Installing Cloudera Observability On-Premises

Date published: 2024-01-31
Date modified: 2024-02-14

CLOUDERA

https://docs.cloudera.com/

Legal Notice

Contents

Cloudera Observability On-Premises installation overview

A brief overview of the tasks required to successfully install Cloudera Observability On-Premises on a dedicated cluster within your environment.

Installing Cloudera Observability On-Premises requires the following tasks that are performed by you:

1. Creating a CDP cluster for Cloudera Observability On-Premises that contains a minimum of 5 nodes and that is managed by Cloudera Manager.

   Note: Cloudera Observability On-Premises must be installed in a dedicated cluster, separate from your development, test, or production workload clusters. This configuration minimizes the impact on the cluster and prevents the need to upgrade your workload clusters to meet the needs of Cloudera Observability On-Premises.
2. Verifying that your environment's system has the required supported software and hardware and the required network services and devices for installing Cloudera Observability On-Premises.
3. Performing the pre-installation tasks.
4. Downloading and deploying the Cloudera Observability On-Premises installation files from the Cloudera Downloads website to the host server on your Cloudera Observability cluster and activating the Cloudera Observability On-Premises parcel file.
5. Enabling secure communication and data encryption between components, Cloudera Observability On-Premises, and your Workload clusters.
6. Enabling the Phoenix service.
7. Enabling the Kafka service.
8. Adding the Cloudera Observability On-Premises service on all nodes in the Cloudera Observability On-Premises cluster and improving performance by enabling multiple devices to share the processing and memory workload.
9. Granting user access with either local or the LDAP protocol authentication.
10. Enabling the Telemetry Publisher service and associating it with your Workload clusters.
11. Performing the post-installation tasks.

Tip: The pre-installation, installation, and deployment tasks collect a series of parameter and property values. These values are used during the Cloudera Observability On-Premises installation and deployment tasks to configure and setup Cloudera Observability On-Premises specifically for your system. Cloudera recommends recording these values before starting a task.

Follow these guidelines to ensure a successful Cloudera Observability On-Premises installation:

- Decide on the type of Cloudera Observability On-Premises environment that best suites your business requirements.
- Read the system requirements. This ensures that your Cloudera Observability On-Premises cluster has the required base hardware and software.
- Read the installation pre-requisites and installation steps. This ensures that you understand the tasks required and how they are completed.
- Understand what software services are required and what account information is needed when configuring dependent services. Third-party software services, such as LDAP, network, and firewall security, must be configured by you. For example, your SSL key pair file locations and private key are required during installation.
- Record all the required configuration values, such as host names, port numbers, user names and passwords.
- During the pre-installation tasks, record any new configuration values as you create them. You will be required to enter these configuration values later when you install and deploy Cloudera Observability On-Premises.
- After installing Cloudera Observability On-Premises, verify that the software stack installed successfully.

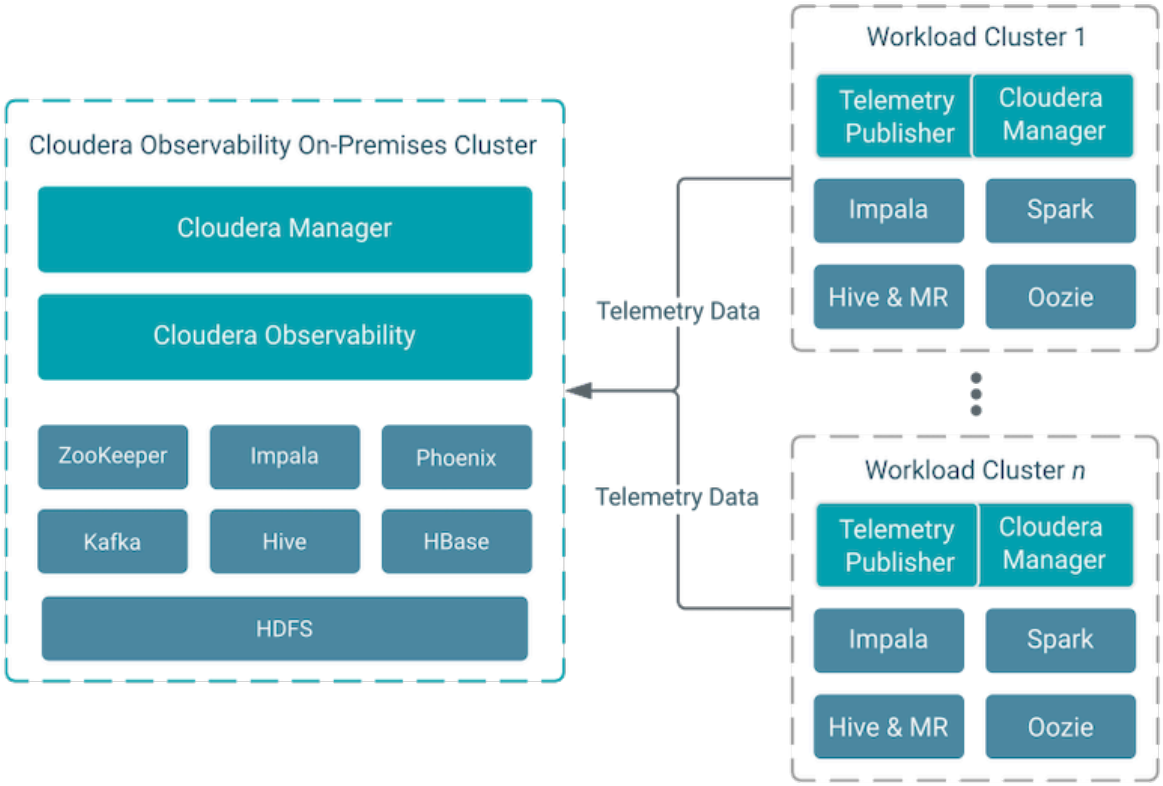
Architecture

Describes the components and architecture of a basic Cloudera Observability On-Premises environment.

Cloudera Observability On-Premises consists of two or more clusters:

- Cloudera Observability On-Premises cluster, which is a CDP cluster that is managed by Cloudera Manager. Cloudera Observability On-Premises and all its main component services are installed and run in this cluster. Users access the Cloudera Observability On-Premises web user interface from the web host server in this cluster.
- Workload Cluster, which is a CDP cluster managed by Cloudera Manager. This cluster is associated with Telemetry Publisher in Cloudera Manager and runs your workload processes.

The below diagram shows the communication between Cloudera Observability On-Premises and your workload clusters through Telemetry Publisher. Where, the Cloudera Observability On-Premises service is installed on the left cluster, including the Cloudera Observability On-Premises main component services, and the Workload clusters on the right contain the services required to run your workload processes. Telemetry data collected by Telemetry Publisher is passed from these clusters to the Cloudera Observability On-Premises cluster.



Estimating your deployment capacity size

Describes tasks that help you estimate the capacity sizing for a Cloudera Observability On-Premises deployment, including a sizing estimate for a five node cluster.

As part of your capacity sizing plan, Cloudera recommends that you test and explore Cloudera Observability in a non production environment on a five node cluster using the base capacity estimate below. This helps you evaluate your Cloudera Observability On-Premises throughput capacity sizing requirements using your existing workloads, which depending on the throughput results can then be scaled up or down. Cloudera also recommends that you use the Cloudera Observability On-Premises Performance metrics and alerts that are available in Cloudera Manager, which can be leveraged to analyze overall usage for your capacity planning.

Considerations and limitations

The Cloudera Observability On-Premises throughput is dependent on the number, frequency, and profile of your workloads and not the size or the number of clusters.

The following Cloudera Observability On-Premises workload resource considerations and known limitations must also be considered as part of your capacity sizing plan:

- Typically, Spark workloads consume the most resources to process, followed by Hive. If you are processing large workloads or a high number of workloads in a Spark or Hive engine adding more resource and processing time is required.
- Large workloads consume more resources than the equivalent number of smaller workloads. Identifying workloads that are using an excessive amount of resources and optimizing large workloads into smaller workloads during your capacity size planning testing will help reduce resource hungry jobs and queries.
- Cloudera Observability On-Premises evaluates the current job against its workload's baseline, which enables you to address performance problems by comparing the performance of your workloads after each run with the Job Comparison feature. Workloads that run frequently will therefore consume and require more system resources for your Cloudera Observability On-Premises cluster.
- A workload's processing time and resources are also dependent on the number of the workload's sub tasks.
- When running large workloads or running multiple jobs and/or queries, distribute their run times evenly throughout the day.

Base capacity estimate for a five node Cloudera Observability On-Premises cluster

Based on an average amount of workload throughput that uses the above considerations and limitations and where the average payload size is based on historical logs and files, such as the Spark history file size, the Impala profile.tgz file size, and the MapReduce job.xml, the estimated job processing capacity for a Five Node Cluster is as follows: [#]

Table 1: Sizing estimate for a five node cluster

Engine	Range
Hive	10,000-20,000 queries per day
Impala	50,000-100,000 queries per day
MapReduce	50,000-100,000 jobs per day
Spark	10,000-20,000 jobs per day
YARN	100,000-200,000 jobs per day

If you require help with your capacity size estimate, such as your cluster exceeds the 5 node base setup, contact your Cloudera sales representative or your Cloudera Account team.

System requirements

Lists the minimum supported system requirements for your Cloudera Observability On-Premises cluster.

Before you install Cloudera Observability On-Premises, you must verify that your environment contains the minimum supported requirements for software, hardware, and networks.

[#] The estimate is based on an average payload size of:

- 3 MB for Hive workloads
- 100 KB for Impala workloads
- 50 KB for MapReduce workloads
- 1 MB for Spark workloads
- 100 KB for YARN workloads

Hardware requirements

Lists the minimum supported hardware requirements for your Cloudera Observability On-Premises cluster.

In addition to the minimum supported hardware requirements for the services that you have installed on your Workload cluster nodes, you must verify that the dedicated cluster for Cloudera Observability On-Premises also contains the recommended minimum hardware requirements.

The recommended minimum hardware requirements for the Cloudera Observability On-Premises cluster are:

- A computer cluster of 5 nodes that hosts Cloudera Observability On-Premises.
- Where, each computer node in the Cloudera Observability On-Premises cluster must contain a minimum of:
  - CPU - 16 CPU cores
  - RAM - 128 GB (For optimal performance, Cloudera recommends 256 GB)
  - Disk Space - 12 TB (For multiple disks the disk space can be divided, for example, the minimum disk space requirement for 6 disks is 2 TB for each disk)

Note: To prevent issues from impacting operations other than those performed by Telemetry Publisher, such as sending data to Cloudera Observability On-Premises, Cloudera highly recommends that the host server on which you assigned the Telemetry Publisher Service role is allocated its own dedicated disk.

Where, depending on the number and size of the jobs run on the cluster, the minimum supported disk drive size is 500GB. This size includes enough disk space for Telemetry Publisher to store data locally when it is unable to send data to Cloudera Observability due to connectivity or other issues.

Supported file systems

Lists the supported file systems for your Cloudera Observability On-Premises cluster.

The following files systems are supported:

- Hadoop Distributed File System (HDFS)
- Amazon Simple Storage Service (S3)
- Azure Data Lake Storage (ADLS)

Supported operating systems

Lists the supported operating systems for your Cloudera Observability On-Premises cluster.

You must verify that the dedicated cluster for Cloudera Observability On-Premises runs on one of the following supported Linux operating systems:

- CentOS Enterprise Linux, version 7, 8, or later
- Red Hat Enterprise Linux, version 7, 8, or later

The OS version requirement for Cloudera Observability On-Premises is the same as the CDP Private Cloud Base.

For more information about the OS requirements and versions for the CDP Private Cloud Base, see the Cloudera support matrix page by clicking the Related Link below.

