Getting started with Hive with Kerberos.

Apache Hive is a powerful data warehousing application built on top of Hadoop; it enables you to access your data using Hive QL, a language that is similar to SQL. Install Hive on your client machine(s) from which you submit jobs; you do not need to install it on the nodes in your Hadoop cluster. If Kerberos authentication is used, authentication is supported between the Thrift client and `HiveServer2`, and between `HiveServer2` and secure HDFS.

If you configure `HiveServer2` to use Kerberos authentication, `HiveServer2` acquires a Kerberos ticket during start-up. `HiveServer2` requires a principal and keytab file specified in the configuration. The client applications (for example JDBC or Beeline) must get a valid Kerberos ticket before initiating a connection to `HiveServer2`.

Grant Permissions to user groups to access `hive`.

Login to the server and create a role. If these roles are not created then we get permission (Privileges) Issues.

Issue as below.

```
Error: Error while compiling statement: FAILED: SemanticException No valid privileges
 Required privileges for this query: Server=server1->action=*; (state=42000,code=40000)
```

Here is how to grant permissions to hive group, so that you can access it.

```
[sas@waepprrkb004 root]$  beeline -u \
                    "jdbc:hive2://hive-server.server.com:10000/default;principal=\
                        hive/hive-server.server.com@XYZ.DOMAIN.COM"

0: jdbc:hive2://hive-server.server.com> create role admin;
1 row affected
0: jdbc:hive2://hive-server.server.com> show roles;
+--------+--+
|  role  |
+--------+--+
| admin  |
+--------+--+

0: jdbc:hive2://hive-server.server.com> GRANT ROLE admin TO GROUP hive;
0: jdbc:hive2://hive-server.server.com> GRANT ALL ON DATABASE default TO ROLE admin;
```

Here is the complete output after the permissions are granted.

```
[sas@waepprrkb004 root]$  beeline -u \
    "jdbc:hive2://hive-server.server.com:10000/default;principal=\
    hive/hive-server.server.com@XYZ.DOMAIN.COM"
scan complete in 1ms
```

```
Connecting to jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                hive/hive-server.server.com@XYZ.DOMAIN.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.5)
Driver: Hive JDBC (version 1.1.0-cdh5.4.5)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.4.5 by Apache Hive
0: jdbc:hive2://hive-server.server.com> show databases;
+----------------+--+
| database_name  |
+----------------+--+
| default        |
+----------------+--+
1 row selected (0.134 seconds)
0: jdbc:hive2://hive-server.server.com> show roles;
+--------+--+
|  role  |
+--------+--+
| admin  |
+--------+--+
1 row selected (0.063 seconds)
0: jdbc:hive2://hive-server.server.com> use default;
No rows affected (0.05 seconds)
0: jdbc:hive2://hive-server.server.com> show tables;
+------------+--+
|  tab_name  |
+------------+--+
| sample_07  |
| sample_08  |
+------------+--+
2 rows selected (0.08 seconds)
0: jdbc:hive2://hive-server.server.com>
```

Adding New Roles and Groups in Hive.

Before we start accessing the data, we need to give users permission.

1. Create a role.
2. Assign role some permissions (SELECT `[readonly]`, INSERT `[rw]`, ALL`[all]`).
3. Add a group to the newly create role.

Creating a new role.

First we create `roles` which we later give permissions to.

```
0: jdbc:hive2://hive-server.server.com> CREATE ROLE admin;
0: jdbc:hive2://hive-server.server.com> CREATE ROLE readonly;
```

Assign role permissions.

We are assigning permission to a role `readonly` to a database (`default`)

```
0: jdbc:hive2://hive-server.server.com> GRANT SELECT ON DATABASE default TO ROLE readonly;
```

Adding a new active directory group to role.

Now we assign the role `readonly` to a group `server-user-access-group`. Here `server-user-access-group` is an Active Directory group which is sync with Linux using SSSD.

```
0: jdbc:hive2://hive-server.server.com> GRANT ROLE readonly TO GROUP server-user-access-group;
```

Adding new External tables

Grant permission to HDFS URI to access the AVRO data.

```
grant all on uri 'hdfs://nameservice1/data/location/hdfs/some_data/ahmed/' to role admin;
```

Creating external table.

```
create external table ahmed-data partitioned by (partition_val1 String,partition_val2 String) \
    stored as avro location '/data/location/hdfs/some_data/ahmed/' \
    TBLPROPERTIES ('avro.schema.url'='data/location/hdfs/some_data_schema/v1/ahmed_schema.avsc');
```

Alter table to partition it.

```
alter table ahmed-data add partition (partition_val1="2015", partition_val2="07");
```

Testing Setup.

