Cloudera Runtime 7.3.2

Ranger Auditing

Date published: 2020-07-28
Date modified: 2026-03-31

https://docs.cloudera.com/

Legal Notice

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

Audit Overview

Apache Ranger provides a centralized framework for collecting access audit history and reporting data, including filtering on various parameters. Ranger enhances audit information obtained from Hadoop components and provides insights through this centralized reporting capability.

Ranger plugins support storing audit data to multiple audit destinations.

Solr

The Solr audit destination is a short term audit destination (with a default TTL of 90 days) managed by Solr which can be configured by a Ranger Admin user. The Ranger Admin Web UI displays the access audit data from the audit data stored in Solr.

HDFS

The HDFS audit destination is a long term audit destination for archival/compliance purposes. The HDFS audit destination has no default retention/purge period. A customer must manage the storage/retention/purge/archival of audit data stored in HDFS manually.

Related Information

Configuring Ranger audit properties for Solr

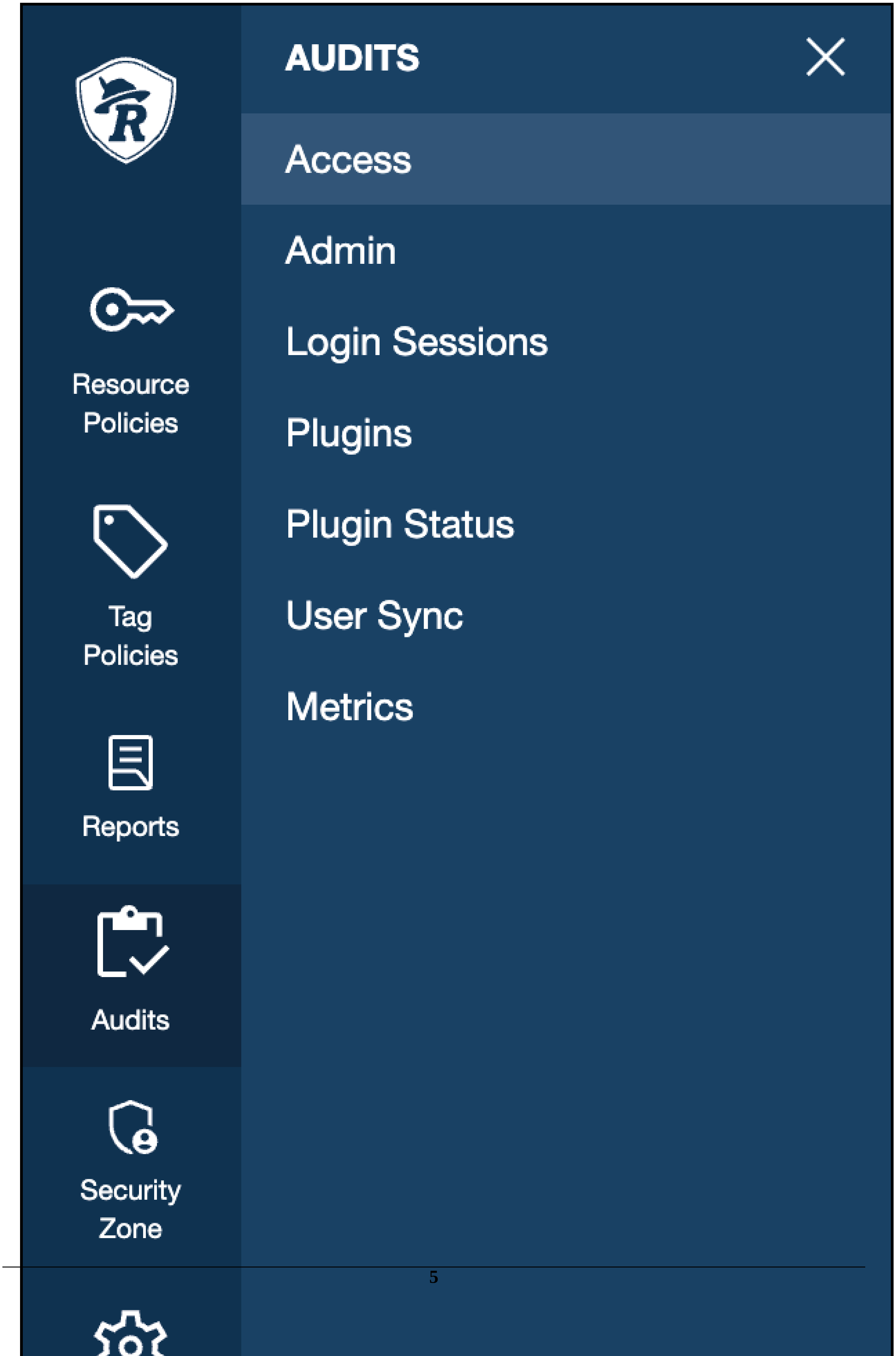
Configuring Ranger audit properties for HDFS

Managing Auditing with Ranger

You can manage auditing using the Audit page in the Ranger Admin Web UI.

To explore options for auditing policies, click Audits

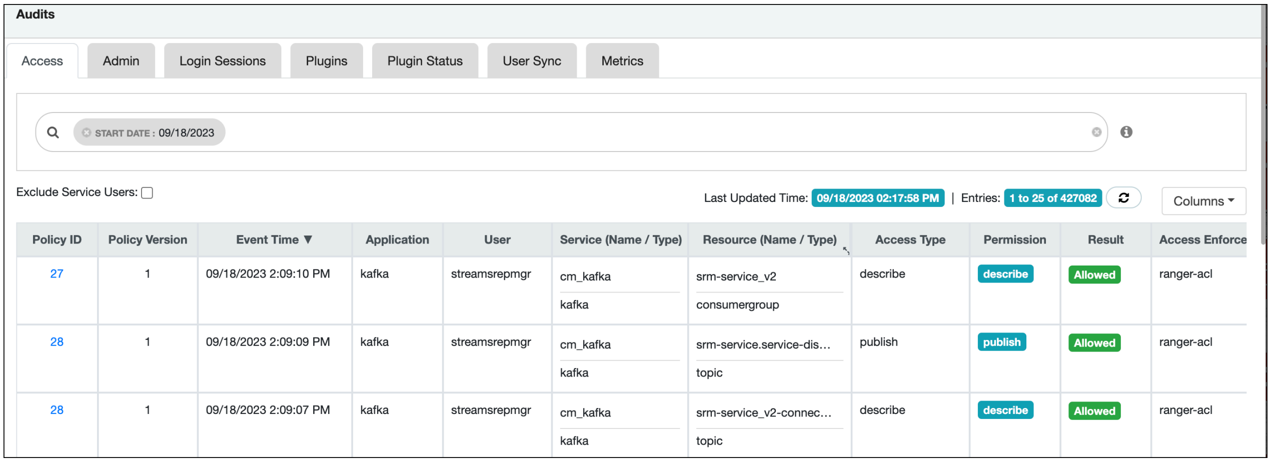
Figure 1: Ranger Admin Audits Menu


AUDITS
Access
Admin
Login Sessions
Plugins
Plugin Status
User Sync
Metrics
Resource Policies
Tag Policies
Reports
Audits
Security Zone

in the left menu of the Ranger Admin Web UI,

then, choose one of the following options:

- Access
- Admin
- Login sessions
- Plugins
- Plugin Status
- User Sync
- Metrics



Exclude Service Users: □

Last Updated Time: 09/18/2023 02:17:58 PM | Entries: 1 to 25 of 427082

Policy ID	Policy Version	Event Time ▼	Application	User	Service (Name / Type)	Resource (Name / Type)	Access Type	Permission	Result	Access Enforce
27	1	09/18/2023 2:09:10 PM	kafka	streamsrepmgr	cm_kafka kafka	srm-service_v2 consumergroup	describe	describe	Allowed	ranger-acl
28	1	09/18/2023 2:09:09 PM	kafka	streamsrepmgr	cm_kafka kafka	srm-service.service-dis… topic	publish	publish	Allowed	ranger-acl
28	1	09/18/2023 2:09:07 PM	kafka	streamsrepmgr	cm_kafka kafka	srm-service_v2-connec… topic	describe	describe	Allowed	ranger-acl

See related topics for more information about using the Audits pages to manage auditing.

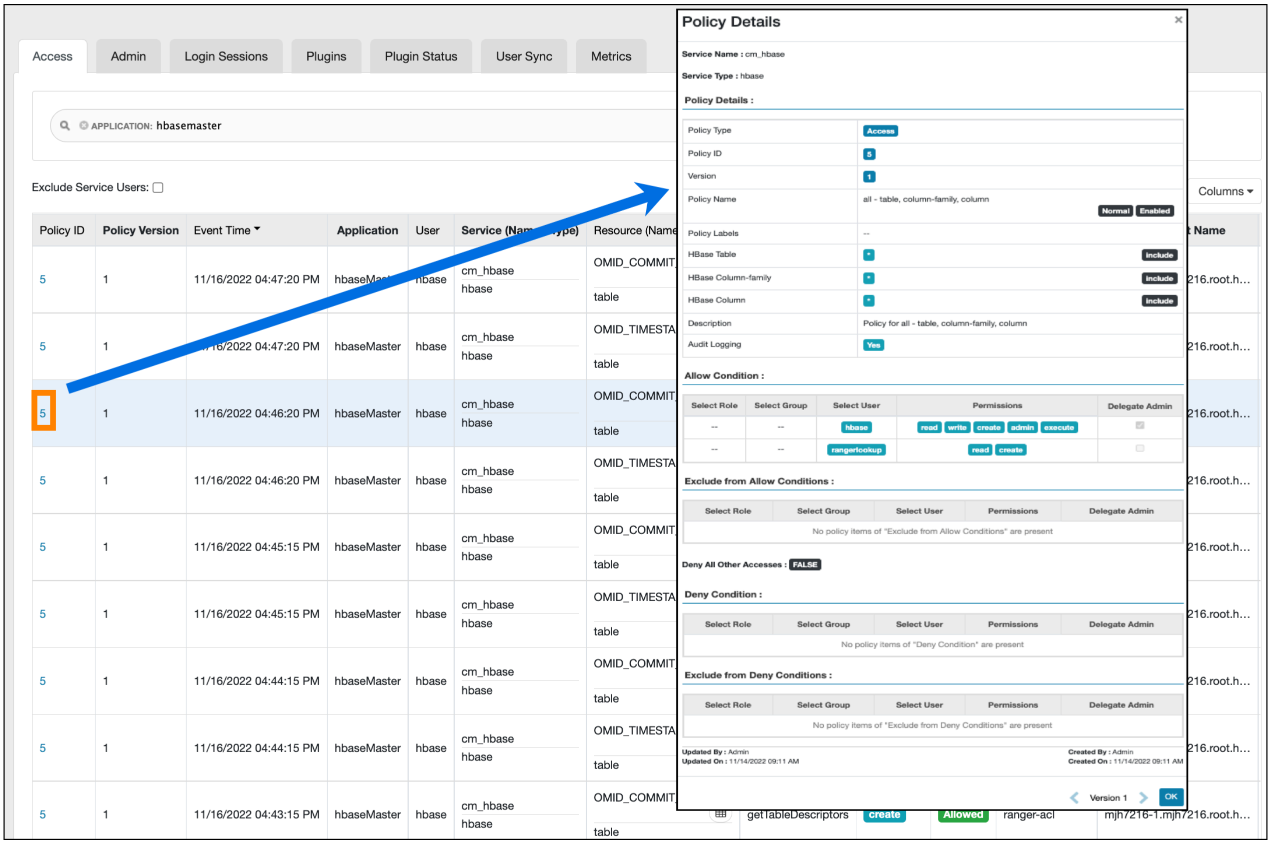
Viewing audit details

How to view policy and audit log details in Ranger audits.

Procedure

To view policy details for a specific audit log, click Access Policy ID .

Audit > Access: hbasemaster



APPLICATION: hbasemaster

Exclude Service Users: □

Policy ID	Policy Version	Event Time	Application	User	Service (Nam ype)	Resource (Name					t Name
5	1	11/16/2022 04:47:20 PM	hbaseMa	hbase	cm_hbase hbase	OMID_COMMIT table					216.root.h...
5	1	/16/2022 04:47:20 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_TIMESTA table					216.root.h...
5	1	11/16/2022 04:46:20 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_COMMIT table					216.root.h...
5	1	11/16/2022 04:46:20 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_TIMESTA table					216.root.h...
5	1	11/16/2022 04:45:15 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_COMMIT table					216.root.h...
5	1	11/16/2022 04:45:15 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_TIMESTA table					216.root.h...
5	1	11/16/2022 04:44:15 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_COMMIT table					216.root.h...
5	1	11/16/2022 04:44:15 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_TIMESTA table					216.root.h...
5	1	11/16/2022 04:43:15 PM	hbaseMaster	hbase	cm_hbase hbase	OMID_COMMIT table	getTableDescriptors	create	Allowed	ranger-acl	mjh7216-1.mjh7216.root.h...

Policy Details ×

Service Name : cm_hbase

Service Type : hbase

Policy Details :

Policy Type	Access
Policy ID	5
Version	1
Policy Name	all - table, column-family, column Normal Enabled
Policy Labels	--
HBase Table	* include
HBase Column-family	* include
HBase Column	* include
Description	Policy for all - table, column-family, column
Audit Logging	Yes

Allow Condition :

Select Role	Select Group	Select User	Permissions	Delegate Admin
--	--	hbase	read write create admin execute	☑
--	--	rangerlookup	read create	□

Exclude from Allow Conditions :

Select Role	Select Group	Select User	Permissions	Delegate Admin
No policy items of "Exclude from Allow Conditions" are present				

Deny All Other Accesses : FALSE

Deny Condition :

Select Role	Select Group	Select User	Permissions	Delegate Admin
No policy items of "Deny Condition" are present				

Exclude from Deny Conditions :

Select Role	Select Group	Select User	Permissions	Delegate Admin
No policy items of "Exclude from Deny Conditions" are present				

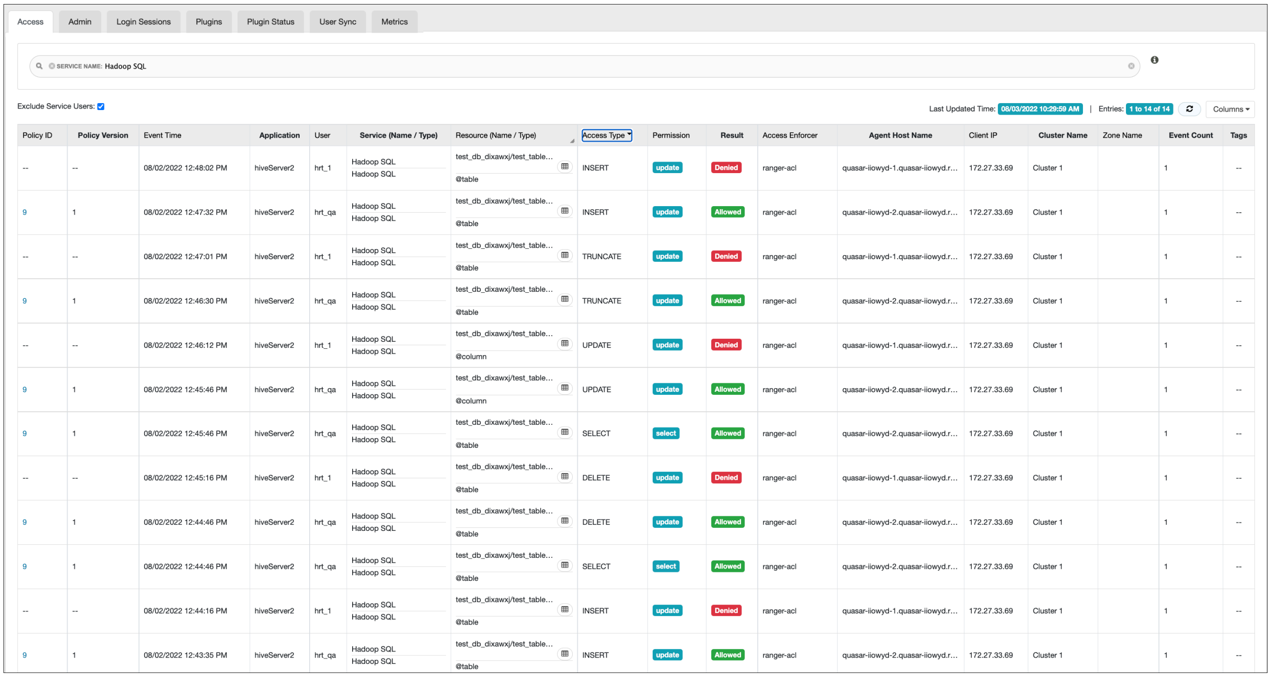
Updated By : Admin
Updated On : 11/14/2022 09:11 AM

Created By : Admin
Created On : 11/14/2022 09:11 AM

Version 1 OK

Audit > Access: HadoopSQL

Note: The Hive plugin audit handler now logs UPDATE operations as INSERT, UPDATE, DELETE, and TRUNCATE specifically.



