Cloudera Runtime 7.2.16

Hue Troubleshooting

Date published: 2020-07-28
Date modified: 2023-01-11

CLOUDERA

Legal Notice

Contents

The Hue load balancer not distributing users evenly across various Hue servers

The Hue load balancer redirects the new users to the newly added Hue servers and the existing users to the existing Hue servers on your cluster. Even though you add more Hue servers to meet the growing user base, the resources might not be utilized effectively.

About this task

The Hue load balancer is tasked to evenly distribute users across the available Hue servers for effective resource utilization. However, due to session persistence, it does not distribute users evenly. To overcome this issue, you can refresh the cookies from Cloudera Manager.

The load balancer uses the cookie ROUTEID with a random string from your browser and is stored in the hue.conf file. This random string is used to redirect a user to the Hue server. To refresh the cookie and set a new random string every time you add a new Hue server, do the following:

Procedure

1. Go to Cloudera Manager Clusters Hue Configuration .
2. Click Scope Load balancer and select the Hue Load Balancer Cookie Refresh checkbox.

   This refreshes the cookie value in the hue.conf file to rebalance the Hue backend connections.
3. In the Instances tab, select all the Hue services and roles, and click Action for Services Restart .

   This creates a new random string for the cookie which the load balancer can now use to evenly distribute users.

Results

Whenever you restart the Hue server, the load balancer will redistribute users evenly based on the server load.

Unable to authenticate users in Hue using SAML

If you have configured SAML to authenticate users, but your users are unable to log into Hue using Single Sign On (SSO), then it is possible that the RSA key format is not supported. To resolve this issue, you can use an unprotected private key and then specify the private key filename in the safety valve.

Procedure

1. Convert the .key file to an unprotected private key file by using the following command:

```
openssl rsa -in /opt/cloudera/security/<file name>.key -out /opt/cloudera/
security/<file name_unprotected>.key
```

```
openssl rsa -in /opt/cloudera/security/hadoop-cpi-prod.key -out /opt/cloud
era/security/hadoop-cpi-prod_unprotected.key
```

2. Update the advanced configuration snippet as shown in the following example:

```
[libsaml]
xmlsec_binary=/usr/bin/xmlsec1
metadata_file=/opt/cloudera/security/saml/idp-openam-metadata.xml
key_file=/opt/cloudera/security/hadoop-cpi-prod_unprotected.key
cert_file=/opt/cloudera/security/hadoop-cpi-prod.pem
```

Cleaning up old data to improve performance

Some tables in Hue retain data indefinitely resulting in slower performance or application crashes. Hue does not automatically clean up data from these tables. You can configure Hue to retain the data for a specific number of days and then schedule a cron job to clean up these tables at regular intervals for improved performance.

About this task

Consider cleaning up old data from the backend Hue database if you face the following problems while using Hue:

- Upgrade times out
- Performance is slower than expected
- Long time to log in to Hue
- SQL query shows a large number of documents in tables
- Hue crashes while trying to access saved documents

> **Important:** The clean-up steps only deletes the unsaved documents and workflows. Saved data and information is not cleaned up.

Before you begin

Back up your database before starting the cleanup activity. Check the saved documents such as Queries and Workflows for a few users to prevent data loss. You can also note the sizes of the tables you want to clean up as a reference by running the following queries:

```
select count(*) from desktop_document;
select count(*) from desktop_document2;
select count(*) from beeswax_session;
select count(*) from beeswax_savedquery;
select count(*) from beeswax_queryhistory;
select count(*) from oozie_job;
```

> **Note:** The optimal number of documents that can be stored in a table is less than or equal to 30,000. Consider this number while specifying the cleanup interval.

Procedure

1. SSH into an active Hue instance.
2. Change to the Hue home directory:

```
cd /opt/cloudera/parcels/CDH/lib/hue
```

3. Run the following command as the root user:

```
./build/env/bin/hue desktop_document_cleanup --keep-days x --cm-managed
```

The --keep-days property is used to specify the number of days for which Hue will retain the data in the backend database.

(Optional) Specify DESKTOP_DEBUG=True if you want to log information for troubleshooting purposes.

```
DESKTOP_DEBUG=True ./build/env/bin/hue desktop_document_cleanup --keep-d
ays 30 --cm-managed
```

In this case, Hue will retain data for 30 days.

The logs are displayed on the console because DESKTOP_DEBUG is set to True. Alternatively, you can view the logs from the following location:

/var/log/hue/desktop_document_cleanup.log

The first run can typically take around 1 minute per 1000 entries in each table.

4. Check whether the table size has decreased by running a query as follows:

```
select count(*) from desktop_document;
```

If the desktop_document_cleanup command has run successfully, the table size should decrease.

What to do next
Set up a cron job that runs at regular intervals to automate the database cleanup. For example, you can set up a cron job to run daily and it purges data older than *x* number of days.

Unable to connect to database with provided credential

Cloudera Manager tests the database connection when you add the Hue service to a cluster. The "Test Database Connection" does not work for Oracle databases that require service name instead of the Oracle System ID (SID). This could stop you from adding the Hue service to your cluster.

About this task
If you encounter the following error while adding the Hue service through Cloudera Manager, then follow the workaround as mentioned in this topic:

Unable to connect to database with provided credential. Able   to find the Database server, but not the specified dat abase. Please check if the   database name is correct and make sure that the user can access the   database.

Procedure

1. Install a different database instance, such as MySql temporarily to use with Hue.

This is referred to as the Hue database.

2. Add the Hue service from Cloudera Manager and specify the Hue database details that you created in the previous step.

This will allow you to get past the Add Service wizard and add the Hue service to your cluster.

3. Modify the Hue instance to use the actual Oracle database as follows:

a) Navigate to Cloudera Manager Clusters Hue service Configuration Category Database .

The database configuration fields are displayed.

b) Set up the Oracle database by configuring the following fields:

1. Select Oracle as the Hue Database Type.

2. In the Hue Database Hostname field, specify the Fully Qualified Domain Name (FQDN) of the host on which you have installed the Oracle database.

3. In the Hue Database Port field, specify the port on the host on which the Oracle databse is running. Typically, this value is 1521.

