Please follow this documentation for adding the Node to PROD Cluster:

Node for PROD cluster:

**ITSUSRALSP07674**

CR DESCRIPTION:

===============================

This CR is to add new node to Production Cluster.

Please find the steps below to be followed :

Steps:

Within host: ITSUSRALSP07674.JNJ.COM

**1 :Add a host to Cloudera Manager**

**. Allocating sufficient space at “/opt""**

Step 1: Create directory in /app/

mkdir -p /app/opt/cloudera

Step 2: Create symlink for /opt/cloudera

ln -s /app/opt/cloudera /opt/cloudera

3 : Change permission of the opt directory in /app/

chmod 755 -R /app/opt/

4: The output shall be like below , showing the symlink has been created successfully:

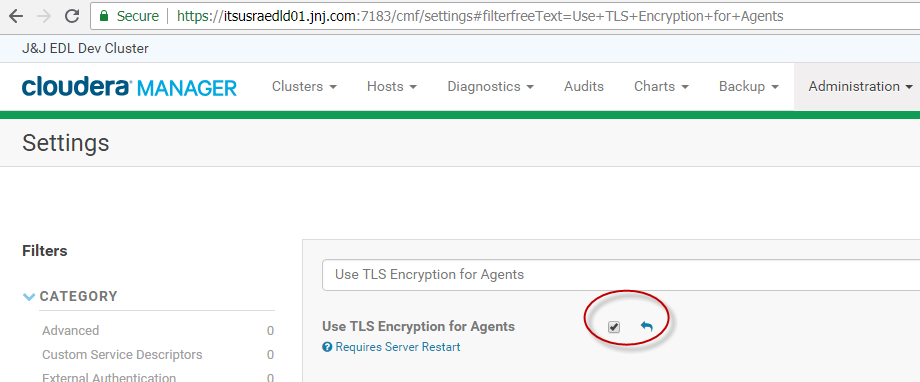
Run: ls -l /opt/

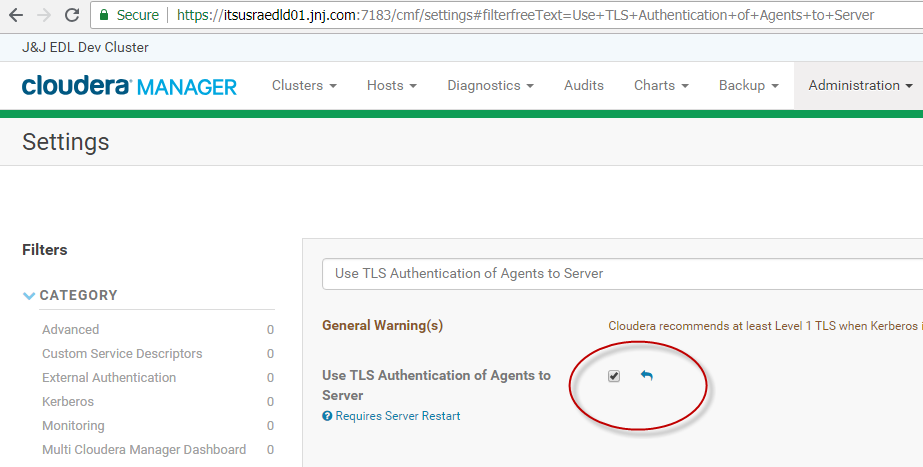
lrwxrwxrwx 1 root root 17 Jun 22 12:13 cloudera -> /app/opt/cloudera

* **Disabling TLS in Cloudera Manager :**

1:Login to Cloudera Manager and go to “Administration" --> “Settings""

2: search for “Use TLS Encryption for Agents” and “Use TLS Authentication of Agents to Server” and “uncheck” them and “save.





* **To Restart Cloudera SCM Server :**

1:Go to ITSSURAEDLP01.JNJ.COM

2: Check the status of Server :

service cloudera-scm-server status

3: Restart the server

service cloudera-scm-server restart

* **Changing the Hostname :**

Type “hostname” in the server :

Editing the Network file in :

cat /etc/sysconfig/network

Change the “HOSTNAME=ITSUSRALSP07674” inside the file.

* **Changing the Server Name:**

Editing the Network file :

vim /etc/sysconfig/network

sysctl kernel.hostname= ITSUSRALSP07674.jnj.com

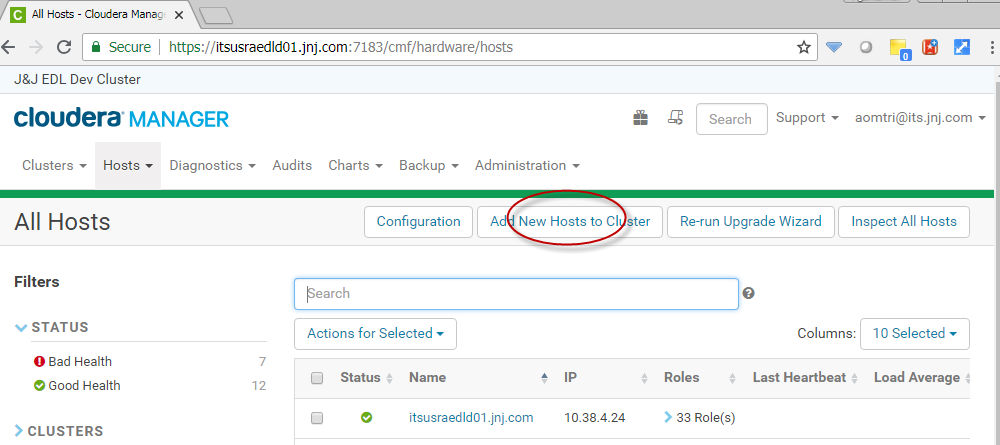
Check for hostname :

