

Profilers

Date published: 2023-12-16

Date modified: 2025-11-10



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.

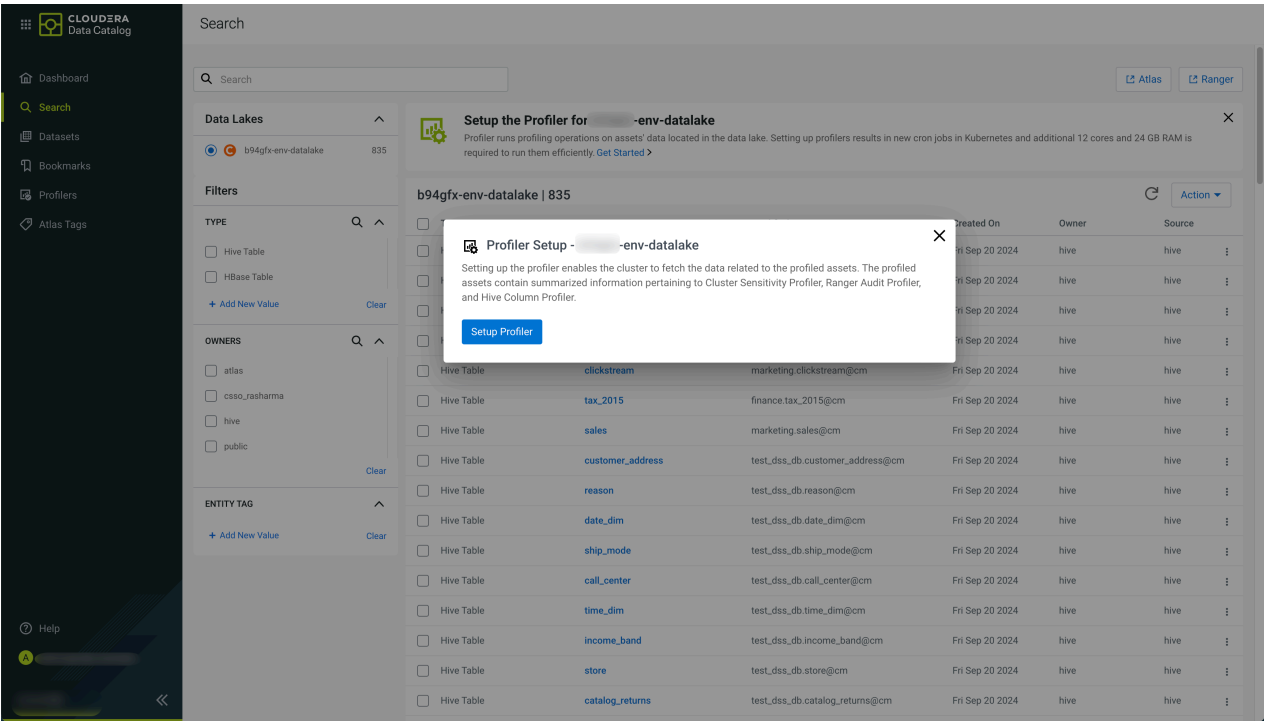
Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

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.

