COMPLETE GUIDILINES TO KERBEROS AUTHENTICATION AND AUTHRIZATION BY SENTRY

Deployment and Configuration of Sentry Service on Cloudera Manager for Authorization using Active Directory Two-way Trust for Authentication

Prepared by:
Shyam Karale,
Sameeha and
Harshal Salunke

(SSH)

Version-01 Rev-00

Dt 21.5.21

Table of Contents:

1. Prerequisite for Security (Authentication and Authorization)

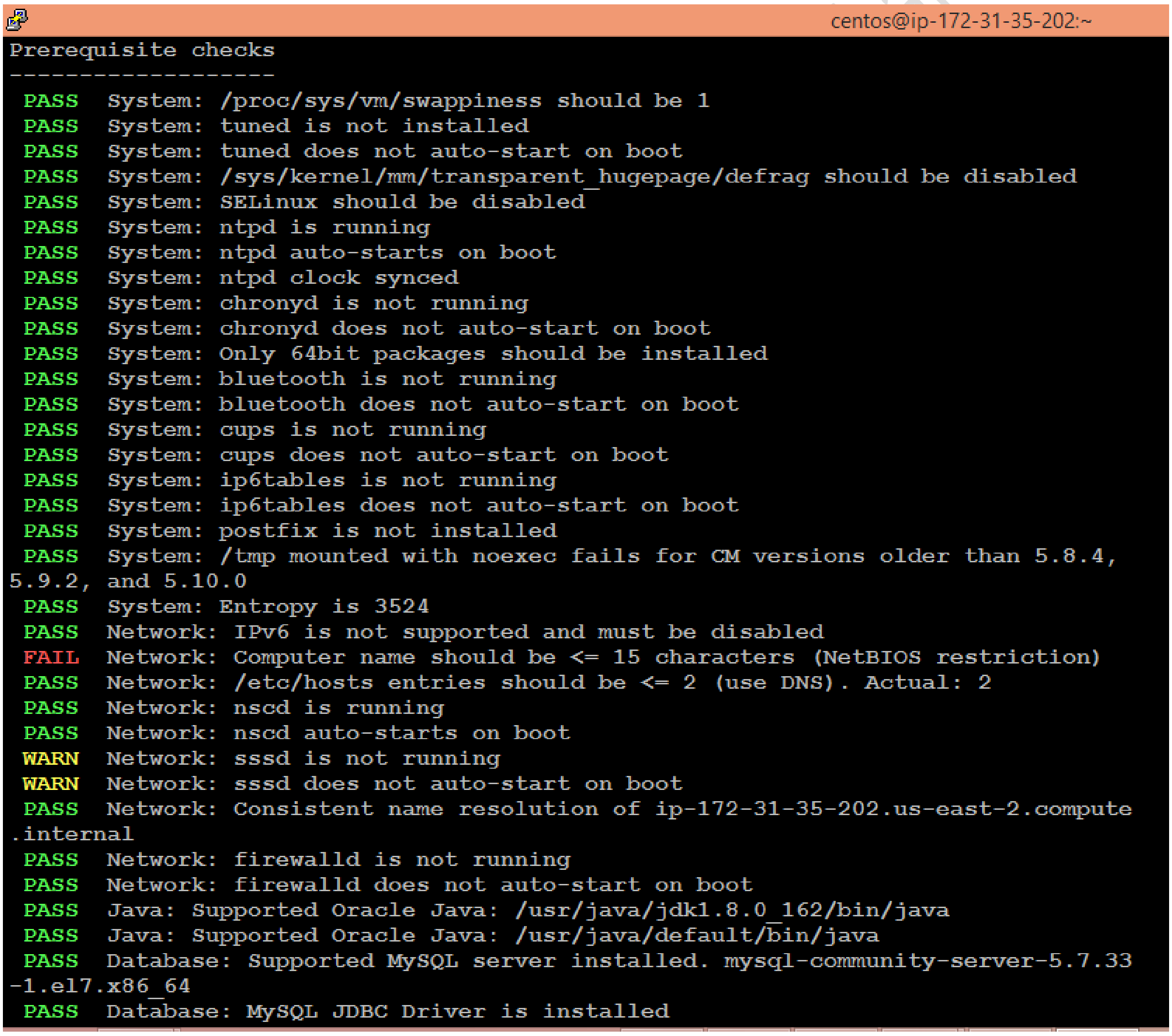
Date – 21.05.21 V1

Following are the requirements for configuring the Kerberos

1. Machine should be ready with Centos 7 Operating system virtual machine and Prerequisite done on it as per below snap

You can run below command to check this

[centos@ip-172-31-35-202 ~]$ sh check-pre-req.sh



```
Prerequisite checks
-------------------
 PASS  System: /proc/sys/vm/swappiness should be 1
 PASS  System: tuned is not installed
 PASS  System: tuned does not auto-start on boot
 PASS  System: /sys/kernel/mm/transparent_hugepage/defrag should be disabled
 PASS  System: SELinux should be disabled
 PASS  System: ntpd is running
 PASS  System: ntpd auto-starts on boot
 PASS  System: ntpd clock synced
 PASS  System: chronyd is not running
 PASS  System: chronyd does not auto-start on boot
 PASS  System: Only 64bit packages should be installed
 PASS  System: bluetooth is not running
 PASS  System: bluetooth does not auto-start on boot
 PASS  System: cups is not running
 PASS  System: cups does not auto-start on boot
 PASS  System: ip6tables is not running
 PASS  System: ip6tables does not auto-start on boot
 PASS  System: postfix is not installed
 PASS  System: /tmp mounted with noexec fails for CM versions older than 5.8.4,
5.9.2, and 5.10.0
 PASS  System: Entropy is 3524
 PASS  Network: IPv6 is not supported and must be disabled
 FAIL  Network: Computer name should be <= 15 characters (NetBIOS restriction)
 PASS  Network: /etc/hosts entries should be <= 2 (use DNS). Actual: 2
 PASS  Network: nscd is running
 PASS  Network: nscd auto-starts on boot
 WARN  Network: sssd is not running
 WARN  Network: sssd does not auto-start on boot
 PASS  Network: Consistent name resolution of ip-172-31-35-202.us-east-2.compute
.internal
 PASS  Network: firewalld is not running
 PASS  Network: firewalld does not auto-start on boot
 PASS  Java: Supported Oracle Java: /usr/java/jdk1.8.0_162/bin/java
 PASS  Java: Supported Oracle Java: /usr/java/default/bin/java
 PASS  Database: Supported MySQL server installed. mysql-community-server-5.7.33
-1.el7.x86_64
 PASS  Database: MySQL JDBC Driver is installed
```

If this is not done please use below link or below file to complete the prerequisite

PreRequisiteOnCentos7_updated march2021.pdf

2. There should be a separate Edge node machine(Instance)
   We also call it a gate way machine where we need to put some configuration file as below

   a. **clustercmd.sh** – ***This is to replicate any command on all cluster***

   b. **putnmove.sh** – ***This is to move any file to cluster***