Hostname

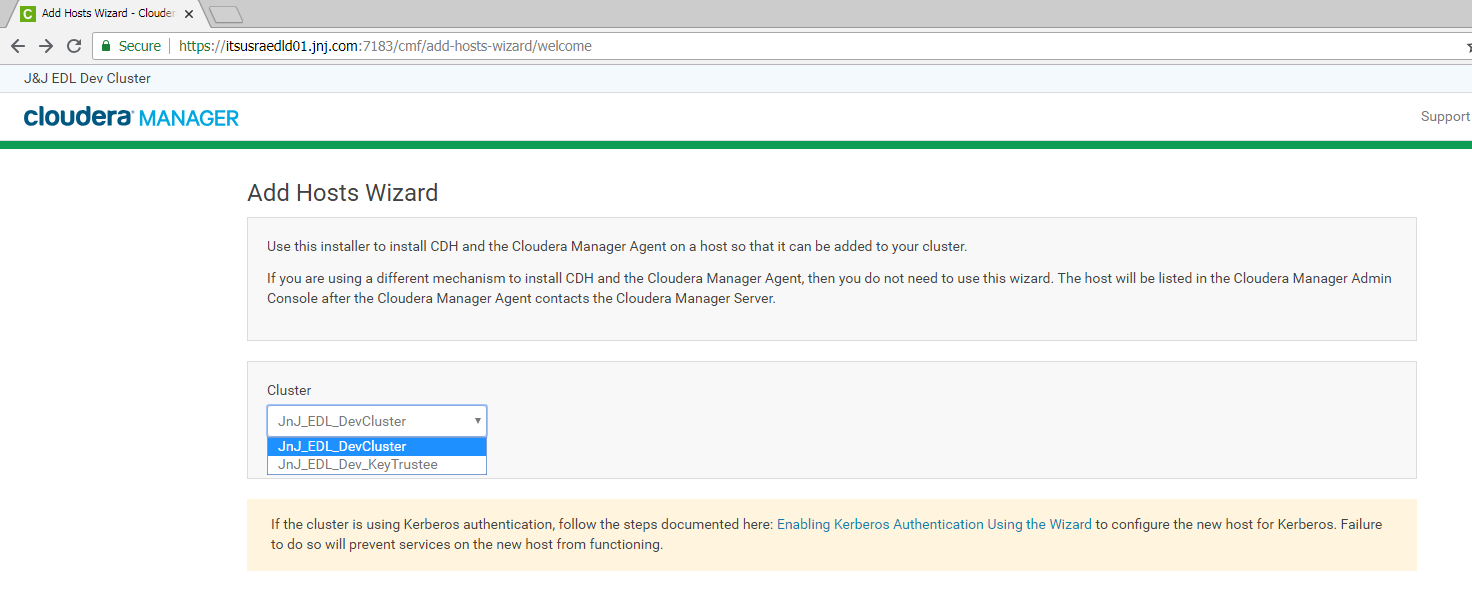
cat /etc/sysconfig/network

HOSTNAME= ITSUSRALSP07674.jnj.com

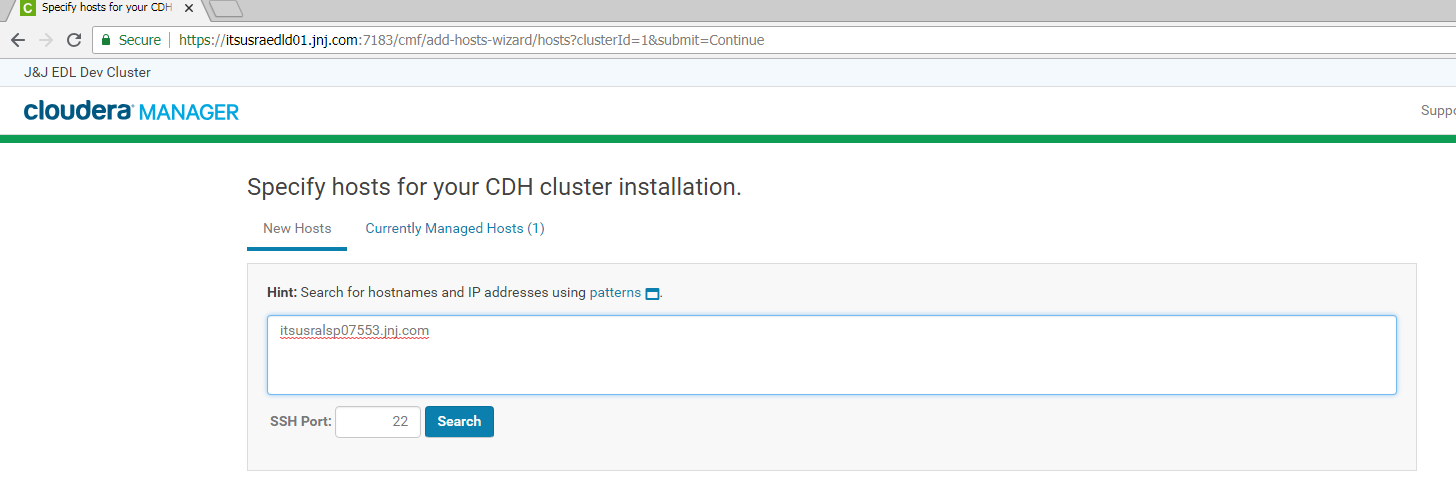
* **Adding the Node to the Cluster : Will be done for all Nodes.** 
  + 1. Login into Cloudera Manager and navigate to “Hosts” 🡪 “Add New Hosts to Cluster”.



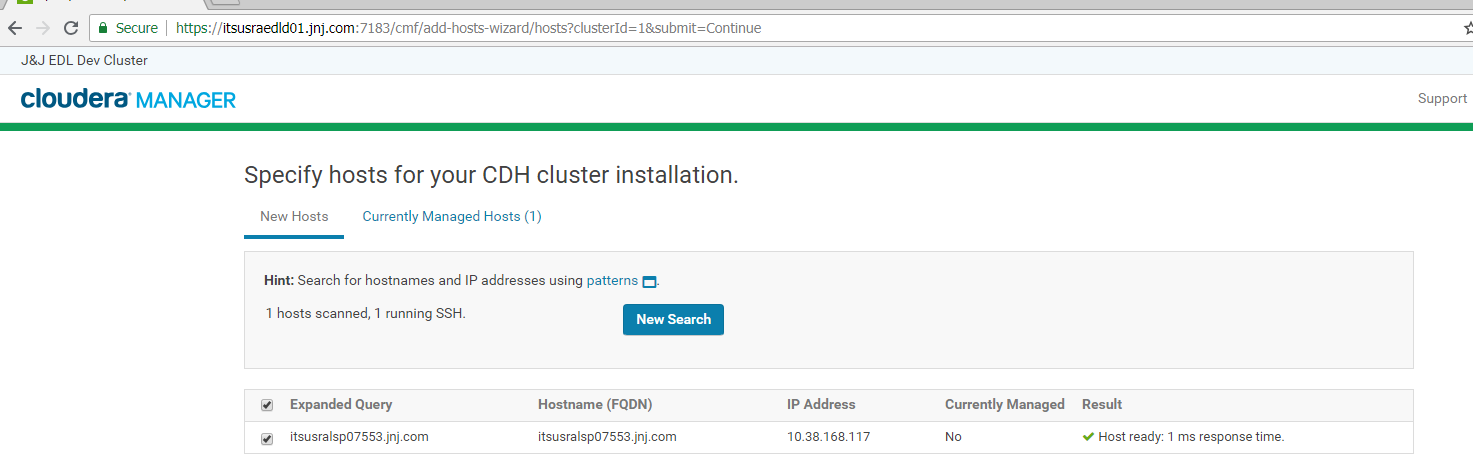
1. Select the appropriate Cluster. Select JnJ\_**EDL**\_PRODCluster in Cluster option, press continue.