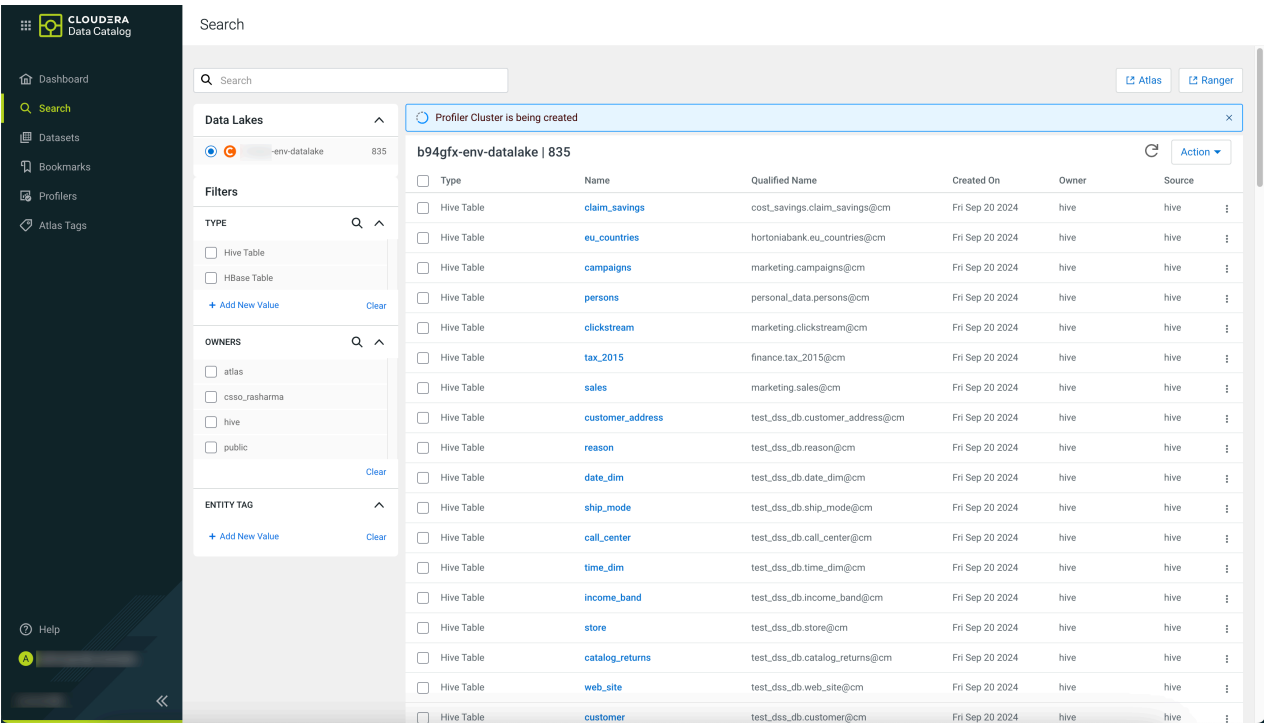
Contents

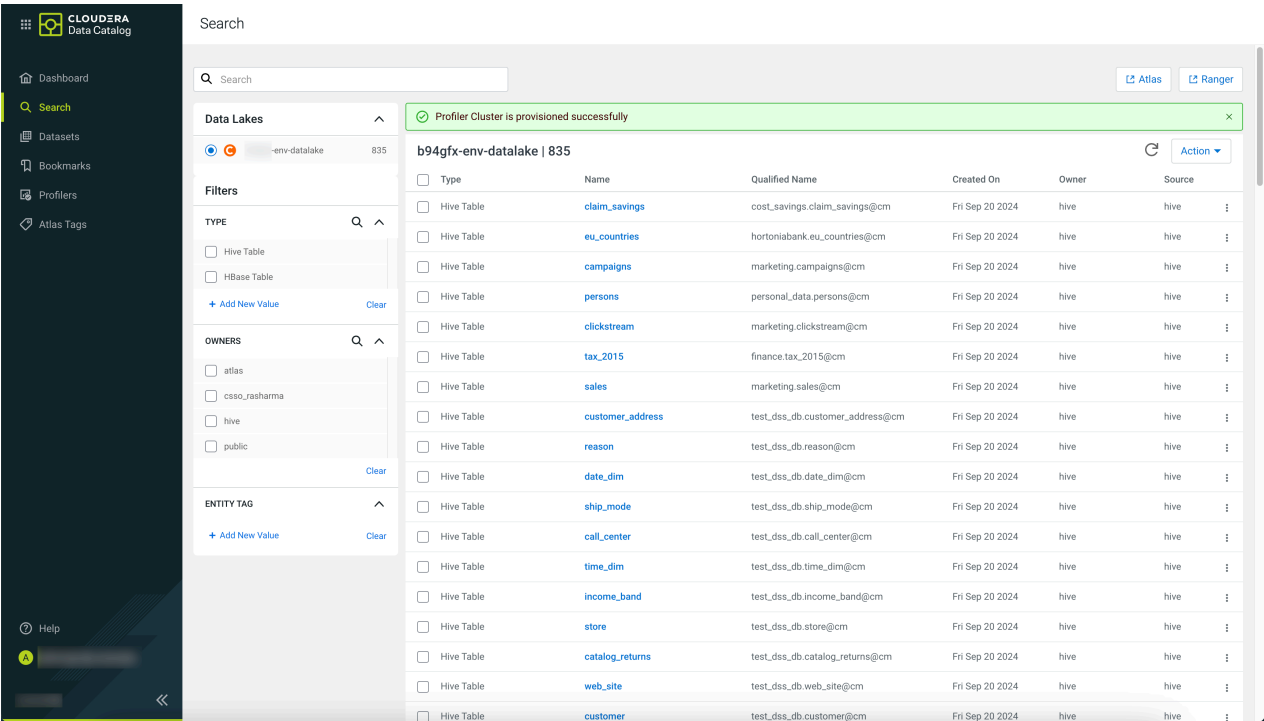
Launching profilers.....	4
Launching profilers using the command-line.....	6
Deleting profilers.....	10
On-Demand Profilers.....	11
Tracking Profiler Jobs.....	14
Viewing profiler configurations.....	15
Ranger Audit Profiler configuration.....	16
Hive Column Profiler configuration.....	17
Cluster Sensitivity Profiler configuration.....	19
Profiler tag rules.....	20
Creating custom profiler rules.....	21
Adding custom regular expressions.....	22
Using DSL grammar.....	23
Understanding the Cron Expression generator.....	24
Enabling or disabling profilers.....	25



The High Availability (HA) feature for profilers, including launching and managing jobs are supported by default. No separate action is required to enable the HA functionality or its components.

 **Note:** Once you schedule the profiler jobs, navigate to the **Profilers** page to view the status of the respective profiler jobs.





- Related Information**
- Cloudera Data Catalog Profilers
 - The Cluster Sensitivity Profiler
 - The Ranger Audit Profiler
 - The Hive Column Profiler

Launching profilers using the command-line

Cloudera Data Catalog supports launching profilers using the Command-Line Interface (CLI) option.

The CLI is one executable and does not have any external dependencies. You can execute some operations in the Cloudera Data Catalog service using the Cloudera CLI commands.

Users must have valid permissions to launch profilers on a data lake.

For more information about the access details, see [Prerequisites to access Cloudera Data Catalog](#).

Prerequisites

You must have the following entitlement granted to use this feature:

DATA_CATALOG_ENABLE_API_SERVICE

The Cloudera Data Catalog CLI

In your Cloudera CLI environment, enter the following command to get started in the CLI mode.

`cdp datacatalog --help`

This command provides information about the available commands in Cloudera Data Catalog.

The output is displayed as:

NAME
datacatalog
DESCRIPTION

Cloudera Data Catalog Service is a web service, using this service user can execute operations like launching profilers in Data Catalog.

AVAILABLE SUBCOMMANDS

launch-profilers

Parameters for profiler launch command

You get additional information about this command by using:

cdp datacatalog launch-profilers --help

NAME

launch-profilers -

DESCRIPTION

Launches DataCatalog profilers in a given datalake.

NAME

launch-profilers - Launches DataCatalog profilers in a given datalake.

DESCRIPTION

Launches DataCatalog profilers in a given datalake.

SYNOPSIS

```
launch-profilers
--datalake <value>
[--enable-ha | --no-enable-ha]
[--profilers <value>]
[--instance-types <value>]
[--max-nodes <value>]
[--cli-input-json <value>]
[--generate-cli-skeleton]
```

OPTIONS

```
--datalake (string)
    The CRN of the Datalake.

--enable-ha | --no-enable-ha (boolean)
    Enables High Availability (HA) for datacatalog profilers (default value is false). The High Availability (HA) Profiler cluster provides failure resilience and scalability but incurs additional cost.

--profilers (array)
    List of profiler names that need to be launched. (Applicable only for compute cluster enabled environments).
```

Syntax:

```
"string" "string" ...
```

```
--instance-types (array)
    List of instance types to be used for the auto-scaling node group setup (Applicable only for compute cluster enabled environments).
```

Syntax:

```
"string" "string" ...
```

```
--max-nodes (integer)
    Maximum number of nodes that can be spawned inside the auto-scaling
```

```

node group, in the range of 30 to 100 (both inclusive). (Applicable
only for compute cluster enabled environments).

--cli-input-json (string)
    Performs service operation based on the JSON string provided. The
    JSON string follows the format provided by --generate-cli-skeleton
on.
    If other arguments are provided on the command line, the CLI value
s
    will override the JSON-provided values.
--generate-cli-skeleton (boolean)
    Prints a sample input JSON to standard output. Note the specified
    operation is not run if this argument is specified. The sample i
nput
    can be used as an argument for --cli-input-json.
OUTPUT
success -> (boolean)
    Status of the profiler launch operation.

datahubCluster -> (object)
    Information about a cluster.

    clusterName -> (string)
        The name of the cluster.

    crn -> (string)
        The CRN of the cluster.

    creationDate -> (datetime)
        The date when the cluster was created.

    clusterStatus -> (string)
        The status of the cluster.

    nodeCount -> (integer)
        The cluster node count.

    workloadType -> (string)
        The workload type for the cluster.

    cloudPlatform -> (string)
        The cloud platform.

    imageDetails -> (object)
        The details of the image used for cluster instances.

        name -> (string)
            The name of the image used for cluster instances.

        id -> (string)
            The ID of the image used for cluster instances. This is
            internally generated by the cloud provider to uniquely
            identify the image.

        catalogUrl -> (string)
            The image catalog URL.

        catalogName -> (string)
            The image catalog name.

    environmentCrn -> (string)
        The CRN of the environment.

