Cloudera Runtime 7.2.8

Administering Hue

Date published: 2020-08-10
Date modified: 2021-03-25

https://docs.cloudera.com/

Legal Notice

Contents

Reference architecture

Hue server can support approximately 25 concurrent users, depending on what tasks the users are performing. Most scaling issues occur as a result of users performing resource-intensive operations and not from the number of users. For example, large downloads of query results can impact resource availability for the other users who are using the same Hue instance during the download operation. During that time, the users can experience slow performance. Another common cause of noticeable performance changes are slow RPC calls between Hue and another service. When this happens, queries may appear to suddenly "stop responding" after they are submitted.

As a guide, 2 Hue servers can support up to:

- 100 unique users per week
- 50 users per hour at peak times executing up to 100 queries

A typical setup is 2 Hue servers.

General guidelines

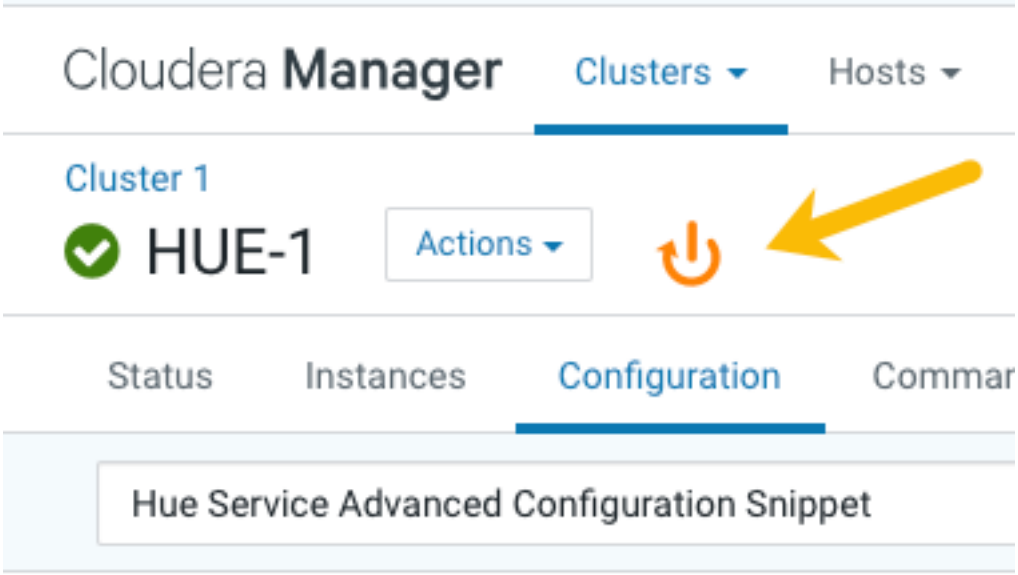
- Deploy a load balancer in front of Hue.
- Use a production-quality database.
- Ensure that other services, such as Impala, Hive, and Oozie, are healthy and not impacted by too few resources. If these services are hanging, it adversely affects Hue performance.
- Consider moving workloads that are subject to SLAs (service-level agreements) or considered "noisy neighbors" to their own compute cluster. Noisy neighbors are workloads that use the majority of available resources and cause performance issues.
- Limit the number of rows that are returned for queries.

  One way to limit the number of rows returned is to specify a value for the download_row_limit configuration property for the Hue Beeswax application. This property can be set in the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini property in Cloudera Manager:

  1. In Cloudera Manager, click HueConfiguration , and enter Hue Service Advanced Configuration Snippet in the search text box.
  2. In the text box for the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini, add the following configuration information:

     ```
     [beeswax]
     download_row_limit=NUMBER_OF_ROWS
     ```

3. Click Save Changes and click the restart icon at the top of the page to restart the Hue service:



Hue configuration files

Hue roles are configured with the following three configuration files: hue.ini, hue_safety_valve.ini, and hue_safety_valve_server.ini.

