Cloudera DataFlow for Data Hub 7.3.1

Setting up your Streams Messaging cluster

Date published: 2019-12-16
Date modified: 2024-12-10

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

Checking prerequisites

Before you start creating your first Streams Messaging Data Hub Cluster, you need to ensure that you have set up the environment properly and have all the necessary accesses to use CDP Public Cloud. Use this checklist to make sure that you meet all the requirements before you start creating your first cluster.

- You have CDP login credentials.
- You have an available CDP environment.
- You have a running Data Lake.
- You have a CDP username and the predefined resource role of this user is EnvironmentAdmin.
- Your CDP user is synchronized to the CDP Public Cloud environment.
- Your CDP user has the correct permissions set up in Ranger allowing access to Kafka, Streams Messaging Manager and Schema Registry.

⚠ **Important:** Make sure that the Runtime version of the Data Lake cluster matches the Runtime version of the Data Hub cluster that you are about to create. If these versions do not match, you may encounter warnings and/or errors.

Related Information

Getting started as a user

AWS environments

Azure environments

GCP environments

Data lakes

Creating your cluster

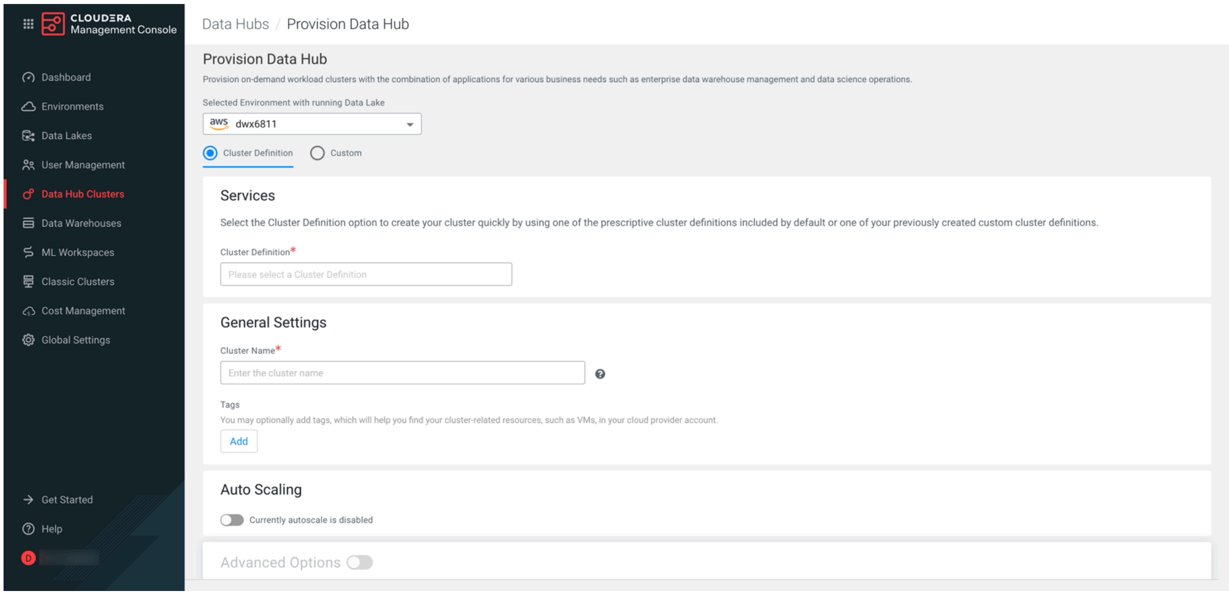
Once you have met the prerequisites, you are ready to create a managed and secured Streams Messaging cluster in CDP Public Cloud by using one of the prescriptive Streams Messaging cluster definitions.

Procedure

1. Log into the CDP web interface.
2. Navigate to Management Console Environments, and select the environment where you would like to create a cluster.

3. Click Create Data Hub.

   The following page is displayed:



Provision Data Hub

Provision on-demand workload clusters with the combination of applications for various business needs such as enterprise data warehouse management and data science operations.

