Cloudera Management Console

AWS Environments

Date published: 2019-08-22
Date modified: 2025-08-18

CLOUDERA

https://docs.cloudera.com/

Legal Notice

Contents

Working with AWS environments

Refer to the following documentation to learn about creating and managing AWS environments in Cloudera:

Related Information
Managing provisioning credentials for AWS
Managing Data Lakes
Managing FreeIPA

Introduction to AWS environments

In Cloudera, an environment is a logical subset of your cloud provider account including a specific virtual private network. You can register as many environments as you require.

The "environment" concept of Cloudera is closely related to the virtual private network in your cloud provider account. Registering an environment provides Cloudera with access to your cloud provider account and identifies the resources in your cloud provider account that Cloudera services can access or provision. A single environment is contained within a single cloud provider region, so all resources deployed by Cloudera are deployed within that region within one specific virtual network. Once you've registered an environment in Cloudera, you can start provisioning Cloudera resources such as clusters, which run on the physical infrastructure in an AWS data center.

The following diagram enumerates the components of an environment:

AWS environment				
Virtual network and subnet	Credential using cross account IAM role	SSH key pair	Cloud storage location	Data Lake cluster
Security groups			Instance profile	FreeIPA server
			S3 Guard	
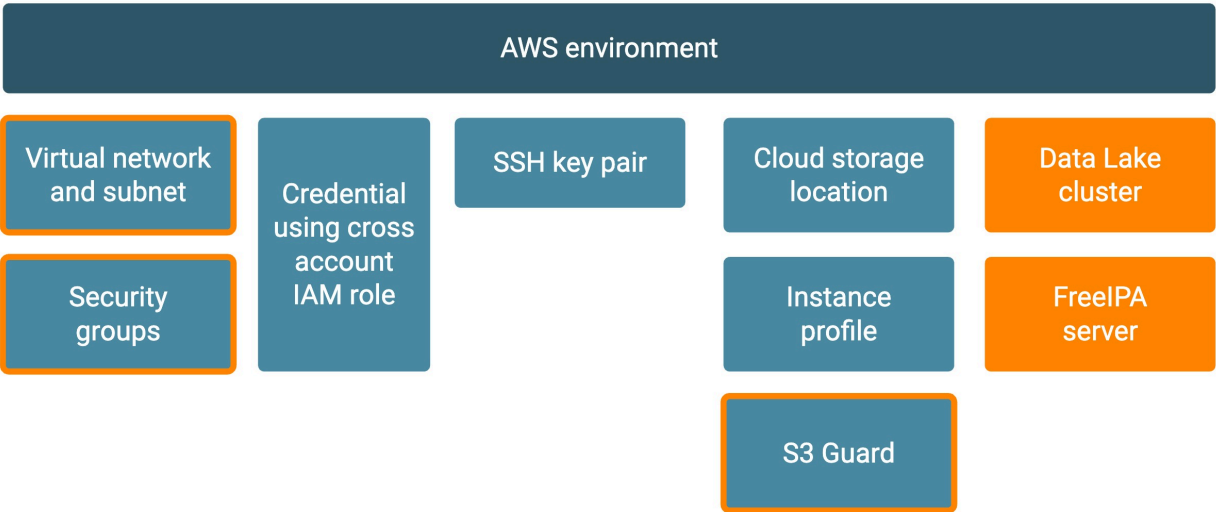
Can optionally be created by Cloudera admin or Cloudera can create it automatically.

Always created by Cloudera admin.

Always created by Cloudera.

The diagram illustrates all major user-created and Cloudera-created components of an environment:

- The items in dark blue boxes with orange outlines can either be automatically provisioned by Cloudera on your AWS account, or your can optionally pre-create them and provide them when registering an environment.
- The items in dark blue boxes must be pre-created by your Cloudera administrator prior to environment registration and then provided when registering an environment.
- The items in orange boxes are automatically provisioned on AWS by Cloudera as part of environment provisioning.

Note: The items that are user-created don't get terminated during environment deletion.

As shown in the diagram, an environment consists of the following resources:

Environment component	Description
Virtual network with subnets	An environment corresponds to one specific virtual network and subnets in which Cloudera resources are provisioned. The network is located within one specified region; therefore, you also need to select a region for each environment.
Security groups	Security groups act as a virtual firewall for your instances to control inbound and outbound traffic. All VM instances provisioned within an environment use your specified security access settings allowing inbound access to your instances from your organization's computers.
Cross-account role for provisioning credential	Cloudera uses a provisioning credential for authorization to provision resources (such as compute instances) within your cloud provider account. On AWS, the credential uses a cross-account IAM role with an attached IAM policy listing all required permissions.
SSH public key	When registering an environment on a public cloud, a Cloudera administrator provides an SSH public key. This way, the administrator has root-level access to the Data Lake instance and Cloudera Data Hub cluster instances.
Cloud storage location and AIM roles and instance profiles allowing access to that location	When registering an environment, you must provide one or more S3 buckets for storing: • All workload cluster data • Ranger audits, and FreeIPA logs • Cluster service logs You must also provide IAM instance profiles that control access to the bucket(s).
Data lake	A data lake is automatically provisioned when an environment is created. It provides a mechanism for storing, accessing, organizing, securing, and managing data.
FreeIPA	A FreeIPA server is automatically provisioned when an environment is created. It is responsible for synchronizing your users and making them available to Cloudera services, Kerberos service principal management, and more.

You may want to register multiple environments corresponding to different regions that your organization would like to use. Once your environment is running, you can provision Cloudera Data Hub clusters, Cloudera Data Warehouses, and other resources in it.

Registering an AWS environment from the Cloudera UI

Once you've met the AWS cloud provider requirements, register your AWS environment.

Before you begin

This assumes that you have already fulfilled the environment prerequisites described in AWS requirements.

Required role: EnvironmentCreator

Steps

1. Navigate to the Cloudera Management Console > Environments > Register environment:
2. On the Register Environment page, provide the following information:

Parameter	Description
General Information	
Environment Name (Required)	Enter a name for your environment. The name: • Must be between 5 and 28 characters long. • Can only include lowercase letters, numbers, and hyphens. • Must start with a lowercase letter.
Description	Enter a description for your environment.
Select Cloud Provider (Required)	Select Amazon.
Credential (Required)	
Select Credential	Select an existing credential or select Create new credential. For instructions on how to create a credential, refer to Creating a role-based credential. **Note:** Activate the Enable Permission Verification button if you want Cloudera to check permissions for your credential. Cloudera will verify that you have the required permissions for your environment.