4. In the Hue Database Username field, specify the username to log in to the Oracle database.

5. In the Hue Database Password field, specify the database password.

6. In the Hue Database Name field, specify the name of the Hue database in the following format:

```
<HUE_DB_HOST>:1521/<servicename>
```

c) Navigate to Cloudera Manager Clusters Hue service Configuration Category Advanced and specify the following in the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini field:

```
[desktop]
[[database]]
port=0
```

4. Click Save Changes.

5. Restart the Hue service by clicking Actions Restart .

Activating Hive query editor on Hue UI

You may not see the Hive query editor on the Hue user interface if you have not installed and selected the HIVE_ON_TEZ service on your cluster. The HIVE_ON_TEZ service is needed to configure and use Hive with Hue.

Note:

In CDH 6 and earlier, the Hive service included the Hive Metastore and HiveServer2. In Cloudera Runtime 7.0 and later, this service includes only Hive Metastore. HiveServer2 and other components of the Hive execution engines are part of the HIVE_ON_TEZ service.

About this task

To enable the Hive query editor on the Hue web UI:

Procedure

1. Sign in to Cloudera Manager as an Administrator.

2. Check whether the HIVE_ON_TEZ service is installed on your cluster.

If it is not installed already, add it as a service.

3. Go to Clusters Hue service Configuration .

The list of Hue configurations is displyed.

4. Search for the HiveServer2 Service field and select the HIVE_ON_TEZ service.

5. Save the changes and restart Hue.

Results

The Hive query editor should now be available on the Hue user interface.

Query execution finished in Hue but shows as executing on Cloudera Manager Impala Queries Page

Cloudera Manager and the Impala demon web page may show a query in an "executing" or "In Flight" state even though the query has finished executing on the Hue web UI. This can happen due to various reasons.

The three main reasons why the completed Hue query still shows as "executing" are:

- Hue does not close the connection to Impala until you click on the **Results** page.

  Clicking the **Results** page in Hue executes the fetchresults call to Impala.
- Impala queries are client-driven. Therefore, the query still remains in a running state until the client sends a fetch command to complete fetching the entire result set.
- If a query has not been closed or unregistered, Impala shows the same in the **In Flight** section on its web UI. Cloudera Manager shows all In Flight queries in the "Executing" state.

Impala query life cycle

When you submit Impala queries, they are first registered by the system. The system identifies the queries with the help of a coordinator. They also have a state, such as CREATED, INITIALIZED, RUNNING, FINISHED, EXCEPTION, and some metadata.

- FINISHED implies that the rows are available but not all rows are ready to be fetched. It is possible that Impala daemons are still executing the query.
- EXCEPTION implies that an error has occurred. For example, if the system runs out of memory, then the query transitions to the EXCEPTION state.

  The query can also go into an EXCEPTION state if it is cancelled.

Query cancellations may be triggered explicitly with a HiveServer2/Beeswax call or if the query times out. Query time-out may be set through a process-wide impalad argument or with a per-query option.

Currently, Impala does not have a state that explicitly indicates whether all Impala daemons have finished executing the query and that all results have been fetched. Let us call it as End of Statement (EOS), temporarily.

When a query is in the EOS (FINISHED) or EXCEPTION state, the query is not doing any more processing, but the query remains registered. It needs to remain registered because clients may need to access the state.

The query is unregistered only in the following two cases:

- The query is explicitly closed by a Close() API call
- The session associated with the query is closed explicitly or the session time-out is set and the session times out

> **Note:** Hue does not close a query until you explicitly close it. When you close a browser tab on which you are running the query in Hue, the browser sends a JavaScript Close() callback request to close the query. If you leave the query unattended, for example by closing the laptop through which you are accessing Hue or if the browser crashes, then the Close() call is never sent to Hue. The query may eventually time out, but because it was not cancelled explicitly, it does not properly clean up the resources.

To optimize resource utilization, configure the Impala daemon to stop the idle sessions by setting the session timeout value in the --idle_session_timeout impalad argument:

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Impala service Configuration .

3. Specify the following in the Impala Command Line Argument Advanced Configuration Snippet (Safety Valve) field:

```
--idle_session_timeout=<maximum lifetime of your queries in seconds>
```

For example,

```
--idle_session_timeout=3600
```

In this case, the query will time out after one hour.

Finding the list of Hue superusers

You can fetch the list of superusers by using the Hue shell with Python code or by running a SQL query on the auth _user table.

Using the Hue shell and Python code to find Hue superusers

1. Connecting to Hue shell by running the following command:

```
/opt/cloudera/parcels/CDH/lib/hue/build/env/bin/hue shell --cm-managed
```

2. Enter the Python code as follows:

```
from django.contrib.auth.models import User
print "%s" % User.objects.filter(is_superuser = True)
```

Sample output:

```
<QuerySet [<User: admin>]>
```

Running a SQL query on the auth_user table to find Hue superusers

1. Connect to Hue database shell by running the following command:

```
/opt/cloudera/parcels/CDH/lib/hue/build/env/bin/hue dbshell --cm-managed
```

2. Run the following SQL query:

```
select username, is_superuser from auth_user where is_superuser=1;
```

Sample output:

```
----------------------+
username is_superuser
----------------------+

admin 1
----------------------+
1 row in set (0.00 sec)
```

Unable to access Hue from Knox Gateway UI

If you are unable to access Hue from the Knox Gateway UI, then it is possible that the KnoxSpnegoDjangoBackend property is not configured for Hue. Cloudera Manager does not set the Authentication Backend to KnoxSpnegoDjangoBackend by default even though Knox is enabled on your cluster.

Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service Configurations and search for the Authentication Backend field.
3. Select desktop.auth.backend.KnoxSpnegoDjangoBackend from the dropdown.
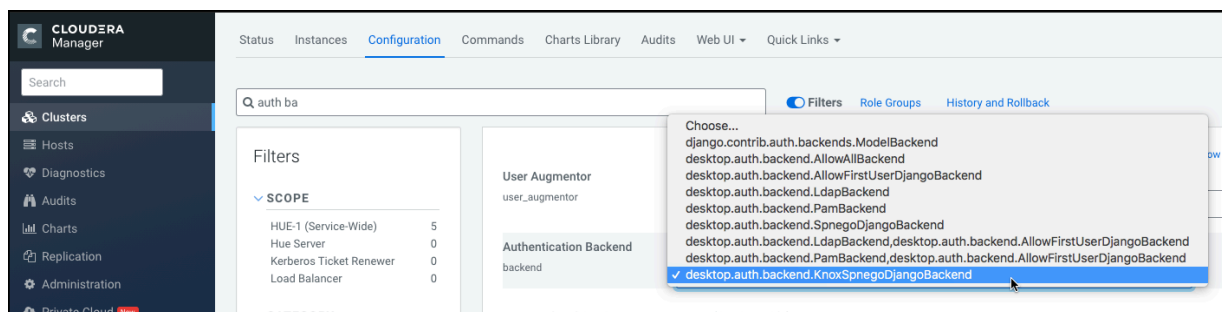
4. Go to Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini) and comment or remove any SAML-specific configurations, if present.