Note: All the nodes within the Cloudera Observability On-Premises cluster must run the same version of the Cloudera Observability On-Premises supported Linux operating system.

Related Information

Cloudera Support Matrix

Supported Cloudera versions

Lists the supported Cloudera platform and software versions for your Cloudera Observability On-Premises cluster and Workload clusters.

The following table lists the supported Cloudera platform and software for running Cloudera Observability On-Premises:

Table 2: Supported Cloudera platform and software for your Cloudera Observability On-Premises cluster

Product	Version
CDP Private Cloud Base	7.1.9 or later
Cloudera Manager	7.11.3 or later

The following table lists the supported Oracle Java Development Kit (JDK) for running Cloudera Observability On-Premises:

Table 3: Supported JDK

Product	Version
Oracle Java Development Kit (JDK)	JDK 17

The following table lists the supported Cloudera platform and software for running your workload clusters:

Table 4: Supported Cloudera platforms and software for your Workload clusters

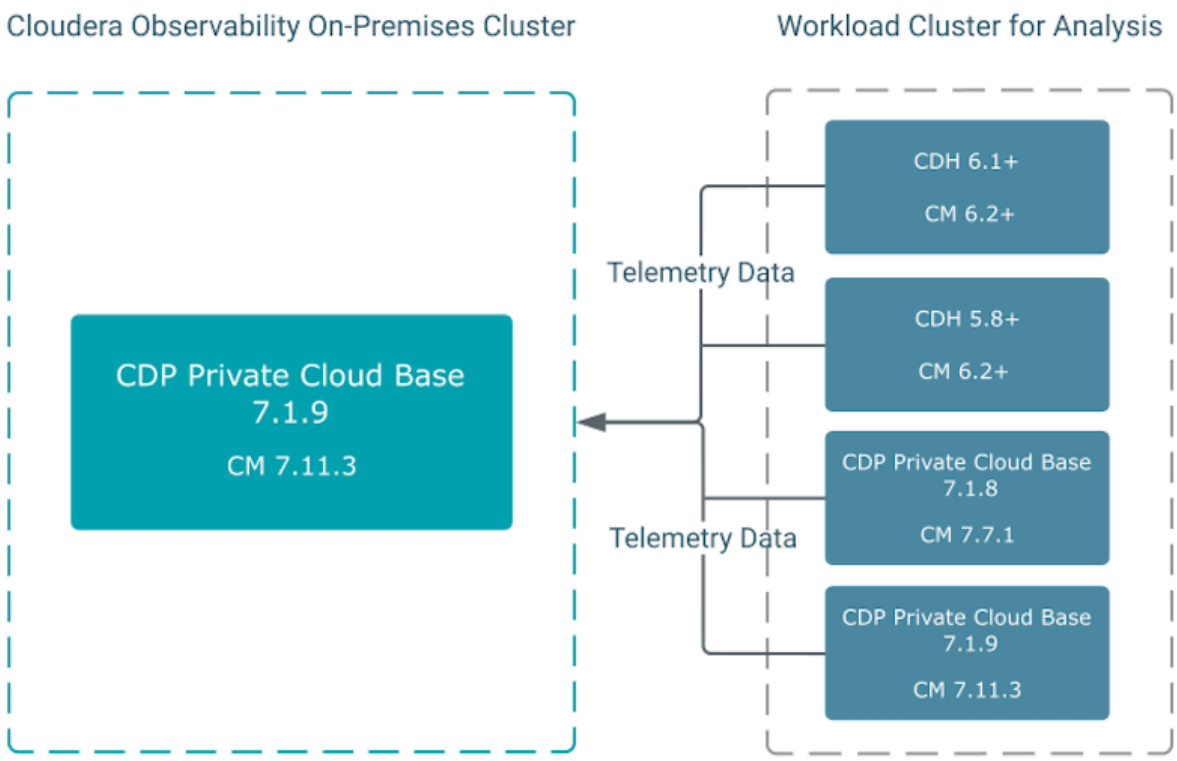
Cluster	Version	Cloudera Manager
CDH 5.x cluster	CDH version 5.8 and later	Cloudera Manager version 6.2 and later
CDH 6.x clusters	Cloudera Manager version 6.1 and later	Cloudera Manager version 6.2 and later
CDP 7.x clusters	Private Cloud Base 7.0.3 or later clusters	Cloudera Manager version 7.1.1 or later

Unsupported versions

The following versions are not supported:

- Cloudera Manager 6.0 and 7.0.3

The following figure shows an example of the Cloudera versions that are supported by Cloudera Observability On-Premises:



Important: To display the most current diagnostic metrics and health statistics collected by Telemetry Publisher in the Cloudera Observability On-Premises web UI, you must upgrade to the latest version of Cloudera Manager and restart Telemetry Publisher.

Related Information

Cloudera Support Matrix

Network port requirements

Lists the network port numbers and their respective protocols used by Cloudera Observability On-Premises and dependent services.

Note: To enable communication, you may need to reconfigure or update your firewall.

Protocols are defined as follows:

- UI Port (ui.port), which serves the Cloudera Observability On-Premises user interface (UI) and communicates using HTTPS when TLS/SSL is enabled, otherwise it communicates using HTTP.
- API Port (api.port), which listens for REST calls to API-based servers. It communicates using HTTPS when TLS/SSL is enabled, otherwise it communications using HTTP.
- Metrics Port (webservice.port), which exposes an interface to the metrics that the Cloudera Observability On-Premises roles collect.
- GRPC Port (grpc.port), which listens for gRPC requests against the backend servers. This protocol is used for inter-role communication.

The following table lists the port numbers that must be enabled for service-to-service network communication with Cloudera Observability On-Premises:

Table 5: Network Port numbers for Cloudera Observability On-Premises

Service	UI Port (ui.port)	API Port (api.port)	Metrics Port (webservice.port)	gRCP Port (grpc.port)
Admin API Server			12111	12112
Analytic Database Server			12031	12032
API Server		12012	12011	
Baseline Server			12041	12042
Console Server	12001			
Databus API Server		12022	12021	
Databus Server			12051	12052
Entities Server			12061	12062
Pipeline Server			12071	12072
SDX Server			12081	12082

The following services are exposed service-wide:

- The Kafka Port (kafka.broker.port), which is the port for Apache Kafka used by Cloudera Observability On-Premises.
- The Phoenix Query Server Port (phoenix.queryserver.port), which is the port for the Phoenix Query Server used by Cloudera Observability On-Premises.
- The Impala Daemon Port (impala.daemon.port), which is the port for the Impala Daemon used by Cloudera Observability On-Premises.

Pre-installation tasks

The tasks that must be completed before you install Cloudera Observability On-Premises.

Configure the Java Heap requirements

Setting the supported Java heap size for the Zookeeper, HBase, HDFS, and Phoenix services, ensures the long-term success of a Cloudera Observability On-Premises deployment.

About this task

Describes how to set the Java heap size in Cloudera Manager for the Cloudera Observability On-Premises services listed in the following table.

Table 6: Java Heap size settings

Service	Size Setting Name	Minimum Value
HBase	Java Heap Size of HBase RegionServer in Bytes - set the value in the RegionServer Default Group field	16 GB minimum (recommended 31 GB)
HDFS	Java Heap Size of NameNode in Bytes	4 GB minimum (recommended 8 GB)
Kafka	Java Heap Size of Broker	4 GB minimum (recommended 8 GB)

Service	Size Setting Name	Minimum Value
Phoenix	Phoenix Query Server Max Heapsize	4 GB minimum (recommended 8 GB)
ZooKeeper	Java Heap Size of ZooKeeper Server in Bytes	4 GB minimum (recommended 8 GB)

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters and then select the *Service* name. For example, Zookeeper.
3. In the *Service* name page, click the Configuration tab and then search for the *Size Setting Name*. For example, in the Search field, enter java heap, which locates the Java Heap Size of ZooKeeper Server in Bytes setting for the Zookeeper product.
4. Change the setting to the minimum supported value or higher for the service. For example, change the Java Heap Size of ZooKeeper Server in Bytes setting, to 4 GB.
5. Click Save.
6. Repeat theses steps for each service using the above Java Heap size settings table.

Configure performance improvement settings

Improve the performance of Cloudera Observability On-Premises by configuring the ZooKeeper and HBase property settings.

About this task

Describes how to set the ZooKeeper and HBase property values listed in the following table:

Table 7: Performance improvement settings

Service	Property	Value
ZooKeeper	maxClientCnxns	300
HBase	hbase.regionserver.handler.count	40
HBase	hbase.hstore.blockingStoreFiles	100
HBase	hbase.ipc.server.max.callqueue.size	2147483648 bytes (2GiB)

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters and then the *Service* name. For example, ZooKeeper.
3. In the *Service* name page, click the Configuration tab, and then search for the *Property* name. For example, in the Search field, enter maxClientCnxn, which locates the Maximum Client Connections setting for ZooKeeper.
4. Change the setting to the value for the service as listed in the Performance improvement settings table. For example, change the Maximum Client Connections setting, to 300.
5. Repeat steps 2 to 4 for the hbase.regionserver.handler.count and the hbase.hstore.blockingStoreFiles properties listed for the HBase service.
6. For the hbase.ipc.server.max.callqueue.size property do the following:
   a. In the HBase Configuration tab, search for safety valve.
   b. In the HBase Service Advanced Configuration Snippet (Safety Valve) for hbase-site.xml section, click Add Another.
   c. Add the hbase.ipc.server.max.callqueue.size setting and set the value to 2147483648 bytes (2GB).
7. Restart HBase, ZooKeeper, and any other dependent services.

Confirm the installation of the main component services

Lists the Cloudera Observability On-Premises main component services. These services must be installed before installing Cloudera Observability On-Premises.

You must verify that the following services are installed on the cluster in which Cloudera Observability On-Premises is to be installed.

- In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
- In Cloudera Manager, select Clusters, and then confirm that your Cloudera Observability On-Premises cluster has the following required services:

- HBase
- HDFS
- Hive
- Optional: Hue

Note: Though an optional service, Cloudera recommends installing Hue on the cluster in which Cloudera Observability On-Premises is to be installed as it provides troubleshooting and extraction services.

- Impala
- Kafka
- Phoenix
- ZooKeeper

Note: To optimize performance, Cloudera highly recommends that you do not install any other services on the Cloudera Observability On-Premises cluster.

Redacting data

Telemetry Publisher collects diagnostic data from logs, job configurations, and SQL queries, and then sends this data to Cloudera Observability On-Premises. As this diagnostic information may contain sensitive information it is important to mask this data before Telemetry Publisher sends it to Cloudera Observability On-Premises.

Data redaction works separately from Cloudera data encryption techniques. Data encryption alone does not preclude administrators with full access to the cluster from viewing sensitive user data. Redaction ensures that cluster administrators, data analysts, and others cannot see personally identifiable information (PII) or other sensitive data that is not within their job domain. At the same time, it does not prevent users with appropriate permissions from accessing data to which they have privileges.

In the event that redaction was disabled, such as during testing, Cloudera recommends that before you configure Telemetry Publisher you verify that redaction has not been disabled.

Redacting log and query data

By default, redaction for log and SQL query data is enabled for Telemetry Publisher.

Note: Only the sensitive data in the actual file is redacted. Metadata, such as the file's name, the file's owner, and information about the data in the file is not redacted.

Redacting Spark data

By default, redaction is enabled in the YARN service for Spark SQL data.

The YARN service redacts Apache Spark SQL sensitive data from event and executor logs.

Important: To ensure that Telemetry Publisher only sends redacted data to Cloudera Observability On-Premises do not change the spark.redaction.regex configuration property.