   c. **cluster** – ***This to configure and define all machine private ip in cluster***

   d. **security_key.pem** – ***This is security key file for Authorization***

   Let’s see how we can transfer these files from our workspace to on the cluster or virtual machines

   There are 3 ways to transfer the files from your local machine to cluster

   **By PowerShell command- the eample of link is given below*

```
.\pscp.exe -P 22 -i .\security.ppk .\security.pem
centos@13.58.37.193:/home/ubuntu/.ssh
```

   **By keeping/uploading all files in in S3 and then taking files from s3 to EC2 instance (Gateway machine)*

   ** By Winscp*

   # Change the ownership of (security key).pem file as per below command

```
[centos@ip-172-31-35-202 ~]$ chmod 400 secuirty.pem
```

3. The next step is to go for path B installation for Cdh cloudera manger installation cluster in GUI
   For this the step by step procedure is available on below link /or file

   scrit path b installation.txt

   Double click on the logo to open Path b installation script

4. Once the Path b installation is complete go to cloudera manager and set Gate way for HDFS and Yarn as per
   Ensure these gateways from cloudera should be deployed in edge node only

Also check cluster health if there is any notification in main window , clear it

2. Kerberos Authentication by Active Directory

1.First we need to take one **Microsoft Windows Server 2012** machine from AWS AMI.

2.Open by RDP and we have to configure or add some new server in this server.

3. Click on Server Manager('Peti pitara') .further steps are as follows.

2.1 Need to change computer name

Click on server manager->local server->computer name->system properties->change computer name to **hadoop-ad**->add->OK->**Restart Later**

2.2 Add DNS Server

Click on Dashboard->Add Roles and features->Next->Next->server roles->Click on **DNS Server**->Add features->continue->next->Install.

Do Restart (we added DNS server so that our application get connected to server)

Reconnect to AD

2.3 Active Directory Domain Service

Click on Dashboard->Add Roles and features->Next->Next->server roles->Click on Active Directory Domain service->Add features->Next….->Install

Go to notification(Exclamation mark!)->click promote server to domain controller->

Deployment configuration window get open->Select add a new forest->Root domain name->**hadoopsecurity.local->**next->give password and confirm->next->install.

Prerequisite check (It will automatically get restart)

Reconnect to AD,Now our AD FQDN=hadoop-ad.hadoopsecurity.local

2.4 Active Directory Certificate Services

Click on Dashboard->Add Roles and features->Next->Next->server roles->Click on Active Directory Domain service->Add features->Next….->Install….->Click on 'Active directory certificate services on the destination server'->Then ADCS Configuration will open->Select Certificate Authority->**Enterprise CA->**next->Root CA->Create new private key->Next->Configure.