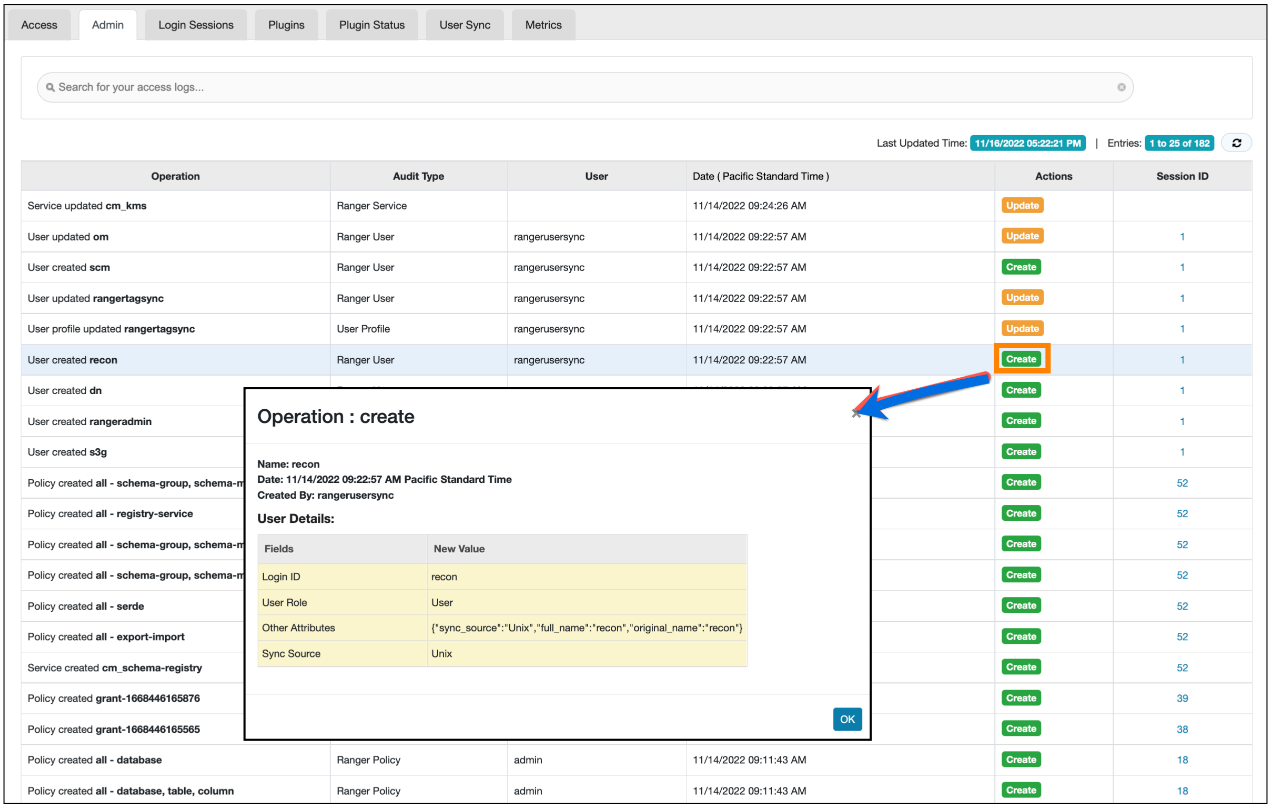
SERVICE NAME: Hadoop SQL

Exclude Service Users: ☑

Last Updated Time: 08/03/2022 10:29:59 AM | Entries: 1 to 14 of 14

Policy ID	Policy Version	Event Time	Application	User	Service (Name / Type)	Resource (Name / Type)	Access Type	Permission	Result	Access Enforcer	Agent Host Name	Client IP	Cluster Name	Zone Name	Event Count	Tags
--	--	08/02/2022 12:48:02 PM	hiveServer2	hrt_1	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	INSERT	update	Denied	ranger-acl	quasar-iiowyd-1.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:47:32 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	INSERT	update	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
--	--	08/02/2022 12:47:01 PM	hiveServer2	hrt_1	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	TRUNCATE	update	Denied	ranger-acl	quasar-iiowyd-1.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:46:30 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	TRUNCATE	update	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
--	--	08/02/2022 12:46:12 PM	hiveServer2	hrt_1	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @column	UPDATE	update	Denied	ranger-acl	quasar-iiowyd-1.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:45:46 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @column	UPDATE	update	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:45:46 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	SELECT	select	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
--	--	08/02/2022 12:45:16 PM	hiveServer2	hrt_1	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	DELETE	update	Denied	ranger-acl	quasar-iiowyd-1.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:44:46 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	DELETE	update	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:44:46 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	SELECT	select	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
--	--	08/02/2022 12:44:16 PM	hiveServer2	hrt_1	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	INSERT	update	Denied	ranger-acl	quasar-iiowyd-1.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--
9	1	08/02/2022 12:43:35 PM	hiveServer2	hrt_qa	Hadoop SQL Hadoop SQL	test_db_dixawxj/test_table... @table	INSERT	update	Allowed	ranger-acl	quasar-iiowyd-2.quasar-iiowyd.r...	172.27.33.69	Cluster 1		1	--

Audit > Admin: Create



Last Updated Time: 11/16/2022 05:22:21 PM | Entries: 1 to 25 of 182

Operation	Audit Type	User	Date (Pacific Standard Time)	Actions	Session ID
Service updated **cm_kms**	Ranger Service		11/14/2022 09:24:26 AM	Update	
User updated **om**	Ranger User	rangerusersync	11/14/2022 09:22:57 AM	Update	1
User created **scm**	Ranger User	rangerusersync	11/14/2022 09:22:57 AM	Create	1
User updated **rangertagsync**	Ranger User	rangerusersync	11/14/2022 09:22:57 AM	Update	1
User profile updated **rangertagsync**	User Profile	rangerusersync	11/14/2022 09:22:57 AM	Update	1
User created **recon**	Ranger User	rangerusersync	11/14/2022 09:22:57 AM	Create	1
User created **dn**				Create	1
User created **rangeradmin**				Create	1
User created **s3g**				Create	1
Policy created **all - schema-group, schema-m**				Create	52
Policy created **all - registry-service**				Create	52
Policy created **all - schema-group, schema-m**				Create	52
Policy created **all - schema-group, schema-m**				Create	52
Policy created **all - serde**				Create	52
Policy created **all - export-import**				Create	52
Service created **cm_schema-registry**				Create	52
Policy created **grant-1668446165876**				Create	39
Policy created **grant-1668446165565**				Create	38
Policy created **all - database**	Ranger Policy	admin	11/14/2022 09:11:43 AM	Create	18
Policy created **all - database, table, column**	Ranger Policy	admin	11/14/2022 09:11:43 AM	Create	18

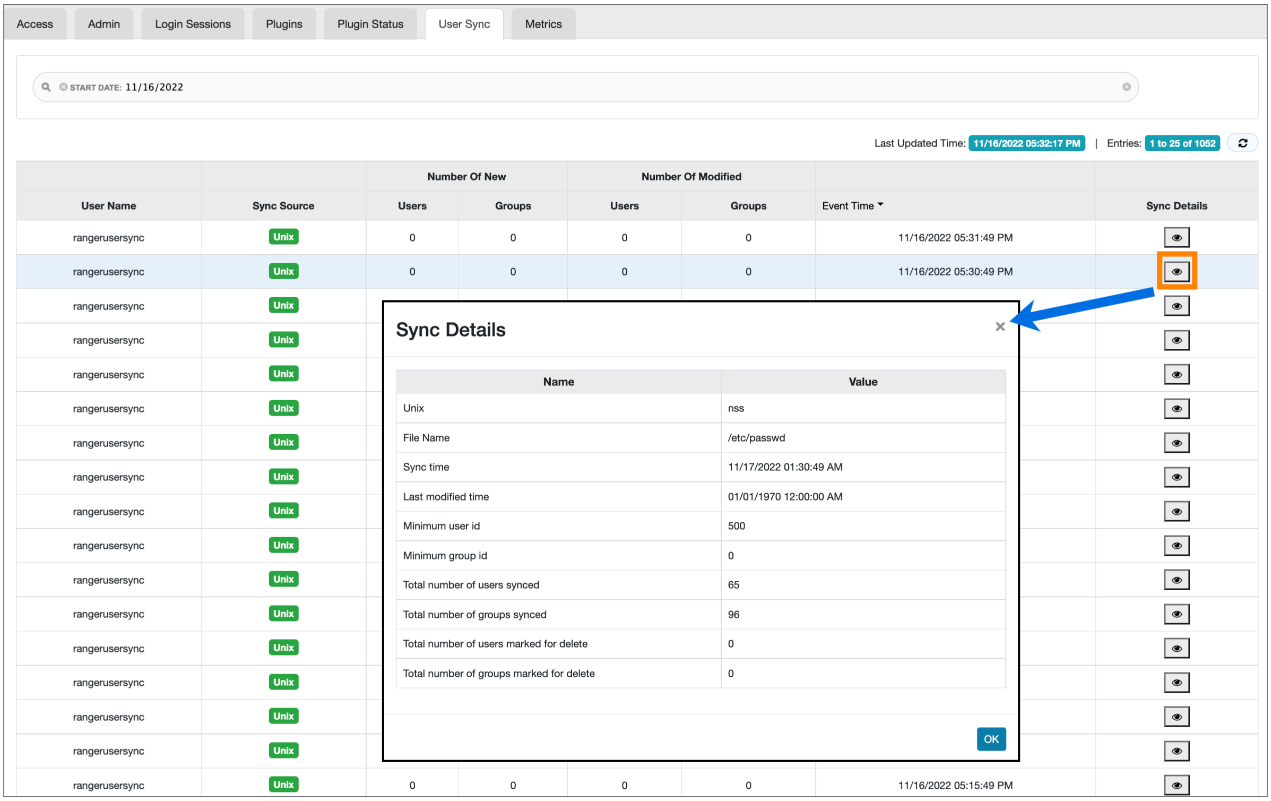
Operation : create

Name: recon
Date: 11/14/2022 09:22:57 AM Pacific Standard Time
Created By: rangerusersync

User Details:

Fields	New Value
Login ID	recon
User Role	User
Other Attributes	{"sync_source":"Unix","full_name":"recon","original_name":"recon"}
Sync Source	Unix

Audit > User Sync: Sync details



Last Updated Time: 11/16/2022 05:32:17 PM | Entries: 1 to 25 of 1052

		Number Of New		Number Of Modified			
User Name	Sync Source	Users	Groups	Users	Groups	Event Time	Sync Details
rangerusersync	Unix	0	0	0	0	11/16/2022 05:31:49 PM	👁
rangerusersync	Unix	0	0	0	0	11/16/2022 05:30:49 PM	👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix						👁
rangerusersync	Unix	0	0	0	0	11/16/2022 05:15:49 PM	👁

Sync Details

Name	Value
Unix	nss
File Name	/etc/passwd
Sync time	11/17/2022 01:30:49 AM
Last modified time	01/01/1970 12:00:00 AM
Minimum user id	500
Minimum group id	0
Total number of users synced	65
Total number of groups synced	96
Total number of users marked for delete	0
Total number of groups marked for delete	0

Viewing audit metrics

How to view audit metrics information using the Ranger Admin Web UI.

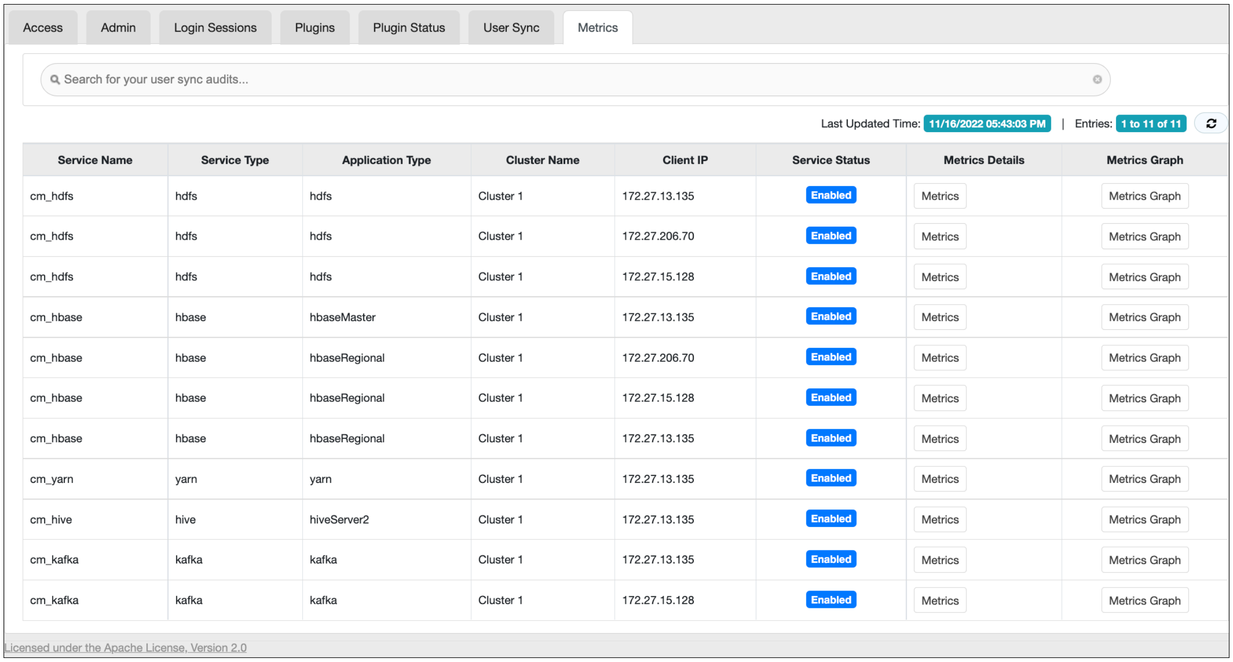
About this task

Metrics provides a high-level view of audit logs as they generate and update in Ranger. Ranger captures audit metrics throughput from the following Ranger services:

- Atlas
- HBase
- Hdfs
- Hive
- Impala
- Kafka
- Knox
- Kudu
- NiFi
- Schema-registry
- Solr
- Streams Messaging Manager
- Yarn

