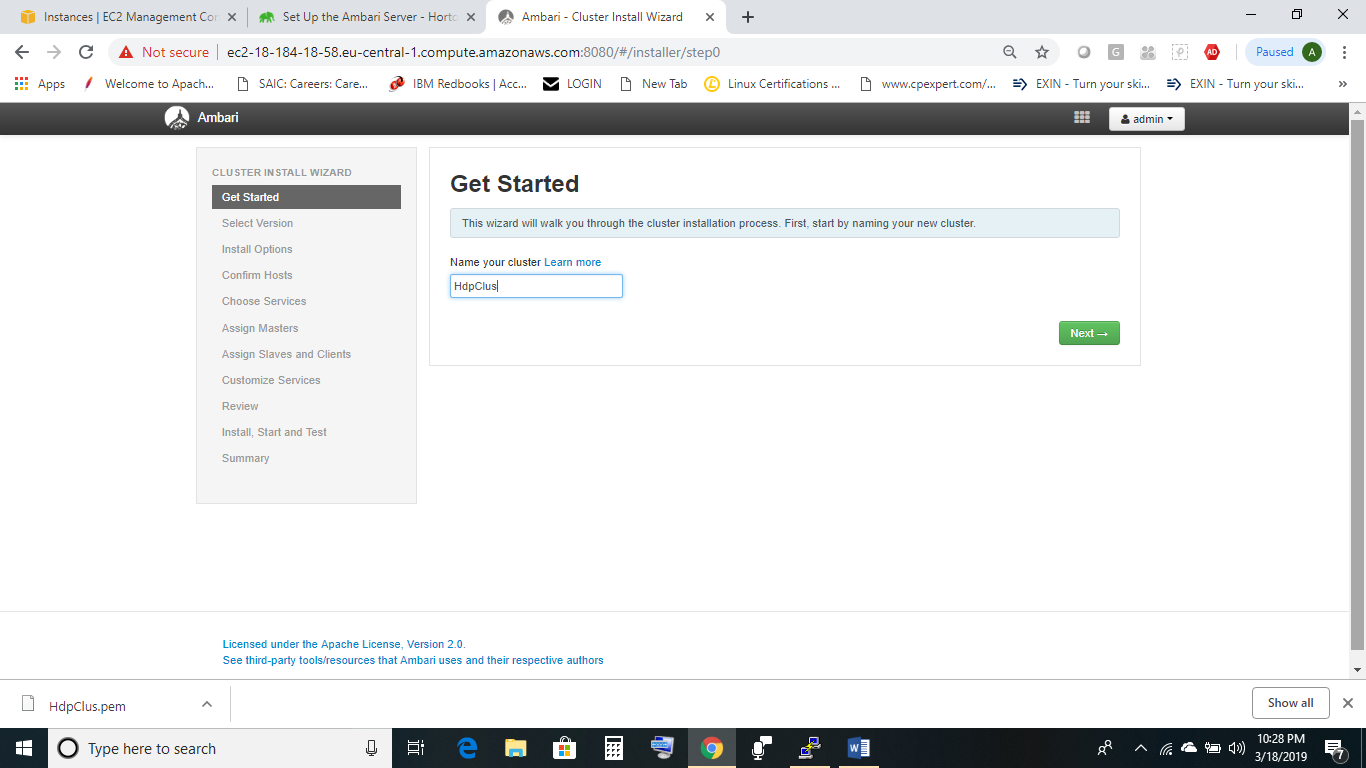
Start ambari-server

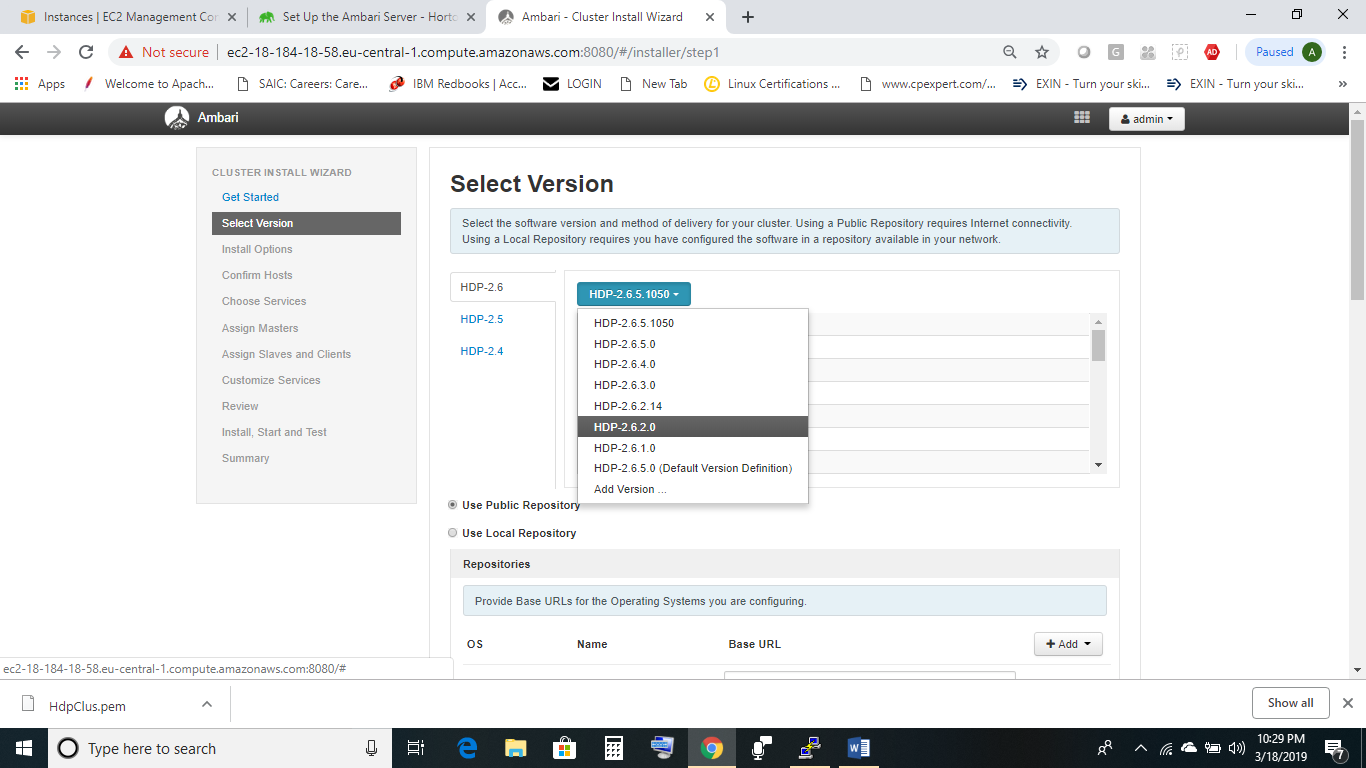
Access <http://hostname:8080>

Login : admin Password : admin

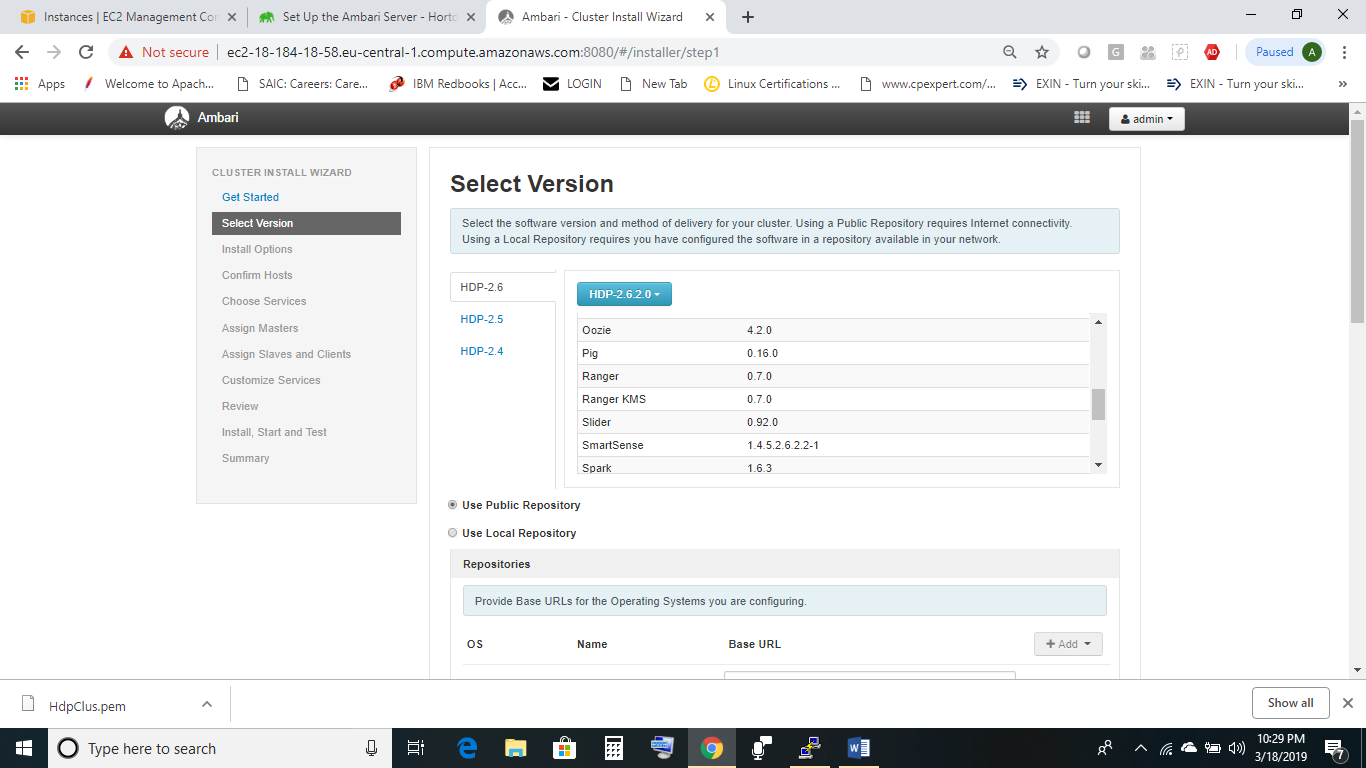


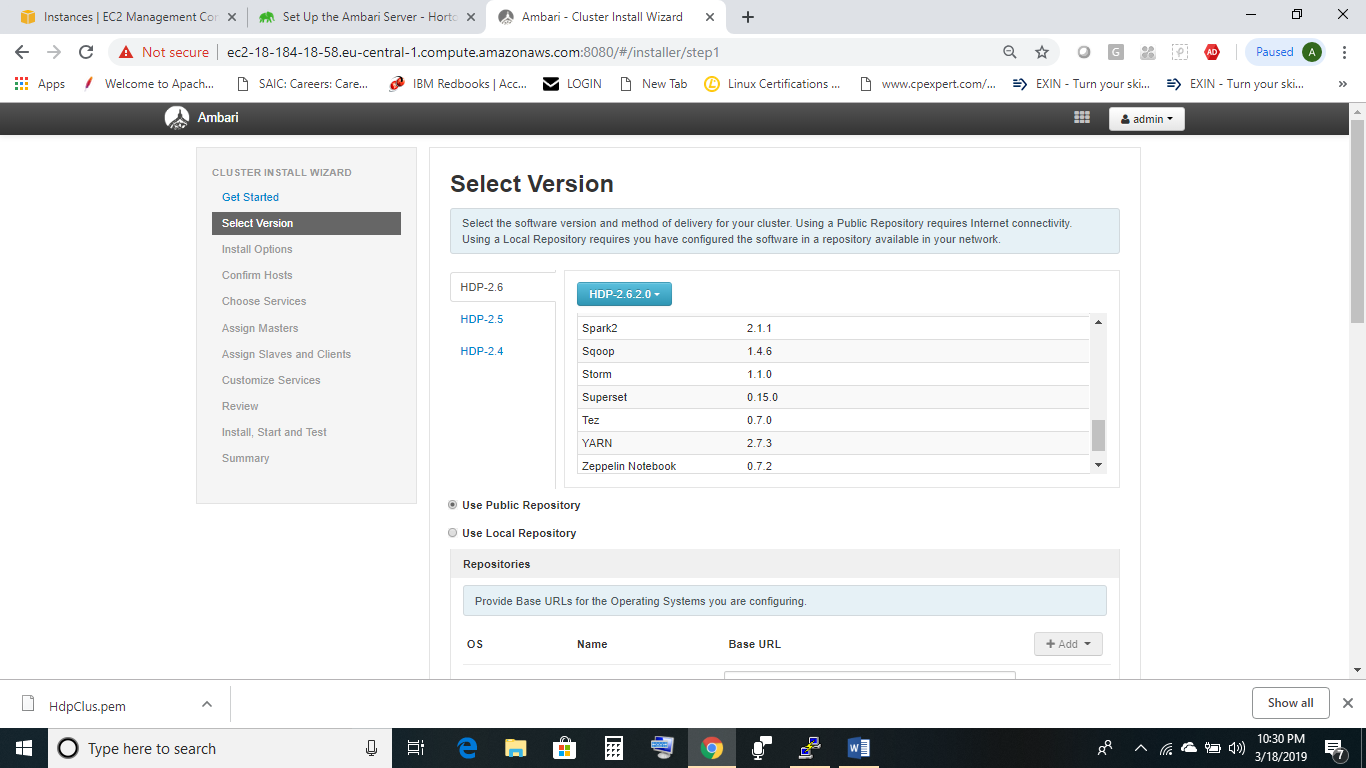


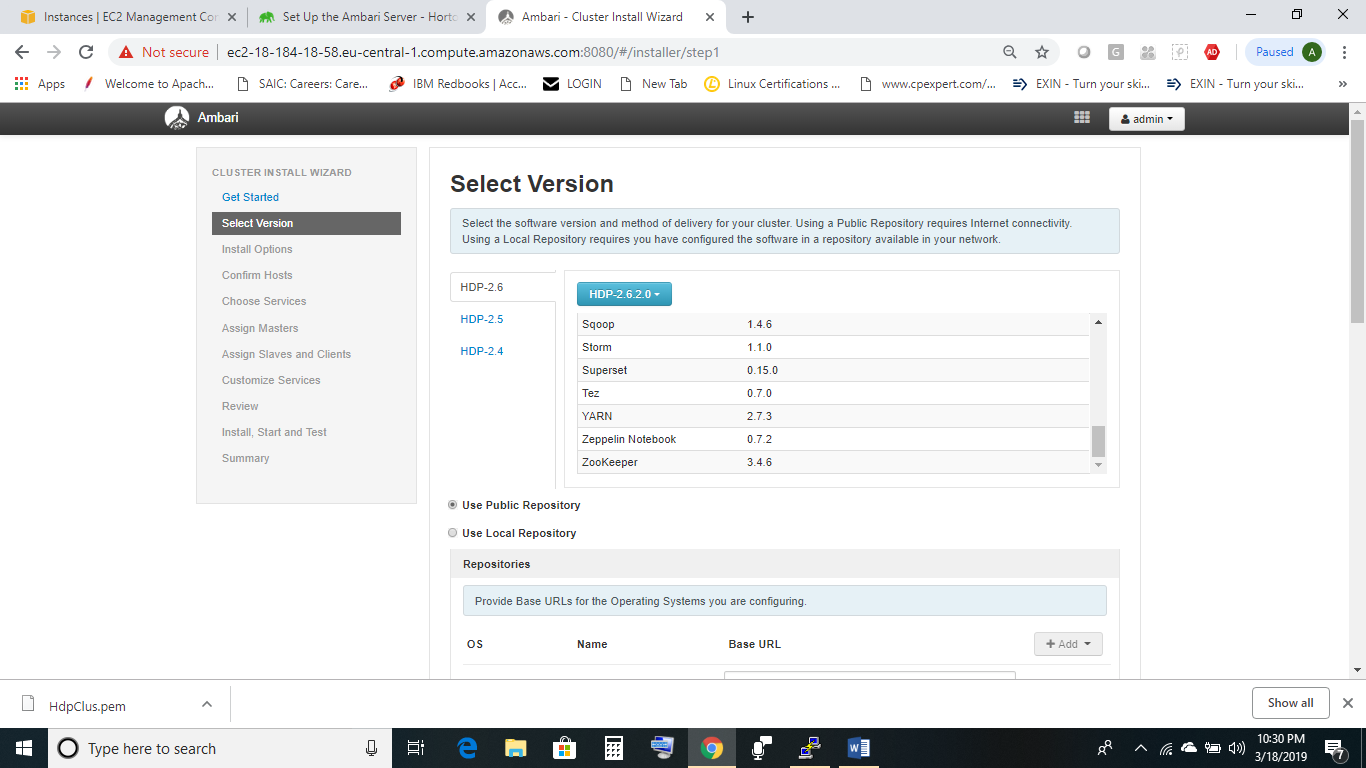


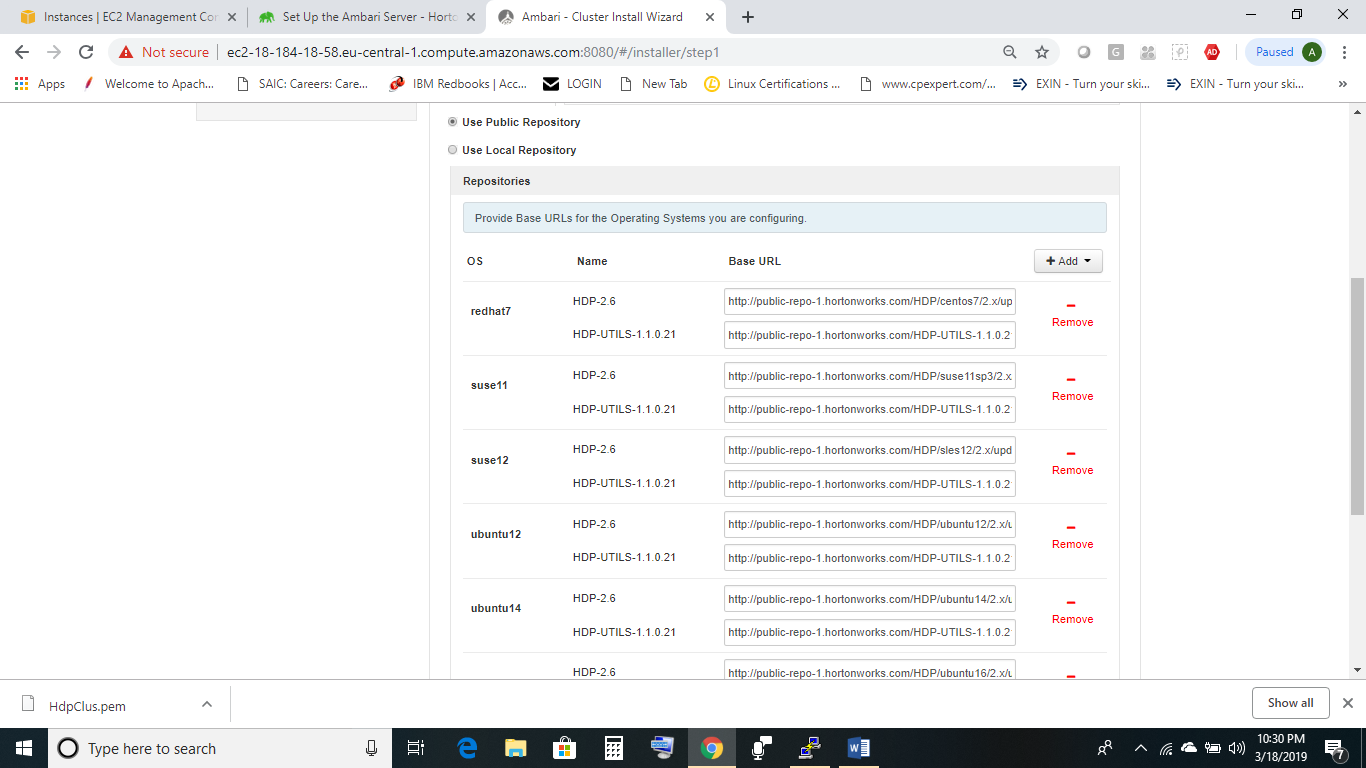