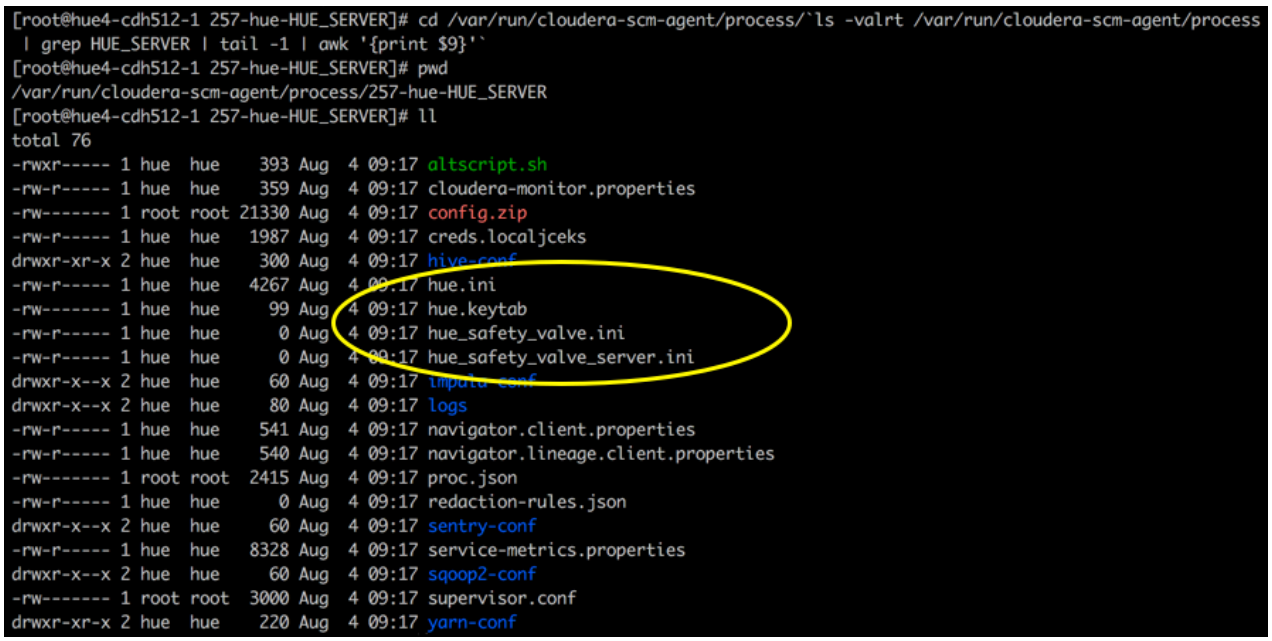
The hue.ini file is the first file that is auto-generated when you add the Hue service to your CDP cluster using Cloudera Manager. The hue_safety_valve.ini file is used to override bugs or configure properties that are missing from the Cloudera Manager UI. The hue_safety_valve_server.ini file is used to configure only the Hue role, and not the full Hue service (which includes the Hue Load Balancer). The hue_safety_valve_server.ini file is not used in practice.

Cloudera recommends that you do not edit the .ini files from the command line because they are stored in dynamic directories named by process ID and populated from the Cloudera Manager database. To add configurations that you cannot add directly from the Cloudera Manager UI, such as Authentication Backend for SAML, use the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini field under the Hue service configurations in Cloudera Manager.

Run the following command to view the .ini files per process ID:

```
ls -ltr /var/run/cloudera-scm-agent/process/`ls -valrt /var/run/cloudera-scm
-agent/process | grep HUE_SERVER | tail -1 | awk '{print $9}'`
```

Figure 1: Terminal showing Hue configuration files



The process directory for any given role is mirrored in Cloudera Manager. Go to Hue Instances , select a role such as Hue Server, and then click the Processes tab.

Related Information

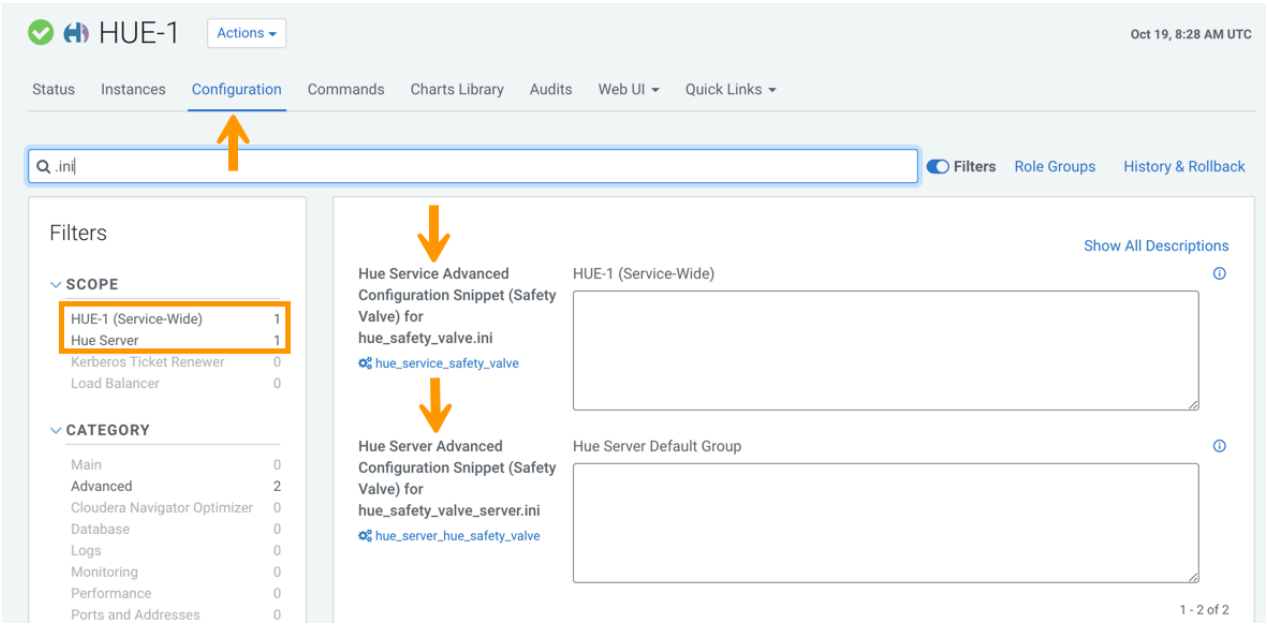
hue.ini

Introduction to Hue Advanced Configuration Snippet (Safety valves)

To customize and configure Hue properties, such as authentication or enabling S3 or ABFS browsers, and so on that you cannot directly configure from the Cloudera Manager UI, you can use the Advanced Configuration Snippet field under the Hue cluster configurations in Cloudera Manager.