```



```

credentialCrn -> (string)
    The CRN of the credential.

datalakeCrn -> (string)
    The CRN of the attached datalake.

clusterTemplateCrn -> (string)
    The CRN of the cluster template used for the cluster creation.

FORM FACTORS
    public

```

**Note:**

- The following parameters are only applicable to Compute Cluster environments (they are ignored in VM-based environments):
 - `--profilers ***VALUE***`
 - `--instance-types ***VALUE***`
 - `--max-nodes ***VALUE***`

Parameters for profiler delete command

You get additional information about this command by using:

```
cdp datacatalog delete-profiler --help
```

```

NAME
    delete-profiler - Deletes DataCatalog profiler in a given datalake.
DESCRIPTION
    Deletes DataCatalog profiler in a given datalake.
SYNOPSIS
    delete-profiler
    --datalake <value>
    [--cli-input-json <value>]
    [--generate-cli-skeleton]
OPTIONS
    --datalake (string)
        The CRN of the Datalake.

    --cli-input-json (string)
        Performs service operation based on the JSON string provided. The
        JSON string follows the format provided by --generate-cli-skeleton
        .
        If other arguments are provided on the command line, the CLI val
        ues
        will override the JSON-provided values.

    --generate-cli-skeleton (boolean)
        Prints a sample input JSON to standard output. Note the specified
        operation is not run if this argument is specified. The sample inp
        ut
        can be used as an argument for --cli-input-json.

OUTPUT
FORM FACTORS
    public

```

Launching the profiler

You can use the following CLI command to launch the data profiler:

```
cdp datacatalog launch-profilers --datalake [***DATA LAKE CRN***]
```

Example:

```
cdp datacatalog launch-profilers --datalake crn:cdp:data  
lake:DATACENTERNAME:c*****b-ccce-4**d-a**1-8*****8:datalake:4*****5e-c**  
1-4**2-8**e-1*****2  
{  
  "success": true  
}
```

Deleting profilers

Deleting the profiler container (pod) jobs removes all the Custom Sensitivity Profiler rules and other updates to the specified profiler.

About this task

To overcome this situation, when you decide to delete the profiler jobs, there is a provision to retain the status of the Cluster Sensitivity Profiler rules:

- If your profiler jobs have rules that are not changed or updated, you can directly delete them.
- If the rules were modified or updated, you have an option to download the modified rules along with deletion. The modified rules consist of the suspended system rules and the deployed custom rules. Using the downloaded rules, you can manually add or modify them to your newly added profiler jobs or the profiler cluster.



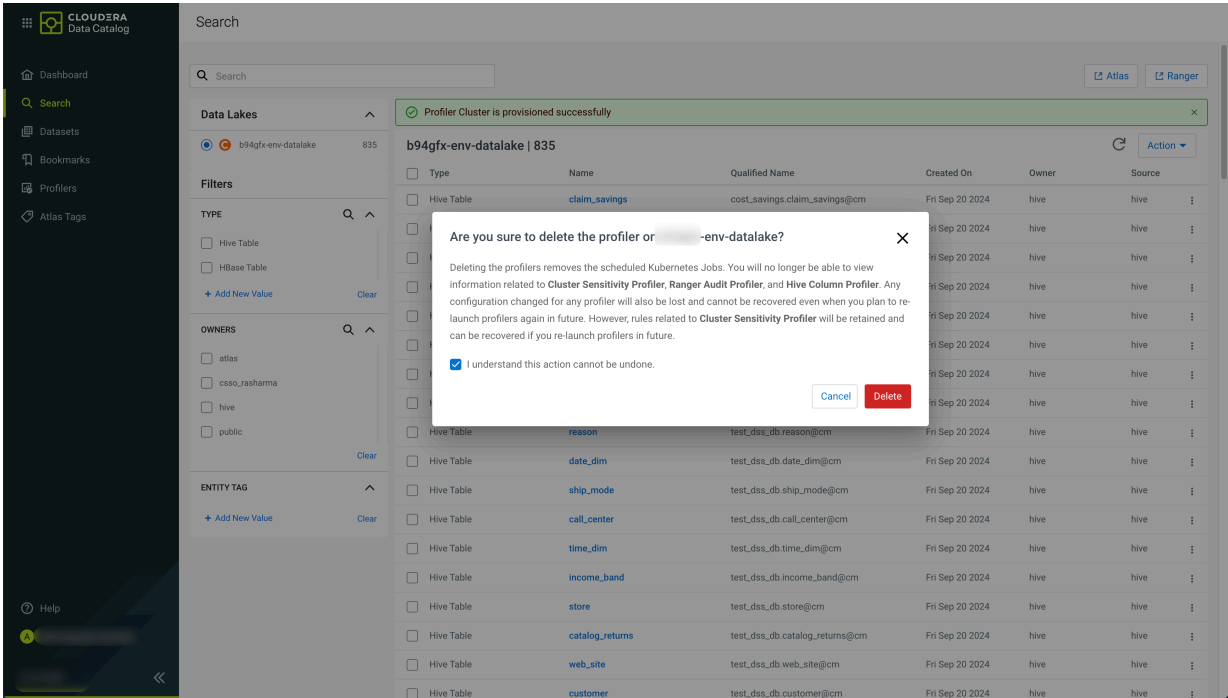
Note:

- When you delete the scheduled jobs, the associated Kubernetes cron job object is deleted from the Kubernetes cluster.
- The associated data of the profilers from the Cloudera Management Console database is also deleted for the specified data lake.

Procedure

1. On the **Search** page, select the data lake.
2. Click the Actions drop-down menu and select Delete Cluster.
3. Click Yes to proceed.

4. If you agree, select the warning message I understand this action cannot be undone.



5. Click **Delete**.


The application displays the following message.

The profiler cluster is deleted successfully.

On-Demand Profilers

You can use On-Demand Profilers to profile specific assets without depending on the cron-based scheduling of profilers jobs. The On-Demand Profiler option is available in the Asset Details of the selected asset.

The following image shows the **Asset Details** page of an asset. You can run an On-Demand Profiler for Hive Column Profiler and Cluster Sensitivity Profiler by clicking on the appropriate Run button. The next scheduled run provides details about the next scheduled profiling for the respective profilers.

 **Note:** You can use the On-Demand Profiler feature to profile both external and managed tables.

Asset Details

raw_bookings

Properties

Type: **HIVE TABLE**

of Columns: **8**

Data Lake: **cloudera-catalog-vm-10**

Datasets: **0**

Owner:

Created On: **10/15/2025 05:38 PM CEST**

Last Updated At: **10/16/2025 01:46 PM CEST**

Table Type: **EXTERNAL_TABLE**

Database: **airline_operations**

DB Catalog: **cm**

Parent: **airline_operations**

Qualified Name

airline_operations.raw_bookings@cm

Comment

[+ Add Comment](#)

Description

[+ Add Description](#)

Profilers | 2

Cluster Sensitivity Profiler

Profiler Job Request Received.

Profiling in progress (1 / 4) ...

Once the profiler is finished, the page is refreshed.

Hive Column Profiler

Last run: **10/16/2025 01:46 PM CEST** | Status: **SUCCESS**

[Run](#)

Next Schedule Run: **10/16/2025 02:45 PM CEST**

Classifications

[+ Add Classification](#)

Managed

System

Propagated

Terms

[+ Add Terms](#)

Figure 1: Tracking on-demand profilers

Profilers

Data Lake

cloudera-catalog-vm-...

