Cloudera DataFlow for Data Hub 7.2.12

Ingesting Data into Apache HBase in CDP Public Cloud

Date published: 2019-12-16
Date modified: 2021-10-25

Legal Notice

© Cloudera Inc. 2024. 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

Ingesting Data into Apache HBase in CDP Cloud.....4
Understand the use case.....4
Meet the prerequisites.....4
Create the HBase target table.....5
Add Ranger policies.....5
Obtain HBase connection details.....6
Build the data flow.....7
Configure the HBase client service.....8
Configure the processor for your data source.....10
Configure the processor for your data target.....12
Start your data flow.....12
Verify your data flow.....13
Next steps.....13

Ingesting Data into Apache HBase in CDP Cloud

You can use an Apache NiFi data flow to ingest data into Apache HBase in the CDP Public Cloud following these steps.

Understand the use case

You can use Apache NiFi to move data from a range of locations into an Operational Database cluster running Apache HBase in CDP Public Cloud.

Today's scalable web applications for use cases like hotel or flight bookings as well as mobile banking applications are relying on an equally scalable database which can serve data at a very low latency. Cloudera's Operational Database in CDP Data Hub is powered by Apache HBase and provides application developers with everything they need to build scalable applications on top of it.

You can use Apache NiFi data flows into Apache HBase in a CDP Public Cloud Operational Database cluster to make sure that the applications you build on top always have access to the latest data.

This use case walks you through the steps associated with creating an ingest-focused data flow from Apache Kafka into HBase. If you are moving data from a location other than Kafka, review the *Getting Started with Apache NiFi* for information about how to build a data flow, and about other data `get` and `consume` processor options. You can also find instructions on building a data flow for ingest into other CDP components.

Related Information

Getting Started with Apache NiFi

Ingesting data into Apache Kafka in CDP Public Cloud

Ingesting Data into Apache Hive in CDP Public Cloud

Ingesting Data into Apache Kudu in CDP Public Cloud

Ingesting Data into Amazong S3 Buckets

Ingesting Data into Azure Data Lake Storage

Meet the prerequisites

Use this checklist to make sure that you meet all the requirements before you start building your data flow.

- You have a CDP Public Cloud environment.
- You have a workload username and password set to access Data Hub clusters. The predefined resource role of this user is at least "EnvironmentUser". This resource role provides the ability to view Data Hub clusters and set the FreeIPA password for the environment.
- Your CDP user is synchronized to the CDP Public Cloud environment.
- You have a Flow Management cluster running Apache NiFi.
- You have a Streams Messaging cluster running Apache Kafka, and have Kafka data available for consumption.
- You have an Operational Database cluster running Apache HBase and Phoenix.
- Your CDP user has the correct permissions set up in Ranger allowing access to the NiFi UI.

Related Information

AWS Onboarding Quickstart

Azure Onboarding Quickstart

Understanding roles and resources

Creating your first Flow Management Cluster in CDP Public Cloud

Create the HBase target table

Before you can ingest data into Apache HBase in CDP Public Cloud, ensure that you have an HBase target table. These steps walk you through creating a simple table with one column family. Modify these instructions based on your ingest target table needs.

Before you begin

Your CDP user has the necessary Ranger permissions to create an HBase table.

Procedure

1. Ssh into one of your HBase nodes, entering your CDP user name and password when prompted.

```
ssh <cdp-user.name><HBase.node.name>
```

2. Use klist to verify your Kerberos ticket, to ensure that the Ranger permissions you have setup to create HBase tables will work with your CDP user.
3. Open the HBase shell.

```
hbase shell
```

4. Create a new table.

   For example, to create a new table called customer, with one column family called customer_data, enter:

```
create 'customer', 'customer_data'
```

5. Verify that you have created your table.

```
list

TABLE
customer
1 row(s)
```

6. Create a Ranger policy that allows the CDP user to ingest data into this HBase table. You will use this CDP user later, when specifying credentials for your NiFi data flow.