    > ⚠️ **Caution:** Knox-SSO and SAML are incompatible and mutually exclusive. Hue authetication may fail with a redirection loop if you have the Knox-SSO and SAML configurations present in the Hue Advanced Configuration Snippet at the same time, as it confuses the authentication redirect to the IdP and back to Hue.

5. Click Save Changes.
6. Go to Clusters $Knox service Instances and note down the hostnames of the Knox Gateways.

    You must provide these details in the next step.

    If you have set up Knox in High-Availablity (HA) mode, then you can see more than one Knox Gateways listed on the Instances tab.

7. Go back to Clusters Hue service Configurations and search for the Knox Proxy Hosts field.
8. Enter the hostname of the Knox Gateway that you noted earlier.

    If you have set up Knox HA, then click + to add another hostname.
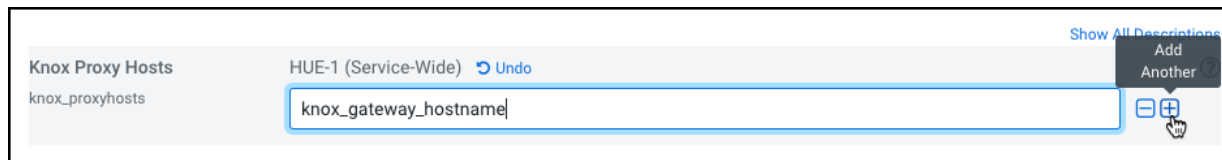
9. If you have deployed a Hue Load Balancer, then you must specify the Load Balancer hostname in the Knox Proxy Hosts field by clicking +.
10. Click Save Changes.

    You would see the following warning:

    ```
    Role is missing Kerberos keytab. Go to the Kerberos Credentials page and
      click the Generate Missing Credentials button.
    ```

11. Click Administration on the Cloudera Manager left navigation panel and select Security.

12. Go to the Kerberos Credentials tab and click Generate Missing Credentials.
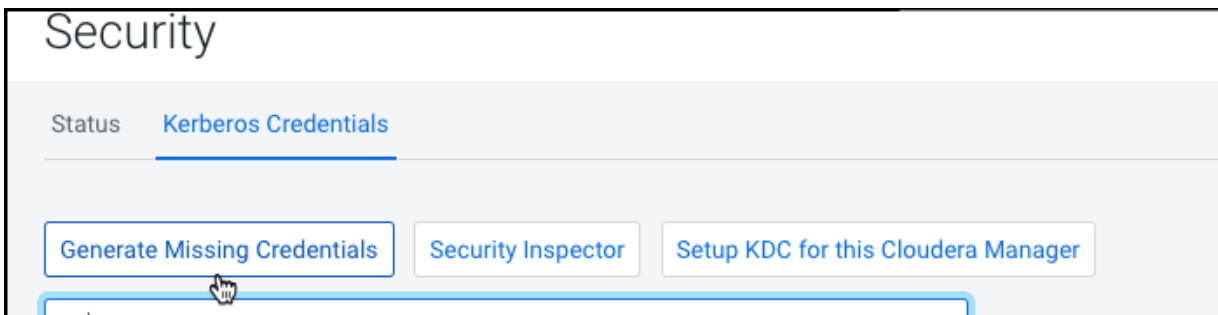
A pop-up showing the status is displayed.
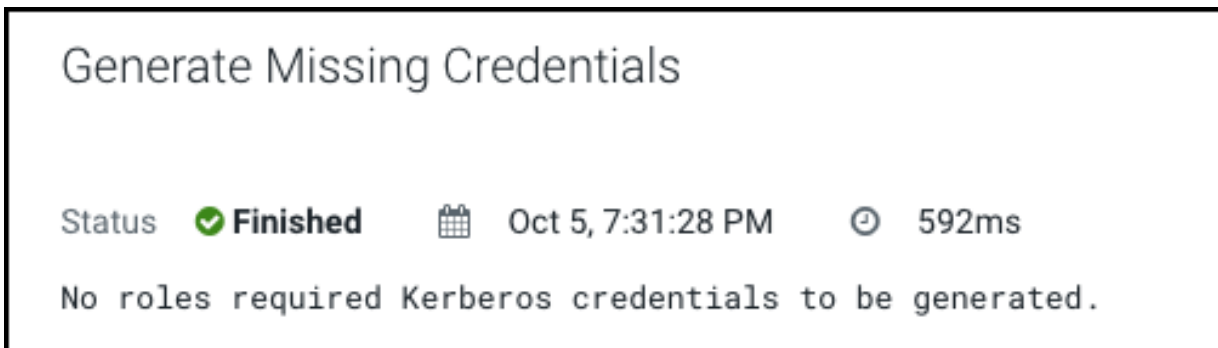
13. Go to Clusters Hue service and click Restart next to Actions.
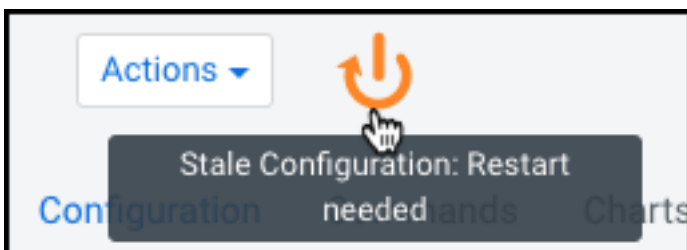
14. On the **Stale Configurations** page, click Restart Stale Services.

The **Restart Stale Services** wizard is displayed.

