

Cloudera Runtime 7.1.5

## Administering Hue

Date published: 2020-08-10

Date modified: 2020-11-30

# CLOUDERA

<https://docs.cloudera.com/>

# Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

# Contents

<b>Reference architecture.....</b>	<b>5</b>
<b>Hue configuration files.....</b>	<b>6</b>
<b>Introduction to Hue Advanced Configuration Snippet (Safety valves).....</b>	<b>7</b>
Advanced safety valves.....	7
Environment safety valves.....	8
<b>Hue logs.....</b>	<b>8</b>
Standard stream logs.....	9
The supervisor.....	9
Hue service Django logs.....	10
Enabling DEBUG.....	10
<b>Hue supported browsers.....</b>	<b>11</b>
<b>Adding a Hue service with Cloudera Manager.....</b>	<b>11</b>
<b>Adding a Hue role instance with Cloudera Manager.....</b>	<b>12</b>
<b>Customizing the Hue web UI.....</b>	<b>13</b>
Adding a custom banner.....	13
Changing the page logo.....	14
Setting the cache timeout.....	16
Enabling or disabling anonymous usage data collection.....	17
<b>Using PostgreSQL database with Hue.....</b>	<b>17</b>
Download and install PostgreSQL.....	18
Configure the PostgreSQL server.....	18
Configure PostgreSQL as the backend database for Hue.....	20
<b>Enabling Hue applications with Cloudera Manager.....</b>	<b>20</b>
<b>Running shell commands.....</b>	<b>21</b>
<b>Downloading and exporting data from Hue.....</b>	<b>23</b>

<b>Backing up the Hue database.....</b>	<b>24</b>
<b>Connect an external database.....</b>	<b>24</b>
<b>Enabling a multi-threaded environment for Hue.....</b>	<b>24</b>
<b>Moving the Hue service to a different host.....</b>	<b>25</b>
Adding and configuring a new Hue service on a new host.....	25
Adding new role instances for Hue server, Hue Load Balancer, and Kerberos Ticket Renewer on new hosts.....	26

