Cloudera Runtime 7.3.3

Tuning Hue

Date published: 2020-07-28
Date modified: 2026-12-13



https://docs.cloudera.com/

Legal Notice

© Cloudera Inc. 2026. 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

Adding a load balancer **4**
Configuring high availability for Cloudera Data Explorer (Hue) **4**
Configuring Hive and Impala for high availability with Cloudera Data Explorer (Hue) 5
Configuring for HDFS high availability **8**
Configuring dedicated Impala coordinator **8**
Configuring the Cloudera Data Explorer (Hue) Query Processor scan frequency **9**
Configuring global Hive JDBC URL for Oozie **9**

Adding a load balancer

Cloudera recommends you to configure load balancers for Data Explorer to improve performance or if you have more than 10 concurrent users, and access Data Explorer through the Data Explorer load balancer port (8889).

Procedure

1. Log into Cloudera Manager as an Administrator.
2. Go to Clusters Hue service .
3. Click Actions and select Add Role Instances.
   The Add Role Instances wizard is displayed.
4. On the **Assign Roles** page, click Select hosts for Load Balancer.
   Select a host and click OK.
5. [Optional] Add 2 additional Hue servers (for a total of 3) to boost performance:
   a. Click Select hosts for Hue Server.
   b. Select a host and click OKContinue.
6. Click Select Hosts for Kerberos Ticket Renewer if running Kerberos and select each host that you selected earlier.
   Every Hue host must have a Renewer.
7. Click Continue.
   The newly added hosts are displayed on the Instances tab.
8. Select all the newly added hosts.
9. Click Actions for Selected and select Start.



Note: Hue servers can share hosts with Load Balancers. But Hue servers must be on distinct hosts from other Hue servers, and Load Balancers must be on distinct hosts from other Load Balancers.

10. Restart the Hue service.
11. Click Hue Web UI Load Balanced Hue Web UI .
12. Log on to Hue and ensure the port is 8889.



Tip: The Load Balancer instance can always be accessed on the Hue Instances tab.

Configuring high availability for Cloudera Data Explorer (Hue)

To enable high availabilty for Data Explorer, you must configure at least two Data Explorer servers, each on a different host and a load balancer.For secure connections, enable TLS/SSL and install Kerberos. Configuring Data Explorer for high availability includes configuring Data Explorer, Hive, and Impala.

Before you begin

- Provide SSH network access to host machines with a Data Explorer Server/Kerberos Ticker Renewer role.
- Configure each Data Explorer server to point to an external database. Cloudera recommends to use an external database for clusters with multiple Data Explorer servers so that no matter which server is being used, the data is always accessible. In a multi-server environment with a default embedded database (one per server), the data on server "A" can appear lost when working on server "B" if the server "A" becomes unavailable (which has the Data Explorer backend database) and vice versa.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Add Hue roles:

   Hue HA requires at least two Hue server roles and one Load Balancer role. If the cluster is authenticating with Keberos, you need one Kerberos Ticker Renewer on each host with a Hue Server.

   a) Go to the Hue service and select Actions Add Role Instances .
   b) Click Hue Server, assign to one or more hosts, and click OK Continue .
   c) Click Kerberos Ticket Renewer, assign to each host with a Hue Server, and click OK Continue .
   d) Click Load Balancer, assign to one or more hosts, and click OK Continue .
   e) Check each role and select Actions for Selected Start and click Start.
3. Enable TLS for the Hue Load Balancer:

   Note: You can configure the Load Balancer for TLS/SSL or each endpoint (HS2, Impalad).

   a) Go to Hue Configuration and search on TLS/SSL.
   b) Check Enable TLS/SSL for Hue for the Hue Server Default Group.
   c) Set other TLS/SSL properties appropriate for your setup.

   Some properties to consider are:

   - Hue Load Balancer Port: Apache Load Balancer listens on this port (default is 8889).
   - Path to TLS/SSL Certificate File: Must be multi-domain with CN = Load Balancer in PEM format.
   - Path to TLS/SSL Private Key File: Must be in PEM format.
4. Select the Hue Load Balancer Cookie Refresh option from the Configuration tab to optimize the Hue Load Balancer to distribute users equally across the Hue instances.
5. Click Save Changes.
6. Restart the Hue service.

Configuring Hive and Impala for high availability with Cloudera Data Explorer (Hue)

To configure Hive for high availability with Data Explorer, you must have two or more HiveServer2 roles. For Impala, you must have two or more Impala daemon (impalad) roles.