Logging as `hive` user to giver permission to a specific `group`.

```
[root@edge-gw-server keytabs]# beeline -u \
            "jdbc:hive2://hive-server.server.com:10000/default;principal=\
            hive/hive-server.server.com@XYZ.DOMAIN.COM"
scan complete in 2ms
Connecting to jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.5)
Driver: Hive JDBC (version 1.1.0-cdh5.4.5)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.4.5 by Apache Hive
0: jdbc:hive2://hive-server.server.com>
0: jdbc:hive2://hive-server.server.com> create role admin;
0: jdbc:hive2://hive-server.server.com> create role readonly;
```

Checking for roles in `hive`.

```
0: jdbc:hive2://hive-server.server.com> show roles;
+-----------+--+
|   role    |
+-----------+--+
| admin     |
| readonly  |
+-----------+--+
2 rows selected (0.349 seconds)
```

Grant `readonly` role to `server-user-access-group` group. (But still we have not given permission to role `readonly` we do that in next step)

```
0: jdbc:hive2://hive-server.server.com> grant role readonly to group server-user-access-group;
No rows affected (0.04 seconds)
```

Assigning role `readonly` `select` permission on the `default` database.

```
0: jdbc:hive2://hive-server.server.com> grant select on database default to role readonly;
No rows affected (0.049 seconds)
0: jdbc:hive2://hive-server.server.com> show role grant group server-user-access-group;
+-----------+---------------+-------------+----------+--+
|   role    | grant_option  | grant_time  | grantor  |
+-----------+---------------+-------------+----------+--+
| readonly  | false         | NULL        | --       |
+-----------+---------------+-------------+----------+--+
1 row selected (0.062 seconds)
0: jdbc:hive2://hive-server.server.com> !quit
Closing: 0: jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
```

Now checking for permission for user `ahmed-user`, since the user is not part of `server-user-access-group` he will still not be able to access the data.

```
[root@edge-gw-server keytabs]# su ahmed-user
[ahmed-user@edge-gw-server keytabs]$ cd ~
[ahmed-user@edge-gw-server ~]$ kinit -kt ahmed-user_new.keytab ahmed-user@ABC.DOMAIN.COM
[ahmed-user@edge-gw-server ~]$ beeline -u \
                "jdbc:hive2://hive-server.server.com:10000/default;principal=\
                 hive/hive-server.server.com@XYZ.DOMAIN.COM"
scan complete in 2ms
Connecting to jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.5)
Driver: Hive JDBC (version 1.1.0-cdh5.4.5)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.4.5 by Apache Hive
0: jdbc:hive2://hive-server.server.com> show tables;
+-----------+--+
| tab_name  |
+-----------+--+
+-----------+--+
No rows selected (1.382 seconds)
0: jdbc:hive2://hive-server.server.com> !quit
Closing: 0: jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
[ahmed-user@edge-gw-server ~]$ exit
exit
```

Now again logging into `hive` superuser to grant permission to a group which `ahmed-user` user is part of.

```
[root@edge-gw-server keytabs]# kinit -kt hive.keytab hive/hive-server.server.com@XYZ.DOMAIN.COM
[root@edge-gw-server keytabs]# beeline -u \
                "jdbc:hive2://hive-server.server.com:10000/default;principal=\
```

```
                hive/hive-server.server.com@XYZ.DOMAIN.COM"
scan complete in 2ms
Connecting to jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.5)
Driver: Hive JDBC (version 1.1.0-cdh5.4.5)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.4.5 by Apache Hive
0: jdbc:hive2://hive-server.server.com> grant role readonly to group ahmed-user-access-group;
No rows affected (0.332 seconds)
0: jdbc:hive2://hive-server.server.com> !quit
Closing: 0: jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
```

Login as `ahmed-user` user. Now we can see the tables.

```
[root@edge-gw-server keytabs]# su ahmed-user
[ahmed-user@edge-gw-server keytabs]$ cd ~
[ahmed-user@edge-gw-server ~]$ kinit -kt ahmed-user_new.keytab ahmed-user@ABC.DOMAIN.COM
[ahmed-user@edge-gw-server ~]$ beeline -u \
                    "jdbc:hive2://hive-server.server.com:10000/default;principal=\
                    hive/hive-server.server.com@XYZ.DOMAIN.COM"
scan complete in 2ms
Connecting to jdbc:hive2://hive-server.server.com:10000/default;principal=
                                                    hive/hive-server.server.com@XYZ.DOMAIN.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.5)
Driver: Hive JDBC (version 1.1.0-cdh5.4.5)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.4.5 by Apache Hive
0: jdbc:hive2://hive-server.server.com> show tables;
+------------+--+
|  tab_name  |
+------------+--+
| ahmed-data |
| sample_07  |
| sample_08  |
+------------+--+
8 rows selected (0.183 seconds)
0: jdbc:hive2://hive-server.server.com>
```

Test Complete.