## 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

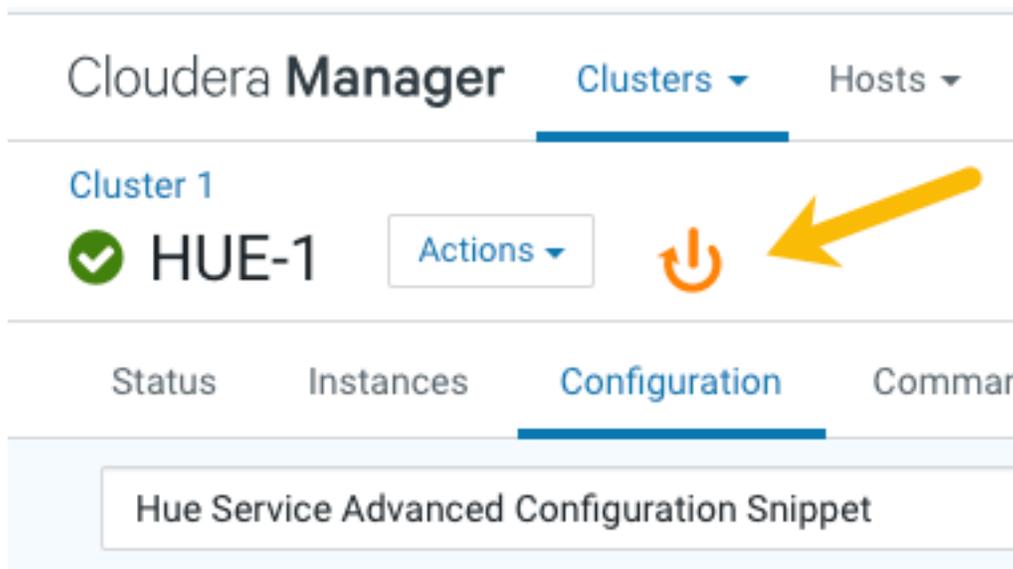
- Deploy a load balancer in front of Hue.
- Use a production-quality database.
- Ensure that other services, such as Impala, Hive, and Oozie, are healthy and not impacted by too few resources. If these services are hanging, it adversely affects Hue performance.
- Consider moving workloads that are subject 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**

```
[root@hue4-cdh512-1 257-hue-HUE_SERVER]# cd /var/run/cloudera-scm-agent/process/`ls -valrt /var/run/cloudera-scm-agent/process | grep HUE_SERVER | tail -1 | awk '{print $9}'`
[root@hue4-cdh512-1 257-hue-HUE_SERVER]# pwd
/var/run/cloudera-scm-agent/process/257-hue-HUE_SERVER
[root@hue4-cdh512-1 257-hue-HUE_SERVER]# ll
total 76
-rwxr----- 1 hue hue 393 Aug 4 09:17 altscript.sh
-rw-r----- 1 hue hue 359 Aug 4 09:17 cloudera-monitor.properties
-rw----- 1 root root 21330 Aug 4 09:17 config.zip
-rw-r----- 1 hue hue 1987 Aug 4 09:17 creds.localjceks
drwxr-xr-x 2 hue hue 300 Aug 4 09:17 hive-conf
-rw-r----- 1 hue hue 4267 Aug 4 09:17 hue.ini
-rw----- 1 hue hue 99 Aug 4 09:17 hue.keytab
-rw-r----- 1 hue hue 0 Aug 4 09:17 hue_safety_valve.ini
-rw-r----- 1 hue hue 0 Aug 4 09:17 hue_safety_valve_server.ini
drwxr-x--x 2 hue hue 60 Aug 4 09:17 impala-conf
drwxr-x--x 2 hue hue 80 Aug 4 09:17 logs
-rw-r----- 1 hue hue 541 Aug 4 09:17 navigator.client.properties
-rw-r----- 1 hue hue 540 Aug 4 09:17 navigator.lineage.client.properties
-rw----- 1 root root 2415 Aug 4 09:17 proc.json
-rw-r----- 1 hue hue 0 Aug 4 09:17 redaction-rules.json
drwxr-x--x 2 hue hue 60 Aug 4 09:17 sentry-conf
-rw-r----- 1 hue hue 8328 Aug 4 09:17 service-metrics.properties
drwxr-x--x 2 hue hue 60 Aug 4 09:17 sqoop2-conf
-rw----- 1 root root 3000 Aug 4 09:17 supervisor.conf
drwxr-xr-x 2 hue hue 220 Aug 4 09:17 yarn-conf
```

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.

The screenshot shows the Hue configuration page for HUE-1. The search bar contains ".ini". The filters on the left show the following results:

SCOPE	Count
HUE-1 (Service-Wide)	1
Hue Server	1
Kerberos Ticket Renewer	0
Load Balancer	0

CATEGORY	Count
Main	0
Advanced	2
Cloudera Navigator Optimizer	0
Database	0
Logs	0
Monitoring	0
Performance	0
Ports and Addresses	0

The search results list two configuration snippets:

- Hue Service Advanced Configuration Snippet (Safety Valve) for hue\_safety\_valve.ini** (hue\_service\_safety\_valve) - HUE-1 (Service-Wide)
- Hue Server Advanced Configuration Snippet (Safety Valve) for hue\_safety\_valve\_server.ini** (hue\_server\_hue\_safety\_valve) - Hue Server Default Group

## Environment safety valves

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

The screenshot shows the Hue configuration page for HUE-1 with the search bar containing "environment". The filters on the left show the following results:

SCOPE	Count
HUE-1 (Service-Wide)	1
Hue Server	1
Kerberos Ticket Renewer	1
Load Balancer	1

CATEGORY	Count
Main	0
Advanced	4
Cloudera Navigator Optimizer	0
Database	0
Logs	0
Monitoring	0
Performance	0
Ports and Addresses	0
Resource Management	0
Security	0

The search results list three configuration snippets:

- Hue Service Environment Advanced Configuration Snippet (Safety Valve)** (hue\_service\_env\_safety\_valve) - HUE-1 (Service-Wide)
- Hue Server Environment Advanced Configuration Snippet (Safety Valve)** (HUE\_SERVER\_role\_env\_safety\_valve) - Hue Server Default Group
- Kerberos Ticket Renewer Environment Advanced Configuration Snippet (Safety Valve)** (KT\_RENEWER\_role\_env\_safety\_valve) - Kerberos Ticket Renewer Default Group