Restart AD machine

Reconnect to AD.

2.5 Organization Unit Creation(ou)

Go to tools->Active Directory Users and Computer->Right click on hadoopsecurity.local->New->ou->enter hadoop.

2.6 User Creation

Right click on hadoop(ou)->new->user->first name(cloudera)->last name(Managers)->user logon name(cm)->next->password->Confirm->next. **[Fig.2.6.2]**

There are four options below password setting , select password never expired option**[Fig.2.6.3]**

Right click on hadoopsecurity.local->Delegate control->next->Add->cm(Check name)->

OK->next->click for modify and create->Finish.**[Fig.2.6.4]**

If you have an AD server running and it's showing as green then you've completed this step.**[Fig.2.6.1]**

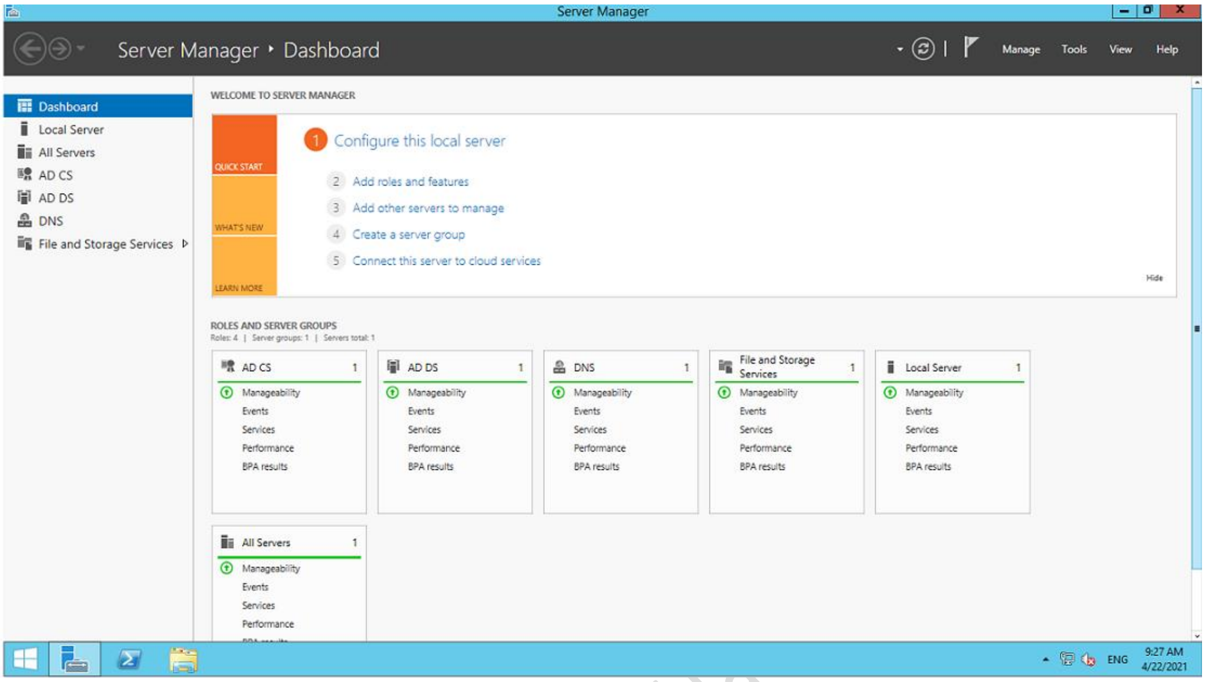
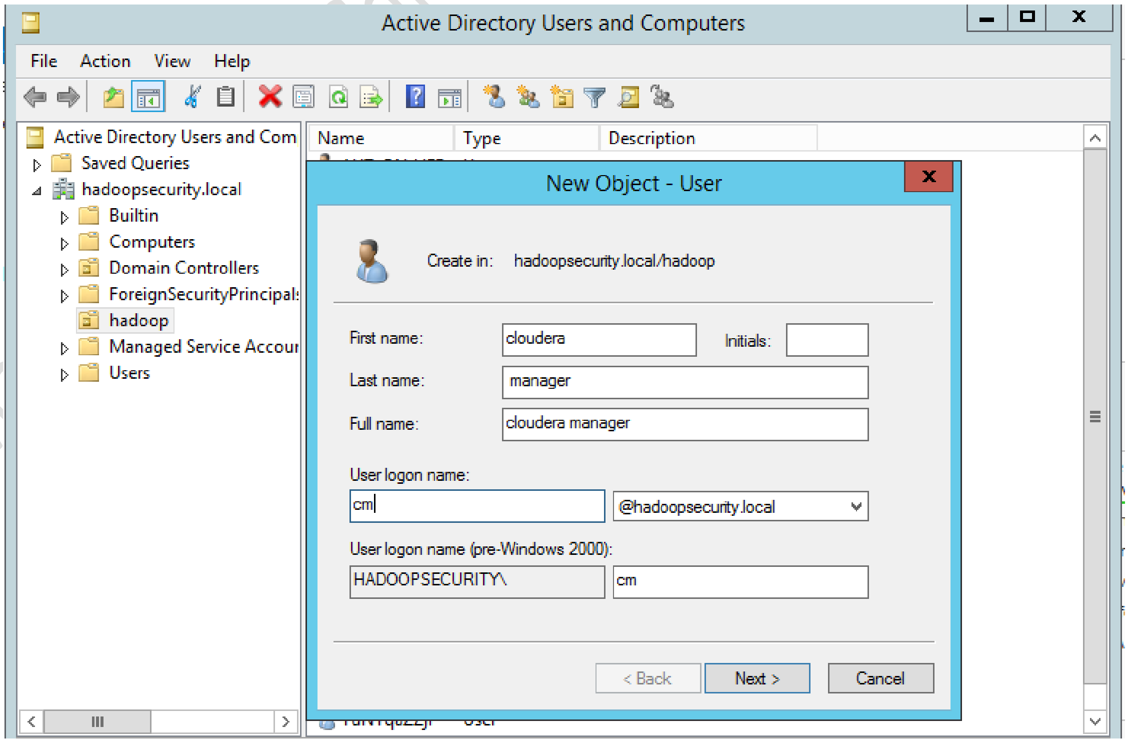


