|  |
| --- |
|  |
| Apache Sentry Configuration using Cloudera |
|  |
|  |
| **Shalaj Shukla** |
| **5/22/2017** |

|  |
| --- |
|  |

Table of Contents

[The Sentry Service 2](#_Toc483229493)

[**Prerequisites** 2](#_Toc483229494)

[**Adding the Sentry Service** 3](#_Toc483229495)

[**Enabling the Sentry Service Using Cloudera Manager** 9](#_Toc483229496)

[Before Enabling the Sentry Service 9](#_Toc483229497)

[Enabling the Sentry Service for Hive 11](#_Toc483229498)

[Enabling the Sentry Service for Hue 12](#_Toc483229499)

[Specify admin group 13](#_Toc483229500)

[Role Management Commands 14](#_Toc483229501)

[Create Role 14](#_Toc483229502)

[Drop Role 14](#_Toc483229503)

[Show Current Roles 14](#_Toc483229504)

[Set Role 14](#_Toc483229505)

[SHOW Statement 15](#_Toc483229506)

[Create and Grant role 16](#_Toc483229507)

[GRANT ALL ON SERVER server1 TO ROLE admin\_role; 17](#_Toc483229508)

[Grant privileges to analyst\_role: 18](#_Toc483229509)

[Grant privileges to junior\_analyst\_role: 18](#_Toc483229510)

[Grant privileges to admin\_role: 18](#_Toc483229511)

[Grant roles to groups: 18](#_Toc483229512)

# The Sentry Service

The Sentry service is a RPC server that stores the authorization metadata in an underlying relational database and provides RPC interfaces to retrieve and manipulate privileges. It supports secure access to services using Kerberos. The service serves authorization metadata from the database backed storage; it does not handle actual privilege validation. The Hive and Impala services are clients of this service and will enforce Sentry privileges when configured to use Sentry.

The motivation behind introducing a new Sentry service is to make it easier to handle user privileges than the existing policy file approach. Providing a database instead, allows you to use the more traditional **GRANT/REVOKE**statements to modify privileges.