Procedure

1. To view audit metrics, in the Ranger Admin Web UI, click Audits Metrics .

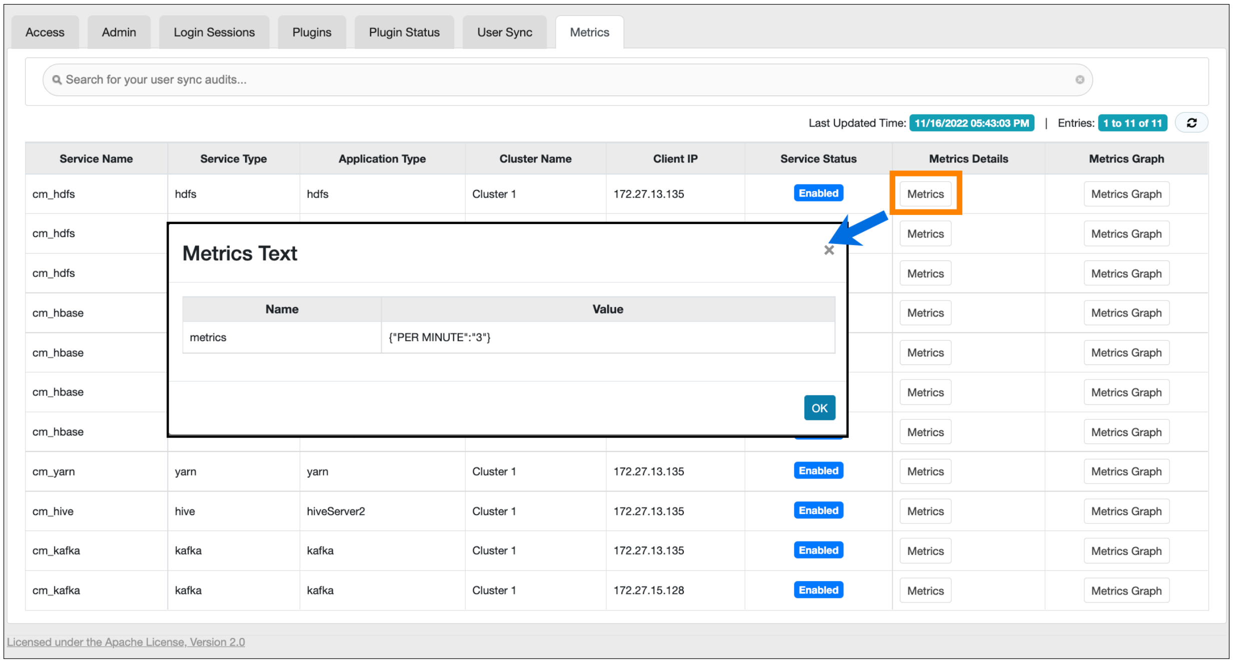


Last Updated Time: 11/16/2022 05:43:03 PM | Entries: 1 to 11 of 11

Service Name	Service Type	Application Type	Cluster Name	Client IP	Service Status	Metrics Details	Metrics Graph
cm_hdfs	hdfs	hdfs	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hdfs	hdfs	hdfs	Cluster 1	172.27.206.70	Enabled	Metrics	Metrics Graph
cm_hdfs	hdfs	hdfs	Cluster 1	172.27.15.128	Enabled	Metrics	Metrics Graph
cm_hbase	hbase	hbaseMaster	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hbase	hbase	hbaseRegional	Cluster 1	172.27.206.70	Enabled	Metrics	Metrics Graph
cm_hbase	hbase	hbaseRegional	Cluster 1	172.27.15.128	Enabled	Metrics	Metrics Graph
cm_hbase	hbase	hbaseRegional	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_yarn	yarn	yarn	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hive	hive	hiveServer2	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_kafka	kafka	kafka	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_kafka	kafka	kafka	Cluster 1	172.27.15.128	Enabled	Metrics	Metrics Graph

Licensed under the Apache License, Version 2.0

2. To view metrics details for a specific service, click Metrics.



Last Updated Time: 11/16/2022 05:43:03 PM | Entries: 1 to 11 of 11

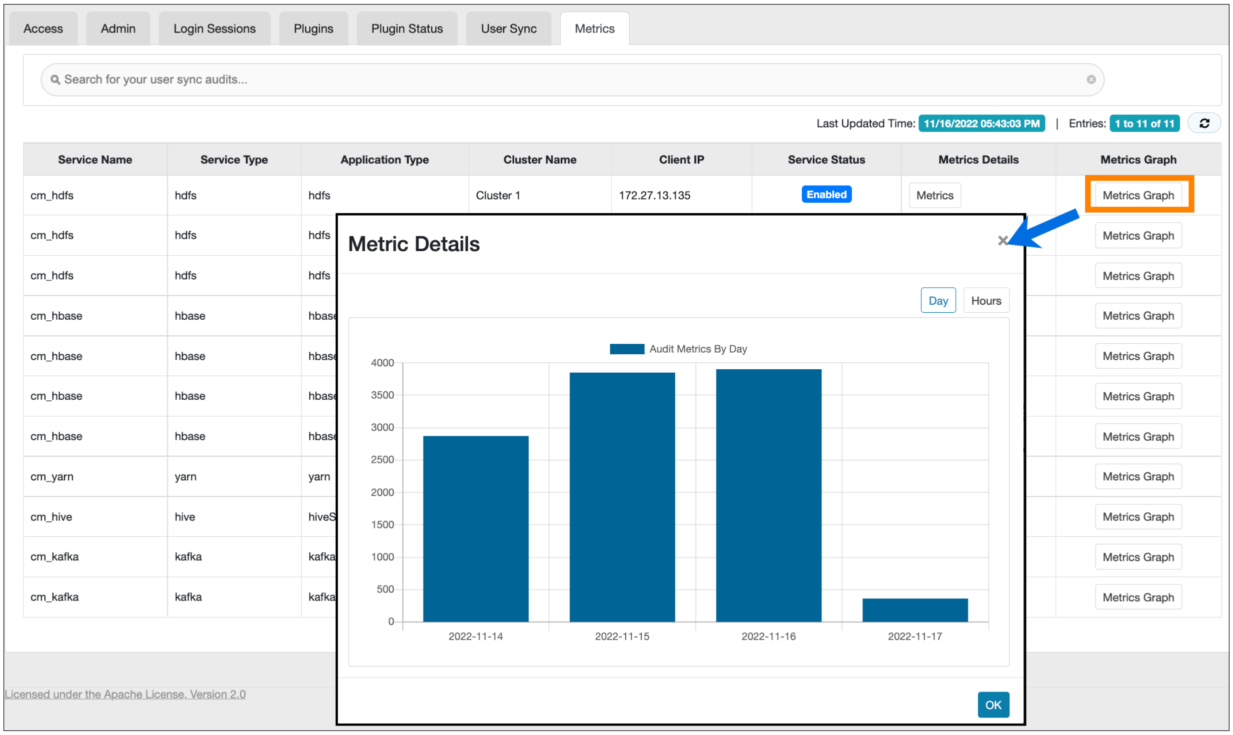
Service Name	Service Type	Application Type	Cluster Name	Client IP	Service Status	Metrics Details	Metrics Graph
cm_hdfs	hdfs	hdfs	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hdfs						Metrics	Metrics Graph
cm_hdfs						Metrics	Metrics Graph
cm_hbase						Metrics	Metrics Graph
cm_hbase						Metrics	Metrics Graph
cm_hbase						Metrics	Metrics Graph
cm_hbase						Metrics	Metrics Graph
cm_yarn	yarn	yarn	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hive	hive	hiveServer2	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_kafka	kafka	kafka	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_kafka	kafka	kafka	Cluster 1	172.27.15.128	Enabled	Metrics	Metrics Graph

Metrics Text

Name	Value
metrics	{"PER MINUTE":"3"}

Licensed under the Apache License, Version 2.0

3. To view hourly or daily metrics as a graphic for a specific service, click Metrics Graph.



Last Updated Time: 11/16/2022 05:43:03 PM | Entries: 1 to 11 of 11

Service Name	Service Type	Application Type	Cluster Name	Client IP	Service Status	Metrics Details	Metrics Graph
cm_hdfs	hdfs	hdfs	Cluster 1	172.27.13.135	Enabled	Metrics	Metrics Graph
cm_hdfs	hdfs	hdfs					Metrics Graph
cm_hdfs	hdfs	hdfs					Metrics Graph
cm_hbase	hbase	hbas					Metrics Graph
cm_hbase	hbase	hbas					Metrics Graph
cm_hbase	hbase	hbas					Metrics Graph
cm_hbase	hbase	hbas					Metrics Graph
cm_yarn	yarn	yarn					Metrics Graph
cm_hive	hive	hiveS					Metrics Graph
cm_kafka	kafka	kafka					Metrics Graph
cm_kafka	kafka	kafka					Metrics Graph

Metric Details

Licensed under the Apache License, Version 2.0

Creating a read-only Admin user (Auditor)

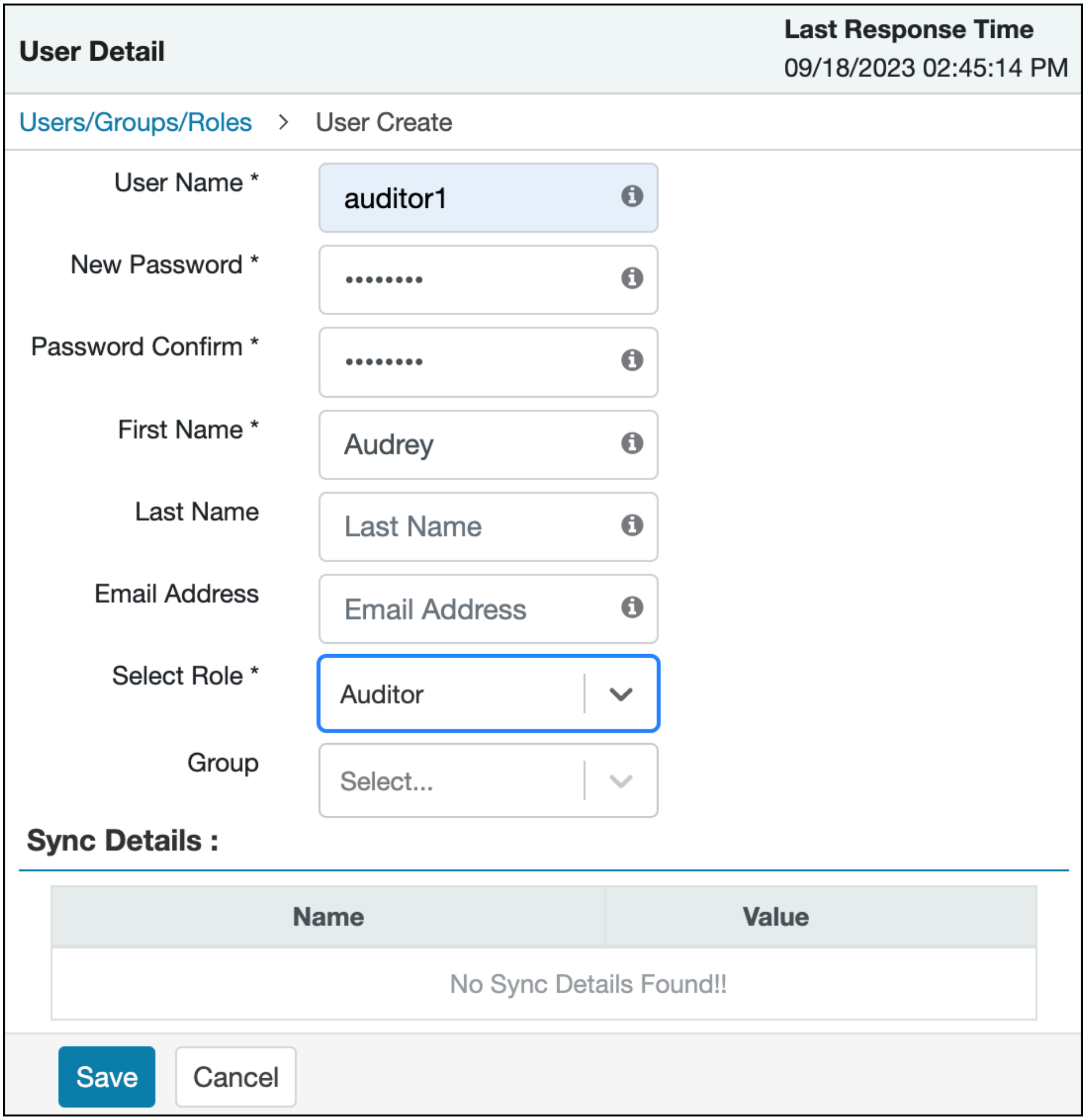
Creating a read-only Admin user (Auditor) enables compliance activities because this user can monitor policies and audit events, but cannot make changes.

About this task

When a user with the Auditor role logs in, they see a read-only view of Ranger policies and audit events. An Auditor can search and filter on access audit events, and access and view all tabs under Audit to understand access events. They cannot edit users or groups, export/import policies, or make changes of any kind.

Procedure

1. In Ranger Admin Web UI, select Settings > Users.
2. Click Add New User.
3. Complete the **User Detail** section, selecting Auditor as the role:



User Detail

Last Response Time
09/18/2023 02:45:14 PM

Users/Groups/Roles > User Create

User Name *	auditor1
New Password *	••••••••
Password Confirm *	••••••••
First Name *	Audrey
Last Name	Last Name
Email Address	Email Address
Select Role *	Auditor
Group	Select...

Sync Details :

Name	Value
No Sync Details Found!!	

4. Click Save.

Configuring Ranger audit properties for Solr

How to change the default time settings that control how long Ranger keeps audit data collected by Solr.

About this task

The Solr audit destination is intended to store short term audit records. You can configure parameters that control how much data collected by Solr that Ranger will store for auditing purposes.

Table 1: Ranger Audit Configuration Parameters for Solr

Parameter Name	Description	Default Setting	Units
ranger.audit.solr.config.ttl	Time To Live for Solr Collection of Ranger Audits **Note:** "Time To Live for Solr Collection of Ranger Audits" is also known as the Max Retention Days attribute.	90	days
ranger.audit.solr.config.delete.trigger	Auto Delete Period in seconds for Solr Collection of Ranger Audits for expired documents	1	days (configurable)

Procedure

1. From Cloudera Manager choose Ranger Configuration .
2. In Search, type ranger.audit.solr.config, then press Return.
3. In ranger.audit.solr.config.ttl, set the the number of days to keep audit data.
4. In ranger.audit.solr.config.delete.trigger set the number and units (days, minutes, hours, or seconds) to keep data for expired documents.
5. Refresh the configuration:
   a) Click Refresh Configuration, as prompted.
   b) In Actions, click Update Solr config-set for Ranger, then confirm.

Limiting solr spool directory growth

To limit stored audit logs, you may set a maximum limit on the solr spool directory size for each service.

Procedure

1. Manually delete the logs under the archive path for the service.
2. Set the log retention value of the archive path from default 100 to 2 .
   a) In Cloudera Manager <service_name> Configuration <service_name> Service Advanced Configuration Snippet (Safety Valve) for <service_name>.xml
   b) Click +
   c) Add the following parameter:

      xasecure.audit.destination.solr.batch.filespool.archive.max.files

      2
3. Restart the service.

Configuring Ranger audit properties for HDFS

How to change the settings that control how Ranger writes audit records to HDFS.

About this task

The HDFS audit destination is intended to store long-term audit records.

You can configure whether Ranger stores audit records in HDFS and at which location.

You must purge long term audit records stored in HDFS manually.

Table 2: Ranger Audit Configuration Parameters for HDFS

Parameter Name	Description	Default Setting	Units
ranger_plugin_hdfs_audit_enabled	controls whether Ranger writes audit records to HDFS	true	T/F
ranger_plugin_hdfs_audit_url	location at which you can access audit records written to HDFS	<hdfs.host_name>/ ranger/audit	string

Note: You can also disable storing ranger audit data to hdfs in each service specifically by setting xasecure.audit.destination.hdfs=false in that service.

Procedure

1. From Cloudera Manager choose Ranger Configuration .
2. In Search, type ranger_plugin, then press Return.
3. In ranger_plugin_hdfs_audit_enabled, check/uncheck RANGER-1 (Service Wide)
4. In ranger_plugin_hdfs_audit_url type a valid directory on the hdfs host.
5. Refresh the configuration, using one of the following two options:
   a) Click Refresh Configuration, as prompted or, if Refresh Configuration does not appear,
   b) In Actions, click Update Solr config-set for Ranger, then confirm.

What to do next

(Optional)

You may want to delete older logs from HDFS. Cloudera provides no feature to do this. You may accomplish this task manually, using a script.

Note:

The following example script is not supported by Cloudera. It is shown for reference only. You must test this successfully in a test environment before implementing it in a production cluster.

You must specify the audit log directory by replacing the 2nd line hdfs dfs -ls /<path_to>/<audit_logs> in the example script.

You may also include the -skipTrash option, if you choose, on 7th line in the script.

```
##################
today=`date +'%s'`
hdfs dfs -ls /<path_to>/<audit_logs> | grep "^d" | while read line ; do
dir_date=$(echo ${line} | awk '{print $6}')
difference=$(( ( ${today} - $(date -d ${dir_date} +%s) ) / ( 24*60*60 ) ))
filePath=$(echo ${line} | awk '{print $8}')

if [ ${difference} -gt 30 ]; then
    hdfs dfs -rm -r $filePath
fi
done
###################
```

Related Information

How to do a cleanup of hdfs files older than a certain date using a bash script

Triggering HDFS audit files rollover

How to configure when HDFS audit files close for each service.

About this task

By default, the Ranger Audit framework closes audit files created in HDFS or other cloud storage inline with audit event triggers. In other words, when an audit event occurs, Ranger checks the configured rollout time and then closes the file if the threshold has reached. Default audit rollout time is 24 hours. If no audit event occurs in a 24 hour period, files remain open beyond the 24 hour period. In some environments, audit log analysis that encounter an audit file open beyond the current date can cause system exceptions. If you want the files to be closed every day, so that the audit log file will have only that day's log and the next day’s log will be in the next day's file, you can configure the audit framework to close files every day. To do this, you must add several configuration parameters to the ranger-<service_name>-audit.xml (safety valve) file for each service, using Cloudera Manager.