Fig.2.6.1.Server Manager



Create in: hadoopsecurity.local/hadoop

First name: cloudera

Initials:

Last name: manager

Full name: cloudera manager

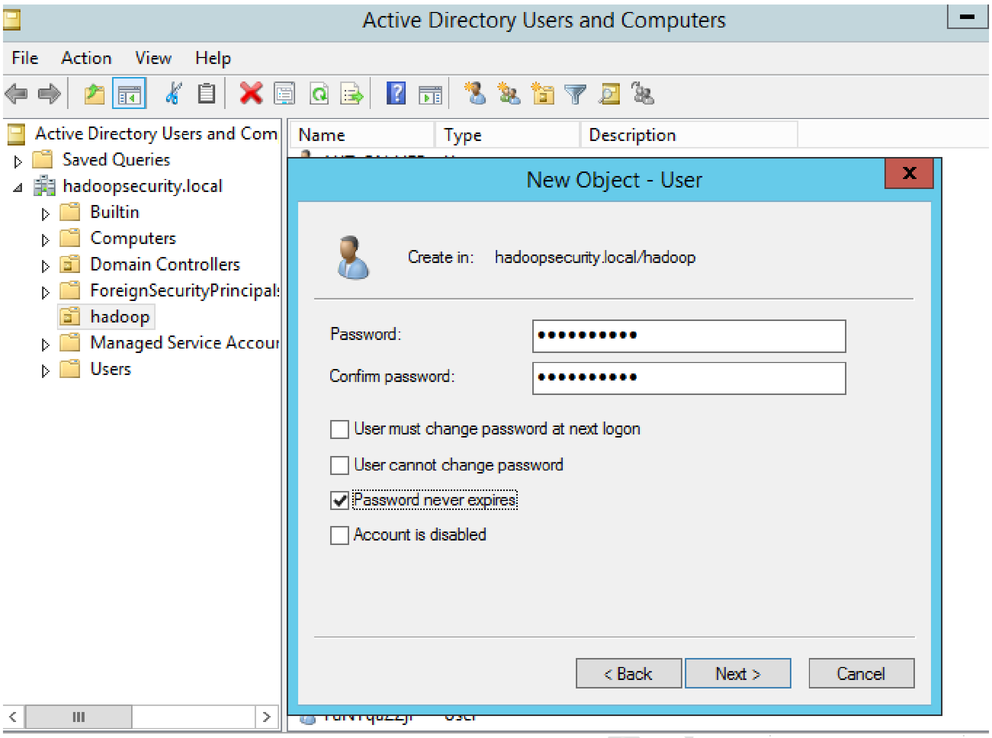
User logon name:

cm @hadoopsecurity.local

User logon name (pre-Windows 2000):

