Authorize Hue User Groups with Sentry (#hue_sec_sentry_auth)

Like Hive and Impala, Hue communicates with Sentry using the thrift protocol; but you can also use the Security Browser in Hue to grant privileges. See Apache Sentry Made Easy (http://gethue.com/apache-sentry-made-easy-with-the-new-hue-security-app/) .
Note: Granting privileges in the Hue Security Browser > Hive Tables > Roles is the same as running `grant role` with HiveServer2 Beeline (https://cwiki.apache.org/confluence/display/Hive/HiveServer2+Clients#HiveServer2Clients-Beeline%E2%80%93CommandLineShell) .

This page demonstrates how to use Sentry in Hue by creating three Hue user groups (**readers**, **writers**, **sysadmins**), creating three corresponding Sentry roles (**reader_role**, **writer_role**, **sysadmin_role**), and granting privileges to those roles.

Continue reading:

Prerequisites (#concept_bh3_sz3_v1b)

To grant privileges with Sentry in Hue:

- CDH services must be authenticated (usually with Kerberos but LDAP is also allowed)
- Hue users and groups must be mapped to the OS with Hadoop User Group Mapping.

Hue users and groups do not need to be authenticated; but in production, LDAP authentication (hue_sec_ldap_auth.html#hue_sec_ldap_auth) is recommended.

Create Hue Users and Groups (#concept_tqb_vz3_t1b)

Note: Because CDH requires POSIX compliant user names (http://pubs.opengroup.org/onlinepubs/000095399/basedefs/xbd_chap03.html#tag_03_426) , Hue should use them too (even though Hue leverages `django.contrib.auth.models.User` which is less strict).

To *demonstrate* Sentry grants, we create three groups and corresponding Sentry roles.

1. Create three user groups (or import from LDAP (hue_sec_ldap_sync.html#hue_sec_ldap_sync)). Log on to Hue as a superuser, expand the user drop down, and select Manage Users.
   - **sysadmins** - user1
   - **writers** - user2
   - **readers** - user3, user4
2. Configure group permissions in Hue as appropriate. See Restrict Group Permissions (hue_sec_ldap_sync.html#concept_g1v_m2j_f1b) .
3. Ensure that users and groups are defined in the OS with Hadoop User Group Mapping. See Hue User Permissions (hue_adm_permissions.html#hue_adm_permissions) .

Enable Sentry in Hue Security Browser (#concept_mhm_dcc_q1b)

This section explains how to configure the Sentry service to work with Hue and CDH services: Hue, Hive, and Impala.

1. Set up an external database for Sentry metadata.
   Note: See Hue Custom Databases (hue_dbs_0.html#hue_database_guide) for guidance and create a table something like this:

   ```
   create database sentry default character set utf8 default collate utf8_general_ci;
   grant all on sentry.* to 'sentry'@'%' identified by 'sentrypassword';
   ```

2. Log on to Cloudera Manager and Add the Sentry Service (sg_sentry_service_install.html#concept_zsk_pr5_1q__section_pwm_fsx_kq) . (Hue does not need a gateway.)
3. Configure Sentry Admin Groups (sg_sentry_service_config.html#concept_z5b_42s_p4__section_vrc_1dk_55) for applicable services and manually add the Hue user group (in this demo, **sysadmins**).

Note: See More on Sentry Admin Groups (hue_sec_sentry_auth.html#concept_b1k_4mh_v1b) for details on user permissions.

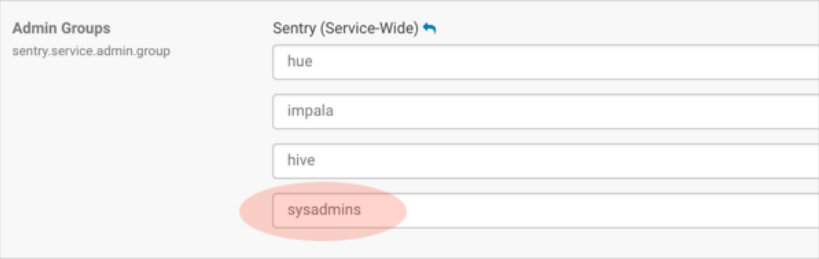
4. Enable Sentry Service for each applicable service installed: Hue (sg_sentry_service_config.html#concept_z5b_42s_p4__enablehue), Hive (sg_sentry_service_config.html#concept_z5b_42s_p4__section_n4d_4g4_rp), Impala (sg_sentry_service_config.html#concept_z5b_42s_p4__sentryserref).
   - Go to Hue > Configuration > Sentry Service, select Sentry radio button, and click Save Changes.
   - Repeat for Hive, Impala.
5. Uncheck Hive > Configuration > HiveServer2 Enable Impersonation.
6. Check HDFS > Configuration > Enable Access Control Lists.
7. Ensure all changes are saved and restart applicable services (or the entire cluster).