3. Click Next.
4. On the Data Access and Data Lake Scaling page, provide the following information:

Parameter	Description
Data Lake Settings	
Data Lake Name (Required)	Enter a name for the Data Lake cluster that will be created for this environment. The name: • Must be between 5 and 100 characters long • Must contain lowercase letters • Cannot contain uppercase letters • Must start with a letter • Can only include the following accepted characters are: a-z, 0-9, -.
Architecture	Select the Architecture type you want to use for the FreeIPA, Data Lake cluster and RDS. You have the option to choose between the following types: • X86_64 • ARM64 You can choose different architecture types for FreeIPA, Data Lake, RDS and Cloudera Data Hub clusters. For example, you can select X86_64 for the Cloudera environment, but ARM64 for the Cloudera Data Hub cluster. **Note:** The following limitations apply when choosing ARM architecture: • ARM architecture is only available for Cloudera Runtime 7.3.1.500 for FreeIPA, Data Lake and RDS. • Ensure that the EC2 instances with ARM processors are supported in your region. For more information, see the AWS Graviton Processors and Amazon EC2 instance types by Region documentation.
Data Lake Version (Required)	Select Cloudera Runtime version that should be deployed for your Data Lake. The latest stable version is used by default. All Cloudera Data Hub clusters provisioned within this Data Lake will be using the same Cloudera Runtime version.
Fine-grained access control on S3	

Parameter	Description
Enable Ranger authorization for AWS S3 Identity	Enable this if you would like to use Fine-grained access control. Next, from the Select AWS IAM role for Ranger authorizer dropdown, select the DATALAKE_ADMIN_ROLE IAM role created in Minimal setup for cloud storage.
Data Access and Audit	
Assumer Instance Profile (Required)	Select the IDBROKER_ROLE instance profile created in Minimal setup for cloud storage.
Storage Location Base (Required)	Provide the S3 location created for data storage in Minimal setup for cloud storage.
Data Access Role (Required)	Select the DATALAKE_ADMIN_ROLE IAM role created in Minimal setup for cloud storage.
Ranger Audit Role (Required)	Select the RANGER_AUDIT_ROLE IAM role created in Minimal setup for cloud storage.
IDBroker Mappings	We recommend that you leave this out and set it up after registering your environment as part of Onboarding Cloudera users and groups for cloud storage. **Note:** If you are using Fine-grained access control, this option is disabled, because you should onboard your users and groups via Ranger instead of using IDBroker mappings.
Scale (Required)	Select Data Lake scale. By default, "Light Duty" is used. For more information on Data Lake scale, refer to Data Lake scale.
Enable **Compute Cluster**	Enable Compute Clusters if you would like to deploy a containerized platform on Kubernetes for data services and shared services.

5. Click on Advanced Options to make additional configurations for your Data Lake. The following options are available:

Parameter	Description
Network and Availability	
Enable Multiple Availability Zones for Data Lake	Click the Enable Multiple Availability Zones for Data Lake toggle button to enable multi-AZ for the Data Lake. This option is disabled by default and is only available when a Medium Duty Data Lake is selected. Refer to Deploying Cloudera in multiple AWS availability zones.
Hardware and Storage	For each host group you can specify an instance type. For more information on instance types, see Amazon EC2 instance types.
Cluster Extensions	
Recipes	You can optionally select and attach previously registered recipes to run on a specific Data Lake host group. For more information, see Recipes.

6. Click Next.

7. On the Region, Networking and Security page, provide the following information:

Parameter	Description
Region	
Select Region (Required)	Select the region that you would like to use for Cloudera. If you would like to use a specific existing virtual network, the virtual network must be located in the selected region.
Customer-managed Keys	
Enable Customer-Managed Keys	Enable this if you would like to provide a Customer-Managed Key (CMK) to encrypt environment's disks and databases. Next, under Select Encryption Key, select an existing CMK. For more information, refer to Customer managed encryption keys.
Select Encryption Key	Select an existing CMK.
Network	
Select Network (Required)	Select the existing virtual network where you would like to provision all Cloudera resources. Refer to VPC and subnet.

Parameter	Description
Select Subnets (Required)	This option is only available if you choose to use an existing network. Multiple subnets must be selected and Cloudera distributes resources evenly within the subnets.
Enable Public Endpoint Access Gateway	When Cluster Connectivity Manager is enabled, you can optionally enable Public Endpoint Access Gateway to provide secure connectivity to UIs and APIs in Data Lake and Cloudera Data Hub clusters deployed using private networking.
	Under Select Endpoint Access Gateway Subnets, select the public subnets for which you would like to use the gateway. The number of subnets must be the same as under Select Subnets and the availability zones must match. For more information, refer to Public Endpoint Access Gateway documentation.
Proxies	
Select Proxy Configuration	Select a proxy configuration if previously registered. For more information refer to Setting up a proxy server.
Security Access Settings	
Select Security Access Type (Required)	This determines inbound security group settings that allow connections to the Data Lake and Cloudera Data Hub clusters from your organization's computers. You have two options:
	• Create new security groups - Allows you to provide custom CIDR IP range for all new security groups that will be created for the Data Lake and Cloudera Data Hub clusters so that users from your organization can access cluster UIs and SSH to the nodes.
	This must be a valid CIDR IP in IPv4 range. For example: 192.168.27.0/24 allows access from 192.168.27.0 through 192.168.27.255. You can specify multiple CIDR IP ranges separated with a comma. For example: 192.168.27.0/24,192.168.28.0/24
	If you use this setting, several security groups will get created: one for each Data Lake host group the Data Lake and one for each host group), one for each FreeIPA host group, and one for RDS; Furthermore, the security group settings specified will be automatically used for Cloudera Data Hub, Cloudera Data Warehouse, and Cloudera AI clusters created as part of the environment.
	• Provide existing security groups (Only available for an existing VPC) - Allows you to select two existing security groups, one for Knox-installed nodes and another for all other nodes. If you select this option, refer to Security groups to ensure that you open all ports required for your users to access environment resources.
Kubernetes	
Select **Private Kubernetes Cluster** or provide **Authorized IP Ranges**	If you have enabled Compute Clusters, you have the following options to configure the necessary networking information for the Kubernetes cluster:
	• Enable Private Kubernetes Cluster to create a private cluster that blocks all access to the API Server endpoint.
	• Provide the CIDRs to the Kubernetes API Server Authorized IP Ranges field to specify a set of IP ranges that will be allowed to access the Kubernetes API server.
	You need to provide the advanced configurations only once when creating your environment. The configurations will be applied to all compute clusters in the environment.
Worker Node Subnets	Uses the same set of subnets provided in **Network** section. You have the option to not use all of the previously provided subnets.
SSH Settings	
New or existing SSH public key (Required)	You have two options for providing a public SSH key:
	• Select a key that already exists on your AWS account within the specific region that you would like to use.
	• Upload a public key directly from your computer.
	Note: Cloudera does not use this SSH key. The matching private key can be used by your Cloudera administrator for root-level access to the instances provisioned for the Data Lake and Cloudera Data Hub.
Add tags	You can optionally add tags to be created for your resources on AWS. Refer to Defining custom tags.

8. Click on Advanced Options to make additional configurations for the FreeIPA cluster. The following options are available:

Parameter	Description
Network and Availability	

Parameter	Description
Enable Multiple Availability Zones for Data Lake	Click the Enable Multiple Availability Zones for Data Lake toggle button to enable multi-AZ for the FreeIPA cluster. Refer to Deploying Cloudera in multiple AWS availability zones.
Hardware and Storage	For each host group you can specify an instance type. For more information on instance types, see Amazon EC2 instance types.
Cluster Extensions	
Recipes	You can optionally select and attach previously registered recipes to run on a specific FreeIPA host group. For more information, see Recipes.