HADOOPSECURITY\ cm

Fig.2.6.2Creating User cm



New Object - User

Create in: hadoopsecurity.local/hadoop

Password: ●●●●●●●●●●

Confirm password: ●●●●●●●●●●

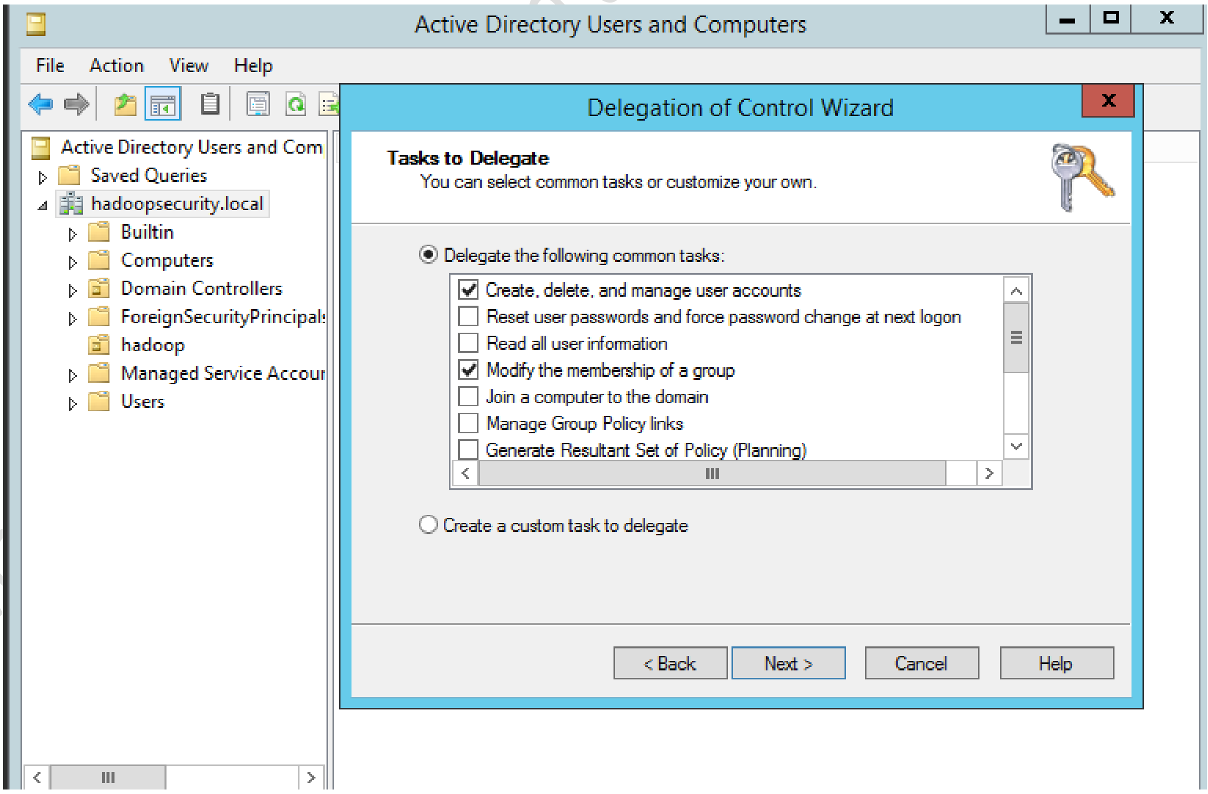
□ User must change password at next logon

□ User cannot change password

☑ Password never expires

□ Account is disabled

Fig.2.6.3 Password



Delegation of Control Wizard

Tasks to Delegate

You can select common tasks or customize your own.

● Delegate the following common tasks:

☑ Create, delete, and manage user accounts

□ Reset user passwords and force password change at next logon

□ Read all user information

☑ Modify the membership of a group

□ Join a computer to the domain

□ Manage Group Policy links

□ Generate Resultant Set of Policy (Planning)

○ Create a custom task to delegate

Fig.2.6.4.Delegation

2.7 Configure /etc/hosts

```
[centos@ip-172-31-35-202 ~]$ sudo hostname gw
```

```
[centos@ip-172-31-35-202 ~]$ bash
```

```
[centos@gw ~]$ sudo nano /etc/hosts
```

#add your AD machine's Private IP and FQDN Address in /etc/host and send it every node like below example

```
172.31.24.56 hadoop-ad.hadooopsecurity.local
```

# Now copy hosts file to our parent directory by below command

```
[centos@gw ~]$ sudo cp /etc/hosts .
```

```
[centos@gw ~]$ sh putnmove.sh /etc/hosts /etc/
```

#Install openldap-clients and krb5-workstation if it's not there. These are libraries

```
[centos@gw ~]$ sh clustercmd.sh sudo yum install openldap-clients -y
```

```
[centos@gw ~]$ sh clustercmd.sh sudo yum install krb5-workstation -y
```

#Configure /etc/krb5.conf for AD

```
[centos@gw ~]$ sudo nano /etc/krb5.conf
```

#paste this below configuration in krb5.conf And make sure that encryption types are different for AD and MIT Kerberos.