Redacting MapReduce data

Telemetry Publisher reads the job configuration file from HDFS. You can enable data redaction for your MapReduce jobs pulled from HDFS by Telemetry Publisher by adding your MapReduce job configuration properties in the Cloudera Manager YARN configuration settings.

About this task

In YARN you add MapReduce job configurations that enable data redaction when MapReduce data is pulled from HDFS.

Procedure

1. In a supported web browser on your Workload cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, YARN, and then click the Configuration tab.
3. Search for the Redacted MapReduce Job Properties property.

   Note: By default, several MapReduce job configuration properties are set for you by the YARN service. Do not change these settings.

4. Add additional MapReduce job configurations by clicking the plus sign (+), which is located after the last configured property, and entering the default gateway group.
5. Click Save Changes.
6. Restart the YARN service.

Disabling redaction for testing

Steps for disabling the Log and Query redaction property in Telemetry Publisher for testing tasks.

About this task

Describes how to disable the Log and Query Redaction property, which by default is enabled for Telemetry Publisher.

Important: To protect sensitive data from being accessed by unauthorized users, Cloudera recommends that log and query redaction is enabled for both HDFS and the Telemetry Publisher service.

The Log and Query Redaction property works with the Log and Query Redaction property in HDFS. Both redaction properties must be disabled for Telemetry Publisher to start.

Note: The Log and Query Redaction configuration property is available in Cloudera Manager version 5.16 and higher.

Procedure

1. In a supported web browser on a Workload cluster, log in to Cloudera Manager with administrative privileges.
2. In Cloudera Manager, select Clusters, HDFS, and then click the Configuration tab.
3. In the Search field, enter redact, which locates the Log and Query redaction properties for HDFS.
4. Deselect the Enable Log and Query Redaction property.
5. Click Save Changes.
6. In the Cloudera Manager Home page, select Clusters, locate and select Cloudera Management Service, and then select the Configuration tab.
7. From the Filters panel in the SCOPE section, select Telemetry Publisher.
8. In the Search field, enter redact, which displays the Log and Query Redaction property.
9. Deselect the Log and Query Redaction property for the Telemetry Publisher Default Group.
10. Click Save Changes.
11. Restart both the HDFS and the Telemetry Publisher services, which disables the log and query redaction feature.

Generating Telemetry Publisher access credentials

Steps for generating access credentials that enable communication between your Workload clusters and Cloudera Observability On-Premises.

About this task

Describes how to create the access and private keys for Cloudera Manager's Telemetry Publisher, which collects and sends your workload information to Cloudera Observability On-Premises. You will be required to supply these values when you enable the Telemetry Publisher service on your Workload Cluster.

Note: Only Cloudera Observability On-Premises administrators (ObservabilityClusterAdmin formerly WXMClusterAdmin) can create Telemetry Publisher access credentials.

Procedure

1. In a supported browser, log in to the Cloudera Data Platform (CDP) as the user with administrative privileges.

   The CDP Cloud web interface landing page opens.
2. From the Your Enterprise Data Cloud landing page, select the Management Console tile.

   The Management Console home page opens.
3. From the Management Console's navigation panel, scroll down, click your user name, and then select Profile.

   Your Profile page opens.
4. Click Generate Access Key.

   The Generate Access Key dialog box opens.
5. In the Generate Access Key dialog box, click Generate an old format access key.

   The Telemetry Publisher Access Key ID and Private Key credentials are created and their respective fields populated with the credential text.
6. Do one of the following:
   - Manually save the access credentials:
     a. Record the Access Key ID credential text and store it somewhere safe. You will be required to supply this value when you enable the Telemetry Publisher service on your Workload Cluster.
     b. In a new text file, copy and paste the Private Key credential text exactly as provided without trailing spaces and then name and save the file somewhere safe. You will be required to supply this file when you enable the Telemetry Publisher service on your Workload Cluster.
   - Download the credentials file.
     a. Click Download Credentials File.
     b. Go to your Downloads directory.
     c. Open the file in a text editor and record the Access Key ID credential text and store it somewhere safe.
     d. Remove the text containing the Access Key ID, starting from [default] and ending with cdp_private_key=.

        The credentials file should now only contain the Private Key credentials text, starting from -----BEGIN P RIVATE KEY----- and ending with -----END PRIVATE KEY-----.
     e. Save the file somewhere safe. You will be required to supply this file when you enable the Telemetry Publisher service on your Workload Cluster.

Related Information

Enabling the telemetry network communication for Cloudera Observability On-Premises

Assigning access roles in Cloudera Observability On-Premises

Deploying the Cloudera Observability On-Premises install files

The tasks that download and deploy the Cloudera Observability On-Premises installation files from either the Cloudera Downloads website or the Cloudera download archive to the host server on your Cloudera Observability On-Premises cluster and activate the Cloudera Observability On-Premises parcel file.

Downloading the installation files using the download archive URL

Provides information about Cloudera download archive URLs for accessing the Cloudera Observability On-Premises installation files.

Important: Download the Cloudera Observability On-Premises installation parcels and files to a computer connected to the same network as the Cloudera Observability On-Premises cluster. To obtain the installation parcels and files, contact your Cloudera sales representative.

Verify that you have an active Cloudera Observability On-Premises subscription agreement, license key, and access authentication credentials for the Cloudera Observability On-Premises download archive repository. To obtain this information, contact your Cloudera sales representative.

Your Cloudera Observability On-Premises download credentials differ from your Cloudera support portal access credentials. Ensure to use the correct download credentials.

Downloading the installation files using the downloads website

Steps for downloading the Cloudera Observability On-Premises installation files from the Cloudera download website.

About this task

Describes how to download the Cloudera Observability On-Premises files from the Cloudera download website for an on-premises installation.

Important: The Cloudera Observability On-Premises files must be downloaded to a computer that is on the same network as the on-premises cluster.

Procedure

1. Verify that you have an active Cloudera Observability On-Premises subscription agreement, license key, and login access to the Cloudera download website. For information on how to obtain these, contact your Cloudera sales representative.
2. In a web browser on a computer that is on the same network as the on-premises cluster, go to the Cloudera download website for Cloudera Observability On-Premises.
3. From the CHOOSE SERVICE INSTALLATION TYPE list in the Get started section, click Cloudera Observability On-Premises Parcel.
4. From the Parcel Type list, select the installation parcel type for your Enterprise Linux operating system. For example, EL7 will install the Cloudera Observability On-Premises parcel on a RedHat RHEL 7 or CentOS 7 operating system.

   Note: For Advanced users who are installing the Cloudera Observability On-Premises installation parcels in a local parcel repository, click manifest.json.

5. Click DOWNLOAD NOW, sign-in or complete the product interest form and click Continue, accept the Cloudera licensing terms, and then click Submit.

6. From the download On Premises parcel section, do the following:
   a. Click On-premises (Parcel), which downloads the Cloudera Observability On-Premises parcel to your computer.
   b. Click On-premises Parcel (Sha), which downloads the Cloudera Observability On-Premises shell archive parcel to your computer.
7. Scroll up to the Get Started section.
8. From the Installation Type list, click Cloudera Observability CSD.
9. Click DOWNLOAD NOW, accept the Cloudera licensing terms, and then click Submit.
10. From the download parcel section, click On-premises CSD, which downloads the custom service descriptor jar file that enables you to install Cloudera Observability On-Premises.

Results

The following files are downloaded to your computer:

- Parcel: OBSERVABILITY-*version_build*-el*version_OS*.parcel
- Parcel SHA: OBSERVABILITY-*version_build*-el*version_OS*.parcel.sha
- CSD: OBSERVABILITY-*version_build*.jar

Deploying the installation files

Steps for copying the Cloudera Observability On-Premises installation files from the computer where the files were downloaded to the Cloudera Manager Server parcel directories on the Cloudera Observability On-Premises cluster.

About this task

Describes how to deploy the downloaded Cloudera Observability On-Premises installation files to the cluster on which you plan to install Cloudera Observability On-Premises.

Procedure

1. Verify that you have the domain name of the Cloudera Manager Server host on the Cloudera Observability On-Premises cluster.
2. In a terminal on the computer where the installation files were downloaded, log in to the Cloudera Manager Server host and verify that you can establish a secure shell (SSH) and a secure copy protocol (SCP) connection between the computer where the installations files were downloaded and the Cloudera Manager Server host.
3. Go the directory where the Cloudera Observability On-Premises installation files were downloaded.
4. As the root user, SSH to the Cloudera Manager Server host.

   For example,

   ```
   ssh root@hostname
   ```

5. In the directory where the Cloudera Observability On-Premises installation files were downloaded, do the following:
   a) Using the SCP protocol, copy the Cloudera Observability On-Premises parcel files to the /opt/cloudera/parcel-repo directory of the Cloudera Manager Server by entering the following command:

   ```
   scp OBSERVABILITY-version_build-elversion_OS.parcel
   root@cm_mgr_server_host:/opt/cloudera/parcel-repo/
   scp OBSERVABILITY-version_build-elversion_OSparcel.sha r
   oot@cm_mgr_server_host:/opt/cloudera/parcel-repo/
   ```

b) Copy the Cloudera Observability On-Premises CSD file to the /opt/cloudera/csd directory of the Cloudera Manager Server by entering the following command:

```
scp OBSERVABILITY-version_build.jar root@cm_mgr_server_host:/opt/cloude
ra/csd/
```

6. In Cloudera Manager Server go to the /opt/cloudera/parcel-repo and the /opt/cloudera/csd directories and set the ownership of the copied files and change the read, write, and execute permissions to 644 by entering the following commands:

```
chown cloudera-scm:cloudera-scm /opt/cloudera/parcel-repo/OBSERVABILITY-
*;
chmod 644 /opt/cloudera/parcel-repo/OBSERVABILITY-*;

chown cloudera-scm:cloudera-scm /opt/cloudera/csd/OBSERVABILITY-*;
chmod 644 /opt/cloudera/csd/OBSERVABILITY-*;
```

7. Restart the Cloudera Manager Server by entering the following command:

```
service cloudera-scm-server restart
```

8. In a supported web browser, log in to Cloudera Manager on the Cloudera Observability On-Premises cluster.
9. In Cloudera Manager, select Clusters, and locate and select Cloudera Management Service.

   The Cloudera Management Service page opens.
10. From the Actions menu, click Restart.
11. In the Restart message, confirm restarting the management roles by clicking Restart.

Activating the Cloudera Observability On-Premises parcel

Distribute the Cloudera Observability On-Premises installation files on all the nodes in the Cloudera Observability On-Premises cluster.

About this task

Describes how to activate the Cloudera Observability On-Premises installation parcel.

Procedure

1. In a supported web browser, log in to Cloudera Manager on the Cloudera Observability On-Premises cluster.
2. In Cloudera Manager, select Hosts and then Parcels.
3. In the Parcels page, verify that Cluster 1 is the Cloudera Observability On-Premises cluster.
4. From the Parcel Name section, locate and select OBSERVABILITY and then click Distribute.
5. When the Distributed indicator appears, click Activate.
6. In the Activate OBSERVABILITY confirmation message, click OK.

Results

The indicators for the OBSERVABILITY parcel are displayed as Distributed and Activated.

Incorrect installation parcel placement

Adding the Cloudera Observability On-Premises installation parcels in the wrong directory on the Cloudera Management Server host causes distribution and activation issues. The Cloudera Observability On-Premises installation parcel files must reside in the /opt/cloudera/parcel-repo directory.