9. Click Next.
10. On the Storage page, provide the following information:

Parameter	Description
Logs	
Logger Instance Profile (Required)	Select the LOG_ROLE instance profile created in Minimal setup for cloud storage.
Logs Location Base (Required)	Provide the S3 location created for log storage in Minimal setup for cloud storage.
Backup Location Base	Provide the S3 location created for FreeIPA and Data Lake backups in Minimal setup for cloud storage. If not provided, the default Backup Location Base uses the Logs Location Base.
Telemetry	
Enable Workload Analytics	Enables Cloudera Observability support for workload clusters created within this environment. When this setting is enabled, diagnostic information about job and query execution is sent to Cloudera Observability. For more information, refer to Enabling workload analytics and logs collection.
Enable Deployment Cluster Logs Collection	When this option is enabled. the logs generated during deployments will be automatically sent to Cloudera. For more information, refer to Enabling workload analytics and logs collection.

11. Click on Register Environment to trigger environment registration.
12. The environment creation takes about 60 minutes. The creation of the FreeIPA server and Data Lake cluster is triggered. You can monitor the progress from the web UI. Once the environment creation has been completed, its status will change to "Running".

After you finish

After your environment is running, perform the following steps:

• You must assign roles to specific users and groups for the environment so that selected users or user groups can access the environment. Next, you need to perform user sync. For steps, refer to Enabling admin and user access to environments.

• You must onboard your users and/or groups for cloud storage. For steps, refer to Onboarding Cloudera users and groups for cloud storage.

• You must create Ranger policies for your users. For instructions on how to access your Data Lake, refer to Accessing Data Lake services. Once you've accessed Ranger, create Ranger policies to determine which users have access to which databases and tables.

Registering an AWS environment from the CDP CLI

Use the CDP CLI to register your AWS environment.

About this task

Unlike in the Cloudera web interface, environment creation in the CDP CLI is a three-step process: creating the environment, setting DBroker mappings, and creating data lakes. Follow these steps to create an AWS environment in Cloudera.

Before you begin

l the environment prerequisites described in AWS requirements

> **Note:** You can use the cdp environments   create-aws-environment help command to get more information about specific CLI command options

Procedure

1. Register your AWS environment in Cloudera with the create-aws-environment command and required input parameters.

```
cdp environments create-aws-environment
--custom-docker-registry '{c-dock}'
--cli-input-json '{
    "environmentName": "test-env",
    "description": "Test AWS environment",
    "credentialName": "test-aws-crd",
    "region": "us-west2",
    "publicKey": "ssh-rsa AAAAB3NzaZ1yc2EAAAADAQABAAABAQDwCI/wmQzbNn9YcA
8vdU+Ot41IIUWJfOfiDrUuNcULOQL6ke5qcEKuboXzbLxV0YmQcPFvswbM5S4FlHjy2VrJ5s
pyGhQajFEm9+PgrsybgzHkkssziX0zRq7U4BVD68kSn6CuAHj9L4wx8WBwefMzkw7uO1Ckfi
fIp8UE6ZcKKKwe2fLR6ErDaN9jQxIWhTPEiFjIhItPHrnOcfGKY/p6OlpDDUOuMRiFZh7qMz
fgvWI+UdN/qjnTlc/M53JftK6GJqK6osN+j7fCwKEnPwWC/gmy8El7ZMHlIENxDut6X0qj9O
kc/JMmG0ebkSZAEbhgNOBNLZYdP0oeQGCXjqdv",
    "enableTunnel": true,
    "usePublicIp": true,
    "existingNetworkParams": {
        "networkName": "eng-private",
        "subnetNames": [
            "private-us-west2"
        ],
        "sharedProjectId": "dev-project"
    },
    "logStorage": {
        "storageLocationBase": "gs://logs",
        "serviceAccountEmail": "logger@dev-project.iam.gserviceaccount.c
om"
    }
}'
```

Parameter	Description
custom-docker-registry	The CRN of the desired custom docker registry for data services to be used.
environmentName	Provide a name for your environment.
credentialName	Provide the name of the credential created earlier.
region	Specify the region where your existing VPC network is located. For example, "us-west2" is a valid region.
publicKey	Paste your SSH public key.

Parameter	Description

existingNetworkParams	Provide a JSON specifying the following:
	
	``
	{
	  "networkName": "string",
	  "subnetNames": ["string", ...],
	  "sharedProjectId": "string"
	}
	``
	
	Replace the values with the actual VPC network name, one or more subnet names, and shared project ID.
	
	The sharedProjectId value needs to be set as follows:
	
	• For a shared VPC, set it to the AWS host project ID.
	• For a non-shared VPC, set it to the AWS project ID of the project where Cloudera is being deployed.

enableTunnel	Enable and disable the Cluster Connectivity Manager. The default value is "true" (enabled). Set it to "false" to disable it. If disabled, you must specify two security groups in your JSON definition:
	
	``
	"securityAccess":
	  {
	    "securityGroupIdForKnox": "string",
	    "defaultSecurityGroupId": "string"
	  }
	``

usePublicIp	Set this to "true" to create public IPs or "false" to use private IPs.

logStorage	Provide a JSON definition specifying your configuration for the cluster and audit logs:
	
	``
	{
	  "storageLocationBase": "string",
	  "serviceAccountEmail": "string"
	}
	``
	
	The storageLocationBase must follow this format: gs://my-bucket-name.

> **Note:**
>
> CDP CLI includes the create-aws-environment   --generate-cli-skeleton command option to generate a CLI JSON template.

2. Once your environment is running, set the IDBroker Mappings with the set-id-broker-mappings command.

```
cdp environments set-id-broker-mappings \
  --environment-name test-env \
  --data-access-role dl-admin@dev-project.iam.gserviceaccount.com \
  --ranger-audit-role ranger-audit@dev-project.iam.gserviceaccount.com \
  --mappings '[{"accessorCrn": "crn:altus:iam:us-west-1:45ca3068-42a6-422
7-8394-13a4493e2ac0:user:430c534d-8a19-4d9e-963d-8af377d16963", "role":
"data-science@dev-project.iam.gserviceaccount.com"},{"accessorCrn":"crn:
altus:iam:us-west-1:45ca3068-42a6-4227-8394-13a4493e2ac0:machineUser:mfox-
aws-idbmms-test-mu/2cbca867-647b-44b9-8e41-47a01dea6c19","role":"data-eng@
dev-project.iam.gserviceaccount.com"}]'
```

Parameter	Description
datalakeName	Provide a name for your Data Lake.
environmentName	Provide the name of the environment created earlier.

Parameter	Description
scale	Provide Data Lake scale. It must be one of: • LIGHT_DUTY or • ENTERPRISE
cloudProviderConfiguration	Provide the name of the data storage bucket and the email of the IDBroker service account.

3. Create the data lake clusters within the environment with the create-aws-datalake command.

```
cdp datalake create-aws-datalake --datalake-name "NAME" --environment-name
  "ENVNAME" --cloud-provider-configuration instanceProfile="INSTANCEPROFILE
",storageBucketLocation="s3://MYBUCKET" --scale MEDIUM_DUTY_HA  --runtime
  7.2.7
```

Parameter	Description
environment-name	Specify the name of the environment created earlier.
data-access-role	Specify the email address of the data lake admin service account created earlier.
ranger-audit-role	Specify the email address of the Ranger audit service account created earlier.

mappings	Map Cloudera users or groups to the GCP service accounts created earlier. Use the following syntax:
	
	``
	[
	{
	"accessorCrn": "string",
	"role": "string"
	}
	...
	]
	``
	
	You can obtain a user or group CRN from the Management Console > User Management by navigating to the details of a specific user or group.
	