Procedure

1. From Cloudera Manager choose <service_name> Configuration .
2. In <service_name> Configuration Search , type ranger-<service_name>, then press Return.
3. In <service_name> Server Advanced Configuration Snippet (Safety Valve) for ranger-<service_name>-audit.xml, do the following steps:
   a) Click + (Add).
   b) In Name, type xasecure.audit.destination.hdfs.file.rollover.enable.periodic.rollover
   c) In Value, type true.

      When this is enabled Ranger Audit Framework will spawn a Scheduler thread which monitors the occurrence of closing threshold and closes the file. By default every night the file gets closed.
   d) Click + (Add another).
   e) In Name, type xasecure.audit.destination.hdfs.file.rollover.sec
   f) In Value, type an integer value in seconds.

      This is the time in seconds when the file has to be closed. The default value is 86400 sec (1 day) which triggers the file to be closed at midnight and opens a new audit log for the next day. You can override the default value can be overridden by setting this parameter. For example, if you set the value 3600 (1 hr), the file gets closed every hour.

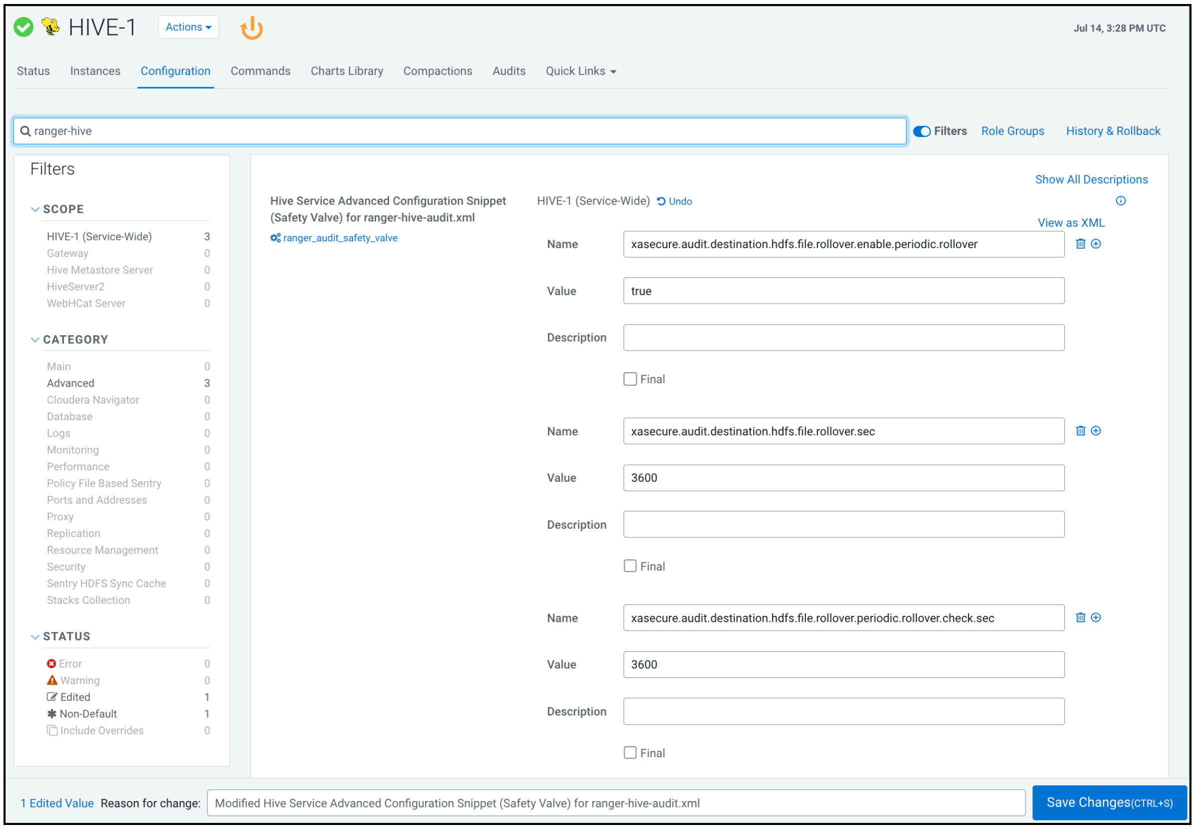
g) Click + (Add another).

h) In Name, type xasecure.audit.destination.hdfs.file.rollover.periodic.rollover.check.sec

i) In Value, type an integer value in seconds.

This is the time frequency of the check to be done whether the threshold time for rollover has occurred. By default the check is done every 60 secs. You can configure this parameter to delay the check time.

Figure 2: Example: Hive service configured to trigger rollover of hdfs audit files



ranger-hive

Hive Service Advanced Configuration Snippet (Safety Valve) for ranger-hive-audit.xml

HIVE-1 (Service-Wide)

Name: xasecure.audit.destination.hdfs.file.rollover.enable.periodic.rollover

Value: true

Description:

□ Final

Name: xasecure.audit.destination.hdfs.file.rollover.sec

Value: 3600

Description:

□ Final

Name: xasecure.audit.destination.hdfs.file.rollover.periodic.rollover.check.sec

Value: 3600

Description:

□ Final

1 Edited Value Reason for change: Modified Hive Service Advanced Configuration Snippet (Safety Valve) for ranger-hive-audit.xml

j) Click Save Changes (CTRL+S).

4. Repeat steps 1-3 for each service.

5. Restart the service.

Configuring Ranger audit log storage to a local file

How to configure a service to write Ranger audit log records to a local file, in addition to the default Solr and HDFS locations.

About this task

Note: This feature works only with the Hive plugin. It is not certified to work with all plugins of Ranger.

The Solr and HDFS audit destinations are intended to store short- and long-term audit records respectively, by default. You can configure whether a service also stores Ranger audit records in a local file destination. You may also further configure a service by appending additional, service-specific configuration properties. To do this, first modify ranger-<service_name>-audit.xml file, using Cloudera Manager. Then, add additional appenders to the configuration. This topic shows how to add appenders to the Hive service, for example.

Table 3: Ranger Audit Destination Configuration Parameters for services

Parameter Name	Description	Default Setting	Units
xasecure.audit.destination.log4j	controls whether Ranger writes audit records to a service	true	T/F
xasecure.audit.destination.log4j.logger	location at which you can access audit records written to HDFS	rangerauditlogger	string

Note: You can also disable storing ranger audit data to hdfs in each service specifically by setting xasecure.audit.destination.hdfs=false in that service.

Procedure

1. Configure Ranger Log4JAuditDestination.
   a) From Cloudera Manager choose <service_name> Configuration .
   b) In Search, type ranger-<service_name>-audit, then press Return.
   c) In Hive Service Advanced Configuration Snippet (Safety Valve) for ranger-hive-audit.xml, click +.
   d) Add the following property/value:

      Name

      xasecure.audit.destination.log4j

      Value

      true
   e) Click Save Changes (CTRL+S).
   f) In Hive Service Advanced Configuration Snippet (Safety Valve) for ranger-hive-audit.xml, click +.
   g) Add the following property/value

      Name

      xasecure.audit.destination.log4j.logger

      Value

      rangerauditlogger
   h) Click Save Changes (CTRL+S).
2. Configure HIVE log4 for ranger audit log appender.

   Add the following properties to the configuration: (?? HiveServer2 Logging Advanced Configuration Snippet (Safety Valve) ??)

```
appenders=console, DRFA, redactorForRootLogger, RANGERAUDIT
appenders=console, DRFA, redactorForRootLogger, RANGERAUDIT
appender.RANGERAUDIT.type=RollingRandomAccessFile
appender.RANGERAUDIT.name=RANGERAUDIT
appender.RANGERAUDIT.fileName=${log.dir}/hive-ranger-audit.log
appender.RANGERAUDIT.filePattern=${log.dir}/hive-ranger-audit.log.%d{yyyy-
MM-dd}-%i
appender.RANGERAUDIT.filePermissions=rwxr--r--
appender.RANGERAUDIT.layout.type=PatternLayout
appender.RANGERAUDIT.layout.pattern=%d{ISO8601} %q %5p [%t] %c{2} (%F:%
M(%L)) - %m%n
appender.RANGERAUDIT.policies.type=Policies
appender.RANGERAUDIT.policies.size.type=SizeBasedTriggeringPolicy
appender.RANGERAUDIT.policies.size.size=250MB
appender.RANGERAUDIT.strategy.type=DefaultRolloverStrategy
appender.RANGERAUDIT.strategy.max=10
appender.RANGERAUDIT.strategy.action.type=DELETE
appender.RANGERAUDIT.strategy.action.basepath=${log.dir}
appender.RANGERAUDIT.strategy.action.maxdepth=1
appender.RANGERAUDIT.strategy.action.PathConditions.glob=hive-ranger-aud
it.*
appender.RANGERAUDIT.strategy.action.PathConditions.type=IfFileName
appender.RANGERAUDIT.strategy.action.PathConditions.nestedConditions.ty
pe=IfAccumulatedFileCount
appender.RANGERAUDIT.strategy.action.PathConditions.nestedConditions.excee
ds=2GB
loggers=Ranger
logger.Ranger.name=rangerauditlogger
logger.Ranger.level=INFO
logger.Ranger.appenderRefs=RANGERAUDIT
logger.Ranger.appenderRef.RANGERAUDIT.ref=RANGERAUDIT
```

3. Restart the service to refresh the configuration.

   The result will be hive-ranger-audit.log getting created in the configured folder. Ranger audit logs for hive operations will be appended.

   Note:

   Change the max number of files to retain the files in the folder. Scripting has to be done to periodically move the hive-ranger-audit.log.* to the processing directory if this feature is used to send these audit logs to other destinations via external applications.

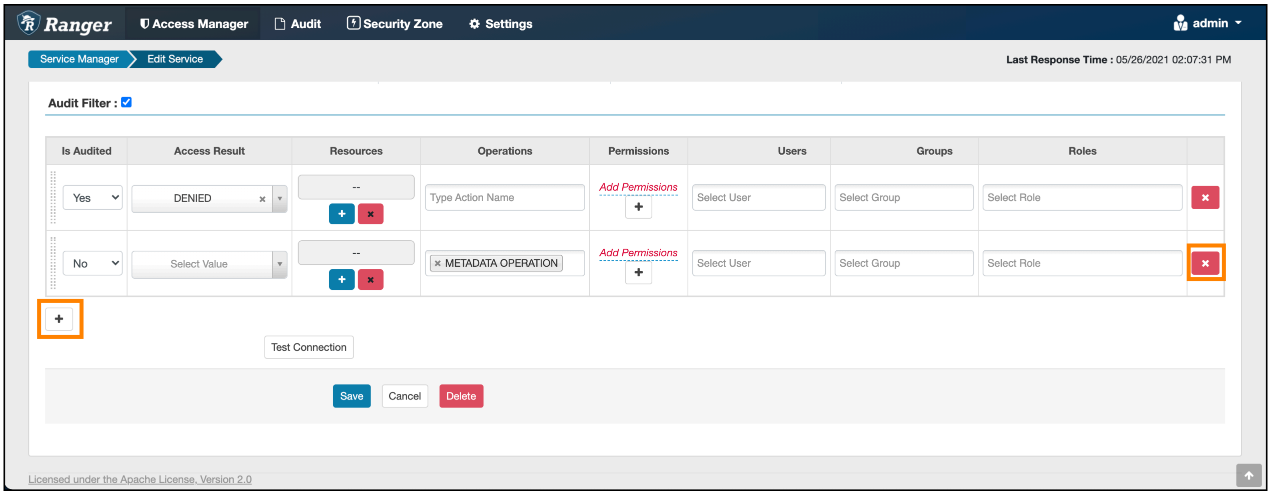
Ranger Audit Filters

You can use Ranger audit filters to control the amount of audit log data collected and stored on your cluster.

About Ranger audit filters

Ranger audit filters allow you to control the amount of audit log data for each Ranger service. Audit filters are defined using a JSON string that is added to each service configuration. The audit filter JSON string is a simplified form of the Ranger policy JSON. Audit filters appear as rows in the Audit Filter section of the Edit Service view for each service. The set of audit filter rows defines the audit log policy for the service. For example, the default audit log policy for the Hadoop SQL service appears in Ranger Admin web UI Service Manager Edit Service when you scroll down to Audit Filter. Audit Filter is checked (enabled) by default. In this example, the top row defines an audit filter that causes all instances of "access denied" to appear in audit logs. The lower row defines a filter that causes no metadata operations to appear in audit logs. These two filters comprise the default audit filter policy for Hadoop SQL service.

Figure 3: Default audit filter policy for the Hadoop SQL service



Audit Filter : ☑

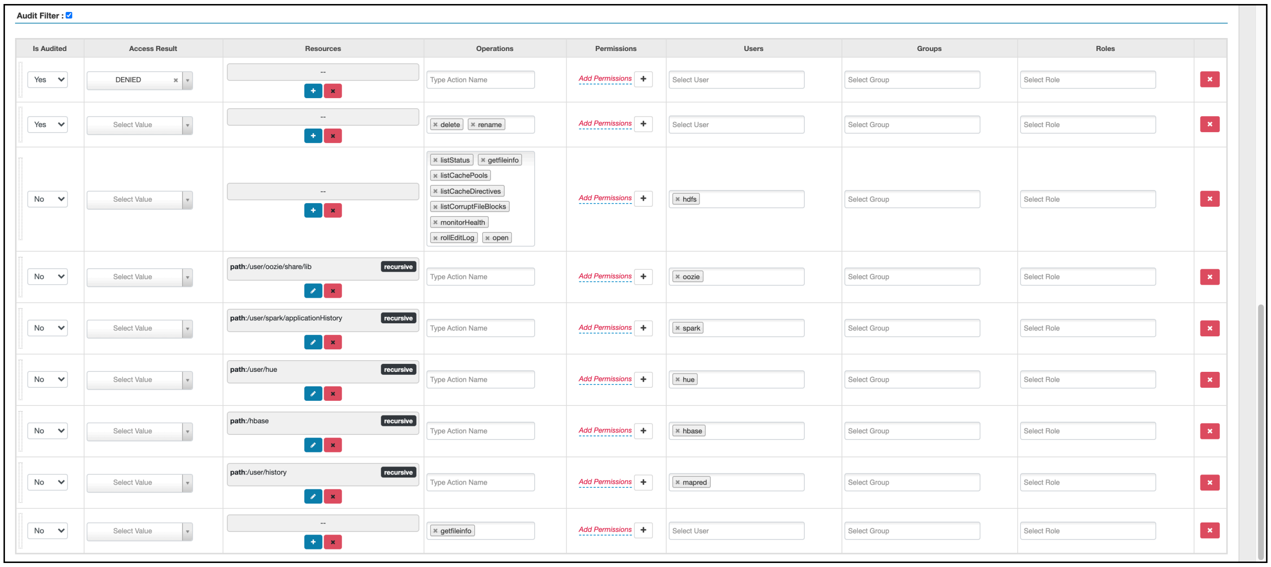
Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	METADATA OPERATION	Add Permissions	Select User	Select Group	Select Role

Default Ranger audit filters

Default audit filters for the following Ranger service appear in Edit Services and may be modified as necessary by Ranger Admin users.