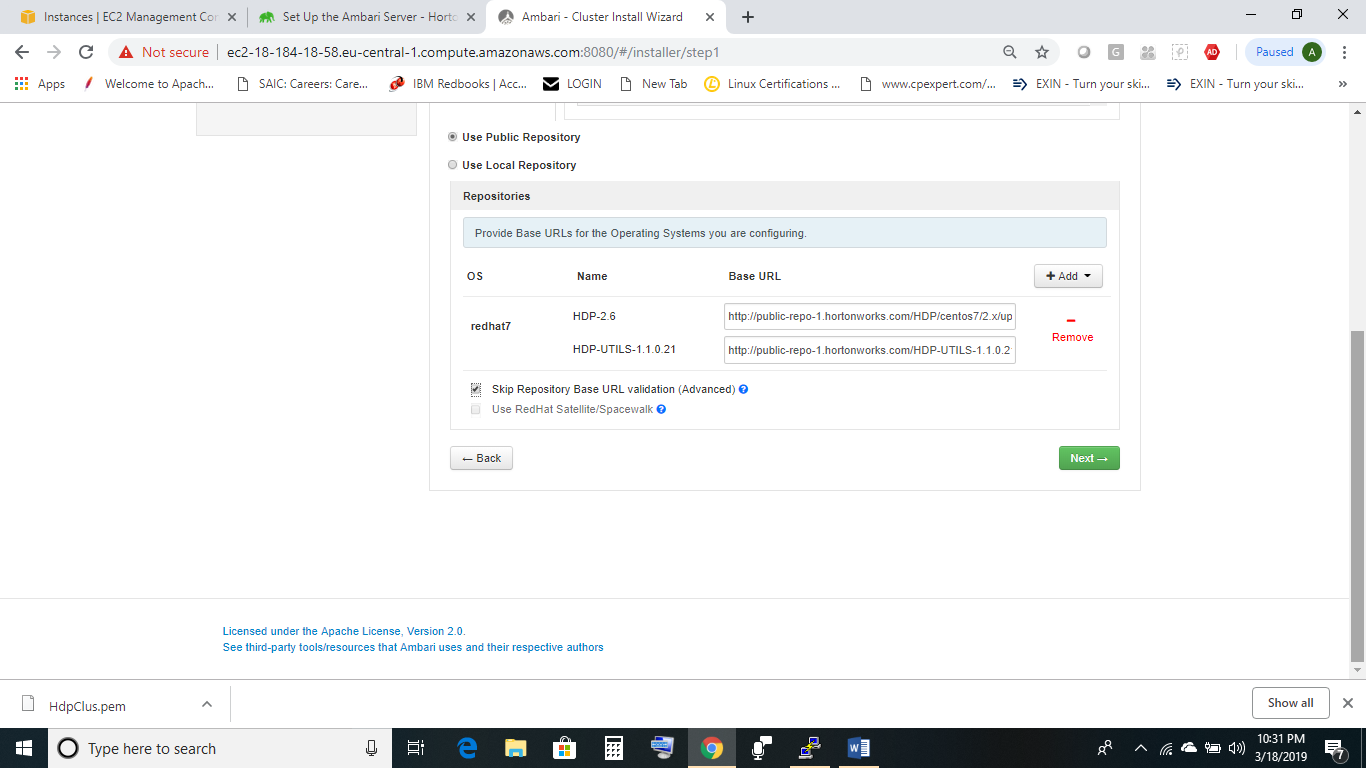


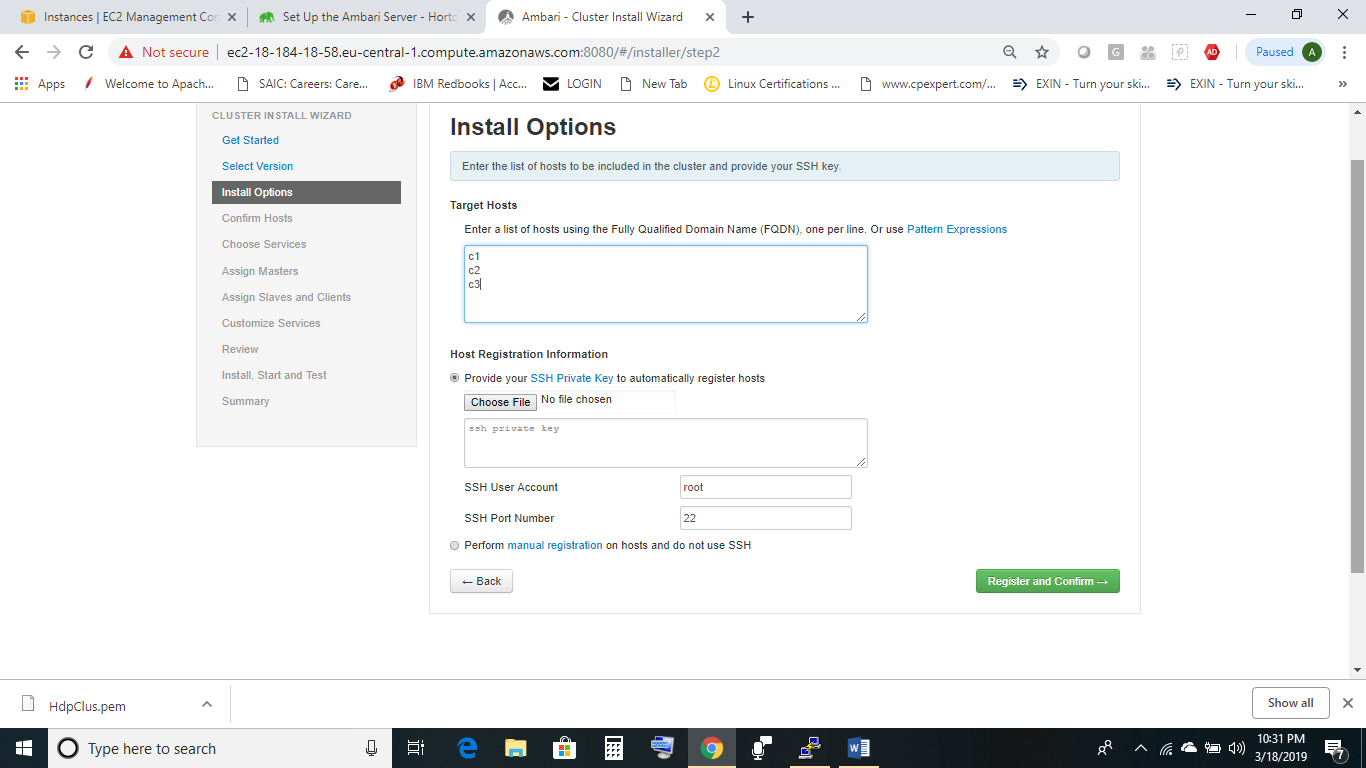




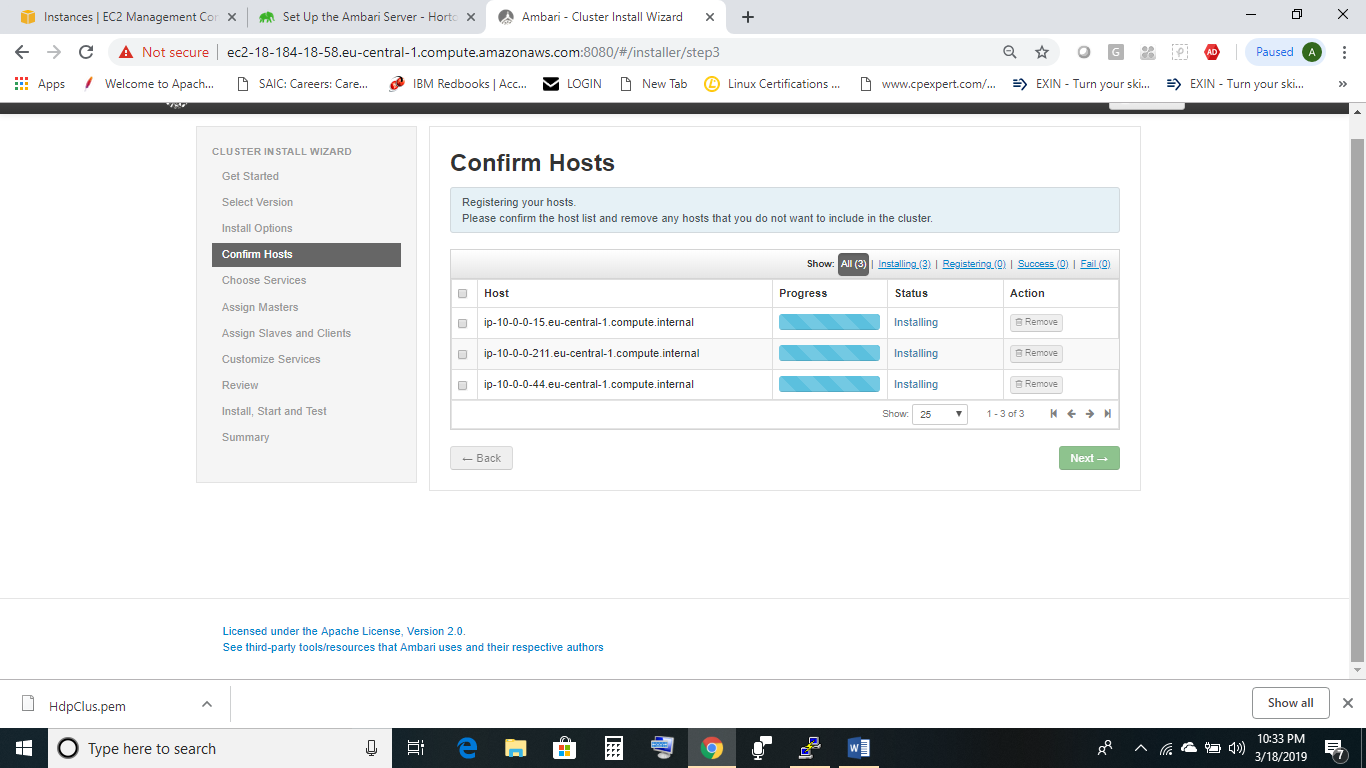


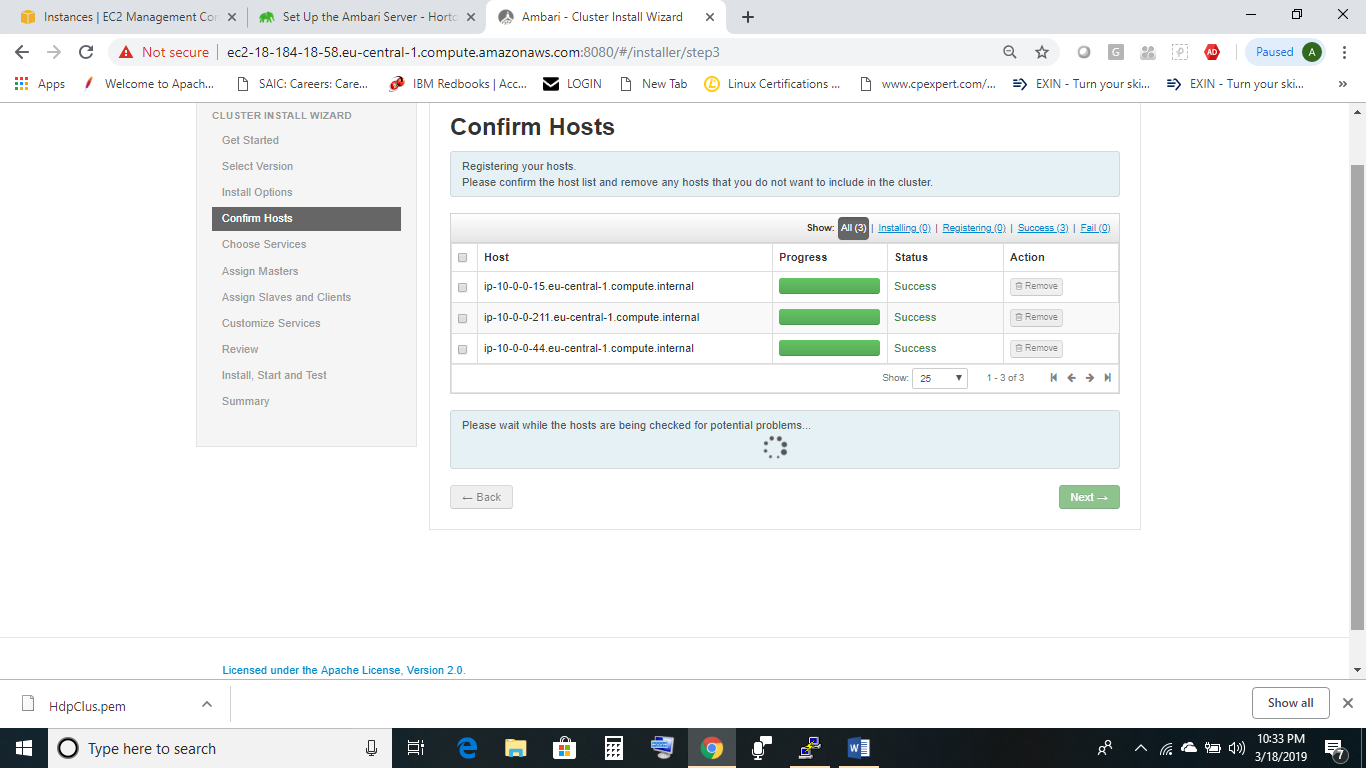


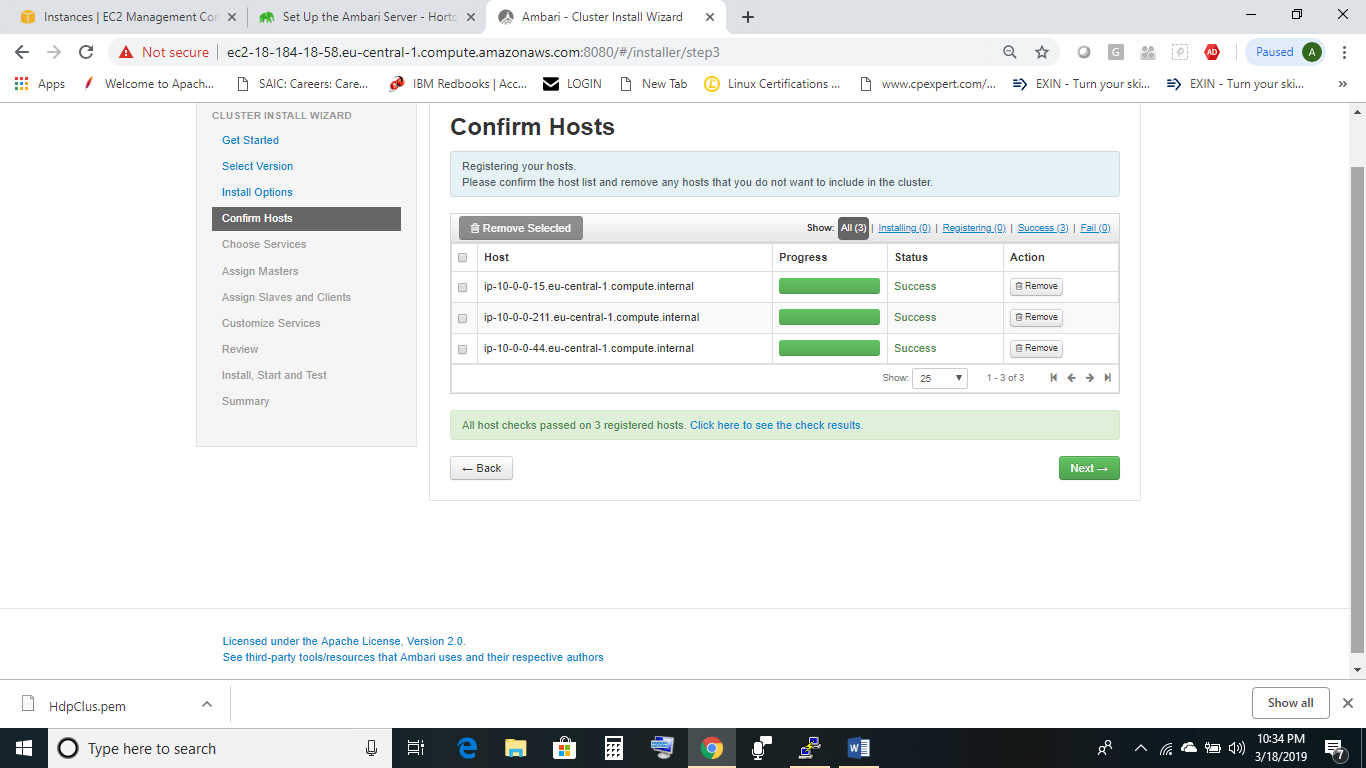






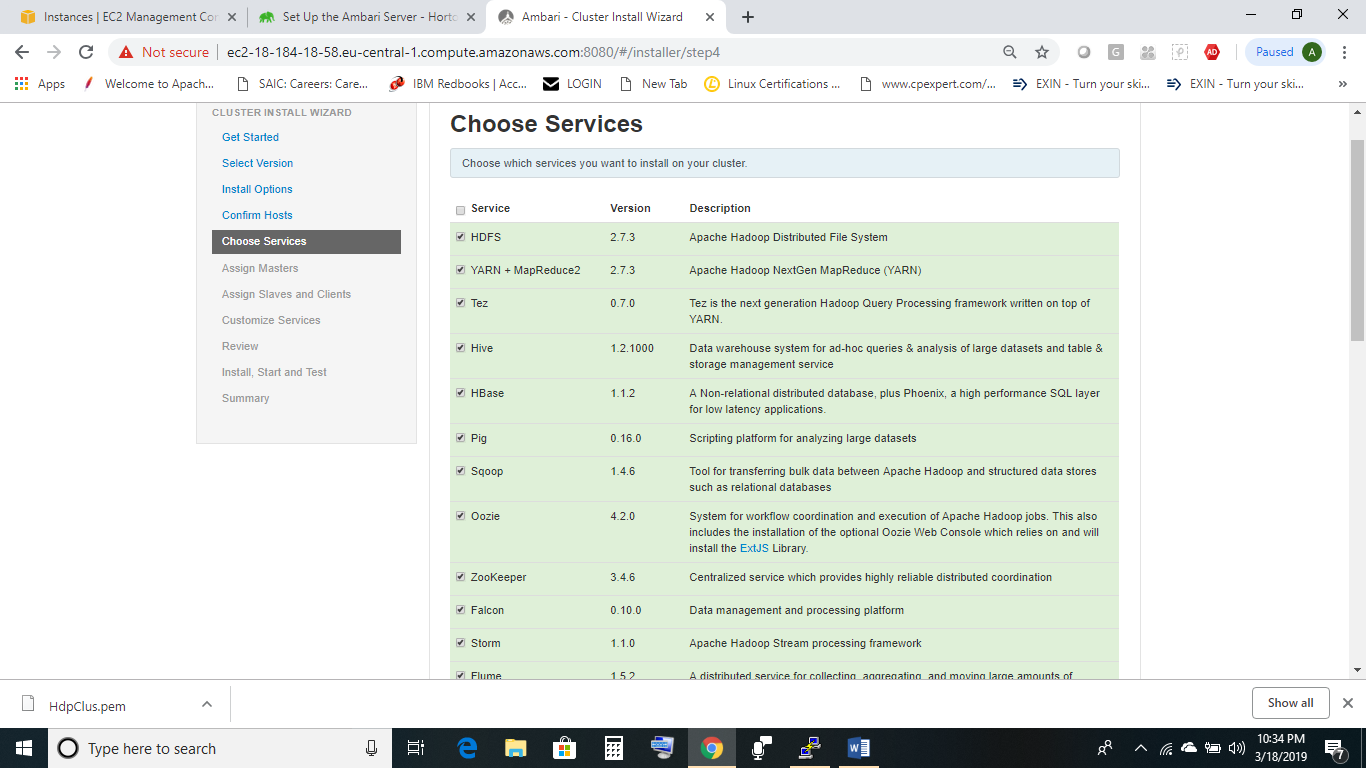


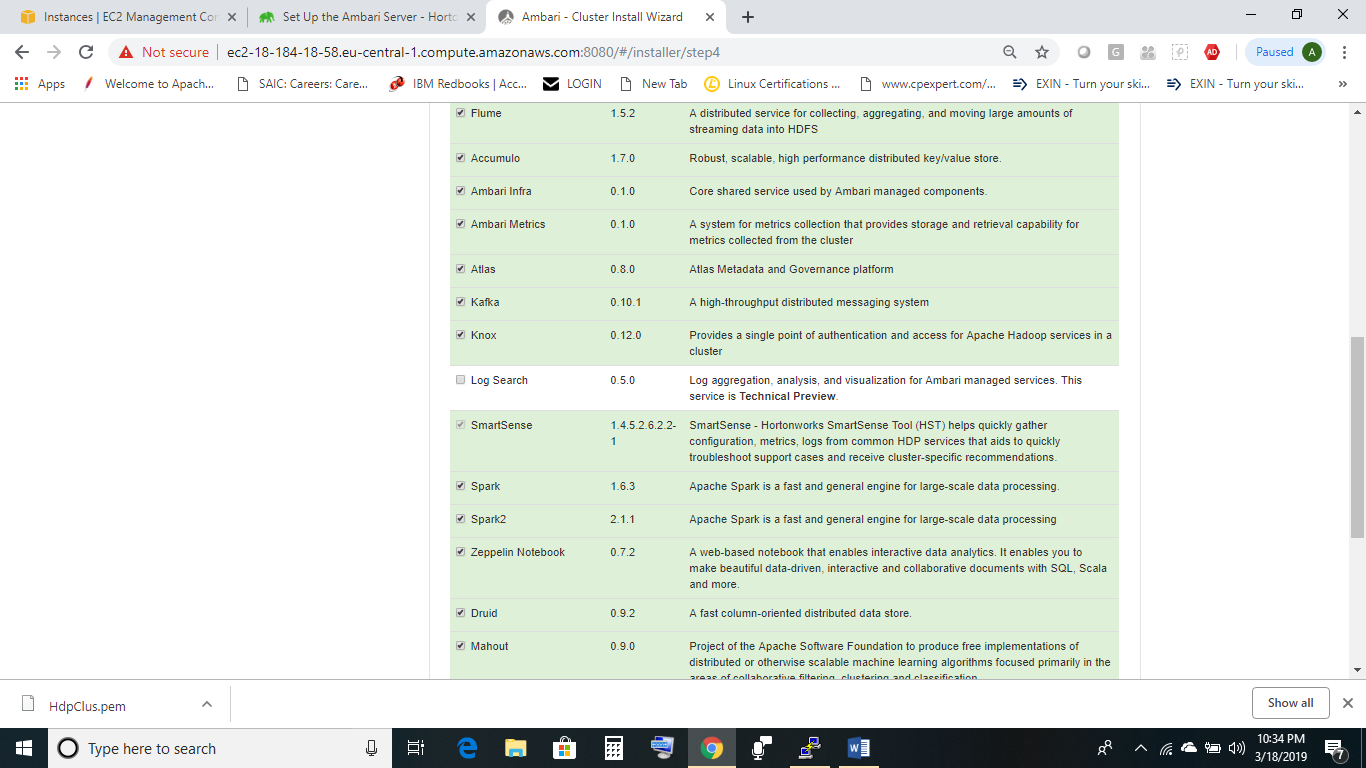