	The role should be specified as a service account email.

4. Verify that your new environment is running:

```
cdp environments list-environments
```

5. Verify the status of the data lake:

```
cdp datalake list-datalakes --environment-name ${ENVNAME}
```

6. Sync the IDBroker Mappings:

```
cdp environments sync-id-broker-mappings --environment-name demo3
```

7. Verify the sync status:

```
cdp environments get-id-broker-mappings-sync-status --environment-name d
emo3
```

What to do next

Once your environment is running:

- You must assign roles to users and groups to allow them access to the environment and perform user sync. For steps, refer to Enabling admin and user access to environments.
- You must onboard your users and/or groups for cloud storage. For steps, refer to Onboarding Cloudera users and groups for cloud storage.
- You must create Ranger policies to determine which users have access to which databases and tables. For instructions on how to access your data lake, refer to Accessing Data Lake services.
- You can use the update-custom-docker-registry command to update/change the custom docker registry on an existing environment. After the update, newly created experiences will use the updated registry.
- You can also use the update-custom-docker-registry command for existing experiences to update the repo details within the environment if the registry query/check from the environment service side is dynamic on the experience service side.

Enabling admin and user access to environments

In order to grant admin and user access to an environment that you registered in Cloudera, you should assign the required roles.

You need to be an EnvironmentCreator in order to register an environment. Once an environment is running, the following roles can be assigned:

- EnvironmentAdmin - Grants all rights to the environment and Cloudera Data Hub clusters running in it, except the ability to delete the environment. The user who registers the environment automatically becomes its EnvironmentAdmin.
- EnvironmentUser - Grants permission to view Cloudera Data Hub clusters and set the workload password for the environment. This role should be used in conjunction with service-specific roles such as DataHubAdmin, DWAdmin, DWUser, MLAdmin, MLUser, and so on. When assigning one of these service-specific roles to users, make sure to also assign the EnvironmentUser role.
- DataSteward - Grants permission to perform user/group management functions in Ranger and Atlas Admin, manage ID Broker mappings, and start user sync for the environment.
- Owner - Grants the ability to mange the environment in Cloudera, including deleting the environment. The user who registers the environment automatically becomes its Owner. The Owner role does not grant access the environment's clusters (Data Lakes, Cloudera Data Hub clusters).

The roles are described in detail in Resource roles. The steps for assigning the roles are described in Assigning resource roles to users and Assigning resource roles to groups.

Related Information
Resource roles
Assigning resource roles to users
Assigning resource roles to groups

Obtaining CLI commands for registering an environment

Although you can obtain CDP CLI commands for environment creation from CDP CLI help, the easiest way to obtain them is from the Cloudera web interface.

You can quickly obtain CDP CLI commands for creating an environment:

- From details of an existing environment
- From the register environment wizard

Create an environment from an existing environment

Obtain an environment template from an existing environment to create an environment with the exact same settings.

Since creating an environment, setting IDBroker mappings, and creating a Data Lake are separate actions in CDP CLI, you need to obtain three commands.

Required role: EnvironmentUser, EnvironmentAdmin, or Owner

Steps

1. Log in to the Cloudera web interface and navigate to the Cloudera Management Console
2. Navigate to environment details > Actions > Show CLI commands:
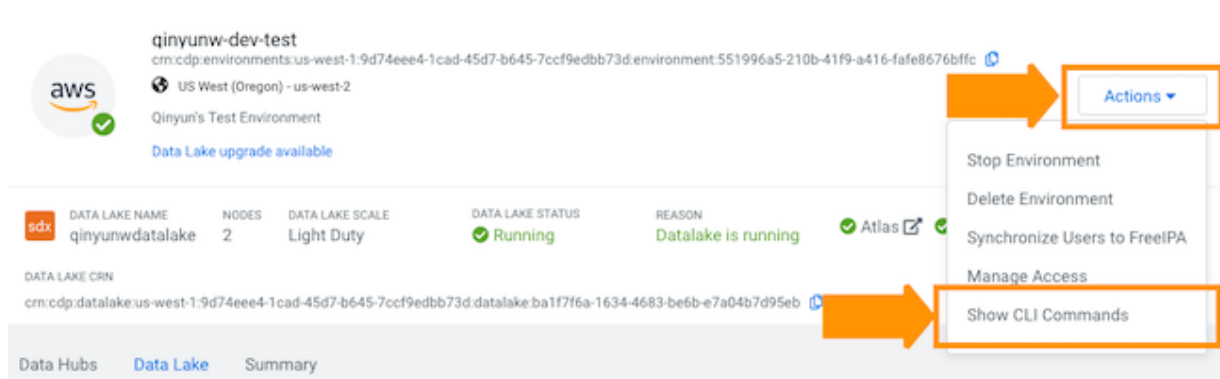
3. Click COPY three times to copy the three commands. These commands allow you to:

    a. Create an environment with the same settings as the existing environment.
    b. Set the same IDBroker mappings as in the original environment.
    c. Create a Data Lake with the same settings.
4. Before you can use these commands, make sure to update the following:

    a. In cdp environments create-<cloud-platform>-environment, update the value of --environment-name. It should be unique within Cloudera.
    b. In cdp environments set-id-broker-mappings, update the value of --environment-name. It should reference the name of the new environment that you are planning to create.
    c. In cdp datalake create-<cloud-platform>-datalake, update the value of --datalake-name. It should be unique within Cloudera.
    d. In cdp datalake create-<cloud-platform>-datalake, update the value of --environment-name. It should reference the name of the new environment that you are planning to create.
5. Run the three commands to:

    a. Register an environment.
    b. Set IDBroker mappings.
    c. Create a Data Lake.

Obtain CLI commands from the register environment wizard

Provide environment parameters in the environment wizard, and then on the last page of the wizard generate a CDP CLI template to create an environment and Data Lake with the parameters specified in the wizard. The obtained cluster template can be used to create an environment with the same settings via CDP CLI.

Since creating an environment, setting IDBroker mappings, and creating a Data Lake are separate actions in CDP CLI, you need to obtain three commands.

Required role: EnvironmentCreator

Steps

1. Log in to the Cloudera web interface and navigate to the Cloudera Management Console
2. Navigate to Environments > Register Environment.
3. Provide all the parameters for your environment.
4. On the last page, click SHOW CLI COMMAND in the bottom of the page.

5. Copy the three commands (for creating an environment, setting IDBroker mappings, and creating a Data Lake).
6. Run the commands:

    a. First run the command that creates the environment.

    b. Once the command finishes and the environment is running, run the command that sets IDBroker mappings.

    c. Next, run the command that creates the Data Lake.

Understanding environment UI options

To access information related to your environment, navigate to the Cloudera Management Console service > Environments and click on your environment.

You need to have the EnvironmentUser role or higher for the environment in order to access details of that environment.

From environments details, you can access the following:

- From the Cloudera Management Console tab, you can create, manage, and access Cloudera Data Hub clusters within the environment.
- From the Data Lake tab, you can monitor, manage, and access the Data Lake cluster.
- From the Cluster Definitions tab, you can access all cluster definitions that can be used with the environment.
- From the Summary tab, you can manage and monitor your environment.