Check realm , kdc name,kdc server name

```
[libdefaults]

 default_realm = HADOOPSECURITY.LOCAL

 dns_lookup_realm = false

 dns_lookup_kdc = false

 ticket_lifetime = 24h

 renew_lifetime = 7d

 forwardable = true
```

```
default_tgs_enctypes = rc4-hmac aes128-cts aes256-cts des-cbc-crc des-cbc-md5

 default_tkt_enctypes = rc4-hmac aes128-cts aes256-cts des-cbc-crc des-cbc-md5

 permitted_enctypes = rc4-hmac aes128-cts aes256-cts des-cbc-crc des-cbc-md5


[realms]

 HADOOPSECURITY.LOCAL = {

  kdc = hadoop-ad.hadoopsecurity.local

  admin_server = hadoop-ad.hadoopsecurity.local

  max_renewable_life = 7d

 }
```

#Then copy krb5.conf to every node in the cluster

```
[centos@gw ~]$ cp /etc/krb5.conf  .

[centos@gw ~]$ sh putnmove.sh krb5.conf /etc/

[centos@gw ~]$ kinit cm

[centos@gw ~]$ klist
```

#Try an openssl connection

```
[centos@gw ~] openssl s_client -connect hadoop-ad.hadoopsecurity.local:636
```

Cloudera Manager enable kerberos wizard

2.8 Administration->security->Enable kerberos

check all 4 checkboxes

Select Active Directory

#DO copy realm ,hostname from /etc/krb5.conf which is a good practice

- kdc server host=hadoop-ad.hadoopsecurity.local
- Kerberos security realm=HADOOPSECURITY.LOCAL
- encryption types from krb5.conf
- Active directory suffix= ou=hadoop,DC=hadoopsecurity,DC=local

setup KDC account

- username=cm
- password=

  Continue

In Cloudera Manager for the hue service

Hue->Configuration->search ldap->

2.9 Enable ldap authentication

set:

backend to ->desktop.auth.backend.ldapBackend

ldap_url =ldaps://hadoop-ad.hadoopsecurity.local

start_tls= checked

create LDAP users on login= checked

LDAP search base = ou=hadoop,dc=hadoopsecurity,dc=local

LDAP bind user =cm

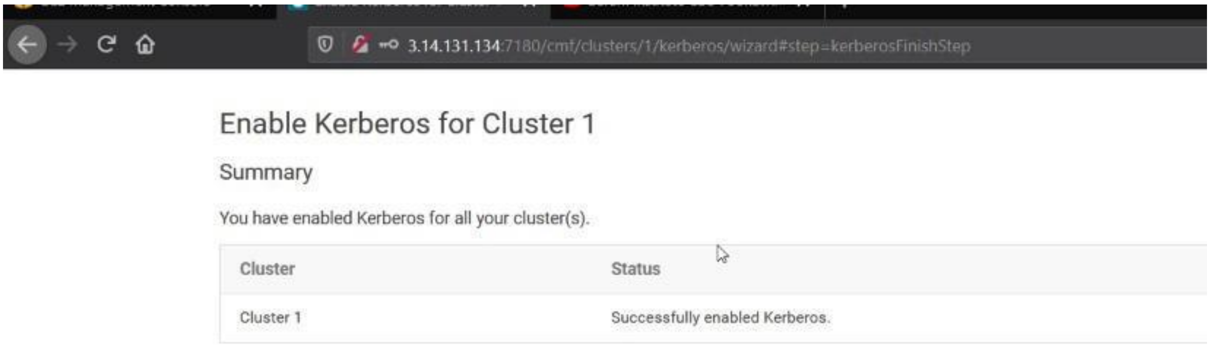
LDAP bind password = #your password for cm

2.10 Set Domain

Hue->Configuration->search domain->

hadoopsecurity.local

Stale Configurtion


3.14.131.134:7180/cmf/clusters/1/kerberos/wizard#step=kerberosFinishStep

Enable Kerberos for Cluster 1

Summary

You have enabled Kerberos for all your cluster(s).

Cluster	Status
Cluster 1	Successfully enabled Kerberos.

Fig.4.successfully enabled Kerberos

3. Authorization using Sentry

Pre requisite:

Start with an AD enabled Kerberized Cluster. Make sure that all the important files are properly configured

3.1 Need for Authorization:

To emphasize the need for authorization using a project like Apache Sentry, let’s demonstrate what is “not” provided when Sentry is absent.
Specifically, to show that even though we have authentication with Kerberos, we do not have any authorization.
For example, after authentication any user has complete admin access over databases and tables in the SQL interfaces (hive and impala). We need to restrict that access.