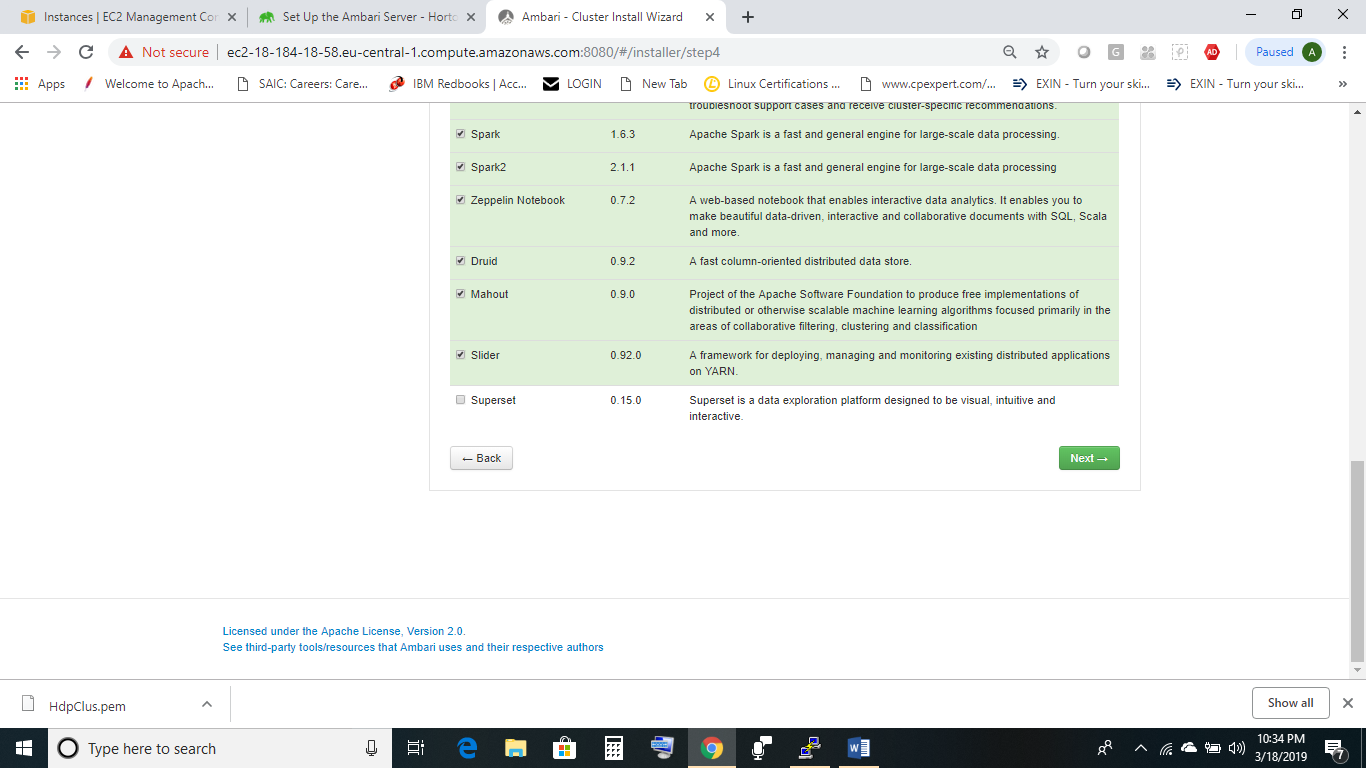


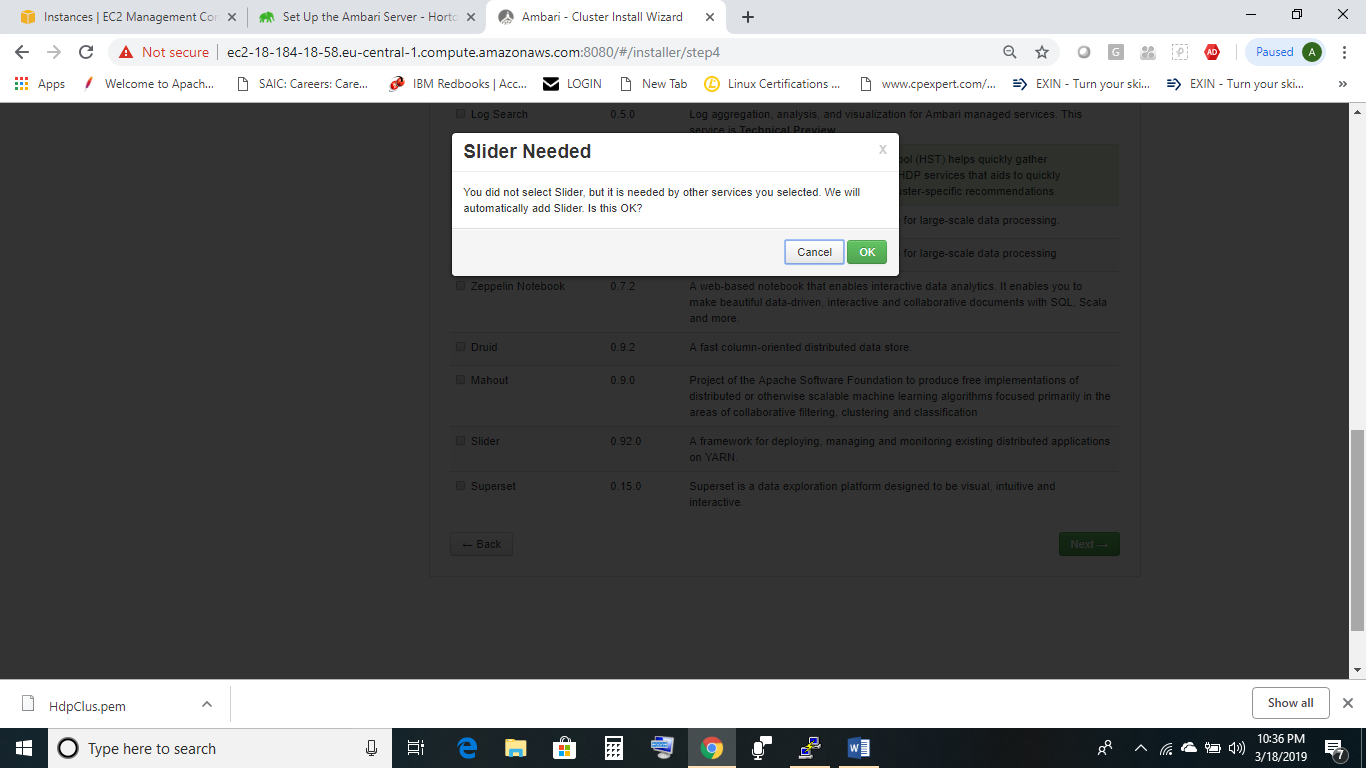
From all services listed, do not choose all services and select services which you would want to use. If services depend on others, you would be prompted for selecting the dependency services.

Initially go for bare minimum services.