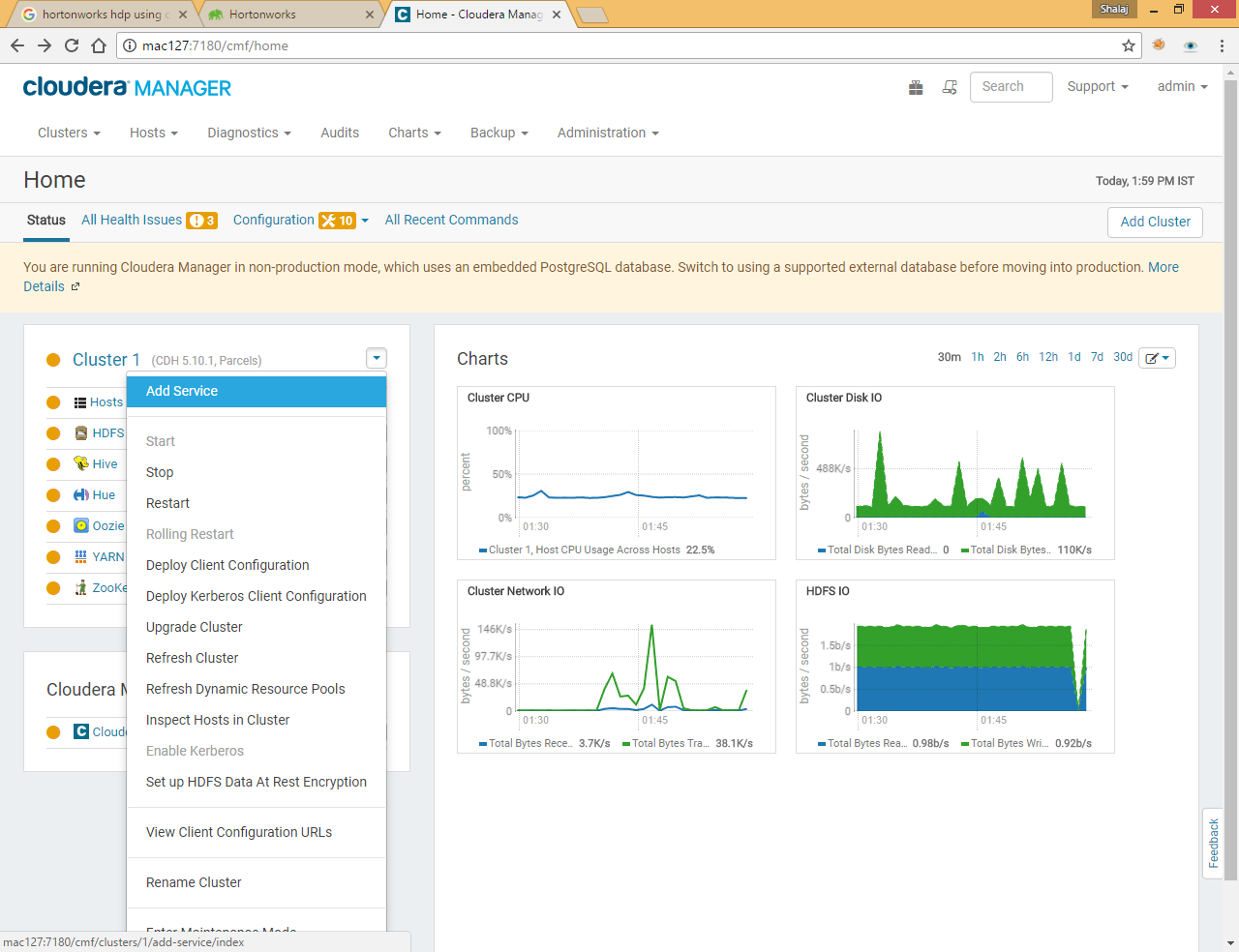
**Prerequisites**

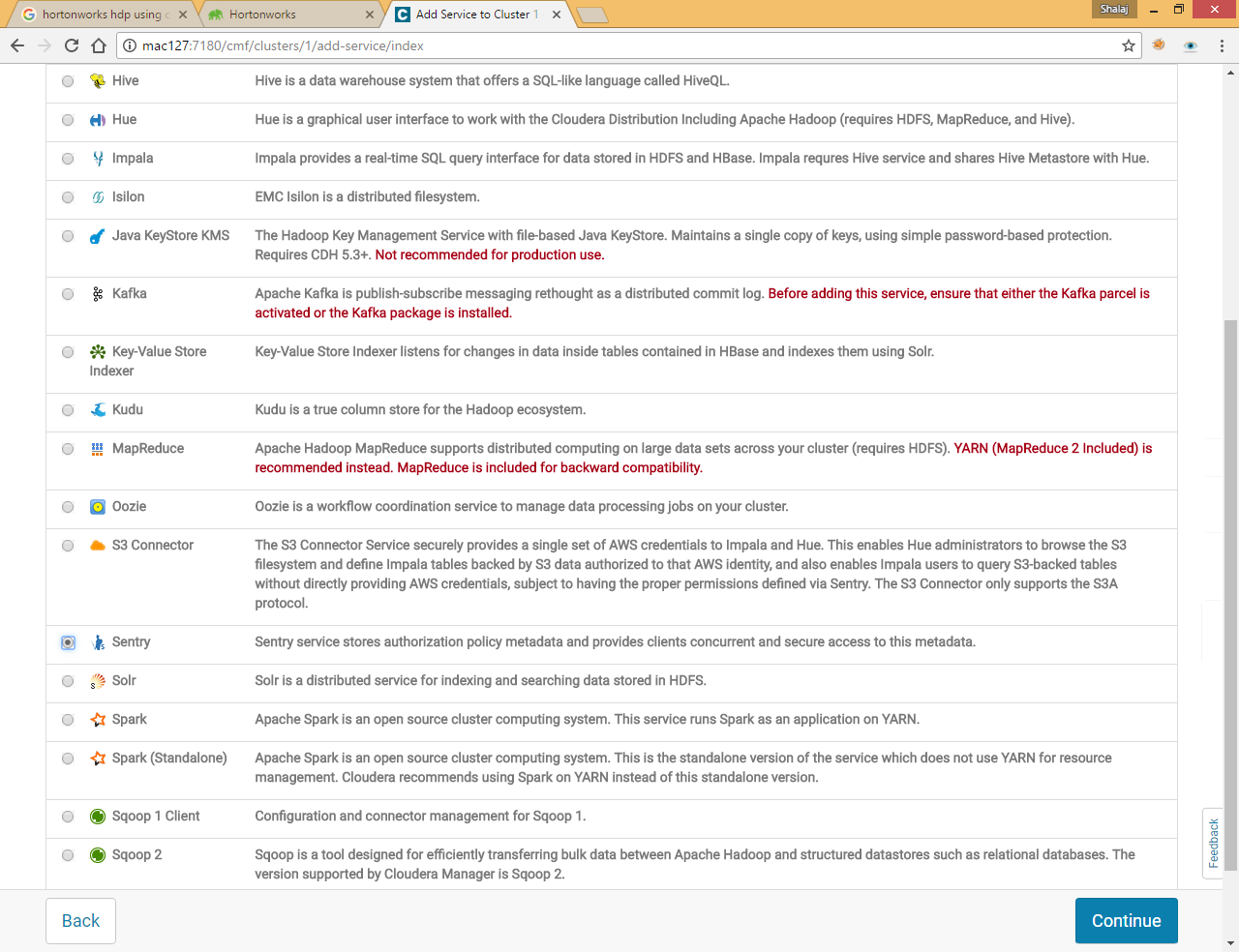
* CDH 5.1.x (or later) managed by Cloudera Manager 5.1.x (or later).
* HiveServer2 and the Hive Metastore running with strong authentication. For HiveServer2, strong authentication is either Kerberos or LDAP. For the Hive Metastore, only Kerberos is considered strong authentication
* Impala 1.4.0 (or later) running with strong authentication. With Impala, either Kerberos or LDAP can be configured to achieve strong authentication.
* **Implement Kerberos authentication on your cluster (check attached doc).**