15. On the Review Changes page, select Redeploy client configuration, and click Restart Now.

The **Command Details** page shows the live status as the service restarts.

When all the steps are complete, click Finish.

16. From the Hue service page, click Web UI Knox Gateway UI .
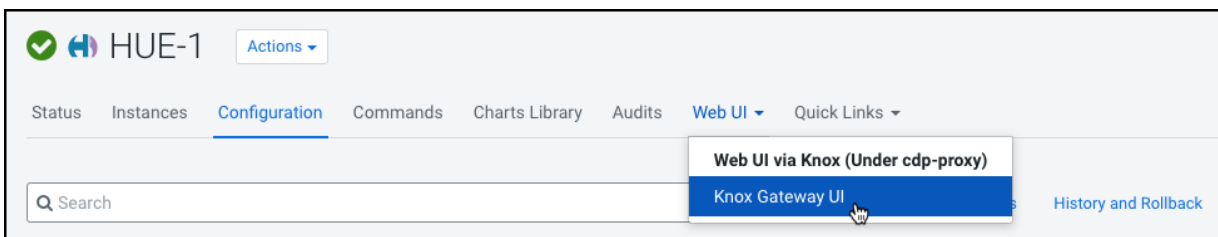
The Knox Gateway UI is displayed.

17. On the **General Proxy Information** page, expand the CDP Proxy topology by clicking + cdp-proxy under Topologies.

The list of services that are configured with the cdp-proxy topology is displayed.

18. Click on the Hue logo.
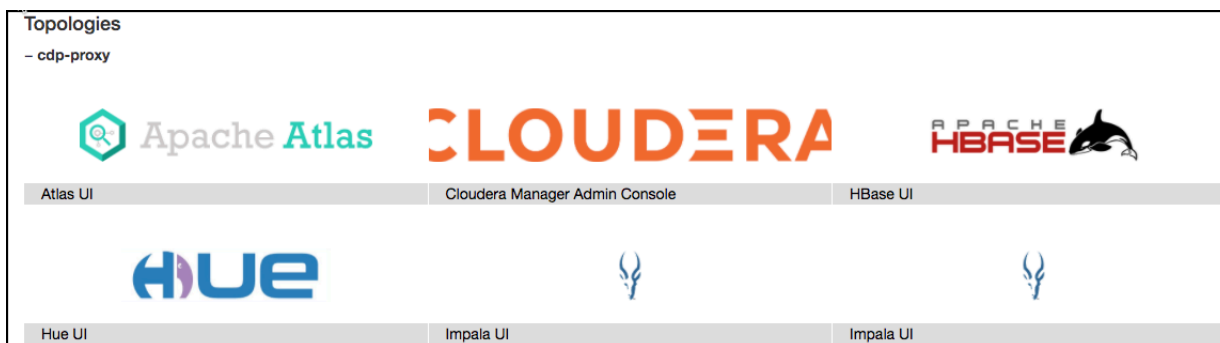
You should be able to log in to the Hue web UI.

You can also log into Hue using the following URL:

```
https://[***HOSTNAME***]:[***PORT***]/gateway/cdp-proxy/hue/
```

19. Go to Clusters Knox Configuration and add the following entries in the Knox Simplified Topology Management - cdp-proxy-api field:

```
HUE:httpclient.socketTimeout=[***TIMEOUT-IN-MINUTES***]
```

```
HUE:httpclient.connectionTimeout=[***TIMEOUT-IN-MINUTES***]
```

Replace *[***TIMEOUT-IN-MINUTES***]* with the actual timeout value depending on the load on your load on the cluster or environment. For example, 20 minutes.

> **Note:** This step is required to prevent query timeouts, and if you are seeing errors such as "Results have expired, rerun the query if needed" in the Hue logs.

20. Restart the Knox service.

Unable to view Snappy-compressed files

You must install the python-snappy library on your cluster to view files compressed with Snappy using the Hue File Browser and the HBase Browser. Post-installation, Hue automatically detects and displays the Snappy-compressed files.

Before you begin

The python-snappy library is incompatible with the python library called snappy. You must uninstall snappy if it is present on your cluster.

Run the following command to check whether the snappy library is installed on your cluster:

```
/usr/bin/pip show snappy
```

No output on the console indicates that the snappy library is not installed on your cluster. If you get any results for snappy, then uninstall it by running the following command:

```
/usr/bin/pip uninstall snappy
```

Next, check whether you have the python-snappy library is installed on your cluster by running the following command:

```
/usr/bin/pip show python-snappy
```

Sample output:

```
Name: python-snappy
Version: 0.5.4
Location: /usr/lib64/python2.7/site-packages
```

Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Stop the Hue service by going to Cluster Hue service Action .
3. Change to the following directory depending on whether you have used parcels or packages to set up your CDH cluster.

   For parcels:

   ```
   cd /opt/cloudera/parcels/CDH/lib/hue
   ```

   For package:

   ```
   cd /usr/lib/hue
   ```

4. Install the python-snappy package by running the following commands:

   ```
   yum install gcc gcc-c++ python-devel snappy-devel
   ./build/env/bin/pip install -U setuptools
   ./build/env/bin/pip install python-snappy
   ```

5. Verify that the python-snappy library is readable by all users by running the following commands:

   ```
   ls -lart `locate snappy.py`
   ```

   The output should be similar to the following:

   ```
   -rw-r--r-- 1 root root 11900 Sep  1 12:25 /usr/lib64/python2.7/site-pack
   ages/snappy.py
   -rw-r--r-- 1 root root 10344 Sep  1 12:26 /usr/lib64/python2.7/site-packa
   ges/snappy.pyc
   ```

6. Start the Hue service by going to Cluster Hue service Action .
7. Verify that the python-snappy library is working for Hue by running the following command:

   ```
   sudo -u hue /bin/bash -c "echo 'import snappy' | python"
   ```

   If the python-snappy library is working as expected, then no output is displayed for this command.

Results

You should be able to view Snappy-compressed files on the Hue File Browser and the HBase Browser using the Hue web interface.

Impala query fails with invalid query handle error

You encounter an "Invalid query handle" error when running Impala queries from the Hue web interface because the connection between Impala Thrift server and the Hue Load Balancer times out. This is governed by the server_conn_timeout property.

