Exercise 1

Apache Ranger Authorization and Auditing

CREATE A RANGER POLICY TO LIMIT ACCESS TO HIVE DATA

1. Login to Ambari <your system IP>:8080 with username/password as raj\_ops
2. We’ll first create a simple table in Hive View and populate it.

create table employee (ssn string, name string, location string)

row format delimited

fields terminated by ','

stored as textfile;

create a .txt file in your local with the below values

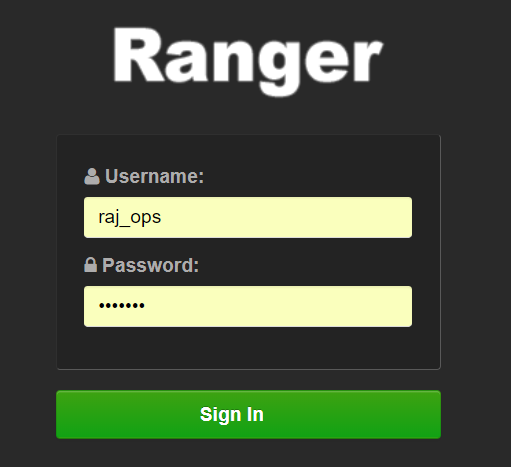
123456789, Josh, Dallas

987654321, Maria, Tampa

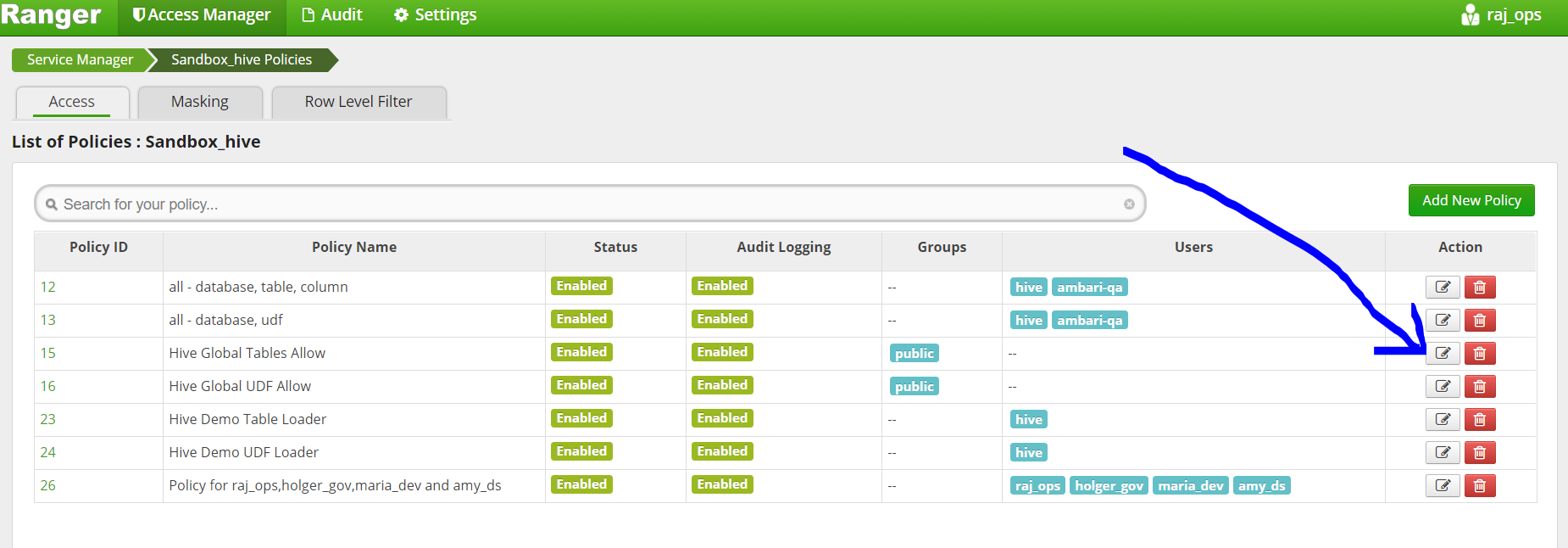
321654987, Vienna, Atlanta

Upload it to HDFS and load this file into the employee table

1. Go to Ranger UI on Ambari - [<your IP>:6080](http://192.168.64.128:6080) – Ranger UI port
2. At the login screen, give username and password as raj\_ops