On AD:

Server Manager > Tools > Active Directory Users and Computers > Add user > Fill in details and logon name as jinga

On GW:

```
[centos@gw ~]$ sh clustercmd.sh sudo useradd jinga
[centos@gw ~]$sudo su
[root@gw ~]$  su jinga
[jinga@gw ~]$ kinit jinga
password
[jinga@gw ~]$ beeline
beeline> !connect jdbc:hive2://ip-172-31-35-202.ec2.internal:10000/default;principal=hive/ip-172-31-35-202.ec2.internal@HADOOPSECURITY.LOCAL
```

(Note: This IP address is of the machine which has HiveServer2(HS2) in its roles, in this case GW.)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create database database1;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal > drop database database1;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal > ^z
[jinga@gw ~]$ exit
```

However, jinga shouldn’t have any such privileges and it should lie with admin. We need a framework for authorization that works not just for hive and impala but also for all of hdfs.

3.2 Adding Sentry to Cluster:

On GW (Assuming all Databases are on GW):

(Creating Database for Sentry)

```
[centos@gw ~]$ mysql -u root –p
password
mysql> create database sentry DEFAULT CHARACTER SET utf8;
mysql> grant all on sentry.* TO 'sentry'@'%' IDENTIFIED BY 'P@ssw0rd';
mysql> exit;
```

Adding Sentry service to Cluster:

Cloudera Manager > Cluster 1 > Add Service > Sentry > Select machines for Sentry service and gateway > Database Setup – Add FQDN of machine where Databases are created (in this case, GW) > Complete the installation.

Dependencies for Sentry are: **HDFS** and **Zookeeper**. Both of them are on NameNode. Hence Sentry service is deployed on NameNode

3.3 Sentry Configurations:

3.3.1. Cloudera Manager > Sentry > Configuration

Admin Groups > + > admin > Save Changes

(admin is now the Master user of Sentry which can manage all the users and permissions in cluster.)

3.3.2. Cloudera Manager > YARN > Configuration

In search bar type 1000 > Minimum User ID > Change it to 1 > Save Changes

(By default CentOS has min user id as 1000. Below 1000 – application users. Above 1000 – human users. By changing it to 1 we are ensuring that applications will also need permissions just like human users to do any job on cluster.)

3.3.3. Cloudera Manager > Hue > Configuration

In search bar type ldap > Authentication Backend >Select desktp.auth.backend.LdapBackend

LDAP URL > ldaps://hadoop-ad.hadoopsecurity.local

LDAP Search Base > dc=hadoop-ad,dc=hadoopsecurity,dc=local

LDAP Bind User > cm (or cm/admin, whichever was created and used while deploying AD)

LDAP Bind Password > (which was given while creating user)

(LDAP mechanism will now be used for Authentication in Hue.)

In search bar type domain > Active Directory Domain > hadoopsecurity.local

(Domain for AD is specified)

In search bar type sentry > Sentry service > Select Sentry > Save Changes

(Now Hue will use Sentry for Authorization)

3.3.4. Cloudera Manager > Hive > Configuration

In search bar type impersonation > HiveServer2 Enable Impersonation > Untick HiveServer2 Default Group

(Since Sentry has been deployed, there is no need for any Default mechanism. Sentry will take care of impersonation)

In search bar type aux > Hive Auxiliary JARs Directory > /opt/cloudera/parcels/CDH/lib/hive/lib

(This path specifies library for Jar files which execute the jobs)

In search bar type sentry > Sentry service > Select Sentry > Save Changes
(Now Hive will use Sentry for Authorization)

Now Restart all the Stale Services – Oozie, YARN, Sentry, Hue and Hive. Restart the Cluster.

Sentry is now configured and ready to use.

3.4 Checking working of Sentry:

On AD:
(Create user hdfs)
Server Manager > Tools > Active Directory Users and Computers > Add user > Fill in details and logon name as hdfs

On GW:

```
[centos@gw ~]$  sudo su hdfs
```

```
[hdfs@gw centos ]$ cd
[hdfs@gw ~]$ klist
```

(There should be no ticket here)

```
[hdfs@gw ~]$ kinit hdfs
password
[hdfs@gw ~]$ wget https://s3.amazonaws.com/cloud-age/dataset
```

(Dataset is now brought on cluster)

```
[hdfs@gw ~]$ hdfs dfs -put dataset /user/hive/warehouse/dataset.csv
[hdfs@gw ~]$ hdfs dfs -ls /user/hive/warehouse
```

(Check ownership of dataset)

```
[hdfs@gw ~]$ hdfs dfs -chown hive:hive /user/hive/warehouse/dataset.csv
```

(Ownership transferred to hive)

To see this dataset in GUI:
Cloudera Manager > HDFS > File Browser > user/hive/warehouse
(Note: Port 50070, NameNode, will not work as we have kerberized the cluster.)

On AD:
(Create user admin)
Server Manager > Tools > Active Directory Users and Computers > Add user > Fill in details and logon name as admin
(Create users for demonstration)
Server Manager > Tools > Active Directory Users and Computers > Add user > Fill in details and logon name as user1 (Similarly create user2, user3, user4)

On GW:

```
[hdfs@gw ~]$ exit
[centos@gw ~]$ cat /etc/passwd
```

(Check if admin user is created.)

