SQL Based authorization in hive

(/users/504/kkulkarni.html)

Kuldeep Kulkarni (/users/504/kkulkarni.html) ✓ created · Nov 21, 2015 at 03:17 PM · **edited (/revisions/4568.html)** · Jan 02 at 07:55 PM

10

☆

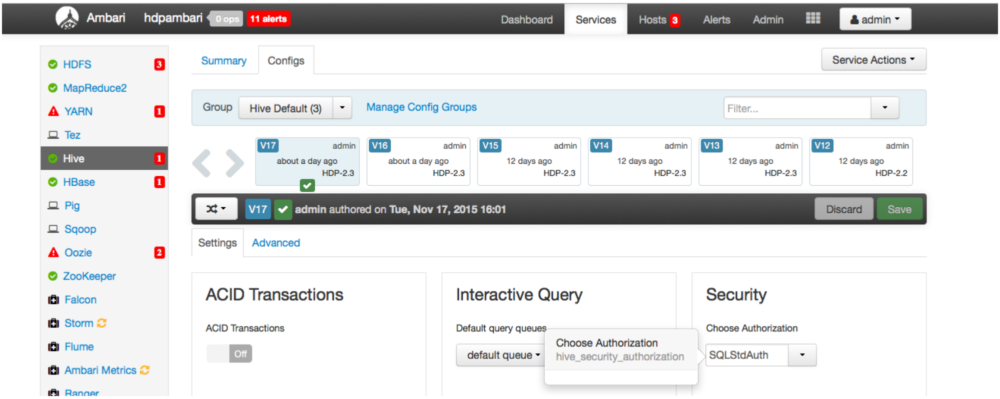
Short Description:

This tutorial is to enable SQL Based authorization for hive.

Article

Step 1 - Goto ambari UI and add/modify below properties

Goto service hive → configs and change autherization to SQLStdAuth



V17 ✓ admin authored on Tue, Nov 17, 2015 16:01

ACID Transactions: Off

Default query queues: default queue

Choose Authorization (hive_security_authorization): SQLStdAuth

Step 2 - In Hive-site.xml, make sure you have set below properties:

```
    hive.server2.enable.doAs --> false


    hive.users.in.admin.role --> root (comma separated list of users)
```

Step 3 - Make sure that you have below properties set in Hiveserver2-site.xml

```
hive.security.authorization.manager --> org.apache.hadoop.hive.ql.security.authorization.plugin.sqlstd.SQLStdConfOnlyAuthorizerFactory

hive.security.authorization.enabled --> true

hive.security.authenticator.manager --> org.apache.hadoop.hive.ql.security.ProxyUserAuthenticator
```

Step 4 - Restart hive services from UI

Step 5 - Testing our setup, in this we will create one readonly user and try to drop table.

5.1 Login to beeline using root (as we have added root in hive.users.in.admin.role)

```
0: jdbc:hive2://localhost:10010> !connect jdbc:hive2://localhost:10010Connecting to jdbc:hive2://localhost:10010

Enter username for jdbc:hive2://localhost:10010: root
Enter password for jdbc:hive2://localhost:10010: ****

Connected to: Apache Hive (version 1.2.1.2.3.2.0-2950)Driver: Hive JDBC (version 1.2.1.2.3.2.0-2950)Transaction isolation:
TRANSACTION_REPEATABLE_READ1: jdbc:hive2://localhost:10010>
```

5.2 Now by default there is only one role - public, you need to run below command to set your role as ADMIN.

```
0: jdbc:hive2://localhost:10010> SHOW CURRENT ROLES;


+---------+--+
|  role  |
+---------+--+
| public  |
+---------+--+
```

5.3 Set role as admin for user root

```
1: jdbc:hive2://localhost:10010> set role ADMIN;
No rows affected (0.445 seconds)1:
```

```
jdbc:hive2://localhost:10010> show roles;
+---------+--+
|  role  |
+---------+--+
| admin  |
| public  |
+---------+--+
2 rows selected (0.165 seconds)
```

5.4 Create new role readonly and add readonly_user in that group

```
0: jdbc:hive2://slave1.hortonworks.com:10010/> create role readonly;
No rows affected (0.071 seconds)
```

5.5 Verify that new role has been created successfully

```
0: jdbc:hive2://slave1.hortonworks.com:10010/> show roles;
+-----------+--+
|  role   |
+-----------+--+
| admin   |
| public   |
| readonly   |
+-----------+--+
3 rows selected (0.051 seconds)
0: jdbc:hive2://slave1.hortonworks.com:10010/>
```

5.6 Add readonly_user into readonly role

```
5: jdbc:hive2://slave1.hortonworks.com:10010> grant role readonly to user readonly_user;
No rows affected (0.088 seconds)
5: jdbc:hive2://slave1.hortonworks.com:10010>
```

5.7 Grant select privileges to role readonly

```
5: jdbc:hive2://slave1.hortonworks.com:10010> grant select on table batting to role readonly;
No rows affected (0.405 seconds)
5: jdbc:hive2://slave1.hortonworks.com:10010>
```

