

Cloudera Runtime 7.1.8

## Hive Troubleshooting

Date published: 2022-12-16

Date modified:

# CLOUdera

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

# Legal Notice

© Cloudera Inc. 2025. 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>HeapDumpPath (/tmp) in Hive data nodes gets full due to .hprof files.....</b>	<b>4</b>
<b>Query fails with "Counters limit exceeded" error message.....</b>	<b>5</b>
<b>HiveServer is unresponsive due to large queries running in parallel.....</b>	<b>5</b>
<b>Whitelisting Configurations at the Session Level.....</b>	<b>6</b>
<b>Mitigating JSON SerDe failures after migration.....</b>	<b>7</b>

# HeapDumpPath (/tmp) in Hive data nodes gets full due to .hprof files

## Condition

The HeapDumpPath (/tmp) in Hive nodes often gets full because of the .hprof files that are generated. Manually deleting these files to free up space does not always help because the files are generated every time there is an exception. Learn how to resolve this issue by modifying the relevant Tez properties.

## Cause

A .hprof file is created when a process exits with an Out Of Memory exception, and the file captures the state of Java Virtual Machine (JVM) when it crashes.

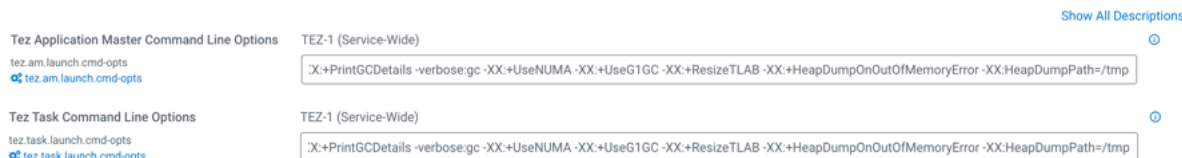
The default value for HeapDumpPath in the Tez Application Master and Tez Task Command Line Options properties is set to /tmp and therefore, Tez saves all the heap dump files in this location.

If you do not want the heap dump files to be saved in /tmp, you can either set the HeapDumpPath value to a different location on the node or choose not to capture heap dump (.hprof) files when a process exits.

## Solution

### Procedure

1. Log in to Cloudera Manager as an administrator.
2. Go to **Clusters Tez Configuration** and search for the 'Tez Application Master Command Line Options' and 'Tez Task Command Line Options' properties.



3. Choose one of the following solutions:

#### If you want to...

**Choose a different location in the Hive node to capture heap dumps**

**Disable automatic capture of heap dumps**

#### Then...

Modify the value of XX:HeapDumpPath in the above properties from /tmp to a different location, for example, /var/tmp.

Remove -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tmp parameters from the above properties.



**Note:** You can add these parameters when you want to work with heap dumps or when you require heap dumps for debugging purposes.

4. Click **Save Changes**.
5. Restart the Tez service.

## Query fails with "Counters limit exceeded" error message

### Condition

After upgrading to CDP Private Cloud Base, you may notice that some Hive queries fail with a "Counters limit exceeded: Too many counters: 10001 max=10000" and "Counters limit exceeded: Too many counter groups: 3001 max=3000".


### Cause

This issue occurs because a lower value is specified for Apache Tez task counters (tez.counters.max) and counter groups (tez.counters.max.groups), which prevents Tez from executing the DAG. If you are running long queries, increase the value for Tez counters and counter groups.

It is recommended that you set the value of tez.counters.max at the session level and also add it to the allowlist.

### Solution

#### Procedure

1. Log in to Cloudera Manager as an administrator.
2. Go to [Clusters Hive on Tez Configuration](#) and search for 'HiveServer2 Advanced Configuration Snippet (Safety Valve) for hive-site.xml'.
3. Click  and add the property key: hive.security.authorization.sqlstd.confwhitelist.append.
4. Provide the property value, or values, to allowlist, for example: tez\counters\.\* or tez\counters\max|tez\counters\max\groups.  
This action appends the parameters to the allowlist.
5. Save the changes and restart the Hive on Tez service.
6. From the Beeline shell, set the tez.counters.max property to a higher value and run the query.  
set tez.counters.max=50000;  
Run the Hive query and if it continues to fail, perform the steps provided below.
7. In Cloudera Manager, go to [Clusters Tez Configuration](#) and search for the tez.counters.max property.
8. Modify the value to 50000, save changes, and refresh the Tez service.
9. Restart the Hive on Tez service and run the Hive query again.  
If you encounter the following error, modify the value for tez.counters.max.groups.

```
ERROR : Counters limit exceeded: org.apache.tez.common.counters.LimitExceededException: Too many counter groups: 3001 max=3000
```