Add Ranger policies

Add Ranger policies to ensure that you have write access to your HBase tables.

Procedure

1. Go to Ranger in your Data Lake, and select the cm_hbase service.
2. From the Ranger cm_hbase service in your Data Lake, create a new policy called customer-ingest that allows the `nifi-hbase-ingest` user to access the HBase table into which you want to ingest data.

   The policy should ensure that the specified user has access to:

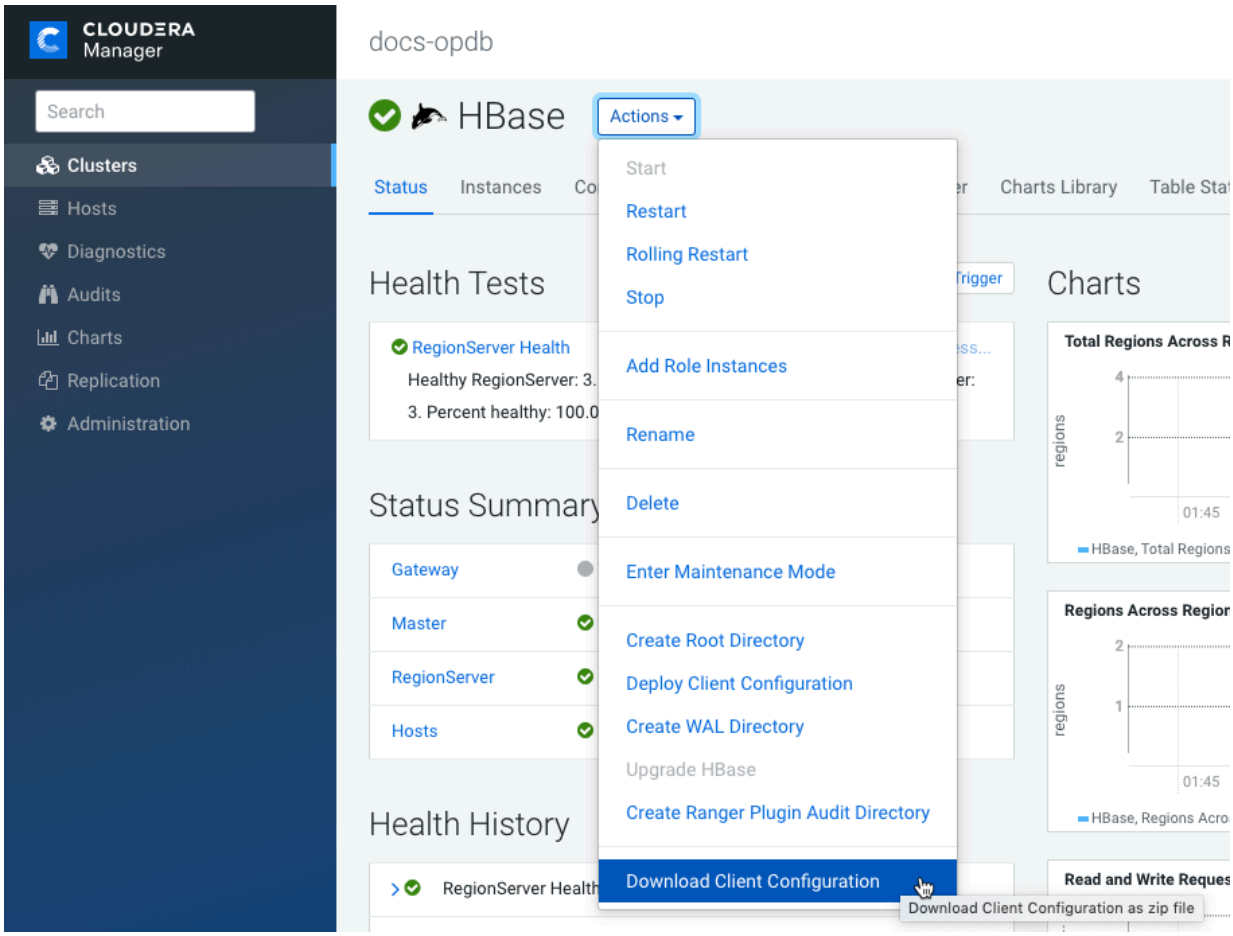
   - The HBase table for data ingest
   - The HBase Column family for data ingest into the specified table
   - All columns within the specified column family
   - Allows read and write permissions