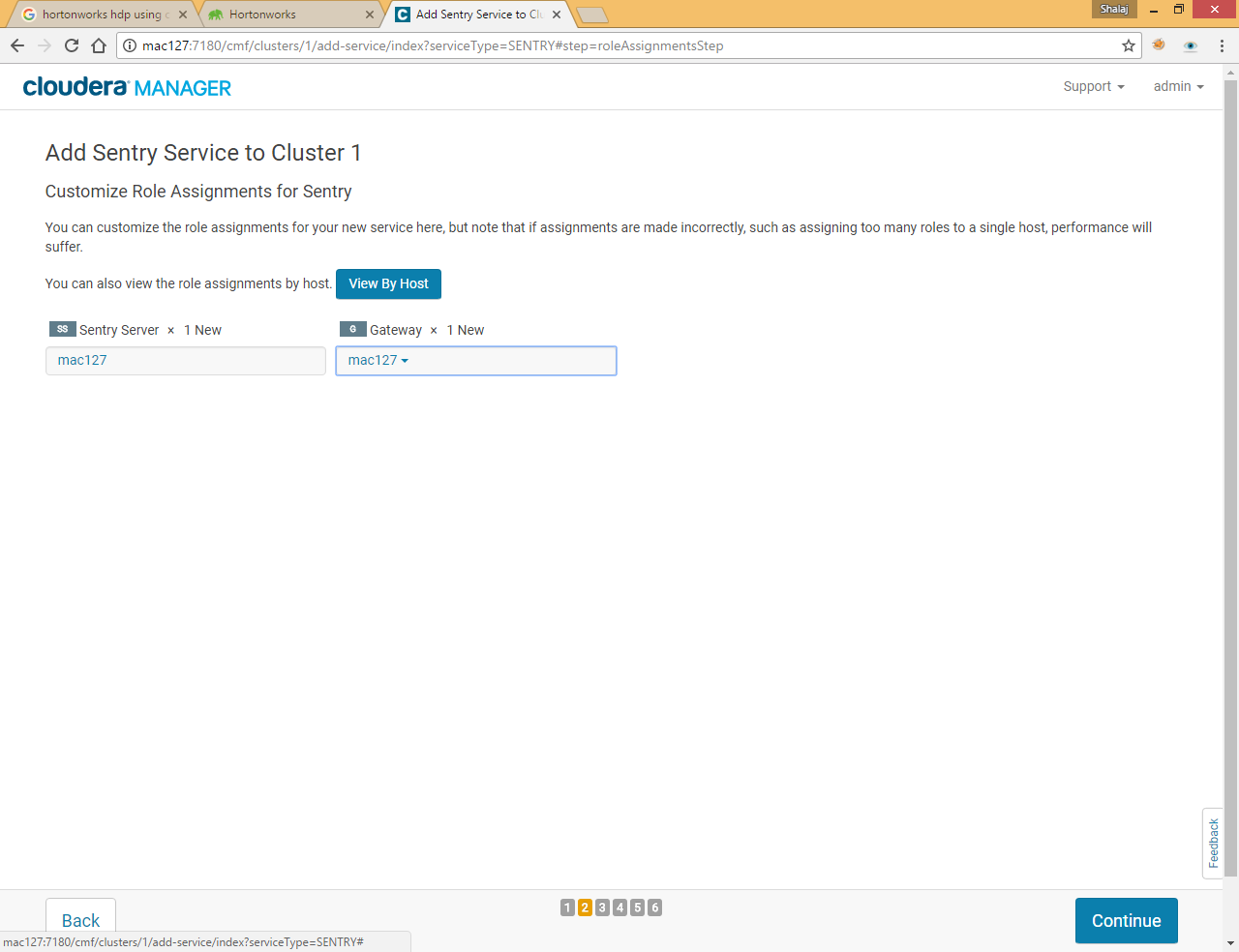
**Adding the Sentry Service**

1. On the Home page, click to the right of the cluster name and select **Add a Service**. A list of service types display. You can add one type of service at a time.