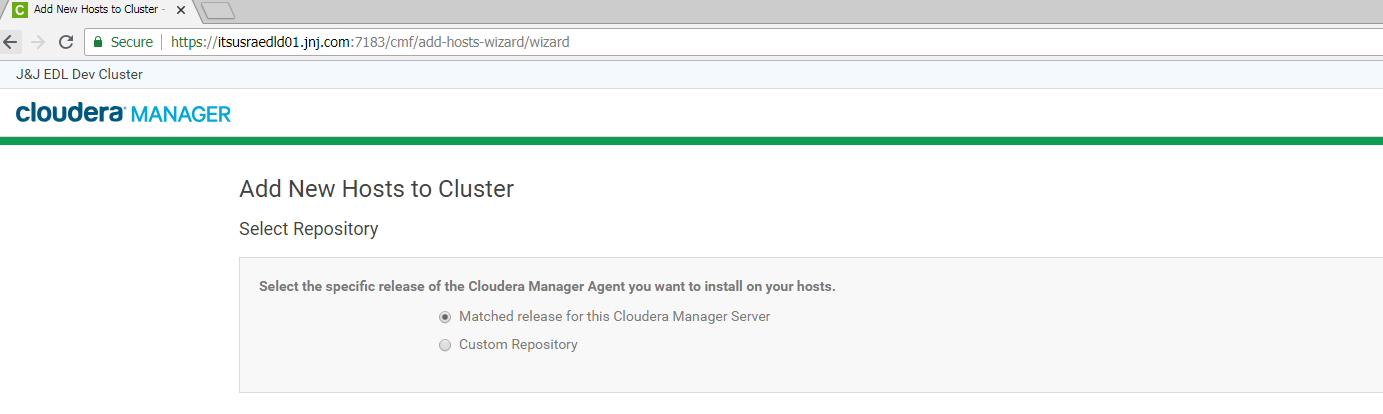


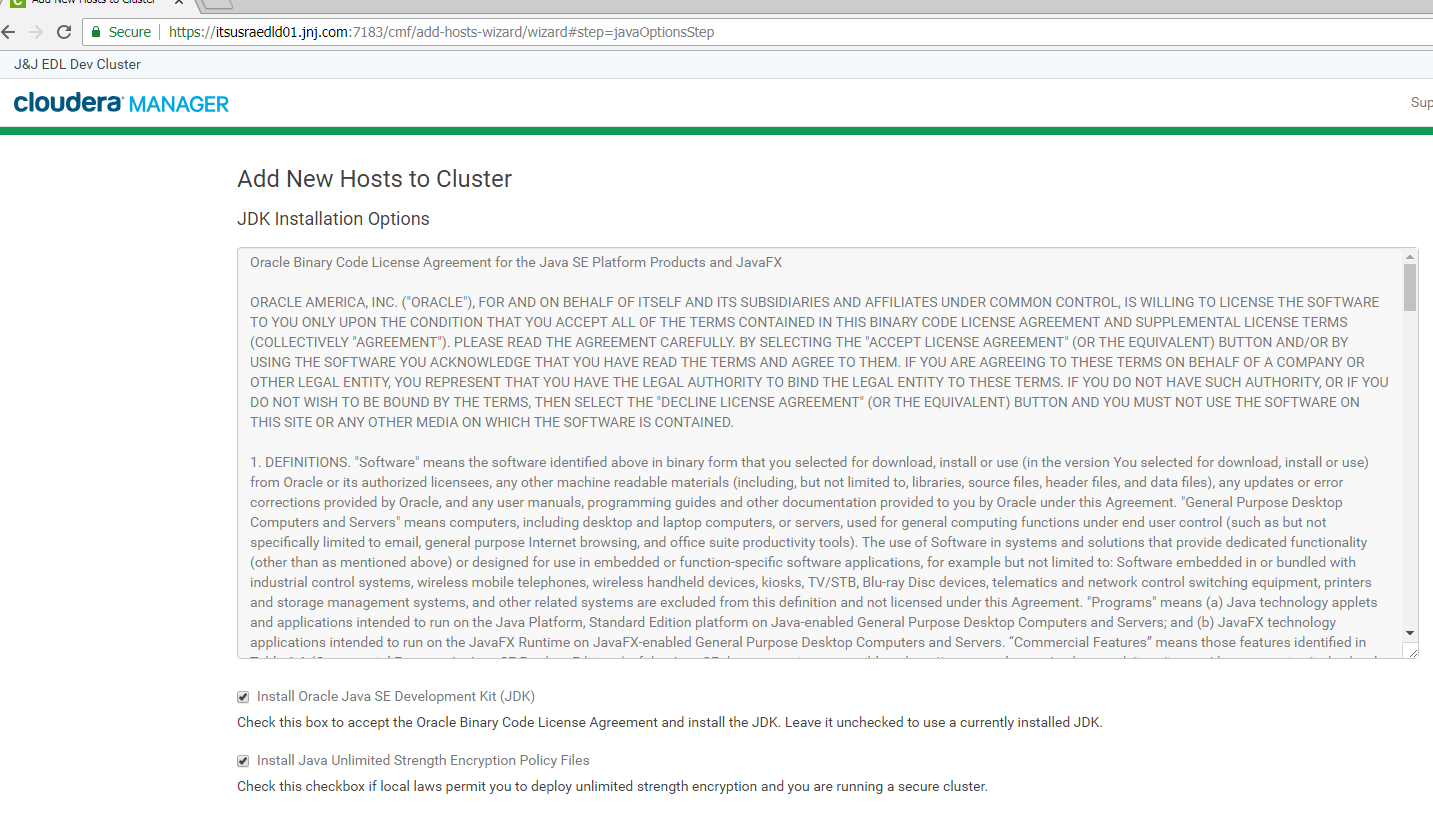
* + 1. Place the “servername” and hit “search”



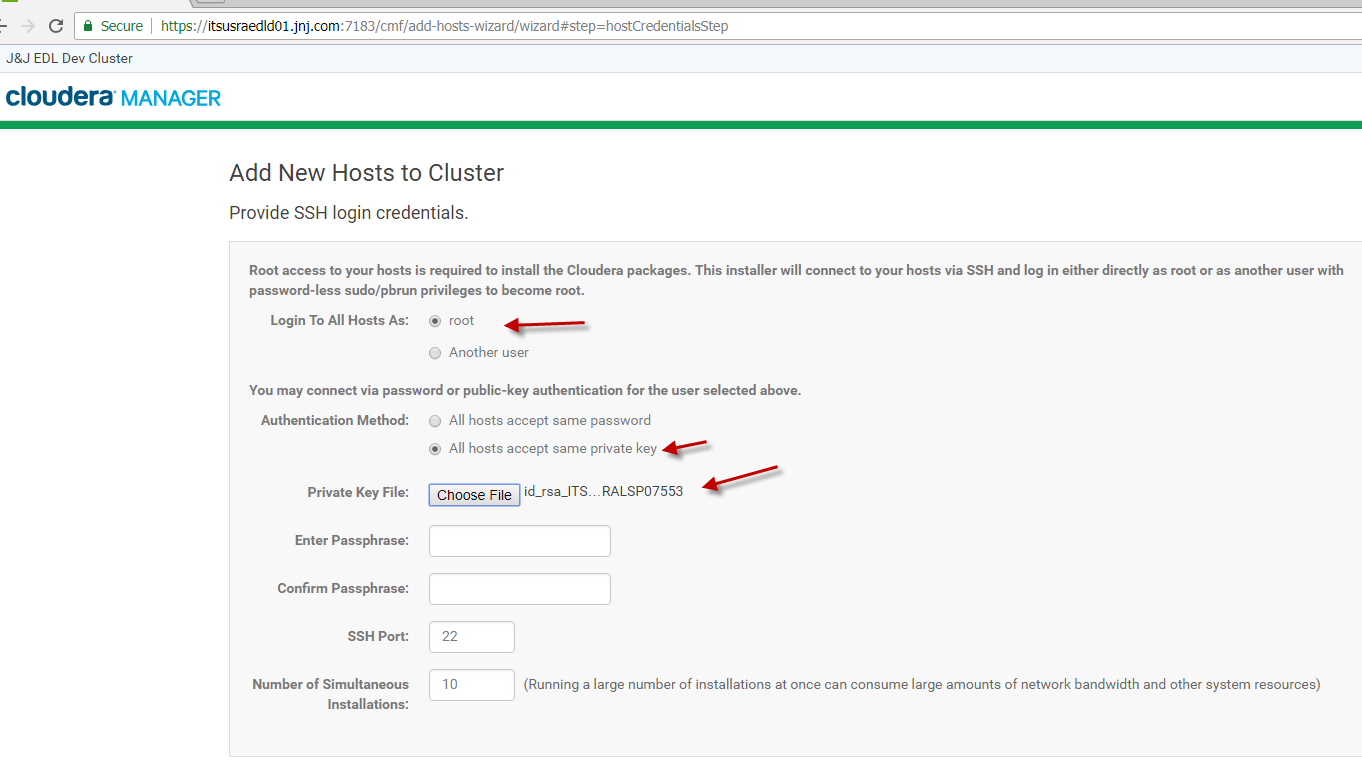
* + 1. pick up the servername. Hit “continue”