Obtain HBase connection details

To enable an Apache NiFi data flow to communicate with HBase, you must obtain the HBase connection details by downloading several client configuration files. The NiFi processors require these files to parse the configuration values and use those values to communicate with HBase.

Procedure

1. From the Services pane in the Operational Database cluster that contains the HBase instance to which you want to ingest data, click CM-UI.
2. Click HBase, located on the Cloudera Manager Clusters pane.
3. From the HBase view, click Actions | Download Client Configurations to download the client configuration files as a zip file.



4. Expand the zip file and upload the following files to each of the NiFi nodes in your Flow Management cluster:
   - hbase-site.xml
   - core-site.xml

   For example:

   ```
   scp hbase-site.xml cdp_username@publiclp:/tmp
   ```

   You may also copy these files to a more permanent location later.

5. Adjust the file permissions to ensure that the NiFi processors can read them.

   For example:

```
chmod 755 tmp/*-site.xml
```

Build the data flow

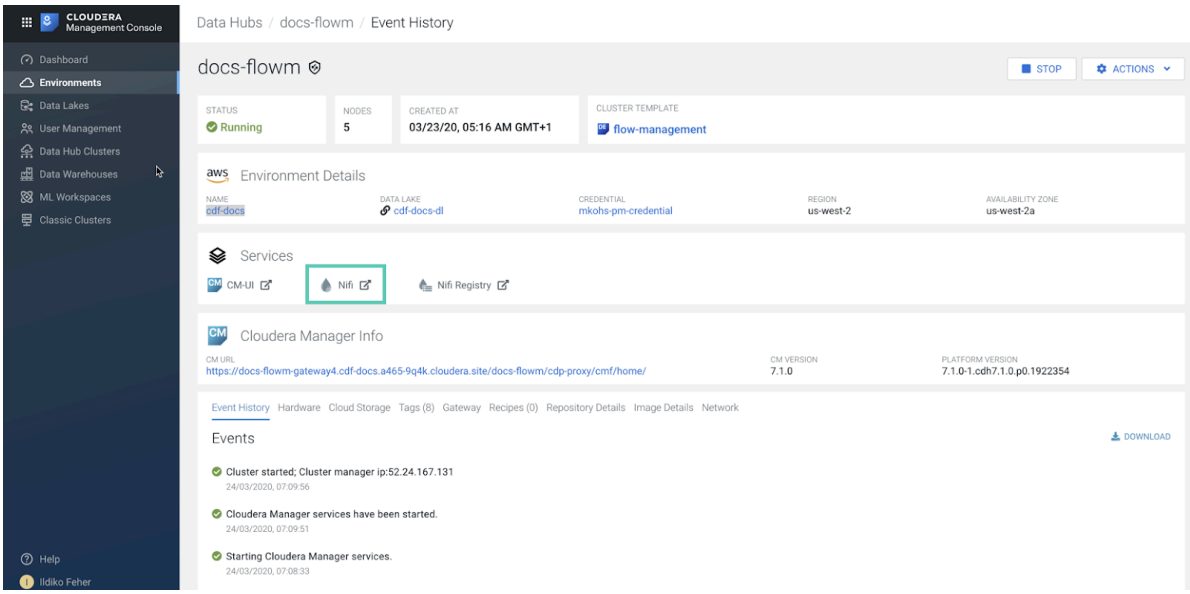
From the Apache NiFi canvas, set up the elements of your data flow. This involves opening NiFi in CDP Public Cloud, adding processors to your NiFi canvas, and connecting the processors.