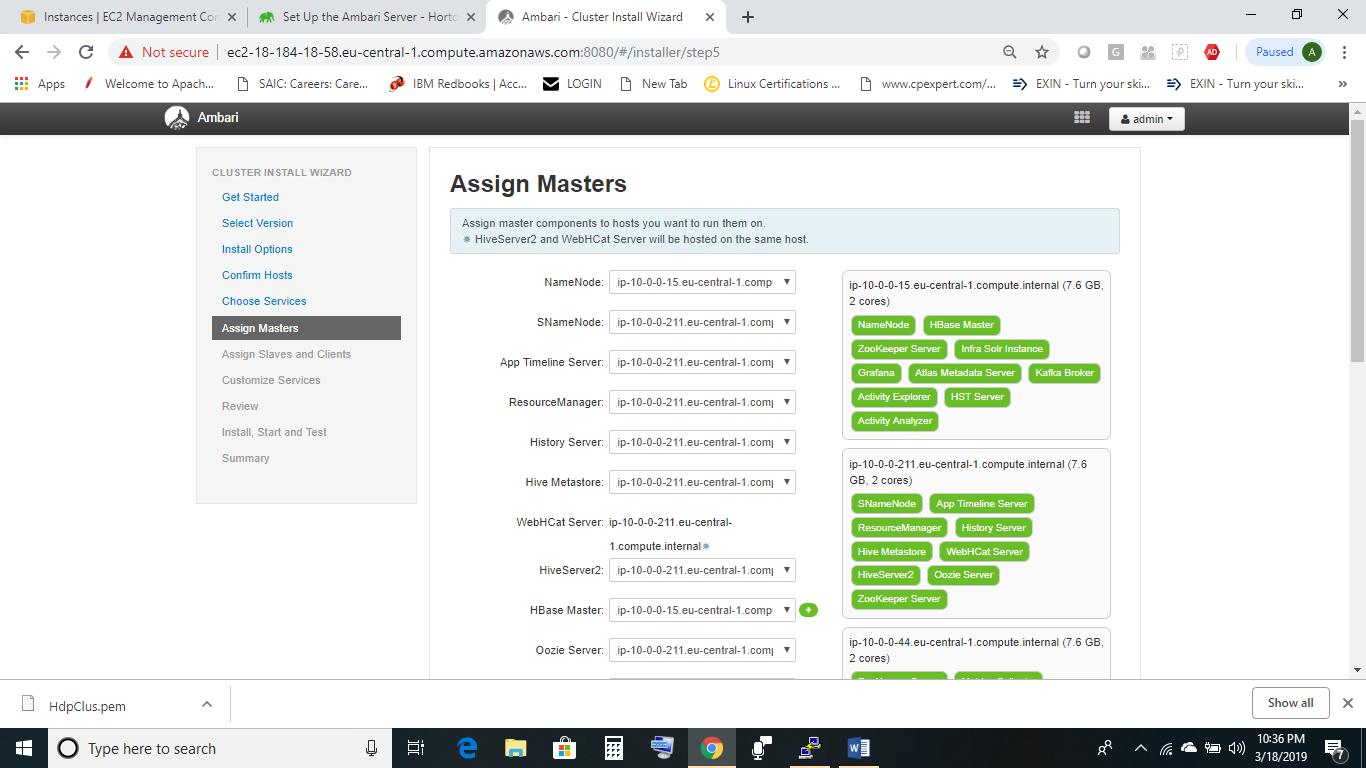


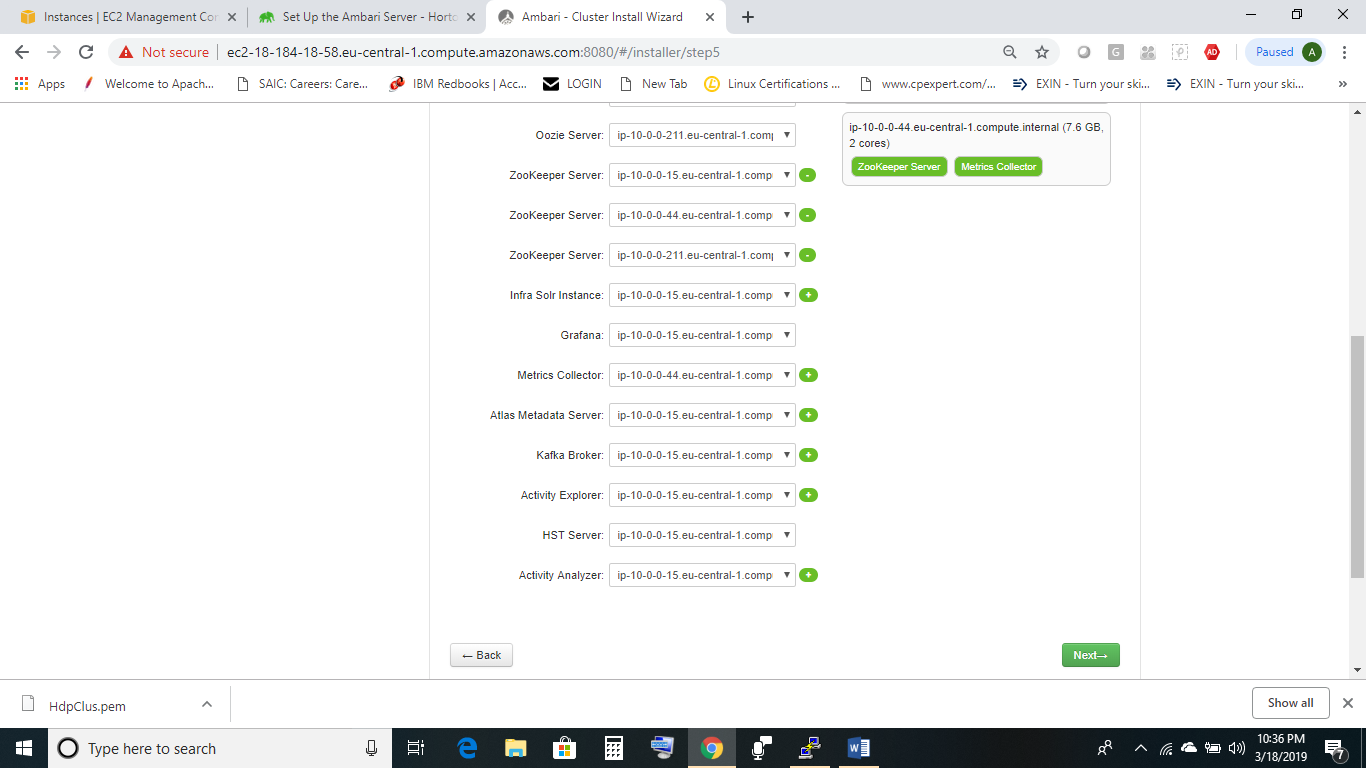


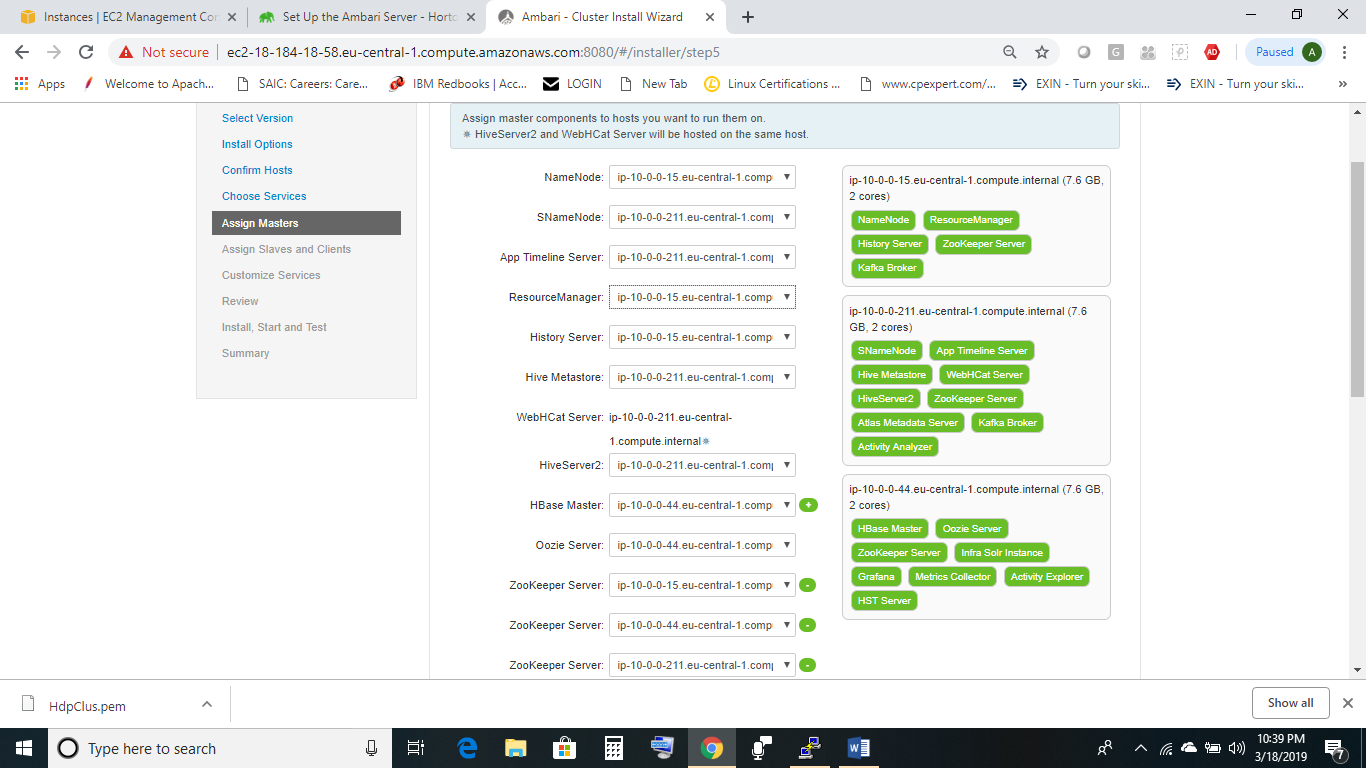




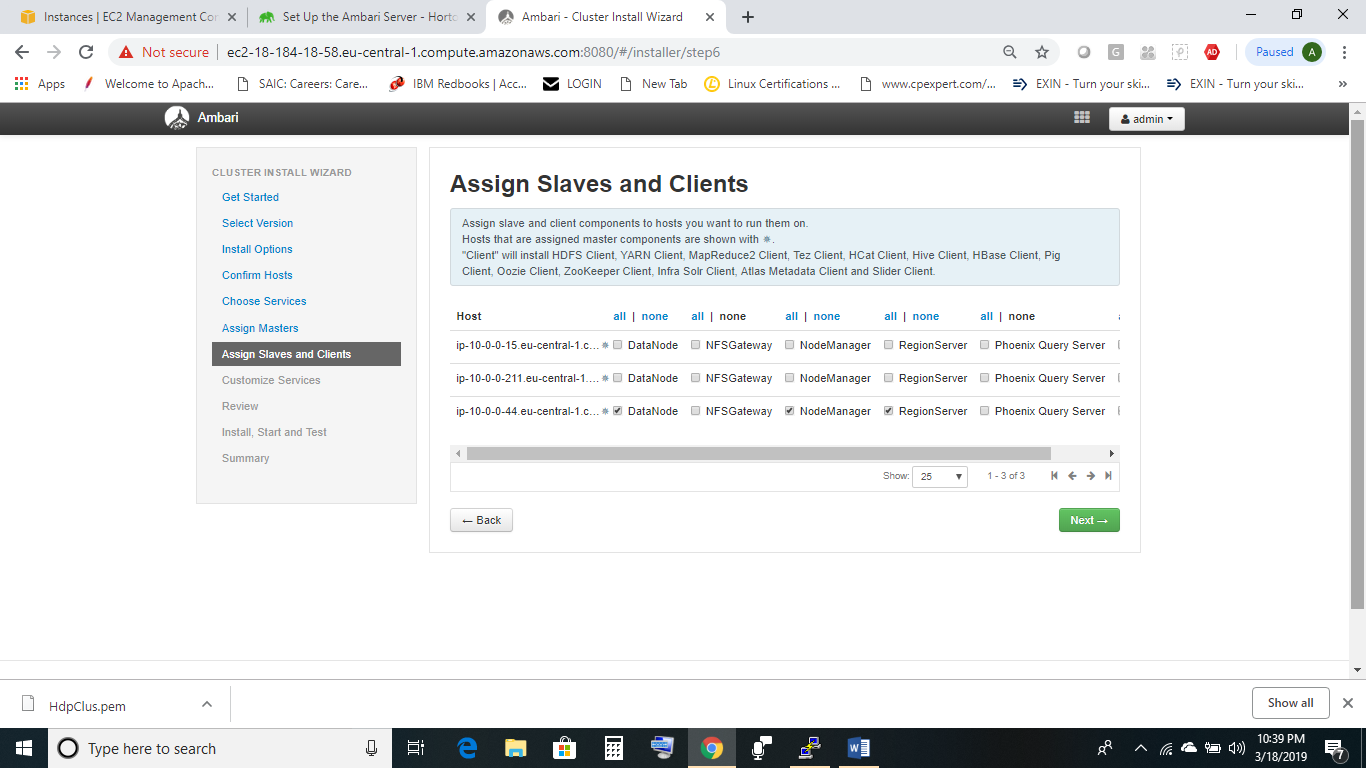
Assign components to nodes. Balancing load would be the key here.