About this task

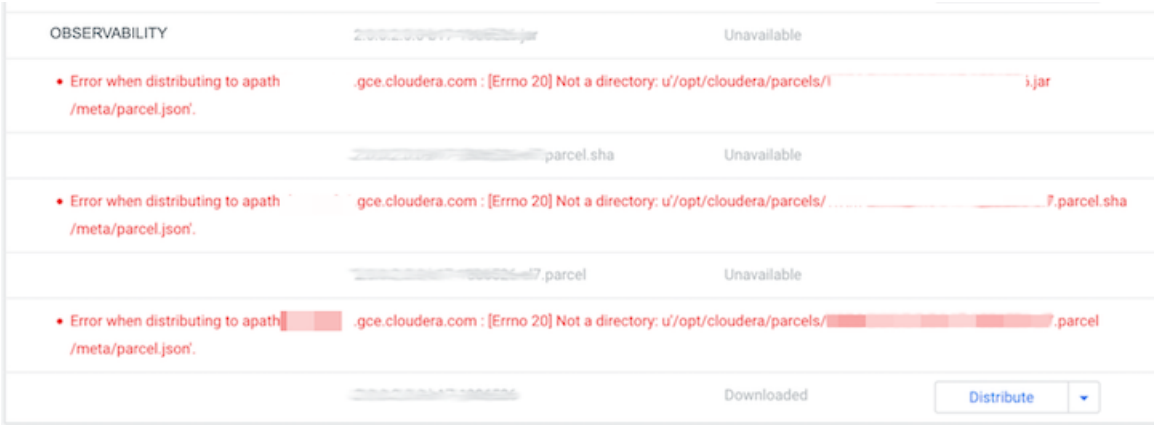
Issues arise when the Cloudera Observability On-Premises installation parcels are incorrectly placed in the wrong directory and the Cloudera SCM server is restarted. This task discusses the type of error messaged generated, where to locate the parcel error messages, and what to do when you receive this type of message.

If your installation of Cloudera Observability On-Premises fails and you receive a message that reports a "getpwnam()" error, do the following:

Procedure

1. Verify that the Cloudera Observability On-Premises installation parcels are residing in the /opt/cloudera/parcel-repo directory of the Cloudera Management Server.
2. Verify whether the parcel is correctly distributed and activated, by going to the Parcels page in Cloudera Manager.

   An example of the errors displayed on the Parcel page during the Cloudera Observability On-Premises parcel distribution process is shown below:



3. In a terminal, move the displaced parcel from the wrong directory to the /opt/cloudera/parcel-repo directory.
4. Restart the Cloudera SCM server and Cloudera SCM agent by using the following commands:

```
service cloudera-scm-server restart
```

```
service cloudera-scm-agent restart
```

Securing the Cloudera Observability On-Premises service data

Describes how to enable secure connections and access authenticity when transferring data between components of Cloudera Observability On-Premises and your data.

Cloudera Observability On-Premises stores your workload data in HDFS and HBase, where the HDFS data is created in the root path and the directories have observability:impala ownership. Configuring Kerberos and TLS/SSL ensures access authenticity and protects connections to your data.

Configuring secure table creation in a Kerberos environment

Cloudera Observability On-Premises must be able to create Phoenix tables in data storage. If you are installing Cloudera Observability On-Premises in a Kerberos environment, it must be able to securely create these tables.

About this task

Describes how to add the observability user as a HBase superuser, which securely enables Cloudera Observability On-Premises to create and store Phoenix tables.

Note: These steps are for a Kerberos environment only.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In the navigation panel, select Clusters and then in the Status page, select the HBASE service.
3. In the HBASE service page, select the Configuration tab.
4. In the Search field, enter hbase superusers, which displays the HBase Superusers property.
5. In the HBASE-1 (Service-Wide) field, enter observability, which adds the observability user as a HBase superuser.

   Tip: If the HBASE-1 (Service-Wide) field is not visible, click the plus icon.

6. Click Save Changes.

Configuring TLS

Enable secure connections for data transfers and user access with either the Transport Layer Security (TLS) protocol or the Secure Socket Layer (SSL) protocol, which ensures access authenticity and securely protects your data.

Note: Cloudera recommends that you configure your cluster to use auto-TLS, which eases the process of configuring TLS/SSL.

TLS/SSL is supported between the following services:

- The supported web browser and the Cloudera Observability On-Premises UI.
- Telemetry Publisher and the Cloudera Observability On-Premises API.
- The Cloudera Observability On-Premises UI and the Cloudera Observability On-Premises API.
- The Cloudera Observability On-Premises Servers and Impala.

Configure the TLS properties based on the edge connection that you want to encrypt.

The following tables list the property settings for enabling TLS/SSL encrypted communication between the Cloudera Observability On-Premises system components:

- The supported web browser connected to the Cloudera Observability On-Premises UI.
- The Console Server and other REST Clients connected to the Admin API Server, the API Server, and the Databus API Server.
- The Pipeline Server, the Analytic Database Server, the Entities Server, the Databus Server, and a SDX Server connected to Impala Server.

Table 8: TLS/SSL parameters for a secure connection between your browser and the Cloudera Observability On-Premises UI

Component	Property	Value
Console Server	TLS/SSL Server Private Key File (PEM)	ssl.privatekey.path
Console Server	TLS/SSL Server Certificate File (PEM)	ssl.cert.path
Console Server	TLS/SSL Private Key Password	ssl.privatekey.password
Console Server	Enable TLS/SSL	ssl.enabled

Table 9: TLS/SSL parameters for a secure connection between the Console Server and other REST clients and the Admin API Server, the API Server, and the Databus API Server

Component	Property	Value
Console Server	TLS/SSL Certificate Trust Store File	ssl.cacert.path
Admin API Server	TLS/SSL Certificate Trust Store File	ssl.trustStore.path
Admin API Server API Server Databus API Server	Enable TLS/SSL	ssl.enabled
Admin API Server API Server Databus API Server	TLS/SSL Server JKS Keystore File Location	ssl.keyStore.path
Admin API Server API Server Databus API Server	TLS/SSL Server JKS Keystore File Password	ssl.keyStore.password
Admin API Server API Server Databus API Server	TLS/SSL Server JKS Keystore Key Password	ssl.keyManager.password

Table 10: TLS/SSL parameters for a secure connection between the Pipeline Server and several other servers to the Impala Server

Component	Property	Value
Pipelines Server Analytic Database Server Entities Server Databus Server SDX Server	TLS/SSL Client Trust Store File	ssl.trustStore.path
Pipelines Server Analytic Database Server Entities Server Databus Server SDX Server	TLS/SSL Client Trust Store Password	ssl.trustStore.password

Configuring Observability service when TLS/SSL enabled in Phoenix

You can use Cloudera Manager to set the Observability service configuration property. This setting is applicable only if TLS/SSL is enabled for Phoenix.

Before you begin

You must have enabled the TLS/SSL for Phoenix.

About this task

Environment safety valves let you configure environment variables across the service and affect all components of Cloudera Observability On-Premises.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.

2. Select Clusters Observability Configuration and filter by SCOPE Observability (Service-Wide) and CATEGORY Advanced .
3. For the Observability Service Environment Advanced Configuration Snippet (Safety Valve) property, enter the following values:
   - Key: PHOENIX_SSL_ENABLED
   - Value: true
4. Click Save Changes.
5. Select Actions Restart to restart the Observability service.

Enabling the Phoenix service

The tasks that enable the Phoenix service for your Cloudera Observability On-Premises environment.

Adding the Phoenix query server role

Assign the Phoenix Query Server role to all the hardware devices in the Cloudera Observability On-Premises environment.

About this task

Describes how to assign the Phoenix Query Server role to all your hosts.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, Phoenix, and then from the Actions menu, select Add Role Instances.
3. In the Add Role Instances to PHOENIX page, click inside the Query Server x *n* field, which opens the Hosts Selected page.
4. Add the Query Server role to all hosts by doing the following:
   a. Select the check box by the side of each host, which adds a Query Server role icon in the Added Roles column for each selected host.
   b. Click OK, which takes you back to the Add Role Instances to PHOENIX page where the Query Server x *n* field is now populated with the selected host names.
   c. Click Continue.
   d. In the Review Changes page, verify the changes and click Continue.
   e. Click Finish.
5. Back in the Cloudera Manager Home page, select Clusters, Phoenix, and then click the Instances tab.
6. Select the check box by the side of each host.
7. From the Actions for Selected list, select Restart.
8. In the Restart message, confirm restarting the hosts by clicking Restart.
9. Monitor the progress until the *Successfully restarted service* message appears for each restarted host and then click Close.

Enabling Phoenix operations in HBase

Describes the steps that configure the Phoenix service operations in HBase for your Cloudera Observability On-Premises environment. The properties are added in Cloudera Manager using safety valves, which safely enable the changes to the HBase service.

About this task

Steps that enable the Phoenix service for the Cloudera Observability On-Premises environment by safely adding Phoenix properties in the HBase service.

Note: This task must be performed by a user who has either cluster or full administrator privileges.

Procedure

1. Verify that the ZooKeeper maxClientCnxns property was set to 300. For more information on how to set this configuration property, click the Related Information link below.
2. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
3. In Cloudera Manager, select Clusters, HBase, and then click the Configuration tab.
4. Search for the HBase Service Advanced Configuration Snippet (Safety Valve) for hbase-site.xml property and do the following:

   Tip: Entering the full property name in the Search field is not always required. For example, in this case you can enter *snippet* to locate the HBase Service Advanced Configuration Snippet (Safety Valve) for hbase-site.xml property.

   a. Above the Name field of the HBase Service Advanced Configuration Snippet (Safety Valve) for hbase-site.xml property, click View as XML.
   b. In the XML field, add the following either before or after the existing XML:

   Tip: Dragging the bottom right corner downwards increases the size of the field.

```
<property>
   <name>hbase.regionserver.wal.codec</name>
<value>org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec</val
ue>
   <description>Set hbase.regionserver.wal.codec to enable custom Write
Ahead Log ("WAL") edits to be written</description>
</property>
   <property>
<name>hbase.region.server.rpc.scheduler.factory.class</name>
<value>org.apache.hadoop.hbase.ipc.PhoenixRpcSchedulerFactory</value>
   <description>Factory to create the Phoenix RPC Scheduler that uses s
eparate queues for index and metadata updates</description>
</property>
<property>
   <name>hbase.rpc.controllerfactory.class</name>
<value>org.apache.hadoop.hbase.ipc.controller.ServerRpcControllerFactory</
value>
   <description>Factory to create the Phoenix RPC Scheduler that uses sepa
rate queues for index and metadata updates</description>
</property>
<property>
  <name>phoenix.functions.allowUserDefinedFunctions</name>
   <value>true</value>
   <description>enable UDF functions</description>
</property>
<property>
   <name>phoenix.queryserver.serialization</name>
   <value>JSON</value>
   <description>serialization format between client and query server</d
escription>
</property>
<property><name>hbase.server.keyvalue.maxsize</name>
   <value>52428800</value>
```

```
   <description>limits max file size for blobs</description>
</property>
<property>
   <name>phoenix.schema.isNamespaceMappingEnabled</name><value>true</val
ue>
</property>
<property>
   <name>hbase.ipc.server.max.callqueue.size</name>
<value>2147483648</value>
</property>
```

5. Search for the HBase Client Advanced Configuration Snippet (Safety Valve) for hbase-site.xml property and do the following:
   a. Click View as XML.
   b. In the XML field, add the following either before or after the existing XML:

```
<property>
<name>phoenix.functions.allowUserDefinedFunctions</name>
<value>true</value>
</property>
<property>
<name>phoenix.schema.isNamespaceMappingEnabled</name>
<value>true</value>
</property>
```

6. Search for the Write-Ahead Log (WAL) Codec Class property and verify that the property is set to the following value:

```
org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec
```

7. Do the following:
   a. Search for the Maximum Size of HBase Client KeyValue property and set the value to 50 Mib.
   b. Search for the HBase RegionServer Handler Count property and set the value to 40.
   c. Search for the HStore Blocking Store Files property and set the value to 100.
8. If you are installing Cloudera Observability On-Premises on a Kerberos environment, search for the HBase Superusers property and verify that the observability user is added.
9. Click, Save Changes.
10. Back in the Cloudera Manager Home page, select Clusters, Phoenix, and then click the Configuration tab.