“ftp” the “id\_rsa” that has been generated for this server to your local system. Choose the below shown options and upload the ftpd “id\_rsa”. Hit “continue



Once the installation is done , it will ask to continue, Click on continue.

Now the parcel will be placed.

Once the parcels are pushed for this one server, hit “continue”. Ignore the errors for the other servers.

Next page would be for the “Host Inspector” and run the below commands on the new server to fix some of the issues.

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

echo 1 > /proc/sys/vm/swappiness

selinuxenabled || echo "disabled"

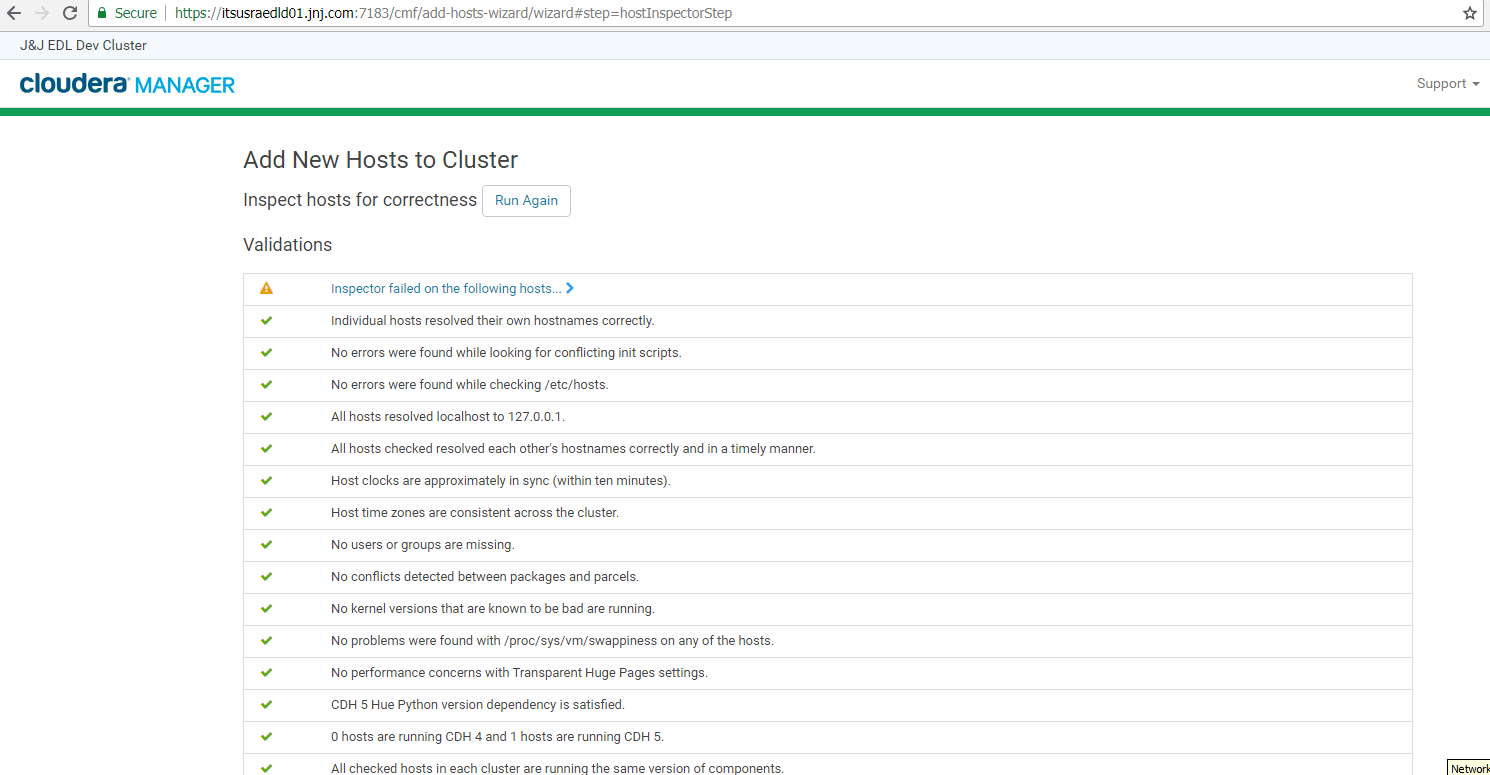
chkconfig cups off

chkconfig postfix off

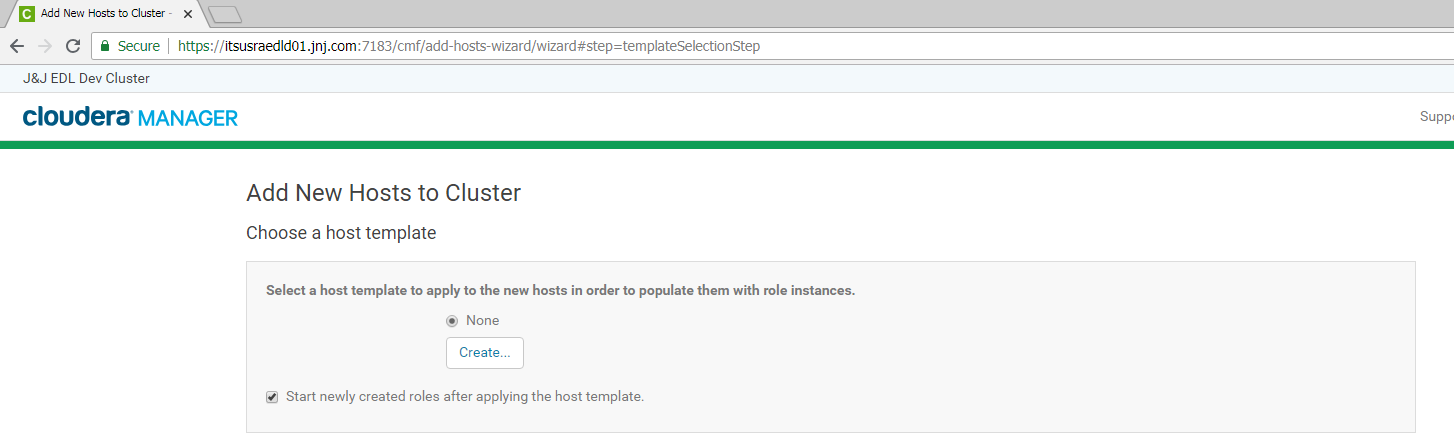
echo never > /sys/kernel/mm/transparent\_hugepage/defrag