HDFS

Figure 4: Default audit filters for HDFS service

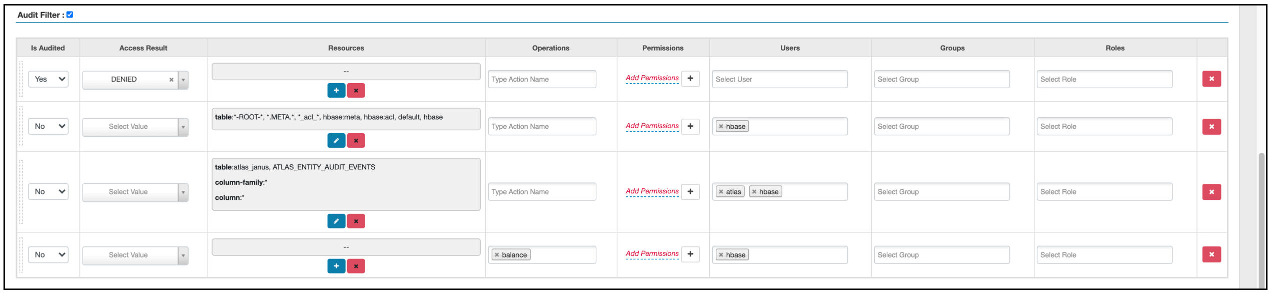


Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
Yes	Select Value	--	delete rename	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	listStatus getfileinfo listCachePools listCacheDirectives listCorruptFileBlocks monitorHealth rollEditLog open	Add Permissions	hdfs	Select Group	Select Role
No	Select Value	**path**:/user/oozie/share/lib recursive	Type Action Name	Add Permissions	oozie	Select Group	Select Role
No	Select Value	**path**:/user/spark/applicationHistory recursive	Type Action Name	Add Permissions	spark	Select Group	Select Role
No	Select Value	**path**:/user/hue recursive	Type Action Name	Add Permissions	hue	Select Group	Select Role
No	Select Value	**path**:/hbase recursive	Type Action Name	Add Permissions	hbase	Select Group	Select Role
No	Select Value	**path**:/user/history recursive	Type Action Name	Add Permissions	mapred	Select Group	Select Role
No	Select Value	--	getfileinfo	Add Permissions	Select User	Select Group	Select Role

Hbase

Figure 5: Default audit filters for the Hbase service

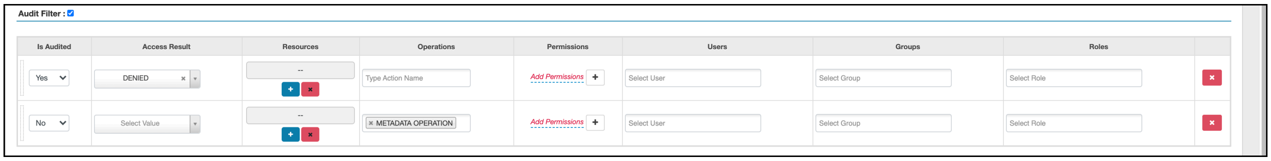


Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	**table**:*-ROOT-*, *.META.*, *_acl_*, hbase:meta, hbase:acl, default, hbase	Type Action Name	Add Permissions	hbase	Select Group	Select Role
No	Select Value	**table**:atlas_janus, ATLAS_ENTITY_AUDIT_EVENTS **column-family**:* **column**:*	Type Action Name	Add Permissions	atlas hbase	Select Group	Select Role
No	Select Value	--	balance	Add Permissions	hbase	Select Group	Select Role

Hadoop SQL

Figure 6: Default audit filters for the Hadoop SQL service

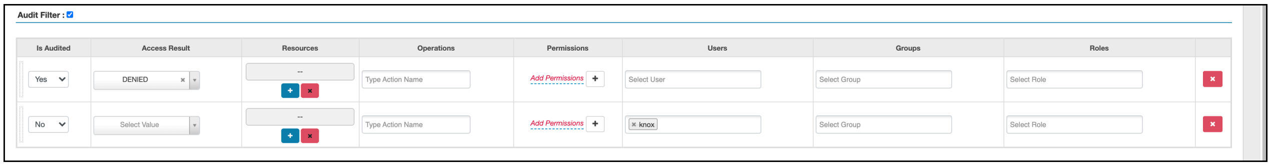


Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	METADATA OPERATION	Add Permissions	Select User	Select Group	Select Role

Knox

Figure 7: Default audit filters for the Knox service

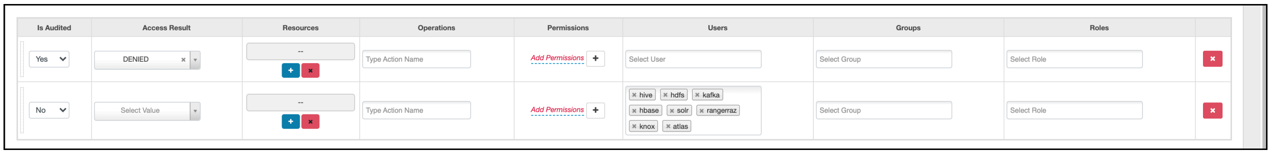


Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	Type Action Name	Add Permissions	knox	Select Group	Select Role

Solr

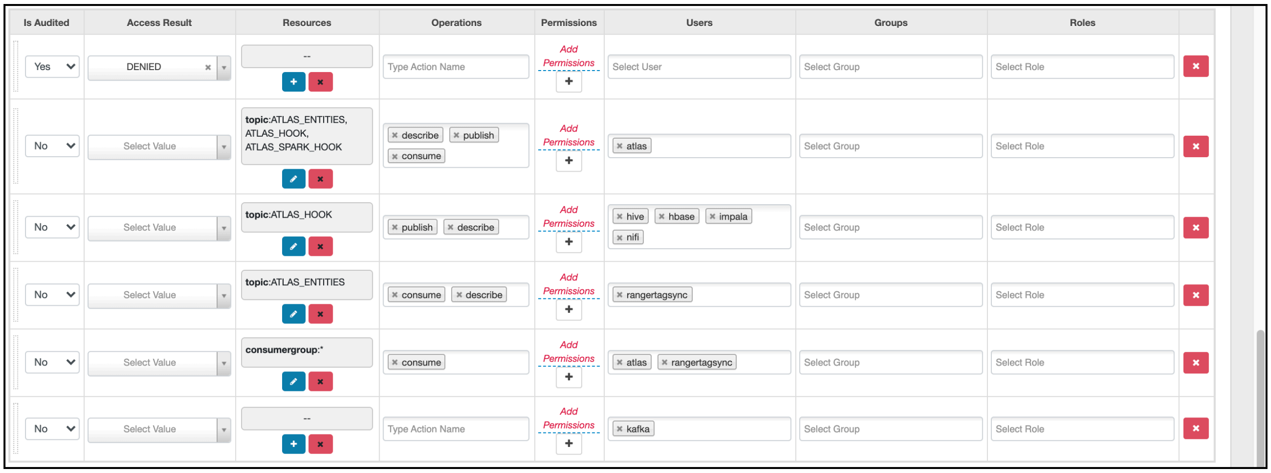
Figure 8: Default audit filters for the Solr service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	Type Action Name	Add Permissions	hive, hdfs, kafka, hbase, solr, rangerraz, knox, atlas	Select Group	Select Role

Kafka

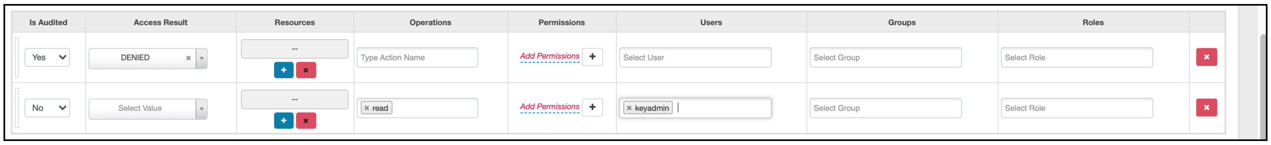
Figure 9: Default audit filters for the Kafka service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	topic:ATLAS_ENTITIES, ATLAS_HOOK, ATLAS_SPARK_HOOK	describe, publish, consume	Add Permissions	atlas	Select Group	Select Role
No	Select Value	topic:ATLAS_HOOK	publish, describe	Add Permissions	hive, hbase, impala, nifi	Select Group	Select Role
No	Select Value	topic:ATLAS_ENTITIES	consume, describe	Add Permissions	rangertagsync	Select Group	Select Role
No	Select Value	consumergroup:*	consume	Add Permissions	atlas, rangertagsync	Select Group	Select Role
No	Select Value	--	Type Action Name	Add Permissions	kafka	Select Group	Select Role

Ranger KMS

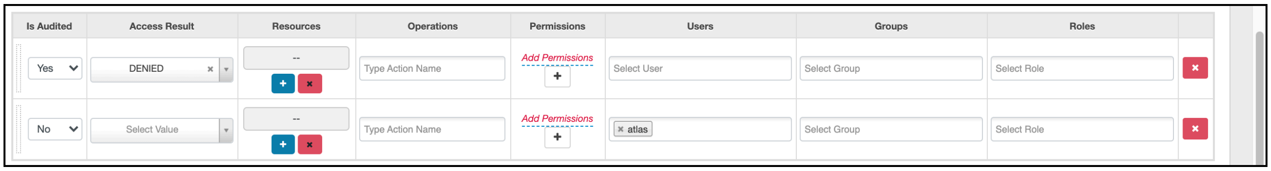
Figure 10: Default audit filters for the Ranger KMS service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	read	Add Permissions	keyadmin	Select Group	Select Role

Atlas

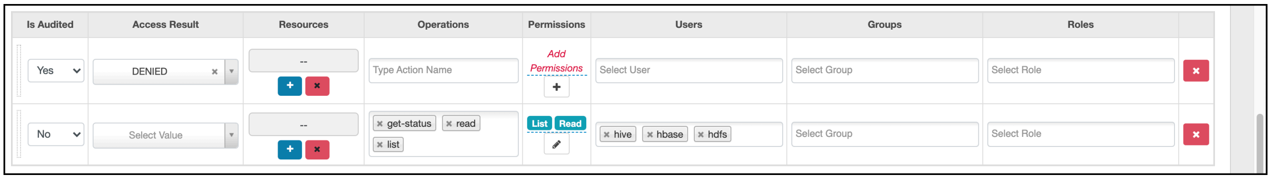
Figure 11: Default audit filters for the Atlas service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role
No	Select Value	--	Type Action Name	Add Permissions	atlas	Select Group	Select Role

ADLS

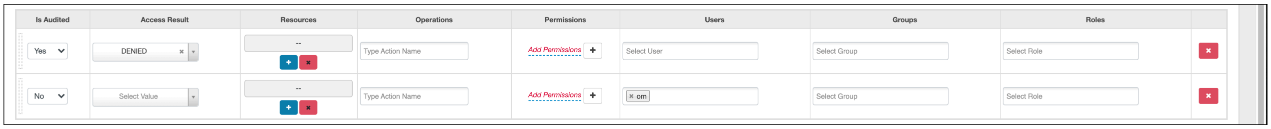
Figure 12: Default audit filters for the ADLS service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role	×
No	Select Value	--	get-status read list	List Read	hive hbase hdfs	Select Group	Select Role	×

Ozone

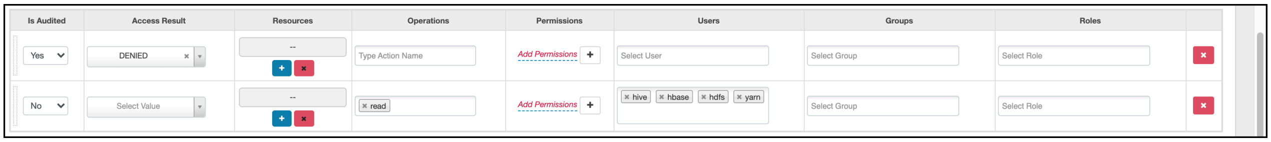
Figure 13: Default audit filters for the Ozone service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role	×
No	Select Value	--	Type Action Name	Add Permissions	om	Select Group	Select Role	×

S3

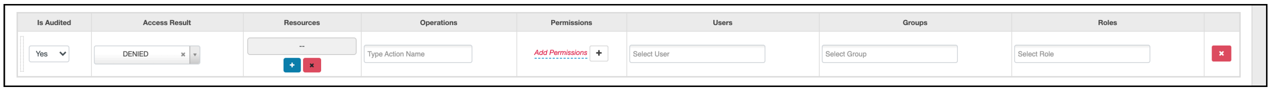
Figure 14: Default audit filters for the S3 service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role	×
No	Select Value	--	read	Add Permissions	hive hbase hdfs yarn	Select Group	Select Role	×

Tag-based services

Figure 15: Default audit filters for a tag-based service



Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	DENIED	--	Type Action Name	Add Permissions	Select User	Select Group	Select Role	×

Note:

Default audit filter policies do not exist for Yarn, NiFi, NiFi Registry, Kudu, or schema registry services.

Configuring audit filters in Ranger Admin Web UI

You can configure audit filter conditions for each service, using the Create/Edit Service option.

About this task

You can stop certain audit logs from being sent to destinations like SOLR and HDFS by setting up audit filters for a service using the Ranger Admin Web UI.

Procedure

1. In the Ranger Admin Web UI Service Manager , click Add New Service or Edit (existing service).

2. On Create/Edit Service, scroll down to Audit Filters.

   a) Verify that the Audit Filter option is checked.

      Optionally, define any of the following to include in the filter definition:

      Is Audited

      To store or not store a filter in the audit logs for a service. Values are:

      - Yes: Stores audit records in the defined audit destination.
      - No: Does not store audit records.

      Access Results

      To include that access result in the audit log filter. Values are:

      - Denied
      - Allowed
      - Not Determined

      Resource

      To include or exclude specific resources, such as databases, tables, or columns, from the audit log filter.

      Operations

      To include the action or operation in the audit log filter. Select specific operations to filter.

      Note: For Operations field, you can refer to the Ranger access audit page and find out the available access types.

      Permissions

      Add or remove permissions:

      1. Click + in Permissions to open the Add dialog.
      2. Select/Unselect required permissions.

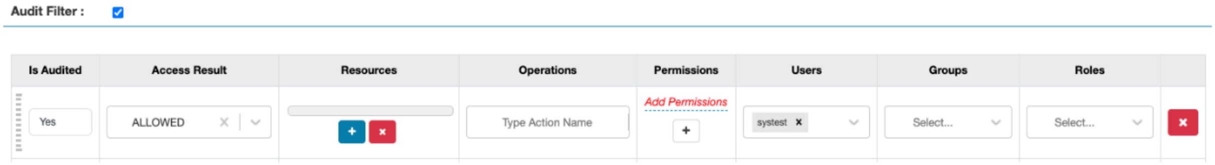
      Users, Groups, Roles

      To include one or multiple users/groups/roles in the audit log filter.

   b) Click Save.

Example

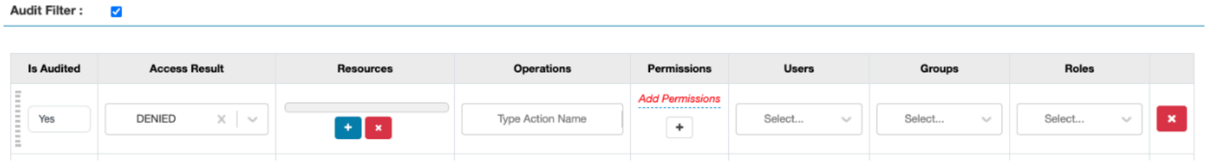
- For a service, the following audit filter is used to capture all ALLOWED audits for user systest:



Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	ALLOWED	+ ×	Type Action Name	Add Permissions +	systest ×	Select...	Select...	×

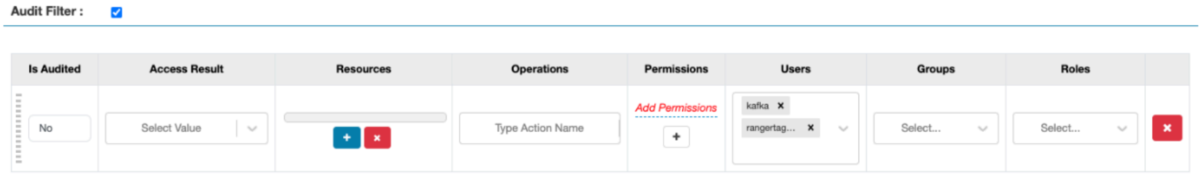
- The following filter specifies all DENIED audits to be audited:



Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
Yes	DENIED	+ ×	Type Action Name	Add Permissions +	Select...	Select...	Select...	×

- The following is an example where all operations for users kafka and rangertagsync are NOT audited, configured for a Kafka service:



Audit Filter : ☑

Is Audited	Access Result	Resources	Operations	Permissions	Users	Groups	Roles	
No	Select Value	+ ×	Type Action Name	Add Permissions +	kafka × rangertag... ×	Select...	Select...	×

Audit filter evaluation order

Audit Filters, implemented as Ranger Policies, determine whether to audit an authorization request after it is processed by Tag/Resource-based Ranger policies. After an Audit Filter condition is fulfilled and executed for a request, no other audit filters are evaluated. If the evaluation dictates that the authorization request should be audited, an audit record is then added to the designated audit destination(s).