10. In Cloudera Manager, go to [Clusters Tez Configuration](#) and search for the tez.counters.max.groups property.
11. Modify the value to 10000, save changes, and refresh the Tez service.
12. Restart the Hive on Tez service and run the Hive query.

## HiveServer is unresponsive due to large queries running in parallel

### Condition

HiveServer (HS2) is blocking user queries or sessions because of multiple Hive sessions running in parallel.

### Cause

The issue occurs when a large query is running and you submit the same query again. The `hive.query.results.cache.wait.for.pending.results` property forces a query to wait for the pending results of an already running query, in order to use the cached result when it is ready.

You can choose to disable this property to avoid blocking queries submitted by other users or you can avoid running the same large queries parallelly.

### Remedy

#### Procedure

1. Go to **Clusters Hive on Tez Configuration** and add the following property name and its value in the **HiveServer2 Advanced Configuration Snippet (Safety Valve)** for `hive-site.xml` field:

Name: `hive.query.results.cache.wait.for.pending.results`

Value: `false`

2. Save the changes and restart the Hive on Tez service.

## Whitelisting Configurations at the Session Level

### Condition

Users encounter errors in Hive on Tez when attempting to modify properties at runtime using tools like Beeline or SQL Workbench. Beeline error include:

```
Error while processing statement: Cannot modify hive.input.format at runtime
.
It is not in the list of params that are allowed to be modified at runtime
(state=42000, code=1).
```

### Cause

The errors occur due to misconfigured or missing properties in the `hive.security.authorization.sqlstd.confwhitelist` or `hive.security.authorization.sqlstd.confwhitelist.append` parameters. Specifically:

1. The required property is not included in the whitelist.
2. Misconfigurations in the whitelist, such as:
  - Using `|` at the beginning of entries.
  - Improper use of `.` in property names, such as failing to escape it or incorrectly adding it to configurations.

### Solution

#### Procedure

1. In `hivesite.xml`, navigate to **Clusters Hive on Tez Configurations**
2. Search for **hive-site.xml** and locate the 'Hive Service Advanced Configuration Snippet (Safety Valve)'
3. Add or update the required properties in the `hive.security.authorization.sqlstd.confwhitelist.append` parameter.

For Example:

```
<property>
<name>hive.security.authorization.sqlstd.confwhitelist.append</name>
<value>hive\input\format\.*|tez\queue\name|my\third\property|principal|sslTrustStore</value>
```

```
</property>
```



**Note:** While adding properties to the whitelist:

- Define each property as a regular expression, separated by |
- Ensure the list does not start with |
- Use \ to escape all . characters

4. Restart the Hive on Tez service for the changes to take effect.

## Mitigating JSON SerDe failures after migration

This topic provides a solution for issues with JSON SerDe functionality that can occur after migrating from older releases.

### Condition

Users migrating from CDH or HDP to who utilize the hive-catalog-core-3.1.3000[...] .jar for JSON SerDe functionality encounter a `java.lang.ClassNotFoundException` post-upgrade.

### Cause

This issue is caused by the use of an outdated SerDe class. The `JsonSerDe` class was moved to Hive from HCatalog and its previous location in the hive-catalog-core jar is deprecated in , leading to the `ClassNotFoundException` [[HIVE-19211](#)]

### Soultion

### Procedure

```
ALTER TABLE <tbl_name> SET SERDE 'org.apache.hadoop.hive.serde2.JsonSerDe';
```

To resolve this issue, you must update the table to use the new `JsonSerDe` class. The recommended solution is to run the above command.