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

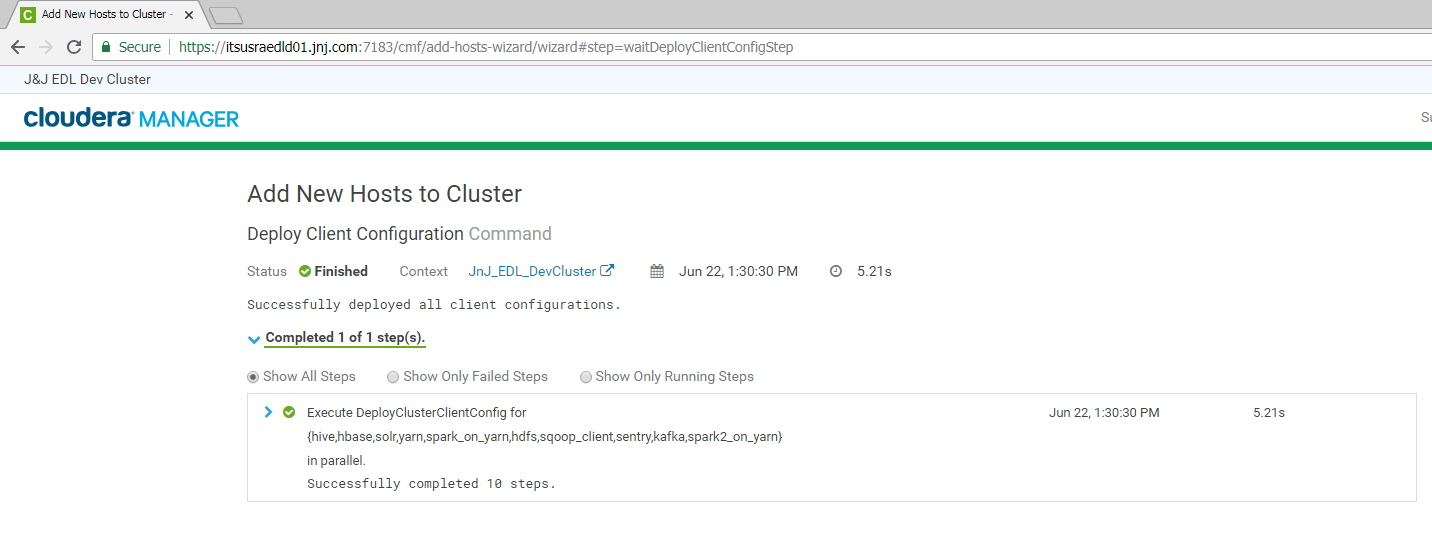
Fix/run any warnings issued by the “Host Inspector” and hit “continue”



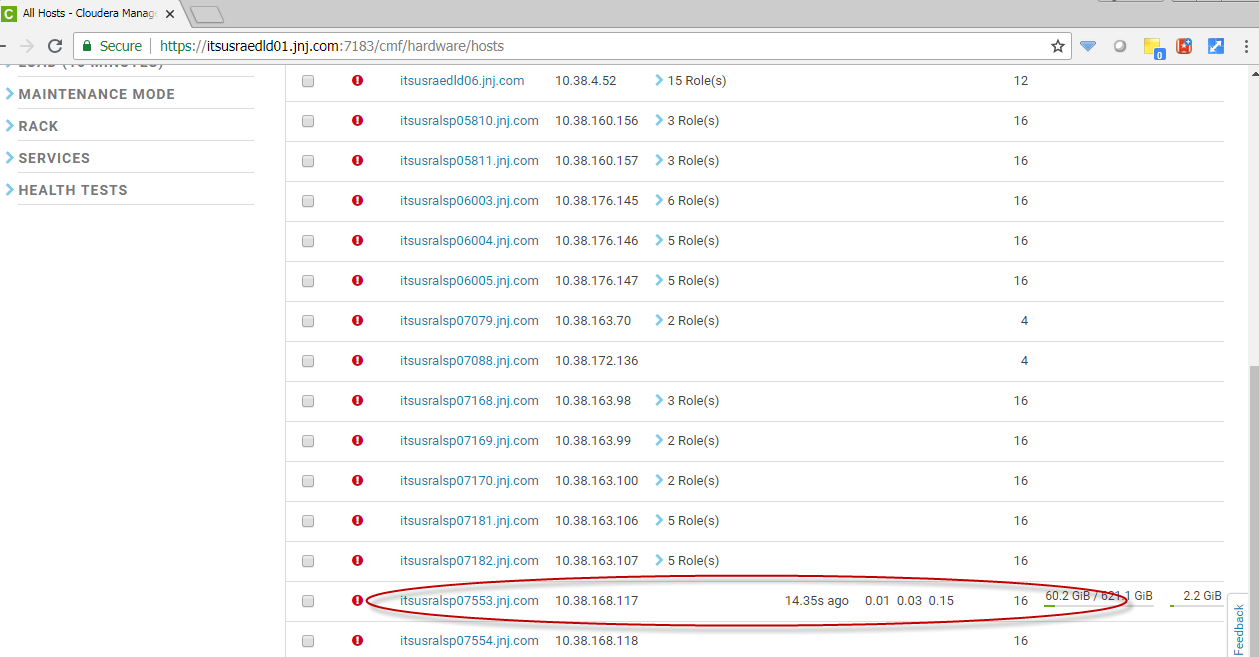
Do not select any Roles :



* Cloudera Manager will push the configs and then activity completes.



Now you should see the system heart beating to the Cloudera Manager. Other nodes will have issues as we have disabled the TLS



**10:Install right version of java and Kerberos**

Cd /usr/java

tar -xvf jdk-8u74-linux-x64.tar.gz

unlink latest; ln -s /usr/java/jdk1.8.0\_74 /usr/java/latest

Kerberos Installation :

yum install krb5-workstation

Make sure “klist” and “kinit” works

Try Klist and Kinit [asingh57@NA.JNJ.COM](mailto:asingh57@NA.JNJ.COM)

Do klist to check its working fine.

**1**1: Generating the Certificates

Make a new directory at your home directory (Ex: “/home/aomtri/7553\_servercert”) and create a file named “openssl.conf” with the below contents. Change servername as per your requirement.