About this task

The default value of the server_conn_timeout property is 30 minutes. You can increase the timeout limit by updating the Hue configuration using Cloudera Manager.

Procedure

1. Log into Cloudera Manager as an Administrator.
2. Go to Clusters Hue service Configuration Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini .
3. Increase the value of the server_conn_timeout property in the Impala section as follows:

```
[impala]
    server_host=[***SERVER-HOST***]
    server_port=[***PORT***]
    server_conn_timeout=[***TIMEOUT-IN-SECONDS***]
```

   You can increase the value of the server_conn_timeout property to 2 hours (7200 seconds).
4. Click Save Changes.
5. Restart the Hue service.

Results

You should be able to run Impala queries from the Hue web interface successfully.

Services backed by PostgreSQL fail or stop responding

When the number of connections between the CDP services and the PostgreSQL database exceeds the preset connection limit, it is possible that a new connection fails, Cloudera Manager stops responding, and you cannot log into Hue. The logs show a "FATAL: remaining connection slots are reserved for non-replication superuser   connections" error.

About this task

The number of connections between the CDP services and the PostgreSQL database is governed by the max_connections setting. By default, the maximum number of available connections to your PostgreSQL database is 115. 15 connections are reserved for the superuser to maintain the state and integrity of your database, and 100 connections are available for CDP and other services.

> **Note:** There is a higher probability for this issue to occur in Hue because Cloudera Manager starts the Hue service after starting other CDP services. Therefore, the Hue service gets relatively fewer connections to PostgreSQL as compared to other services sharing the same database.

Procedure

1. Check the number of available and idle connections:
    a) SSH into the PostgreSQL database from the command-line client psql as an admin user.
    b) Run the following query to check the number of idle connections:

    ```
    SELECT datname, count(datname) FROM pg_stat_activity WHERE state = 'idle
    ' GROUP BY datname;
    ```

    c) Run the following query to check the number of connections currently in use:

    ```
    SELECT datname, count(datname) FROM pg_stat_activity GROUP BY datname;
    ```

    d) Run the following command to view the maximum number of connections:

    ```
    show max_connections;
    ```

    e) Run the following query to know where the connections are going:

    ```
    SELECT datname, numbackends FROM pg_stat_database;
    ```

2. If most connections are idle and the max_connections value is less than 100, then increase the max_connections value in the postgresql.conf file:
    a) Log into Cloudera Manager and stop all services that use the PostgreSQL database.
    b) SSH into the host on which the PostgreSQL server is running.
    c) Open the postgresql.conf file for editing.

    The postgresql.conf file is typically present in the /var/lib/pgsql/data directory. But this may vary depending on where you have installed the database.

    d) Increase the value of max_connections as per the following recommendation:

    Allow a maximum of 100 connections for each database and add 50 extra connections. For example, for two databases, set the maximum connections to 250.

    If you store five databases on one host (the databases for Cloudera Manager Server, Activity Monitor, Reports Manager, Apache Atlas, and Hive Metastore), set the maximum connections to 550.

    e) Save the changes and exit.
    f) Restart the PostgreSQL database by running the following command:

    ```
    pg_ctl restart
    ```

    g) Restart all the affected services from Cloudera Manager.

What to do next

If increasing the connection limit does not solve your problem and you see a need to scale up, then add new PostgreSQL instances on other hosts and migrate the services to those hosts with the help of your Database Administrator (DBA).

Invalid method name: 'GetLog' error after submitting Hive queries

The Invalid method name: 'GetLog' (code THRIFTAPPLICATION): None error can occur after you submit Hive queries from the Hue editor and while Hue tries to fetch the result set. If you encounter this error, then set the beeswax use_get_log_api property to false using Cloudera Manager.

Procedure

1. Log into Cloudera Manager as an Administrator.

2. Go to Clusters Hue service Configuration .

3. Add/update the beeswax section as follows in the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini field:

```
[beeswax]
use_get_log_api=false
```

4. Click Save Changes.

5. Restart the Hue service.

"Authorization Exception" error on submitting queries in Hue

If you have secured your cluster using Ranger, then you must grant the required permissions to your users and groups from the Ranger web UI. If your users do not have proper permissions, then they may not be able to access certain databases or tables from the Hue editor.

About this task

If your users see the "Authorization Exception: User does not have privileges to execute..." error on submitting queries from the Hue editor, then grant them the proper permissions using the Ranger web UI.

Procedure

1. Log into Cloudera Manager as an administrator.

2. Go to Clusters Ranger service Instances tab and note down the hostname corresponding to the "Ranger Usersync" role type.

3. Open the Ranger web UI by clicking Ranger Admin Web UI.

4. SSH into the Ranger Usersync host that you noted in step 2 and add the user or the group as follows:

```
ssh root@example.domain.site useradd [***USERNAME/GROUP-NAME***] passwd
[***PASSWORD***]
```

5. On the Ranger web UI, click Hadoop SQL listed under the HADOOP SQL service.

The **Hadoop SQL Policies** page is displayed.

6. On the **Hadoop SQL Policies** page, you can grant the new user access to all the databases or to specific databases by adding a new policy.

- To grant the permission on all databases:

  a. Click the policy ID corresponding to "all - database, table,   column".

  b. On the **Edit Policy** page, add the user whom you want to grant the permission in the Select User field under the Allow Conditions section as shown in the following image:

  To grant permissions to a group, enter the group name in the Select Group field.

  c. Click Save.

- To grant permission on specific database:

  a. Click Add New Policy.

  The **Create Policy** page is displayed.

  b. Under the Policy Details section, specify the policy name and select the database, table, and column that you want your user to access as shown in the following image:

  c. Under the Allow Conditions section, enter the username in the Select User field and click Add Permissions and select the permissions that your user must have.

> To grant permissions to a group, enter the group name in the Select Group field.
>
> **d.** Click Add.

7. Start the Hue service from Cloudera Manager.

Results

The user or the group should be able to run any query on any entities as defined in the policy.

Cannot alter compressed tables in Hue

Due to a known bug in the Oracle database (12c and higher), you cannot perform ALTER TABLE operations (add, delete, drop, modify) on compressed tables. If you have compressed tables in the Hue schema, then you may see the "ORA-39726: unsupported add/drop column operation on compressed tables" error.