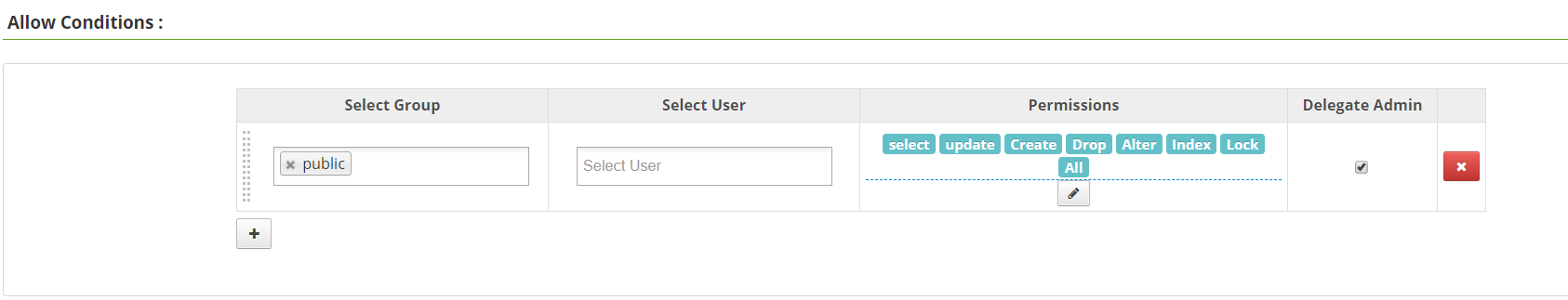


1. Since we are going to work on limiting access to Hive Tables, go to Sandbox\_Hive in the Ranger dashboard
2. You’ll see various policy ID’s, names and Users in the Ranger--->AccessManager tab



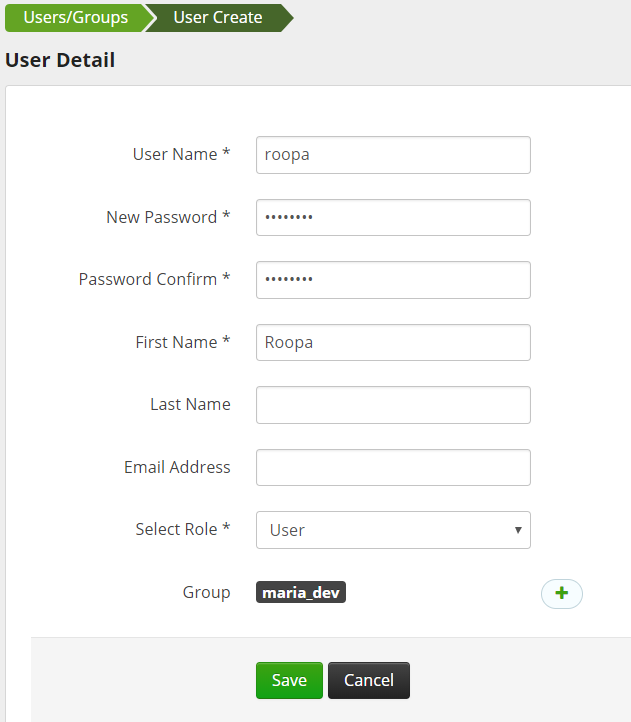
Click on ‘Edit’ button under ‘ACTIONS’ for the policy name - ‘HIVE GLOBAL TABLES ALLOW’

In this screen you’ll see for permissions it has all the permissions selected for ‘PUBLIC GROUP’

