Cloudera Data Engineering 1.5.5

Configuring Data Connectors

Date published: 2020-07-30
Date modified: 2025-11-08

https://docs.cloudera.com/

Legal Notice

Contents

Using Ozone storage with Cloudera Data Engineering on premises..... 4
Adding Ozone data connector for Cloudera Data Engineering service..... 4
Deleting Ozone data connector for Cloudera Data Engineering service..... 6
Backing up and Restoring Data Connectors..... 6
Backing up the data connector..... 6

Using Ozone storage with Cloudera Data Engineering on premises

Apache Ozone is an object store available on the Cloudera Base on premises cluster which enables you to optimize storage for big data workloads. You can configure Ozone as the backend storage for workloads of Cloudera Data Engineering clusters.

Data connectors enable you to access different storage using only a few configurations specific to storage. Data Connectors are bound to a Cloudera Data Engineering service. Hence, you must first create a Cloudera Data Engineering service before configuring Ozone storage through Data Connectors. All the virtual clusters managed by that Cloudera Data Engineering service can use the same data connectors.

Related Information

Adding a Cloudera Data Engineering service

Adding Ozone data connector for Cloudera Data Engineering service

You can configure Ozone as the backend storage for Cloudera Data Engineering workloads.

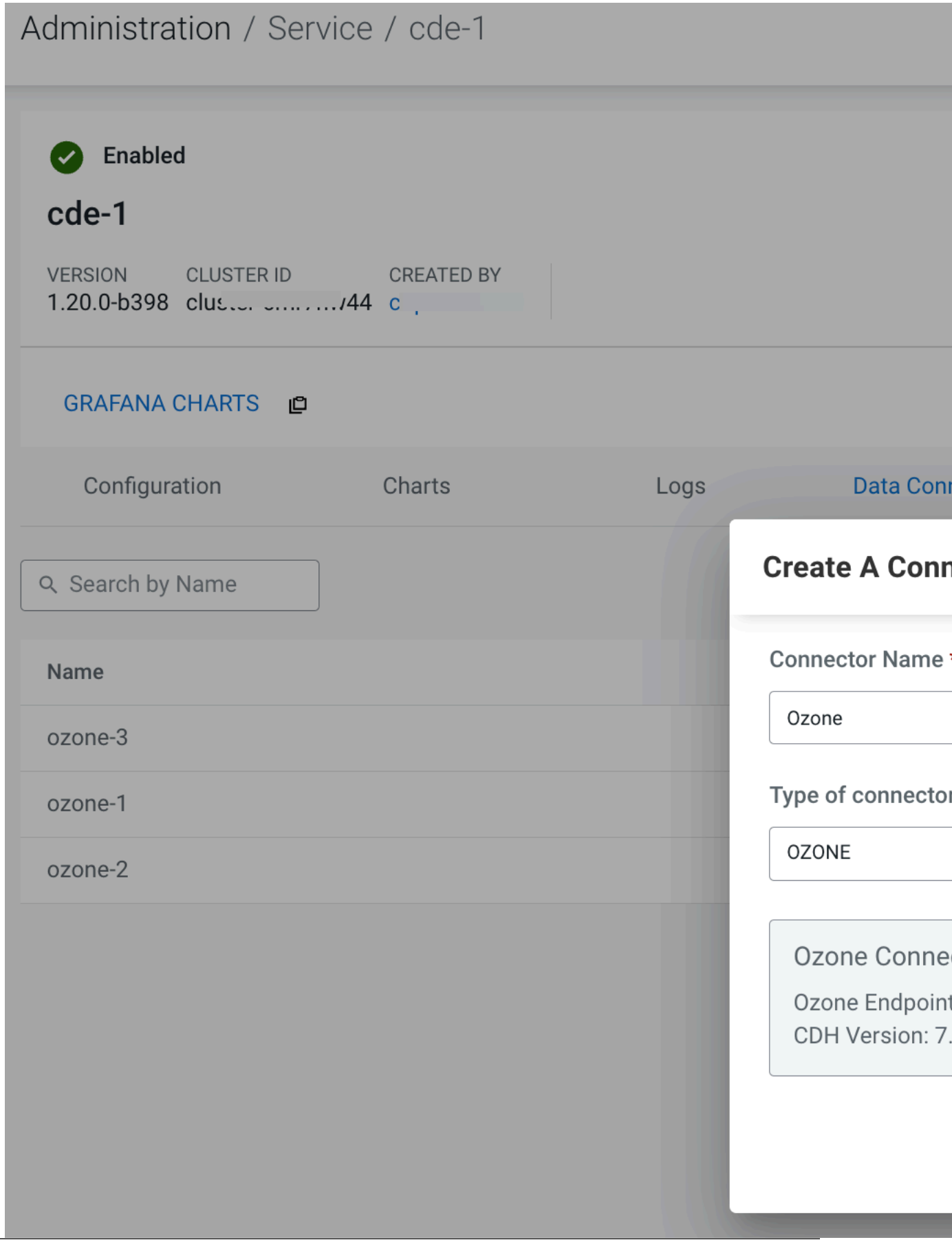
Before you begin

Data connectors are bound to a Cloudera Data Engineering service and you must first create a Cloudera Data Engineering service before configuring Ozone storage.

