Cloudera Runtime 7.3.3

Configuring YARN Log Aggregation

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

YARN log aggregation overview....4
Log aggregation file controllers....4
Configuring log aggregation....5
Log aggregation properties....5
Configuring debug delay....7

YARN log aggregation overview

The YARN Log Aggregation feature enables you to move local log files of any application onto HDFS or a cloud-based storage depending on your cluster configuration.

Application logs has great significance: meaningful information can be extracted from them, can be used to debug issues or can be kept for historical analysis. YARN can move local logs securely onto HDFS or a cloud-based storage, such as AWS. This allows the logs to be stored for a much longer time than they could be on a local disk, allows faster search for a particular log file and optionally can handle compression.

The Log Aggregation Retention Period is set using the yarn.log-aggregation.retain-seconds property. The default value of Java Heap Size (JHS) of JobHistory Server is 1GB. Cloudera recommends to use approximately 10KB per job of JHS Heap memory. For example, 50,000 jobs uses at least 0.5GB of memory. Ensure that your JHS heap is large enough to cache all of your jobs.

Increasing the yarn.log-aggregation.retain-seconds property increases the number of jobs. For example, if yarn.log-aggregation.retain-seconds is set to 180 days, and there are 3000 jobs daily and each job requires 10kB. The heap size is 180*3000*10 = 5.4 GB.



Note: When the value of yarn.log-aggregation.retain-seconds is set to more than 15 days , ensure to increase the heap for the JHS to prevent a JHS crash.

Log aggregation file controllers

By default, log aggregation supports two file controllers, TFile and IFile. You can also add your own custom file controller.

By default IFile is used to write the aggregated logs.



Note: TFile and IFile are commonly referred as file formats. However, they do not only define the format of the file, but also how the log aggregation is managed. Therefore referring to them as file controllers specifies them more clearly.

TFile

TFile is the legacy file controller in YARN. It is reliable and well tested. Its buffer and chunk sizes are configurable.

TFile provides the following features:

- Block compression
- Named metadata blocks
- Sorted or unsorted keys
- Seek by key or by file offset

IFile

IFile is a newer file controller than TFile. It also uses TFile internally so it provides the same features as TFile.

In an IFile the files are indexed so it is faster to search in the aggregated log file than in a regular TFile. It uses checksums and temporary files which help to prevent failures. Its buffer sizes and rollover file size are configurable on top of the configuration options of TFile.

Configuring log aggregation

Log aggregation is enabled by default. You can configure it using Cloudera Manager.

Procedure

1. In Cloudera Manager, select the YARN service.
2. Click the Configuration tab.
3. Select the Log Aggregation filter under Category.
   Every log aggregation related property is displayed.
4. Find the Enable Log Aggregation property and ensure that it is selected, meaning that log aggregation is enabled.
5. Configure the log aggregation properties as applicable.

   

   Important:

   The remote-app-log-dir and remote-app-log-dir-suffix configuration properties are not backward compatible. If you change their value later, the logs of applications that are already finished cannot be read.
6. Click Save Changes.

Related Information

Log aggregation properties

Log aggregation properties

You can configure YARN log aggregation for your cluster by specifying values of specific properties depending on your requirements.

Name	Property	Default value	Description
Enable Log Aggregation	yarn.log-aggregation-enable	Enabled	Specifies if log aggregation is enabled. If log aggregation is disabled, YARN container logs are not aggregated.
Log Aggregation Retention Period	yarn.log-aggregation.retain-seconds	7 days	Specifies how long aggregation logs are kept before they are deleted.
Log Aggregation Status Timeout	yarn.log-aggregation-status.time-out.ms	10 minutes	Specifies the maximum amount of time that the NodeManager has for reporting a container's log aggregation status. If no log aggregation status is sent by the NodeManager within the allotted time, the ResourceManager reports a TIME_OUT log aggregation status for that container.

Name	Property	Default value	Description
Supported Log Aggregation File Formats	yarn.log-aggregation.file-formats	IFile and TFile	Specifies which log file formats are supported. The first file format in the list is used to write the aggregated logs. TFile format is always added to the end of the list, to support backward compatibility.
Remote App Log Directory for TFile Format	yarn.log-aggregation.TFile.remote-app-log-dir	empty	Specifies the path of the directory where application logs are stored after an application is completed if TFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix for TFile Format	yarn.log-aggregation.TFile.remote-app-log-dir-suffix	empty	The remote log directory is created at {remote-app-log-dir}/${user}/{thisParam} if TFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir-suffix). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory for IFile Format	yarn.log-aggregation.IFile.remote-app-log-dor	empty	Specifies the path of the directory where application logs are stored after an application is completed if IFile format is given as the file format for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix for IFile Format	yarn.log-aggregation.IFile.remote-app-log-dir-suffix	empty	The remote log directory is created at {remote-app-log-dir}/${user}/{thisParam} if IFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir-suffix). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.

Name	Property	Default value	Description
Log Aggregation Compression Type	yarn.nodemanager.log-aggregation.compression-type	none	Specifies the TFile compression type used to compress aggregated logs. Its value can be: gz, lzo or none.
Remote App Log Directory	yarn.nodemanager.remote-app-log-dir	/tmp/logs	Specifies the path of the directory where application logs are stored after an application is completed. This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix	yarn.nodemanager.remote-app-log-dir-suffix	logs	The remote log dir will be created at {yarn.nodemanager.remote-app-log-dir}/${user}/{thisParam}. This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Log Aggregation Policy	yarn.nodemanager.log-aggregation.policy.class	All	Specifies the types of container logs that are uploaded during the log aggregation. Valid values are: Only AM, AM or Failed, All, Only Failed, Failed or Killed, None, or Sample.
Log Aggregation by NodeManager	yarn.nodemanager.logaggregation.threadpool-size-max	100	Specifies the threadpool size, that is, the number of threads used for the Log Aggregation by NodeManager.

Related Information

Configuring log aggregation

Configuring debug delay

You can keep the application's localized files and log directory on the local disk for a predefined time before the NodeManager deletes them.

About this task

This feature can be useful for example when there is a problem with the log aggregation feature or if you want to diagnose YARN application issues.

Procedure

1. In Cloudera Manager, select the YARN service.
2. Click the Configuration tab.
3. Search for deletion delay.
4. Find the Localized Dir Deletion Delay property.
5. Define the delay time in seconds.

   If you want to diagnose YARN application issues, set this property to a large value, for example to 600 seconds, to permit examination of these directories.

6. Click Save Changes.