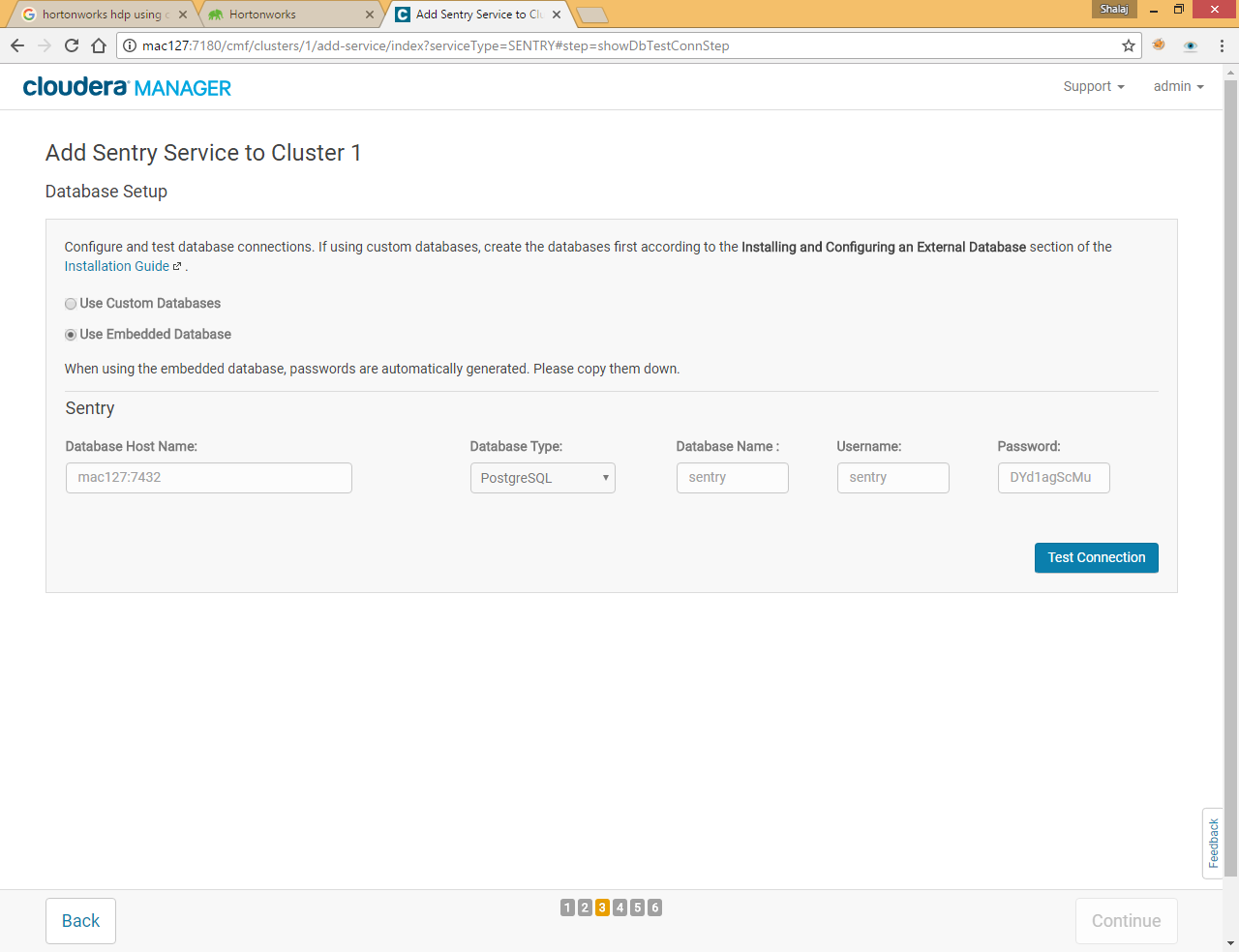




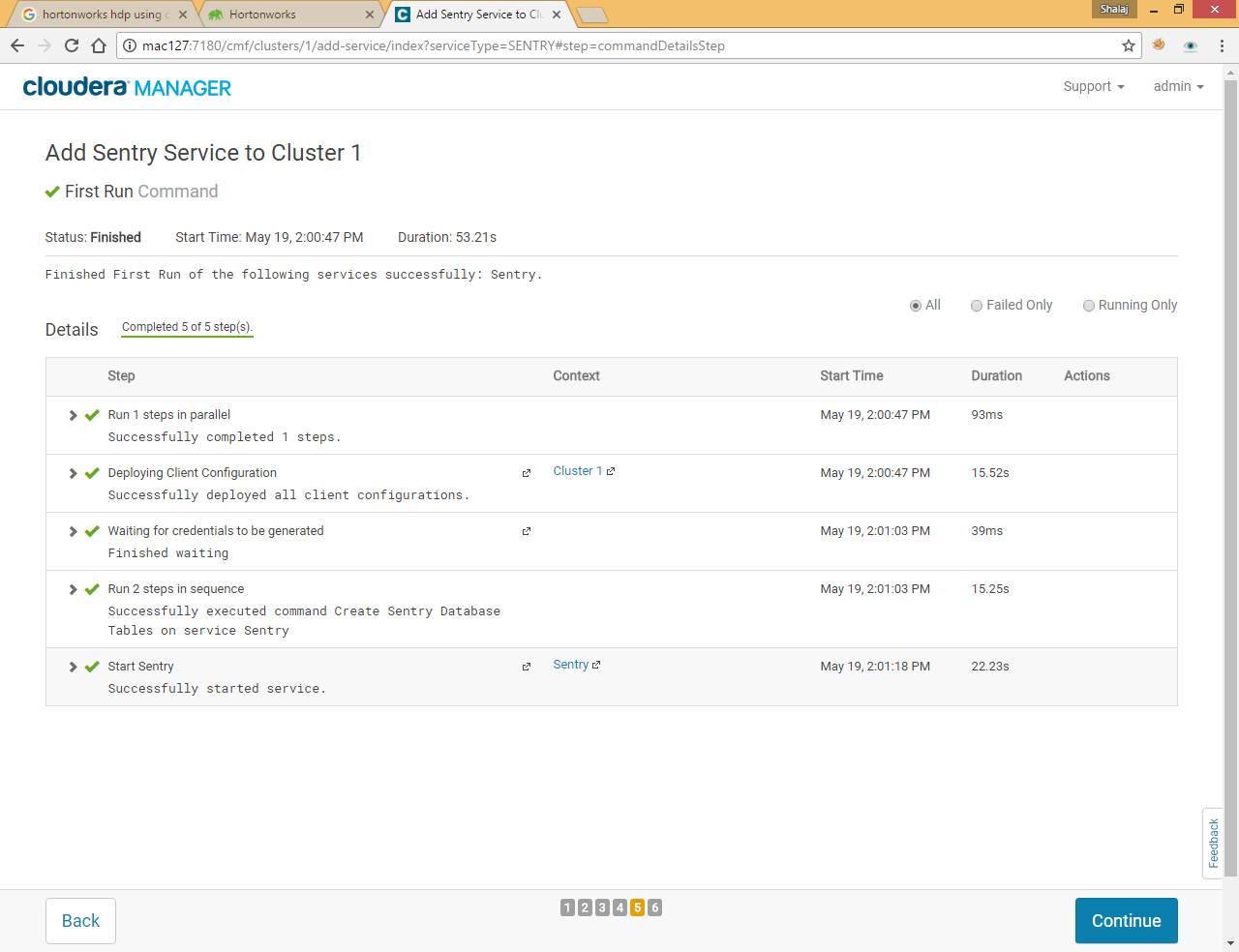
Select the **Sentry** service and click **Continue**.