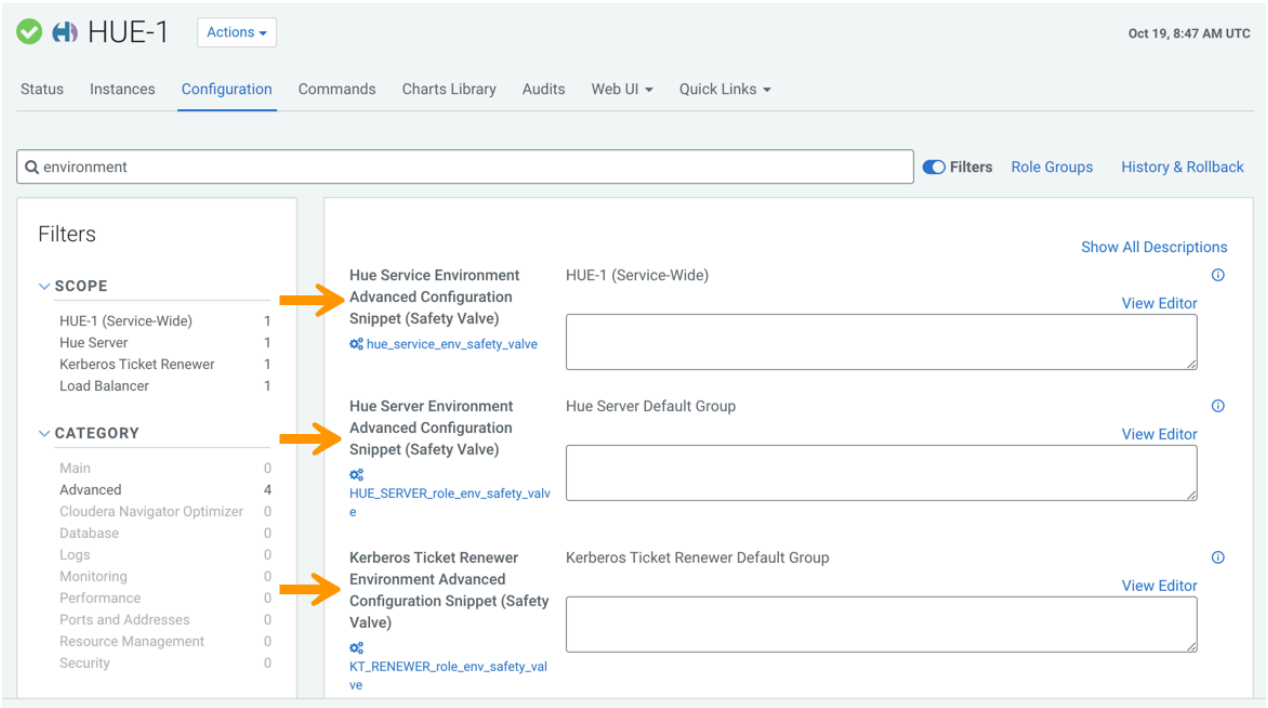
Advanced safety valves

The field for hue_safety_valve.ini is service-wide and affects all Hue roles listed on the Instances tab. The field for hue_safety_valve_server.ini only affects Hue Server role instances.



Environment safety valves

Environment safety-valves let you configure environment variables across the service or for specific role instances. These safety-valves lack headers.



Hue logs

Cloudera Manager generates standard stream logs when each Hue role instance starts and stops. The Hue service, which is built on the Django framework, generates log4j logs when the service is running.

Standard stream logs

Cloudera Manager logs the start and stop of each supervised Hue process in standard stream logs (stdout.log, stderr.log

When the Hue service restarts, Cloudera Manager generates a new directory for each supervised process of a Hue role instance:

```
ls -vrl /var/run/cloudera-scm-agent/process | grep HUE
```

```
[root@hue4-cdh512-1 ~]# ls -vrl /var/run/cloudera-scm-agent/process | grep HUE
drwxr-x--x 4 hue          hue          320 Aug  6 14:33 289-hue-HUE_LOAD_BALANCER
drwxr-x--x 8 hue          hue          440 Aug  6 14:33 288-hue-HUE_SERVER
drwxr-x--x 4 hue          hue          280 Aug  6 14:33 258-hue-HUE_LOAD_BALANCER
drwxr-x--x 8 hue          hue          460 Aug  6 14:33 257-hue-HUE_SERVER
drwxr-x--x 4 hue          hue          280 Aug  4 09:17 242-hue-HUE_LOAD_BALANCER
drwxr-x--x 8 hue          hue          460 Aug  4 09:17 241-hue-HUE_SERVER
drwxr-x--x 4 hue          hue          280 Aug  4 08:41 239-hue-HUE_LOAD_BALANCER
drwxr-x--x 8 hue          hue          460 Aug  4 08:41 238-hue-HUE_SERVER
drwxr-x--x 4 hue          hue          280 Aug  4 08:38 236-hue-HUE_LOAD_BALANCER
drwxr-x--x 8 hue          hue          460 Aug  4 08:38 235-hue-HUE_SERVER
```

It writes to a nested logs directory for each running instance:

```
[root@hue4-cdh512-1 logs]# pwd
/var/run/cloudera-scm-agent/process/289-hue-HUE_LOAD_BALANCER/logs
[root@hue4-cdh512-1 logs]# ll
total 16
-rw-r--r-- 1 root root 11148 Aug  6 14:33 stderr.log
-rw-r--r-- 1 root root   447 Aug  6 14:33 stdout.log
```

Configuration errors are written here because they prevent Hue servers and load balancers from starting properly.