About this task

Even if you uncompress an existing table, you may not be allowed to alter the columns. To resolve this issue:

Procedure

1. SSH into the host on which you have installed the Oracle database.

2. Create a new uncompressed table with the same structure as the compressed table.

3. Copy the data from the compressed table to the new uncompressed table.

4. Rename or delete the compressed table.

5. Rename the uncompressed table with the name of the original compressed table.

Results

You should now be able to perform ALTER TABLE operations (add, delete, drop, modify) on the Hue tables.

MySQL: 1040, 'Too many connections' exception

If Hue displays the "1040, Too many connections" exception, then it is possible that the Hue backend database is overloaded and out of maximum available connections. To resolve this issue, you can increase the value of the max_connections property for your database.

About this task

The 1040, 'Too many connections' exception occurs on a MySQL database when it runs out of maximum available connections. If you are using the Impala engine, you may see the following error message on the Hue web interface: OperationalError   at /desktop/api2/context/computes/impala("1040: too many connections"). A similar error may be displayed for Hive. The exception is also captured in the Hue server logs.

Before you begin

The max_connections property defines the maximum number of connections that a MySQL instance can accept. Uncontrolled number of connections can crash the server. Following are some guidelines for tuning the value of the max_connections property:

• Set the value of the max_connections property according to the size of your cluster.

• If you have less than 50 hosts, then you can store more than one database (for example, both the Activity Monitor and Service Monitor) on the same host. If you have more than 50 hosts, then use a separate host for each database/ host pair. The hosts need not be reserved exclusively for databases, but each database must be on a separate host.

- For less than 50 hosts:

  - Place each database on its own storage volume.
  - Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases, set the maximum connections to 250. If you store five databases on one host (the databases for Cloudera Manager Server, Activity Monitor, Reports Manager, Atlas, and Hive MetaStore), then set the maximum connections to 550.

To increase the number of maximium available connections and to resolve the "1040, Too many connections" exception:

Procedure

1. Log in to Cloudera Manager and stop the Hue service.
2. SSH in to your database instance as a root user.
3. Check the number of available connections by running the following command:

```
grep max_conn /etc/my.cnf
```

/etc/my.cnf is the default location of the options file (my.cnf).

4. Set the new value of the max_connections property from the MySQL shell as per the guidelines provided above. For example:

```
mysql> SET GLOBAL max_connections = 550;
```

5. Restart the Hue service.

Increasing the maximum number of processes for Oracle database

While using Oracle as a backend database for Hue, if you face issues connecting to the Hue service after restarting the database, then it is possible that the Hue service is not able to get a new database connection. The following error in the Hue logs indicates that the maximum number of connections have exhausted: "ORA-12519: TNS:no appropriate service handler found". This can be resolved by increasing the number of available processes.

After restarting the Oracle database, if you are not able to connect to the Hue service, check the Hue logs for the ORA-12519: TNS:no appropriate service handler found error. If you see the ORA-12519: TNS:no appropriate service handler found error in the logs, then work with your database administrator to check whether the maximum number of processes have exceeded. If the maximum number of processes have exceeded, then you see the following error: ORA-00020: maximum number of processes exceeded. Increase the number of processes to resolve this issue.

How to calculate the number of database processes, transactions, and sessions?

Cloudera recommends that you allow 100 maximum connections for each service that requires a database and then add 50 extra connections. For example, for two services, set the maximum connections to 250. If you have five services that require a database on one host (the databases for Cloudera Manager Server, Activity Monitor, Reports Manager, Cloudera Navigator, and Hive metastore), set the maximum connections to 550.

From the maximum number of connections, you can determine the number of anticipated sessions using the following formula:

```
sessions = (1.1 * maximum_connections) + 5
```

For example, if a host has a database for two services, anticipate 250 maximum connections. If you anticipate a maximum of 250 connections, plan for 280 sessions.

Based on the number of sessions, you can determine the number of anticipated transactions using the following formula:

```
transactions = 1.1 * sessions
```

Continuing with the previous example, if you anticipate 280 sessions, you can plan for 308 transactions.

Work with your Oracle database administrator to apply these derived values to your system.

Using the sample values above, Oracle attributes would be set as follows:

```
alter system set processes=250;
alter system set transactions=308;
alter system set sessions=280;
```

UnicodeDecodeError: UTF-8 codec cannot decode byte in position

You may see an error such as the following while querying a table using the Impala editor in Hue: "'utf8' codec can't decode byte 0x91 in position 6: invalid start   byte". This occurs because Hue does cannot handle columns containing non-UTF-8 data.

To resolve this issue, contact Cloudera Support to apply the following software patches: CDPD-22129 and CDPD-27412.

UnicodeDecodeError: 'ascii' codec cannot decode byte in position

You may see an error such as the following while downloading Impala query results in CSV format from Hue: "Unic odeDecodeError: 'ascii' codec can't decode byte 0xc2 in   position 49: ordinal not in range(128)". This occurs when the query results contain special characters of non-ASCII encoding.

About this task

Note: You may not face this issue if you export the query result from Hue to Excel or if you are querying the table using the Hive editor.

In CDP, Hue is compatible with Python 2.7.x, but the Tablib library for Hue has been upgraded from 0.10.x to 0.14.x, which is generally used with the Python 3 release. If you try to download Impala query results having special characters in the result set in a CSV format, then the download may fail with the ASCII unicode decode error. To fix this issue, downgrade the Tablib library to 0.12.x as follows:

Procedure

1. SSH into the Hue server host.

2. Change directory to the following:

```
cd /opt/cloudera/parcels/CDH-7.x/lib/
```

3. Back up the hue directory.

```
cp -R hue hue_orginal
```

4. Change to the hue directory.

```
cd hue
```

5. Install the Wheel package using pip.

```
./build/env/bin/pip install wheel
```

The Wheel package is used to avoid recompiling your software during every install.

6. Install the Python Setuptools package for Hue:

```
./build/env/bin/pip install setuptools==44.1.0
```

7. Install Tablib version 0.12.1:

```
./build/env/bin/pip install tablib==0.12.1
```

8. Update the permissions of the Tablib library and other packages that were installed during the process by using the following command:

```
chmod 0755 [***TABLIB-LIBRARY-NAME***]
```

⚠️ **Attention:** The Tablib library and other packages should have the drwxr-sr-x permission. Failing to have this permission can result into an "tablib module not found" error.