The Summary includes the following information:

Option	Description
General	This includes your environment's CRN. CRN is an ID that Cloudera uses to identify a resource.
Credential	This links the provisioning credential associated with the environment and includes the option to change the credential.
Region	This lists the region in which your environment is deployed.
Network	This lists the networking resources used by your environment, provided by you or created by Cloudera during environment registration. You can add additional subnets for Cloudera Data Hub clusters deployed in the future.

Option	Description
Security Access	This lists the security groups used by your environment, provided by you or created by Cloudera during environment registration. You can provide new security groups for Data Hub clusters deployed in the future.
FreeIPA	This includes details of a FreeIPA server running in the environment and includes an Actions menu with FreeIPA management options.
Log Storage and Audits	This lists the cloud storage location used for logs and audits that you provided during environment registration. There is no way to update this location once your environment is running.
Telemetry	This includes your environment's telemetry settings. You can change them for any Cloudera Data Hub clusters created in the future.
Advanced	This lists the name of your root SSH key. You can add an additional SSH key if needed. The newly added SSH key will only be used forCloudera Data Hub clusters created in the future.

Related Information

Understanding Cloudera Data Hub details

Understanding Data Lake details

Using Compute Clusters

Compute Clusters enable you to deploy a containerized platform on Kubernetes for Data Services and shared services.

The Compute Cluster architecture offers simplified management, enhanced efficiency, and centralized control that leads to faster deployments, reduced configuration errors and improved system reliability. As multiple Data Services can optionally share the same Compute Cluster, it also lowers the cost of ownership.

When registering a Compute Cluster enabled environment, a default Compute Cluster is created. This default Compute Cluster is tied to the environment, and is used for running critical applications. The default Compute Cluster is labeled as **Default Cluster** in your environment to distinguish it from the other Compute Clusters.

You can create additional Compute Clusters that inherit the configuration of the default Compute Cluster. You can manage the lifecycle of the additional Compute Clusters as they are independent of the environment.

Setting up Compute Cluster IAM permissions

Before enabling Compute Clusters for your environment, you need to ensure that the required IAM roles and policies are set up.

Setting up the Compute Cluster IAM permissions are only needed when using **Reduced access policies** detailed on Cross-account access IAM role page, and completing these steps are not required when using default policies.

Creating IAM roles and instance profile for EKS

Complete the steps to create the required IAM roles and profile for EKS.

Procedure

1. Apply the following CloudFormation template to create the following:

   - IAM role called cdp-eks-master-role
   - IAM role and instance profile pair called cdp-liftie-instance-profile

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Creates Liftie IAM resources
Parameters:
  TelemetryLoggingEnabled:
    Description: Telemetry logging is enabled
    Type: String
  TelemetryLoggingBucket:
    Description: Telemetry logging bucket where Liftie logs will be store
d.
    Type: String
  TelemetryKmsKeyARN:
    Description: KMS Key ARN For Telemetry logging bucket.
    Type: String
    Default: ""
  TelemetryLoggingRootDir:
    Description: Telemetry logging root directory inside telemetry logg
ing bucket used for storing logs.
    Default: "cluster-logs"
    Type: String
Conditions:
  TelemetryLoggingEnabled:
    Fn::Equals:
      - {Ref: TelemetryLoggingEnabled}
      - true
  KMSKeyARNForTelemetryLoggingBucketIsEmpty: !Not [!Equals [!Ref Telemetry
KmsKeyARN, ""]]
Resources:
  AWSServiceRoleForAmazonEKS:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service:
                - eks.amazonaws.com
            Action:
              - sts:AssumeRole
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonEKSServicePolicy
        - arn:aws:iam::aws:policy/AmazonEKSClusterPolicy
      RoleName: cdp-eks-master-role
  NodeInstanceRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
```

```
                    - Effect: Allow
                      Principal:
                        Service:
                          - ec2.amazonaws.com
                      Action:
                        - sts:AssumeRole
              Path: "/"
              ManagedPolicyArns:
                - arn:aws:iam::aws:policy/AmazonEKSWorkerNodePolicy
                - arn:aws:iam::aws:policy/AmazonEKS_CNI_Policy
                - arn:aws:iam::aws:policy/AmazonEC2ContainerRegistryReadOnly
              RoleName: cdp-liftie-instance-profile
              Policies:
                - PolicyName: ssm-required
                  PolicyDocument:
                    Version: 2012-10-17
                    Statement:
                      - Effect: Allow
                        Action:
                          - ssm:GetParameters
                        Resource:
                          - "*"
                - PolicyName: cluster-autoscaler
                  PolicyDocument:
                    Version: 2012-10-17
                    Statement:
                      - Effect: Allow
                        Action:
                          - autoscaling:DescribeAutoScalingGroups
                          - autoscaling:DescribeAutoScalingInstances
                          - autoscaling:DescribeLaunchConfigurations
                          - autoscaling:DescribeScalingActivities
                          - autoscaling:DescribeTags
                          - ec2:DescribeImages
                          - ec2:DescribeInstanceTypes
                          - ec2:DescribeLaunchTemplateVersions
                          - ec2:GetInstanceTypesFromInstanceRequirements
                          - eks:DescribeNodegroup
                        Resource:
                          - "*"
                      - Effect: Allow
                        Action:
                          - autoscaling:SetDesiredCapacity
                          - autoscaling:TerminateInstanceInAutoScalingGroup
                        Resource:
                          - "*"
                        Condition:
                          StringEquals:
                            "aws:ResourceTag/k8s.io/cluster-autoscaler/enabled":
"true"
                - PolicyName: ebs-csi
                  PolicyDocument:
                    Version: 2012-10-17
                    Statement:
                      - Effect: Allow
                        Action:
                          - ec2:CreateSnapshot
                          - ec2:AttachVolume
                          - ec2:DetachVolume
                          - ec2:ModifyVolume
                          - ec2:DescribeAvailabilityZones
                          - ec2:DescribeInstances
                          - ec2:DescribeSnapshots
                          - ec2:DescribeTags
```

19

```
                              - ec2:DescribeVolumes
                              - ec2:DescribeVolumesModifications
                          Resource: "*"
                      - Effect: Allow
                        Action:
                          - ec2:CreateTags
                        Resource:
                          - "arn:aws:ec2:*:*:volume/*"
                          - "arn:aws:ec2:*:*:snapshot/*"
                        Condition:
                          StringEquals:
                            "ec2:CreateAction":
                                - CreateVolume
                                - CreateSnapshot
                      - Effect: Allow
                        Action:
                          - ec2:DeleteTags
                        Resource:
                          - "arn:aws:ec2:*:*:volume/*"
                          - "arn:aws:ec2:*:*:snapshot/*"
                      - Effect: Allow
                        Action:
                          - ec2:CreateVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "aws:RequestTag/ebs.csi.aws.com/cluster": "true"
                      - Effect: Allow
                        Action:
                          - ec2:CreateVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "aws:RequestTag/CSIVolumeName": "*"
                      - Effect: Allow
                        Action:
                          - ec2:CreateVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "aws:RequestTag/kubernetes.io/cluster/*": "owned"
                      - Effect: Allow
                        Action:
                          - ec2:DeleteVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "ec2:ResourceTag/ebs.csi.aws.com/cluster": "true"
                      - Effect: Allow
                        Action:
                          - ec2:DeleteVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "ec2:ResourceTag/CSIVolumeName": "*"
                      - Effect: Allow
                        Action:
                          - ec2:DeleteVolume
                        Resource: "*"
                        Condition:
                          StringLike:
                            "ec2:ResourceTag/kubernetes.io/created-for/pvc/name":
  "*"
                      - Effect: Allow
```