Tip: Testing the LDAP configuration from Cloudera Manager (Clusters Hue service Test LDAP Configuration) also writes to standard stream logs which you can search using the following command: ls -vrl /var/run/cloudera-scm-agent/process | grep ldaptest

The supervisor

The supervisor is a watchdog process and supervisor.conf manages all Hue processes; its only purpose is to spawn and monitor other processes. A standard Hue installation starts and monitors the runcpserver process, which provides the core web functionality for Hue.

Note: To see active supervisor processes, run: ps -f -u hue.

For each Hue role, Cloudera Manager looks to the appropriate supervisor.conf for instructions on how to start the server.

```
# Hue Server Process Directory
cd /var/run/cloudera-scm-agent/process/`ls -valrt /var/run/cloudera-scm-agent/process | grep HUE_SERVER | tail -1 | awk '{print $9}'`
```

```
cat supervisor.conf
```

```
[program:288-hue-HUE_SERVER]
command=cmf-redactor "/usr/lib64/cmf/service/hue/hue.sh" "runcpserver"
autostart=true
directory=/run/cloudera-scm-agent/process/288-hue-HUE_SERVER
stdout_logfile=/run/cloudera-scm-agent/process/288-hue-HUE_SERVER/lo
gs/stdout.log
stdout_logfile_maxbytes=10MB
stdout_logfile_backups=10
stderr_logfile=/run/cloudera-scm-agent/process/288-hue-HUE_SERVER/lo
gs/stderr.log
stderr_logfile_maxbytes=10MB
stderr_logfile_backups=10
environment= ...
```

```
# Hue Load Balancer Process Directory
cd /var/run/cloudera-scm-agent/process/`ls -valrt /var/run/cloudera-scm-ag
ent/process | grep HUE_LOAD | tail -1 | awk '{print $9}'`
cat supervisor.conf
```

```
[program:258-hue-HUE_LOAD_BALANCER]
command=cmf-redactor "/usr/lib64/cmf/service/hue/httpd.sh"
...
```

Note: Currently, maxbytes=10MB, is hard-coded and cannot be changed for stdout or stderr.

If you installed other applications into your Hue instance, you may see other daemons running under the supervisor as well. Supervisor automatically restarts these processes if they fail for any reason. If they fail repeatedly in a short period of time, the supervisor itself shuts down.

Hue service Django logs

When the Hue service is running, Hue generates logs in /var/log/hue using log4j. Load balancer logs are in /var/run/httpd. You can view these logs in Hue at http://hueserver:port/logs.

Table 1: Hue service logs

Log Name	Description
access.log	Filtered list of successful attempts to access Hue Web UI
audit/hue_server_audit_wal.log	Audit log visible in Apache Atlas.
error.log	Filtered list of all nontrivial errors
kt_renewer.log	Kerberos ticket renews
metrics-hue_server/metrics.log	Populates charts in Cloudera Manager
migrate.log	Database and table migrations + First Run of Hue server
runcpserver.log	Hue (CherryPy) web server info (CP server runs Django core)
hue_install.log	Contains the log produced during installation

Enabling DEBUG

DEBUG is available for the Hue Django logs in /var/log/hue. By default, the Hue service writes INFO level messages and keeps a small buffer of log messages at all levels in memory.

There are two ways to enable DEBUG messages for all the logs in /var/log/hue :

- Cloudera Manager: Go to Hue Configuration, check Enable Django Debug Mode, and Save ChangesRestart.
- Hue Web UI: Go to the Home page, select Server Logs, and check Force Debug Level. Debug is enabled on-the-fly.

Hue supported browsers

Hue works with the two most recent LTS (long term support) or ESR (extended support release) browsers. Cookies and JavaScript must be enabled.

The lists the minimum tested versions of the most common browsers:

- Chrome: (Version history)
- Firefox: (Version history)
- Safari (Mac only): Version history
- Microsoft Edge: (Version history)

Hue can display in other browsers and in older versions of the common browsers, but you might not have access to all features.

Customizing the Hue web UI

To customize the Hub Web UI, add configuration properties in Cloudera Manager. You can customize the banner, the page logo, the splash screen, the cache timeout setting, and you can enable or disable anonymous usage data collection.

Adding a custom banner

Add a custom banner to the Hue web UI by adding your custom HTML to the Top Banner Custom HTML property in Cloudera Manager.