## 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 runcpsrvr 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/logs/stdout.log
stdout_logfile_maxbytes=10MB
stdout_logfile_backups=10
stderr_logfile=/run/cloudera-scm-agent/process/288-hue-HUE_SERVER/logs/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-agent/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
<code>access.log</code>	Filtered list of successful attempts to access Hue Web UI
<code>audit/hue_server_audit_wal.log</code>	Audit log visible in Apache Atlas.
<code>error.log</code>	Filtered list of all nontrivial errors
<code>kt_renewer.log</code>	Kerberos ticket renews
<code>metrics-hue_server/metrics.log</code>	Populates charts in Cloudera Manager
<code>migrate.log</code>	Database and table migrations + First Run of Hue server
<code>runcpserver.log</code>	Hue (CherryPy) web server info (CP server runs Django core)
<code>hue_install.log</code>	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.

## Adding a Hue service with Cloudera Manager

Hue is part of the CDP Private Cloud Base stack, but is not installed by default. To use Hue, you must add it to your cluster using the "Add Service to Cluster" wizard in Cloudera Manager.

### Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Actions and click Add Service.  
The **Add Service to Cluster** wizard is displayed.
3. Select Hue as the Service Type and click Continue.  
The **Select Dependencies** page is displayed.
4. On the **Select Dependencies** page, select the optional dependencies that are required for your cluster and click Continue.

The required dependency is selected by default.



#### Note:

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

If you need to run Hive queries, then select an optional dependency containing the `HIVE_ON_TEZ` service.

If you need to run only Impala queries, then the `HIVE_ON_TEZ` service is not required.

5. On the **Assign Roles** page, you can perform one of the following actions:

- To accept the default role assignments, click Continue in the lower right corner of the page. To view the default role assignments by host, click View By Host.
- To customize role assignments, click the field below Hue Server and Load Balancer roles. This launches a dialog box where you can select hosts where you want to add the role. Click OK after making your selection.

The wizard evaluates host hardware configurations to determine the best hosts for each role. All worker roles are automatically assigned to the same set of hosts as the HDFS DataNode. You can reassign if necessary. Specify hostnames by IP address, rack name, or by range:

Range Definition	Matching Hosts
10.1.1.[1-4]	10.1.1.1, 10.1.1.2, 10.1.1.3, 10.1.1.4
host[1-3].company.com	host1.company.com, host2.company.com, host3.company.com
host[07-10].company.com	host07.company.com, host08.company.com, host09.company.com, host10.company.com

6. On the **Setup Database** page:

- Select a database vendor from the Type field and specify the Database Hostname, Database Name, Username, and Password.
- Click Test Connection, and when the success message appears, click Continue and Cloudera Manager starts the Hue service.

7. Review your configurations on the **Review Changes** page and click Continue.

8. On the **Command Details** page, expand the running commands to view the details of any step, including log files and command output. You can filter the view by selecting Show All Steps, Show Only Failed Steps, or Show Only Running Steps.

After the First Run command completes, click Continue to go to the **Summary** page.

9. After the service is started, click Continue and then click Finish.

If your cluster uses Kerberos, Cloudera Manager automatically adds a Hue Kerberos Ticket Renewer role to each host where you assigned the Hue Server role instance. See "Enabling Kerberos Authentication" for more information.

## Adding a Hue role instance with Cloudera Manager

Roles are functions that comprise a service. Role instances must be assigned to one or more hosts. You can assign roles to hosts by using Cloudera Manager.

### Procedure

1. In the Cloudera Manager Home page, select the Hue service.
2. On the Hue service page, click the Instances tab.
3. On the Instances tab page, click Add Role Instances on the right side of the page.
4. On the Assign Roles page:
  - a) Click Select hosts under the role names to select hosts where you want to add the role. This launches a dialog box where you can select hosts where you want to add the role.