About this task

You should use the `PutHBaseRecord` processor to build your HBase ingest data flows.

Procedure

1. Open NiFi in CDP Public Cloud.
   a) To access the NiFi service in your Flow Management cluster, navigate to Management Console service > Data Hub Clusters.
   b) Click the tile representing the Flow Management cluster with which you want to work.
   c) Click the NiFi icon in the Services section of the Cluster overview page to access the NiFi UI.

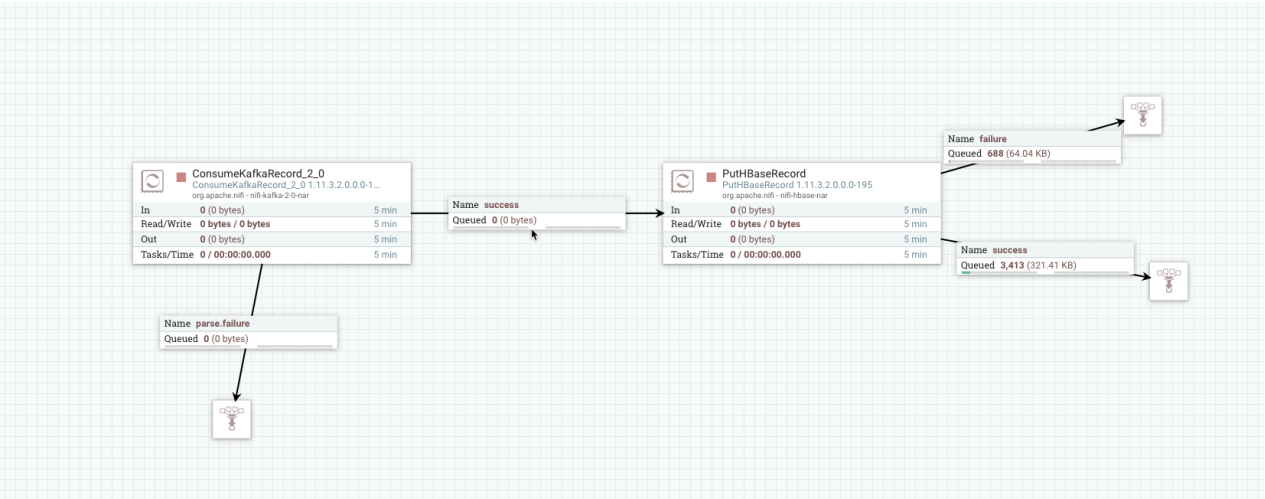


2. Add the NiFi Processors to your canvas.
   a) Select the Processor icon from the Cloudera Flow Management actions pane, and drag a processor to the Canvas.
   b) Use the Add Processor filter box to search for the processor you want to add, and then click Add.
   c) Add each of the processors you want to use for your data flow.
3. Connect the two processors to create a flow.
   a) Click the connection icon in the first processor, and drag it to the second processor.
   b) A Create Connection dialog displays. It has Details and Settings tabs. You can configure the connection's name, FlowFile expiration time period, thresholds for back pressure, load balance strategy, and prioritization.
   c) Click Add to close the dialog box and add the connection to your flow.

      Optionally, you can add success and failure funnels to your data flow, which help you see where flow files are routed when your data flow is running.

Results

Your data flow may look similar to the following:



What to do next

Create the Controller service for your data flow. You will need these services later on as you configure your data flow target processor.

Related Information

Ingesting data into Apache Kafka in CDP Public Cloud

ConsumeKafkaRecord_2_0

PutHBaseRecord

Building a data flow

Configure the HBase client service

You can add Controller Services to provide shared services to be used by the processors in your data flow. In the case of data ingest into HBase, use a Controller Service to configure the HBase client service.

About this task

When ingesting data into HBase in CDP Public Cloud, use the `HBase_2_ClientService` Controller Service.