More on Sentry Admin Groups (#concept_b1k_4mh_v1b)

On startup, Hue reads `sentry_conf/sentry-site.xml` and looks for the property, `sentry.service.admin.group`.

In this demo, group **sysadmins** can grant Sentry roles within Hue. Members of sysadmins must be defined in the OS and also within Hue via Manage Users.

```
<property>
  <name>sentry.service.admin.group</name>
  <value>hue,hive,impala,sysadmins</value>
</property>
```



If user1 in our example is part of sysadmins on the backend, but not in Hue Manage Users > Groups, user1 will *not* be able to grant roles in the Hue UI (only in Beeline). If user1 is a Hue superuser, user1 can view roles in theSecurity Browser > Hive Tables, but cannot edit them.

Conversely, if we add user2 to sysadmins in Manage Users > Groups, user2 will not be able edit roles in Hue nor in Beeline because user2 is not part of sysadmins in the OS.
Note: To view logs, Enable DEBUG (hue_adm_logs.html#concept_jtf_jtg_jw) and run:

```
cat /var/log/hue/runcpserver.log | grep Sentry
```

Create Sentry Roles and Grant Privileges (#concept_jwb_fcc_q1b)

In this section, configure the **sysadmin_role**, and assign it to the **sysadmins** group, because it requires the most permissions.

System Administrator Requirements (#concept_fxc_dnn_v1b)

To create roles and run grants with Sentry in Hue, system administrators must be configured with:

- User/group membership defined in the OS with **Hadoop User Group Mapping**.
- User/group membership defined in Hue Manage Users.
- Superuser access configured in Hue Manage Users.
- Sentry **Database privileges** set to ALL (for select, insert, create privileges).
- Sentry **URI privileges** to all user directories in HDFS.
  Note: By default, every user has access to their own HDFS directory in `/user`; but permissions through Hive/Impala must be granted with a URI.
- **Default ACL** set for hive with `r-w-x` permissions so that it can load files into hdfs at `/hive/warehouse`.
- **ACL** set for the same to ensure recursive attempts are covered.

Create Roles and Grants (#concept_emz_fnn_v1b)

1. Log on to Hue as a user with Sentry Admin and Hue Superuser privileges (in this demo, `user1`).
2. Go to Security > HiveTables > Roles: `http://<your_hostname>:8889/hue/security/hive#@roles`
3. Click Add, enter a role name (`sysadmin_role`), and select a group from the drop down (`sysadmins`).
   Note: If group sysadmins exists but does not display in the drop down, manually enter it and press `return`.
4. Click the plus icon to begin assigning privileges.
5. Select the database radio button:
   - Enter a database name.
   - Select ALL for create database and table privileges.
   - Check the box, grant permissions to give others permission on this database.
6. Select the URI radio button, and enter the path to which you want hive to have access:

```
hdfs://<your hostname>:8020/user/
```