```
                    Action:
                      - ec2:DeleteSnapshot
                    Resource: "*"
                    Condition:
                      StringLike:
                        "ec2:ResourceTag/CSIVolumeSnapshotName": "*"
                  - Effect: Allow
                    Action:
                      - ec2:DeleteSnapshot
                    Resource: "*"
                    Condition:
                      StringLike:
                        "ec2:ResourceTag/ebs.csi.aws.com/cluster": "true"
          - PolicyName: efs-csi
            PolicyDocument:
              Version: 2012-10-17
              Statement:
                - Effect: Allow
                  Action:
                    - elasticfilesystem:DescribeAccessPoints
                    - elasticfilesystem:DescribeFileSystems
                    - elasticfilesystem:DescribeMountTargets
                  Resource: "*"
                - Effect: Allow
                  Action:
                    - elasticfilesystem:CreateAccessPoint
                  Resource: "*"
                  Condition:
                    StringLike:
                      "aws:RequestTag/efs.csi.aws.com/cluster": "true"
                - Effect: Allow
                  Action:
                    - elasticfilesystem:DeleteAccessPoint
                  Resource: "*"
                  Condition:
                    StringEquals:
                      "aws:ResourceTag/efs.csi.aws.com/cluster": "true"
          - !If
            - TelemetryLoggingEnabled
            - PolicyName: telemetry-s3-list-bucket
              PolicyDocument:
                Version: 2012-10-17
                Statement:
                  - Effect: Allow
                    Action:
                      - s3:ListBucket
                    Resource:
                      - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}'
                      - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}/${Tele
metryLoggingRootDir}/*'
            - !Ref 'AWS::NoValue'
          - !If
            - TelemetryLoggingEnabled
            - PolicyName: telemetry-s3-read-write
              PolicyDocument:
                Version: 2012-10-17
                Statement:
                  - Effect: Allow
                    Action:
                      - s3:*Object
                      - s3:AbortMultipartUpload
                      - s3:GetBucketAcl
                    Resource:
                      - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}'
```

```
                         - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}/${Tel
emetryLoggingRootDir}/*'
                 - !Ref 'AWS::NoValue'
             - !If
               - KMSKeyARNForTelemetryLoggingBucketIsEmpty
               - PolicyName: s3-kms-read-write-policy
                 PolicyDocument:
                   Version: 2012-10-17
                   Statement:
                     - Effect: Allow
                       Action:
                         - kms:CreateGrant
                         - kms:Decrypt
                         - kms:GenerateDataKey
                         - kms:GenerateDataKeyWithoutPlainText
                       Resource:
                         - !Sub ${TelemetryKmsKeyARN}
               - !Ref 'AWS::NoValue'
         - PolicyName: calico-cni
           PolicyDocument:
             Version: 2012-10-17
             Statement:
               - Effect: Allow
                 Action:
                   - ec2:ModifyInstanceAttribute
                 Resource:
                   - "*"
                 Condition:
                   StringEquals:
                     "ec2:Attribute": "SourceDestCheck"
  NodeInstanceProfile:
    Type: AWS::IAM::InstanceProfile
    Properties:
      Path: /
      InstanceProfileName: cdp-liftie-instance-profile
      Roles:
        - !Ref NodeInstanceRole
```

2. In the AWS console Cloudformation wizard, provide values for the following properties:

- Stack Name: Provide an appropriate name. Example: compute-precreated-roles-and-instanceprofile)
- TelemetryLoggingBucket: Name of the log bucket. Example: compute-logging-bucket
- TelemetryLoggingEnabled: Set it to true.
- TelemetryLoggingRootDir: Verify that it is set to the default value cluster-logs.
- TelemetryKMSKeyARN: If the telemetry bucket is encrypted, specify the KMS Key ARN. The default value is null.

3. On the last page in the wizard process, click the I acknowledge... checkbox to allow creation of IAM resources with special names.

4. Click Create stack.

Results

On the Cloudformation **Resources** tab, you find the precreated role and instance profile.

What to do next

Update the environment role to use the restricted role and policy.

Creating Compute Restricted IAM policy

Complete the steps to attach the Compute Restricted IAM policy with the cross-account role associated with your environment.

Procedure

1. Go to the **Environments** page.

Environments / Environments

2. In the Create Cross-account Access Policy field, attach the Compute Restricted IAM policy:

Replace the following placeholders in the JSON file:

- *[YOUR-ACCOUNT-ID]* with your account ID in use.
- *[YOUR-IAM-ROLE-NAME]* with the IAM restricted role associated with this policy.
- *[YOUR-SUBNET-ARN-*]* supplied during the Cloudera Environment(s) creation.

  Note: Provide all the subnets present in all the Cloudera Environment(s) that you intend to use it for the experience. If at any point a new Cloudera Environment is created or an existing one is updated for subnets, provide it here.

- *[YOUR-IDBROKER-ROLE-NAME]* with the ID Broker Role name in use.
- *[YOUR-LOG-ROLE-NAME]* with the Log Role name in use.
- *[YOUR-KMS-CUSTOMER-MANAGED-KEY-ARN]* with KMS key ARN.

```
{
  "Version": "2012-10-17",
  "Id": "ComputePolicy_v12",
  "Statement": [
    {
      "Sid": "SimulatePrincipalPolicy",
      "Effect": "Allow",
      "Action": [
```

```
                  "iam:SimulatePrincipalPolicy"
              ],
              "Resource": [
                "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-IAM-ROLE-NAME]"
              ]
          },
          {
              "Sid": "RestrictedPermissionsViaClouderaRequestTag",
              "Effect": "Allow",
              "Action": [
                "cloudformation:CreateStack",
                "cloudformation:CreateChangeSet",
                "ec2:createTags",
                "eks:TagResource"
              ],
              "Resource": "*",
              "Condition": {
                "StringLike": {
                  "aws:RequestTag/Cloudera-Resource-Name": [
                    "crn:cdp:*"
                  ]
                }
              }
          },
          {
              "Sid": "RestrictedPermissionsViaClouderaResourceTag",
              "Effect": "Allow",
              "Action": [
                "autoscaling:DeleteTags",
                "autoscaling:DetachInstances",
                "autoscaling:ResumeProcesses",
                "autoscaling:SetDesiredCapacity",
                "autoscaling:SuspendProcesses",
                "autoscaling:TerminateInstanceInAutoScalingGroup",
                "autoscaling:UpdateAutoScalingGroup",
                "cloudformation:DeleteChangeSet",
                "cloudformation:DeleteStack",
                "cloudformation:DescribeChangeSet",
                "cloudformation:DescribeStacks",
                "cloudformation:CancelUpdateStack",
                "cloudformation:ContinueUpdateRollback",
                "cloudformation:DescribeStackEvents",
                "cloudformation:DescribeStackResource",
                "cloudformation:DescribeStackResources",
                "cloudformation:ExecuteChangeSet",
                "cloudformation:ListStacks",
                "cloudwatch:deleteAlarms",
                "cloudwatch:putMetricAlarm",
                "ec2:AttachVolume",
                "ec2:CreateNetworkInterface",
                "ec2:CreateVolume",
                "ec2:DeleteVolume",
                "ec2:RunInstances",
                "eks:DescribeUpdate",
                "eks:ListUpdates",
                "eks:UpdateClusterConfig",
                "eks:UpdateClusterVersion",
                "iam:GetRolePolicy",
                "iam:ListInstanceProfiles",
                "iam:ListRoleTags",
                "iam:RemoveRoleFromInstanceProfile",
                "iam:TagRole",
                "iam:UntagRole",
                "logs:DescribeLogStreams",
```