Selected Environment with running Data Lake: dwx6811

● Cluster Definition ○ Custom

Services

Select the Cluster Definition option to create your cluster quickly by using one of the prescriptive cluster definitions included by default or one of your previously created custom cluster definitions.

Cluster Definition*: Please select a Cluster Definition

General Settings

Cluster Name*: Enter the cluster name

Tags

You may optionally add tags, which will help you find your cluster-related resources, such as VMs, in your cloud provider account.

Auto Scaling

Currently autoscale is disabled

4. Select Cluster Definition.
5. Select the appropriate Streams Messaging cluster definition from the Cluster Definition dropdown depending on your operational objectives.

   The list of services is automatically shown below the selected cluster definition name. For more information on templates, see *Data Hub cluster definitions* and *Streams Messaging cluster layout*.
6. Give the cluster a name and add any tags you might need.

   You can define tags that will be applied to your cluster-related resources on your cloud provider account. For more information about tags, see Tags.
7. If using the Streams Messaging High Availability definitions, ensure that you select multiple subnets.

   Multiple subnets can be selected by navigating to Advanced Options Network And Availability .
8. Add Kafka KRaft to your cluster.

   Note: Kafka KRaft is available in this version of CDP but is not ready for production deployment. Cloudera encourages you to explore this technical preview feature in non-production environments and provide feedback on your experiences through the Cloudera Community Forums. For more information regarding KRaft limitations and unsupported features, see Unsupported Streams Messaging features.

   Kraft nodes are not provisioned by default in any of the Streams Messaging cluster definitions. If you want to run Kafka in KRaft mode, you must set the **Instance Count** of the **Kraft Nodes** host group to an odd number. An odd number of nodes is required because KRaft requires an odd number of instances to function. Cluster provisioning fails if the number of KRaft nodes is set to an even number.

   Note: While it is possible to run Kafka in KRaft mode with a single KRaft node, Cloudera recommends that you provision a minimum of three nodes to avoid having a single point of failure. Additionally, note that ZooKeeper service instances are still deployed even in KRaft mode. You can optionally delete ZooKeeper after the cluster is deployed.

   a) Go to Advanced Options Hardware and Storage .
   b) Locate the **Kraft Nodes** host group and click ✎ to edit host group details.
   c) Set the Instance Count to at an odd number
   d) Click ▤ to save your changes.

9. Add SRM or Kafka Connect to Heavy Duty clusters.

SRM and Kafka Connect nodes are not provisioned by default when using the Streams Messaging Heavy Duty definition. If you are using this definition and want to have SRM, Kafka Connect, or both deployed in the cluster, set the Instance Count of the Srm Nodes or Connect Nodes host group to at least one.

a) Go to Advanced OptionsHardware and Storage.

b) Locate the appropriate host group and click ✎ to edit host group details.

c) Set the Instance Count to at least 1.

> **Important:** If you want to deploy SRM in high availability mode, set the instance count of Srm Nodes to two or higher.

d) Click ▤ to save your changes.

10. Use the Advanced Options section to customize the infrastructure settings.

> **Important:** If you customize the hardware and storage properties of the cluster, ensure that Attached Volume per Instances is set to the same value for both the **Broker** and **Core_brokers** host groups. Alternatively, if you want to provision a cluster where the number of volumes is not identical for the two host groups, ensure that you complete *Configure data directories for clusters with custom disk configurations* after the cluster is provisioned. Otherwise, Kafka does not utilize all available volumes. Additionally, scaling the cluster might also not be possible.

11. Click Provision Cluster.

Results

You will be redirected to the Data Hub cluster dashboard, and a new tile representing your cluster will appear at the top of the page.

What to do next

What steps you take next depends on how you configured your cluster

- If you have provisioned KRaft nodes in your cluster, you can optionally choose to remove ZooKeeper from your cluster. For more information, see *Deleting ZooKeeper from Streams Messaging clusters*.
- If you configured Attached Volume per Instances and the volume count is not identical for the **Broker** and **Core_brokers** host groups, continue with *Configuring data directories for clusters with custom disk configurations*.
- If you did not customize Attached Volume per Instances or do not want to remove ZooKeeper, continue with *Give users access to your cluster*.