Jobs

Configs

Tag Rules

Filters

Clear All

Job Status

☐ Finished

24

☐ Running

0

☐ Failed

0

Profilers

☐ Cluster Sensitivity Profiler

9

☐ Ranger Audit Profiler

6

☐ Hive Column Profiler

9

Profiler	Stage	Status	Job ID	Start On	Last Updated On
On-Demand Cluster Sensitivity	Livy	Running	87	10/16/2025 01:49 PM CEST	10/16/2025 01:49 PM CEST
On-Demand Cluster Sensitivity	Scheduler Service	Finished	86	10/16/2025 01:49 PM CEST	10/16/2025 01:49 PM CEST
On-Demand Cluster Sensitivity	Admin Service	Finished	85	10/16/2025 01:48 PM CEST	10/16/2025 01:48 PM CEST
Table Stats	Metrics Service	Finished	83	10/16/2025 01:46 PM CEST	10/16/2025 01:46 PM CEST
Table Stats	Livy	Finished	81	10/16/2025 01:45 PM CEST	10/16/2025 01:46 PM CEST
Table Stats	Scheduler Service	Finished	78	10/16/2025 01:45 PM CEST	10/16/2025 01:45 PM CEST
Cluster Sensitivity	Metrics Service	Finished	84	10/16/2025 01:48 PM CEST	10/16/2025 01:48 PM CEST
Cluster Sensitivity	Livy	Finished	82	10/16/2025 01:45 PM CEST	10/16/2025 01:48 PM CEST
Cluster Sensitivity	Scheduler Service	Finished	79	10/16/2025 01:45 PM CEST	10/16/2025 01:45 PM CEST
Ranger Audit	Livy	Finished	80	10/16/2025 01:45 PM CEST	10/16/2025 01:46 PM CEST
Ranger Audit	Scheduler Service	Finished	77	10/16/2025 01:45 PM CEST	10/16/2025 01:45 PM CEST
Cluster Sensitivity	Metrics Service	Finished	76	10/16/2025 12:48 PM CEST	10/16/2025 12:48 PM CEST
Cluster Sensitivity	Livy	Finished	74	10/16/2025 12:45 PM CEST	10/16/2025 12:48 PM CEST
Cluster Sensitivity	Scheduler Service	Finished	70	10/16/2025 12:45 PM CEST	10/16/2025 12:45 PM CEST
Table Stats	Metrics Service	Finished	75	10/16/2025 12:46 PM CEST	10/16/2025 12:46 PM CEST
Table Stats	Livy	Finished	73	10/16/2025 12:45 PM CEST	10/16/2025 12:46 PM CEST

12

Asset Details

enriched_flight_data

Properties

Type: ICEBERG TABLE
of Columns: 8
Data Lake: cloudera-catalog-hue-lake-9
Datasets: 0
Owner:
Created On: 10/15/2025 05:30 PM CEST
Last Updated At: 10/15/2025 05:43 PM CEST
Table Type: EXTERNAL_TABLE
Database: airline_operations
DB Catalog: cm
Parent: airline_operations

Qualified Name
airline_operations.enriched_flight_data@cm

Comment
+ Add Comment

Description
+ Add Description

Profilers | 2

Data Compliance Profiler

Last run: 10/15/2025 06:50 PM CEST | Status: SUCCESS [Run](#)
Next Schedule Run: 10/16/2025 06:48 PM CEST

Statistics Collector Profiler

Processing the asset.
Once the profiler is finished, the page is refreshed.

Classifications

[Managed](#) [System](#) [Propagated](#)
+ Add Classification

Terms

+ Add Terms

Figure 2: Tracking on-demand profilers

Profilers Details

Profilers

Profilers Details

✓ Data Compliance Profiler

cloudera-catalog-hue-lake-9

RECENT JOB ID
JDMTBVME

TOTAL JOBS
15

TOTAL PROFILED ASSETS
53

LAST RUN
10/16/2025 01:44 PM CEST

NEXT RUN
10/16/2025 06:48 PM CEST

SCHEDULE FREQUENCY (UTC)
At 04:48 PM

Disable Profiler

Job History

Configuration

Tag Rules

Search by Job Id

Status

Time Range

Job Type

Clear All

Refresh

The Job History shows the profiling jobs started in the last 30 days by default.

Status	Job Id	Job Type	Started On	Finished On	Profiled Assets
✓	JDMTBVME	On Demand	10/16/2025 01:44 PM CEST	10/16/2025 01:45 PM CEST	1 / 1
✓	WLVSQ3KE	On Demand	10/16/2025 01:43 PM CEST	10/16/2025 01:45 PM CEST	1 / 1
✓	2RXKPSH4	Scheduled	10/15/2025 06:48 PM CEST	10/15/2025 06:50 PM CEST	6 / 6
✓	PNVZG29V	Scheduled	10/15/2025 05:40 PM CEST	10/15/2025 05:42 PM CEST	7 / 7
✓	Z2FKKSX8	Scheduled	10/15/2025 04:49 PM CEST	10/15/2025 04:50 PM CEST	7 / 7
✓	GKDTJA9V	On Demand	10/15/2025 04:39 PM CEST	10/15/2025 04:40 PM CEST	1 / 1
✓	9C6M3JKY	On Demand	10/15/2025 04:38 PM CEST	10/15/2025 04:38 PM CEST	1 / 1
✓	YDZVTJPR	On Demand	10/15/2025 04:36 PM CEST	10/15/2025 04:37 PM CEST	1 / 1
✓	UP998XNU	Scheduled	10/15/2025 03:49 PM CEST	10/15/2025 03:51 PM CEST	7 / 7

13

Asset Details

enriched_flight_data

Properties

Type: **HIVE TABLE**
of Columns: **9**
Data Lake:
Datasets: **0**
Owner: **admin**
Created On: **Wed Oct 15 2025 18:07:14 GMT+0200 (Centr...**
Last Access Time: **Wed Oct 15 2025 18:07:14 GMT+0200...**
Table Type: **EXTERNAL_TABLE**
Database: **airline_operations**
DB Catalog: **cm**
Parent: **airline_operations**

Qualified Name
airline_operations.enriched_flight_data@cm

Comment
[+ Add Comment](#)

Description
[+ Add Description](#)

Classifications

[+ Add Classification](#)

Profilers | 2

Cluster Sensitivity Profiler

Last run: **NA** | Status: **NA**
Next Schedule Run: **10/17/2025, 8:45:00 AM (UTC)**
[Run](#)

Hive Column Profiler

Profiler Job Request Received.
Profiling in progress...
Once the profiler is complete the page will refresh

Terms

[+ Add Terms](#)

Figure 3: Tracking on-demand profilers

Profilers / Jobs

-datalake

Jobs

Configs

Tag Rules

Filters

Clear All

-datalake

Job Status

☐ Finished 4

☐ Running 0

☐ Failed 0

Profilers

☐ Ranger Audit Profiler 0

☐ Hive Column Profiler 3

☐ Cluster Sensitivity Profiler 1

Profiler	Status	Job ID	Started On	Last Updated On
Table Stats (On-Demand)	Finished	hzgi7yaW	Oct 16 2025 14:04:51	Oct 16 2025 14:07:04
Cluster Sensitivity (On-Demand)	Finished	WmFv8hfs	Oct 16 2025 14:01:36	Oct 16 2025 14:04:12
Table Stats (On-Demand)	Finished	pAM6NJGx	Oct 16 2025 14:01:36	Oct 16 2025 14:05:02
Table Stats	Finished	RbT9QEX8	Oct 16 2025 10:45:29	Oct 16 2025 10:52:16

D

W

M

 **Note:**

Tracking Profiler Jobs

Use the Profilers > Jobs page for tracking the respective profiler jobs.