Procedure

1. In the Cloudera console, click the Data Engineering tile. The Cloudera Data Engineering Home page displays.
2. Click Administration in the left navigation menu. The Administration page displays.
3. In the Services column, click  for the Cloudera Data Engineering service you want to configure for accessing the Ozone file system.
4. Click the Data Connectors tab. On the Data Connectors tab, you can view details about the existing data connectors, if any, including the Cloudera Data Engineering service name, URL for the storage, CDH Version, and the Cloudera environment.
5. Click Create A Connector. The Create A Connector dialog box is displayed.
6. Specify the connector's name.

7. Select Ozone from the Type of Connectordrop-down list.



8. Review the summary and click Create Connector.

Results

After configuring the Cloudera Data Engineering service to access the Ozone filesystem, use this data connector during job creation to read and write from Apache Ozone Object Store through your workload Spark job.

Related Information

Adding a Cloudera Data Engineering service

Creating jobs in Cloudera Data Engineering

Deleting Ozone data connector for Cloudera Data Engineering service

You can delete an existing data connector.

Procedure

1. In the Cloudera console, click the Data Engineering tile. The Cloudera Data Engineering Home page displays.
2. Click Administration in the left navigation menu. The Administration page displays.
3. In the Services column, click icon for the Cloudera Data Engineering service you want to configure for accessing the Ozone file system.
4. Click the Data Connectors tab. On the Data Connectors tab, you can view details about the existing data connectors, if any, including the service name and Cloudera environment.
5. Click in the Actions column next to the data connector, and then click Delete.
6. In the Delete Connector dialog box, click Confirm to delete the data connector.

Backing up and Restoring Data Connectors

Before you upgrade your clusters or delete an existing cluster to create a new cluster, you can back up the data connectors and restore those data connectors post-upgrade on the new cluster. The data connector pre-upgrade jobs work seamlessly on the upgraded cluster after you restore those jobs.

Backing up the data connector

You must back up your data connectors before you delete or upgrade your cluster.

Before you begin

Install `jq` on the host machine where you want to run the backup and restore commands.

Procedure

1. Get the Cloudera Data Engineering Service URL.
   a. In the Cloudera console, click the Data Engineering tile. The Cloudera Data Engineering Home page displays.
   b. In the Services column, click for the Cloudera Data Engineering service.
   c. Click on GRAFANA CHARTS and get the domain name.
   d. Note the URL. For example, if the URL copied is *HTTPS://SERVICE.CDE-N82XSQPR.APPS.APPS.SHARED-RKE-DEV-01.KCLOUD.EXAMPLE.COM/GRAFANA/D/SK1XDUSZZ/KUBERNETES*, use only the *HTTPS://SERVICE.CDE-N82XSQPR.APPS.APPS.SHARED-RKE-DEV-01.KCLOUD.EXAMPLE.COM* as the Cloudera Data Engineering Service URL.

2. Get the Cloudera Data Engineering token for your Cloudera Data Engineering service URL.

```
export CDE_TOKEN=$(curl -k -L -u <user-name>:<password> https://<cde ser
vice url>/gateway/authtkn/knoxtoken/api/v1/token | jq -r .access_token)
```

Example:

```
export CDE_TOKEN=$(curl -k -L -u dexssoadmin:Password@123 https://servic
e.cde-4x6s29mx.apps.apps.shared-os-qe-01.kcloud.cloudera.com/gateway/aut
htkn/knoxtoken/api/v1/token | jq -r .access_token)
```

3. Note the cluster-ID.
   a. In the Cloudera console, click the Data Engineering tile. The Cloudera Data Engineering Home page displays.
   b. In the Services column, click  for the Cloudera Data Engineering service.
   c. Click on GRAFANA CHARTS and get the domain name.
   d. Note the Cluster ID shown on the page.
4. Provide the Cloudera Data Engineering token and cluster-ID and run the data connector backup command.

```
curl --location --request POST 'https://<cde service url>/data-connectors/
v1/backupDataConnections' \
--header "Authorization: Bearer ${CDE_TOKEN}" \
--data-raw '{
  "clusterId": "<cluster id of cluster that needs backing up>"
}' --insecure | jq .zipFile | tr -d '"'  | base64 -d > backup-dataconnect
ors.zip
```

Example:

```
curl --location --request POST 'https://service.cde-4x6s29mx.apps.apps.s
hared-os-qe-01.kcloud.cloudera.com/data-connectors/v1/backupDataConnecti
ons' \
--header "Authorization: Bearer ${CDE_TOKEN}" \
--data-raw '{
  "clusterId": "cluster-4x6s29mx"
}' --insecure | jq .zipFile | tr -d '"'  | base64 -d > backup-dataconnect
ors.zip
```

Related Information

Installing jq