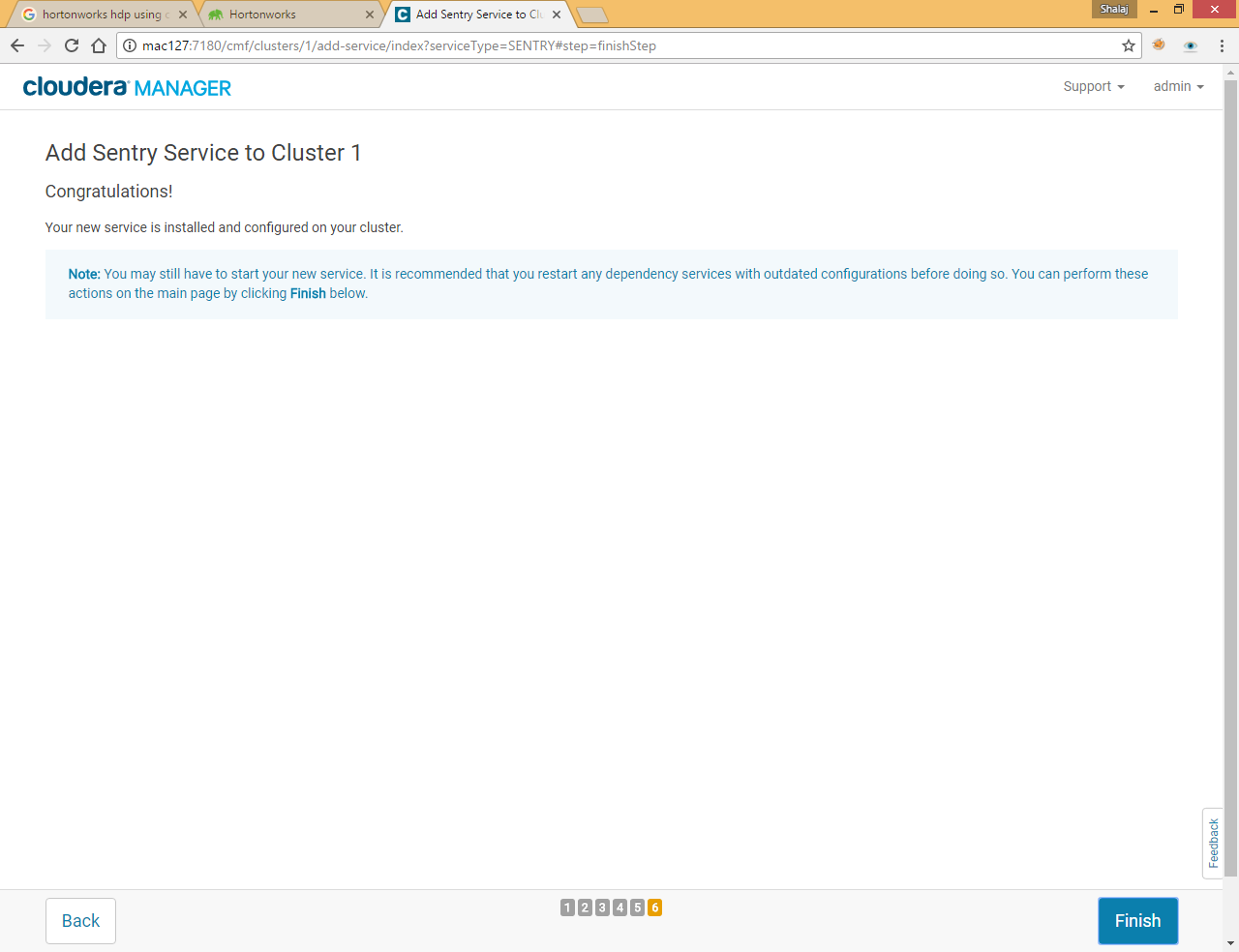


Click Continue



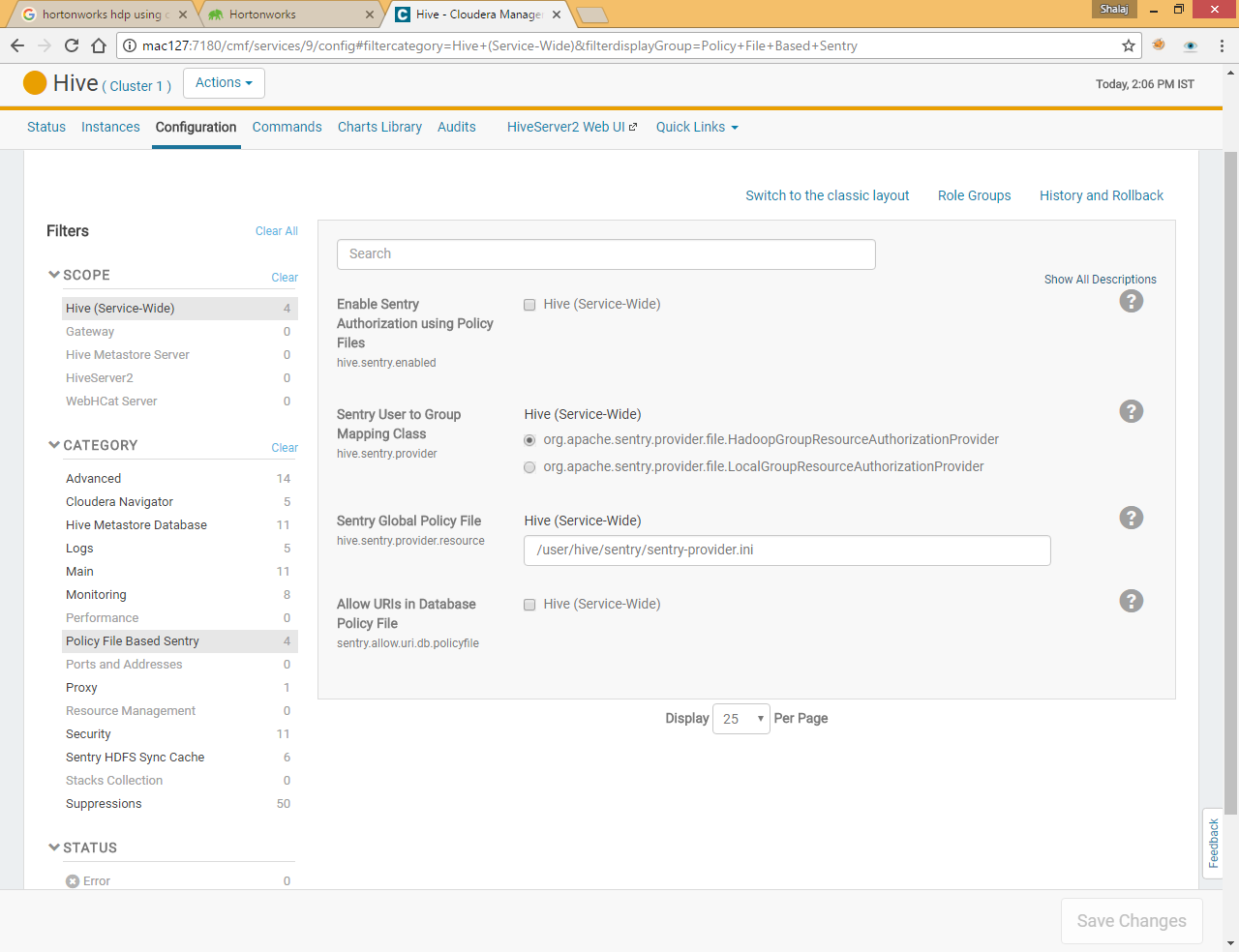
mac127:7432 sentry sentry DYd1agScMu





Disable the existing Sentry policy file for any Hive or Impala services on the cluster. To do this:

* Go to the Hive or Impala service.
* Click the **Configuration** tab.
* Select **Scope** > ***Service Name* (Service-Wide)**.
* Select **Category** > **Policy File Based Sentry**.
* Deselect **Enable Sentry Authorization using Policy Files**. Cloudera Manager will throw a validation error if you attempt to configure the Sentry service while this property is checked.
* Repeat for any remaining Hive or Impala services.



By default it is deselected

**Enabling the Sentry Service Using Cloudera Manager**

### Before Enabling the Sentry Service

* Ensure you satisfy all the **Prerequisites** for the Sentry service.
* The Hive warehouse directory (/user/hive/warehouse or any path you specify as hive.metastore.warehouse.dir in your hive-site.xml) must be owned by the Hive user and group.
* Permissions on the warehouse directory must be set as follows :
  + - **771** on the directory itself (for example, /user/hive/warehouse)
    - **771** on all subdirectories (for example, /user/hive/warehouse/mysubdir)
    - All files and subdirectories should be owned by hive:hive