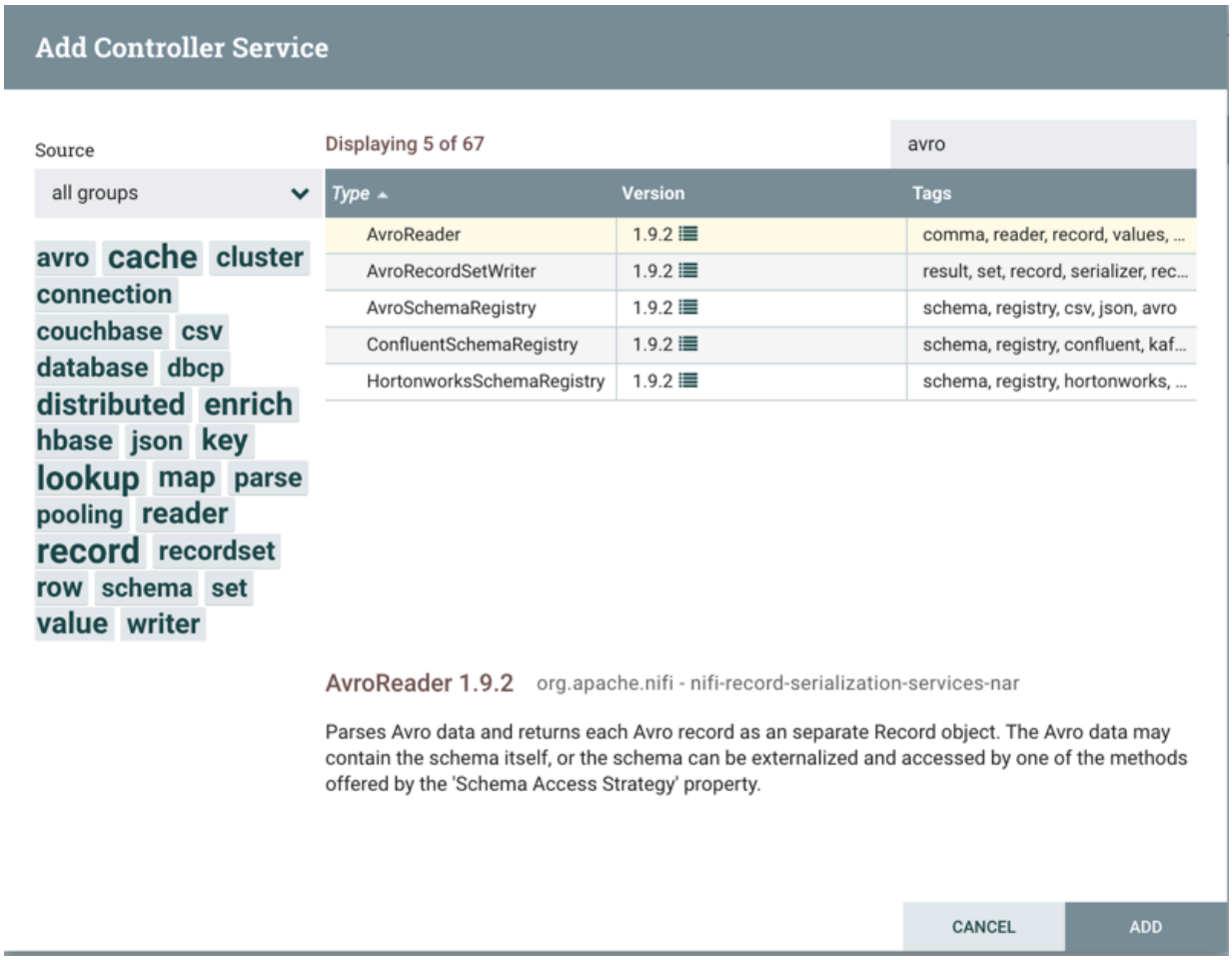
You must define the Controller Services for the processors in your data flow in the configuration of the root process group where they will be used.