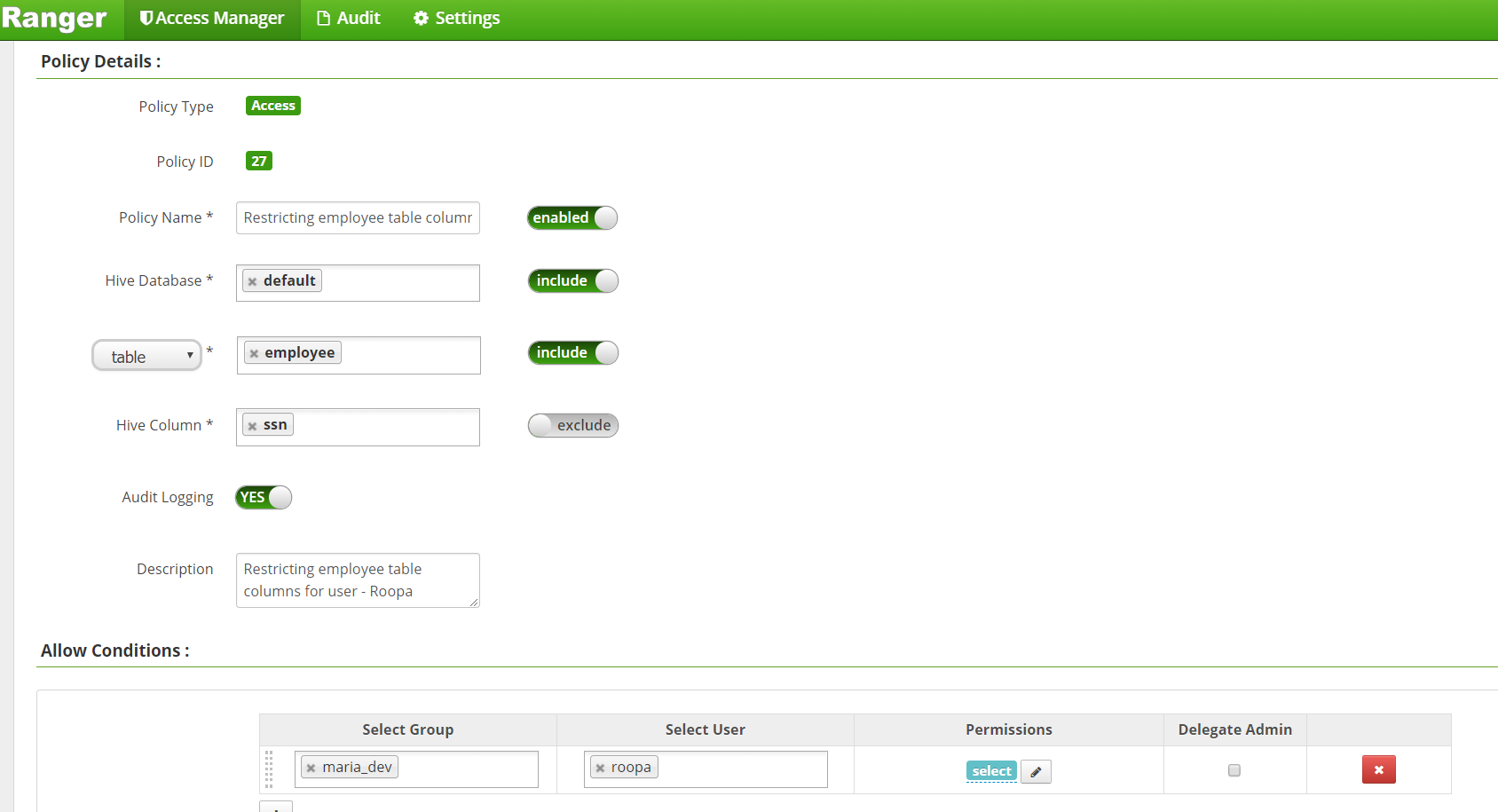


1. Come back to the Ranger UI page and on top you’ll see under settings → Users/Groups

Add a new user using the option add user. A user called ‘Roopa’ is created under group ‘maria\_dev’. Hit save.



1. So, for this user ‘Roopa’ we are going to restrict access to a particular column ‘ssn’ in the ‘EMPLOYEE’ table. The other fields are accessible. Also the user is given only ‘SELECT’ access and no other access.
2. Back in Access Manager → sandbox\_Hive click on ‘Add New Policy’



The options chosen are

Policy name – Give a policy name and a description for this policy at the bottom

Hive Database –select default

Table –choose the newly created employee table

Hive column – choose ssn and hit the ‘exclude’ button which means other than this column, the other columns are accessible