And edit the file as below:

cat openssl.cnf

[ req ]

default\_bits = 2048

distinguished\_name = req\_distinguished\_name

req\_extensions = req\_ext

[ req\_distinguished\_name ]

countryName = US

stateOrProvinceName = NJ

localityName = RARITAN

organizationName = JNJ

commonName = ITSUSRALSP07674

[ req\_ext ]

subjectAltName = @alt\_names

[alt\_names]

DNS.1 = ITSUSRALSP07674

DNS.2   = ITSUSRALSP07674.jnj.com

DNS.3   = itsusralsp07674.jnj.com

[root@ ITSUSRALSP07674:/home/aomtri/7553\_servercert]#

Execute the below command

openssl req -new -newkey rsa:2048 -nodes -out ITSUSRALSP07674\_jnj\_com.txt -keyout ITSUSRALSP07674\_jnj\_com.key -subj "/C=US/ST=NJ/L=Raritan/O=JNJ/OU=EDL/CN=ITSUSRALSP07674.jnj.com" -config /home/aomtri/7553\_servercert/openssl.cnf

* Copy the contents of the “.txt” file

For Example:

#-> cat ITSUSRALSP07674\_jnj\_com.txt

-----BEGIN CERTIFICATE REQUEST-----

MIIC5DCCAcwCAPRODwajELMAkGA1UEBhMCVVMxCzAJBgNVBAgMAk5KMRAwDgYDVQQHDAdSYXJpdGFuMQwwCgYDVQQKDANKTkoxDDAKBgNVBAsMA0VETDEgMB4GA1UEAwwXaXRzdXNyYWxzcDA3NTUzLmpuai5jb20wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC5n1/PG97FLqiuFufJPFONZQXSk51r1gg6TZcP71+EgMprodEEvzCLWSQeypHqK7zN//Zx63mBnIHc/legEOh4UGzCS1zF2Oq/bpnQzcwJ9ATNnrDyD//VSn+yrwZrLSyTu6t4cocuYCuyZgSdiU2qvCZAIqs46SrF9uqGCDSNSs57BZD1cWUjaE



E6vjkefvrPqy2spymrbgSqVthUGd5OfY

-----END CERTIFICATE REQUEST-----

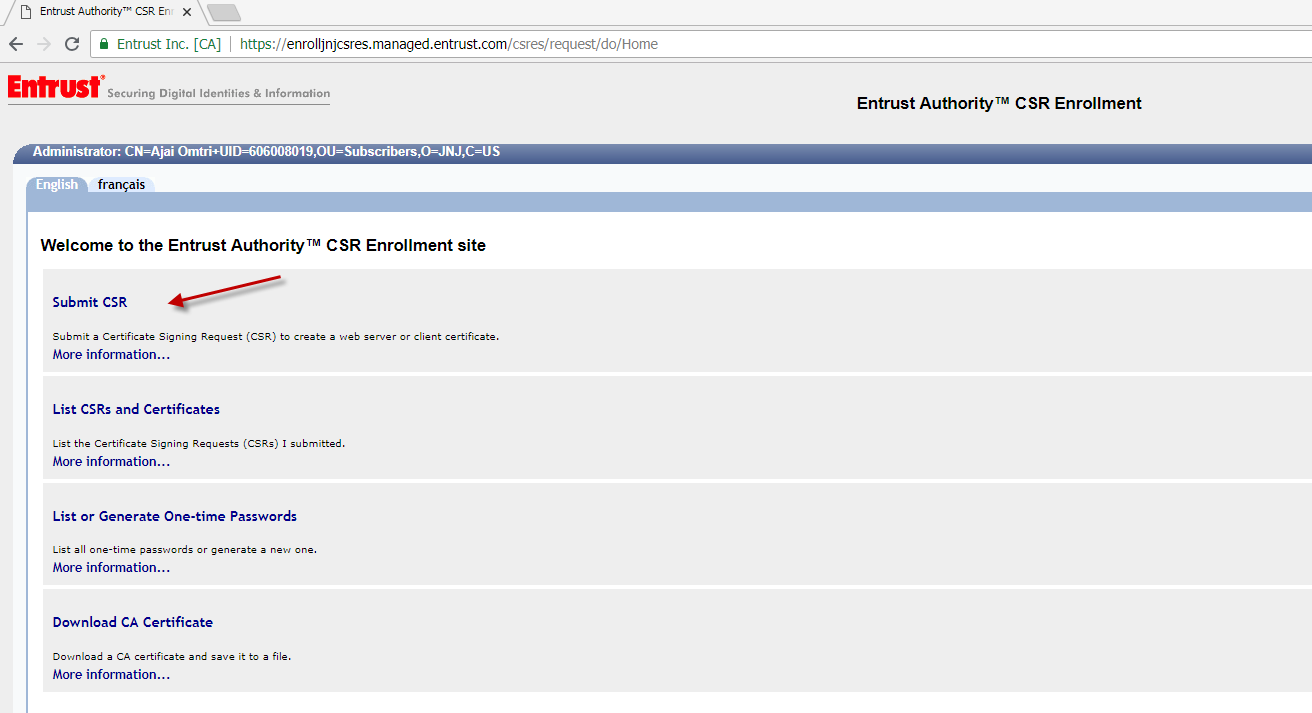
Make sure there are no Enter or space is present at the end of the file.

* Go to the URL <https://enrolljnjcsres.managed.entrust.com/csres/request/do/Home>

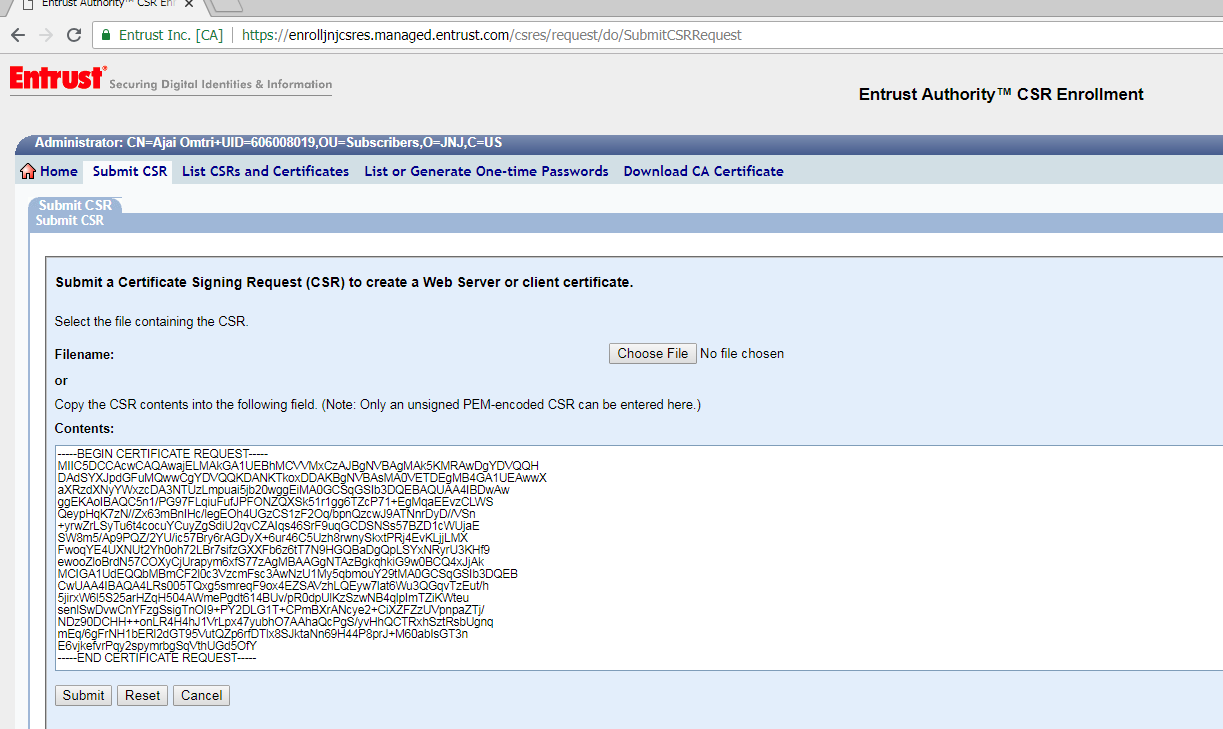
It would require user’s Entrust password. Select “Submit CSR”