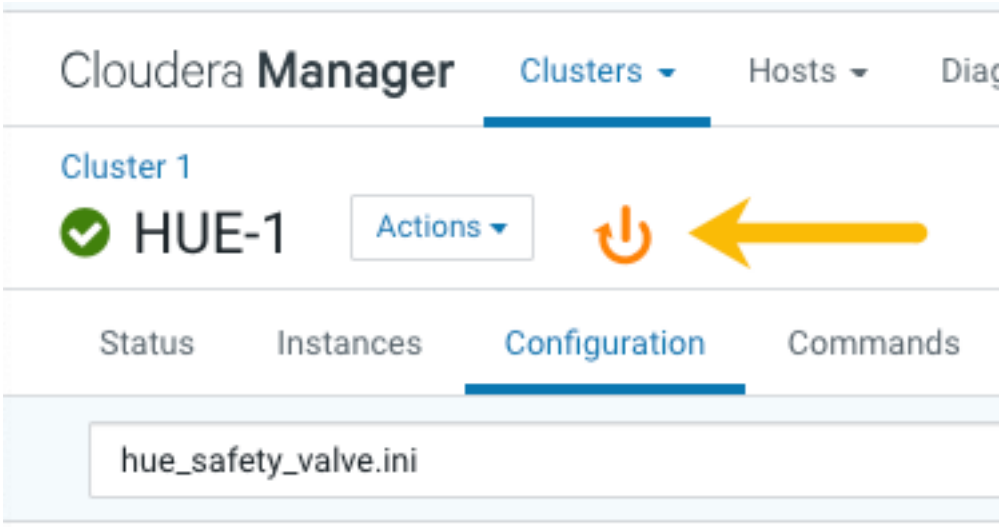
About this task

To add a custom banner to the Hue web UI:

Procedure

1. In the Cloudera Manager Admin Console, select ClustersHueConfiguration to navigate to the configuration page for Hue.
2. In the Search text box, type top banner to locate the Top Banner Custom HTML banner_top_html configuration parameter.
3. Add your custom HTML to the text box for the configuration parameter.
4. Click Save Changes at the bottom of the page to save the configuration change.

5. Refresh the browser page and click the restart icon at the top of the page so the new configuration changes can be read by the server:



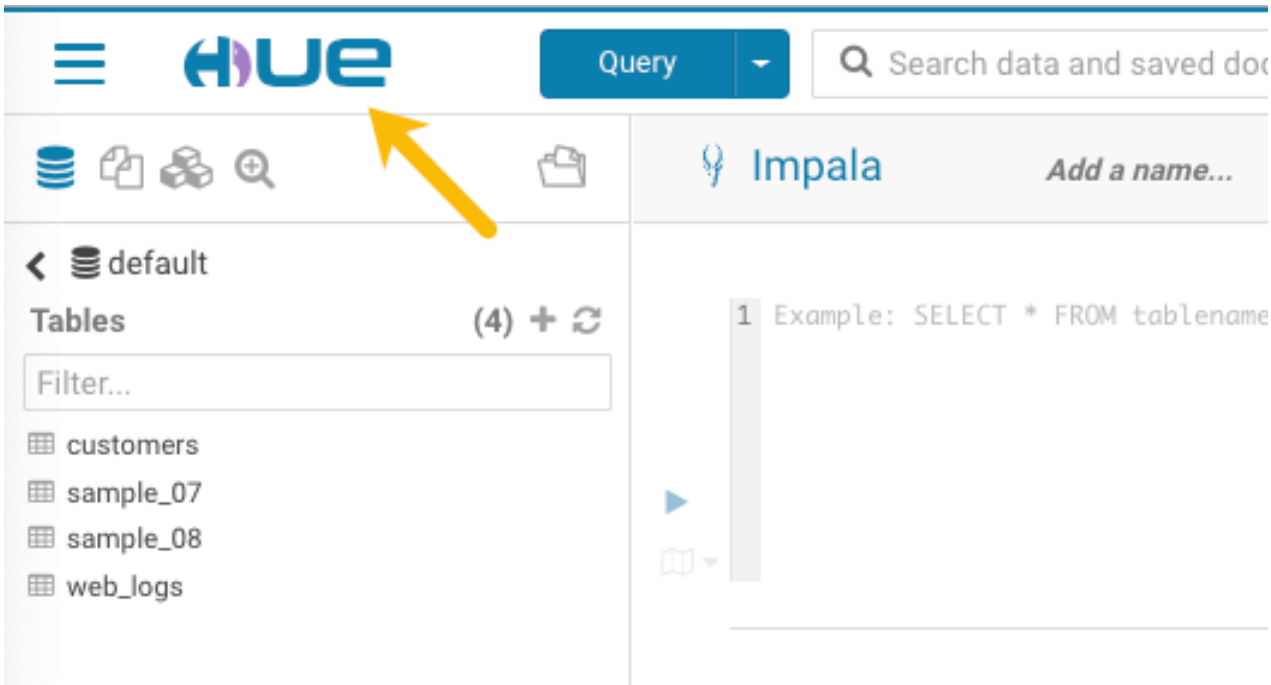
6. In the Hue configuration page of Cloudera Manager, select Web UIHue Load Balanced to load Hue and view your custom banner.

Changing the page logo

You can replace the Hue web UI logo with a custom log that is created with SVG code. Add any type of logo you want, but your custom logo should be designed to fit into a 160 x 40 pixel space.

About this task

For example, here is the Hue logo shown below:

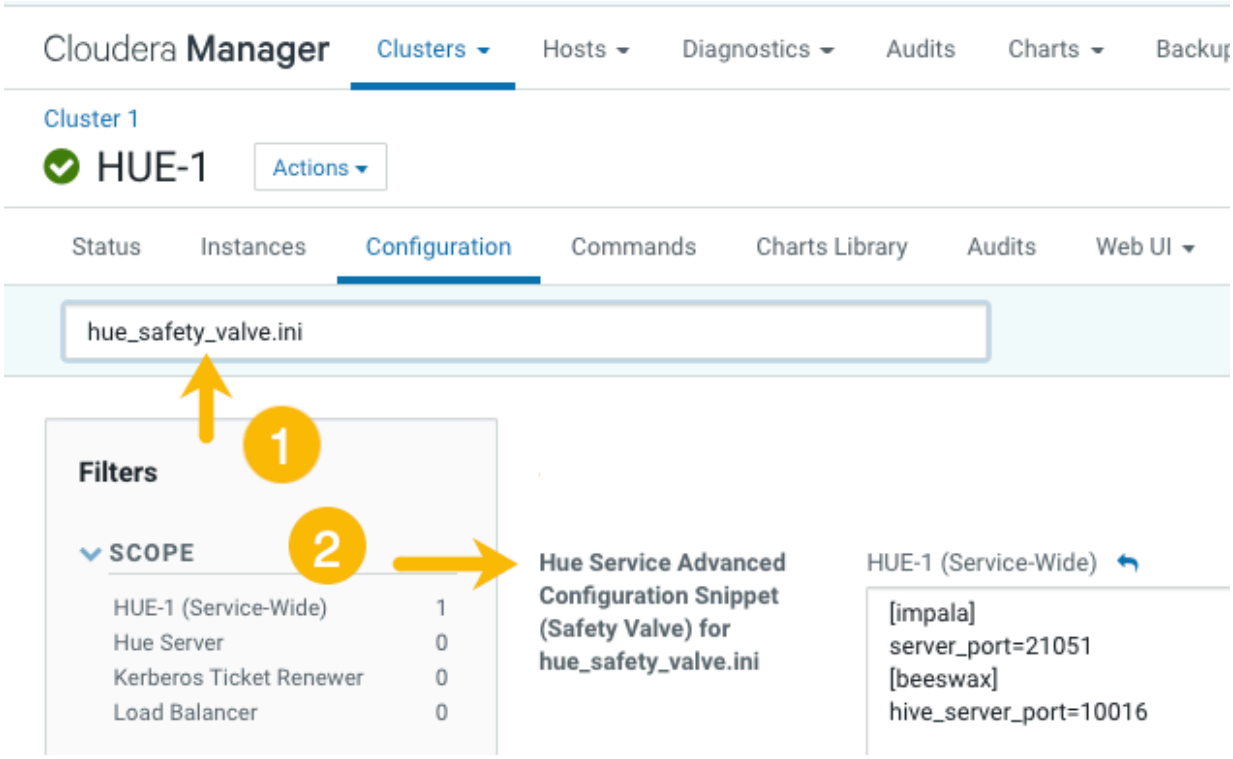


You can change this Hue logo by adding the appropriate SVG code to the logo_svg property under [desktop] [[cust om]] in the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini configuration parameter in Cloudera Manager as follows:

To replace the Hue logo with a custom logo:

Procedure

1. In the Cloudera Manager Admin Console, select ClustersHueConfiguration to navigate to the configuration page for Hue.
2. In the Search text box type hue_safety_valve.ini to locate the configuration parameter:



3. Add the following parameters with your custom logo SVG code to the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini configuration parameter:

```
[desktop]
[[custom]]
logo_svg='<CUSTOM_SVG_CODE_FOR_LOGO>'
```

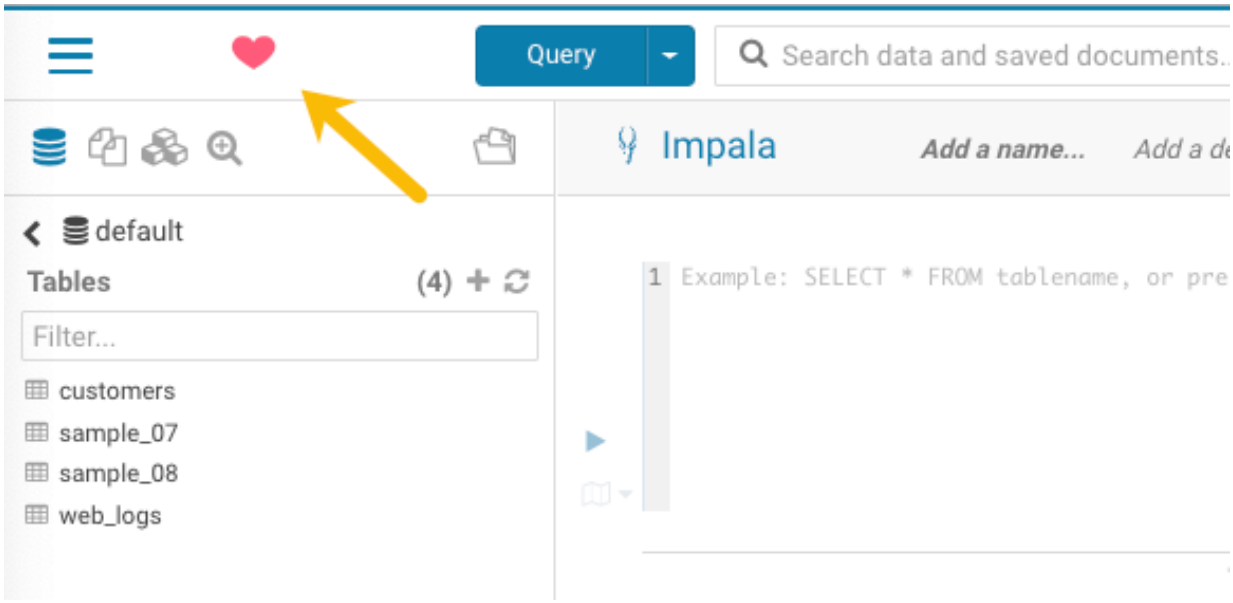
For example, the following SVG code replaces the Hue logo with a red heart:

```
[desktop]
[[custom]]
logo_svg='<g><path stroke="null" id="svg_1" d="m44.41215,11.43463c-4.05
017,-10.71473
-17.19753,-5.90773 -18.41353,-0.5567c-1.672,-5.70253 -14.497,-9.95663
-18.411,0.5643c-4.35797,11.71793 16.891,22.23443 18.41163,23.95773c1.518
1,-1.36927 22.7696,-12.43803
18.4129,-23.96533z" fill="#ffffff"/> <path stroke="null" id="svg_2"
d="m98.41246,10.43463c-4.05016,-10.71473 -17.19753,-5.90773 -18.41353,-
0.5567c-1.672,-5.70253
-14.497,-9.95663 -18.411,0.5643c-4.35796,11.71793 16.891,22.23443 18.4116
4,23.95773c1.5181,-1.36927
22.76959,-12.43803 18.41289,-23.96533z" fill="#FF5A79"/> <path stroke="nu
ll" id="svg_3"
```

```
d="m154.41215,11.43463c-4.05016,-10.71473 -17.19753,-5.90773 -18.41353,-0
.5567c-1.672,-5.70253
-14.497,-9.95663 -18.411,0.5643c-4.35796,11.71793 16.891,22.23443 18.41164
,23.95773c1.5181,-1.36927 22.76959,-12.43803 18.41289,-23.96533z" fill="
#ffffff"/> </g>'
```