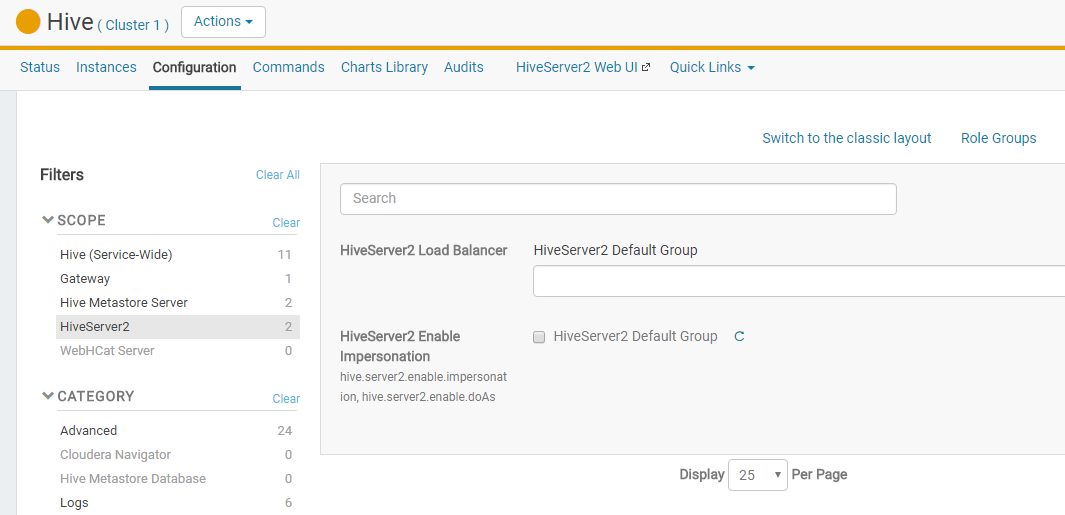
For example:

$ sudo -u hdfs hdfs dfs -chmod -R 771 /user/hive/warehouse

$ sudo -u hdfs hdfs dfs -chown -R hive:hive /user/hive/warehouse

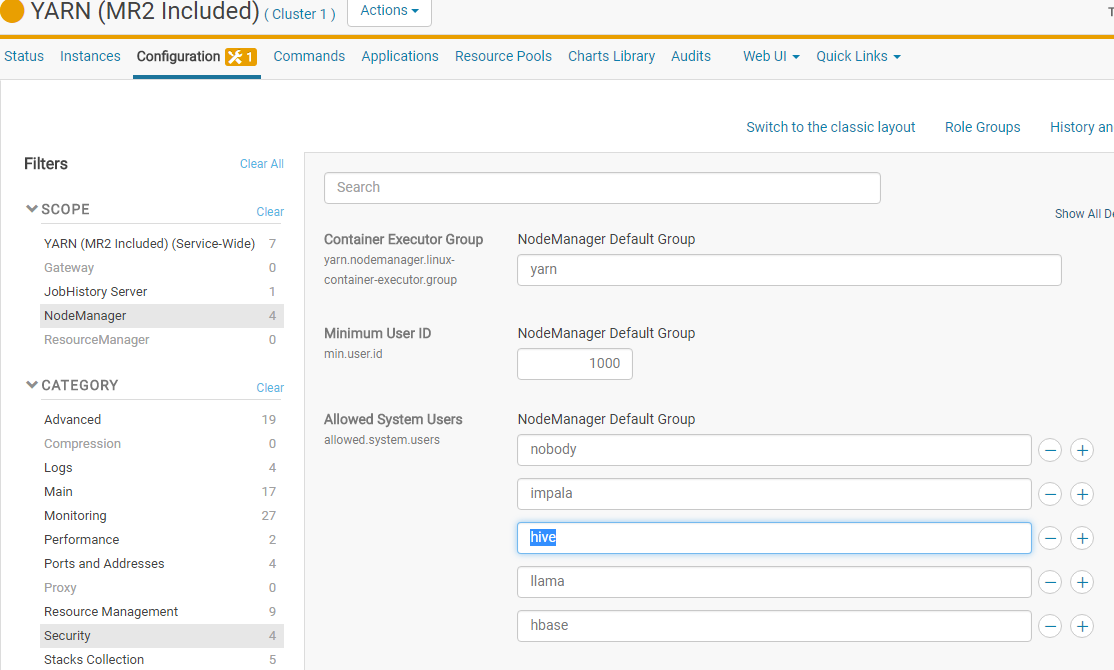
* Disable impersonation for HiveServer2 in the Cloudera Manager Admin Console:

1. Go to the Hive service.
2. Click the **Configuration** tab.
3. Select **Scope** > **HiveServer2**.
4. Select **Category** > **Main**.
5. Uncheck the **HiveServer2 Enable Impersonation** checkbox.
6. Click **Save Changes** to commit the changes.



* If you are using YARN, enable the Hive user to submit YARN jobs.

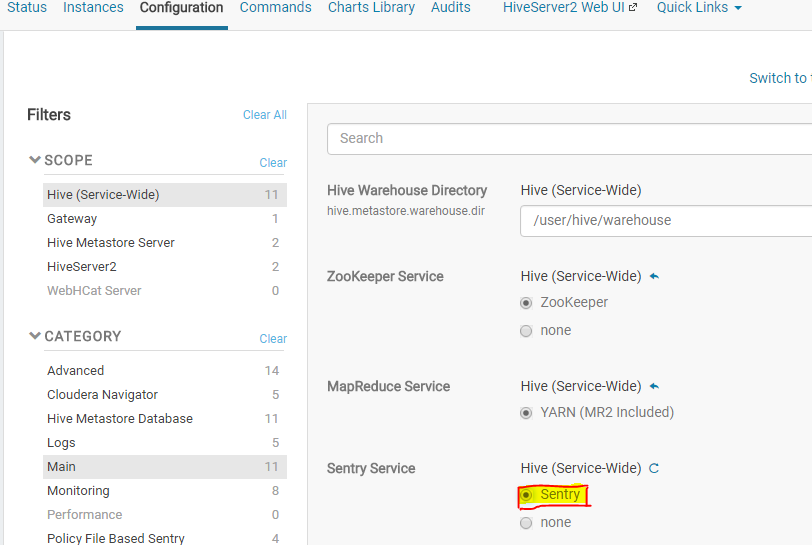
1. Open the Cloudera Manager Admin Console and go to the YARN service.
2. Click the **Configuration** tab.
3. Select **Scope** > **NodeManager**.
4. Select **Category** > **Security**.
5. Ensure the **Allowed System Users** property includes the hive user. If not, add hive.



1. Click **Save Changes** to commit the changes.
2. Repeat steps 1-6 for *every* NodeManager role group for the YARN service that is associated with Hive, if more than one exists.
3. Restart the YARN service.