    Tip: Clicking the CLOUDERA Manager icon in the upper-left corner takes you back to the Cloudera Manager Home page.

11. Search for the Query Server Advanced Configuration Snippet (Safety Valve) for phoenix-site.xml property by entering *snippet* and do the following:
    a. Click View as XML.
    b. In the XML field, add the following either before or after the existing XML:

```
<property>
   <name>phoenix.queryserver.serialization</name>
   <value>JSON</value>
   <description>serialization format between client and query server</de
scription>
</property>
<property>
   <name>phoenix.schema.isNamespaceMappingEnabled</name>
   <value>true</value>
</property>
```

c. Click, Save Changes.

Note: This step ensures that the configuration setting for the phoenix.schema.isNamespaceMappingEnabled property is consistent on both the client and the server.

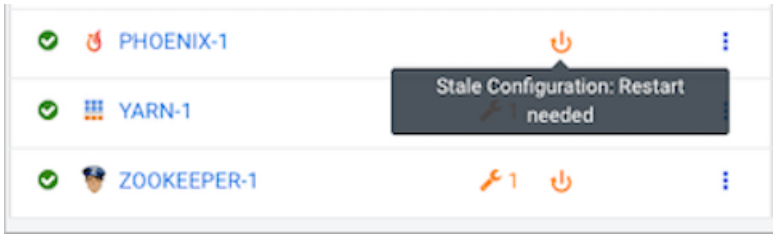
12. Apply your changes by doing the following:

a) Back in the Cloudera Manager Home page, select the HBASE service from the Clusters Status page.
b) From the Actions menu in the HBASE page, select Deploy Client Configuration.
c) In the Deploy Client Configuration message, confirm deployment by clicking Deploy Client Configuration.
d) Monitor the progress of the client's configuration deployment until you see the *successfully deployed* message.
e) Click Close.

13. Back in the Cloudera Manager Home page, restart the HBase and Phoenix services by doing one of the following from the Clusters Status page:

- If no Stale Configuration: Restart needed indicator icon is displayed, do the following:
  a. From the HBase service row, select its vertical ellipses icon, and then select Restart.
  b. Monitor the restart progress until the *Successfully restarted service* message appears and then click Finish.
  c. Repeat steps a and b for the Phoenix service.
- If a Stale Configuration: Restart needed indicator icon is displayed, do the following:
  a. From the HBase or Phoenix service row, click the service's Stale Configuration indicator icon.

  Tip: A Stale Configuration: Restart needed indicator next to a service denotes that the service requires a restart and can be used to restart the service. As shown in the following image:

  b. In the Stale Configurations page, click Restart Stale Services.
  c. In the Restart Stale Services page, select the Re-deploy client configuration check box and click Restart Now.
  d. Monitor the restart progress until the *All requested services successfully restarted* message appears and then click Finish.

Related Information

Configure performance improvement settings

Failure creating a Phoenix schema

Mapping a Phoenix schema to a HBase namespace enables multitenancy. Before running a Phoenix job you must verify that namespace mapping is enabled in the HBase safety valve. Once enabled, tables that are created with the Phoenix schema are mapped to the HBase namespace.

The following example shows a stack track error report that was generated after running a Phoenix job. It shows that the phoenix schema namespace mapping property is not enabled:

```
Role Log
 at org.apache.phoenix.shaded.org.eclipse.jetty.util.thread.strategy.Execut
eProduceCon
sume.executeProduceConsume(ExecuteProduceConsume.java:303)
 at org.apache.phoenix.shaded.org.eclipse.jetty.util.thread.strategy.Exec
uteProduceConsume.produceConsume(ExecuteProduceConsume.java:148)
```

```
 at org.apache.phoenix.shaded.org.eclipse.jetty.util.thread.strategy.Execut
eProduceConsume.run(ExecuteProduceConsume.java:136)
 at org.apache.phoenix.shaded.org.eclipse.jetty.util.thread.QueuedThreadPool
.runJob(QueuedThreadPool.java:671)
 at org.apache.phoenix.shaded.org.eclipse.jetty.util.thread.QueuedThreadPool
$2.run(QueuedThreadPool.java:589)
 at java.lang.Thread.run(Thread.java:748)
Caused by: java.sql.SQLException: ERROR 725 (43M08): Cannot create schema
 because config phoenix.schema.isNamespaceMappingEnabled for enabling name s
pace mapping isn't enabled. schemaName=SIGMA_DB
```

Solution:

Add the following property in the HBase safety valve:

```
<property><name>phoenix.schema.isNamespaceMappingEnabled</name><value>true</
value></property>
```

After adding the property, redeploy the client configurations and restart HBase and the dependent services.

Enabling the Kafka service

The tasks that enable the Apache Kafka service for your Cloudera Observability On-Premises environment.

Enabling the Kafka service operations

Describes the steps that configure the Apache Kafka service operations for your Cloudera Observability On-Premises environment. The properties are enabled and added in Cloudera Manager.

About this task

Steps that enable the Kafka service operations for the Cloudera Observability On-Premises environment by safely adding Kafka properties in Cloudera Manager.

Note: This task must be performed by a user who has either cluster or full administrator privileges.

Before you begin

- Cloudera Observability On-Premises requires at least three Apache Kafka Broker hosts. Verify that you have the hostnames for these instances.
- Ensure that you have added Apache Kafka service to the cluster in Cloudera Manager.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, KAFKA, and then click the Configuration tab.

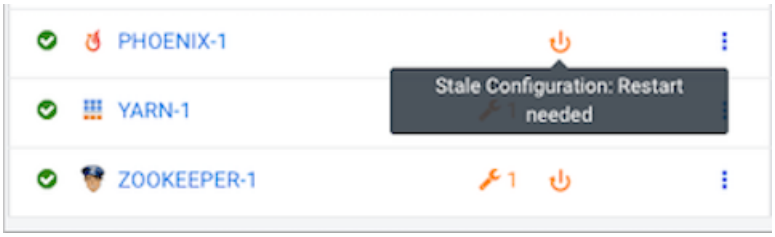
3. Set the Java heap size, partitions, and retention values and record the TLS/SSL port number by doing the following:
   a. Search for the Java Heap Size of Broker property and in the Kafka Broker Default Group field, set the value to a minimum of 4 GB. Cloudera recommends a setting of 8 GB.

      Tip: Entering the full property name in the Search field is not always required. For example, in this case you can enter *java* to locate the Java Heap Size of Broker property.

   b. Search for the Default Number of Partitions property and in the Kafka (Service-Wide) field, set the value to 48.

      Important: This step must be completed.

   c. Increase the Kafka default retention times by doing the following:
      - Search for the Offset Retention Time property and in the Kafka (Service-Wide) field, increase the number of days to 14.
      - Search for the Data Retention Time property and in the Kafka Broker Default Group field, increase the number of days to 14.
      - Search for the Data Retention Check Interval property and in the Kafka Broker Default Group field, increase the time to 1 hr.
   d. Search for the TLS/SSL Port property and do one of the following:
      - If the Kafka Broker Default Group field contains a value, record and store the port number somewhere safe as you will be required to supply this value when you set the OBSERVABILITY service properties.
      - If the Kafka Broker Default Group field is empty, search for the TCP Port property and from the Kafka Broker Default Group field, record and store its port number somewhere safe as you will be required to supply this value when you set the OBSERVABILITY service properties.
   e. Click Save Changes.
4. Restart the KAFKA services, by doing the following:
   a) Select the Status tab and then from the Actions menu, select Deploy Client Configuration.
   b) In the Deploy Client Configuration message, confirm deployment by clicking Deploy Client Configuration.
   c) Monitor the progress of the client's configuration deployment until you see the *successfully deployed client configuration* message.
   d) Click Close.
5. To restart stale services do the following:
   a. Back in the Cloudera Manager Home page, select the Status tab.
   b. Locate a Stale Configuration: Restart needed indicator. As shown in the image below:



   c. Restart the service by clicking the service's vertical ellipses icon, selecting Restart, and in the Stale Configurations page, clicking Restart Stale Services.
   d. In the Restart Stale Services page, select the Re-deploy client configuration check box and click Restart Now.
   e. Monitor the restart progress until the *All requested services successfully restarted* message appears and then click Finish.

Adding the Kafka Broker role

Assign the Kafka Broker role to at least three hosts in the Cloudera Observability On-Premises environment.

About this task

Describes how to assign the Apache Kafka Broker role to at least three hosts.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, KAFKA, and then from the Actions menu, select Add Role Instances.
3. In the Add Role Instances to KAFKA page, click inside the Kafka Broker x *n* field, which opens the Hosts Selected page.
4. Add the Kafka Broker role to at least three hosts by doing the following:
   a. Select the check box by the side of at least three hosts, which adds a Kafka Broker role icon in the Added Roles column for each selected host.
   b. Click OK, which takes you back to the Add Role Instances to KAFKA page where the Kafka Broker x *n* field is now populated with the selected host names.
   c. Click Continue.
   d. In the Review Changes page, verify the changes and click Continue.
   e. Click Finish.
5. Back in the KAFKA page, click the Instances tab.
6. Select the check box by the side of each host.
7. From the Actions for Selected list, select Restart.
8. In the Restart message, confirm restarting the hosts by clicking Restart.
9. Monitor the progress until the *Successfully restarted service* message appears for each restarted host and then click Close.

Adding the Cloudera Observability On-Premises service

The tasks that deploy the Cloudera Observability On-Premises service to all nodes in the Cloudera Observability On-Premises cluster and distribute the Cloudera Observability On-Premises components for optimum performance.

Deploying Cloudera Observability On-Premises

Steps for successfully configuring and installing Cloudera Observability On-Premises on all nodes in the Cloudera Observability On-Premises cluster.

About this task

Describes how to install Cloudera Observability On-Premises on all nodes in the Cloudera Observability On-Premises cluster.

Before you begin

The following tasks must be completed before deploying Cloudera Observability On-Premises.

- Verify that you successfully downloaded, distributed, and activated the OBSERVABILITY parcel.
- Verify that you assigned the Phoenix Query Server role to all the hosts in the Cloudera Observability On-Premises environment, by selecting the Clusters and then Roles in the Cloudera Home page, and then confirming that each host displays the QS icon.

- Optional: If you are not using autoTLS and manually configuring the TLS/SSL protocol, verify that you have the following TLS/SSL key pair values, which you will be required to supply during the deployment task:
  - The location of your TLS/SSL private key file.
  - The location of your TLS/SSL certificate file.
  - The password of your TLS/SSL private key.
- Verify that you recorded a Phoenix and Impala Daemon host name, which you will be required to supply during the deployment task, by doing the following:
  1. From the Cloudera Manager's Home page, select Phoenix and then Instances. Record the host name of one of the Query Server hosts.
  2. From the Cloudera Manager's Home page, select Impala and then Instances. Record the host name of one of the Query Server hosts.

  Tip: Click the CLOUDERA Manager icon to go back to Cloudera Manager's Home page.

Procedure

1. Verify that you are in a supported web browser on the Cloudera Observability On-Premises cluster and have logged in to Cloudera Manager.
2. From the Cloudera Manager's Home page, select the Status tab.
3. From the cluster Actions menu, denoted by the vertical ellipses icon, select Add Service.
4. From the Service Type column in the Service to Cluster page, locate Observability.
5. Select the Observability option and click Continue.