Under Profilers Jobs , you can have an overview of your started profiler jobs. By using the D, W, M filters, you can go back up to a day, week or a month, to see your previous jobs. Use this page to quickly check if your profiler jobs are failing.

Profilers / Jobs

datalake

Jobs

Configs

Filters

Clear All

Job Status

☐ Finished

3

☐ Running

0

☐ Failed

1

Profilers

☐ Ranger Audit Profiler

0

☐ Cluster Sensitivity Profiler

2

☐ Hive Column Profiler

2

datalake

D

W


M

Profiler	Status	Job ID	Start Time	Last Updated
Cluster Sensitivity	Failed	J8dKWBHi	Sep 10 2024 15:03:17	Sep 10 2024 15:03:21
Table Stats (On-Demand)	Finished	3Dq8SWnJ	Sep 10 2024 14:50:43	Sep 10 2024 14:50:47
Table Stats (On-Demand)	Finished	5wqVCKX8	Sep 10 2024 14:48:39	Sep 10 2024 14:49:36
Cluster Sensitivity (On-Demand)	Finished	iw7vFnFW	Sep 10 2024 14:46:32	Sep 10 2024 14:47:35

For each profiler job, you can view the details about:

- Profiler type
- Job Status
- Job ID
- Start Time
- Last Updated Time

Using this data can help you to troubleshoot failed jobs or even understand how the jobs were profiled and other pertinent information that can help you to manage your profiled assets.



Note: More than one occurrence of Scheduler Service or Livy indicates that there could be more assets to be profiled. For example, if an HBase schedule has about 80 assets to be profiled, the first 50 assets would be profiled in the first Livy batch and the other assets get profiled in the next batch.

In case of Ranger Audit profiling, there could be a “NA” status for the total number of assets profiled. It indicates that the auditing that happens is dependent on the Ranger policies. In other words, the Ranger policies are actually profiled and not the assets.

Viewing profiler configurations

You can monitor the last status of individual profilers by viewing them in Profiler > Configs. Also, you can change their resources, sensitivity and scheduling.

Profilers / Configs

Jobs

Configs

Profiler Configuration

Name	Last Run Time	Last Run Status	Next Scheduled Run	Config Version	Status
Ranger Audit Profiler	NA	NA	Tomorrow at 12:00 AM (UTC)	1	Active
Cluster Sensitivity Profiler	an hour ago	SUCCESS	Today at 4:03 PM (UTC)	9	Active
Hive Column Profiler	NA	NA	Tomorrow at 12:00 AM (UTC)	2	Active

Monitoring the profiler configurations has the following uses:

- Verify which profilers are active or inactive.
- Verify the status of the profiler runs.

- View the last run time and status and the next scheduled run.



Note: You can also filter your profilers by job status, type for the last day, week and month.

Ranger Audit Profiler configuration

In addition to the generic configuration, there are additional parameters for the Ranger Audit Profiler that can optionally be edited. You can configure the scheduling and the available resources for your profiler.

Procedure

1. Go to **Profilers** and select your data lake.
2. Go to **Profilers Configs**.
3. Select Ranger Audit Profiler.
The **Detail** page is displayed.
- 4.



Use the toggle button to enable or disable the profiler.

5. Select a schedule to run the profiler using a quartz cron expression.
6. Set the **Input block size**.



Note: The minimum allowed value is 100000, the maximum is 10000000.

7. Continue with the **Pod Configurations** and set the Kubernetes job resources:

Pod configurations specify the resources that would be allocated to a pod when the profiler job starts to run. As all profilers are submitted as Kubernetes jobs, you must decide if you want to add or reduce resources to handle workload of various sizes.

- **Pod CPU Limit:** Indicates the maximum number of cores that can be allocated to a Pod. The accepted values range from one through eight.
- **Pod CPU Requirement:** This is the minimum number of CPUs that will be allocated to a pod when its provisioned. If the node where a pod is running has enough resources available, it is possible (and allowed) for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from one through eight.
- **Pod Memory Limit:** The maximum amount of memory can be allocated to a Pod. The accepted values range from 1 through 256.
- **Pod Memory Requirement:** This is the minimum amount of RAM that will be allocated to a pod when it is provisioned. If the node where a pod is running has enough resources available, it is possible (and allowed) for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from 1 through 256.

8. In **Executor Configurations**, update the following:

Executor configurations are the runtime configurations. These configuration must be changed if you are changing the pod configurations and when there is a requirement for additional compute power.

- **Number of workers:** Indicates the number of processes that are used by the distributed computing framework. The accepted values range from one through eight.
- **Number of threads per worker:** Indicates the number of threads used by each worker to complete the job. The accepted values range from one through eight.

- **Worker Memory limit in GB:** To avoid over utilization of memory, this parameter forces an upper threshold memory usage for a given worker. For example, if you have a 8 GB Pod and 4 threads, the value of this parameter must be 2 GB. The accepted values range from one through four.
9. Click Save to apply the configuration changes to the selected profiler.

Related Information

[The Ranger Audit Profiler](#)

[Understanding the Cron Expression generator](#)

Hive Column Profiler configuration

In addition to the generic configuration, there are additional parameters for the Hive Column Profiler that can optionally be edited.

Procedure

1. Go to Profilers Configs .
2. Select Hive Column Profiler.
The **Detail** page is displayed.
- 3.



Use the toggle button to enable or disable the profiler.

4. Select a schedule to run the profiler. This is implemented as a quartz cron expression.
For more information, see [Understanding the Cron Expression generator](#) on page 24.
5. Select Last Run Check and set a period



Note:

The Last Run Check enables profilers to avoid profiling the same asset on each scheduled run.

If you have scheduled a cron job, for example set to start in about an hour, and have enabled the Last Run Check configuration for two days, this setup ensures that the job scheduler filters out any asset which was already profiled in the last two days.

If the Last Run Check configuration is disabled, assets will be picked up for profiling as per the job cron schedule, honoring the asset filter rules.

6. Set the sampling configurations. When an asset or table is profiled, instead of scanning the whole table, the profiler sample selects only a subset of records as it finds them.
 - a) Set the Sample Count: Indicates the number of times a table must be sampled for profiling. A value less than 3 and higher than 30 is not recommended.
 - b) Set the Sample Factor: Controls the randomization of records. Less value promote better random samples and higher values results in poor samples. A value 0.001 indicates that the data that is retrieved from Hive and a new random number is generated. If the value is less than or equal to the provided proportion (0.001), it will be chosen in the result set. If the value is greater, it is ignored. The accepted values range from 0,001 through 0,5.
 - c) Set the Sample Records: Indicates the number of records to be retrieved in a given sample. Consider this as LIMIT clause of the SQL query. The accepted values range from 100 through 100000.