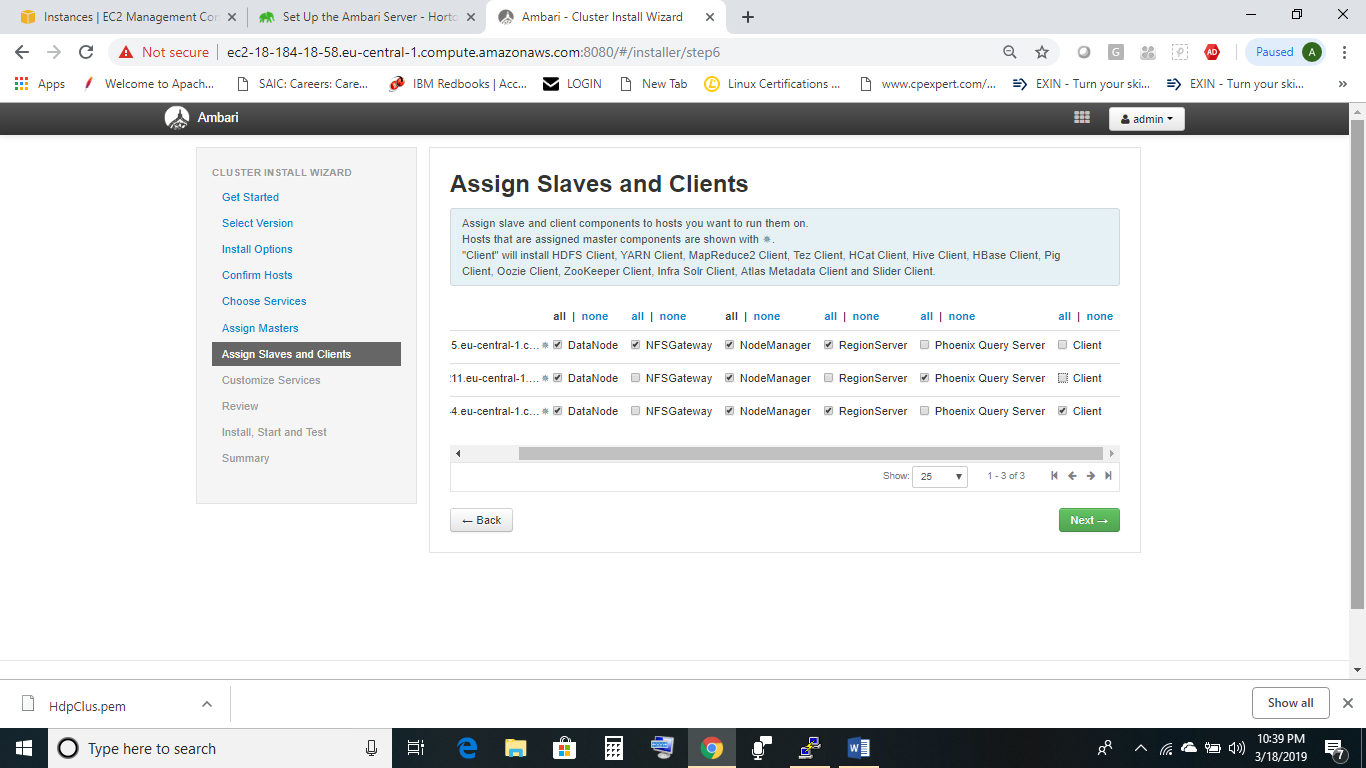


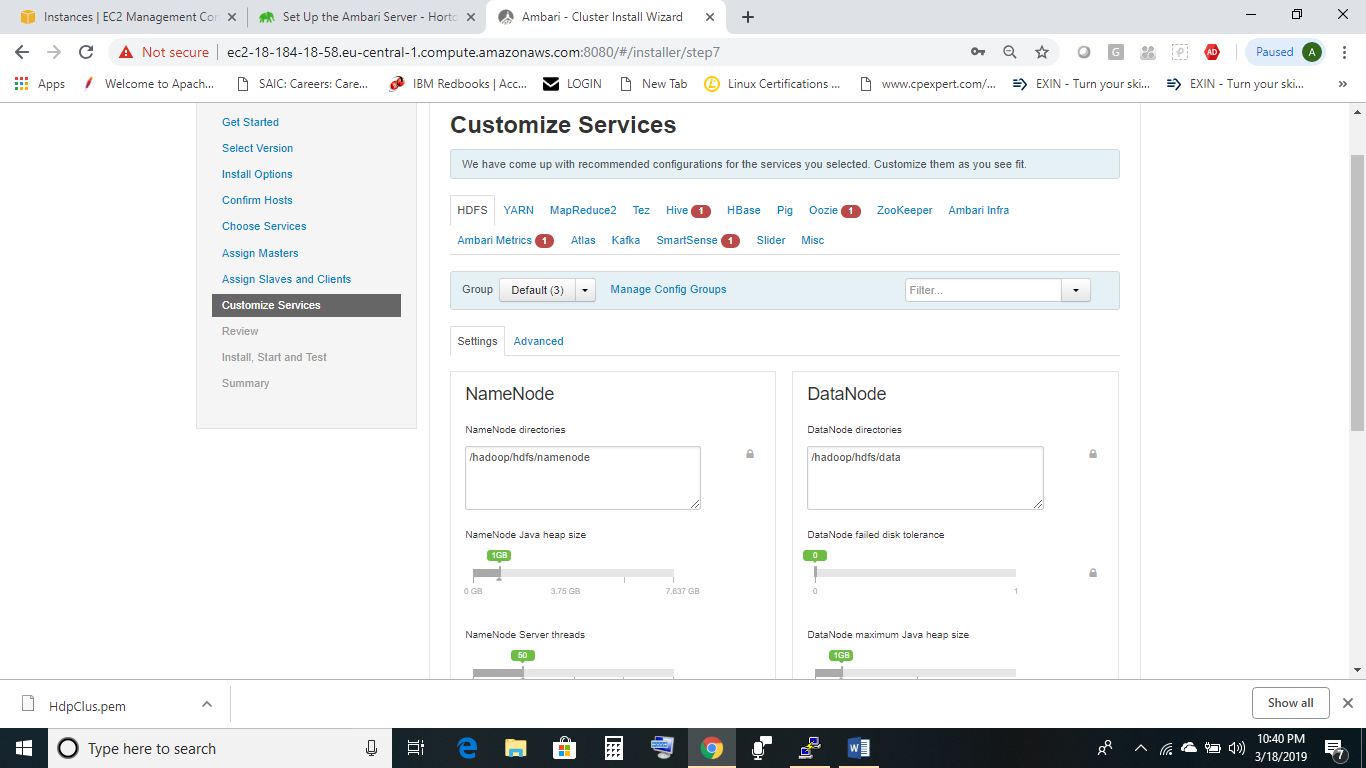


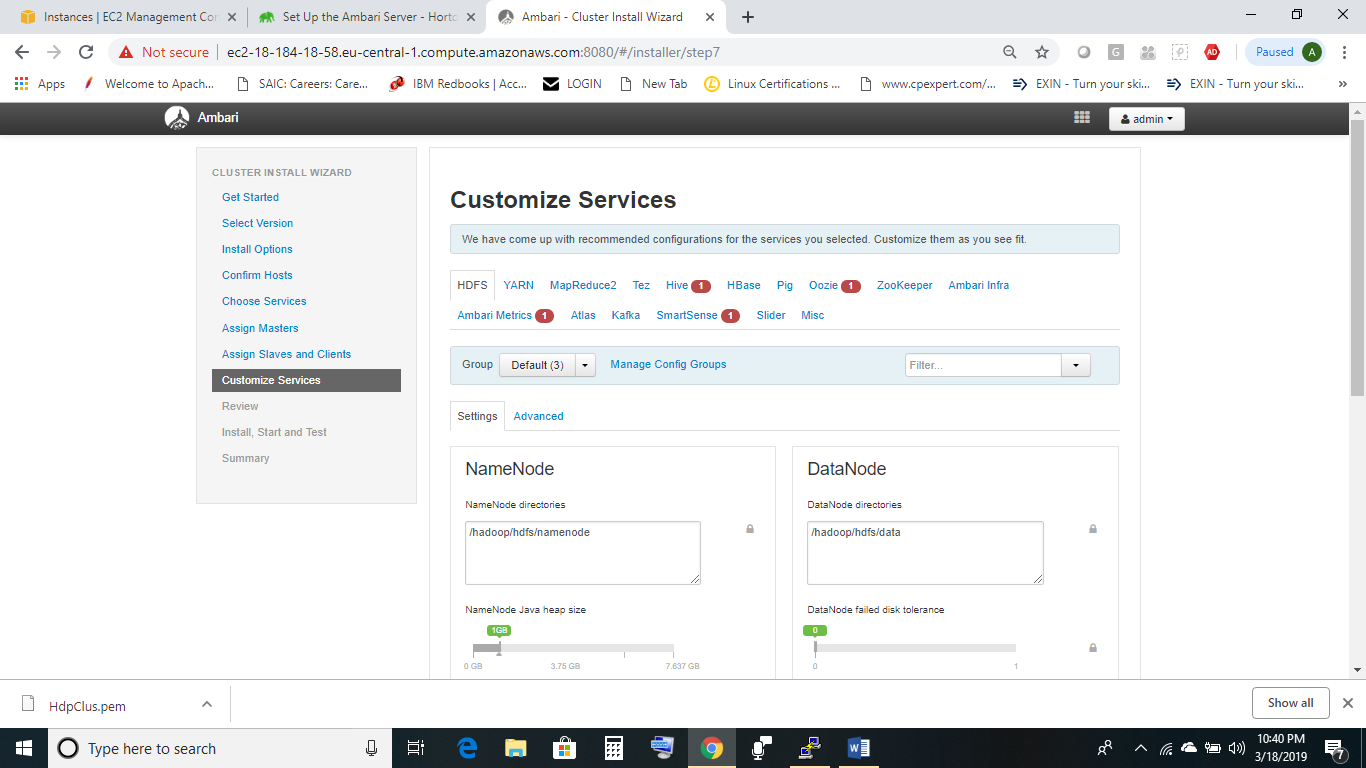


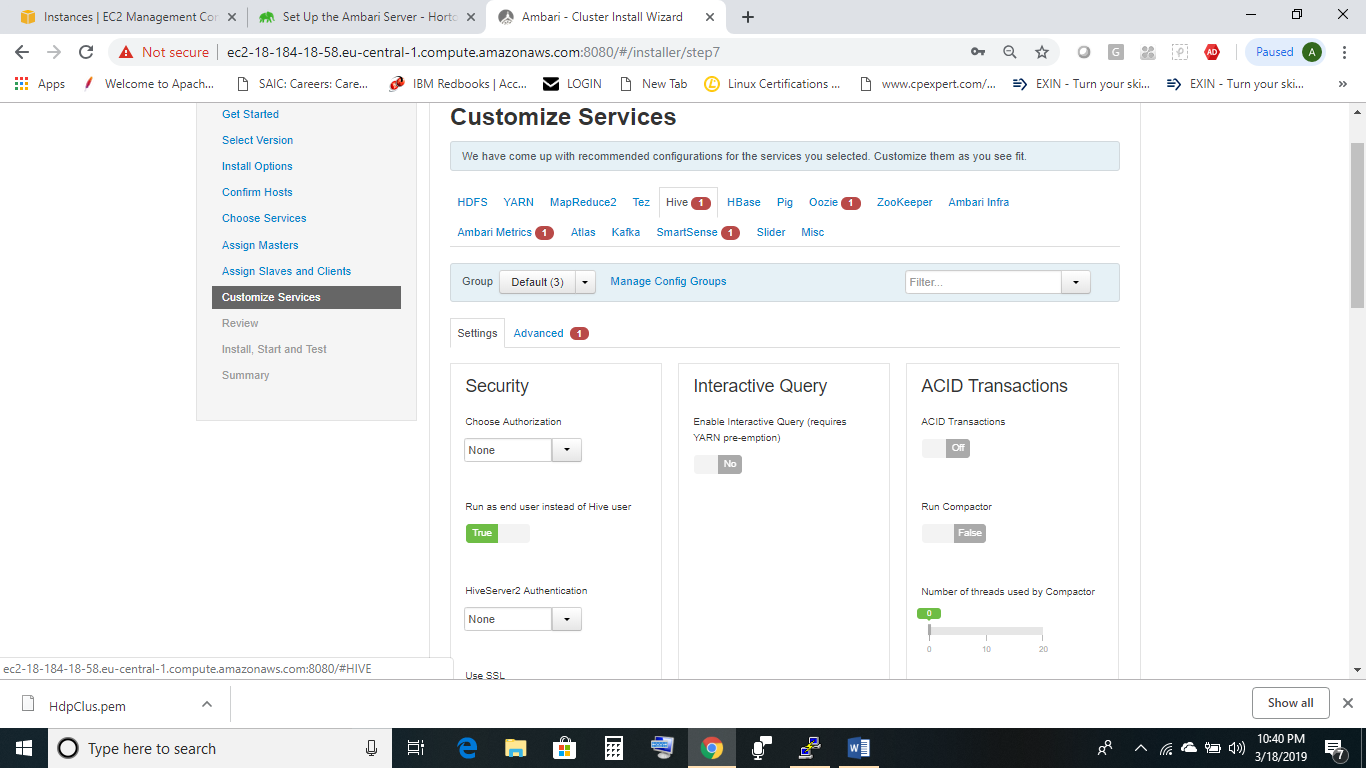
Do not assign clients on all nodes.