Procedure

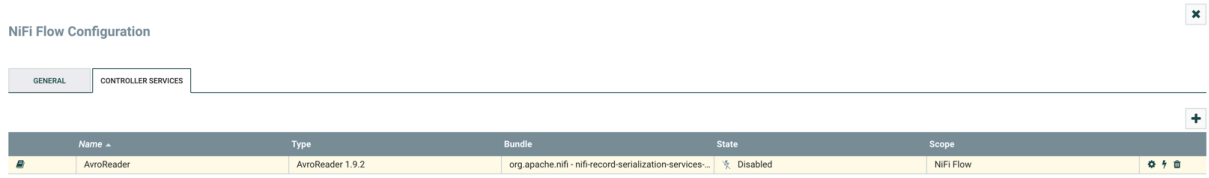
1. To add a Controller Service to your flow, right-click on the Canvas and select Configure from the pop-up menu. This displays the Controller Services Configuration window.
2. Select the Controller Services tab.

3. Click the + button to display the Add Controller Service dialog.

4. Select the required Controller Service and click Add.



5. Perform any necessary Controller Service configuration tasks by clicking the Configure icon in the right-hand column.



6. When you have finished configuring the options you need, save the changes by clicking the Apply button.

7. Enable the Controller Service by clicking the Enable button (flash) in the far-right column of the Controller Services tab.

Example

In this example, we are ingesting data into HBase, and will configure the `HBase_2_ClientService` Controller Service to create the HBase client service. For ingest data flows configure `HBase_2_ClientService` with the following required information. For a complete list of configuration properties, see the *Controller Service documentation*.

Property	Description	Example value for ingest data flow
Kerberos Principle	Specify the user name that should be used for authenticating with Kerberos. Use your CDP workload username to set this Authentication property.	srv-hbase-ingest-user
Kerberos Password	Provide the password that should be used for authenticating with Kerberos. Use your CDP workload password to set this Authentication property.	password
Hadoop Configuration Files	Comma-separated list of Hadoop Configuration files, such as hbase-site.xml and core-site.xml for kerberos, including full paths to the files.	These are the files you downloaded in *Obtain the HBase connection details*.

Note: You can also use the KerberosPasswordUserService controller service instead of entering the Kerberos principle and password.

Related Information

HBase_2_ClientService

Configure the processor for your data source

You can set up a data flow to move data from many locations into Apache HBase. This example assumes that you are configuring `ConsumeKafkaRecord_2_0`. If you are moving data from a location other than Kafka, review *Getting Started with Apache NiFi* for information about how to build a data flow, and about other data consumption processor options.

Before you begin

- Ensure that you have the Ranger policies required to access the Kafka consumer group.
- This use case demonstrates a data flow using data in Kafka, and moving it to HBase. To review how to ingest data to Kafka, see *Ingesting Data to Apache Kafka*.

Procedure

1. Launch the Configure Processor window, by right-clicking the processor and selecting Configure.
2. Click the Properties tab.

3. Configure `ConsumeKafkaRecord_2_0` with the required values.

   The following table includes a description and example values for the properties required to configure an ingest data flow. For a complete list of `ConsumeKafkaRecord_2_0`, see the *processor documentation*.