And follow the step as shown below :

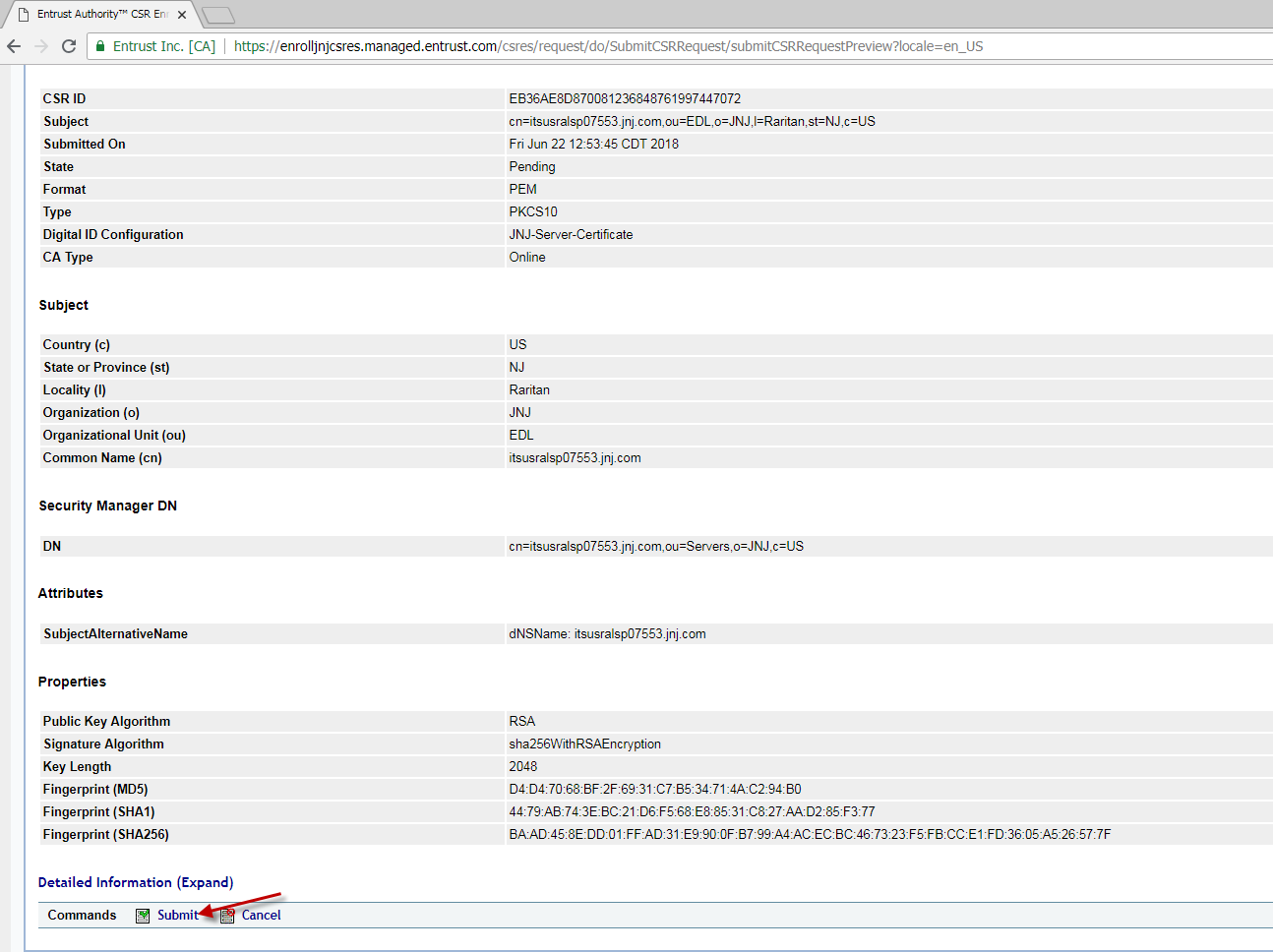
Select Submit CSR:



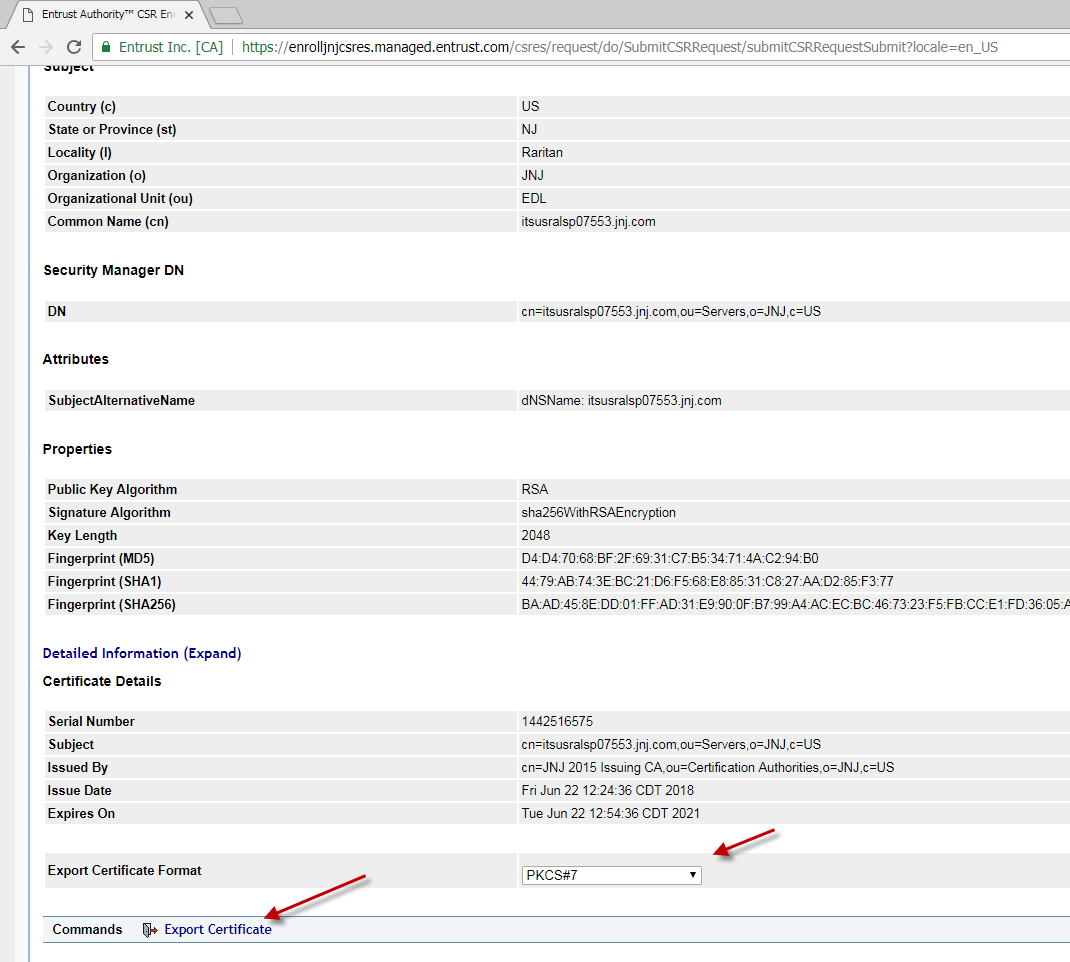
Paste the content, copied from Certificate ,shown as below :