Related Information

Create a cluster from a definition on AWS

Create a cluster from a definition on Azure

Create a cluster from a definition on GCP

Tags

Data Hub cluster definitions

Streams Messaging cluster layout

Configuring data directories for clusters with custom disk configurations

Give users access to your cluster

Deleting ZooKeeper from Streams Messaging clusters

Deleting ZooKeeper from Streams Messaging clusters

The ZooKeeper instances deployed in Streams Messaging clusters can be manually deleted if the cluster uses KRaft for metadata management. Learn how to delete ZooKeeper from a Streams Messaging cluster.

About this task

If you use the default cluster definitions to deploy a Streams Messaging cluster, you have the option to provision the cluster with KRaft nodes. If KRaft nodes are deployed in a cluster, Kafka runs in KRaft mode and uses KRaft for metadata management instead of ZooKeeper. By default, clusters like this still include ZooKeeper instances. However, the ZooKeeper instances are not required and can be deleted. The ZooKeeper instances can be removed using Cloudera Manager.

Before you begin

Warning: Ensure that your cluster is provisioned with KRaft nodes. If you complete the following steps in a cluster that does not include KRaft nodes, the cluster will fail.

- Ensure that the cluster, its hosts, and all its services are healthy.
- Deleting the service does not remove the service data from your hosts. It only removes the service from management by Cloudera Manager. All role groups under the service are removed from host templates.

Procedure

1. Access the Cloudera Manager instance managing the cluster.
2. Remove ZooKeeper service dependencies:
   a) On the Cloudera Manager **Home** page, go to Configuration Service Dependencies .
   b) Locate and clear the ZooKeeper Service property.
   c) Click Save Changes.
3. Go to Clusters and select the ZooKeeper service.
4. Stop the ZooKeeper service:
   a) Select Actions Stop .
   b) Click Stop to confirm the action.
   c) Click Close after the process is finished.
5. Delete the ZooKeeper service:
   a) Select Actions Delete .
   b) Click Delete to confirm the action.

Results

The ZooKeeper service is deleted from the cluster.

What to do next

- If you configured Attached Volume per Instances and the volume count is not identical for the **Broker** and **Core_brokers** host groups, continue with *Configuring data directories for clusters with custom disk configurations*.
- If you did not customize Attached Volume per Instances, or customized it in a way that the volume count remained identical for the **Brokers** and **Core_brokers** host groups, continue with *Give users access to your cluster*.

Related Information

Give users access to your cluster

Configuring data directories for clusters with custom disk configurations

Configuring data directories for clusters with custom disk configurations

Data Hub clusters provisioned with the default Streams Messaging cluster definitions that have non-identical volume numbers configured for the Broker and Core_brokers host groups require additional configuration after cluster provisioning. If configuration is not done, the Kafka service deployed on these clusters does not utilize all available volumes. Additionally, scaling the cluster might not be possible.

About this task

Clusters provisioned with version 7.2.12 or later of the Streams Messaging cluster definitions have two host groups that contain Kafka brokers (broker type groups). These are as follows:

- **Core_brokers**: This host group is provisioned by default, it contains a core set of brokers, and is not scalable.
- **Broker**: This host group is not provisioned by default (instance count is 0), but is scalable.

By default both broker type groups have the same disk configuration. The disk configuration can be changed during cluster provisioning in Advanced ConfigurationHardware and Storage. If you customize the Attached Volume per Instance property and the volume numbers set for the **Broker** and **Core_broker** groups are not identical, manually configuring the data directories (log.dirs) that Kafka uses is required after the cluster is provisioned.

If configuration is not done, Kafka does not utilize all volumes that are available. For example, assume you set Attached Volume per Instances to 2 for the **Broker** host group and 3 for the **Core_brokers** host group. In a case like this, Kafka by default utilizes two out of the three volumes. That is, it will utilize the lower number of volumes out of the two host group configurations. Additionally, scaling the cluster might also not be possible.