The wizard evaluates host hardware configurations to determine the best hosts for each role. All worker roles are automatically assigned to the same set of hosts as the HDFS DataNode. You can reassign if necessary. Specify hostnames by IP address, rack name, or by range:

Range Definition	Matching Hosts
10.1.1.[1-4]	10.1.1.1, 10.1.1.2, 10.1.1.3, 10.1.1.4
host[1-3].company.com	host1.company.com, host2.company.com, host3.company.com

Range Definition	Matching Hosts
host[07-10].company.com	host07.company.com, host08.company.com, host09.company.com, host10.company.com

- b) Click OK to save your hostname selection for a role.
  - c) When you are finished assigning roles to hosts, click Continue in the lower right corner of the page.
5. Cloudera Manager automatically adds the roles and returns you to the Hue Instances tab.



**Note:** If your cluster uses Kerberos, Cloudera Manager automatically adds a Hue Kerberos Ticket Renewer role to each host where you assigned the Hue Server role instance. Cloudera Manager throws a validation error if the new Hue Server role does not have a co-located Kerberos Ticket Renewer role. See "Enabling Kerberos for Authentication" for more information.

## 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.

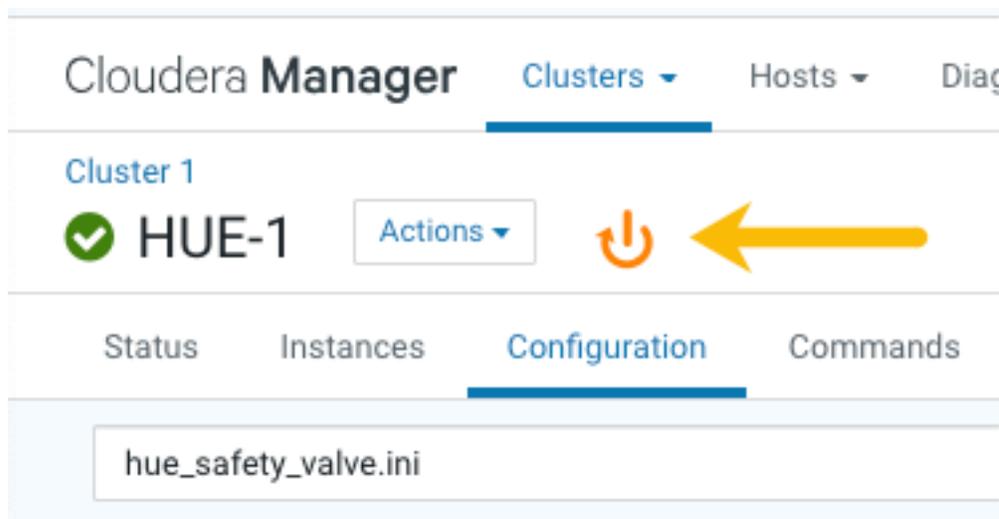
#### About this task

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.

- 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:



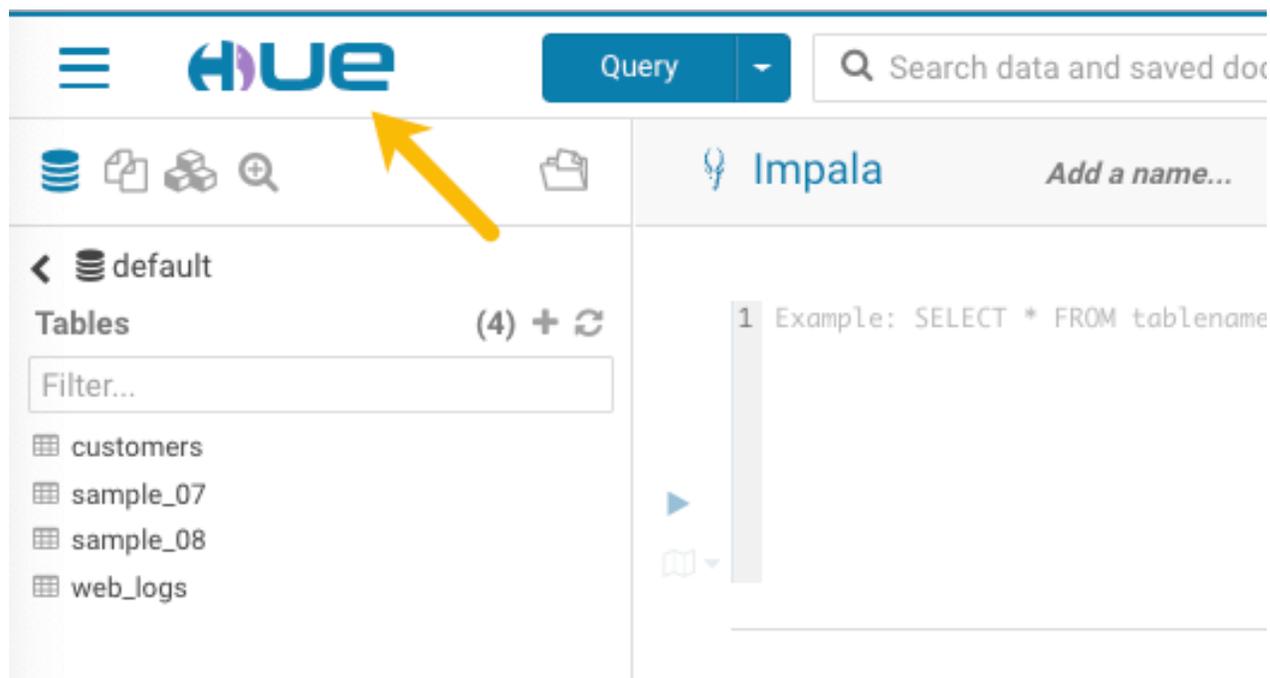
- In the Hue configuration page of Cloudera Manager, select `Web UIHue Load Balanced` to load Hue and view your custom banner.

## 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]` `[[custom]]` 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:

The screenshot shows the Cloudera Manager Admin Console interface. At the top, there are navigation tabs: Clusters, Hosts, Diagnostics, Audits, Charts, and Backup. Below this, the cluster name 'Cluster 1' is shown with a green checkmark and 'HUE-1'. A search bar contains the text 'hue\_safety\_valve.ini'. Below the search bar, there is a 'Filters' section with a 'SCOPE' filter. The filter table shows the following items:

SCOPE	Count
HUE-1 (Service-Wide)	1
Hue Server	0
Kerberos Ticket Renewer	0
Load Balancer	0

The 'HUE-1 (Service-Wide)' filter is selected, and an arrow points to the configuration snippet for 'hue\_safety\_valve.ini'. The snippet shows the following parameters:

```
[impala]
server_port=21051
[beeswax]
hive_server_port=10016
```