Before you begin

- SSH network access to host machines with a HiveServer2 or Impala Daemon role.
- External database configured for each H2S and Impala Daemon.
- Data Explorer Load Balancer Hive/Impala Load Balancer configured with Source IP Persistence.

Source IP Persistence

Without IP Persistence, you may encounter the error, “Results have expired, rerun the query if needed.”.

Data Explorer supports High Availability through a "load balancer" to HiveServer2 and Impala. Because the underlying Data Explorer thrift libraries reuse TCP connections in a pool, a single user session may not have the same TCP connection. If a TCP connection is balanced away from a HiveServer2 or Impalad instance, the user session and its queries (running or returned) can be lost and trigger the “Results have expired" error.

To prevent sessions from being lost, configure the Hive/Impala Load Balancer with Source IP Persistence so that each Data Explorer instance sends all traffic to a single HiveServer2/Impala instance. Of course, this is not true load balancing, but a configuration for failover High Availability.

To prevent sessions from timing out while in use, add more Data Explorer Server instances, so that each can be pinned to another HiveServer2/Impala instance. And for both HiveServer2/Impala, set the affinity timeout (that is, the timeout to close persisted sessions) to be longer than the Impala query and session timeouts.

For the best load distribution, create multiple profiles in your load balancer, per port, for both non-Data Explorer clients and Data Explorer clients. Have non-Data Explorer clients distribute loads in a round robin and configure Data Explorer clients with source IP Persistence on dedicated ports, for example, 21000 for impala-shell, 21050 for impala-jdbc, and 21051 for Data Explorer.

Procedure

1. In Cloudera Manager, add roles for HiveServer2 and for Impala daemon:
   a. Configure the cluster with at least two roles for HiveServer2:
      1. Go to the Hive service and select Actions Add Role Instances .
      2. Click HiveServer2, assign one or more hosts, and click OK Continue .
      3. Check each role and select Actions for Selected Start and click Start.
   b. Configure the cluster with at least two roles for Impala Daemon:
      1. Go to the Impala service and select Actions Add Role Instances .
      2. Click Impala Daemon, assign one or more hosts, and click OK Continue .
      3. Check each role and select Actions for Selected Start and click Start.
2. Install a proxy service:

   This is an example of how to add a proxy server for each HiveServer2 and Impala Daemon with multiple profiles using the open source TCP/HTTP load balancer, HAProxy.

   a. Install HAProxy for your operating system:

   ```
   yum install haproxy
   ```

   ```
   apt-get install haproxy
   ```

   ```
   zypper addrepo http://download.opensuse.org/repositories/server:http/SLE
   _12/server:http.repo
   zypper refresh
   zypper install haproxy
   ```

   b. Configure HAProxy for each role, for example:

   ```
   vi /etc/haproxy/haproxy.cfg
   ```

   ```
   listen impala-shell
      bind :21001
      mode tcp
      option tcplog
      balance roundrobin
      stick-table type ip size 20k expire 5m
   server impala_0 shortname-2.domain:21000 check
   server impala_1 shortname-3.domain:21000 check

   listen impala-jdbc
      bind :21051
      mode tcp
      option tcplog
      balance roundrobin
      stick-table type ip size 20k expire 5m
   server impala_0 shortname-2.domain:21050 check
   server impala_1 shortname-3.domain:21050 check
   ```

```
listen impala-hue
   bind :21052
   mode tcp
   option tcplog
   balance source
server impala_0 shortname-2.domain:21050 check
server impala_1 shortname-3.domain:21050 check

listen hiveserver2-jdbc
   bind :10001
   mode http
   option tcplog
   balance roundrobin
   stick-table type ip size 20k expire 5m
server hiveserver2_0 shortname-1.domain:10000 check
server hiveserver2_1 shortname-2.domain:10000 check
listen hiveserver2-hue
   bind :10002
   mode tcp
   option tcplog
   balance source
    stick-table type ip size 20k expire 5m
server hiveserver2_0 host shortname-1.domain:10000 check
server hiveserver2_1 host shortname-2.domain:10000 check
```

Replace shortname-#.domain with those in your environment:

```
sed -i "s/host shortname/your host shortname/g" /etc/haproxy/haproxy.cfg
sed -i "s/domain/your domain/g" /etc/haproxy/haproxy.cfg
```

Hue does not maintain any state for the connection to HS2. If Hue connects to one HS2 instance and the query is being executed on another HS2 instance in an HA setup, then commands including listing tables or databases could fail. Enabling a sticky session using stick table is used to avoid this issue.