Important: Configuration is required even if you provision the cluster with no **Broker** host group. This is needed to avoid issues in case the **Broker** host group is upscaled at a later point in time.

Procedure

1. Access the Cloudera Manager instance managing the cluster.
2. Select the Kafka service.
3. Configure the Data Directories property for the default Kafka Broker role group.
   a) Go to Configuration.
   b) Find and configure the Data Directories property.

   The Data Directories property for the default role group must reflect the disk configuration (number of volumes) of the **Broker** host group. The property is configured by adding a unique data directory for each of the volumes. Each unique directory must be added as a separate entry. New entries can be added by clicking the + (add) icon next to the property. If configured correctly, the number of entries in the property should be identical with the number of volumes configured for the **Broker** host group.

   The data directories you specify must be /hadoopfs/fs*[***NUMBER***]*/kafka. For example, in case of two volumes, the data directories you need to specify are as follows:

   ```
   /hadoopfs/fs1/kafka
   /hadoopfs/fs2/kafka
   ```

   c) Click Save Changes.
4. Go to Instances.

5. **5.** Configure role level overrides for the Data Directories property.

   The following substeps must be repeated for each Kafka Broker role instance that is part of the **Core_brokers** host group. Ensure that the override you configure is identical for all Kafka Broker role instances.

   a) Select a Broker role instance that is deployed in the **Core_brokers** host group.

   b) Go to Configuration.

   c) Find and configure the Data Directories property.

      In this step you are configuring an override for the property. The value you set here is only be used by the role instance you have selected in substep 5.a on page 9.

      Configure the override so that it reflects the disk configuration (number of volumes) of the **Core_brokers** host group. The property override is configured by adding a unique data directory for each of the volumes. Each unique directory must be added as a separate entry. New entries can be added by clicking the + (add) icon next to the property. If configured correctly, the number of entries in the property should be identical with the number of volumes configured for the **Core_brokers** host group.