Property	Description	Value for ingest to HBase data flow
Kafka Brokers	Provide a comma-separated list of known Kafka Brokers in the format <host>:<port>.	`Docs-messaging-broker1.cdf-docs.a465-9q4k.cloudera.site:9093, docs-messaging-broker2.cdf-docs.a465-9q4k.cloudera.site:9093, docs-messaging-broker3.cdf-docs.a465-9q4k.cloudera.site:9093`
Topic Name(s)	Enter the name of the Kafka topic from which you want to get data.	*customer*
Topic Name Format	Specify whether the topics are a comma separated list of topic names, or a single regular expression.	*names*
Record Reader	Specifies the record reader to use for incoming Flow Files. This can be a range of data formats.	*AvroReader*
Record Writer	Specifies the format you want to use in order to serialize data before for sending it to the data target. Note: Ensure that the source record writer and the target record reader are using the same Schema.	*CSVRecordSetWriter*
Honor Transactions	Specifies whether or not NiFi should honor transactional guarantees when communicating with Kafka.	*false*
Security Protocol	Specifies the protocol used to communicate with brokers. This value corresponds to Kafka's 'security.protocol' property.	*SASL_SSL*
SASL Mechanism	The SASL mechanism to use for authentication. Corresponds to Kafka's 'sasl.mechanism' property.	*plain*
Username	Specify the username when the SASL Mechanism is PLAIN or SCRAM-SHA-256.	*srv_nifi-hbase-ingest*
Password	Specify the password for the given username when the SASL Mechanism is PLAIN or SCRAM-SHA-256.	*Password1!*
SSL Context Service	Specifies the SSL Context Service to use for communicating with Kafka.	*default*
Group ID	A Group ID is used to identify consumers that are within the same consumer group. Corresponds to Kafka's 'group.id' property.	*nifi-hbase-ingest*

What to do next

Configure the processor for your data target.

Related Information

ConsumeKafka_2_0
Building a data flow

Ingesting data into Apache Kafka in CDP Public Cloud

Configure the processor for your data target

You can set up a data flow to move data into many locations. This example assumes that you are moving data into Apache HBase using `PutHBaseRecord`. If you are moving data into another location, review *Getting Started with Apache NiFi* for information about how to build a data flow, and about other data ingest processor options.

About this task

If you are moving data into HBase, `PutHBaseRecord` is the preferred HBase processor to use.

Procedure

1. Launch the Configure Processor window, by right-clicking the processor and selecting Configure.
2. Click the Properties tab.
3. Configure `PutHBaseRecord` with the required values.

   The following table includes a description and example values for the properties required to configure an ingest data flow into HBase. For a complete list of `PutHBaseRecord`, see the *processor documentation*.

Property	Description	Value for ingest to HBase data flow
Table name	The name of the HBase table into which you want to put data. The format namespace:table should be used to get the correct data lineage in Atlas.	default:customer
Column family	The column family to use when inserting data into HBase.	customer_data
Row Identifier Field Name	Specifies the name of a record field whose value should be used as the row id for the given record.	customer_id
Record Reader	Specifies the Controller Service to use for parsing incoming data and determining the data's schema	RecordCustomerCSVReader
HBase Client Service	Specifies the Controller Service to use for accessing HBase.	Select the Controller Service you configured in *Configure the HBase client service*.

What to do next

Start your data flow.

Related Information

PutHBaseRecord

Start your data flow

Start your data flow to verify that you have created a working dataflow and to begin your data ingest process.

Procedure

1. To initiate your data flow, select all the flow components you want to start.
2. Click the Start icon in the Actions toolbar.

   Alternately, right-click a single component and choose Start from the context menu.

Verify your data flow

Once you have configured and started your data flow, you can verify that you are successfully ingesting data into HBase.

Before you begin

You have confirmed that NiFi processors are not producing errors.

Procedure

1. Open the HBase shell.

```
hbase shell
```

2. List your available HBase tables. You should see the customer table you created at the beginning of this workflow.

   For example:

```
list

TABLE
customer
1 row(s)
```

3. Scan the customer table you created.

   For example:

```
scan 'customer'
```

Results

A successful data flow results in data moving into the customer table.

Next steps

Provides information on what to do once you have moved data into HBase in CDP Public Cloud.

You have built a simple data flow for an easy way to move data to HBase. This example data flow enables you to easily design more complex data flows for moving and processing data in HBase. Here are some ideas for next steps:

- Review *NiFi documentation* to learning more about building and managing data flows.
- Review *NiFi Registry documentation* for more information about versioning data flows.
- Review the *Cloudera Runtime HBase documentation* for information about working with HBase.

Related Information

NiFi documentation

NiFi Registry documentation

Cloudera Runtime HBase information