   The Add Observability Service to Cluster 1 page opens.
6. Click Continue.
7. In the Assign Roles panel, verify that the Cloudera Observability On-Premises components are distribute evenly on your cluster nodes for optimum performance by comparing your existing layout with the layout described in the Component distribution for a five node Cloudera Observability On-Premises cluster table. For more information, click the Related Information link below.
8. Click Continue.
9. In the Review Changes panel do the following:
   a. If not visible, display the Phoenix Query Server Host and the Impala Daemon Host value entry fields by clicking the plus (+) icon under Observability (Service-Wide).
   b. In the Phoenix Query Server Host field, enter all the host names that you recorded where the Phoenix Query Server is installed.
   c. In the Impala Daemon Host field, enter the host name of the Impala Daemon that you recorded.
   d. Optional: If you are not using autoTLS and manually configuring the TLS/SSL protocol, do the following:
      1. In the Console Service TLS/SSL Server Private Key File field, enter the location of your TLS private key file.
      2. In the Console Service TLS/SSL Server Certificate File field, enter the location of your TLS certificate file.
      3. In the Console Service TLS/SSL Private Key password field, enter the password of your TLS private key.
      4. For the Console Service TLS/SSL Server CA Certificate field, do nothing by leaving this field blank.
   e. Scroll through the rest of the properties and make your changes.
   f. When satisfied with your changes, click Continue.

   The Cloudera Observability On-Premises service is deployed and configured on the Cloudera Observability On-Premises cluster and its progress is displayed.
10. When completed, as denoted by the Status field displaying Finished, click Continue.
11. In the Summary panel, click Finish.
12. In the Cloudera Manager Home page, select the Status tab and locate the Cloudera Management Service section.

13. From the Actions menu, select Restart.
14. In the Restart message, confirm restarting the Cloudera Management Service by clicking Restart.
15. Monitor the restart progress until the *Successfully restarted service* message appears and then click Close.

Results
On the Cloudera Manager Home page, the OBSERVABILITY service appears in the list of services.

Related Information
Distributing the Cloudera Observability On-Premises components

Distributing the Cloudera Observability On-Premises components

Horizontal scaling improves performance by enabling multiple devices to share the processing and memory workload. Cloudera recommends that you leverage the Cloudera Observability On-Premises cluster resources by installing its components as described.

About this task
Describes how to display your current layout and how to distribute the Cloudera Observability On-Premises services for optimum performance.

The following table lists the components and the layout for a five node cluster. Where,

- One node must include *all* the Cloudera Observability On-Premises component role types.
- The Databus API Server, Databus Server, Analytic Database Server, Baseline Server, Entities Server, SDX Server, and Pipelines Server role types can scale out to multiple nodes. As listed in the Node 2, 3, and 4 columns.
- Due to inter service dependencies, the following role types are grouped. Where, if one of the components is on a host then all the other components in that group must be on the host, which is enforced by Cloudera Manager:
  - Databus API Server and Databus Server.
  - Analytic Database Server, Baseline Server, Entities Server, SDX Server, and Pipelines Server.
  - Admin API Server, API Server, and Console Server.

  For example, if you add a new Databus API Server, you must also add a Databus Server to that node.
- Configure multiple Phoenix Query Server hosts, which reduces bottlenecks. Where, the number of Phoenix Query Server hosts should be proportional to the number of Cloudera Observability On-Premises roles.

  For example, if you have roles on 5 nodes, at least 5 Query Servers are recommended for Phoenix. Cloudera Observability On-Premises internally balances loads on those hosts.

Important: Only one host must be configured for Impala.

Table 11: Component distribution for a five node Cloudera Observability On-Premises cluster

Service	Node 1 (All master components of all services)	Node 2, 3, 4 (Worker nodes + ZooKeeper + Cloudera Observability processing components)	Node 5 (Worker nodes + Cloudera Observability processing components + Cloudera Observability UI)
Cloudera Management	• Alert Publisher • Event Server • Host Monitor • Reports Manager • Service Monitor		

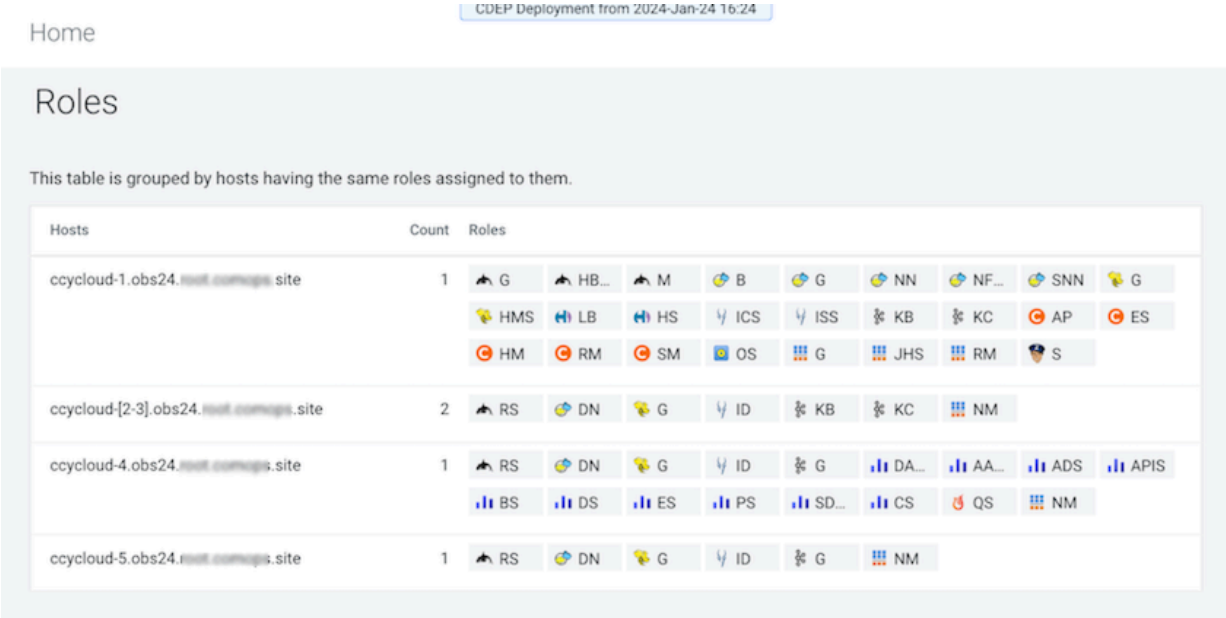
Service	Node 1 (All master components of all services)	Node 2, 3, 4 (Worker nodes + ZooKeeper + Cloudera Observability processing components)	Node 5 (Worker nodes + Cloudera Observability processing components + Cloudera Observability UI)
HBase	• Gateway • Master • Thrift Server (optional)	• Gateway • RegionServer	• Gateway • RegionServer
HDFS	• Balancer • Gateway • NameNode • NFS Gateway (optional) • SecondaryNameNode	• DataNode • Gateway	• DataNode • Gateway
Hive	• Gateway • Metastore Server • HiveServer	• Gateway	• Gateway
Hue (Optional)	• Load Balancer • Hue Server		
Impala	• Catalog Server • StateStore	• Impala Daemon	• Impala Daemon
Kafka		• Kafka Broker	
Phoenix	• Query Server	• Query Server	• Query Server
Observability		• Analytic Database Server • Baseline Server • Databus API Server • Databus Server • Entities Server • Pipelines Server • SDX Server	• Admin API Server • Analytic Database Server • API Server • Baseline Server • Console Server • Databus API Server • Databus Server • Entities Server • Pipelines Server • SDX Server
ZooKeeper		• Server	

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.

2. In Cloudera Manager, select Hosts and then Roles.

   The roles assigned to each node appear as shown in the below example:



3. Compare your existing layout with the layout described in the Component distribution for a five node Cloudera Observability On-Premises cluster table above.
4. To leverage resources, spread the Cloudera Observability On-Premises (observability) roles throughout the cluster. For more information on how to assign roles, see the Cloudera Manager documentation.

Configuring the Kafka broker host and port values

Describes the steps that configure the Apache Kafka broker server host and port values for receiving and sending data for your Cloudera Observability On-Premises environment. The properties are enabled and added in Cloudera Manager.

About this task

After Cloudera Observability On-Premises has been deployed these steps set the Kafka broker host and port values for the Kafka service operations for the Cloudera Observability On-Premises environment.

Note: This task must be performed by a user who has either cluster or full administrator privileges.

Before you begin

Verify that you have set the property values for the Kafka Broker service on at least three hosts. For more information, click the Related Information link below.

Procedure

1. Verify that you are in a supported web browser on the Cloudera Observability On-Premises cluster and have logged in to Cloudera Manager.
2. From the Cloudera Manager Home page, select Clusters, OBSERVABILITY, and then click the Configuration tab.

3. Set the Kafka broker host and port values by doing the following:
   a. Search for the KAFKA Service property and under the OBSERVABILITY (Service-Wide) setting, select the KAFKA check box.
   b. Search for the Dbus Kafka Streams enabled property and verify that the OBSERVABILITY check box is enabled.
   c. Search for the Kafka Broker Host property and in the OBSERVABILITY (Service-Wide) field, enter the name of the host in which you require the Kafka broker server that will receive and send data. Repeat this step to add more host names by clicking the plus sign.

      Important: At least three instances of the Kafka Broker must be installed.

   d. Search for the Kafka Broker Port property and in the OBSERVABILITY field, enter the port number that you saved.
   e. Click Save Changes.
4. Apply your changes and restart the OBSERVABILITY and KAFKA services, by doing the following:
   a) Back in the Cloudera Manager Home page, select the Status tab and then from the Actions menu, select Deploy Client Configuration.
   b) In the Deploy Client Configuration message, confirm deployment by clicking Deploy Client Configuration.
   c) Monitor the progress of the client's configuration deployment until you see the *successfully deployed all client configurations* message.
   d) Click Close.

Related Information

Enabling the Kafka service operations

Granting user access

Cloudera Observability On-Premises supports two authentication methods for granting user access; Local authentication and the Lightweight Directory Access Protocol (LDAP). You configure user access to Cloudera Observability On-Premises in one of these supported authentication methods.

Granting Local authentication

Granting user access using local authentication.

About this task

Describes how to locate the Cloudera Observability On-Premises local authentication directory and user authentication files in Cloudera Manager, and how to add a user, or remove or list existing users using the Console Server executable tool.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, OBSERVABILITY, and then the Configuration tab.

3. Search for the following properties:
   - User Authorization File Directory (user-file.dir), which is the local directory for storing the user authorization file required by the Console Server. By default, /etc/observability/conf.
   - User Authorization File Name (user-file.name), which is the name of the user authorization file required by the Console Server. By default, user-file.json.

     Note: If this file does not exist, it is created during the service startup and is then stored in the directory set by the user-file.dir parameter.
4. In a terminal, SSH to the cluster node that has the Cloudera Observability On-Premises Console role.
5. On the Cloudera Observability On-Premises host, go to the following directory by entering the following command:

   ```
   ${PARCELS_ROOT}/observability/lib/thunderhead-sigma-console
   ```
6. According to your task, enter one of the following commands:
   - To add a user, enter the following command and then follow the prompts to create the user's user name and password:

     ```
     ./onprem-linux user add --user-file user-file.dir user-file.name
     ```
   - To remove a user, enter the following command:

     ```
     ./onprem-linux user remove --user-file user-file.dir user-file.name
     ```
   - To list existing users, enter the following command:

     ```
     ./onprem-linux user list --user-file user-file.dir user-file.name
     ```
7. To access the help for other commands, enter the following command:

   ```
   ./onprem-linux -h
   ```

   Note: You cannot change a user's user name or password, instead you must first remove the user and then recreate the user with their new credentials. Also, if you attempt to edit a nonexistent user file, a prompt appears asking if you would like to create the file.