Validate and Submit

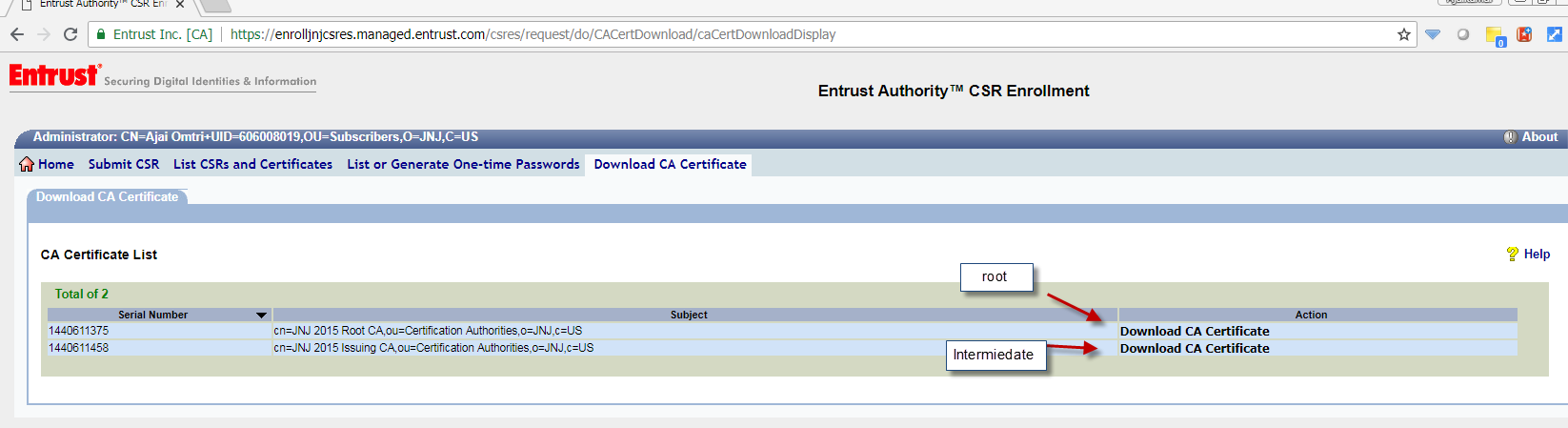


Now select the the pkcs#7 format file and Export.



Save the certificate.

Now download the “root” and “intermediate” certs from <https://enrolljnjcsres.managed.entrust.com/csres/request/do/CACertDownload/caCertDownloadDisplay>



Rename the files downloaded in the local system:

rename "JNJ 2015 Issuing CA.cer" entrustintermediate.cer

rename "JNJ 2015 Root CA.cer" entrustroot.cer

**Now “ftp” the files to the linux Servers:**

Execute the below commands to convert all 3 certificates into PEM format:

openssl pkcs7 -in EB36AE8D870081236848761997447072.p7b -inform DER -print\_certs -out ITSUSRALSP07674.pem

openssl x509 -inform der -in entrustroot.cer -out entrustroot.pem

openssl x509 -inform der -in entrustintermediate.cer -out entrustintermediate.pem

**Create a single PEM certificate with the root, issuing and server certificates:**

cat ITSUSRALSP07674.pem entrustintermediate.pem entrustroot.pem > cert-chain.pem

* **Change this PEM certificate into a PKCS12 file**

openssl pkcs12 -export -inkey ITSUSRALSP07674\_jnj\_com.key -in cert-chain.pem -out cert-chain.pkcs12

Enter the password as “Cloudera@5”

**Create a keystore from the PKCS12 file. This keystore will have the private key and the 3 certs that are required. Set the PATH for JAVA so that “keytool” is found.**

**cd /usr/java**

export PATH=/usr/java/jdk1.8.0\_74/bin:$PATH

cd /home/aomtri/7553\_servercert

keytool -importkeystore -srckeystore cert-chain.pkcs12 -srcstoretype PKCS12 -destkeystore keystore.jks

\*\* Enter the destination password as “Cloudra@5” and the source as “Cloudera@5”

**Create the “security” related folders**

#-> mkdir /opt/cloudera/security

[root@ITSUSRALSP07674:/opt/cloudera]#

#-> cd /opt/cloudera/security

[root@ITSUSRALSP07674:/opt/cloudera/security]#

#-> mkdir CAcerts

[root@ITSUSRALSP07674:/opt/cloudera/security]#

#-> mkdir jks

[root@ITSUSRALSP07674:/opt/cloudera/security]#

#-> mkdir x509

[root@ITSUSRALSP07674:/opt/cloudera/security]#

* Copy the generated key as “server.key” and “server.pem”  
   to “x509” folder

cp ITSUSRALSP07674\_jnj\_com.key /opt/cloudera/security/x509/server.key

cp ITSUSRALSP07674.pem /opt/cloudera/security/x509/server.pem

Create below password file inside x509

* Create a password file “serverkey.pw” with content “Cloudera@5”. That’s the password we gave for destination.

#-> cat serverkey.pw

Cloudera@5

* Create “combinedtrust.pem”

[root@ITSUSRALSP07674:/home/aomtri/7553\_servercert]#

#-> cat entrustroot.pem > combinedtrust.pem

[root@ITSUSRALSP07674:/home/aomtri/7553\_servercert]#

#-> cat entrustintermediate.pem >> combinedtrust.pem

* Copy the “combinedtrust.pem to “CAcerts

cp combinedtrust.pem /opt/cloudera/security/CAcerts/

* Copy the “server.keystore”

cp keystore.jks /opt/cloudera/security/jks/server.keystore

## SET File Permissions:

cd /opt/cloudera/security

cd CAcerts

chmod 644 combinedtrust.pem

cd ..

cd jks/

chmod 666 server.keystore

chown cloudera-scm:cloudera-scm server.keystore

cd ..

cd x509/

chmod 440 server.key

chmod 440 serverkey.pw

chmod 444 server.pem

**Copy trustorre file from DEV Server on all servers:**

scp /opt/cloudera/security/jks/truststore.jks abaseerm@ITSUSRALSP08797:/home/abaseerm/

cp truststore.jks /opt/cloudera/security/jks

chmod 775 /opt/cloudera/security/jks/truststore.jks

## 12: Copy the agent config file

Just copy the working copy of Cloudera-scm-agent config file from another server to this file. Here we are copying the file “/etc/cloudera-scm-agent/ config.ini” from “itsusraedld01” to “ITSUSRALSP07674”.

## 13: Enable TLS in Cloudera Manager

* Login to Cloudera Manager and go to “Administration” 🡪 “Settings and search for “Use TLS Encryption for Agents” and “Use TLS Authentication of Agents to Server” and “check” them and “save”.
* Restart the “cloudera-scm-server” and “Cloudera-scm-agent” on server and the host that we are adding.
* Now you should see the new server along with the old server heartbeating to the Cloudera Manager.

Restart the cluster.

cd /opt/cloudera/security/x509

cat server.pem ../CAcerts/combinedtrust.pem server.key > impalacombined.cert

chown impala:impala impalacombined.cert

chmod a+r impalacombined.cert