7. Continue with the **Pod Configurations** and set the Kubernetes job resources:

Pod configurations specify the resources that would be allocated to a pod when the profiler job starts to run. As all profilers are submitted as Kubernetes jobs, you must decide if you want to add or reduce resources to handle workload of various sizes.

- **Pod CPU limit:** Indicates the maximum number of cores that can be allocated to a Pod. The accepted values range from one through eight.
- **Pod CPU Requirements:** This is the minimum number of CPUs that will be allocated to a Pod when its provisioned. If the node where a Pod is running has enough resources available, it is possible (and allowed) for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from one through eight.
- **Pod Memory limit:** Maximum amount of memory can be allocated to a Pod. The accepted value format examples are: The accepted values range from 1 through 256.
- **Pod Memory Requirements:** This is the minimum amount of RAM that will be allocated to a Pod when it is provisioned. If the node where a Pod is running has enough resources available, it is possible (and allowed) for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from 1 through 256.

8. In **Executor Configurations**, update the following:

Executor Configurations are the runtime configuration. These configuration must be changed if you are changing the Pod configurations and when there is a requirement for additional compute power.

- **Number of workers:** Indicates the number of processes that are used by the distributed computing framework. The accepted values range from one through eight.
- **Number of threads per worker:** Indicates the number of threads used by each worker to complete the job. The accepted values range from one through eight.
- **Worker Memory limit in GB:** To avoid over utilization of memory, this parameter forces an upper threshold memory usage for a given worker. For example, if you have a 8 GB Pod and 4 threads, the value of this parameter must be 2 GB. The accepted values range from one through four.

9. Add **Asset Filter Rules** as needed to customize the selection and deselection of assets which the profiler profiles.

a) Set your **Deny List** and **Allow-list**.

The profiler will skip profiling assets that meet any criteria in the **Deny List** and will include assets that meet any criteria in the **Allow List**.

1. Select the **Deny-list** or **Allow List** tab.
2. Click Add New to define new rules.
3. Select the key from the drop-down list. You can select from the following:
 - Database name
 - Asset name
 - Asset owner
 - Path to the asset
 - Created date
4. Select the operator from the drop-down list. Depending on the keys selected, you can select an operator such as equals, contains. For example, you can select the name of assets that contain a particular string.
5. Enter the value corresponding to the key. For example, you can enter a string as mentioned in the previous example.
6. Click Done. Once rule is added, you can toggle the state of the new rule to enable it or disable it as needed.

10. Click Save to apply the configuration changes to the selected profiler.

Related Information

[The Hive Column Profiler](#)

[Understanding the Cron Expression generator](#)

Cluster Sensitivity Profiler configuration

In addition to the generic configuration, there are additional parameters for the Cluster Sensitivity Profiler that can optionally be edited. You can configure the scheduling and the available resources for your profiler.

Procedure

1. Go to **Profilers** and select your data lake.
2. Go to **Profilers Configs**.
3. Select Cluster Sensitivity Profiler.
The **Detail** page is displayed.
- 4.



Use the toggle button to enable or disable the profiler.

5. Select a schedule to run the profiler. This is implemented as a quartz cron expression.
For more information, see [Understanding the Cron Expression generator](#) on page 24.
6. Select Last Run Check and set a period if needed.



Note:

The Last Run Check enables profilers to avoid profiling the same asset on each scheduled run.

If you have scheduled a cron job, for example set to start in about an hour, and have enabled the Last Run Check configuration for two days, this setup ensures that the job scheduler filters out any asset which was already profiled in the last two days.

If the Last Run Check configuration is disabled, assets will be picked up for profiling as per the job cron schedule, honoring the asset filter rules.

7. Set the sampling configurations. When an asset or table is profiled, instead of scanning the whole table, the profiler sample selects only a subset of records as it finds them.
 - a) Set the Sample Count: Indicates the number of times a table must be sampled for profiling. A value less than 3 and higher than 30 is not recommended.
 - b) Set the Sample Factor: Controls the randomization of records. Less value promote better random samples and higher values results in poor samples. A value 0.001 indicates that the data that is retrieved from Hive and a new random number is generated. If the value is less than or equal to the provided proportion (0.001), it will be chosen in the result set. If the value is greater, it is ignored. The accepted values range from 0,001 through 0,5.
 - c) Set the Sample Records: Indicates the number of records to be retrieved in a given sample. Consider this as LIMIT clause of the SQL query. The accepted values range from 100 through 100000.
8. Continue with the **Pod Configurations** and set the Kubernetes job resources:

Pod configurations specify the resources that would be allocated to a pod when the profiler job starts to run. As all profilers are submitted as Kubernetes jobs, you must decide if you want to add or reduce resources to handle workload of various sizes.

 - Pod CPU limit: Indicates the maximum number of cores that can be allocated to a Pod. The accepted values range from one through eight.
 - Pod CPU Requirements: This is the minimum number of CPUs that will be allocated to a Pod when its provisioned. If the node where a Pod is running has enough resources available, it is possible (and allowed)

for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from one through eight.

- **Pod Memory limit:** Maximum amount of memory can be allocated to a Pod. The accepted value format examples are: The accepted values range from 1 through 256.
- **Pod Memory Requirements:** This is the minimum amount of RAM that will be allocated to a Pod when it is provisioned. If the node where a Pod is running has enough resources available, it is possible (and allowed) for a container to use more resource than its request for that resource specifies. However, a container is not allowed to use more than its resource limit. The accepted values range from 1 through 256.

9. In **Executor Configurations, update the following:**

Executor Configurations are the runtime configuration. These configuration must be changed if you are changing the Pod configurations and when there is a requirement for additional compute power.

- **Number of workers:** Indicates the number of processes that are used by the distributed computing framework. The accepted values range from one through eight.
- **Number of threads per worker:** Indicates the number of threads used by each worker to complete the job. The accepted values range from one through eight.
- **Worker Memory limit in GB:** To avoid over utilization of memory, this parameter forces an upper threshold memory usage for a given worker. For example, if you have a 8 GB Pod and 4 threads, the value of this parameter must be 2 GB. The accepted values range from one through four.

10. Add **Asset Filter Rules as needed to customize the selection of assets to be profiled.**

a) Set your **Deny List** and **Allow-list**.

The profiler will skip profiling assets that meet any criteria in the **Deny List** and will include assets that meet any criteria in the **Allow List**.

1. Select the **Deny-list** or **Allow List** tab.
2. Click Add New to define new rules.
3. Select the key from the drop-down list. You can select from the following:
 - Database name
 - Asset name
 - Asset owner
 - Path to the asset
 - Created date
4. Select the operator from the drop-down list. Depending on the keys selected, you can select an operator such as equals, contains. For example, you can select the name of assets that contain a particular string.
5. Enter the value corresponding to the key. For example, you can enter a string as mentioned in the previous example.
6. Click Done. Once rule is added, you can toggle the state of the new rule to enable it or disable it as needed.