In the Allow Conditions

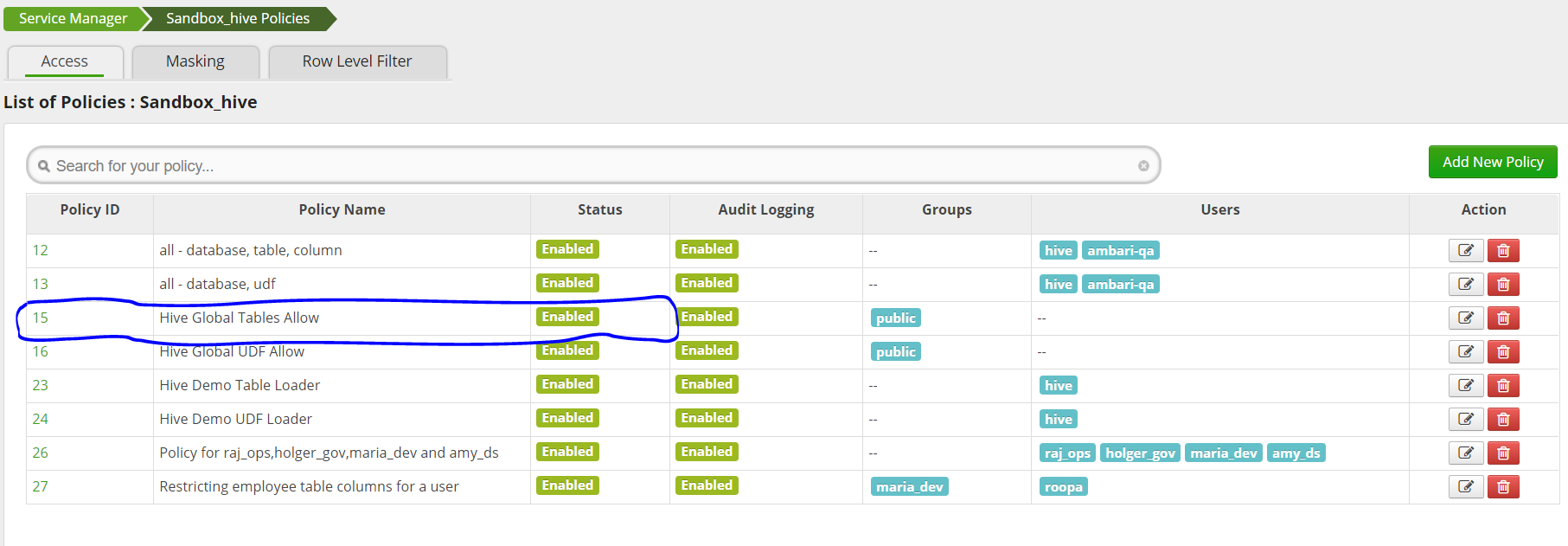
Select group as ‘maria\_dev’ select user as ‘Roopa’ and under permissions only give ‘SELECT’ permission

Hit Add policy

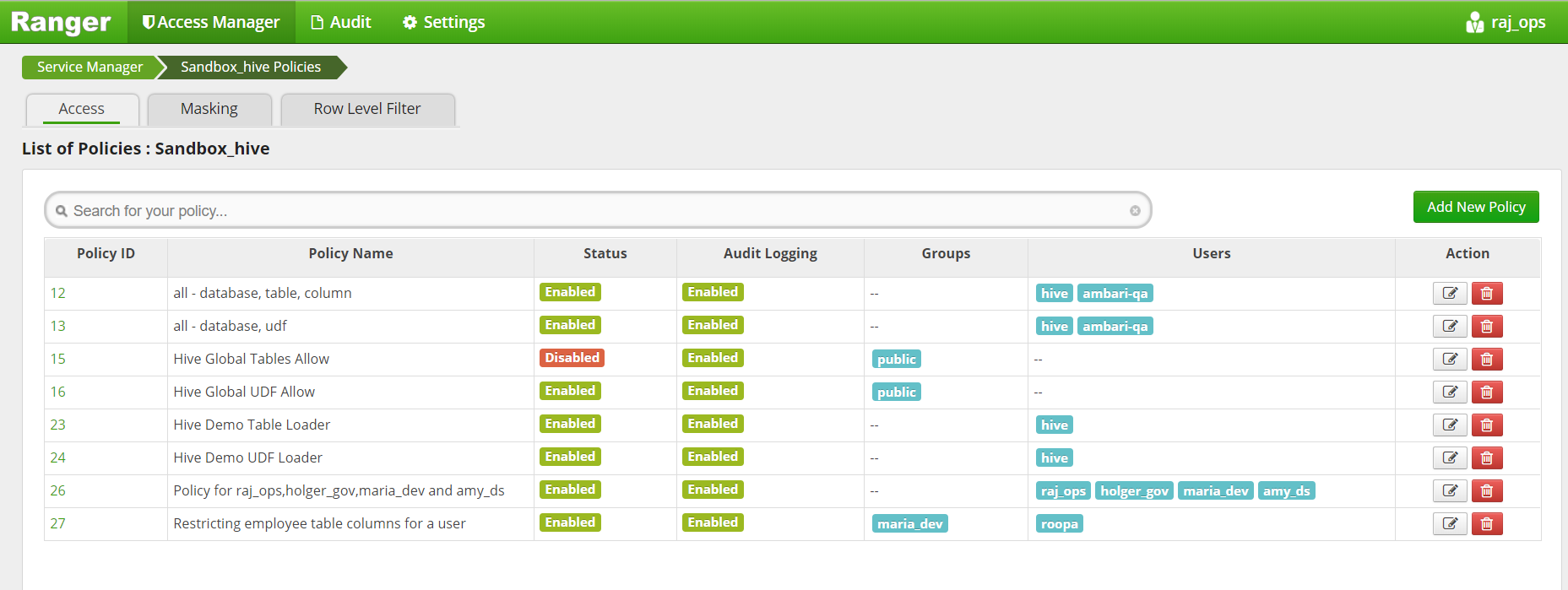
So a new policy for the user ‘Roopa’ has been created.