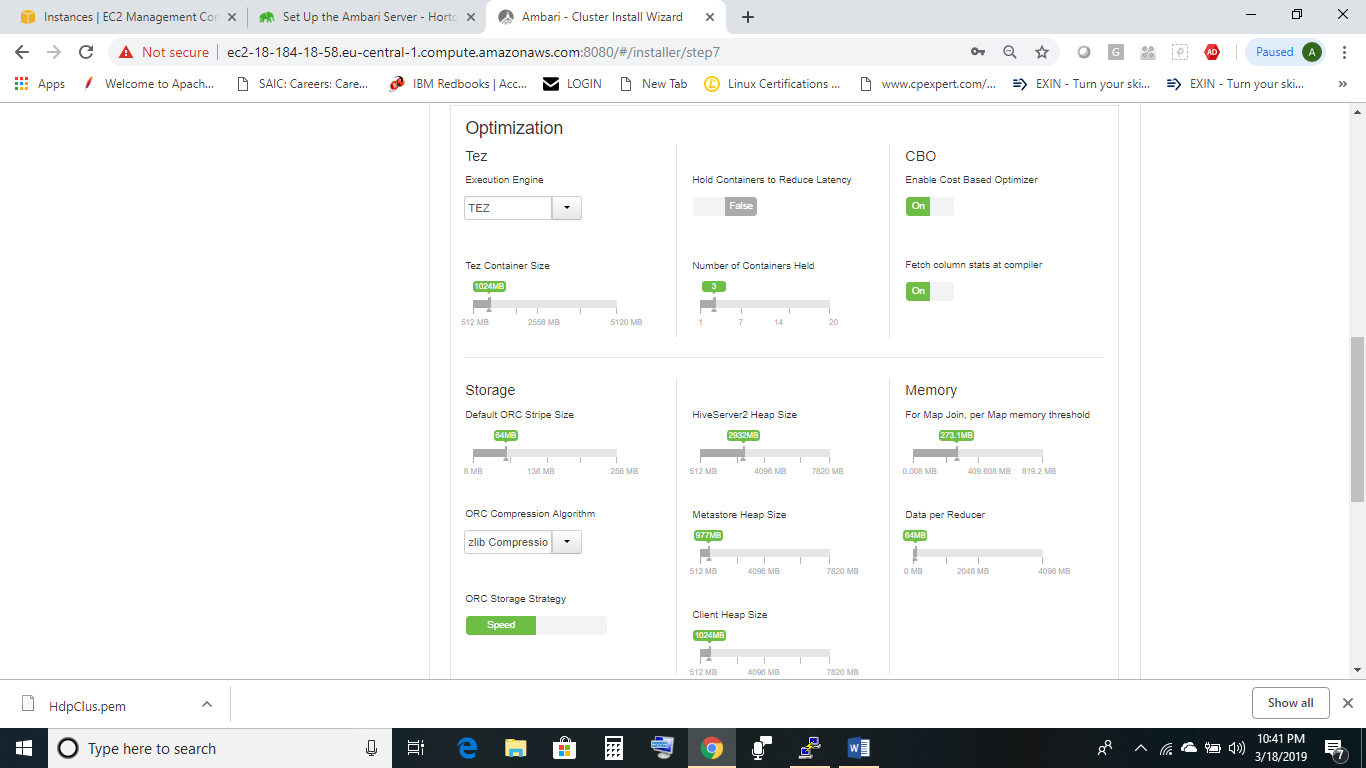


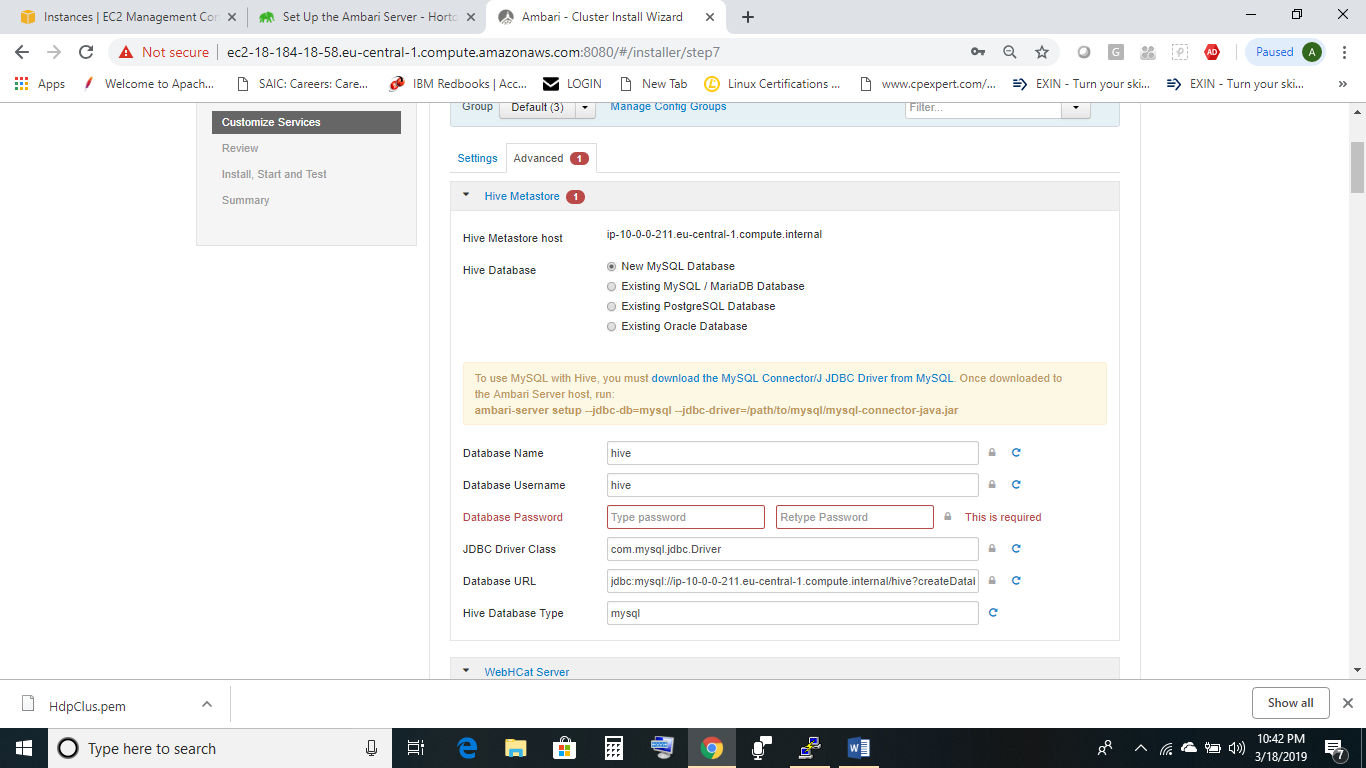


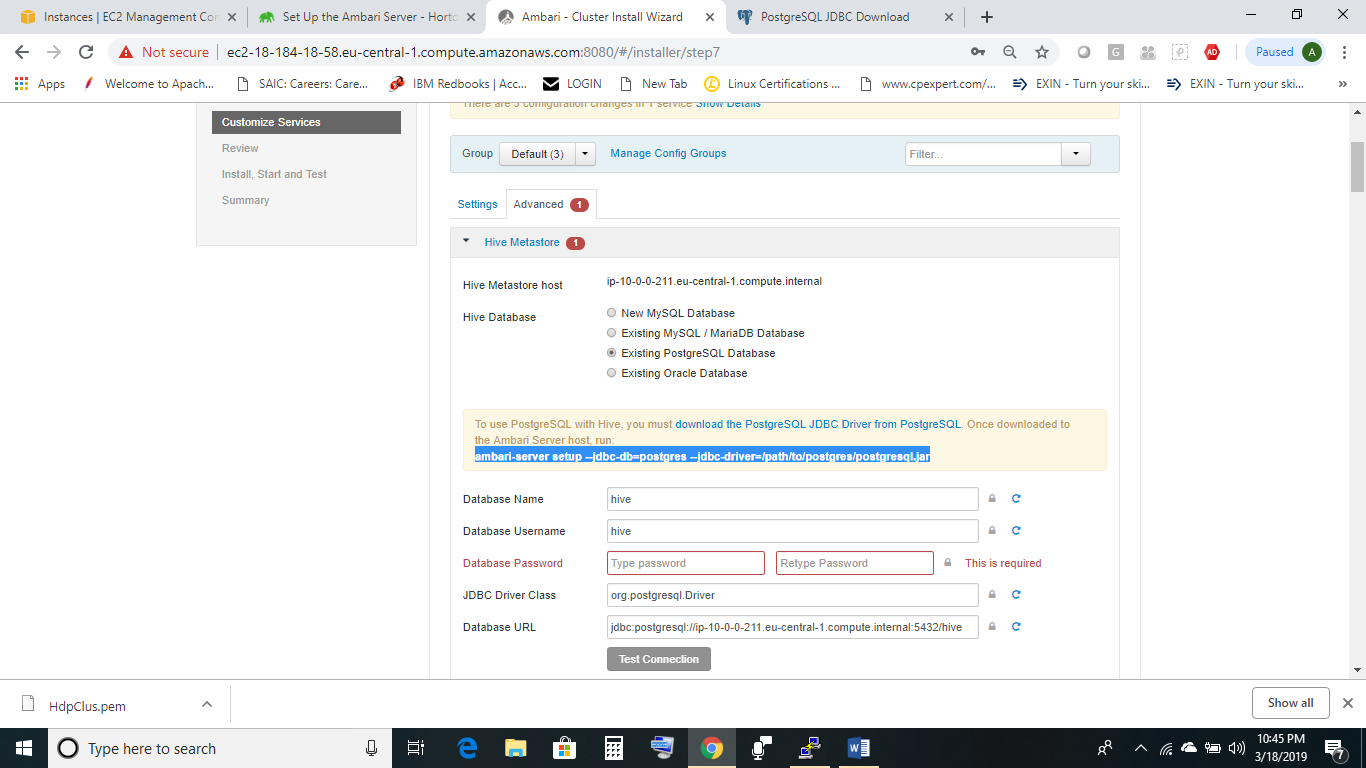




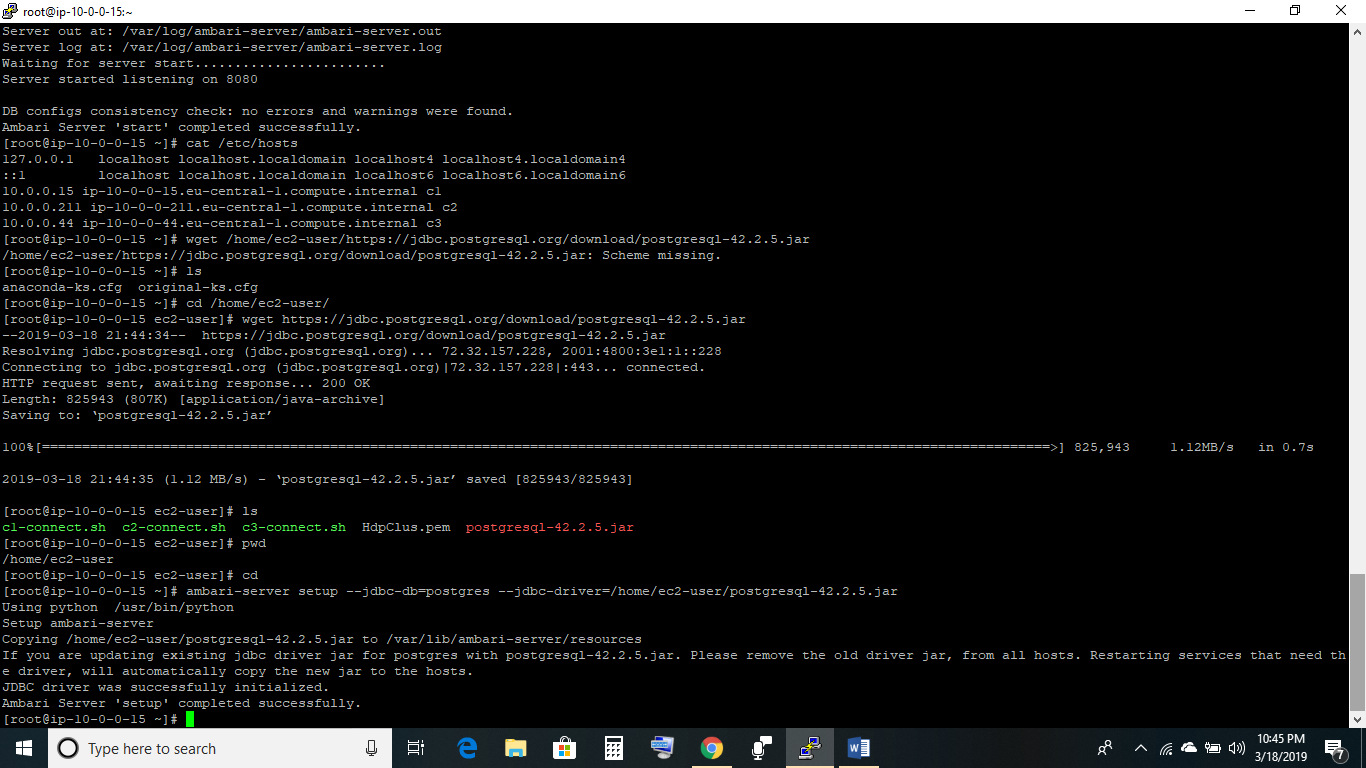


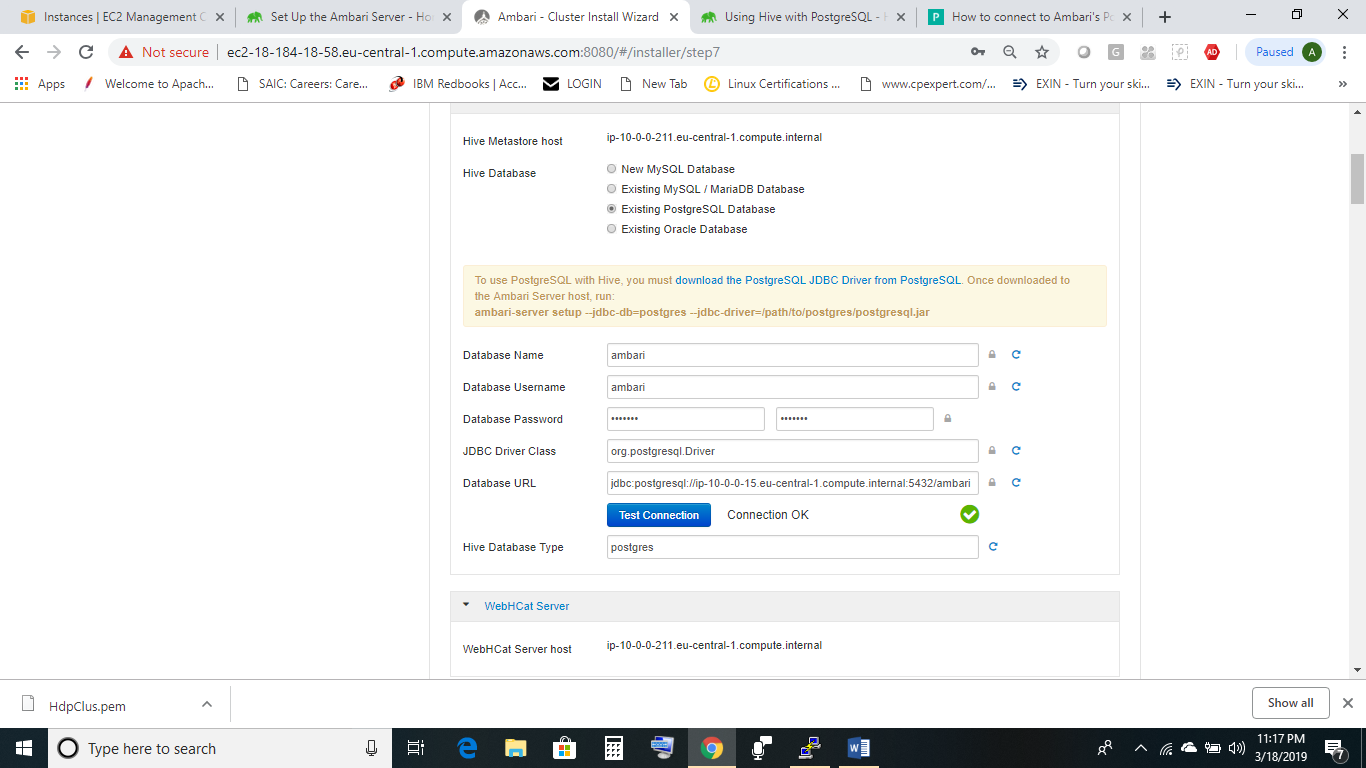


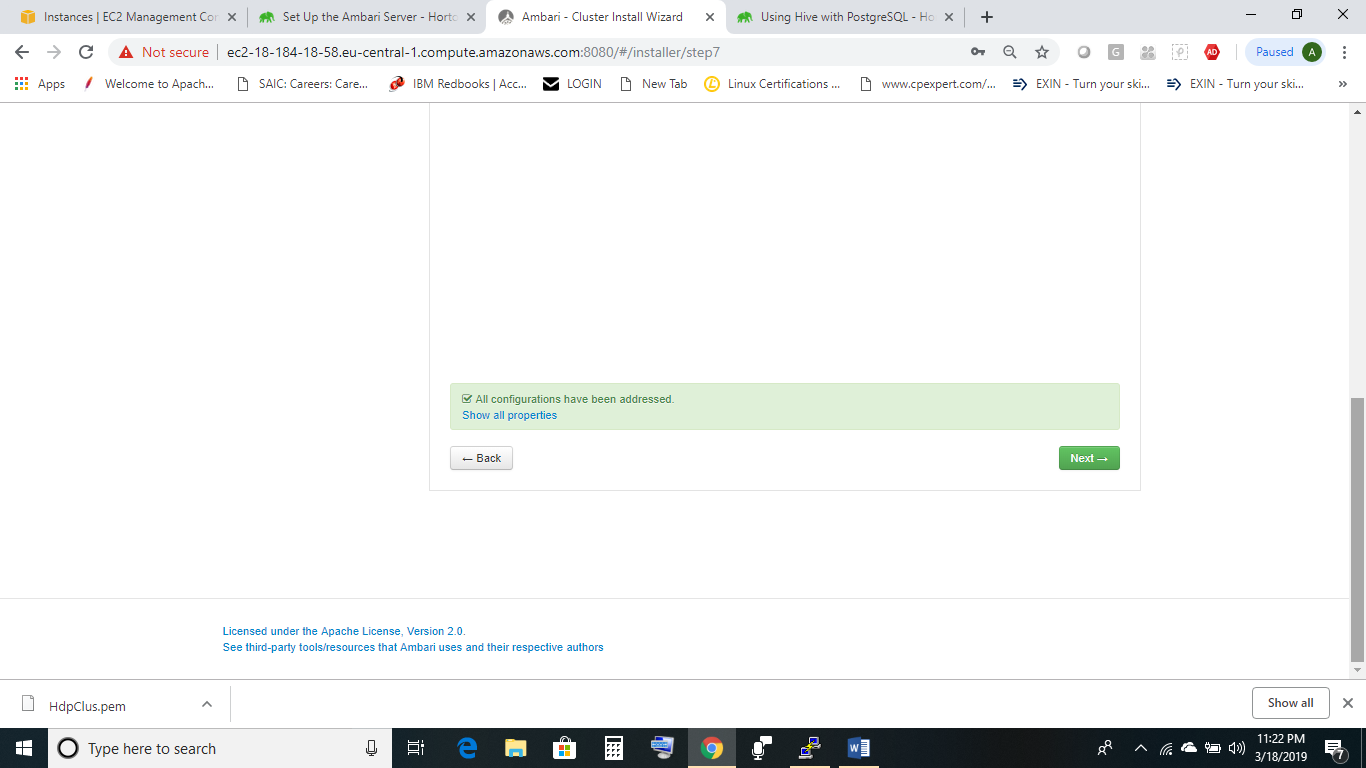


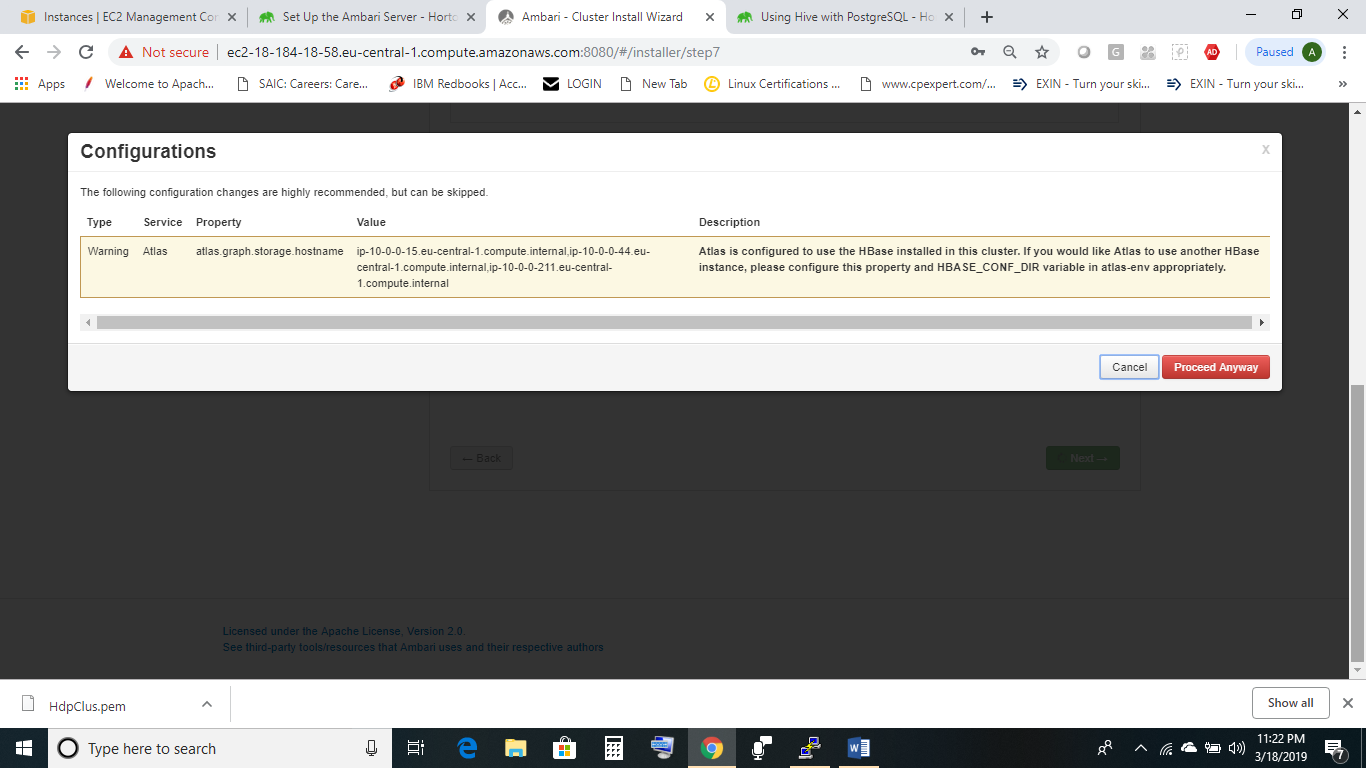


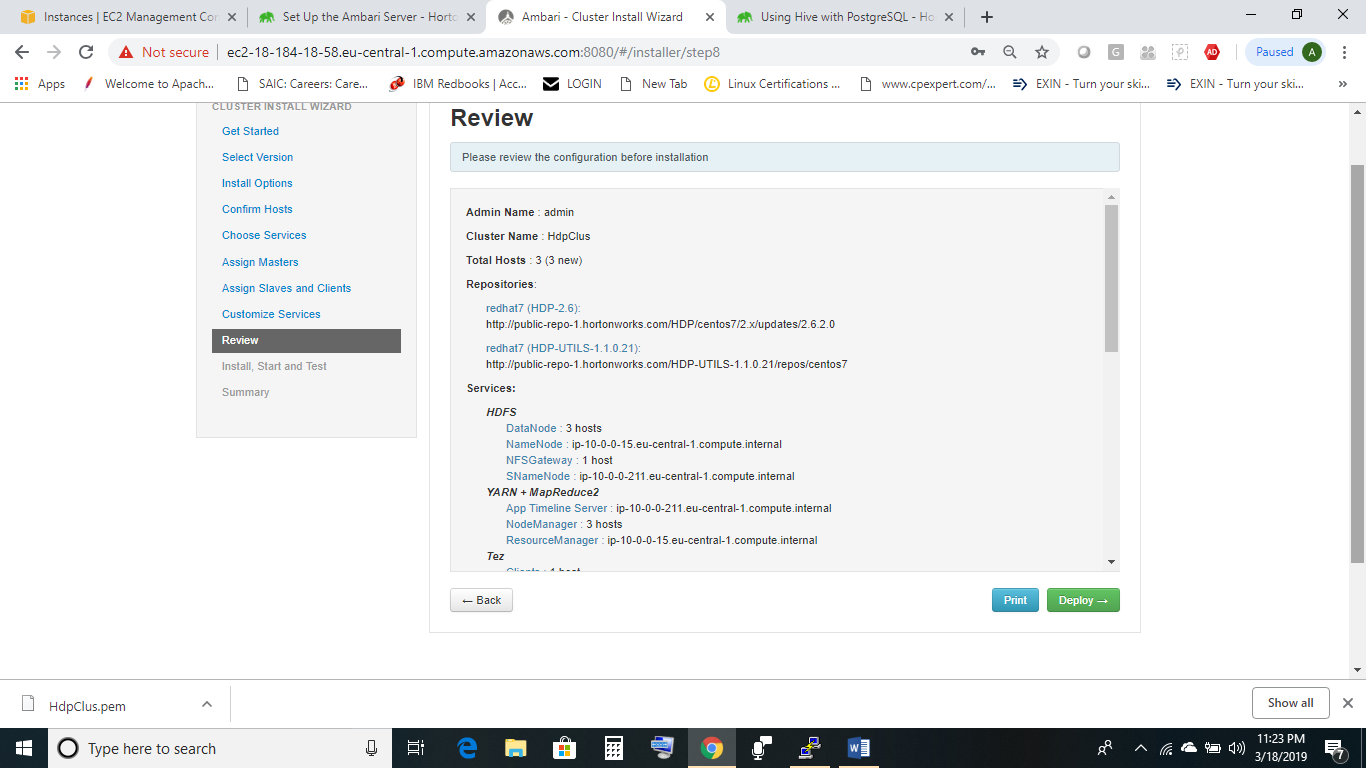
For all services you can use ambari database in beginning and later customize services to use a different database .

