Cloudera Runtime 7.3.3

Starting and Stopping Apache Impala

Date published: 2020-11-30
Date modified: 2026-11-13



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

Modifying Impala Startup Options....4
Shut Down Impala....4
Setting Timeouts in Impala....5
Setting Timeout and Retries for Thrift Connections to Backend Client....5
Increasing StateStore Timeout....6
Setting the Idle Query and Idle Session Timeouts....6
Adjusting Heartbeat TCP Timeout Interval....7
Setting Impala query cancellation on shut down....7

Modifying Impala Startup Options

You can view and edit the configuration options for the Impala daemons to customize your Impala environment, such as to specify which hosts and ports to use, to assign directories for logging,and to control resource usage and security.

Configuring Impala Startup Options

Navigate to the following page to configure the settings for all the Impala-related daemons:

ClustersImpalaConfiguration.

If the Cloudera Manager interface does not yet have a form field for a newly added option, or if you need to use special options for debugging and troubleshooting, the Advanced category page for each daemon includes one or more Safety Valve fields where you can enter option names directly.

Checking the Values of Impala Startup Options

You can check the current runtime value of all these settings through the Impala Web UI, available by default at:

- http://*IMPALA_HOSTNAME*:25000/varz for the `impalad` daemon
- http://*IMPALA_HOSTNAME*:25010/varz for the `statestored` daemon
- http://*IMPALA_HOSTNAME*:25020/varz for the `catalogd` daemon

Related Information

Web User Interface for Debugging

Shut Down Impala

Explains how to gracefully shut down Impala Daemons by first allowing running queries a specified amount of time to complete the process.

The flow to gracefully shut down an Impala Daemon is as follows:

1. The shutdown is initiated.
2. The grace period starts. The Impala Daemon informs other coordinators not to schedule any new queries on it. This allows queries already scheduled to run on this daemon by other coordinators to start executing.
3. The grace period expires.
4. The Impala Daemon continuously checks if there are no queries or fragments running.
5. If there are no queries or fragments running, it shuts down.
6. Otherwise, when it reaches the IMPALA_GRACEFUL_SHUTDOWN_DEADLINE duration, Impala Daemon shuts down.

Once Cloudera Manager initiates the stop/shutdown command, the Impala Daemon starts up the graceful shutdown process, and the process cannot be reverted. However, if you need to change the hard deadline in Cloudera Manager, you can cancel the shutdown command, change the Impala Daemon Graceful Shutdown, and start the shutdown command again.

Procedure

1. Optionally, set the grace period.
   a) In Cloudera Manager, navigate to Impala Service Configuration Scope Impala Daemon .
   b) In the Impala Daemon Command Line Argument Advanced Configuration Snippet (Safety Valve) field, specify the grace period:

--shutdown_grace_period_s=<new grace period in seconds>

The default grace period is 2 minutes.

It is strongly recommended that you use the default value and not change the setting.

2. Optionally, set the hard deadline after which Impala is shut down regardless of whether queries are still running on it.
   a) In Cloudera Manager, navigate to Impala Service Configuration Scope Impala Daemon .
   b) In the Impala Graceful Shutdown Deadline field, specify the time to wait for running queries.

      The default is 60 minutes.

      If you specify 0, Impala will shutdown immediately without waiting for running queries
3. In Cloudera Manager, navigate to Impala Service Instances .
4. Click an Impala Daemon role.
5. Click ActionsImpala Daemon Graceful Shutdown.

Related Information

SHUTDOWN statement

Setting Timeouts in Impala

Depending on how busy your cluster is, you might increase or decrease various timeout values. Increase timeouts if Impala is cancelling operations prematurely, when the system is responding slower than usual but the operations are still successful if given extra time. Decrease timeouts if operations are idle or hanging for long periods, and the idle or hung operations are consuming resources and reducing concurrency.

Related Information

Impala datastream sender timeout

Setting Timeout and Retries for Thrift Connections to Backend Client

Impala connections to the backend client are subject to failure in cases when the network is momentarily overloaded.

About this task

To avoid failed queries due to transient network problems, you can configure the number of Thrift connection retries using the following option:

Procedure

1. In Cloudera Manager, navigate to Impala service Configuration .
2. In the Impala Daemon Command Line Argument Advanced Configuration Snippet (Safety Valve) field, specify the following.

   To avoid failed queries due to transient network problems, you can configure the number of Thrift connection retries using the following option:

   - The --backend_client_connection_num_retries option specifies the number of times Impala will try connecting to the backend client after the first connection attempt fails. By default, `impalad` will attempt three re-connections before it returns a failure.

   You can configure timeouts for sending and receiving data from the backend client. Therefore, if for some reason a query does not respond, instead of waiting indefinitely for a response, Impala will terminate the connection after a configurable timeout.

- The --backend_client_rpc_timeout_ms option can be used to specify the number of milliseconds Impala should wait for a response from the backend client before it terminates the connection and signals a failure. The default value for this property is 300000 milliseconds, or 5 minutes.