LDAP authentication properties

Granting user access using the Lightweight Directory Access Protocol (LDAP).

Cloudera Observability On-Premises supports LDAP authentication through the following properties, which are set on the OBSERVABILITY Configuration page of Cloudera Manager:

- Enable LDAP (ldap.enabled)
- LDAP URL (ldap.url)
- LDAP Bind User Distinguished Name (ldap.bind_dn)
- LDAP Bind Password (ldap.bind_password)
- LDAP Group Search Filter (ldap.group_search_filter)
- LDAP Search Base (ldap.search_base)
- LDAP Search Filter (ldap.search_filter)
- LDAP Search Filter Property (ldap.search_filter_property)
- LDAP Server CA Certificate (ldap.ca_cert)

For information on how to set the values for these LDAP properties, click the help icon by the side of each property.

Configuring Telemetry Publisher

Tasks for enabling Cloudera Telemetry Publisher, which collects and transmits diagnostic information about job and query processes to Cloudera Observability On-Premises.

Cloudera Telemetry Publisher is a role in the Cloudera Manager Management Service that collects and sends your workload information to Cloudera Observability On-Premises. For example, when new clusters are added with Cloudera Manager, Telemetry Publisher automatically sends the new cluster information to Cloudera Observability On-Premises.

Note: Cloudera recommends that you assign a dedicated disk for the Telemetry Publisher Service role on your Workload cluster. This prevents any issues when sending data to Cloudera Observability On-Premises from affecting operations other than those performed by Telemetry Publisher.

Depending on the number and size of the jobs run on the cluster, the minimum supported disk drive size is 500GB. This size includes enough disk space for Telemetry Publisher to store data locally when it is unable to send data to Cloudera Observability On-Premises due to connectivity or other issues.

Enabling the telemetry network communication for Cloudera Observability On-Premises

Learn how to enable the network communication between Telemetry Publisher and Cloudera Observability On-Premises.

About this task

Describes how to configure the host URL and access credentials for Telemetry Publisher.

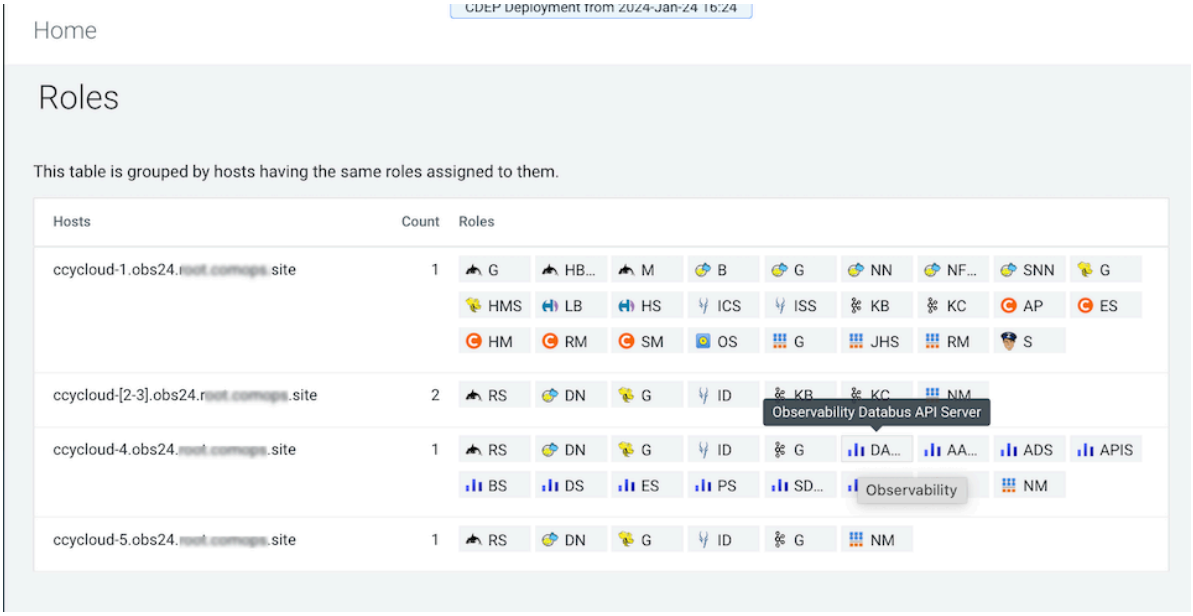
Before you begin

Verify that you have the following values before enabling the Telemetry Publisher service, as you will be required to supply their values during this task.

- The Telemetry Publisher access credentials, which are required to register the Telemetry Publisher account.

- The name of the node that contains the Observability Databus API Server role, by doing the following:
  1. In Cloudera Manager, select Hosts and then Roles.
  2. Search for the Observability Databus API Server role and record its host name. For example:

Figure 1: Roles on the Cloudera Observability On-Premises Cluster



Procedure

1. In a supported web browser on a Workload cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, locate and select Cloudera Management Service, and then select the Configuration tab.
3. Search for the Telemetry Publisher Advanced Configuration Snippet (Safety Valve) for telemetrypublisher.conf property and in its text field, enter the following using the Cloudera Observability On-Premises Database API Server host name that you recorded as a prerequisite for these steps:

```
telemetry.upload.job.logs=true
telemetry.altus.url=http|
https://Databus_API_Server_hostname:Databus_API_Server_port_number
```

   Where,

   - If you have enabled TLS/SSL for the Databus API Server (ssl.enabled), enter https.
   - If you have not enabled TLS/SSL for the Databus API Server, enter http.

   Note: By default, the Databus API Server port number is 12022.

4. Click Save Changes.
5. Back in the Cloudera Manager Home page, from the navigation panel, select Administration and then External Accounts.

   Tip: You can refresh Cloudera Manager and come back to the Cloudera Manager Home page by clicking on the Cloudera Manager icon at the top of the navigation panel.

6. In the External Accounts page, click the Altus Credentials tab, which displays your resource access account certificates.

7. Add a new Telemetry Publisher access account certificate by clicking Add Access Key Authentication and then in the **Add Access Key Authentication** dialog box do the following:
   a) In the Name field, enter an identifiable name for the Telemetry Publisher access key account.
   b) In the Access Key ID field, enter the Telemetry Publisher access key text exactly as provided without trailing spaces.
   c) From the Private Key list, select Choose File and then browse and select your Telemetry Publisher private key file.

      Note: The Cloudera Observability On-Premises Telemetry Publisher credentials are not related to Altus, but act as a pay-wall mechanism to use Cloudera Observability On-Premises.

   d) Click Add, which saves the credentials as an Altus account certificate using the account name you provided and adds it on the Altus Credentials External Accounts page.
8. Back in the Cloudera Manager Home page, from the navigation panel, select Administration and then Settings.
9. Under the Filter CATEGORY section, select Altus, which populates the Settings page with your Telemetry Publisher access key accounts.
10. In the filtered result, select the Telemetry Publisher Altus account credential that you require for this Workload Cluster. In this case, the name you provided in step 7a.
11. Click Save Changes.

(Optional) Renaming the Workload cluster

Describes how to rename the Workload cluster with a human-readable name in Cloudera Manager.

About this task

Cloudera Observability On-Premises identifies the cluster from a random string of 32 characters, such as 44a6e75e-8630-4773-9ea9-6272478e84c2, which is difficult to identify and manage. Cloudera recommends completing the following task to rename your Workload cluster before you add and start the Telemetry Publisher role instance.

Procedure

1. In a supported web browser on a Workload cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters, and then select the Workload cluster that requires a human-readable name.
3. From the Actions menu, select Rename Cluster.
4. In the Name field of the Rename Cluster dialog box, enter a new name that is easily identifiable by you.
5. Click Rename Cluster.

What to do next

Add and start a role instance of the Telemetry Publisher service on the Cloudera Manager Server node.

Adding and starting an instance of Telemetry Publisher for Cloudera Observability On-Premises

Describes how to associate a Workload cluster with Telemetry Publisher by designating a host cluster with the Telemetry Publisher service role and starting the Telemetry Publisher service for Cloudera Observability On-Premises.

About this task

After configuring and adding the Telemetry Publisher credentials, the Telemetry Publisher service must be associated with your Working cluster by adding the Telemetry Publisher service role to a designated host and starting the Telemetry Publisher role instance.

Note: If you are using Java 7, additional steps are required for adding the Telemetry Publisher service role.

Before you begin

The following pre-tasks must be completed before associating a Workload cluster with Telemetry Publisher.

- Verify that you have configured and added the Telemetry Publisher credentials in Cloudera Manager.
- Verify that you have the JCE Policy installed before enabling the role in the Cloudera Manager Service.

Note: If you are using JDK version 1.8.0_160 or earlier, verify that you have installed the JCE policy file as described in the Cloudera Manager documentation.

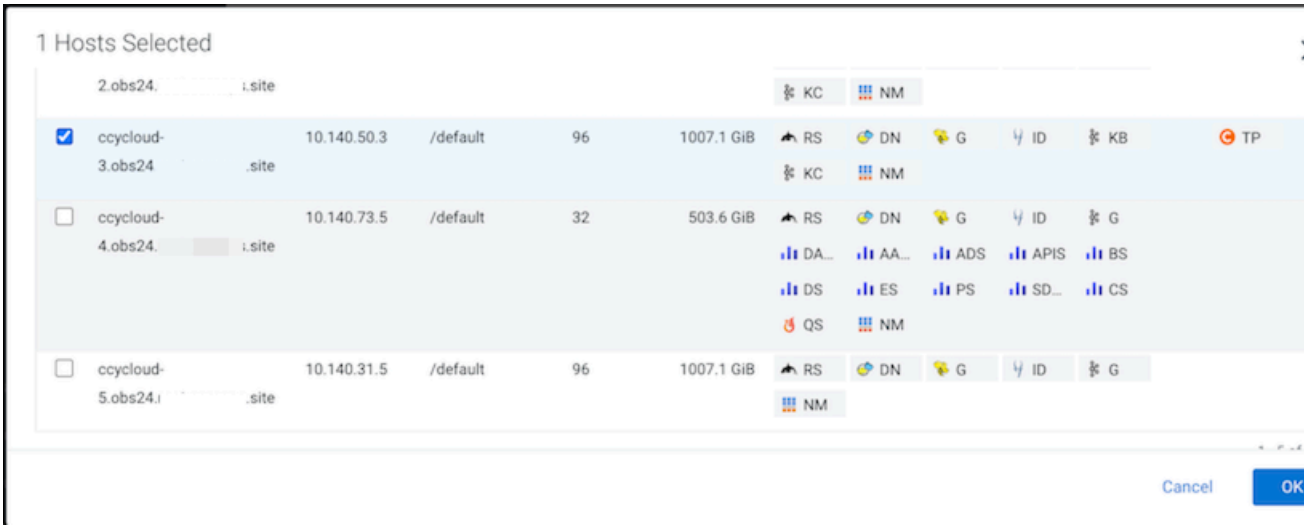
Procedure

1. In a supported web browser on a Workload cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters and then locate and select Cloudera Management Service.

3. Add the Telemetry Publisher role on the Cloudera Manager Server nodes by doing one of the following:
   - If you are using Java 8:
     a. From the Actions menu, select Add Role Instances.

        The Add Role Instances for Cloudera Management Service wizard opens.
     b. Click inside the Telemetry Publisher field.

        The Hosts Selected dialog box opens.
     c. Select the check box of the host for the Telemetry Publisher and click OK, as shown in the below image.