11. Click Save to apply the configuration changes to the selected profiler.

Related Information

[The Cluster Sensitivity Profiler](#)


[Understanding the Cron Expression generator](#)

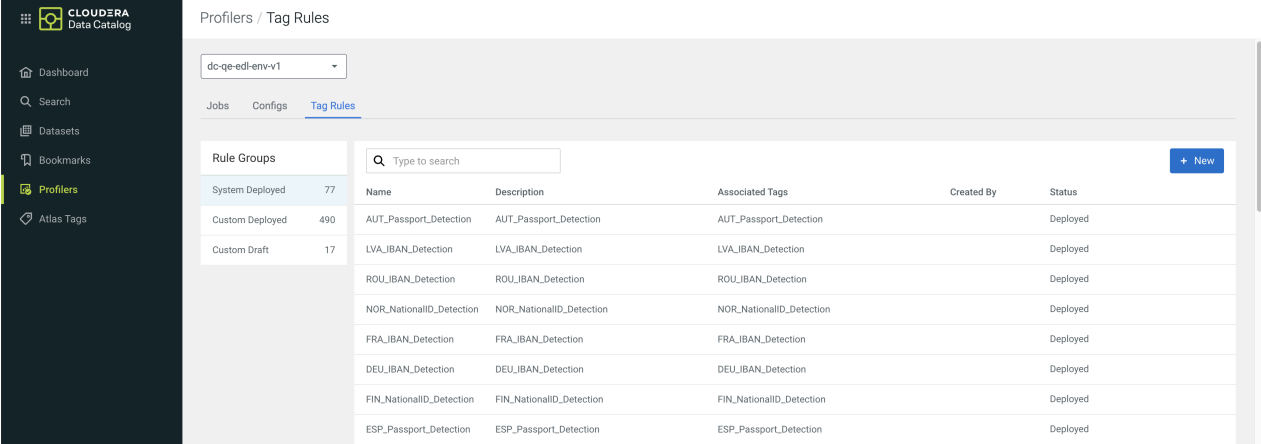
Profiler tag rules

You can use preconfigured tag rules or create new rules based on regular expressions and values in your data to limit the number of assets to be profiled by the Cluster Sensitivity Profiler. When a tag rule is matching your data, the selected Apache Atlas classification (also known as a Cloudera Data Catalog tag) is applied. This way you can save compute resources instead of running the profiler on the whole of your data.

Tag rule types

Tag Rules are categorized based on their type into the following groups:

- **System Deployed:** These are built-in rules that cannot be edited. You can only enable or disable them for your data.
- **Custom Deployed:** Tag rules that you create, edit and deploy on clusters after validation will appear under this category. Hover your mouse over the tag rules to deploy or suspend them as needed. Click the  icon in the **Action** column to enable your custom tag rules. You can also edit these tag rules.
- **Custom Draft:** You can create new tag rules and save them for later validation and deployment on clusters. Such rules appear under this category.

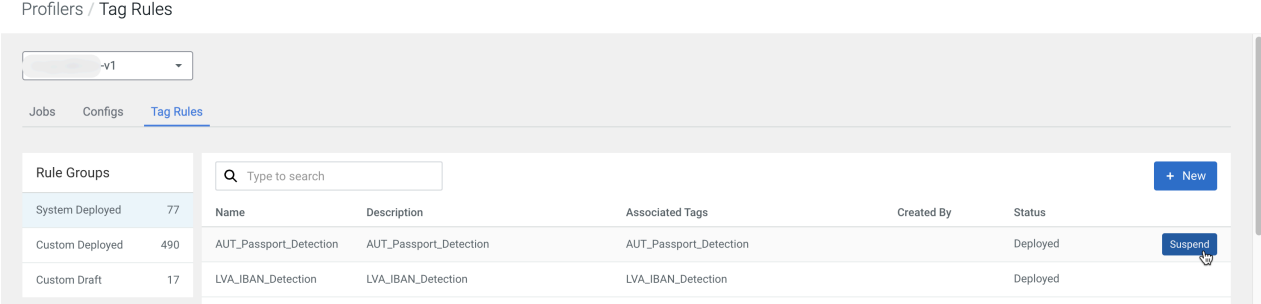


Rule Groups	Name	Description	Associated Tags	Created By	Status
System Deployed 77	AUT_Passport_Detection	AUT_Passport_Detection	AUT_Passport_Detection		Deployed
Custom Deployed 490	LVA_IBAN_Detection	LVA_IBAN_Detection	LVA_IBAN_Detection		Deployed
Custom Draft 17	ROU_IBAN_Detection	ROU_IBAN_Detection	ROU_IBAN_Detection		Deployed
	NOR_NationalID_Detection	NOR_NationalID_Detection	NOR_NationalID_Detection		Deployed
	FRA_IBAN_Detection	FRA_IBAN_Detection	FRA_IBAN_Detection		Deployed
	DEU_IBAN_Detection	DEU_IBAN_Detection	DEU_IBAN_Detection		Deployed
	FIN_NationalID_Detection	FIN_NationalID_Detection	FIN_NationalID_Detection		Deployed
	ESP_Passport_Detection	ESP_Passport_Detection	ESP_Passport_Detection		Deployed

After creating your rule, you have to validate them with test data and, then Deploy them from **Custom Draft**.



Note: Tag Rules can be temporarily suspended.



Rule Groups	Name	Description	Associated Tags	Created By	Status
System Deployed 77	AUT_Passport_Detection	AUT_Passport_Detection	AUT_Passport_Detection		Deployed
Custom Deployed 490	LVA_IBAN_Detection	LVA_IBAN_Detection	LVA_IBAN_Detection		Deployed
Custom Draft 17					

Related Information

[Atlas tag management](#)

Creating custom profiler rules

You can create a custom profiler by adding the required tags, regex entries to specific columns within your tables. You can evaluate the tags and regex entries to Boolean values.

Procedure

1. On the **Profilers** page, click Tag Rules.
2. On the Tag Rules tab, click New to create a new profiler tag rule.
3. Enter the name of the new custom profiler tag rule.
4. Enter the description for the custom tag rule.

5. Select the tags to apply to assets which match your regex expressions. You can select tags from the drop-down list and or enter a new value to create a new tag.

New tags that you create here are added with a `dp_` prefix in the list of Atlas tags. For example, if you add a new tag called `credit_card`, this tag will be added as `dp_credit_card` in Apache Atlas.

6. Enter the rule for the column name. As you enter the values, regex name and resource names are auto populated. Select the column that is needed for your custom profiler.
7. Enter the column value for the DSL.

Based on your entry, Cloudera Data Catalog auto-populates values from the entries already available in the **Resources** tab. For more information about behaviors, see [Using DSL Grammar](#).

8. Click Save and Validate.

Data Catalog / Profilers

Custom Rule

Name *

Description

Tags *

Column Name Expression

Column Value Expression *

Resources

Regex Q +

SampleRegex_1580209003967
SampleRegex_1.58020939186e+12
DeployRegex1580209681238
SampleRegex_1.58020999412e+12
SampleRegex_1.58021014275e+12
SampleRegex_1.58021014308e+12
DeployRegex1580210288950
SampleRegex_1580276618318
SampleRegex_1580277217453

In the validation pop up window that appears, enter data to validate your custom profiler tag rule. Make sure you separate each data entry with a new line.

9. Click Save to create a tag rule and validate and deploy it later.

Adding custom regular expressions

To use custom regex entries within your new custom profiler tag rules, you can also add new regex values. If a match is found, the expression is evaluated to True.