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-4.05017,-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 18.41163,23.95773c1.5181,-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 18.41164,23.95773c1.5181,-1.36927
22.76959,-12.43803 18.41289,-23.96533z" fill="#FF5A79"/> <path stroke="null" id="svg_3">
```

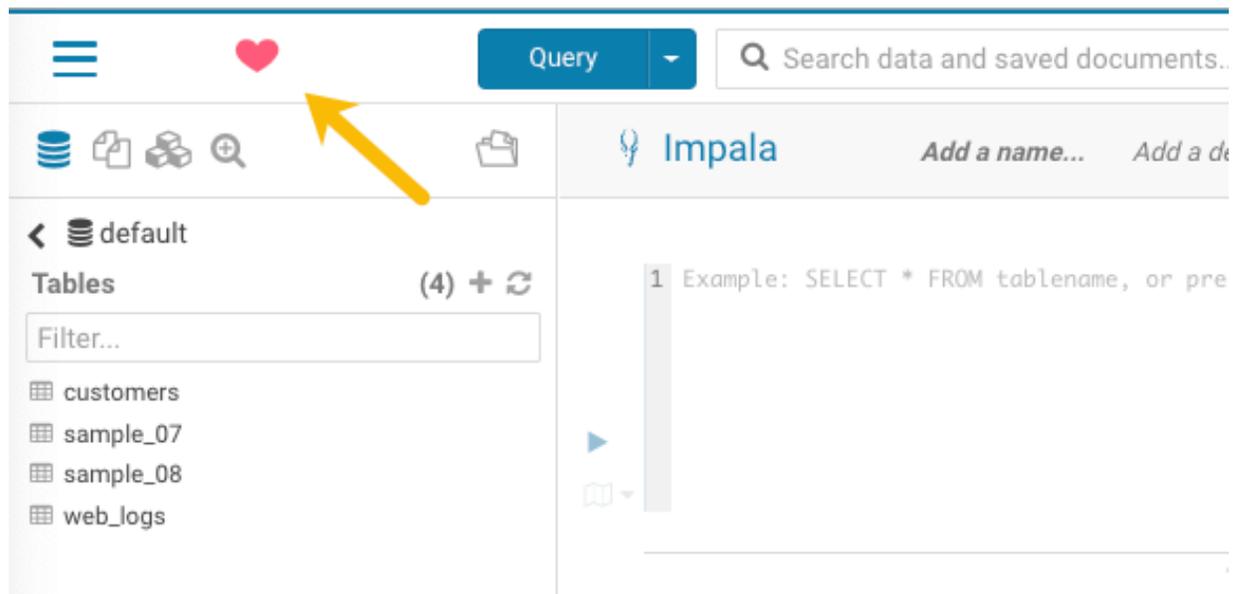
```
d="m154.41215,11.43463c-4.05016,-10.71473 -17.19753,-5.90773 -18.41353,-0
.5567c-1.672,-5.70253
-14.497,-9.95663 -18.411,0.5643c-4.35796,11.71793 16.891,22.23443 18.41164
,23.95773c1.5181,-1.36927 22.76959,-12.43803 18.41289,-23.96533z" fill="
#ffffff"/> </g>'
```

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 value in milliseconds. The default is 10 days or 864000000 milliseconds. Set the timeout to 0 to disable caching.

### About this task

When you browse tables using the left assist panel or run queries, Hue caches this information for fetching information faster and query autocompletion. You can configure the time for which you want to Hue to cache this information by setting the value of the `cacheable_ttl` property under the `[desktop][[custom]]` section in the Hue Service Advanced Configuration Snippet (Safety Valve) for `hue_safety_valve.ini` configuration property in Cloudera Manager.

### Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue Configuration
3. 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.

4. 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
```

5. Click Save Changes.
6. Restart the Hue service.
- 7.
- 8.
- 9.

## Enabling or disabling anonymous usage data 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.

## Using PostgreSQL database with Hue

Hue is packaged with a lightweight embedded PostgreSQL database for proof-of-concept deployments with one Hue server. In production environments, you can install and use a custom PostgreSQL database as the backend database to store Hue metadata, job and query history, and account information.

## Download and install PostgreSQL

RHEL and CentOS 7 include PostgreSQL version 9.2. However, on these OS versions, Hue is compatible only with PostgreSQL versions 10, 11.x, and 12. Therefore, you must download and install the PostgreSQL distribution separately.

### Procedure

1. Download the PostgreSQL distribution from <https://www.postgresql.org/download/> based on your OS version.
2. Install the PostgreSQL server using the following command:

```
sudo yum install postgresql-server
```

## Configure the PostgreSQL server

By default, PostgreSQL only accepts connections on the loopback interface. You must reconfigure PostgreSQL to accept connections from the fully qualified domain names (FQDN) of the hosts hosting the services for which you are configuring databases. If you do not make these changes, the services cannot connect and use the database on which they depend.

### Procedure

1. Make sure that LC\_ALL is set to en\_US.UTF-8 and initialize the database as follows:

```
echo 'LC_ALL="en_US.UTF-8"' >> /etc/locale.conf  
sudo su -l postgres -c "postgresql-setup initdb"
```

2. Enable MD5 authentication. Edit the pg\_hba.conf file typically located at /var/lib/pgsql/data or /etc/postgresql/[\*\*\*VERSION\*\*\*]/main. Add the following line:

```
host all all 127.0.0.1/32 md5
```

3. If the default pg\_hba.conf file contains the "host all all 127.0.0.1/32 ident" line, then the host line specifying MD5 authentication shown above must be inserted before this line. Failure to do so may cause an authentication error when running the scm\_prepare\_database.sh script. You can modify the contents of the MD5 line shown above to support different configurations. For example, if you want to access PostgreSQL from a different host, replace 127.0.0.1 with your IP address and update the postgresql.conf file, which is typically found in the same place as pg\_hba.conf, to include the following line:

```
listen_addresses = '*'
```