        The wizard populates the Telemetry Publisher field with the selected host name.
     d. Click Continue.
     e. In the Review Changes page, review your selection and click Finish.
   - If you are using Java 7, configure Telemetry Publisher as follows:
     a. Go back to Cloudera Management Service and click the Configuration tab.
     b. Under Scope, select Telemetry Publisher.
     c. In the Search field, enter java configuration, which displays the Java Configuration Options for Telemetry Publisher filter.
     d. In Telemetry Publisher Default Group field, add the following property:

```
-Dhttps.protocols=TLSv1.2 -Dhttps.cipherSuites=TLS_RSA_WITH_AES_256_
CBC_SHA256
```

     e. Click Save Changes.
4. Start Telemetry Publisher by selecting the Instances tab, selecting the Telemetry Publisher check box, and then from the Actions for Selected menu, select Start.
5. In the Start message, confirm starting the Telemetry Publisher Service on your cluster by clicking Start.
6. Monitor the progress until the *Successfully started service* message appears and then click Close.

Post-installation tasks

Optional tasks that can be completed after installing Cloudera Observability On-Premises.

HDFS file access requirements

Describes how to access files from HDFS with Telemetry Publisher when your access keys are stored in the Cloudera Key Trustee Server.

By default, when keys are stored in the Key Trustee Server, the HDFS user for Telemetry Publisher (hdfs) does not have permission to access files.

To enable access to your files in HDFS, the Telemetry Publisher user must belong to the user groups that authenticate user access for the Job History Server and the Spark History Server. For example, if the hadoop user group authenticates access for the Job History Server and the spark user group authenticates access for the Spark History Server, then the Telemetry Publisher user must belong to the hadoop group and the spark group to download files from HDFS.

Adding a proxy server

Steps for configuring a proxy server, which adds extra security by enabling an intermediary gateway for sending your workload data to Cloudera Observability On-Premises.

About this task

Describes how to add a proxy server as an intermediary gateway.

Note: You cannot upload data from Amazon Web Services (AWS) using a proxy server.

You can configure the Telemetry Publisher service to send data by way of a proxy server for database and metric data uploads. By default, this configuration property is disabled.

Telemetry Publisher uses the TLS and HTTPS protocols to send telemetry information to Cloudera Observability On-Premises, which ensures that the data is encrypted. The proxy you use must support the HTTP CONNECT method to be able to pass through the encrypted messages. For more information, see the associated HTTP CONNECT Request for Comments (RFC) document.

Note: Telemetry Publisher support for proxy servers is only available in Cloudera Manager version 5.16.2 and higher.

Procedure

1. In a supported web browser on a Workload cluster, log in to Cloudera Manager with administrator privileges.
2. In Cloudera Manager, select Clusters, locate and select Cloudera Management Service, and then select the Configuration tab.
3. From the Filters panel in the SCOPE section, select Telemetry Publisher.
4. In the Search field, enter proxy, which displays the proxy configuration properties.
5. In the Proxy Support for Telemetry Publisher property, select the Telemetry Publisher Default Group check box and do the following:
   a. In the Proxy Server field, enter the proxy server name.
   b. In the Proxy Port field, enter the port number for the proxy server.
   c. In the Proxy User field, enter the proxy server user name, which is used for access authentication.
   d. In the Proxy Password field, enter the password for the proxy server user name.

Note: If these properties do not appear, search for the Java Configuration Options for Telemetry Publisher property and in its entry field, enter the following:

```
-Djdk.http.auth.tunneling.disabledSchemes=""
```

6. Click Save Changes, and then restart the Telemetry Publisher service.

Enabling the Auto Actions feature in Telemetry Publisher

Steps for enabling the Cloudera Observability On-Premises Auto Actions feature in the Telemetry Publisher service.

About this task

Describes how to access and enable the Telemetry Publisher Auto Actions property settings.

Note: This task must be performed by a user who has either cluster or full administrator privileges.

Procedure

1. Verify that you have enabled the Telemetry Publisher service.
2. In a supported web browser, log in to Cloudera Manager as a user with full system administrator privileges.
3. From the Cloudera Manager navigation panel, click Clusters and then scroll down, locate, and select Cloudera Management Service.
4. In the Status Summary section on the Cloudera Management Service page, click Telemetry Publisher.
5. In the Telemetry Publisher page, select the Configuration tab.
6. In the Alert message that appears, click Continue Editing Role Instance.
7. In the Search field, enter Telemetry Publisher Advanced Configuration Snippet (Safety Valve) for telemetrypublisher.conf.

   Tip: Entering the full property name in the Search field is not mandatory. For example, in this case you can enter *safety* to locate the Telemetry Publisher Advanced Configuration Snippet (Safety Valve) for telemetrypublisher.conf configuration property.
8. In the Telemetry Publisher Default Group text box, enter the Telemetry Publisher Auto Actions configuration property settings that you require.

   For example, to enable Auto Actions for Spark applications, set the following:

   ```
   autoactions.yarn.app.collector.enabled=true
   autoactions.collection.spark.enabled=true
   ```

   For the full list of available Auto Actions property settings for Telemetry Publisher, see *Telemetry Publisher configuration settings for Auto Actions*.
9. Click Save Changes.
10. Restart the Telemetry Publisher service on your cluster by doing the following:
    a) Go back to the Cloudera Manager Home page.

       Tip: Clicking the CLOUDERA Manager icon in the upper-left corner takes you back to the Cloudera Manager Home page.

    b) From the Cloudera Manager Navigation panel, click Clusters and then scroll down, locate, and select Cloudera Management Service.
    c) In the Status Summary section on the Cloudera Management Service page, click Telemetry Publisher.
    d) From the Actions menu, select Restart this Telemetry Publisher.
    e) In the Restart message, confirm restarting the Telemetry Publisher Service on your cluster by clicking Restart.
    f) Monitor the restart progress until the Successfully restarted role message appears and then click Close.

Related Information

Triggering action alerts across jobs and queries

Telemetry Publisher configuration settings for Auto Actions

Lists the Cloudera Observability On-Premises Telemetry Publisher configuration property settings. Set in the Telemetry Publisher Safety Valve section in Cloudera Manager they enable Telemetry Publisher to collect data that is required by the Auto Actions feature.

Table 12: Telemetry Publisher configuration settings for Auto Actions

Property Name	Default Value	Description	Example
autoactions.yarn.app.collector.enabled	FALSE	• When set to true, the Telemetry Publisher collector is enabled and tests for workloads related to YARN. • When set to false, no YARN related collection configurations are considered.	autoactions.yarn.app.collector.enabled=true
autoactions.impala.collector.enabled	FALSE	• When set to true, the Telemetry Publisher collector is enabled and tests for workloads related to Impala. • When set to false, no Impala related collection configurations are considered.	autoactions.impala.collector.enabled=true
autoactions.collection.yarn.enabled	FALSE	• When set to true, the Telemetry Publisher collector evaluates all the YARN application Auto Actions. This is the starting point for an Auto Actions evaluation. • When set to false, no YARN application Auto Actions are evaluated on the cluster.	autoactions.collection.yarn.enabled=true
autoactions.collection.mr.enabled	FALSE	• When set to true, the Telemetry Publisher collector evaluates all the MapReduce Auto Actions. This is the starting point for an Auto Actions evaluation. • When set to false, no MapReduce Auto Actions are evaluated on the cluster.	autoactions.collection.mr.enabled=true
autoactions.collection.spark.enabled	FALSE	• When set to true, the Telemetry Publisher collector evaluates all the Spark application Auto Actions. This is the starting point for an Auto Actions evaluation. • When set to false, no Spark application Auto Actions are evaluated on the cluster.	autoactions.collection.spark.enabled=true
autoactions.collection.hive.enabled	FALSE	• When set to true, the Telemetry Publisher collector evaluates all the Hive query Auto Actions. This is the starting point for an Auto Actions evaluation. • When set to false, no Hive query Auto Actions are evaluated on the cluster.	autoactions.collection.hive.enabled=true
autoactions.definition.cache.refresh.minutes	5	This setting controls the number of minutes an Auto Action definition is stored in cache on the cluster. Your Auto Action definitions can be stored in cache on the cluster. As an Auto Action definition rarely changes, setting this value increases the delivery speed of requests by reducing the number of calls to Cloudera Observability On-Premises.	autoactions.definition.cache.refresh.minutes=5

Accessing the Cloudera Observability On-Premises web user interface URL

Steps for accessing the Cloudera Observability On-Premises web user interface URL for the first time.

About this task

Describes how locate the Cloudera Observability On-Premises web UI URL.

Before you begin

Verify that the following is completed:

- Cloudera Observability On-Premises is installed.
- Telemetry Publisher is enabled for Cloudera Observability On-Premises and your Workload clusters are associated with the service.
- Cloudera Manager is connected to Cloudera Observability On-Premises.

Note: This task is performed by a user who is assigned full access rights and system administrator privileges across all clusters within the Cloudera Observability On-Premises environment.

Procedure

1. In a supported web browser on the Cloudera Observability On-Premises cluster, log in to Cloudera Manager.
2. In Cloudera Manager, select Clusters and then OBSERVABILITY.
3. Do one or more of the following:
   - To forward the Cloudera Observability On-Premises web UI URL to your Cloudera Observability On-Premises users. Right-click on the Observability UI tab and copy and paste its URL link into an email and send to the Cloudera Observability On-Premises users.
   - To test the Cloudera Observability On-Premises URL and web user interface. Click the Go To icon in the Observability UI tab, which opens the Cloudera Observability On-Premises login page for you to enter and log in to the Cloudera Observability On-Premises web UI with your Cloudera Observability On-Premises user name and password access credentials.

     Tip: By default, for testing purposes, you can enter admin for both the user name and password. Cloudera recommends changing this default setting as soon as possible.

   The Cloudera Observability On-Premises web UI Landing page opens to the main navigation panel.
4. Testing tasks:
   a. From the main navigation panel, select **Financial Governance**, which opens the **Chargeback** page. When configured, by you, this page displays the total costs and the hourly CPU and memory usage for all of your cost centers, including the unutilized resource usage costs from the **Uncategorised** section. For more information about the Financial Governance feature and how to configure your cost centers and assign them to your resources, click the Related Information link below.

b. To verify that all the Workload clusters in your Cloudera Observability On-Premises environments are visible in the Cloudera Observability On-Premises web UI and are accessible by Telemetry Publisher and Cloudera Observability On-Premises.

1. From the main navigation panel, select Analytics.

   The Cloudera Observability On-Premises **Environments** page opens displaying the Workload clusters in your environment.
2. Select an environment required for analysis.

   The **Environment** navigation panel opens, which hierarchically lists the environment's cluster, engines, and if applicable the Hive Metastore category.

   Note: If no cluster names appear or a specific Workload cluster is not displaying, verify that you have associated Cloudera Manager Telemetry Publisher with the Workload cluster.

c. To view a Workload cluster's workload metrics, do the following:

1. Verify that the **Cluster Summary** page is displayed for the environment's cluster required for analysis.

   The **Cluster Summary** page, which is displayed as the title in your browser's tab, displays total, failed, and slow jobs or queries and the engine in which they were executed, and performance trends about the processed jobs and queries. It also enables you to view historical trends for analysis when you select a predefined or custom time period from the Time-Range filter list.
2. From the cluster's ENGINES, select a workload engine of interest. When an engine is selected, the engine's page opens. The name of the engine is displayed in the browser tab and the engine's chart widgets display information about the workload jobs run by the selected engine, such as which jobs or queries have failed or are slow, their processing time, missed SLAs (thresholds), user and pool metrics, and outlier issues.
3. In the workload engine's page, review its chart widgets and then select a chart widget, such as Suboptimal. Select a link or bar and drill down further to view more information, such as health checks, execution details, baselines, and trends.

   Tip: Breadcrumbs are displayed at the top of each page, which displays the name of your current location and its preceding page levels. You can move between these levels by clicking on a breadcrumb location.

Related Information

Analyzing your environment costs with Cloudera Observability On-Premises

About the Cloudera Observability On-Premises user interface hierarchy