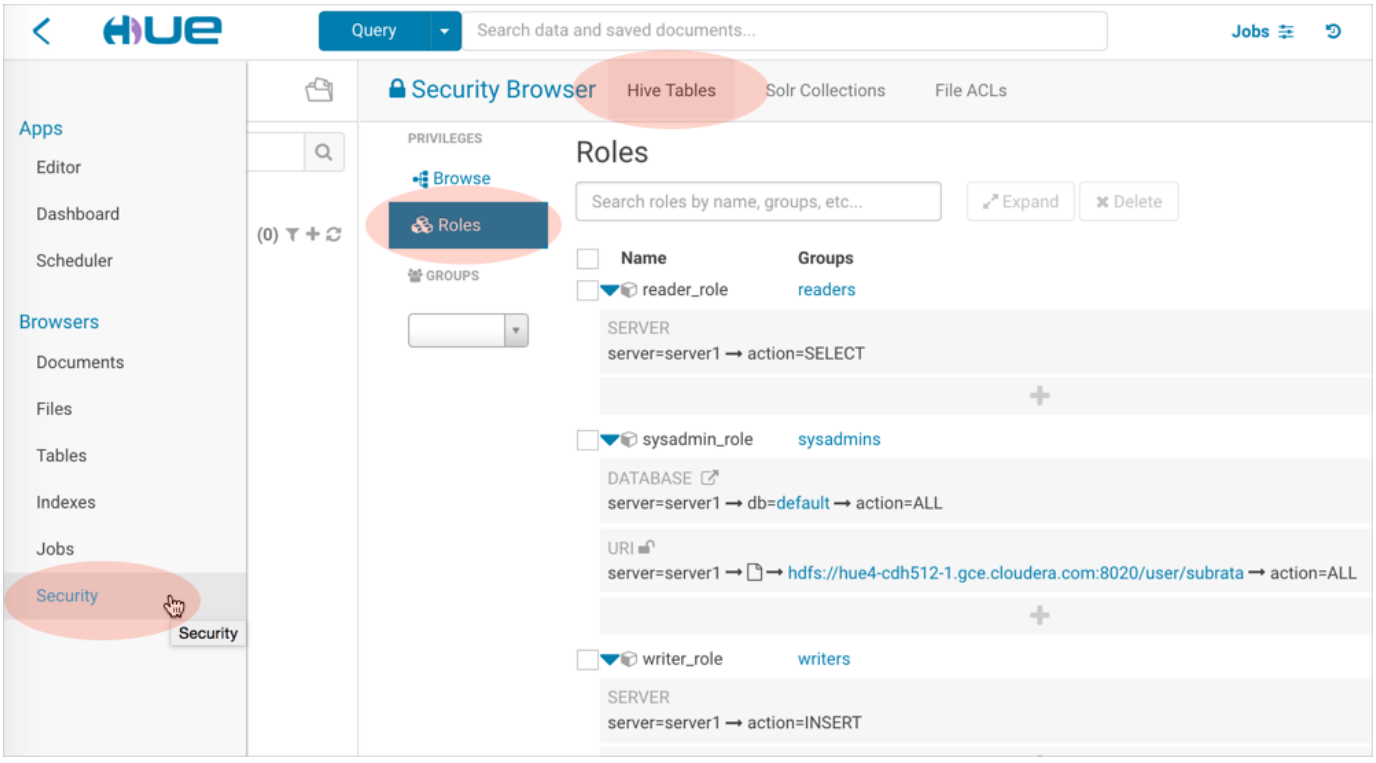
7. Go to the tab, File ACLs, in the Security Browser. ACLs give hive r-w-x permissions so that it can load files into /hive/warehouse in hdfs.
8. Add a Default ACL. For individual users:
   - Expand the /user directory and select an individual user directory.
   - Click the plus icon under default ACL.
   - Give hive r-w-x- permissions and save. If hive is not in the drop down, manually add it.

   For system administrators, go to the command line of your host and give hive r-w-x privileges on all /user.

```
## Edit location of Java path as necessary
export JAVA_HOME=/usr/java/jdk1.7.0_67
kinit hdfs
hdfs dfs -setfacl -m -R default:user:hive:rwx /user
hdfs dfs -setfacl -m -R user:hive:rwx /user
```

9. Create an identical ACL to cover any recursive cases.
10. Create roles and ACLs for groups writers (INSERT) and readers (SELECT).

Her, "subrata" = user1 who is a member of sysadmins with sysadmin_role privileges



Deconstruct Hue Actions (#concept_rtj_3nz_p1b)

Now that we have our three groups, let us analyze how the services operate when users take actions.

1. When user2, in group writers, creates a hive table from a file:

```
> hue asks hive to doas user2 and create table with this hdfs /dir/file
> hive asks sentry if user2 can create tables in this database (DB)
> hive asks sentry if user2 has sentry level creds on this /dir/file (URI)
> table gets created as hive (not user2) in hive metastore
> hive asks hdfs if it can move file into table as hive, not user2 (ACL)
```

2. When user2, in group writers, creates, saves, and runs a hive query:

```
> hue asks hive to doas user2 and run this query
> hive asks sentry if user2 has permission to run queries
```

3. When user2, in group writers, creates an oozie job to run the hive query on a schedule:

```
> hue asks oozie to doas user2 and run this job
> oozie does not authorize and runs job as user2
> oozie spawns mapred job that runs hive query as user2
> mapred job asks hive to run this query as user2
> hive asks sentry if user2 has permission to run queries
```

Categories: Administrators (../categories/hub_administrators.html) | Authorization (../categories/hub_authorization.html) | Hue (../categories/hub_hue.html) | Sentry (../categories/hub_sentry.html) | All Categories (../categories/hub.html)

Page generated August 10, 2018.