Audit filters with REST APIs

Audit filters can be configured through REST when creating or updating a service as well. The following are examples where you can use JSON to configure audit filters:

- **Audit all denied accesses**

  The following audit filter specifies that any access with accessResult = DENIED will be audited:

```
{
  "accessResult": "DENIED",
  "isAudited": true
}
```

- **Do not audit access to path /unaudited**

  The following audit filter specifies that access to the resource path /unaudited (recursively) will not be audited:

```
{
  "resources": {
    "path": {
      "values": ["/unaudited"],
      "isRecursive": true
    }
  },
  "isAudited": false
}
```

- **Do not audit access to database sys, table dump, by user user2**

  The following audit filter specifies that access to resource database=> sys table=> dump by user “user2” will not be audited:

```
{
  "resources": {
    "database": {
      "values": ["sys"]
    },
    "table": {
      "values": ["dump"]
    }
  },
  "users": ["user2"],
  "isAudited": false
}
```

- **Do not audit actions listStatus and getfileinfo with access type execute**

  The following audit filter specifies that access result in actions => listStatus, getfileInfo and accessType => execute will not be audited:

  ```
  {
    "actions": ["listStatus", "getfileinfo"],
    "accessTypes": ["execute"],
    "isAudited": false
  }
  ```
- **Do not audit access by user superuser1 and group supergroup1**

  The following audit filter specifies that access by user "superuser1" and group "supergroup1" will not be audited:

  ```
  {
    "users": ["superuser1"],
    "groups": ["supergroup1"],
    "isAudited": false
  }
  ```
- **Do not audit access to any resource tagged as NO_AUDIT**

  The following audit filter specifies that access to any resource tagged as NO_AUDIT will not be audited:

  ```
  {
    "resources": {
      "tag": {
        "values": ["NO_AUDIT"]
      }
    },
    "isAudited": false
  }
  ```

Above are examples of individual filters. You can apply them together in JSON when using REST to create/update a service as well.

Filter service access logs from Ranger UI

You can limit display of system access/audit log records generated by service users in each service.

About this task

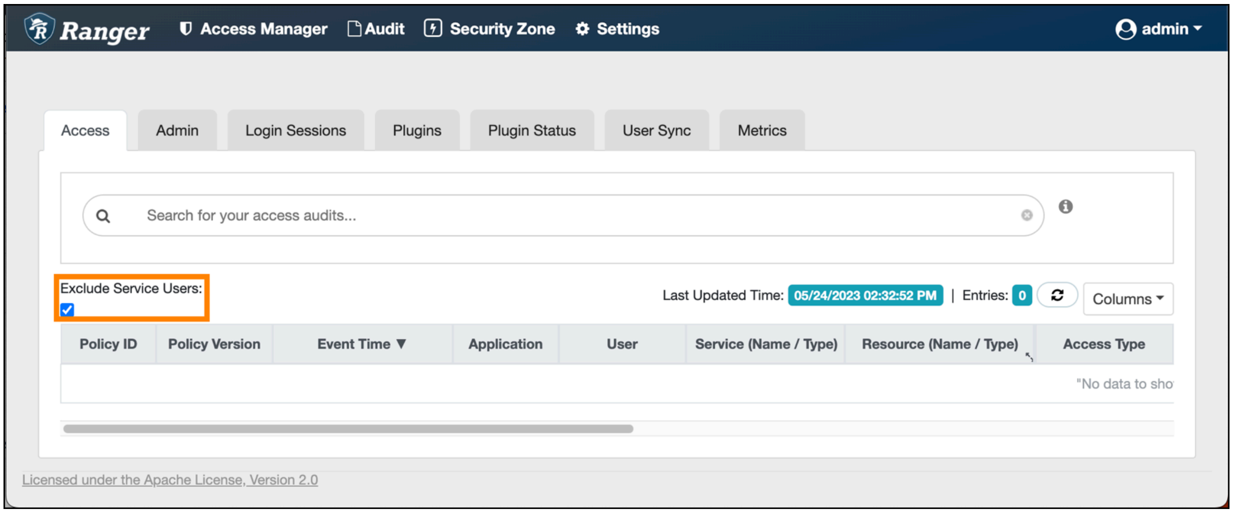
This topic describes how to limit the display of access log records on the Access tab in the Ranger Admin Web UI.

Procedure

1. Go to Ranger Admin Web UI Audit Access .

2. Check the Exclude Service Users box, as shown in:

Figure 16: Setting the Exclude Service Users flag to true



Exclude Service Users: ☑

Last Updated Time: 05/24/2023 02:32:52 PM | Entries: 0

Policy ID	Policy Version	Event Time ▼	Application	User	Service (Name / Type)	Resource (Name / Type)	Access Type
							"No data to sho

3. Define specific component services and users for access logs to filter out, in ranger-admin-site.xml.

a) Go to Cloudera Manager Ranger Configuration

b) In Search, type ranger-admin-site.

c) Define the following properties:

Name

ranger.plugins.<service_name>.serviceuser

Value

<service_name>

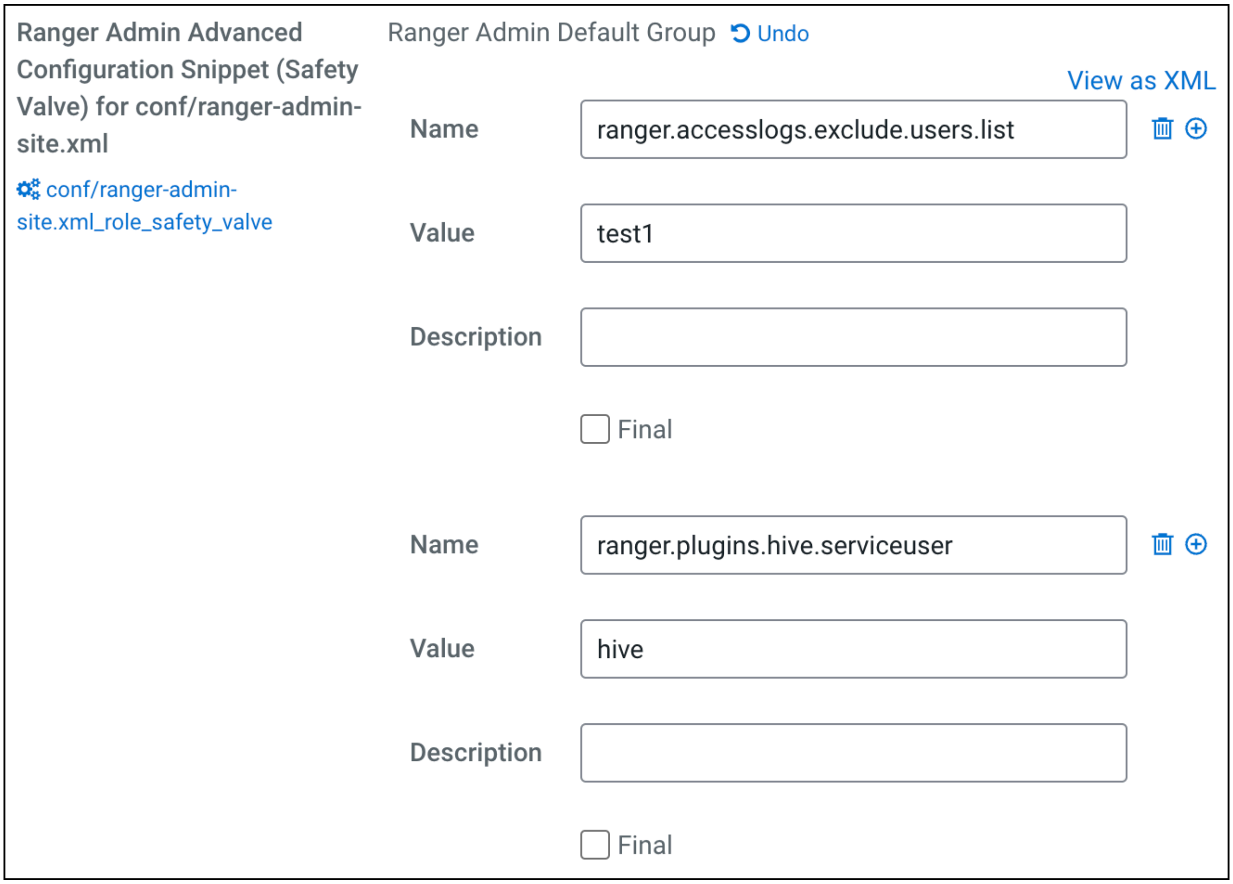
Name

ranger.accesslogs.exclude.users.list

Value

user1, user2

Figure 17: Filtering out service and user logs for Hive service



Ranger Admin Advanced Configuration Snippet (Safety Valve) for conf/ranger-admin-site.xml

conf/ranger-admin-site.xml_role_safety_valve

Ranger Admin Default Group ⟲ Undo

View as XML

Name: ranger.accesslogs.exclude.users.list

Value: test1

Description:

□ Final

Name: ranger.plugins.hive.serviceuser

Value: hive

Description:

□ Final

4. Click Save Changes (CTRL+S).

5. Restart the Ranger service.

Results

Setting Exclude Service Users to true and defining specific filtering properties prevents audit logs from service users from appearing on Ranger Admin Web UI Audit Access , but does NOT prevent access logs for service users from being generated in Solr.

Configuring audit spool alert notifications

You can enable and configure alerts for Ranger plugin supported services on Cloudera Manager that notify when audit spool files are accumulated.

About this task

Ranger stores plugin access audit events (audit logs) in Solr and in HDFS. Typically, you store audit logs in Solr for short-term auditing purposes and in HDFS for longer-term purposes. When the Solr server is down, Ranger plugin audit logs are stored in the spool directory as spool files. You configure the spool directory location using the following properties:

For Solr

xasecure.audit.destination.solr.batch.filespool.dir = /var/log/<service_name>/audit/solr/spool

For HDFS

xasecure.audit.destination.hdfs.batch.filespool.dir = /var/log/<service_name>/audit/hdfs/spool

If the Solr server goes down for a long period of time or, if a large number of audit events occur while the Solr server is down; then spool files accumulate in the spool directory. Spool file accumulation consumes system memory. Sometimes, audit records in spool files become corrupted, and may not be restored when the Solr server returns to a running state. Corrupted, un-restored records also cause spool file accumulation. This requires manual cleanup of corrupted spool files. An unnoticed large accumulation or "piling up" of spool files may fill the local filesystem and result in service failure.

After you enable spool directory metric usage for a service, an alert appears on the Cloudera Manager UI which notifies the user when spool files have piled up in the spool directory. The Cloudera Manager agent measures the disk usage of the spool directory and registers it as a metric value. This metric value is compared against a threshold value. The spool alert appears on the Cloudera Manager UI if the metric value is greater than the threshold value.

Single-level metrics for collecting the disk usage of Solr and HDFS Ranger plugin spool directories are registered in Cloudera Manager, using the following disk usage metric names:

Solr

ranger_plugin_solr_spool_directory_size

HDFS

ranger_plugin_hdfs_spool_directory_size

The following table lists Ranger plugin supported service names and roles that support spool alerts.

Table 4: Ranger plugin supported services and their roles supporting audit spool alerts

Services	Roles
HDFS	NAMENODE
HIVE	HIVEMETASTORE
HIVE_ON_TEZ	HIVESERVER2
HBASE	MASTER, REGIONSERVER
YARN	RESOURCEMANAGER
IMPALA	IMPALAD, CATALOGSERVER
ATLAS	ATLAS_SERVER
KAFKA	KAFKA_BROKER
KNOX	KNOX_GATEWAY
KUDU	KUDU_MASTER

Services	Roles
RANGER_KMS	RANGER_KMS_SERVER
RANGER_RAZ	RANGER_RAZ_SERVER
SCHEMAREGISTRY	SCHEMA_REGISTRY_SERVER
STREAMS_MESSAGING_MANAGER	STREAMS_MESSAGING_MANAGER_SERVER

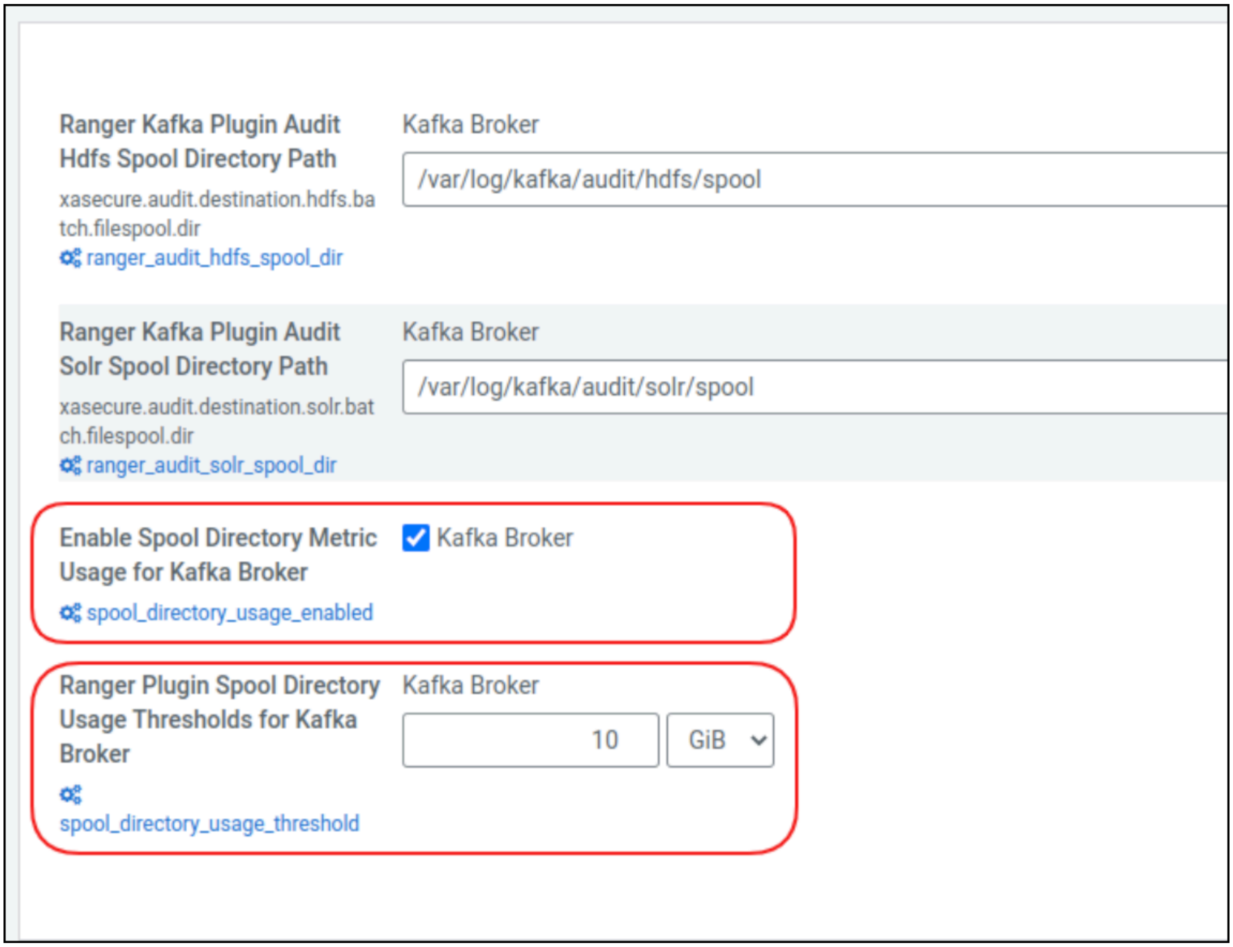
To enable or disable spool directory alerts:

Procedure

1. Go to Cloudera Manager Ranger Plugin Supported Services page Configuration .
2. In Search, type spool directory.

   The following configuration properties display (this example uses the Kafka Broker role from Kafka service)

   Figure 18: Audit Spool Alert Configurations for Kafka Broker

   Ranger Kafka Plugin Audit Hdfs Spool Directory Path
   xasecure.audit.destination.hdfs.batch.filespool.dir
   ranger_audit_hdfs_spool_dir
   Kafka Broker
   /var/log/kafka/audit/hdfs/spool

   Ranger Kafka Plugin Audit Solr Spool Directory Path
   xasecure.audit.destination.solr.batch.filespool.dir
   ranger_audit_solr_spool_dir
   Kafka Broker
   /var/log/kafka/audit/solr/spool

   Enable Spool Directory Metric Usage for Kafka Broker
   spool_directory_usage_enabled
   ☑ Kafka Broker

   Ranger Plugin Spool Directory Usage Thresholds for Kafka Broker
   spool_directory_usage_threshold
   Kafka Broker
   10 GiB

3. In Enable Spool Directory Metric Usage for <service-name>, check the box.

   (un-checking the Enable box disables spool directory metric usage)

   a) Refresh the role.

      For example, the Kafka service which supports Ranger Plugin in Kafka Broker role, go to Cloudera Manager Kafka Instances Kafka Broker Actions Refresh Kafka Broker .