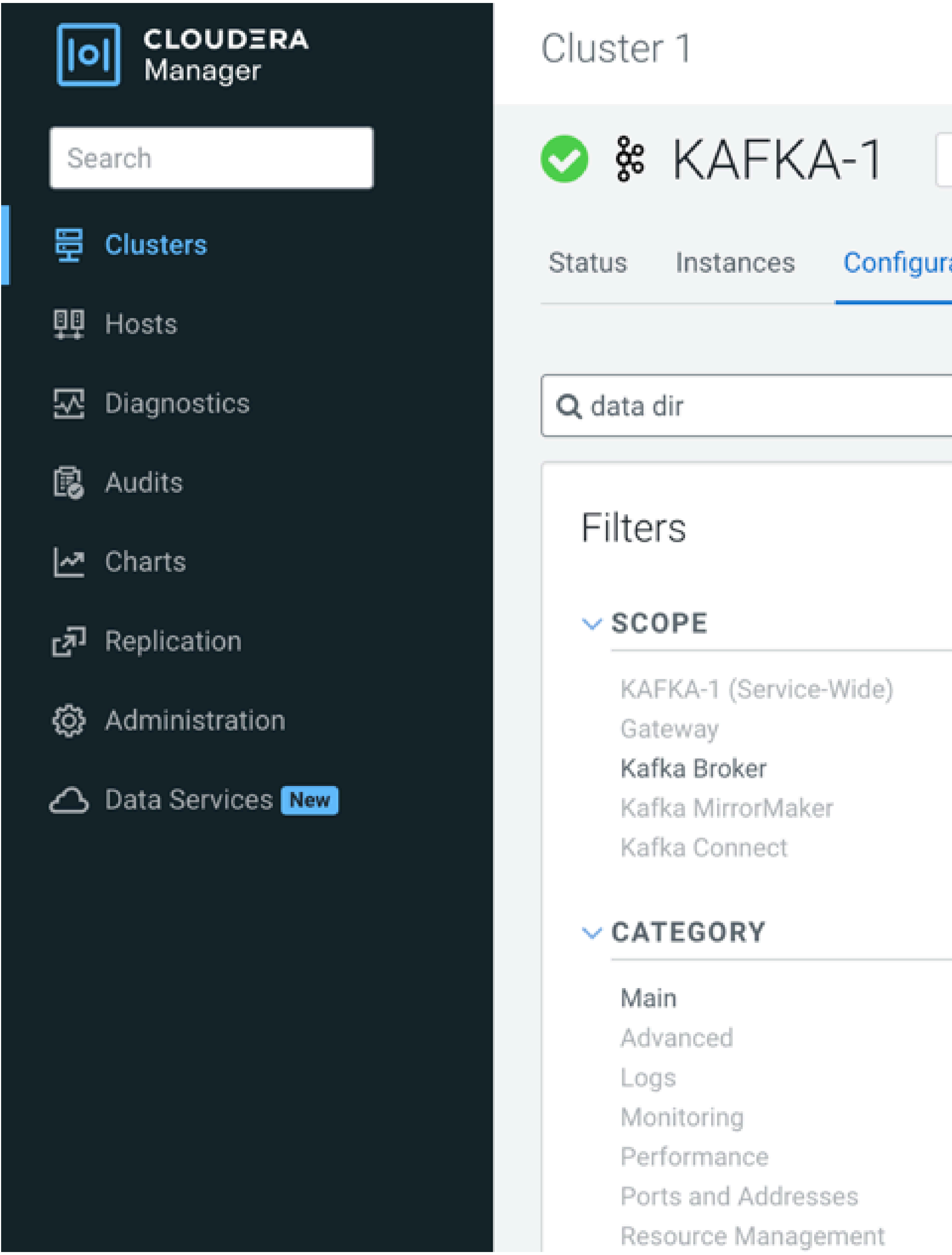
      The data directories you specify must be /hadoopfs/fs*[***NUMBER***]*/kafka. For example, in case of three volumes, the data directories you need to specify are as follows:

      ```
      /hadoopfs/fs1/kafka
      /hadoopfs/fs2/kafka
      /hadoopfs/fs3/kafka
      ```

   d) Click Save Changes.

6. **6.** Review and verify your configuration.

   After both the default role group configuration and the overrides are specified, navigate to the service level configuration page of Kafka and find the Data Directories property. The property is displayed with the full configuration including the overrides. For example, if **Core_brokers** had three volumes and **Broker** had two volumes, the correct configuration would be similar to the following example:



Results

Kafka data directories are configured. The Kafka service now uses all volumes available for the cluster.

What to do next

Continue with *Giving access to your cluster*.

Related Information

Give users access to your cluster

Giving access to your cluster

As an EnvironmentAdmin, you need provide the users access to your environment and to the new Streams Messaging cluster by assigning user roles and adding users to Ranger policies.

About this task

The cluster you have created using the Streams Messaging cluster definition is kerberized and secured with SSL. Users can access cluster UIs and endpoints through a secure gateway powered by Apache Knox. Before you can begin working with Kafka, Schema Registry, Streams Messaging Manager, and Streams Replication Manager, you must give users access to the Streams Messaging cluster components.

Procedure

1. Assign the EnvironmentUser role to the users to grant access to the CDP environment and the Streams Messaging cluster.
2. Sync the updated user permissions to the Data Lake (Ranger) using ActionsSynchronize Users to FreeIPA.

   This synchronizes the user to the FreeIPA identity management system to enable SSO.
3. Add the user to the appropriate pre-defined Ranger policies.

   These policies are specified within the Ranger instance that provides authorization to the Kafka service in your Streams Messaging cluster.

   If needed, you can also create a custom Ranger access policy and add the user to it.

What to do next

After you have created your Streams Messaging Data Hub cluster, you are ready to start working with Kafka workloads in CDP Public Cloud.

Related Information

Enable authorization in Kafka with Ranger

Configure the resource-based Ranger service used for authorization