Cloudera Runtime 7.1.6

Tuning Hue

Date published: 2019-07-28
Date modified: 2021-03-19

CLOUDERA

Legal Notice

Contents

Adding a load balancer

Cloudera recommends you to configure load balancers for Hue to improve performance or if you have more than 10 concurrent users, and access Hue through the Hue 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 UILoad 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 Hue

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

Before you begin

- Provide SSH network access to host machines with a Hue Server/Kerberos Ticker Renewer role.
- Configure each Hue server to point to an external database. Cloudera recommends to use an external database for clusters with multiple Hue 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 Hue backend database) and vice versa.

Procedure

1. 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. Log in to Cloudera Manager and go to the Hue service.
   b. Go to the Hue service and select ActionsAdd Role Instances.
   c. Click Hue Server, assign to one or more hosts, and click OKContinue.
   d. Click Kerberos Ticket Renewer, assign to each host with a Hue Server, and click OKContinue.
   e. Click Load Balancer, assign to one or more hosts, and click OKContinue.
   f. Check each role and select Actions for SelectedStart and click Start.

2. Enable TLS for the Hue load balancer:

   > **Note:** You can configure the Load Balancer for TLS/SSL or each endpoint (H2S, Impalad).

   a. Go to HueConfiguration 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 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.

   d. Click Save Changes and Restart Hue.

Configuring Hive and Impala for high availability with Hue

To configure Hive for high availability with Hue, 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.
- Hue 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.

Hue supports High Availability through a "load balancer" to HiveServer2 and Impala. Because the underlying Hue 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 Hue 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 Hue 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-Hue clients and Hue clients. Have non-Hue clients distribute loads in a round robin and configure Hue clients with source IP Persistence on dedicated ports, for example, 21000 for impala-shell, 21050 for impala-jdbc, and 21051 for Hue.

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 ActionsAdd Role Instances.
      2. Click HiveServer2, assign one or more hosts, and click OKContinue.
      3. Check each role and select Actions for SelectedStart and click Start.

   b. Configure the cluster with at least two roles for Impala Daemon:

      1. Go to the Impala service and select ActionsAdd Role Instances.
      2. Click Impala Daemon, assign one or more hosts, and click OKContinue.
      3. Check each role and select Actions for SelectedStart 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 databased 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
```

Configuring for HDFS high availability

You can use Cloudera Manager to configure Hue 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.