```
[centos@gw ~]$ sh clustercmd.sh sudo useradd admin
```

(User admin is created on Linux level)

```
[centos@gw ~]$ sh clustercmd.sh sudo useradd user1
```

(Users for demonstration are created on Linux level)

```
[centos@gw ~]$ sh clustercmd.sh sudo useradd user2
[centos@gw ~]$ sh clustercmd.sh sudo useradd user3
[centos@gw ~]$ sh clustercmd.sh sudo useradd user4
[centos@gw ~]$ sudo su admin
[admin@gw centos]$ cd
[admin@gw ~]$ klist
[admin@gw ~]$ kinit
[admin@gw ~]$ klist
[admin@gw ~]$ beeline
beeline> !connect jdbc:hive2://ip-10-0-0-215.us-east-2.compute.internal:10000/default;principal=hive/ip-10-0-0-215.us-east-2.compute.internal@HADOOPSECURITY.LOCAL
```

(Note: This IP address if of the machine which has HiveServer2(HS2) in its roles, in this case GW.)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role admin_role;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant all on server server1 to role admin_role;
```

(All permissions on server level are granted to admin_role)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant all on database default to role admin_role;
```

(All permissions on database level are granted to admin_role)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role admin_role to group admin;
```

(The role admin_role is now granted to user admin)

To see the above in GUI:

Cloudera Manager > Hue > Web UI > username- admin, password- which was entered in AD > databases and tables will be blank

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> show databases;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> use default;
```

(default database will be used)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> show tables;
```

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table foo(a string);
```

(A table named foo has been created)

#Refresh the Hue screen to see table foo there

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table sightex (
group1 string, group2 string, group3 string, group4 string,
group5 string, group6 string, group7 string, group8 string,
group9 string, group10 string)
row format serde 'org.apache.hadoop.hive.contrib.serde2.RegexSerDe'
with serdeproperties (
"input.regex"="(\\d*),(\\d*),(\".*\"),((\".*\")|([^,]*)),((\".*\")|([^,]*)),(\".*\")"
) stored as textfile;
```

(Another table sightex is created with specified properties)

#Refresh the Hue screen to see table sightex there

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> load data inpath
'/user/hive/warehouse/dataset.csv' into table sightex;
```

(Data is loaded into table sightex)

#Refresh the Hue screen and check in sample data that the data is there

On GUI:

Log out of admin from Hue.
Log in as user1.

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table sightings_parquet as
select group1 as sightex, group2 as reported, group3 as loc, group4 as shape, group7
as duration, group10 as description from sightex where group1 is not null;
```

(Table sightings_parquet is created by importing values from table sightex)

To see this job running in GUI:

Cloudera Manager > YARN > WebUI

(All the completed tasks are shown here)

Refresh the Hue screen to see the table sightings_parquet

#user1 is Authorized to view only sightings_parquet

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> show * from sightings_parquet limit
10;
```

(Shows 10 items form sightings_parquet, sample data)

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role analyst;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant select on table sightings_parquet
to role analyst;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role analyst to group user1;
```

#Role analyst is created and granted to user1 which can be seen on Hue Browser

Now we will check Authorization for user2:

On GUI:

Log out from user1 in Hue
Log in as user2

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create view sightings_ltd as select sightex, reported, loc, shape, duration from sightings_parquet;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role ltd_reader;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant select on sightings_ltd to role ltd_reader;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role ltd_reader to group user2;
```

(sightings_ltd is created from sightings_parquet)
(Role ltd_reader is created and granted to user2)
#This all can be seen in Hue Browser

For user3:

On GUI:

Log out from user2 in Hue
Log in as user3

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create database sightings_parquet;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table jersey as select * from sightings_parquet where loc LIKE "%NJ%";
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role nj;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant select on jersey to role nj;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role nj to group user3;
```

(User3 is given access to only data of NJ(New Jersey) from sightings_parquet)
#This all can be seen in Hue Browser

For user4:

On GUI:

Log out from user3 in Hue
Log in as user4

On GW:

```
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table washington as select * from sightings_parquet where loc LIKE "%WA%";
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role wa;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant select on washington to role wa;
```

0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role wa to group user4;
(User4 is given access to only data of WA(Washington) from sightings_parquet)
#This all can be seen in Hue Browser

For user5:

On GUI:
Log out from user4 in Hue
Log in as user5

On GW:
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create table florida as select * from sightings_parquet where loc LIKE "%FL%";
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> create role fl;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant select on florida to role fl;
0: jdbc:hive2://ip-172-31-35-202.ec2.internal> grant role fl to group user5;
(User5 is given access to only data of Fl(Florida) from sightings_parquet)
#This all can be seen in Hue Browser

While practicing, skip adding a user in CLI and another in AD and then try to access the tables from Hue and see the difference.

Try and Fail but do not fail to try

Effort is the key for success !!!

------------------End------------------