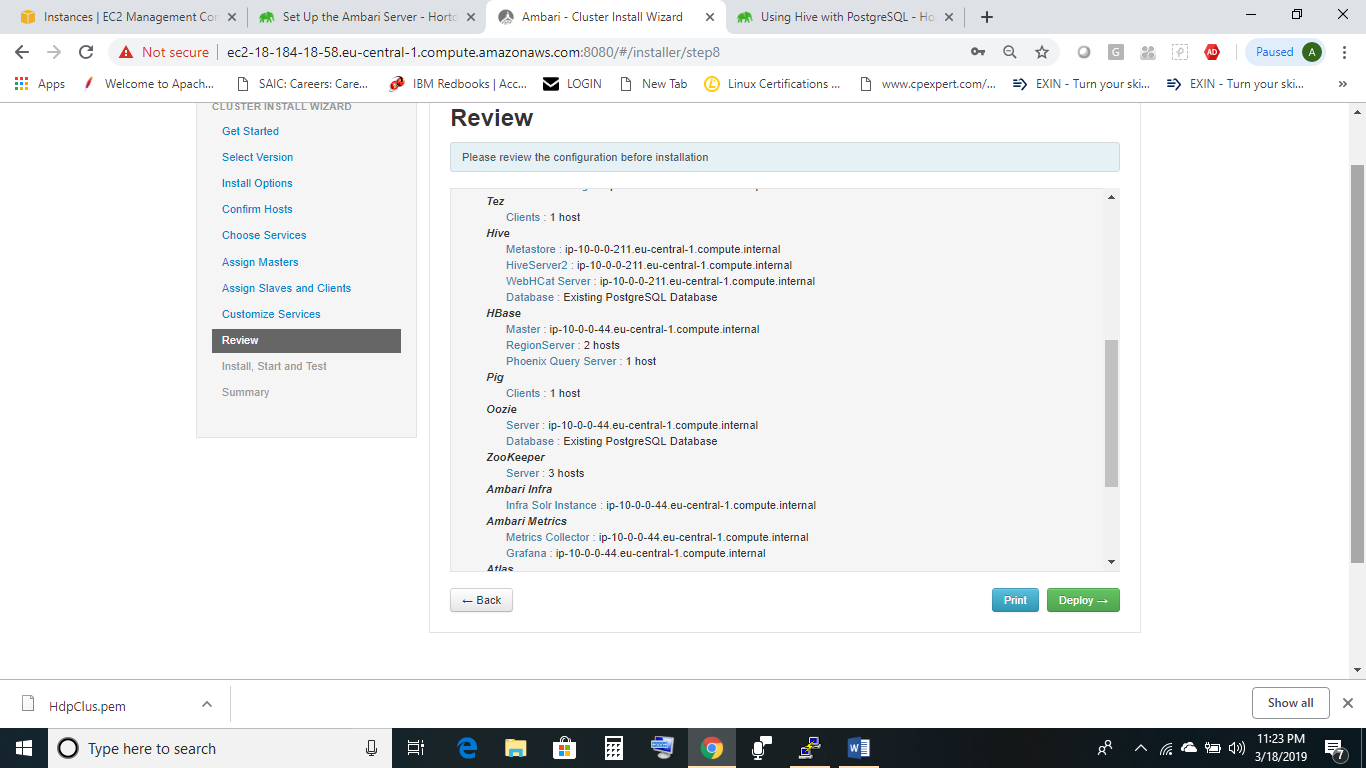


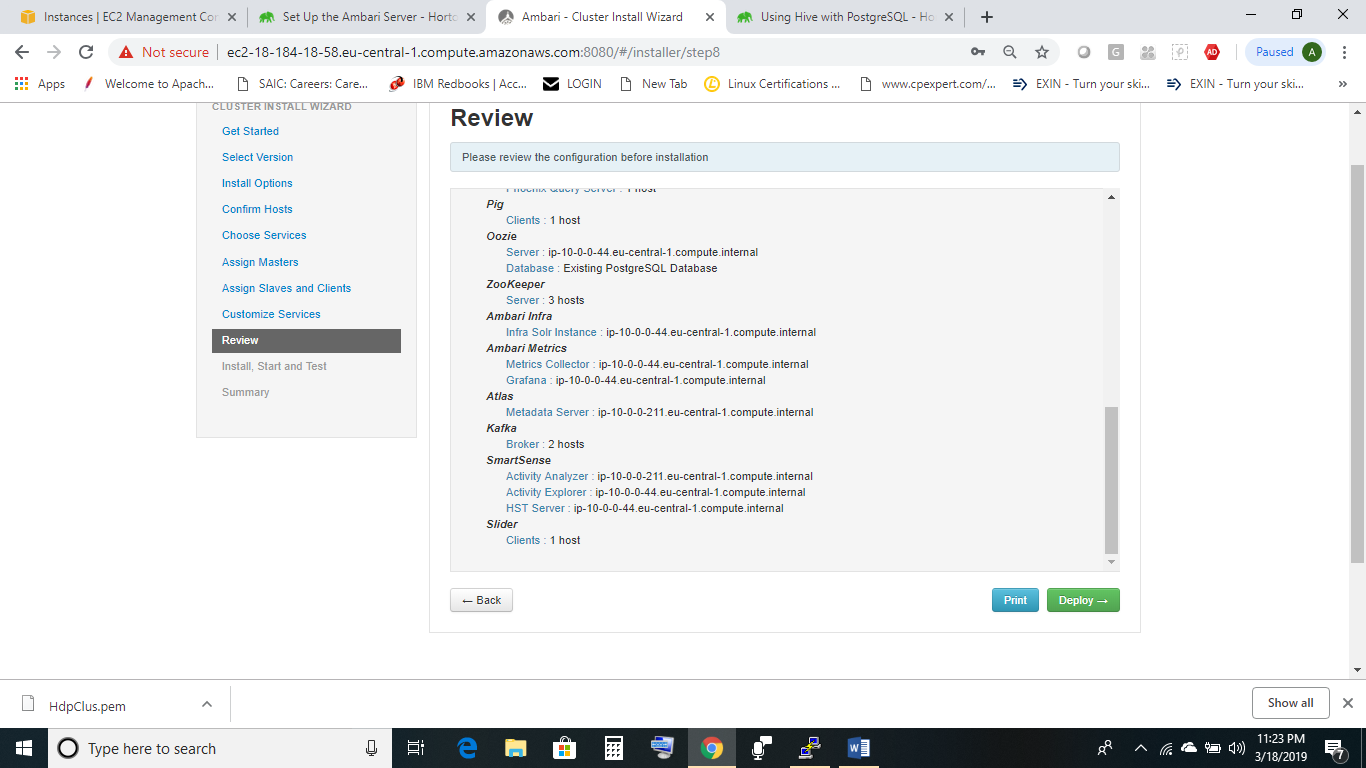


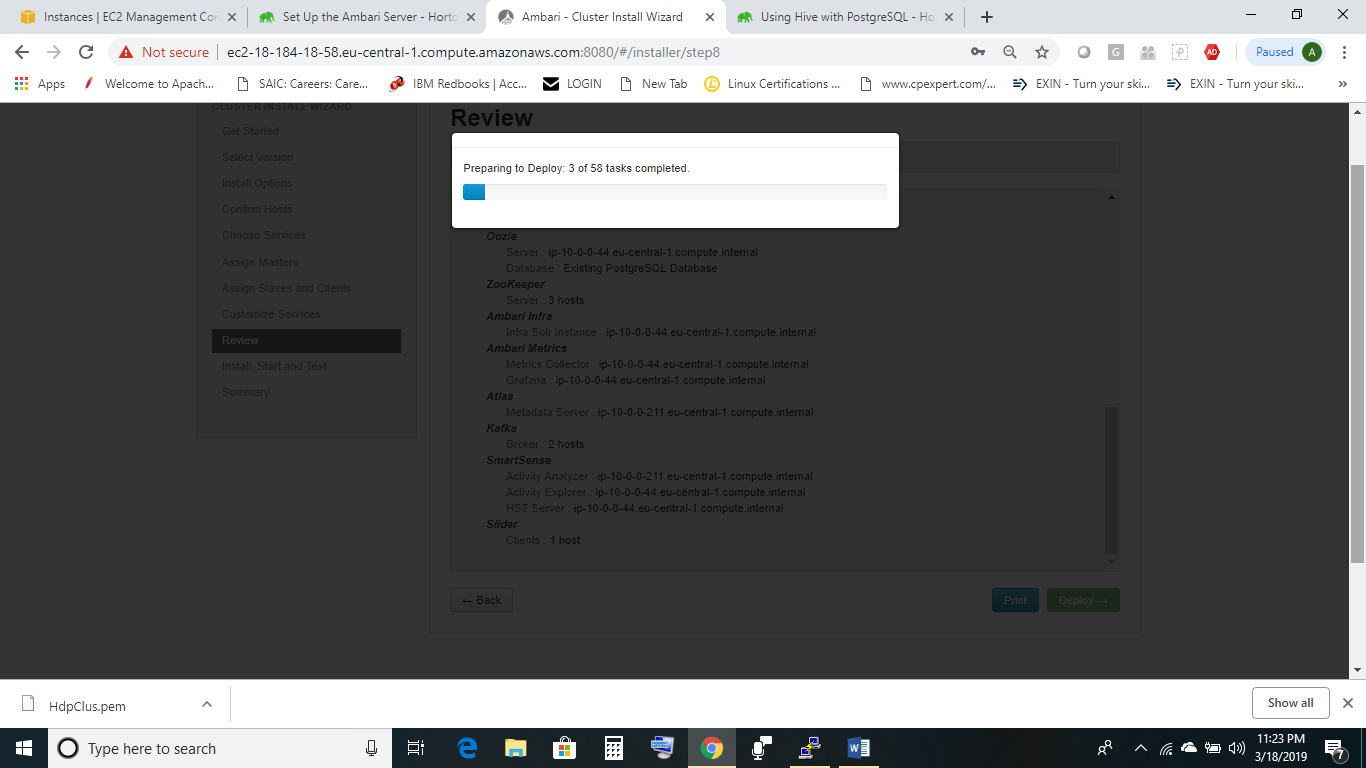


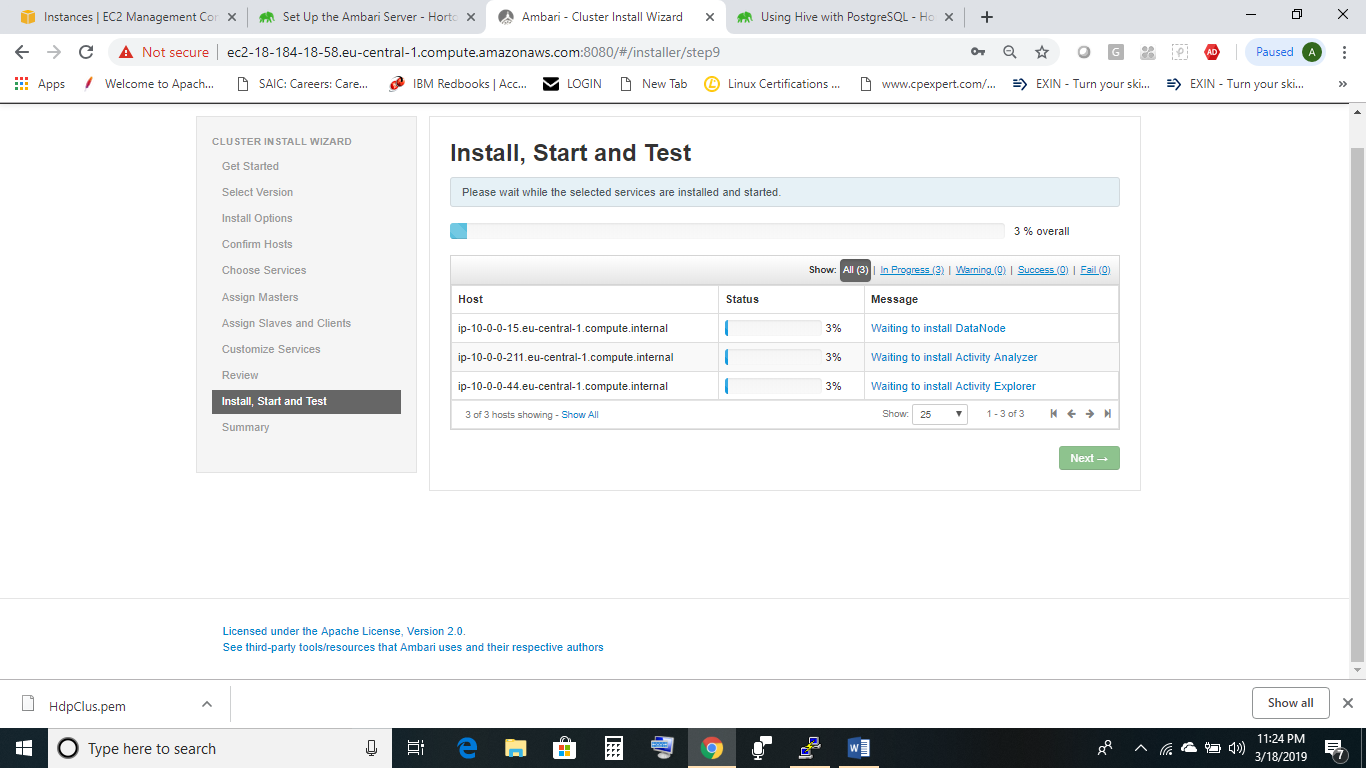


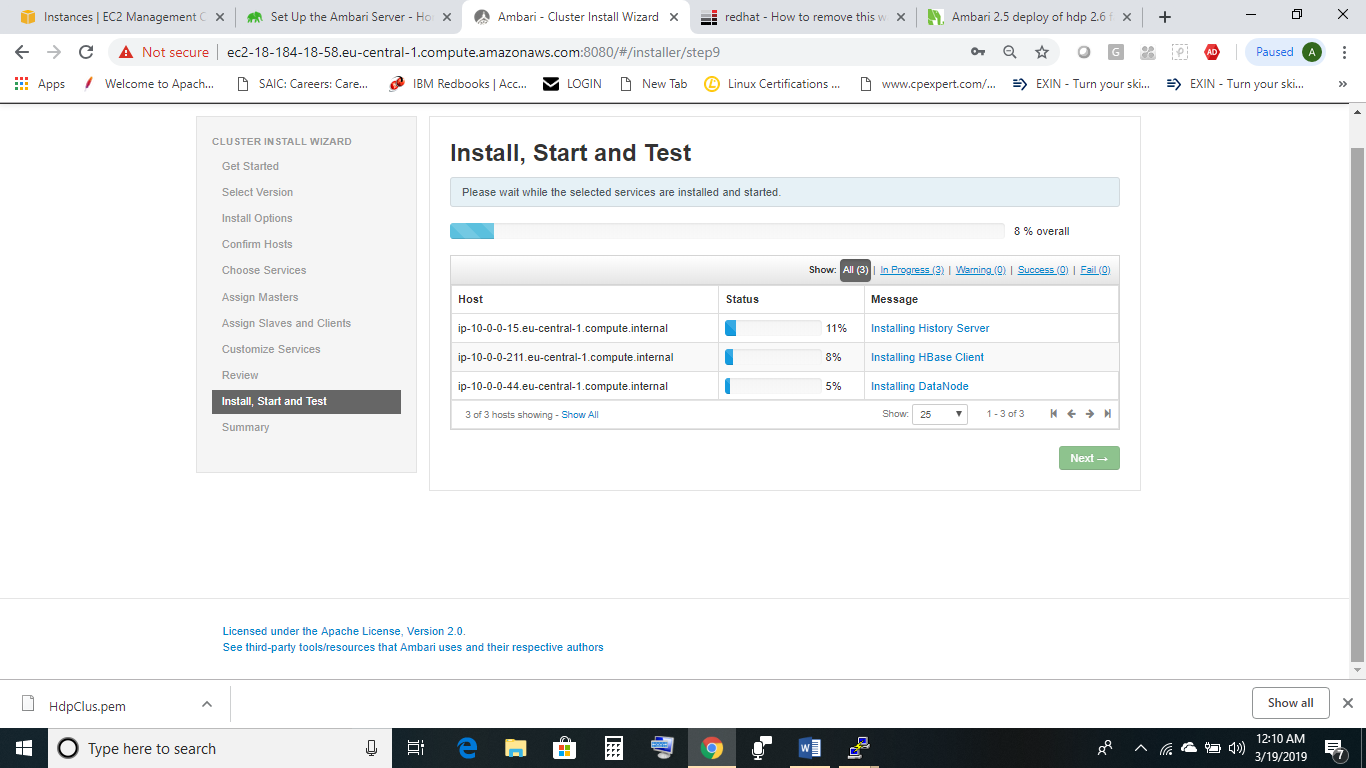


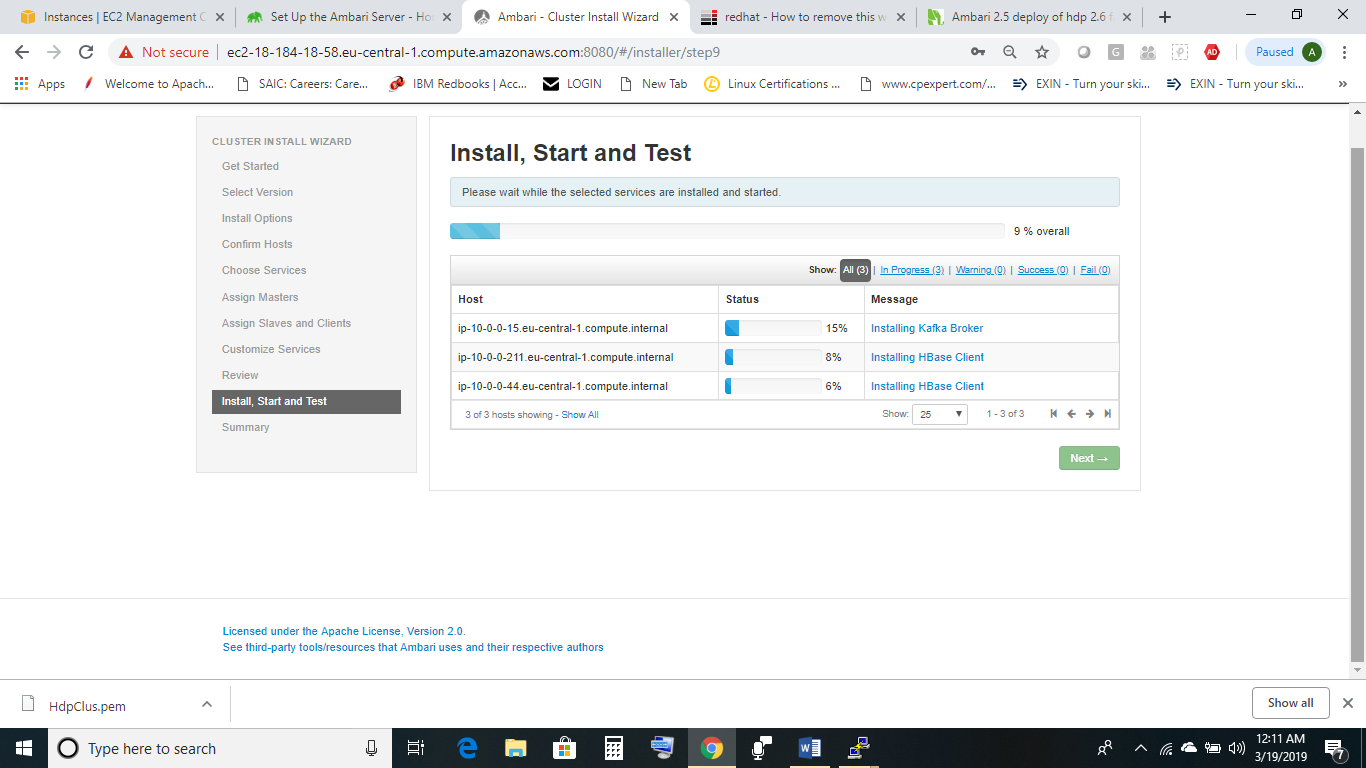


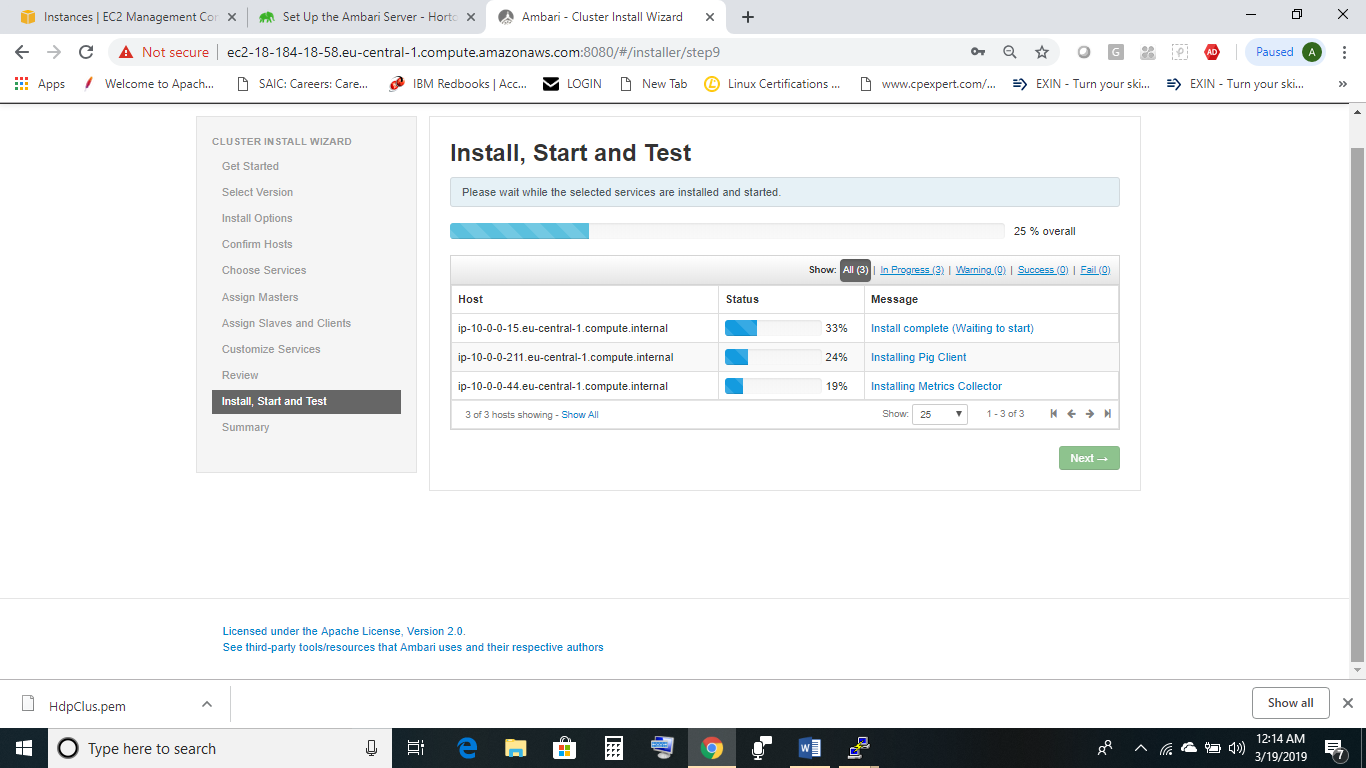


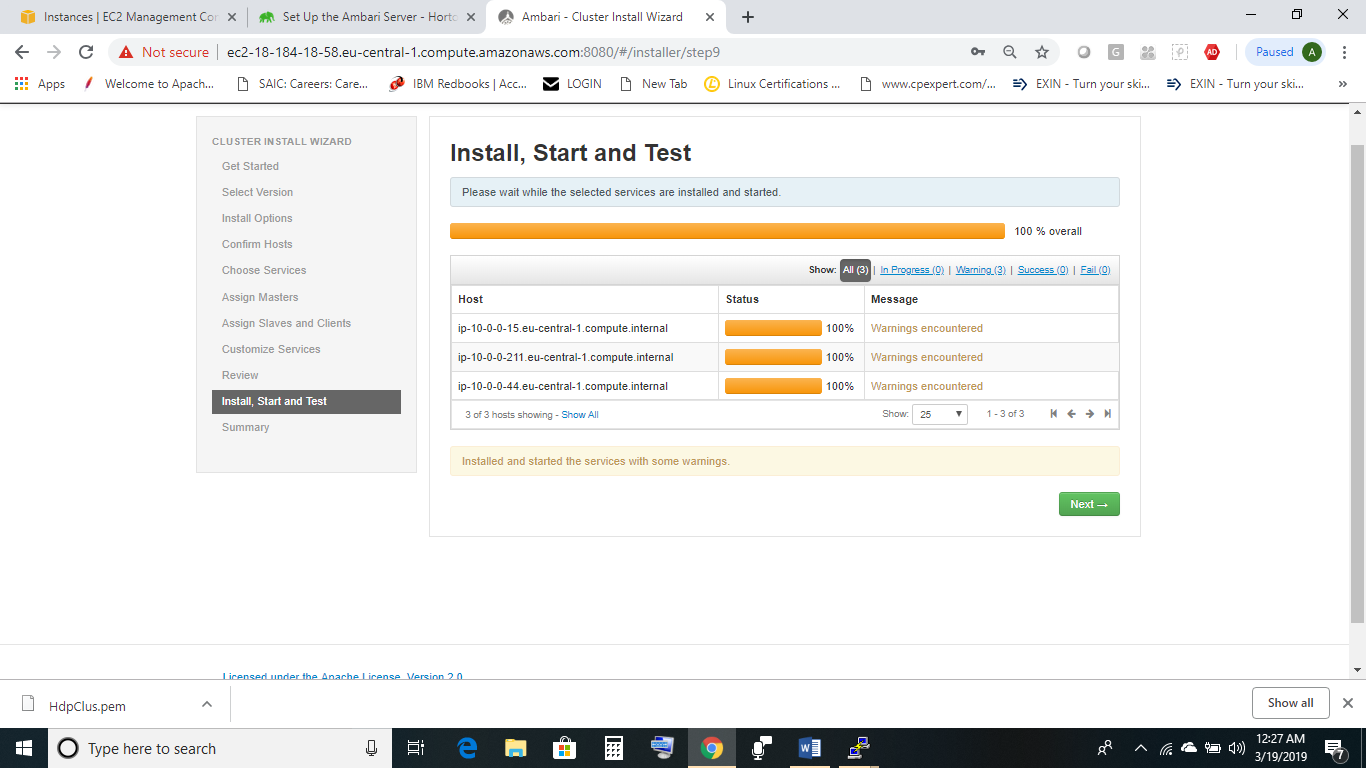


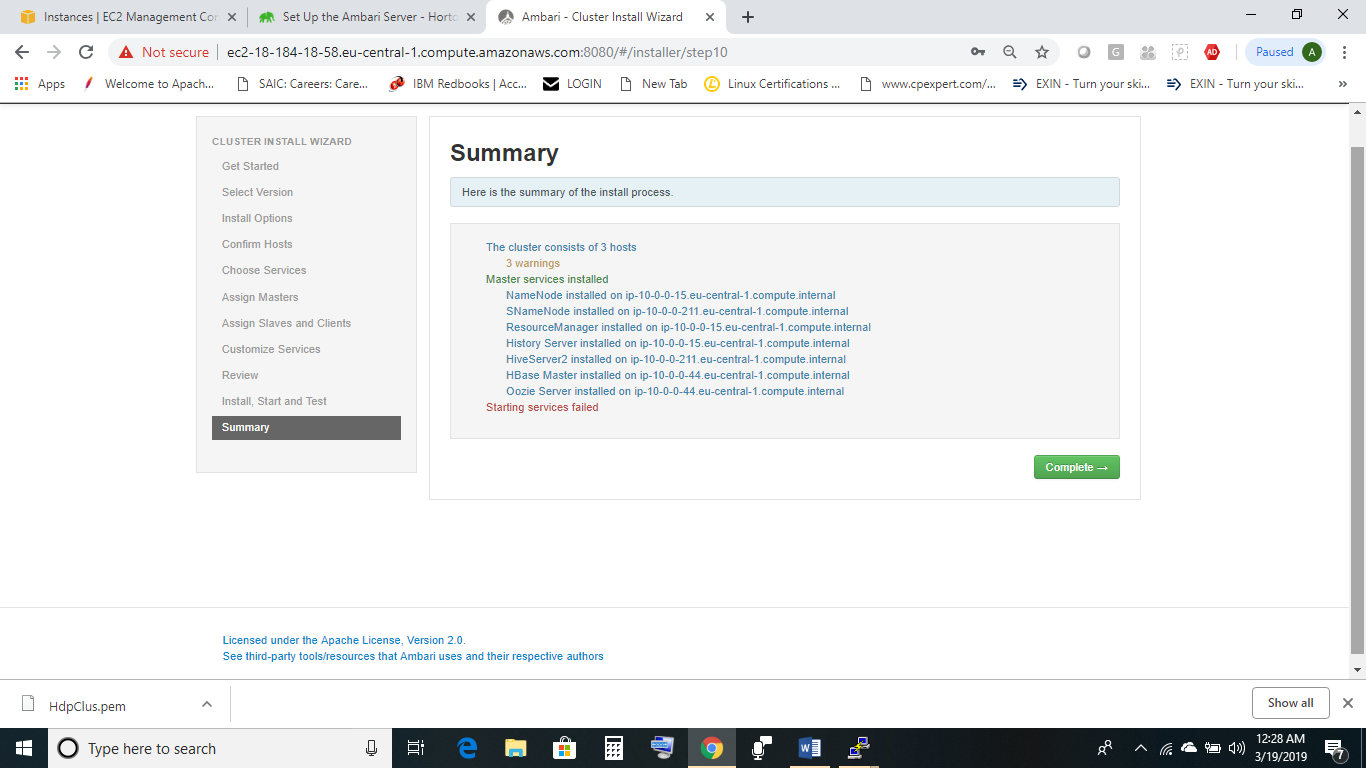


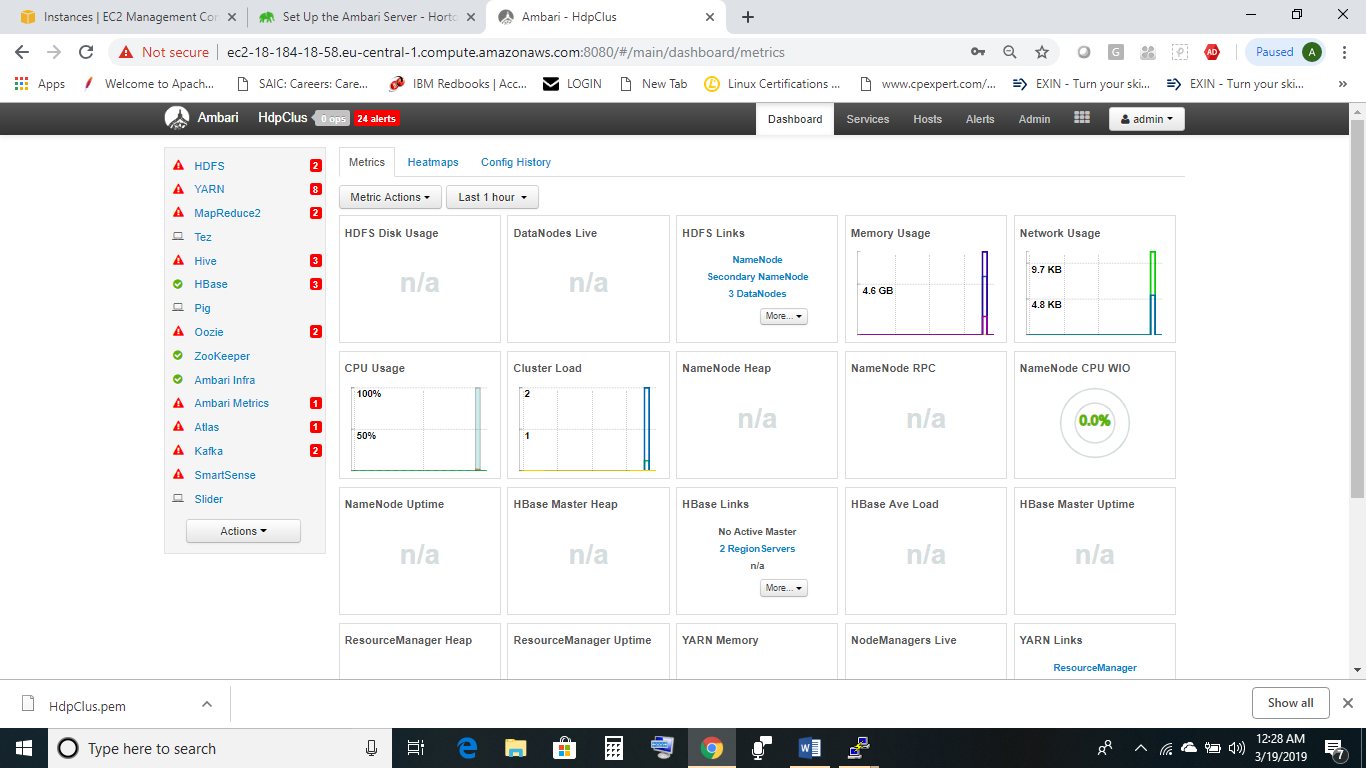












Some of the **Issues encountered were due to libtirpc-devel package missing :**

-to-remove-this-warning “this-system-is-not-registered-to-red-hat-subscription entitlement server”.

two solutions

1. in the plugin configuration file disable plugin

vim /etc/yum/pluginconf.d/subscription-manager.conf

enabled=0

1. or register to the satellite

If package missing

sudo yum-config-manager --enable rhui-REGION-rhel-server-optional  
sudo yum install libtirpc-devel

If using Mysql/MariaDB for setup

**U**sing Ambari with MySQL/MariaDB

To set up MySQL/MariaDB for use with Ambari:

**Steps**

1. On the Ambari Server host:
   1. [Download the MySQL Connector/JDBC driver from MySQL](https://dev.mysql.com/downloads/connector/j/).
   2. On the Ambari Server host run:

ambari-server setup --jdbc-db=mysql --jdbc-driver=/path/to/mysql/mysql-connector-java.jar

* 1. Confirm that .jar is in the Java share directory.

ls /usr/share/java/mysql-connector-java.jar

* 1. Make sure the .jar file has the appropriate permissions - 644.

1. Create a user for Ambari and grant it permissions.