4. Configure settings to ensure your system performs as expected. Update these settings in the /var/lib/pgsql/data/postgresql.conf or /var/lib/postgresql/data/postgresql.conf file based on cluster size and resources as follows:  
Small to mid-sized clusters: Consider the following settings as starting points. If resources are limited, consider reducing the buffer sizes and checkpoint segments further. Ongoing tuning may be required based on each host's

resource utilization. For example, if the Cloudera Manager server is running on the same host as other roles, the following values may be acceptable:

- `max_connection`: In general, allow each database on a host a maximum of 100 connections and then add 50 extra connections. You may have to increase the system resources available to PostgreSQL.
- `shared_buffers`: 256 MB
- `wal_buffers`: 8 MB
- `checkpoint_segments`: 16. Replace configuration parameter `checkpoint_segments` with `min_wal_size` and `max_wal_size`.

If you previously adjusted `checkpoint_segments`, the following formula will give you an approximately equivalent setting:

```
max_wal_size = (3 * checkpoint_segments) * 16MB
```



**Note:**

The `checkpoint_segments` setting is removed in PostgreSQL 9.5 and higher. This setting is replaced by `min_wal_size` and `max_wal_size`.

The default setting for `max_wal_size` is much higher than the default setting for `checkpoint_segments`. So, adjusting it might no longer be necessary.

- `checkpoint_completion_target`: 0.9

Large clusters: Can contain up to 1000 hosts. Consider the following settings as starting points:

- `max_connection`: for large clusters, allow each database on a host a maximum of 100 connections and then add 50 extra connections. You may have to increase the system resources that is available to PostgreSQL.
- `shared_buffers`: 1024 MB. This requires that the operating system can allocate sufficient shared memory.
- `wal_buffers`: 16 MB. This value is derived from the `shared_buffers` value. Setting `wal_buffers` to be approximately 3% of `shared_buffers`, up to a maximum of approximately 16 MB is sufficient in most cases.
- `checkpoint_segments`: 128. The PostgreSQL Tuning Guide recommends values between 32 and 256 for write-intensive systems, such as this one. Replace configuration parameter `checkpoint_segments` with `min_wal_size` and `max_wal_size`.

If you previously adjusted `checkpoint_segments`, the following formula will give you an approximately equivalent setting:

```
max_wal_size = (3 * checkpoint_segments) * 16MB
```



**Note:**

The `checkpoint_segments` setting is removed in PostgreSQL 9.5 and higher. This setting is replaced by `min_wal_size` and `max_wal_size`.

The default setting for `max_wal_size` is much higher than the default setting for `checkpoint_segments`. So, adjusting it might no longer be necessary.

- `checkpoint_completion_target`: 0.9

5. Configure the PostgreSQL server to start when the system boots.

OS	Command
RHEL 7 compatible, SLES, and Ubuntu	<pre>sudo systemctl enable postgresql</pre>

6. Restart the PostgreSQL database.

OS	Command
RHEL 7 compatible, SLES, and Ubuntu	<pre>sudo systemctl restart postgresql</pre>

## Configure PostgreSQL as the backend database for Hue

After installing and configuring the PostgreSQL database, you can configure Hue to use it as its backend database.

### Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service and stop the Hue service by clicking **ACTIONS Stop**.
3. Go to Clusters Hue service Configuration Category Database .  
All the database configurations for Hue are displayed.
4. Select PostgreSQL as the Hue Database Type.
5. Specify the fully qualified domain name (FQDN) of the host on which you have installed the PostgreSQL database in the Hue Database Hostname field.
6. Enter 5432 in the Hue Database Port field.
7. Enter user name and password in Hue Database Username and Hue Database Password fields.
8. Specify the database name in the Hue Database Name field.
9. Click **Save Changes** and restart the Hue service.

### Results

If the database setup is successful and Hue starts without errors, then you should be able to log into the Hue web interface and view the database and tables.

## 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 webservice:

- CentOS/RHEL:

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

- Ubuntu:

```
for line in `strings /proc/$(lsof -i :8888|grep -ml 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 password> /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/build/env/bin/hue changepassword admin
```

- Promote Hue user to superuser

```
HUE_IGNORE_PASSWORD_SCRIPT_ERRORS=1 /opt/cloudera/parcels/CDH/lib/hue/build/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/build/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.

Required 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
```

## Backing up the Hue database

Cloudera Manager provides you (the Hue service administrator) an option to easily back up your database without the need to engage your database administrators or perform this operation manually.