Procedure

1. Go to Profilers Tag Rules .
2. Click New.
3. In the **Resources** in the panel on the **Custom Profiler** page, click +. The **Regular Expression Editor** page appears.
4. Enter the name of the new regular expression.

5. Enter a valid regular expression.

For example:

```
\b((([a-zA-Z0-9_-\.]+)@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.))|(([a-zA-Z0-9_-]+\.)+))([a-zA-Z]{2,4}|[0-9]{1,3})\b)?\b
```

6. Enter the list of test strings to evaluate the new regular expression.

If the test string is valid, then the match information gets auto populated in the **Match Information** box.

7. Click Save to add the new regular expression to the list of regex Resources.

Using DSL grammar

Using Domain-specific Language (DSL) grammar, you can combine different regex expressions in intuitive ways to bring out new functionality while creating custom profiler rules.

The two behaviors available in this framework are the following:

1. `falseIdentity` - Always evaluates to false, regardless of the input.
2. `trueIdentity` - Always evaluates to true, regardless of the input.

These two behaviors are used in the following examples and descriptions.

Binary AND operator

Keyword: AND

AND works the same way it does other languages. Hence following observations.

```
falseIdentity and trueIdentity == falseIdentity
```

```
falseIdentity and falseIdentity == falseIdentity
```

```
trueIdentity and trueIdentity == trueIdentity
```

```
trueIdentity and falseIdentity == falseIdentity
```

Here we are using `==` to show their equality.

Binary OR operator

The OR operator works the same way it does in other languages.

```
falseIdentity or trueIdentity == trueIdentity
```

```
falseIdentity or falseIdentity == falseIdentity
```

```
trueIdentity or trueIdentity == trueIdentity
```

```
trueIdentity or falseIdentity == trueIdentity
```

Expand DSL to use as follows.

```
val rule1= falseIdentity and trueIdentity and trueIdentity
```

```
val rule2= trueIdentity and trueIdentity and trueIdentity
```

```
val rule3=rule1 and rule2
```

```
rule3 or trueIdentity
```

The above expression evaluates to true.

Unary NOT operator

The NOT operator negates the value of a regex expression (from TRUE to FALSE and vice versa).

```
not(falseIdentity) == trueIdentity
```

```
not(trueIdentity) == falseIdentity
```

Use the not operator as follows.

```
val rule1= falseIdentity and trueIdentity and trueIdentity
```

```
val rule2= trueIdentity and trueIdentity and trueIdentity
```

```
val rule3=rule1 and rule2
```

```
val rule4=rule3 or trueIdentity
```

```
rule4 and not(falseIdentity)
```

The above expression evaluates to true.

Understanding the Cron Expression generator

In the Profiler > Configs > Detail page, a cron expression defines when the profiler schedule executes and visualizes the next execution dates of your profiling jobs.

The Unix cron expression uses the following typical format:

Each * in the cron represents a unique value.


Cron Expression: 0 18 * * *

In this format the * characters represent the following units:Minute hour day(month) month day(week)

For example, consider a cron with the following values:

CRON Expression: 30 10 15 5 *


This cron expression is scheduled to run the profiler job at: “At 10:30 on 15th day-of-month in May.”

 **Note:** The * character is a replacement for the "day-of-the-week". It is not specified on which day of the week the job has to run.

Consider another example:

30 10 * 5 7

This cron expression is scheduled to run the profiler job at: “At 10:30 on Sunday in May”.

 **Note:** The * character is a replacement for the "day-of the month". It is not specified on which day of the month the job has to run.

You can change the value of cron as and when it is required depending on how you want to schedule your profiler job.

Related Information


[The Open Group Base Specifications Issue 8: crontab](#)

Enabling or disabling profilers

By default profilers are scheduled to run at every 24 hours at midnight UTC timezone.

Procedure

1. Go to Profilers Configs .
2. Select the profiler to proceed further.



Dashboard

Search

Datasets

Bookmarks

Profilers

Atlas Tags

Profilers / Configs

Jobs

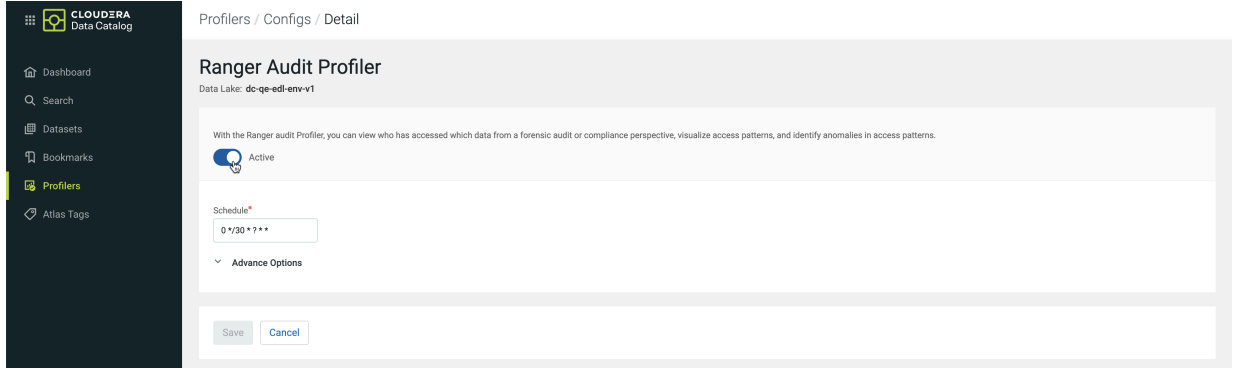
Configs

Tag Rules

Profiler Configuration

Name	Last Run Time	Last Run Status	Next Scheduled Run	Config Version	Status
Ranger Audit Profiler	09/12/2024 06:30 PM CEST	SUCCESS	09/12/2024 07:00 PM CEST	1	Active
Hive Column Profiler	09/12/2024 08:00 AM CEST	SUCCESS	09/12/2024 08:00 PM CEST	1	Active
Cluster Sensitivity Profiler	09/11/2024 06:20 PM CEST	SUCCESS	09/12/2024 07:20 PM CEST	1	Active

3. Switch the toggle to the desired state.



The screenshot shows the Cloudera Data Catalog interface. On the left is a dark sidebar with navigation links: Dashboard, Search, Datasets, Bookmarks, Profilers (highlighted), and Atlas Tags. The main content area is titled 'Ranger Audit Profiler' and shows the configuration for a specific Data Lake: 'dc-qe-edl-env-v1'. The page includes a description of the profiler's purpose, a toggle switch currently set to 'Active', a 'Schedule' field with a cron expression '0 */30 * * * *', and an 'Advance Options' section. At the bottom are 'Save' and 'Cancel' buttons.

CLUSTERA
Data Catalog

Profilers / Configs / Detail

Ranger Audit Profiler

Data Lake: dc-qe-edl-env-v1

With the Ranger audit Profiler, you can view who has accessed which data from a forensic audit or compliance perspective, visualize access patterns, and identify anomalies in access patterns.

☒ Active

Schedule*

0 */30 * * * *

▼ Advance Options

Save Cancel