### Enabling the Sentry Service for Hive

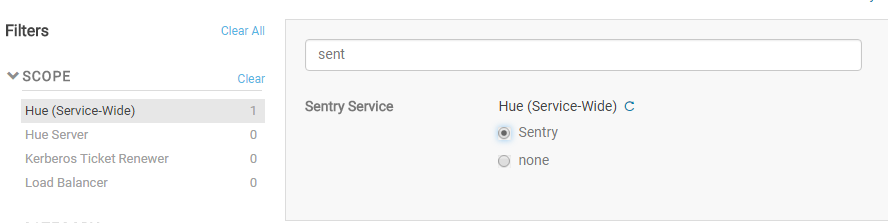
1. Go to the Hive service.
2. Click the **Configuration** tab.
3. Select **Scope** > **Hive (Service-Wide)**.
4. Select **Category** > **Main**.
5. Locate the **Sentry Service** property and select Sentry.
6. Click **Save Changes** to commit the changes.
7. Restart the Hive service.



### Enabling the Sentry Service for Hue

To interact with Sentry using Hue, enable the Sentry service as follows:

1. Enable the Sentry service for Hive and Impala (as instructed above).
2. Go to the Hue service.
3. Click the **Configuration** tab.
4. Select **Scope** > **Hue (Service-Wide)**.
5. Select **Category** > **Main**.
6. Locate the **Sentry Service** property and select Sentry.
7. Click **Save Changes** to commit the changes.
8. Restart Hue.

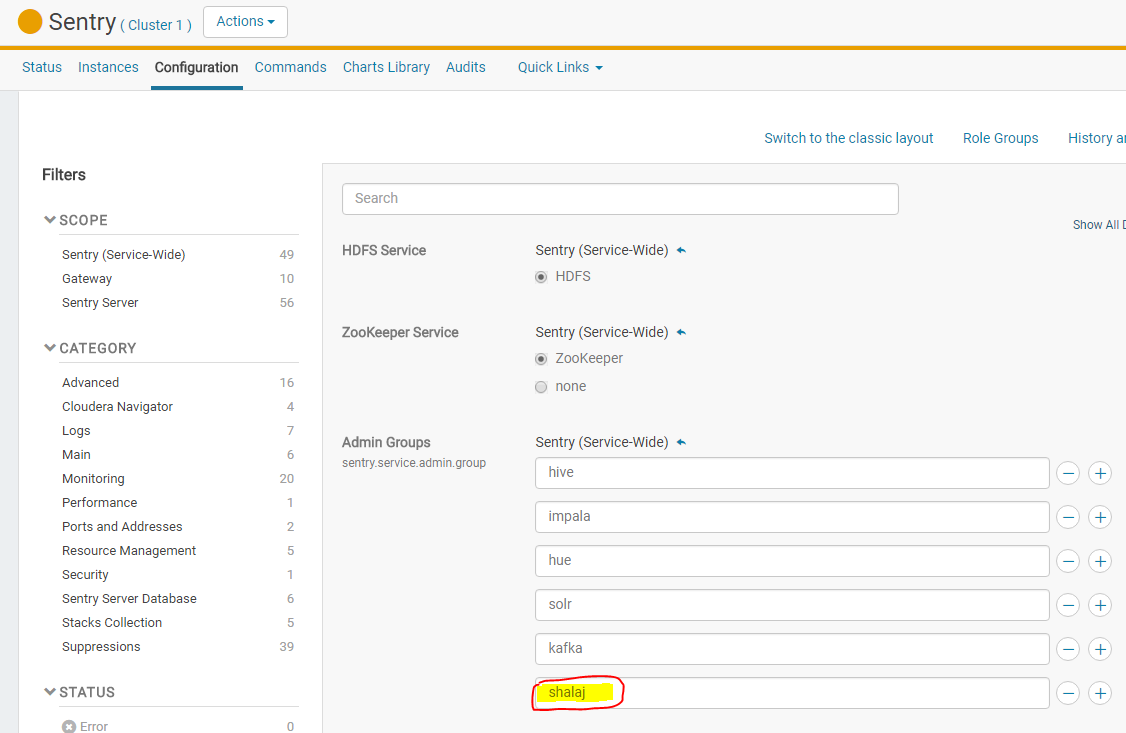


### Specify admin group

We have already added user shalaj under group shalaj , here we need to add this group under admin group so that we can create and drop role using this group

Go to Sentry>>configuration

Under **sentry.service.admin.group** add Group shalaj and restart the service



## ****Role Management Commands****

### Create Role

|  |
| --- |
| CREATE ROLE role\_name; |

Create a new role. Only the **admin** role has privilege for this.

The role names **ALL**, **DEFAULT** and **NONE** are reserved.

### Drop Role

|  |
| --- |
| DROP ROLE role\_name; |

Drop the given role. Only the **admin** role has privilege for this.

### Show Current Roles

|  |
| --- |
| SHOW CURRENT ROLES; |

Show the list of the user's current roles. All actions of the user are authorized by looking at the privileges of the user and all current roles of the user.

The default current roles have all roles for the user except for the admin role (even if the user belongs to the admin role as well).

Any user can run this command.

### Set Role

|  |
| --- |
| SET ROLE (role\_name|ALL|NONE); |

The SET ROLE statement can be used to specify a role to be enabled for the current session.

If a role\_name is specified, then that role becomes the only role in current roles.

Setting role\_name to ALL refreshes the list of current roles (in case new roles were granted to the user) and sets them to the default list of roles.

Setting role\_name to NONE will remove all current roles from the current user.

If a role the user does not belong to is specified as the role\_name, it will result in an error.

### SHOW Statement

* To list the database(s) for which the current user has database, table, or column-level access:

SHOW DATABASES;

* To list the table(s) for which the current user has table or column-level access:

SHOW TABLES;

* To list the column(s) to which the current user has SELECT access:

SHOW COLUMNS;

* To list all the roles in the system (only for sentry admin users):

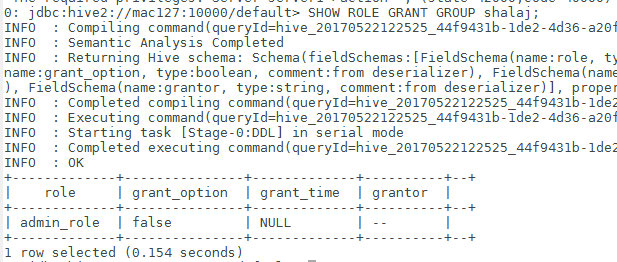
SHOW ROLES;

* To list all the roles in effect for the current user session:

SHOW CURRENT ROLES;

* To list all the roles assigned to the given <groupName> (only allowed for Sentry admin users and others users that are part of the group specified by <groupName>):