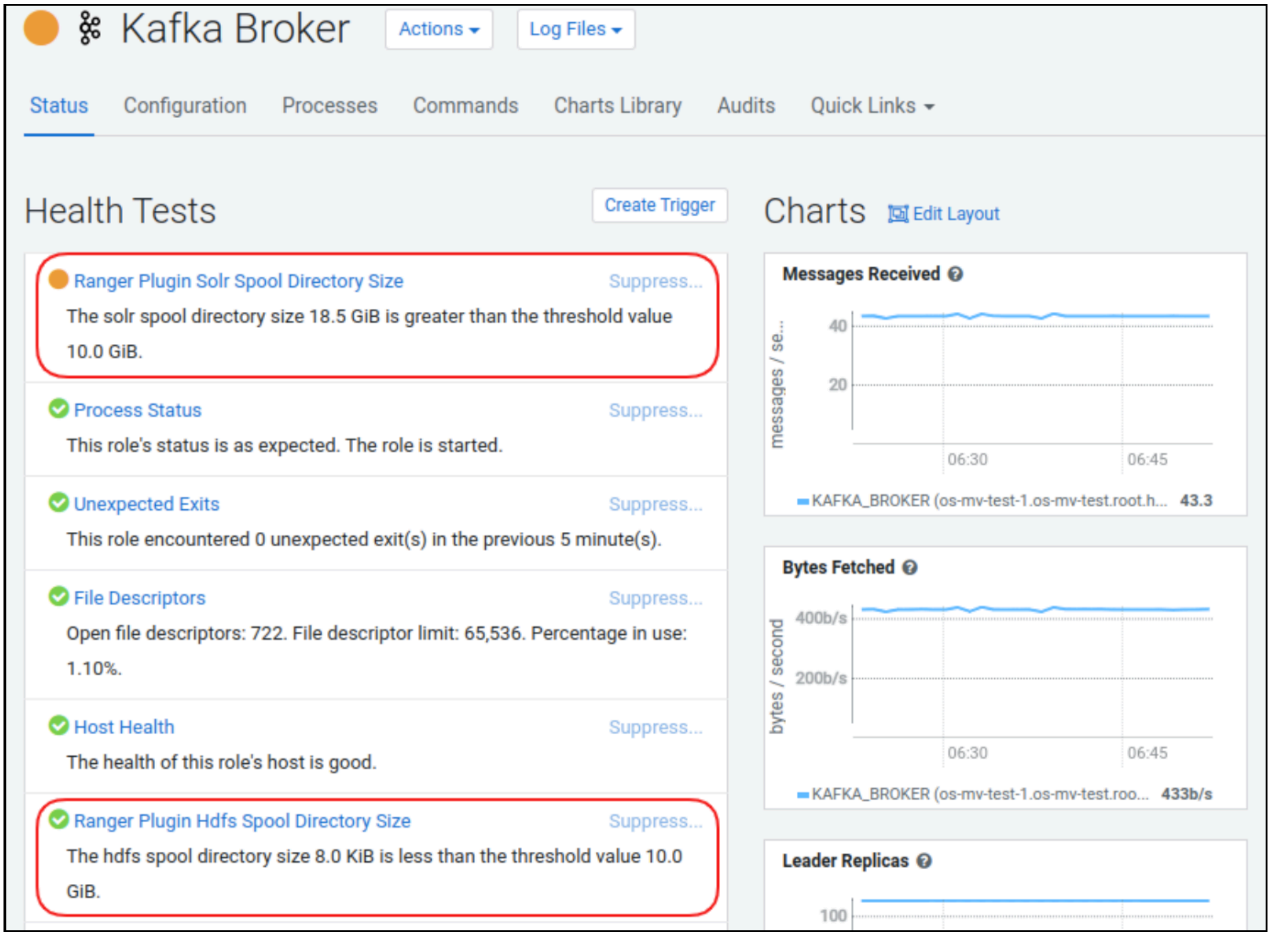
4. In Ranger Plugin Spool Directory Usage Thresholds for <service-name>, type values and select units.

   By default the threshold is set to 1 GB. If the disk usage of the spool directory exceeds this threshold, an alert will be shown.

Results

These spool alert details appear on the Ranger plugin service Role status page in the Cloudera Manager UI, as shown for Kafka Broker role in the following example:

Figure 19: Audit spool details for the Kafka Broker role



Ranger Plugin Solr Spool Directory Size Suppress...

The solr spool directory size 18.5 GiB is greater than the threshold value 10.0 GiB.

Process Status Suppress...

This role's status is as expected. The role is started.

Unexpected Exits Suppress...

This role encountered 0 unexpected exit(s) in the previous 5 minute(s).

File Descriptors Suppress...

Open file descriptors: 722. File descriptor limit: 65,536. Percentage in use: 1.10%.

Host Health Suppress...

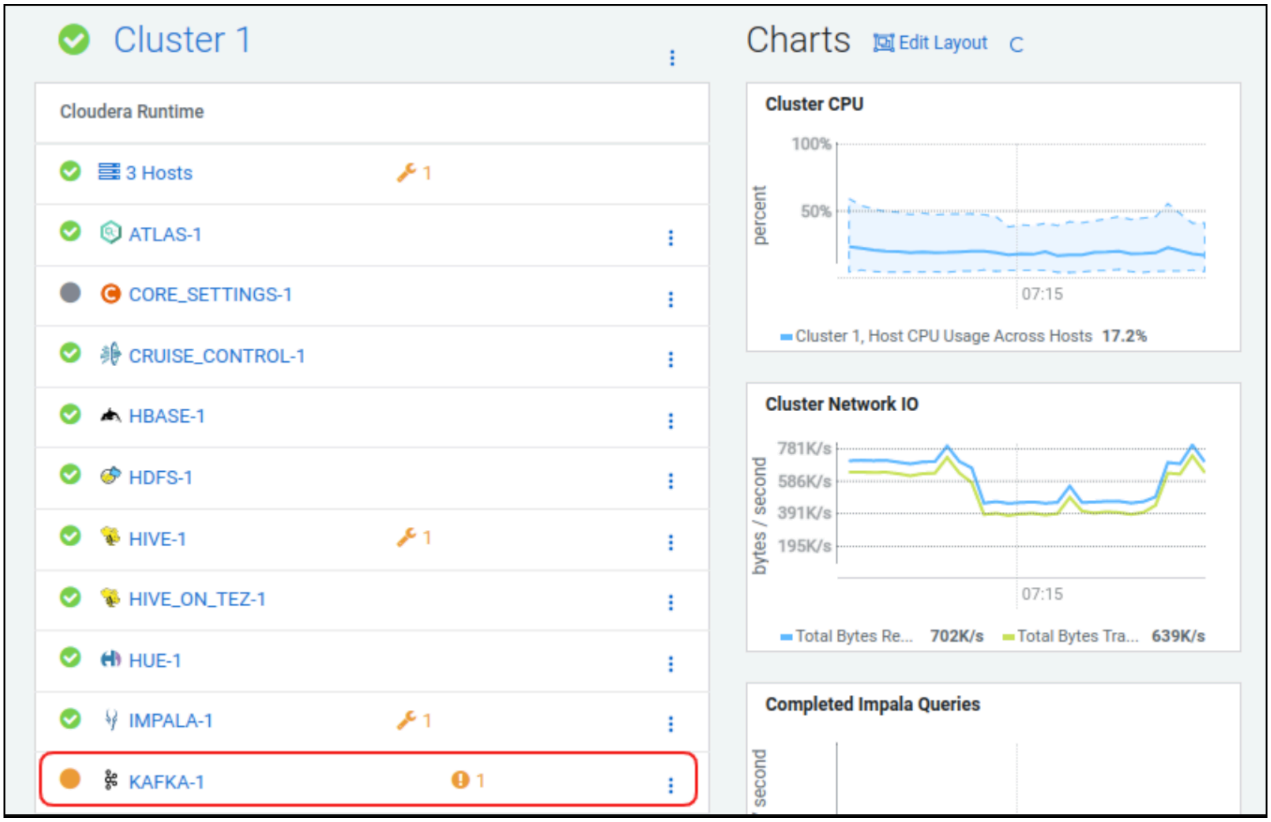
The health of this role's host is good.

Ranger Plugin Hdfs Spool Directory Size Suppress...

The hdfs spool directory size 8.0 KiB is less than the threshold value 10.0 GiB.

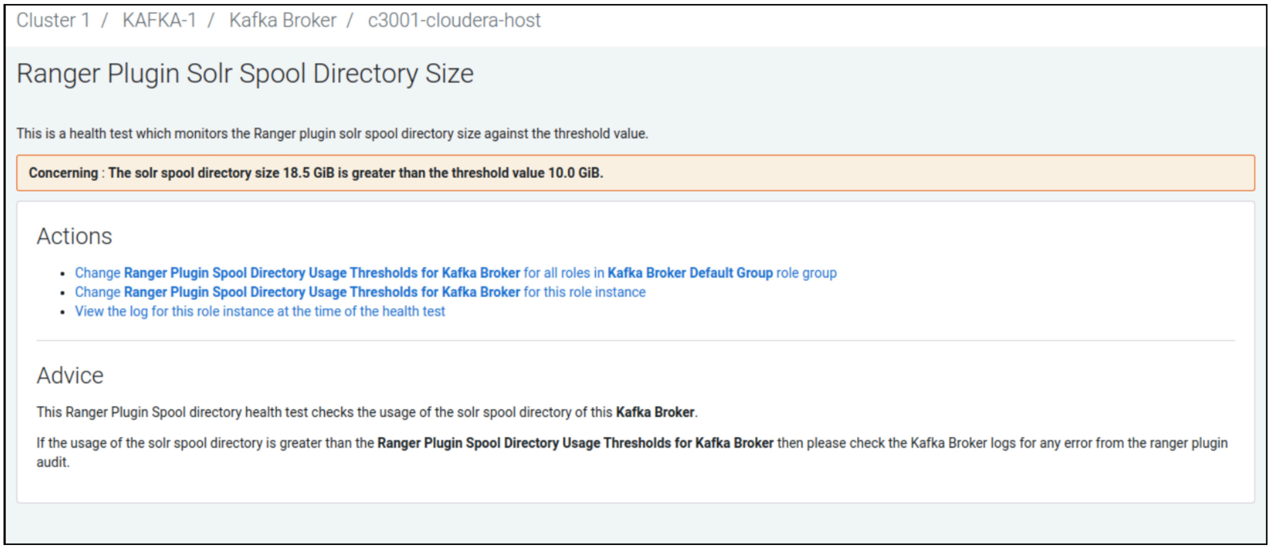
Also, an alert appears on the Cloudera Manager Home page next to the service name, as shown for Kafka service in the following example.

Figure 20: Audit file spool notification for Kafka service plugin on Cloudera Manager home page.



Also, Cloudera Manager shows Cluster Heath Actions and Advice for the role, as shown in the following example:

Figure 21: Health Test For Ranger Plugin Spool Alert



Cluster 1 / KAFKA-1 / Kafka Broker / c3001-cloudera-host

Ranger Plugin Solr Spool Directory Size

This is a health test which monitors the Ranger plugin solr spool directory size against the threshold value.

Concerning : The solr spool directory size 18.5 GiB is greater than the threshold value 10.0 GiB.

Actions

- Change **Ranger Plugin Spool Directory Usage Thresholds for Kafka Broker** for all roles in **Kafka Broker Default Group** role group
- Change **Ranger Plugin Spool Directory Usage Thresholds for Kafka Broker** for this role instance
- View the log for this role instance at the time of the health test

Advice

This Ranger Plugin Spool directory health test checks the usage of the solr spool directory of this **Kafka Broker**.

If the usage of the solr spool directory is greater than the **Ranger Plugin Spool Directory Usage Thresholds for Kafka Broker** then please check the Kafka Broker logs for any error from the ranger plugin audit.

What to do next

Optionally, you can create a graph for each spool alert metric.

Related Information

Charting spool alert metrics

Ranger audit log event summarization

You can summarize Ranger audit events that differ only by timestamp to reduce the amount of events logged in a busy system.

When summarization is enabled, if a Ranger plugin logs consecutive audit events that differ only by timestamp, it coalesces all such events into a single event and set event_count to the number of events logged and event_dur_ms to the time difference in milliseconds between the first and the last event.

To enable this feature you must set the following properties in the Ranger plugin's configuration (For example, if you want to enable Audit Summary for Hive plugin, you need to add these configurations to the ranger-hive-audit.xml file in Hive configurations):

Configurations	Description
xasecure.audit.provider.summary.enabled	To enable summarization set this property to true. This would cause audit messages to be summarized before they are sent to various sinks. By default it is set to false, which means audit summarization is disabled.
xasecure.audit.provider.queue.size	If unspecified, this value defaults to 1048576, which means the queue is sized to store 1M (1024 * 1024) messages. Note the difference in property name that controls the size of summary queue.
xasecure.audit.provider.summary.interval.ms	The maximum time interval at which messages would be summarized. If unspecified, it defaults to 5000, which means 5 seconds.
Summarization Batch size	Note that regardless of this time interval while summarizing at most 100k messages at a time are considered for aggregation. Thus, if more than 100k messages are logged during this interval then similar messages could show up as multiple summarized audit messages even though they are logged within the configured time interval. Currently, this value of 100k is not user configurable. It is mentioned here for better understanding of Summarization logic.

Charting spool alert metrics

Cloudera Manager supports chart visualization of spool directory disk usage metrics.

Single-level metrics for collecting the disk usage of Solr and Hdfs Ranger plugin spool directories are registered in Cloudera Manager, using the following disk usage metric names:

Solr

ranger_plugin_solr_spool_directory_size

HDFS

ranger_plugin_hdfs_spool_directory_size

You can chart disk usage of Solr and HDFS spool directory size as a time-series, using Cloudera Manager Chart Builder .

For example, to chart the disk usage for the Solr spool directory defined for Kafka Broker, use the folllowing query syntax:

SELECT ranger_plugin_solr_spool_directory_size WHERE roleType = KAFKA_BROKER, as shown in

Figure 22: Charting the disk usage of the Ranger Solr plugin spool directory for Kafka Broker

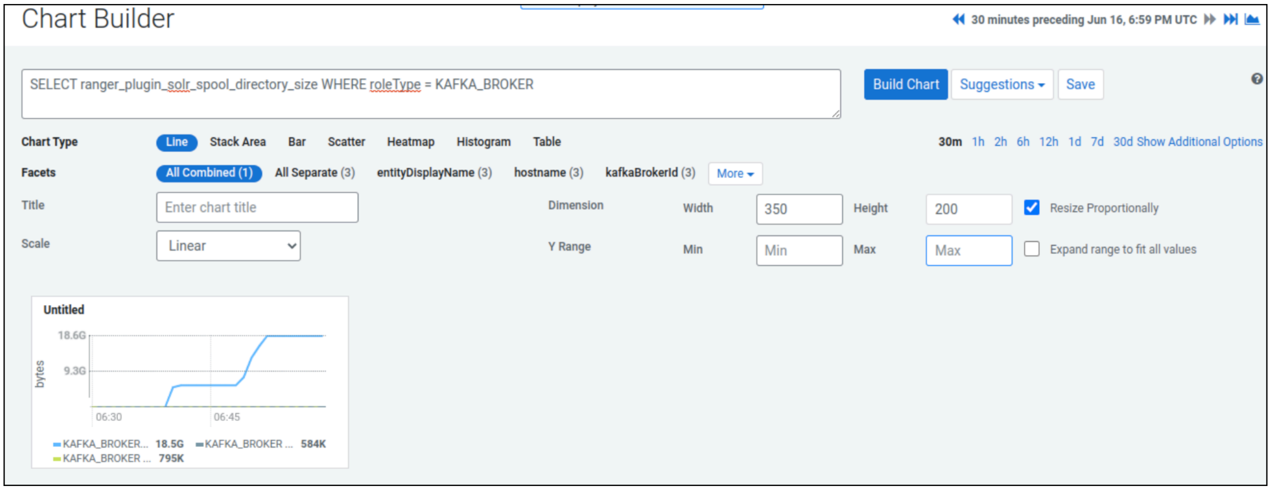


Chart Builder

```
SELECT ranger_plugin_solr_spool_directory_size WHERE roleType = KAFKA_BROKER
```

Chart Type: Line

Facets: All Combined (1)

Title: Enter chart title

Scale: Linear

Dimension: Width 350, Height 200, ☑ Resize Proportionally

Y Range: Min, Max, ☐ Expand range to fit all values

For more specific information about charting time-series data, see:

Related Information

Building a Chart with Time-Series Data

Excluding audits for specific users, groups, and roles

You can exclude audit records for specific users, groups, and roles from each service from appearing in the Ranger UI.