4. Click Save Changes at the bottom of the page to save the configuration change.
5. Refresh the browser page and click the restart icon  at the top of the page so the new configuration changes can be read by the server.
6. In the Hue configuration page of Cloudera Manager, select Web UIHue Load Balanced to load Hue and view your custom logo.

   If you added the sample SVG code that defines a red heart as the logo, your Hue web UI looks like this:



Setting the cache timeout

Enable Hue UI caching by setting a timeout in milliseconds. The default is 86400000 milliseconds or one day. Set the timeout to 0 to disable caching. You can set the cache timeout using the cacheable_ttl property under [desktop][[custom]] in the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini configuration property in Cloudera Manager as follows.

About this task

To set the cache timeout value:

Procedure

1. In the Cloudera Manager Admin Console, select ClustersHueConfiguration to navigate to the configuration page for Hue.
2. In the Search text box, type hue_safety_valve.ini to locate the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini configuration parameter.
3. Add the following parameters with the cache timeout value to the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini configuration parameter:

```
[desktop]
```

```
[[custom]]
cacheable_ttl=<VALUE_IN_MILLISECONDS>
```

For example, the following configuration sets the cache timeout to the default value of 86400000 milliseconds:

```
[desktop]
[[custom]]
cacheable_ttl=86400000
```

4. Click Save Changes at the bottom of the page to save the configuration change.
5. Refresh the browser page and click the restart icon  at the top of the page so the new configuration changes can be read by the server and the new cache timeout limit takes effect.

Enabling or disabling anonymous usage date collection

Hue tracks anonymized pages and application versions to gather information about application usage levels. The data collected does not include hostnames or IDs. For example, the data collected has the format /2.3.0/pig or /2.5.0/beeswax/execute.

About this task

To enable or disable anonymous usage data collection:

Procedure

1. In the Cloudera Manager Admin Console, select ClustersHueConfiguration to navigate to the configuration page for Hue.
2. In the Search text box, type usage to locate the Enable Usage Data Collection check box:
   - To enable anonymous data collection, check the box, which is the default setting.
   - To disable anonymous data collection, clear the check box.
3. Enter a Reason for change..., and then click Save Changes at the bottom of the page to save the configuration change.
4. Refresh the browser page and click the restart icon  at the top of the page so the new configuration changes can be read by the server and the new data collection setting takes effect.

Enabling Hue applications with Cloudera Manager

Most Hue applications are configured by default, based on the services you have installed. Cloudera Manager selects the service instance that Hue depends on. If you have more than one service, you may want to verify or change the service dependency for Hue. If you add a service such as Oozie after you have set up Hue, you must set the dependency because it is not done automatically.

About this task

To add a dependency in Hue:

Procedure

1. In the Cloudera Manager Admin Console, select ClustersHueConfiguration to navigate to the configuration page for Hue.

2. Filter by ScopeHue (Service-Wide) and CategoryMain .
3. Select the <*SERVICE_NAME*> Service property that you want to set a dependency for. Select none to remove a dependency.
4. Enter a Reason for change..., and then click Save Changes at the bottom of the page to save the configuration change.
5. Refresh the browser page and click the restart icon  at the top of the page so the new configuration changes can be read by the server and the new setting takes effect.

Running shell commands

You can run shell commands to administer Hue programmatically. For example, to reset the superuser password or to assign an LDAP user superuser permissions.