1. See how on the access manager tab ‘Hive Global Tables Allow’ has been enabled for the ‘public’ group so this will enable access to all users to all the tables. We will disable this first to test our user access limits.

Click edit against this policy

1. 

And against the policy name, disable it and save the policy.

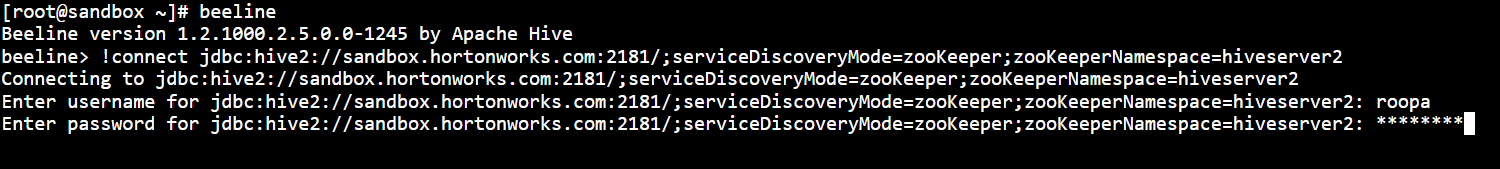


1. Now we get to test this policy for user ‘Roopa’

For this we log into Hive on linux using the beeline client and connect to HiveServer2

In the command prompt type beeline

Next copy the JDBC URL from ambari → Hive → Hiveserver2 JDBC URL

In the beeline command line type !connect <JDBC URL that was copied>

At the prompt type username as ‘Roopa’ and password as whatever was set at the user creation time.

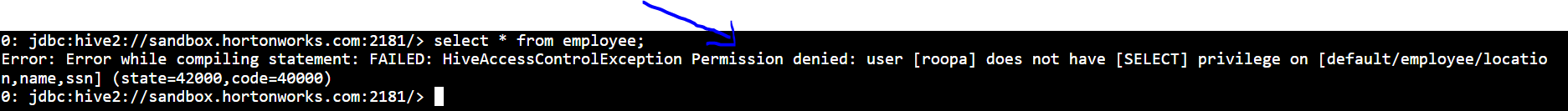
You can now run queries.

Let’s test the access policy.

First type showdatabases; command and see the databases displayed

Next → SELECT\* FROM EMPLOYEE;

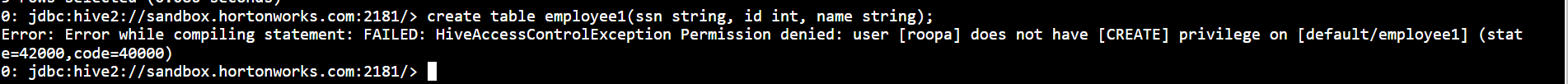
You’ll get an error as shown below because user Roopa only has access to 2 columns in the table.