5.8 Verify grants for role readonly

```
0: jdbc:hive2://slave1.hortonworks.com:10010/> show grant role readonly;

+-----------+----------+------------+---------+-----------------+-----------------+------------
+---------------+---------------+---------+--+
| database  |  table  | partition  | column  | principal_name  | principal_type  | privilege  |
 grant_option  |  grant_time  | grantor  |
+-----------+----------+------------+---------+-----------------+-----------------+------------
+---------------+---------------+---------+--+
| default   | batting  |   |   | readonly  | ROLE   | SELECT   | false   | 1447877696000   | root   |
+-----------+----------+------------+---------+-----------------+-----------------+------------
```

5.9 Now login to beeline by user readonly_user and try to drop table batting

```
beeline> !connect jdbc:hive2://slave1.hortonworks.com:10010/
Connecting to jdbc:hive2://slave1.hortonworks.com:10010/
Enter username for jdbc:hive2://slave1.hortonworks.com:10010/: readonly_user
Enter password for jdbc:hive2://slave1.hortonworks.com:10010/: ********
Connected to: Apache Hive (version 1.2.1.2.3.2.0-2950)Driver: Hive JDBC (version 1.2.1.2.3.2.0-295
0)
Transaction isolation: TRANSACTION_REPEATABLE_READ
```

```
0: jdbc:hive2://slave1.hortonworks.com:10010/> drop table batting;
Error: Error while compiling statement: FAILED: HiveAccessControlException Permission denied: Prin
cipal [name=readonly_user, type=USER] does not have following privileges for operation DROPTABLE
[[OBJECT OWNERSHIP] on Object [type=TABLE_OR_VIEW, name=default.batting]] (state=42000,code=40000)
```

Note - we are getting an error here because readonly_user does not have permission to drop table batting!

5.10 Let's try to access some rows from table batting

```
    0: jdbc:hive2://slave1.hortonworks.com:10010/> select * from batting limit 5;
    +-------------------+---------------+---------------+--+
    | batting.player_id  | batting.year  | batting.runs  |
    +-------------------+---------------+---------------+--+
    | playerID   | NULL   | NULL   |
    | aardsda01  | 2004  | 0  |
    | aardsda01  | 2006  | 0  |
    | aardsda01  | 2007  | 0  |
    | aardsda01  | 2008  | 0  |
    +-------------------+---------------+---------------+--+
```

We can see that grants are working and user can see the contents but cannot delete the table.

Attached my hive-site.xml (https://www.dropbox.com/s/eh6npsn6ylv7gb2/hive-site.xml?dl=0) for more details.

How-To/Tutorial (/topics/How-To%2FTutorial.html) **Hive (/topics/Hive.html)** **hive-authorization (/topics/hive-authorization.html)** **hive-sql (/topics/hive-sql.html)**

Add comment · 💬 Show 2

ARTICLE

Contributors

(/users/504/kkulkarni.html)

FOLLOW

+ FOLLOW (/FOLLOW/4568/KBENTRY.HTML)

(/use[illegible]html)

(/use[illegible]ml)

(/use[illegible]html)

(/use[illegible]en.html)

(/use[illegible]2 akansha-

(/use[illegible]html))

NAVIGATION

SQL Based authorization in hive (/articles/4568/sql-based-authorization-in-hive.html)

RELATED ARTICLES

Investigating LLAP cache hit rate (/articles/149901/investigating-llap-cache-hit-rate.html)

LLAP debugging overview - logs, UIs, etc (/articles/149896/llap-debugging-overview-logs-uis-etc.html)

Using Regular Expressions to Extract Fields for Hive Tables (/articles/58591/using-regular-expressions-to-extract-fields-for-hi.html)

Investigating when LLAP doesn’t start (/articles/149899/investigating-when-llap-doesnt-start.html)

Investigating when the queries on LLAP are slow or stuck (/articles/149900/investigating-when-the-queries-on-llap-are-slow-or.html)

Hive on Tez Performance Tuning - Determining Reducer Counts (/articles/22419/hive-on-tez-performance-tuning-determining-reducer.html)

Phoenix Timestamp - Leverage the core ROWKEY data model (/articles/53377/phoenix-timestamp-leverage-the-core-rowkey-data-mo.html)

Part 1 - Ad-hoc Query Workloads on HDP (/articles/121879/part-1-ad-hoc-query-workloads-on-hdp.html)

Oozie - Simulating looping with sub workflows (/articles/75118/oozie-simulating-looping-with-sub-workflows.html)

Apache Hive with Apache Hivemall (/articles/67983/apache-hive-with-apache-hivemall.html)

HCC Guidelines (http://community.hortonworks.com/page/community+guidelines) | HCC FAQs (http://community.hortonworks.com/page/faq.html) | HCC Privacy Policy (http://hortonworks.com/wp-content/uploads/2015/11/Hortonworks-Community-Connection-Beta-Terms.pdf) | Privacy Policy (http://hortonworks.com/agreements/privacy-policy/) | Terms of Service (http://hortonworks.com/agreements/terms-of-service/)