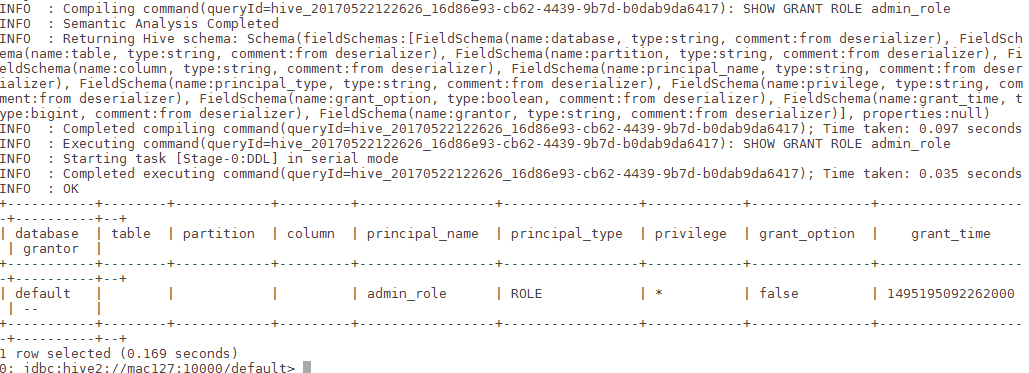
SHOW ROLE GRANT GROUP <groupName>;



* The SHOW statement can also be used to list the privileges that have been granted to a role or all the grants given to a role for a particular object.

To list all the grants for the given <roleName> (only allowed for Sentry admin users and other users that have been granted the role specified by <roleName>). The following command will also list any column-level privileges:

SHOW GRANT ROLE <roleName>;



* To list all the grants for a role on the given <objectName> (only allowed for Sentry admin users and other users that have been granted the role specified by <roleName>). The following command will also list any column-level privileges:

SHOW GRANT ROLE <roleName> on OBJECT <objectName>;

# Create and Grant role

Here we have already added two OS users- shalaj and shalajs , both have been assigned valid Kerberos ticket, please check Kerberos document attached in prerequisite section

Group name is same as user name

Check **/etc/passwd** file



Hive CLI is not allowed as it is not used HIVE\_SERVER2, so login with beeline

|  |
| --- |
| [root@mac127 ~]# su - shalaj  Last login: Fri May 19 15:55:31 IST 2017 on pts/3 |
| #assign Kerberos ticket  [shalaj@mac127 ~]$ kinit -kt /etc/security/keytabs/shalaj.keytab shalaj@CYBAGE.COM |
| [shalaj@mac127 ~]$ beeline  Beeline version 1.1.0-cdh5.10.1 by Apache Hive  beeline> !connect jdbc:hive2://mac127:10000/default;principal=hive/mac127@CYBAGE.COM  scan complete in 3ms  Connecting to jdbc:hive2://mac127:10000/default;principal=hive/mac127@CYBAGE.COM  Connected to: Apache Hive (version 1.1.0-cdh5.10.1)  Driver: Hive JDBC (version 1.1.0-cdh5.10.1)  Transaction isolation: TRANSACTION\_REPEATABLE\_READ  0: jdbc:hive2://mac127:10000/default> |

Run below commands to grant ALL rights to group shalaj on default database

**Grant privileges to** admin\_role

|  |
| --- |
| **CREATE ROLE** admin\_role**;** |
| **GRANT ALL ON DATABASE** default **to role** admin\_role**;** |
| **GRANT ROLE** admin\_role **TO GROUP** shalaj**;** |

To create external table first we need to grant access on URI, later we can create external table on this location

|  |
| --- |
| **GRANT ALL ON URI** 'hdfs://mac127:8020/user/shalaj/test' **TO ROLE** admin\_role; |
| **CREATE EXTERNAL TABLE** tmp (a string) location 'hdfs://mac127:8020/user/shalaj/test'; |

To Revoke Role

|  |
| --- |
| **REVOKE aLL ON URI** 'hdfs://mac127:8020/user/shalaj/test' **FROM ROLE** admin\_role; |

Give all permissions to admin\_role.

|  |
| --- |
| GRANT ALL ON SERVER server1 TO ROLE admin\_role; |
| GRANT ROLE admin\_role TO GROUP shalaj; |

Now group shalaj has authority to do anything like create database, create external table, drop database etc.

User Shalaj is now administrator so he can create different roles and assign to different group

|  |
| --- |
| **CREATE ROLE** dummy\_role; |
| **GRANT ALL ON DATABASE** dummy to ROLE dummy\_role; |
| **GRANT ROLE** dummy\_role to **GROUP** shalajs; |

Now user shalaj (Group – shalaj) has all permissions on dummy database.

Now if you login to hue with user shalaj, he can see all the databases and can access all the tables

But if you login via shalajs, he can access only dummy database

Please find below some examples of how to grant permission to some group

## Grant privileges to analyst\_role:

CREATE ROLE analyst\_role;

GRANT ALL ON DATABASE analyst1 TO ROLE analyst\_role;

GRANT SELECT ON DATABASE jranalyst1 TO ROLE analyst\_role;

GRANT ALL ON URI 'hdfs://ha-nn-uri/landing/analyst1' \

TO ROLE analyst\_role;

## Grant privileges to junior\_analyst\_role:

CREATE ROLE junior\_analyst\_role;

GRANT ALL ON DATABASE jranalyst1 TO ROLE junior\_analyst\_role;

GRANT ALL ON URI 'hdfs://ha-nn-uri/landing/jranalyst1' \

TO ROLE junior\_analyst\_role;

## Grant privileges to admin\_role:

CREATE ROLE admin\_role

GRANT ALL ON SERVER server1 TO ROLE admin\_role;

## Grant roles to groups:

GRANT ROLE admin\_role TO GROUP admin;

GRANT ROLE analyst\_role TO GROUP analyst;

GRANT ROLE jranalyst\_role TO GROUP jranalyst;

Refer links

<http://boopathi.me/blog/hive-sentry-role-grant-cheat-sheet/>

<https://cwiki.apache.org/confluence/display/Hive/SQL+Standard+Based+Hive+Authorization>

<https://www.cloudera.com/documentation/enterprise/5-5-x/topics/sg_hive_sql.html>

<https://cwiki.apache.org/confluence/display/Hive/SQL+Standard+Based+Hive+Authorization>