About this task

To run Hue shell commands:

Procedure

1. Set HUE_CONF_DIR to the latest Hue process directory:

```
export HUE_CONF_DIR="/var/run/cloudera-scm-agent/process/`ls -alrt /var/
run/cloudera-scm-agent/process | grep HUE_SERVER | tail -1 | awk '{print
 $9}'`"
echo $HUE_CONF_DIR
```

2. Set environment variables used to run the Hue webserver:
   - CentOS/RHEL:

```
for line in `strings /proc/$(lsof -i :8888|grep -m1 python|awk '{ print
 $2 }')/environ|egrep -v "^HOME=|^TERM=|^PWD="`;do export $line;done
```

   - Ubuntu:

```
for line in `strings /proc/$(lsof -i :8888|grep -m1 hue|awk '{ print $2
 }')/environ|egrep -v "^HOME=|^TERM=|^PWD="`;do export $line;done
```

3. Run shell subcommands

   When true, HUE_IGNORE_PASSWORD_SCRIPT_ERRORS runs the Hue shell even if hue.ini contains passwords generated by Cloudera Manager (such as bind_password and ssl_password).

   Note: Do not export HUE_IGNORE_PASSWORD_SCRIPT_ERRORS or HUE_DATABASE_PASSWORD to ensure that they are not stored and only apply to this command.

   For CDH parcel deployments:

   - Run the interactive Hue Python shell (Ctrl+D to quit)

     ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /opt/cloudera/parcels/CDH/lib/hue/
     build/env/bin/hue shell
     ```

     Or with the database password:

     ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 HUE_DATABASE_PASSWORD=<your db passw
     ord> /opt/cloudera/parcels/CDH/lib/hue/build/env/bin/hue shell
     ```

   - Change a user password

     ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /opt/cloudera/parcels/CDH/lib/hue/bu
     ild/env/bin/hue changepassword admin
     ```

   - Promote Hue user to superuser

     ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /opt/cloudera/parcels/CDH/lib/hue/bu
     ild/env/bin/hue shell
     ```

     ```
     from django.contrib.auth.models import User
     a = User.objects.get(username='gwen')
     a.is_superuser = True
     a.save()
     ```

   - Count all of the documents of a certain user:

     ```
     from django.contrib.auth.models import User
     from desktop.models import Document2

     user=User.objects.get(username='demo')
     Document2.objects.documents(user=user).count()

     Out[X]: 1167
     ```

   - List available subcommands

     ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /opt/cloudera/parcels/CDH/lib/hue/bu
     ild/env/bin/hue
     ```

   For CDH package deployments:

   - ```
     HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /usr/lib/hue/build/env/bin/hue shell
     ```
     ```

Downloading and exporting data from Hue

Hue enables you to download or export data from Hue to HDFS or to an external storage location from Hue Editor, Hue Dashboard, and the Hue File browser. You can limit the number of rows or bytes that are downloaded or disable the export feature altogether so that you do not run out of storage space.

Requied Role: Administrator.

For a service-wide change, go to Cloudera Manager Clusters Hue service Configuration and specify the configurations in the Hue Service Advanced Configuration Snippet (Safety valve) for hue_safety_valve.ini field.

By default, Hue users can download the query results from the Hue Editor, the Hue Dashboard, and the File browser.

Limiting the number of rows to download

Specify the following in the Hue Service Advanced Configuration Snippet (Safety valve) for hue_safety_valve.ini to limit the number of rows that can be downloaded from a query before it is truncated:

```
[beeswax]
download_row_limit=X
```

X represents the number of rows that you can download.

By default, there is no download limit, and you can configure this by setting the value to “-1”:

```
[beeswax]
download_row_limit=-1
```

Limiting the number of bytes to download

Specify the following in the Hue Service Advanced Configuration Snippet (Safety valve) for hue_safety_valve.ini to limit the number of bytes that can be downloaded from a query before it is truncated:

```
[beeswax]
download_bytes_limit=X
```

X represents the number of bytes that you can download.

By default, there is no download limit, and you can configure this by setting the value to “-1”:

```
[beeswax]
download_bytes_limit=-1
```

Disabling the data download feature

Specify the following in the Hue Service Advanced Configuration Snippet (Safety valve) for hue_safety_valve.ini field to disable your users from downloading query results:

```
[desktop]
enable_download=false
```

Specify the following in the Hue Service Advanced Configuration Snippet (Safety valve) for hue_safety_valve.ini field to hide the Download button from the Hue File browser:

```
[filebrowser]
show_download_button=false
```

Connect an external database

Hue needs its own database to store items such as user account information, job submissions, and SQL queries. It is packaged with a lightweight embedded PostgreSQL database, which is intended only for proof-of-concept deployments with one Hue server. For production environments, Cloudera recommends connecting Hue to an external database.

There are two ways to connect Hue to an external database:

- During a new CDP installation with the Cloudera Manager Installation Wizard at the Database Setup step. The database must be installed, configured, and running.
- After CDP is installed with Cloudera Manager, go to the HueConfiguration tab. You can migrate from an old database and connect to a new one. Or you can just connect to the new database without saving the data in the old database.

It is only necessary to migrate to a new database if you want to save the data in your current database. Otherwise, connect to the new database and restart Hue without migrating the old database data. External databases can be remote, but ensure that the database server is properly configured to listen on the correct address to receive requests from Hue.

Enabling a multi-threaded environment for Hue

A multi-threaded environment can help reduce the time it takes to migrate data from one database to other. By default, operations such as migrating data run on a single thread. For example, if you are switching from MySQL as the backend database for Hue to Oracle, then enabling a multi-threaded environment significantly reduces the data transfer time.

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 .
3. Locate the [[database]] section under [desktop] and set threaded to true:

```
[desktop]
[[database]]
options={"threaded":true}
```

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