### About this task

You may have the need to back up the Hue database before upgrading or migrating to a new database, or for any other purposes. You can do so from Cloudera Manager as follows:

### Procedure

1. Log in to Cloudera Manager as Administrator.
2. Go to Clusters Hue service .
3. Stop the Hue service by clicking Actions Stop .
4. After the Hue service has successfully stopped, click Actions Dump Database .
5. On the **Dump Database** popup, click Dump Database to confirm the operation.  
The database dump is written to the `/tmp/hue_database_dump.json` file on the Hue server host.
6. Start the Hue service by clicking Actions Start .

## 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 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. 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.

## Moving the Hue service to a different host

After adding new hosts to your CDP cluster if you need to move the Hue service from one host to another, then either add a new Hue service and manually configure it on the new host or add new role instances for Hue server, Hue Load Balancer, and Kerberos Ticket Renewer on new hosts.

## Adding and configuring a new Hue service on a new host

To migrate the Hue service from one host to another within the same cluster, you can add a new Hue service on a new host using the Add Service wizard and manually configure the Hue service as needed.

### Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service and stop the Hue service by clicking Actions Stop .
3. Go to Clusters `***CLUSTER-NAME***` and click Add Service from the menu.
4. On the **Add Service** wizard, select Hue.
5. On the **Select Dependencies** page, select the Optional Dependencies.



**Important:** In CDH 6 and earlier, the Hive service included the Hive Metastore and HiveServer2. In Cloudera Runtime 7.0 and later, this service only includes the Hive Metastore. HiveServer2 and other components of the Hive execution engines are part of the HIVE\_ON\_TEZ service.

If you need to run Hive queries, then select an optional dependency containing the HIVE\_ON\_TEZ service. If you need to run only Impala queries, then the HIVE\_ON\_TEZ service is not required.

6. On the **Assign Roles** page, click the field below Hue Server and Load Balancer roles. This launches a dialog box where you can select hosts where you want to add the role.  
Click OK after making your selection.
7. On the **Setup Database** page, specify the details of your existing Hue database.
  - a) Select your database vendor from the Type field and specify the Database Hostname, Database Name, Username, and Password of your existing database.
  - b) Click Test Connection, and when the success message appears, click Continue.
 The wizard configures and starts Hue and its dependent services.
8. After the new Hue service is in the running state, configure the Hue Advanced Configuration Snippet as it is in the existing Hue service on the older host.
9. Click Save Changes and restart the Hue service.
10. Verify that the new Hue service functions as expected.
11. Delete the Hue service from the old host by clicking Actions Delete .

## Adding new role instances for Hue server, Hue Load Balancer, and Kerberos Ticket Renewer on new hosts

To migrate the Hue service from one host to another within the same cluster, you can add new hosts to your CDP cluster and then add the Hue Server, Hue Load Balancer, and Kerberos Ticket Renewer to the new host while inheriting the configurations from the existing roles.

### Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Add new hosts for Hue server, Hue Load Balancer, and Kerberos Ticket Renewer to your cluster by going to Hosts Add Hosts .
3. Go to Clusters Hue service Instances and click Add Role Instances.
4. On the **Assign Roles** page, specify the role assignments for Kerberos Ticket Renewer, Load Balancer, and Hue Server by clicking Select hosts.
5. Select the new hosts that you have added to your CDP cluster for each of these roles.
6. Click Continue, review your changes, and exit the wizard.
7. Go back to the Instances tab and start the new role instances by selecting the newly added roles and click Actions for Selected Start .

The newly configured roles inherit settings and configurations from the existing roles, including the configuration in the Advanced Configuration Snippet.

8. Verify that the new roles have the same configuration as the existing roles and the Hue service functions as expected.



**Important:** If you have applied host-specific configuration, then you must manually apply the same configuration to your new host.

9. Delete the Hue roles from the old host by selecting them and clicking Actions for Selected Delete .
10. Restart the Hue service.