1. This shows that we have effectively hidden the ‘ssn’ column from user ‘Roopa’

Now do Select name, location from employee;

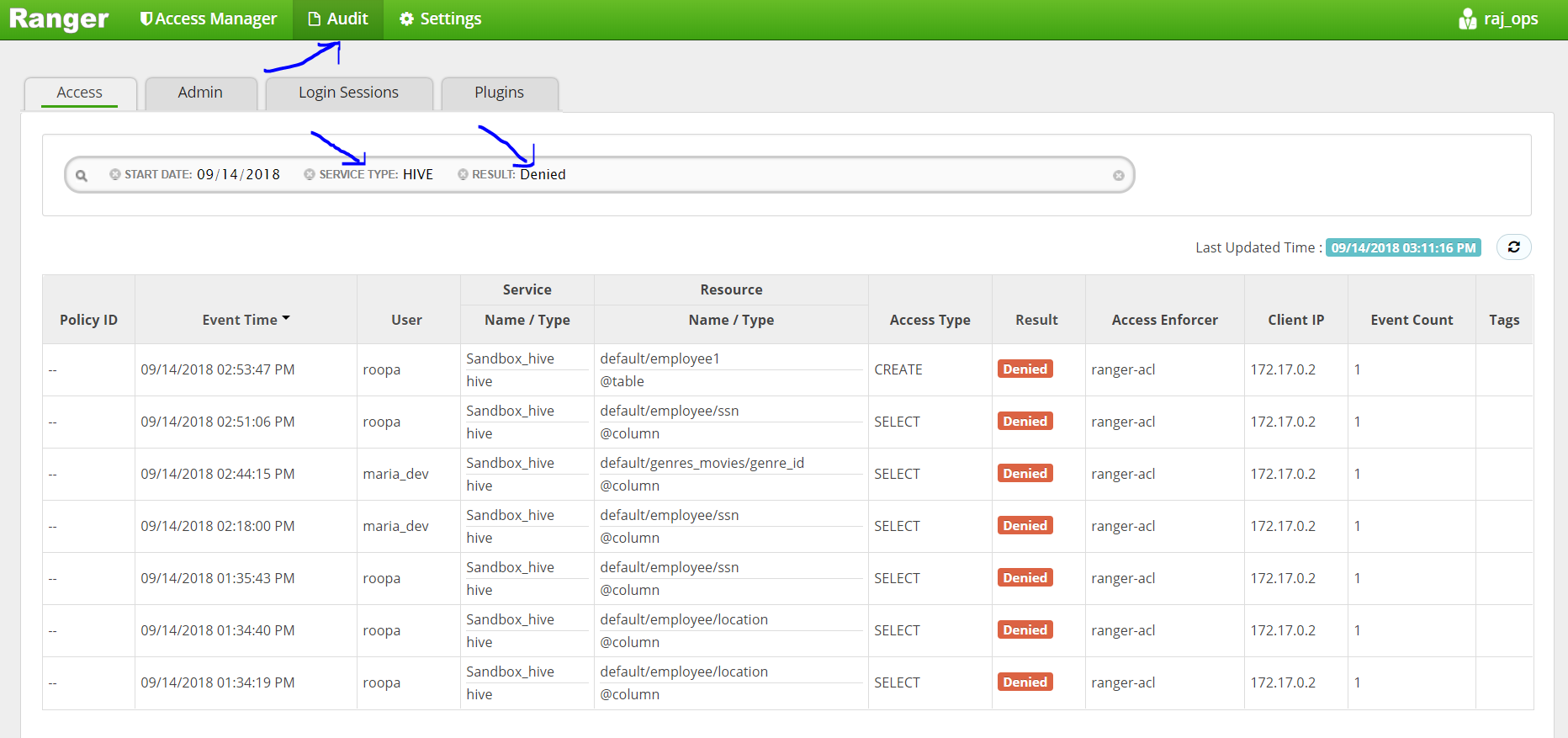
You should be able to see the output.

1. 

While here check for other DDL options to ensure user doesn’t have access.

1. We can look into Ranger Audit reports to look for all the times users tried to access certain tables, what was the result, the event times etc.
2. For looking at the Audit reports, go to Ranger UI and select ‘AUDIT’ on top

Under access, its selected on date, filter by ‘service type = HIVE’ and filter again on Result=’DENIED’ and you’ll see something like this



1. On the next tab – admin in the Audits, you can see what operations were done by which user like so ---