9. Go to Cloudera Manager and restart the Hue service.

Fixing authentication issues between HBase and Hue

An HBase feature improvement to the Thrift Server (HBASE-19852) may cause authentication issues between HBase and Hue, and you may see the following error while accessing the HBase tables from Hue: "Failed to authenticate to HBase Thrift Server,   check authentication configurations."

About this task

You may also see the following error in the Hue logs: "RestException: Unable to authenticate <Response [401]>". To fix this issue, update the HBase configurations using Cloudera Manager.

Procedure

1. Log in to Cloudera Manager as an Administrator.

2. Go to Clusters HBase service Instances and note the hostname of the host on which the HBase Thrift Server is running.

If multiple Thrift Servers are configured, then find the one that Hue is configured to use.

3. Go to Configuration HBase Thrift Server Advanced Configuration Snippet (Safety Valve) for hbase-site.xml and add the following properties:

Field	Property
Name	hbase.thrift.spnego.principal
Value	HTTP/[***HOSTNAME-FROM-STEP2***]@REALM Substitute @REALM with the actual Kerberos realm.
Name	hbase.thrift.spnego.keytab.file
Value	hbase.keytab

4. Select the following options to enable the properties:

- Enable HBase Thrift Http Server
- Enable HBase Thrift Proxy Users

5. Deselect the following properties:

- Enable HBase Thrift Server Compact Protocol
- Enable HBase Thrift Server Framed Transport

6. If you have not enabled SSL on your cluster, but if you have Kerberized your cluster, then add the following HBase configurations, without which you may encounter a NullPointerException error while starting the HBase Thrift Server:

a) Add the following properties in the HBase Client Advanced Configuration Snippet (Safety Valve) for hbase-site.xml field:

Field	Property
Name	hbase.thrift.ssl.enabled
Value	false

b) Add the following properties in the HBase Thrift Server Advanced Configuration Snippet (Safety Valve) for hbase-site.xml field:

Field	Property
Name	hbase.thrift.ssl.enabled
Value	false

7. Click Save Changes.

8. Go to Clusters Hue service Configuration Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini and add the following lines:

```
[hbase]
thrift_transport=buffered
```

9. Click Save Changes.

10. Restart the HBase and Hue services to apply stale configurations.

Hue Load Balancer does not start due to lengthy BalancerMember Route length

The Hue Load Balancer may not start if the route name in the ROLES table exceeds 64 characters. You must manually reduce the length of the route name for each Hue server in the ROLES table to be less than or equal to 64 characters, and also ensure that it is unique.

About this task

Cloudera Manager creates the Role name in the following format, which may sometimes cause the route name to exceed 64 characters: "Service-name-Role-name-Cluster-name-Unique-hash". You may see the following error while starting the Hue Load Balancer:

```
BalancerMember Route length must be < 64 characters
+ '[' 1 '!=' 0 ']'
+ die '/var/run/cloudera-scm-agent/process/482-hue-HUE_LOAD_BALANCER/http
d.conf is invalid.'
+ echo '/var/run/cloudera-scm-agent/process/482-hue-HUE_LOAD_BALANCER/httpd.
conf is invalid.'
```

```
/var/run/cloudera-scm-agent/process/482-hue-HUE_LOAD_BALANCER/httpd.conf is
  invalid.
```

To resolve this issue, modify the route name column in the ROLES table in the scm database for every Hue server to be less than or equal to 64 characters, and ensure that it is unique.

Procedure

1. SSH into the database instance as an administrator.
2. Query the ROLES table to view table content:

```
SELECT * FROM `ROLES`;
```

3. Update the values in the "NAME" column so that the route name is less than or equal to 64 characters:

```
UPDATE ROLES SET NAME='[***HUE-ROLE-NAME***]' WHERE ROLE_ID=[***ROLE-ID-
NUMBER***];
```

```
UPDATE ROLES SET NAME='hue-6c02f47dbd7e181d293c078ea293f3da' WHERE ROLE_
ID=10;
```

Enabling access to HBase browser from Hue

If HBase impersonation is not allowed in HDFS, then you may get an API error while trying to access the HBase browser from Hue. To resolve this issue, you must allow proxy users on the Thrift gateway, and also allow all groups form all hosts in HDFS to impersonate the hbase user.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters HBase service Configuration and search for the hbase.thrift.support.proxyuser property.
3. Enable the Enable HBase Thrift Proxy Users option.
4. Click Save Changes.
5. Go to Clusters HDFS service Configuration .
6. Enter the following lines in the Cluster-wide Advanced Configuration Snippet (Safety Valve) for core-site.xml field:

```
<property>
<name>hadoop.proxyuser.hbase.hosts</name>
<value>*</value>
</property>
<property>
<name>hadoop.proxyuser.hbase.groups</name>
<value>*</value>
</property>
```

7. Click Save Changes.
8. Restart the HBase and HDFS services.

Fixing a warning related to accessing non-optimized Hue

If you are using a third-party load balancer for Hue, then you may see a warning such as "You are accessing a non-optimized Hue, please switch to one of the available addresses: https://[***LB-HOST-FQDN***]:[***LB-PORT***]". To fix this issue, you must add the URL of the Hue instance for which you are seeing the warning to the list of hue load balancers in the Advanced Configuration Snippet.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service Configuration Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini and add the following line under the [desktop] section:

    ```
    [desktop]
    hue_load_balancer=https://[***LB-HOST-FQDN***]:[***LB-PORT***]
    ```

    You can specify a comma-separated list of URLs for which you see the warning in the hue_load_balancer property.
3. Click Save Changes.
4. Restart the Hue service.

Hue load balancer does not start after enabling TLS

If the Hue load balancer does not start after you enable TLS, then check whether your TLS private key file is password protected. To resolve this issue, you must configure the Hue load balancer to use the corresponding key password using Cloudera Manager.

About this task

At startup, the Hue load balancer reads the private key file as defined in the Cloudera Manager Clusters Hue service Configuration Hue Load Balancer TLS/SSL Server Private Key File (PEM Format) configuration property. If the TLS private key file is password protected, then you may see error such as Fatal error initialising mod_ssl, exiting and Some of your private key files are encrypted for security reasons. In order to read them you have to provide the pass phrases. in the /var/log/hue-httpd/error_log Hue load balancer log file.