About this task

Ranger default log functionality creates audit log records for access and authorization requests, specifically around service accounts such as hbase, atlas and solr. Writing so much data to solr can limit the availability of Solr for further usage. This topic describes how to exclude audit records for specific users, groups, and roles from each service from appearing in the Ranger UI. Excluding specific users, groups or roles is also known as creating a blacklist for Ranger audits.

Procedure

1. In the Ranger Admin Web UI Service Manager , click Add New Service or Edit (existing service).
2. On Create/Edit Service, scroll down to Config Properties Add New Configurations .
3. Remove all audit filters from the existing service.

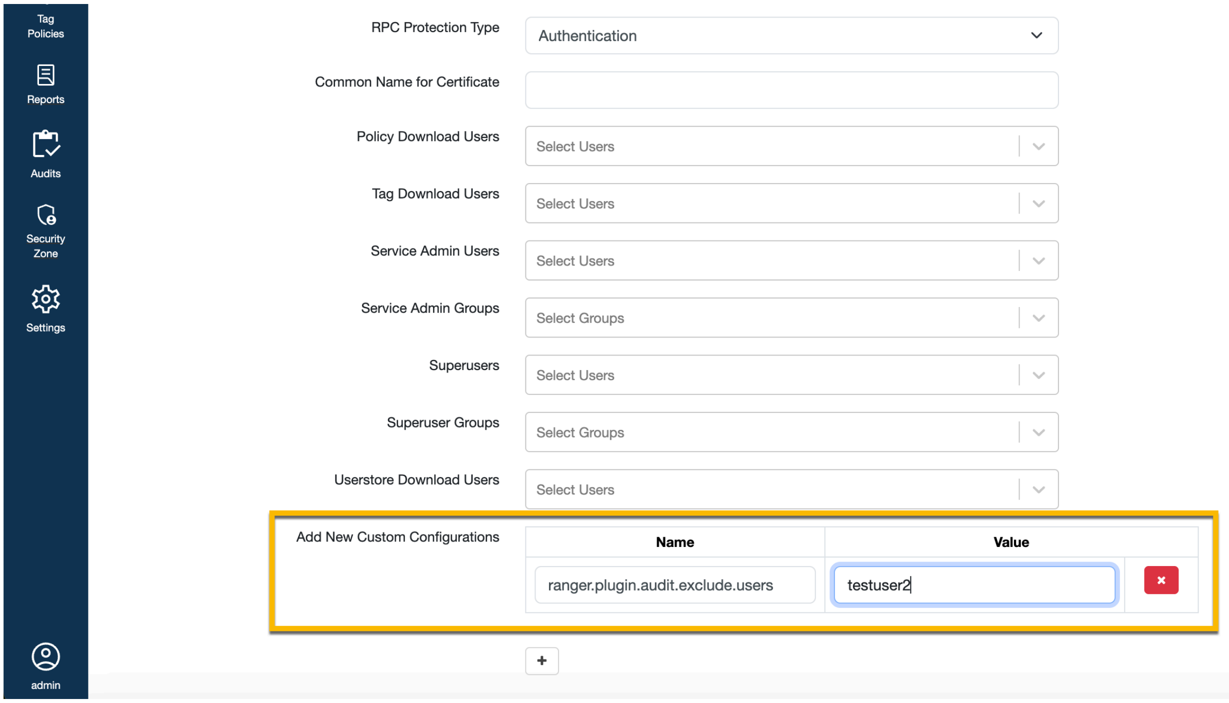
4. Click +, then type one of the following property names:

   - ranger.plugin.audit.exclude.users
   - ranger.plugin.audit.exclude.groups
   - ranger.plugin.audit.exclude.roles

   followed by one or more values.

   Note: You can include multiple values for each exclude property using a comma-separated list.

Figure 23: Adding an exclude users property to the HadoopSQL service



RPC Protection Type: Authentication

Common Name for Certificate:

Policy Download Users: Select Users

Tag Download Users: Select Users

Service Admin Users: Select Users

Service Admin Groups: Select Groups

Superusers: Select Users

Superuser Groups: Select Groups

Userstore Download Users: Select Users

Add New Custom Configurations

Name	Value
ranger.plugin.audit.exclude.users	testuser2

After adding the above configuration; if testuser2 user performs any actions for HadoopSQL service, Audit Access logs will not appear in the Ranger UI, but are still sent to Solr.

Similarly, you can exclude (or blacklist) users belonging to a particular group or role by adding a user-specific or role-specific configuration.

Changing Ranger audit storage location and migrating data

How to change the location of existing and future Ranger audit data collected by Solr from HDFS to a local file system or from a local file system to HDFS.

Before you begin

- Stop Atlas from Cloudera Manager.
- If using Kerberos, set the SOLR_PROCESS_DIR environment variable.

```
# export SOLR_PROCESS_DIR=$(ls -1dtr /var/run/cloudera-scm-agent/process/
*SOLR_SERVER | tail -1)
```

About this task

Starting with Cloudera Runtime version 7.1.4 / 7.2.2, the storage location for ranger audit data collected by Solr changed to local file system from HDFS, as was true for previous versions. The default storage location Ranger audit data storage location for Cloudera Runtime-7.1.4+ and Cloudera Runtime-7.2.2+ installations is local file system. After upgrading from an earlier Cloudera platform version, follow these steps to backup and migrate your Ranger audit data and change the location where Solr stores your future Ranger audit records.

- The default value of the index storage in the local file system is /var/lib/solr-infra. You can configure this, using Cloudera Manager Solr Configuration "Solr Data Directory" .
- The default value of the index storage in HDFS is /solr-infra. You can configure this, using Cloudera Manager Solr Configuration "HDFS Data Directory" .

Procedure

1. Create HDFS Directory to store the collection backups.

   As an HDFS super user, run the following commands to create the backup directory:

   ```
   # hdfs dfs -mkdir /solr-backups
   # hdfs dfs -chown solr:solr /solr-backups
   ```
2. Obtain valid kerberos ticket for Solr user.

   ```
   # kinit -kt solr.keytab solr/$(hostname -f)
   ```
3. Download the configs for the collection.

   ```
   # solrctl instancedir --get ranger_audits /tmp/ranger_audits
   # solrctl instancedir --get atlas_configs /tmp/atlas_configs
   ```
4. Modify the solrconfig.xml for each of the configs for which data needs to be stored in HDFS.

   In /tmp/<config_name>/conf created during Step 3., edit properties in the solrconfig.xml file as follows:

   - When migrating your data storage location from a local file system to HDFS, replace these two lines:

     ```
     <directoryFactory name="DirectoryFactory"
      class="${solr.directoryFactory:solr.NRTCachingDirectoryFactory}">
     <lockType>${solr.lock.type:native}</lockType>
     ```

     with

     ```
     <directoryFactory name="DirectoryFactory"
      class="${solr.directoryFactory:org.apache.solr.core.HdfsDirectoryFactory}">
     <lockType>${solr.lock.type:hdfs}</lockType>
     ```
   - When migrating your data storage location from HDFS to a local file system, replace these two lines:

     ```
     <directoryFactory name="DirectoryFactory"
      class="${solr.directoryFactory:org.apache.solr.core.HdfsDirectoryFactory}">
     <lockType>${solr.lock.type:hdfs}</lockType>
     ```

     with

     ```
     <directoryFactory name="DirectoryFactory"
      class="${solr.directoryFactory:solr.NRTCachingDirectoryFactory}">
     <lockType>${solr.lock.type:native}</lockType>
     ```
5. Backup the Solr collections.

   - When migrating your data storage location from a local file system to HDFS, run:

     ```
      # curl -k --negotiate -u : "https://$(hostname
      -f):8995/solr/admin/collections?action=BACKUP&name=vertex_backup&coll
      ection=vertex_index&
     ```

```
 location=hdfs://<Namenode_Hostname>:8020/solr-backups&async=vertex_back
 up"
```

In the preceding command, the important points are name, collection, and location:

name

specifies the name of the backup. It should be unique per collection

collection

specifies the collection name for which the backup will be performed

location

specifies the HDFS path, where the backup will be stored

Repeat the curl command for different collections, modifying the parameters as necessary for each collection.

The expected output would be -

```
"responseHeader":{
 "status":0,
 "QTime":10567},
"success":{
 "Solr_Server_Hostname:8995_solr":{
   "responseHeader":{
   "status":0,
   "QTime":8959}}}}
```

- When migrating your data storage location from HDFS to a local file system:

  Refer to Back up a Solr collection for specific steps, and make the following adjustments:

  - If TLS is enabled for the Solr service, specify the trust store and password by using the ZKCLI_JVM_FLAGS environment variable before you begin the procedure.

    ```
    # export ZKCLI_JVM_FLAGS="-Djavax.net.ssl.trustStore=/path/to/
    truststore.jks -Djavax.net.ssl.trustStorePassword="
    ```
  - Create Snapshot

    ```
    # solrctl --jaas $SOLR_PROCESS_DIR/jaas.conf collection --create-
    snapshot <snapshot_name> -c <collection_name>
    ```
  - or use the Solr API to take the backup:

    ```
    curl -i -k --negotiate -u : "https://(hostname -f):8995/solr/admin/
    collections?
    action=BACKUP&name=ranger_audits_bkp&collection=ranger_audits&location=/
    path/to/solr-backups"
    ```
  - Export Snapshot

    ```
    # solrctl --jaas $SOLR_PROCESS_DIR/jaas.conf collection
     --export-snapshot <snapshot_name> -c <collection_name> -d
     <destination_directory>
    ```

    Note: The <destination_directory> is a HDFS path. The ownership of this directory should be solr:solr.

6. Update the modified configs in Zookeeper.

```
# solrctl --jaas $SOLR_PROCESS_DIR/jaas.conf instancedir --update
 atlas_configs /tmp/atlas_configs
# solrctl --jaas $SOLR_PROCESS_DIR/jaas.conf instancedir --update
 ranger_audits /tmp/ranger_audits
```

7. Delete the collections from the original location.

   All instances of Solr service should be up, running, and healthy before deleting the collections. Use Cloudera Manager to check for any alerts or warnings for any of the instances. If alerts or warnings exist, fix those before deleting the collection.

   ```
   # solrctl collection --delete edge_index
   # solrctl collection --delete vertex_index
   # solrctl collection --delete fulltext_index
   # solrctl collection --delete ranger_audits
   ```
8. Verify that the collections are deleted from the original location.

   ```
   # solrctl collection --list
   ```

   This will give an empty result.
9. Verify that no leftover directories for any of the collections have been deleted.

   - When migrating your data storage location from a local file system to HDFS:

     ```
     # cd /var/lib/solr-infra
     ```

     Get the value of "Solr Data Directory, using Cloudera Manager Solr Configuration .

     ```
     # ls -ltr
     ```
   - When migrating your data storage location from HDFS to a local file system, replace these two lines:

     ```
     # hdfs dfs -ls /solr/<collection_name>
     ```

     Note: If any directory name which starts with the collection name deleted in Step 7. exists, delete/ move the directory to another path.

10. Restore the collection from backup to the new location.

    Refer to Restore a Solr collection, for more specific steps.

    ```
    # curl -k --negotiate -u : "https://$(hostname
    -f):8995/solr/admin/collections?
    action=RESTORE&name=<Name_of_backup>&location=hdfs:/
    <<Namenode_Hostname>:8020/solr-
    backups&collection=<Collection_Name>&async=<any_unique_name>"
    ```

    ```
    # solrctl collection --restore ranger_audits
     -l hdfs://<Namenode_Hostname>:8020/solr-backups
     -b ranger_backup -i ranger1
    ```

    The request id must be unique for each restore operation, as well as for each retry.

    To check the status of restore operation:

    ```
    # solrctl collection --request-status <requestId>
    ```

    Note: If the Atlas Collections (vertex_index, fulltext_index and edge_index) restore operations fail, restart the solr service and rerun the restore command. Now, the restart operations should complete successfully.

11. Verify the Atlas and Ranger functionality.

Verify that both Atlas and Ranger audits functions properly, and that you can see the latest audits in Ranger Web UI and latest lineage in Atlas.

- To verify Atlas audits, create a test table in Hive, and then query the collections to see if you are able to view the data.
- You can also query the collections every 20-30 seconds (depending on how other services utilize Atlas/Ranger), and verify if the "numDocs" value increases at every query.

```
# curl -k --negotiate -u : "https://$(hostname -f):8995/solr/edge_index/
select?q=*%3A*&wt=json&ident=true&rows=0"
# curl -k --negotiate -u : "https://$(hostname -f):8995/solr/vertex_index/
select?q=*%3A*&wt=json&ident=true&rows=0"
# curl -k --negotiate -u : "https://$(hostname -f):8995/solr/
fulltext_index/select?q=*%3A*&wt=json&ident=true&rows=0"
# curl -k --negotiate -u : "https://$(hostname -f):8995/solr/
ranger_audits/select?q=*%3A*&wt=json&ident=true&rows=0"
```

Configuring Ranger audits to show actual client IP address

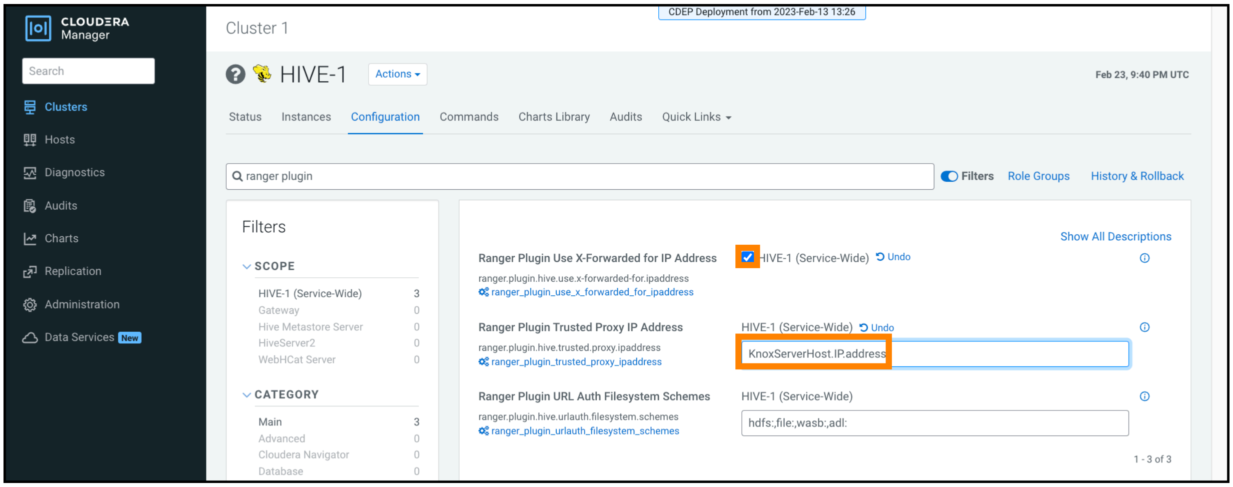
How to forward the actual client IP address to audit logs generated from a Ranger plugin.

About this task

Ranger audit logs record the IP address through which Ranger policies grant/authorize access. When Ranger is set up behind a Knox proxy server, the proxy server IP address appears in the audit logs generated for each Ranger plugin. You can configure each plugin to forward the actual client IP address on which that service runs, so that the audit logs for that service more specifically reflect access/authorization activity. You must configure each plugin individually. This topic uses the Hive (Hadoop SQL) service as an example.

Procedure

1. From Cloudera Manager choose <service_name> Configuration .
2. In <service_name> Configuration Search , type ranger-plugin, then press Return.
3. In Ranger Plugin Use X-Forwarded for IP Address, check the box.
4. In Ranger Plugin Trusted Proxy IP Address, type the IP address of the Knox proxy server host.



ranger plugin

Ranger Plugin Use X-Forwarded for IP Address ☑ HIVE-1 (Service-Wide)
ranger.plugin.hive.use.x-forwarded-for.ipaddress
ranger_plugin_use_x_forwarded_for_ipaddress

Ranger Plugin Trusted Proxy IP Address HIVE-1 (Service-Wide): KnoxServerHost.IP.address
ranger.plugin.hive.trusted.proxy.ipaddress
ranger_plugin_trusted_proxy_ipaddress

Ranger Plugin URL Auth Filesystem Schemes HIVE-1 (Service-Wide): hdfs:,file:,wasb:,adl:
ranger.plugin.hive.urlauth.filesystem.schemes
ranger_plugin_urlauth_filesystem_schemes

Results

Hive audit logs will now show the IP address of the host on which Hive service runs.