   1. For example, using the MySQL database admin utility:

# mysql -u root -p

CREATE USER '<AMBARIUSER>'@'%' IDENTIFIED BY '<AMBARIPASSWORD>';

GRANT ALL PRIVILEGES ON \*.\* TO '<AMBARIUSER>'@'%';

CREATE USER '<AMBARIUSER>'@'localhost' IDENTIFIED BY '<AMBARIPASSWORD>';

GRANT ALL PRIVILEGES ON \*.\* TO '<AMBARIUSER>'@'localhost';

CREATE USER '<AMBARIUSER>'@'<AMBARISERVERFQDN>' IDENTIFIED BY '<AMBARIPASSWORD>';

GRANT ALL PRIVILEGES ON \*.\* TO '<AMBARIUSER>'@'<AMBARISERVERFQDN>';

FLUSH PRIVILEGES;

* 1. Where <AMBARIUSER> is the Ambari user name, <AMBARIPASSWORD> is the Ambari user password and<AMBARISERVERFQDN> is the Fully Qualified Domain Name of the Ambari Server host.

1. Load the Ambari Server database schema.
   1. You must pre-load the Ambari database schema into your MySQL/MariaDB database using the schema script. Run the script in the same location where you find the Ambari-DDL-MySQL-CREATE.sql file. You should find the Ambari-DDL-MySQL-CREATE.sql file in the /var/lib/ambari-server/resources/ directory of the Ambari Server host, after you have installed Ambari Server.

mysql -u <AMBARIUSER> -p

CREATE DATABASE <AMBARIDATABASE>;

USE <AMBARIDATABASE>;

SOURCE Ambari-DDL-MySQL-CREATE.sql;

* 1. Where <AMBARIUSER> is the Ambari user name and <AMBARIDATABASE> is the Ambari database name.

1. When setting up the Ambari Server, select Advanced Database Configuration > Option [3] MySQL/MariaDB and enter the credentials you defined in Step 2. for user name, password and database name.