3. Click Save Changes and restart Impala.

Increasing StateStore Timeout

If you have an extensive Impala schema, for example, with hundreds of databases, tens of thousands of tables, you might encounter timeout errors during startup as the Impala catalog service broadcasts metadata to all the Impala nodes using the StateStore service. To avoid such timeout errors on startup, increase the StateStore timeout value from its default of 10 seconds.

About this task

Increase the timeout value of the StateStore service if you see messages in the impalad log such as:

```
Connection with state-store lost
Trying to re-register with state-store
```

Procedure

1. In Cloudera Manager, navigate to Impala service Configuration .
2. In the search field, type -statestore_subscriber_timeout_seconds.
3. In the StateStoreSubscriber Timeout field, specify a new timeout value larger than the current value.
4. Click Save Changes and restart Impala.

Setting the Idle Query and Idle Session Timeouts

To keep long-running queries or idle sessions from tying up cluster resources, you can set timeout intervals for both individual queries, and entire sessions.

About this task

Procedure

1. In Cloudera Manager, navigate to Impala service Configuration .
2. In the search field, type idle.
3. In the Idle Query Timeout field, specify the time in seconds after which an idle query is cancelled.

   This could be a query whose results were all fetched but was never closed, or one whose results were partially fetched and then the client program stopped requesting further results. This condition is most likely to occur in a client program using the JDBC or ODBC interfaces, rather than in the interactive `impala-shell` interpreter. Once a query is cancelled, the client program cannot retrieve any further results from the query.

   You can reduce the idle query timeout by using the QUERY_TIMEOUT_S query option at the query level. Any non-zero value specified in this field serves as an upper limit for the QUERY_TIMEOUT_S query option.

   The value of 0 disables query timeouts.
4. In the Idle Session Timeout field, specify the time in seconds after which an idle session expires.

   A session is idle when no activity is occurring for any of the queries in that session, and the session has not started any new queries. Once a session is expired, you cannot issue any new query requests to it. The session remains open, but the only operation you can perform is to close it.

The default value of 0 specifies sessions never expire.

You can override this setting with the IDLE_SESSION_TIMEOUT query option at the session or query level.

5. Click Save Changes and restart Impala.
6. Under Configuration files, select the flagfile.

Results

Impala checks periodically for idle sessions and queries to cancel. The actual idle time before cancellation might be up to 50% greater than the specified configuration setting. For example, if the timeout setting was 60, the session or query might be cancelled after being idle between 60 and 90 seconds.

Adjusting Heartbeat TCP Timeout Interval

Using the TCP flag, you can prevent the Statestore from waiting indefinitely for a response from the subscribers that fail to respond to the heartbeat RPC within the set period.

About this task

This flag statestore_heartbeat_tcp_timeout_seconds defines the time that may elapse before a heartbeat RPC connection request from a Statestore server to an Impalad or a Catalog server (subscribers) should be considered dead.

You can increase the flag value if you see intermittent heartbeat RPC timeouts listed in the statestore's log. You may find the max value of "statestore.priority-topic-update-durations" metric on the statestore to get an idea of a reasonable value to be used in this flag.



Note: The priority topic updates are only small amounts of data that take little time to process, similar to the heartbeat complexity.

Procedure

1. In Cloudera Manager, navigate to Impala service Configuration .
2. In the Impala Daemon Command Line Argument Advanced Configuration Snippet (Safety Valve) field, add the flag statestore_heartbeat_tcp_timeout_seconds with an appropriate value.
3. You can also control the maximum number of consecutive heartbeat messages an impalad can miss before being declared failed by the statestore by adding this flag statestore_max_missed_heartbeats. Typically, you will not have to change this value.
4. Click Save Changes and restart Impala.

Setting Impala query cancellation on shut down

Learn how to configure Impala to cancel running queries before the shut down deadline, ensuring a controlled shut down and preventing resource leaks.

About this task

The Impala query cancellation on shut down is available from Cloudera Runtime 7.3.1.500 SP3.

When the graceful shutdown deadline is reached, the Impala Daemon exits immediately, leaving queries unfinished. This can lead to unreleased resources, such as scratch files in remote storage.

To improve this process, a new state has been added to the graceful shut down procedure. Before reaching the shut down deadline, Impala attempts to cancel any remaining running queries within a configurable time limit, shutdown _query_cancel_period_s.

- The default value is 60 seconds.
- If the specified value exceeds 20% of the total shut down deadline, it is automatically capped at that value.
- If this value is set to 0, the shutdown query cancellation state is disabled, and Impala waits directly until the shut down deadline.

How query cancellation works during shut down

1. The shut down process is initiated.
2. Before the shutdown deadline, Impala enters a query cancellation phase if shut down query cancellation state is enabled.
3. Any remaining running queries are canceled within the shutdown_query_cancel_period_s duration.
4. If all queries are successfully canceled within this period, Impala shuts down immediately.
5. If queries are still running when the deadline is reached, Impala shuts down with those queries still active.

To configure query cancellation before the shut down deadline, follow these steps:

Results

Configuring this setting, Impala attempts to cancel running queries before the shut down deadline, ensuring a more controlled shut down process.