```
              "logs:FilterLogEvents"
            ],
            "Resource": "*",
            "Condition": {
              "StringLike": {
                "aws:ResourceTag/Cloudera-Resource-Name": [
                  "crn:cdp:*"
                ]
              }
            }
          },
          {
            "Sid": "RestrictedPermissionsViaCloudFormation",
            "Effect": "Allow",
            "Action": [
              "autoscaling:CreateAutoScalingGroup",
              "autoscaling:CreateLaunchConfiguration",
              "autoscaling:CreateOrUpdateTags",
              "autoscaling:DeleteAutoScalingGroup",
              "autoscaling:DeleteLaunchConfiguration",
              "autoscaling:DescribeAutoScalingInstances",
              "autoscaling:DescribeLaunchConfigurations",
              "autoscaling:DescribeScalingActivities",
              "autoscaling:DescribeScheduledActions",
              "autoscaling:DescribeTags",
              "dynamodb:DescribeTable",
              "ec2:AuthorizeSecurityGroupEgress",
              "ec2:CreateLaunchTemplate",
              "ec2:CreateSecurityGroup",
              "ec2:DeleteLaunchTemplate",
              "ec2:DeletePlacementGroup",
              "ec2:DeleteSecurityGroup",
              "ec2:DescribeAccountAttributes",
              "ec2:DescribeImages",
              "ec2:DescribeInstanceStatus",
              "ec2:DescribeInstances",
              "ec2:DescribeKeyPairs",
              "ec2:DescribeLaunchTemplateVersions",
              "ec2:DescribeLaunchTemplates",
              "ec2:DescribePlacementGroups",
              "ec2:DescribeRegions",
              "ec2:DescribeRouteTables",
              "ec2:DescribeSecurityGroups",
              "ec2:DescribeVolumes",
              "ec2:RevokeSecurityGroupEgress",
              "ec2:RevokeSecurityGroupIngress",
              "eks:CreateCluster",
              "eks:DeleteCluster"
            ],
            "Resource": "*",
            "Condition": {
              "ForAnyValue:StringEquals": {
                "aws:CalledVia": [
                  "cloudformation.amazonaws.com"
                ]
              }
            }
          },
          {
            "Sid": "RestrictedEC2PermissionsViaClouderaResourceTag",
            "Effect": "Allow",
            "Action": [
              "ec2:RebootInstances",
              "ec2:StartInstances",
```

```
                        "ec2:StopInstances",
                        "ec2:TerminateInstances"
                     ],
                     "Resource": [
                        "*"
                     ],
                     "Condition": {
                        "ForAnyValue:StringLike": {
                           "ec2:ResourceTag/Cloudera-Resource-Name": [
                              "crn:cdp:*"
                           ]
                        }
                     }
                  },
                  {
                     "Sid": "RestrictedIamPermissionsToClouderaResources",
                     "Effect": "Allow",
                     "Action": [
                        "iam:PassRole"
                     ],
                     "Resource": [
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-IDBROKER-ROLE-NAME]",
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-LOG-ROLE-NAME]",
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/liftie-*-eks-service-role",
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/liftie-*-eks-worker-nodes",
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/cdp-eks-master-role",
                        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/cdp-liftie-instance-profile"
                     ]
                  },
                  {
                     "Sid": "RestrictedKMSPermissionsUsingCustomerProvidedKey",
                     "Effect": "Allow",
                     "Action": [
                        "kms:CreateGrant",
                        "kms:DescribeKey",
                        "kms:Encrypt",
                        "kms:Decrypt",
                        "kms:ReEncrypt*",
                        "kms:GenerateDataKey*"
                     ],
                     "Resource": [
                        "[YOUR-KMS-CUSTOMER-MANAGED-KEY-ARN]"
                     ]
                  },
                  {
                     "Sid": "AllowCreateDeleteTagsForSubnets",
                     "Effect": "Allow",
                     "Action": [
                        "ec2:CreateTags",
                        "ec2:DeleteTags"
                     ],
                     "Resource": [
                        "arn:aws:ec2:[YOUR-SUBNET-REGION]:[YOUR-ACCOUNT-ID]:subnet/*"
                     ]
                  },
                  {
                     "Sid": "ModifyInstanceAttribute",
                     "Effect": "Allow",
                     "Action": [
                        "ec2:ModifyInstanceAttribute"
                     ],
                     "Resource": [
                        "*"
                     ],
```

```
        "Condition": {
          "StringEquals": {
            "ec2:Attribute": "SourceDestCheck"
          }
        }
      },
      {
        "Sid": "OtherPermissions",
        "Effect": "Allow",
        "Action": [
          "autoscaling:DescribeAutoScalingGroups",
          "ec2:AuthorizeSecurityGroupIngress",
          "ec2:CreateLaunchTemplateVersion",
          "ec2:CreatePlacementGroup",
          "ec2:DeleteKeyPair",
          "ec2:DeleteNetworkInterface",
          "ec2:DescribeAvailabilityZones",
          "ec2:DescribeInstanceTypes",
          "ec2:DescribeNetworkInterfaces",
          "ec2:DescribeSubnets",
          "ec2:DescribeVpcAttribute",
          "ec2:DescribeVpcs",
          "ec2:ImportKeyPair",
          "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
          "ec2:GetInstanceTypesFromInstanceRequirements",
          "eks:DescribeCluster",
          "elasticloadbalancing:DescribeLoadBalancers",
          "iam:GetRole",
          "iam:ListRoles",
          "iam:GetInstanceProfile"
        ],
        "Resource": [
          "*"
        ]
      },
      {
        "Sid": "AllowSsmParams",
        "Effect": "Allow",
        "Action": [
          "ssm:DescribeParameters",
          "ssm:GetParameter",
          "ssm:GetParameters",
          "ssm:GetParameterHistory",
          "ssm:GetParametersByPath"
        ],
        "Resource": [
          "arn:aws:ssm:*:*:parameter/aws/service/eks/optimized-ami/*"
        ]
      },
      {
        "Sid": "CfDeny",
        "Effect": "Deny",
        "Action": [
          "cloudformation:*"
        ],
        "Resource": [
          "*"
        ],
        "Condition": {
          "ForAnyValue:StringLike": {
            "cloudformation:ImportResourceTypes": [
              "*"
            ]
          }
```

```
      }
    },
    {
      "Sid": "ForAutoscalingLinkedRole",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/autoscaling-
plans.amazonaws.com/AWSServiceRoleForAutoScalingPlans_EC2AutoScaling"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "autoscaling-plans.amazonaws.com"
        }
      }
    },
    {
      "Sid": "ForEksLinkedRole",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/eks.amazona
ws.com/AWSServiceRoleForEKS"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "eks.amazonaws.com"
        }
      }
    }
  ]
}
```

3. Provide and verify your Customer Managed Key (CMK) to be used for EBS encryption.

   Along with providing the KMS Customer Managed Key (CMK) for volume encryption in the policy section with Sid: RestrictedKMSPermissionsUsingCustomerProvidedKey, you need to verify that the policy for the Customer Managed Key (CMK) at KMS (this is not an IAM policy) has the following three permission blocks defined for AWSServiceRoleForAutoScaling.