Procedure

1. SSH into the Hue host as an Administrator.
2. Create a password file in your chosen security directory and insert the private key password, as shown in the following example:

    ```
    # echo "abc123" > /etc/security/password.txt
    # chown hue:hue password.txt
    # chmod 700 password.txt
    ```

3. Log in to Cloudera Manager as an Administrator.
4. Go to Clusters Hue service Configuration and enter the file path and the filename of the password file in the Hue Load Balancer TLS/SSL Server SSLPassPhraseDialog field.
5. Click Save Changes.
6. Restart the Hue service.

Unable to log into Hue with Knox

Opening Hue from Cloudera Manager or Knox Gateway Home may fail if the Kerberos principal name for the Knox service is different from the default Kerberos principal defined in the hue.ini file. To resolve this issue, you must set the value of the knox_principal property to your custom Kerberos principal name for Knox in the Hue Advanced Configuration Snippet, so that Hue's login mechanism (KnoxSpenegoDjangoBackend) can authenticate using Knox's Kerberos principal name.

About this task

If logging into Hue fails due to a mismatch between the custom Kerberos principal name for Knox and the default Kerberos principal name defined in the hue.ini file, then you may see the following error in the access.log file: Failed to verify provided username set(['*KNOX-PRINCIPAL-NAME*']) with set(['knox']).

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Note the Kerberos principal name for the Knox service from Clusters Knox service Configuration Kerberos Principal .
3. Go to Clusters Hue service Configuration Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini and enter the following lines:

```
[desktop]
[[knox]]
knox_principal=[***KNOX-PRINCIPAL-NAME***]
```

   Replace [***KNOX-PRINCIPAL-NAME***] with the Kerberos principal name for the Knox service that you obtained earlier.
4. Click Save Changes.
5. Restart the Hue service.

Results
You should be able to log into Hue from the Knox Gateway UI cdp-proxy topology.

LDAP search fails with invalid credentials error

LDAP authentication fails with an "Invalid credentials" error, even if you input valid login credentials on the Hue login page, and you are unable to log into Hue. To resolve this issue, verify and update the LDAP Bind User credentials using Cloudera Manager.

About this task

This issue may happen if the credentials for the LDAP Bind User for Hue configured in Cloudera Manager are invalid. The invalid credentials could either be the "LDAP Bind Password" or "LDAP Bind User Distinguished Name". If the credentials are valid and the issue persists, verify that LDAP Search Base option in Cloudera Manager Hue Configurations is valid. The LDAP search base should be similar to 'dc=hadoop,dc=mycompany,dc=com'.

Before you begin

This task assumes that the Use Search Bind Authentication option is enabled in Cloudera Manager Hue Configurations . Search Bind Authentication connects to the LDAP server using the credentials provided in the 'bind_dn' and 'bind_password' configurations. If these configurations are not set, then an anonymous search is performed.

If the Use Search Bind Authentication option is not enabled in Cloudera Manager Hue Configurations , then do not set the LDAP Bind User credentials as described in this task. You must use the LDAP Username Pattern field for configuring the LDAP credentials, and verify whether the authentication works as expected.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue Configurations .
3. Set the LDAP Bind User credentials in the following fields:

   • LDAP Bind User Distinguished Name
   • LDAP Bind Password

   You can specify the LDAP Bind User Distinguished Nameeither in the generic LDAPv3 Distinguished Name ("CN=binduser,OU=users,DC=Example,dc=com") format or the Active Directory style (binduser@EXAMPLE.COM) format.
4. Click Save Changes.
5. Restart the Hue service.

Unable to execute queries due to atomic block

Condition

You may see the following error after submitting a query from Hue, while logging into Hue, or while saving documents and workflows: "TransactionManagementError: An error occurred in the current transaction. You can't execute queries until the end of the 'atomic' block."

Cause

If there is a load on Hue's backend database and slower processing of operations such as saving documents or workflows, running INSERT or UPDATE queries from multiple users, then the database puts an atomic block on the table rows. If you are using MySQL or MariaDB as the backend database for Hue, then you can increase the value of the innodb-lock-wait-timeout parameter along with a few other timeout parameters in the my.cnf file.

Solution

Procedure

1. SSH into the database host as an Administrator.
2. Back up the my.cnf configuration file as follows:

   ```
   cp /etc/my.cnf  /[***BACKUP-DIRECTORY***]
   ```
3. Open the my.cnf file for editing and add the following two lines under the [mysqld] section:

   ```
   vi /etc/my.cnf
   ```

   ```
   [mysqld]
   wait_timeout = 28800
   interactive_timeout = 28800
   ```
4. Save the file and exit from the editor.
5. Restart the database server.

Disabling the web metric collection for Hue

Condition

You see the following error in the Hue server logs: DatabaseError: ORA-01000: maximum open cursors exceeded, when you are using Oracle as the backend database for Hue.

Cause

Cloudera Manager Agent collects web metrics from the Hue server to monitor its health. At times, the number of metrics that are collected can exhaust the number of available cursors on the Oracle database, resulting in the ORA-01000: maximum open cursors exceeded error.

Solution

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue Configuration and deselect the Web Metric Collection option.

   > **Note:** Restarting the Hue service from time to time can also clear the cursors, but temporarily. For a long-term resolution, consider turning off the Web Metric Collection option or reducing the frequency of the metric collection using the Web Metric Collection Duration option in Cloudera Manager.

3. Click Save Changes.
4. Restart the Hue service.

Resolving "The user authorized on the connection does not match the session username" error

Condition

In CDP, if you try to open two Hue sessions on different ports on the same browser window, then you may see the "The user authorized on the connection does not match the session username" error on the Hue web interface and in the Hue logs.

Cause

On CDH, if you opened Hue web URL on port 8889 on one tab and on port 8888 on another tab, you would be logged out of Hue on the port 8889, and you would be forced to refresh the page. In CDP, Hue depends on the ZooKeeper quorum for how it connects to Impala or Hive and how the Hue load balancer distributes the connections to the Hue server nodes and controls switching between the different sesions.

Solution

After migrating/upgrading from CDH to CDP, note this change in behaviour and always open Hue web UI using the Hue Load Balanced - recommended option from Cloudera Manager.