When you specify “stick-table type ip size 20k expire 5m”, the stick table tracks the HTTP request rate of each client that passes through the load balancer.

- The table's primary key is of the type ip, which implies that the keys are IP addresses
- The table holds a maximum of 20 thousand records
- A record expires after 5 minutes unless it is accessed during that time

c. Restart haproxy:

```
service haproxy restart
```

d. Run netstat to ensure your proxies are running:

```
netstat | grep LISTEN
```

e. To configure Hue to connect through the HAProxy load balancer:

```
[beeswax]
hive_server_host=<LOAD_BALANCER_HOSTNAME>
hive_server_port=<HIVESERVER2_HUE_PROXY_PORT>
[impala]
server_host=<LOAD_BALANCER_HOSTNAME>
server_port=<IMPALA_HUE_PROXY_PORT>
```

After configuring HAProxy, update the Hue service configuration to use the HAProxy hostname and ports. HAProxy routes requests to the backend services based on the load-balancing policy defined in the proxy configuration.

f. Restart the Hue service.

Configuring for HDFS high availability

You can use Cloudera Manager to configure Data Explorer to use HDFS high availability NameNodes.

Procedure

1. Add the HttpFS role.
2. After the command has completed, go to the Hue service.
3. Click the Configuration tab.
4. Locate the HDFS Web Interface Role property or search for it by typing its name in the Search box.
5. Select the HttpFS role you just created instead of the NameNode role, and save your changes.
6. Restart the Hue service.

Related Information

Using HttpFS to provide access to HDFS

Configuring Cloudera Data Explorer (Hue) to use a dedicated Impala coordinator

If you have separate Impala executors and coordinators on your Cloudera cluster, then you can configure Data Explorer to use a dedicated Impala coordinator for running queries.

About this task

Note: Data Explorer must connect to the same Impala daemon for fetching query results on which the query was executed. Therefore, do not configure the load balancer using a simple round-robin algorithm because the Impala daemons are only aware of the queries they handle.

Procedure

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

```
[impala]
#The Impala coordinator server
server_host=[***COORDINATOR-SERVER***]:[***PORT***]
```

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

Results

Hue uses the dedicated Impala coordinator server to execute queries.

Configuring the Cloudera Data Explorer (Hue) Query Processor scan frequency

You can configure the frequency at which the Data Explorer Query Processor fetches the data from a Hadoop Compatible File System by setting the time in milliseconds (ms) in the “hue.query-processor.event-pipeline.folder-scan-delay-millis” property in the Query Processor configurations.

About this task

By default, the value of the hue.query-processor.event-pipeline.folder-scan-delay-millis property is 5 * 60 * 1000 ms (5 minutes). For a faster data refresh rate, you can set this value to less than 5 * 60 * 1000 ms (5 minutes). But this can result in higher storage read costs. However, if you want stronger performance, and a faster refresh rate is not a priority, then you can set it to a higher value to reduce storage read costs.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Query Processor service Configuration and add the following line in the Query Processor Extra Configurations field:

```
"hue.query-processor.event-pipeline.folder-scan-delay-millis" : [***TIME-IN-MS***]
```

```
"hue.query-processor.event-pipeline.folder-scan-delay-millis" : 2000
```

3. Click Save Changes.
4. Restart the Query Processor service.

Configuring global Hive JDBC URL for Oozie

Learn how to configure a global Hive JDBC URL for Oozie workflows built using Data Explorer. This allows you to set a default connection string across the cluster while retaining the flexibility to override it for individual workflows.

About this task

Data Explorer automatically generates the Hive JDBC URL for Oozie workflows based on the default Hive service configuration. If you require a different connection string, such as a ZooKeeper quorum for High Availability (HA), you previously had to manually specify the URL for every individual Oozie workflow.

You can now define a global oozie_hs2_jdbc_url that applies to all Oozie workflows built using Data Explorer across the cluster. While this setting provides a global default, you can still override it at the individual workflow level if a specific project requires a different connection.

Procedure

1. Log in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue Configuration .
3. Search for the Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini.
4. Add the oozie_hs2_jdbc_url property under the [oozie] section as follows:

```
[oozie]
# Example: Global ZooKeeper quorum-based URL
```

```
oozie_hs2_jdbc_url=jdbc:hive2://<zk_host1>:2181,<zk_host2>:2181/default
;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2
```

Replace the placeholders with your actual ZooKeeper hosts and namespace or the Hive JDBC URL intended to be used by the Oozie workflows.

5. Click Save Changes.
6. Restart the Data Explorer service.