```
{
  "Statement": [
    {
      "Sid": "AllowAutoscalingServiceLinkedRoleForAttachmentOfPersisten
tResources",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/auto
scaling.amazonaws.com/AWSServiceRoleForAutoScaling"
      },
      "Action": "kms:CreateGrant",
      "Resource": "*",
      "Condition": {
        "Bool": {
          "kms:GrantIsForAWSResource": "true"
        }
      }
    },
    {
      "Sid": "AllowAutoscalingServiceLinkedRoleUseOfTheCMK",
```

```
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/autos
caling.amazonaws.com/AWSServiceRoleForAutoScaling"
        },
        "Action": [
          "kms:Encrypt",
          "kms:Decrypt",
          "kms:ReEncrypt*",
          "kms:GenerateDataKey*",
          "kms:DescribeKey"
        ],
        "Resource": "*"
      },
      {
        "Sid": "Allow EKS access to EBS.",
        "Effect": "Allow",
        "Principal": {
          "AWS": "*"
        },
        "Action": [
          "kms:CreateGrant",
          "kms:Encrypt",
          "kms:Decrypt",
          "kms:ReEncrypt*",
          "kms:GenerateDataKey*",
          "kms:DescribeKey"
        ],
        "Resource": "*",
        "Condition": {
          "StringEquals": {
            "kms:CallerAccount": "[YOUR-ACCOUNT-ID]",
            "kms:viaService": "ec2.[YOUR-ACCOUNT-REGION].amazonaws.com"
          }
        }
      }
    ]
```

```
}
```

After the policy is attached, the KMS service page will show the CMS as having the policy attached as shown in the following example:

Enabling default Compute Cluster for new environments

When creating your environment, you can enable the default Compute Cluster using the Cloudera Management Console or CDP CLI to be able to run your data and shared services on the containerized platform.

Required role: EnvironmentAdmin

Before you begin

• Ensure that your AWS account has all the resources required by Cloudera.

For more information, see AWS account requirements.
• Ensure that the IAM permissions are correctly set up for your environment.

For more information, see Setting up Compute Cluster IAM permissions.

Using Cloudera Management Console

When creating your environment in **Cloudera Management Console**, ensure that you use the **Enable Compute Cluster** setting to create the Compute Cluster enabled environment.

> ⓘ Enable Compute Cluster to set up a containerized platform for all data services.

🔵 Enable Compute Cluster

Deploy a standard, uniform Kubernetes platform that can host any data services and shared services.

After completing the step for **Data Access and Data Lake Scaling**, configure the networking settings for Kubernetes with either selecting the **Private Kubernetes Cluster** or providing **Authorized IP Ranges** on the **Region, Networking and Security** page. **Worker Node Subnets** are automatically pre-filled with the same set of subnets provided in **Network** section, but you have the option to not use all of the available subnets.

⚙️ **Kubernetes**

⚪ Private Kubernetes Cluster

Kubernetes API Server Authorized IP Ranges

[   ] ❓

> ⓘ Please select a network subnet first in the Network section!

Worker Node Subnets*

[Please select subnet(s)   ▼] ❓

For more information about creating your environment, see the Register environment (UI) documentation.

Using CDP CLI

Run the following command to create the Compute Cluster enabled environment:

For Without private cluster

```
cdp environments create-aws-environment
--environment-name [***ENVIRONMENT NAME***] \
--credential-name [***CREDENTIAL NAME***] \
--region [***REGION***] \
--security-access [***SECURITY CONTROL CONFIGURATIONS***] \
--authentication [***PUBLIC SSH KEY***] \
--log-storage [***STORAGE CONFIGURATION***] \
--enable-compute-cluster \
--compute-cluster-configuration \
privateCluster=FALSE, \
kubeApiAuthorizedIpRanges=[***CIDR1***],[***CIDR2***]
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

For With private cluster

```
cdp environments create-aws-environment
--environment-name [***ENVIRONMENT NAME***] \
--credential-name [***CREDENTIAL NAME***] \
--region [***REGION***] \
--security-access [***SECURITY CONTROL CONFIGURATIONS***] \
--authentication [***SSH KEY***] \
```

```
--log-storage [***STORAGE CONFIGURATION***] \
--enable-compute-cluster \
--compute-cluster-configuration \
privateCluster=TRUE
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

After the command runs, you can verify if the environment was successfully created with the default Compute Cluster with using the following commands:

- Describing the environment:

```
cdp environments describe-environment --env-name-or-crn [***ENVIRONMENT
 NAME OR CRN***]

...
        "awsComputeClusterConfiguration": {
            "privateCluster": false,
            "kubeApiAuthorizedIpRanges": [
                "0.0.0.0/0"
            ]
        },
        "enableComputeCluster": "true"
...
```

- Listing Compute Clusters:

```
cdp compute list-clusters --env-name-or-crn [***ENVIRONMENT NAME OR
 CRN***]
```

Note: FreeIPA must be created and running before the default Compute Cluster is created.

You can use the following command to retry the environment creation with the default Compute Cluster:

For Without private cluster

```
cdp environments initialize-aws-compute-cluster
--environment-name [***ENVIRONMENT NAME***] \
--compute-cluster-configuration \
privateCluster=FALSE, \
kubeApiAuthorizedIpRanges=[***CIDR1***],[***CIDR2***]
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

For With private cluster

```
cdp environments initialize-aws-compute-cluster
--environment-name [***ENVIRONMENT NAME***] \
--compute-cluster-configuration \
privateCluster=TRUE
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

Enabling default Compute Cluster for existing environments

In case you already have an environment, but would like to have your services to run on the containerized platform enabled by Compute Clusters, you can add the default Compute Cluster to your existing environment.

Required resource role: EnvironmentAdmin

Before you begin

- Ensure that the environment has no default Compute Cluster provisioned.
- Ensure that the environment is started and available.
- Ensure that IAM permissions are correctly set up during the environment creation.

    For more information, see Setting up Compute Cluster IAM permissions.

Using Cloudera Management Console

1. Navigate to your environment.
2. Click Compute Clusters.

    The following message will indicate that Compute Cluster are not enabled for your environment:

3. Click Enable Compute Clusters.
4. Provide the necessary networking information for the **Kubernetes** cluster.

    a. If you need to create a Private Cluster, enable Private Kubernetes Cluster to create a private cluster that blocks all access to the API Server endpoint.

    b. If you do not need to create a Private Cluster, provide the CIDRs to the Kubernetes API Server Authorized IP Ranges field to specify a set of IP ranges that will be allowed to access the Kubernetes API server.

    c. **Worker Node Subnets** are automatically pre-filled with the same set of subnets provided during environment registration, but you have the option to not use all of the available subnets.

5. Click Submit.

    You will be redirected to the **Compute Clusters** tab, where you can track the creation process of the default Compute Cluster.

Using CDP CLI

Run the following command to add the default Compute Cluster to the environment:

For Without private cluster

```
cdp environments initialize-aws-compute-cluster
--environment-name [***ENVIRONMENT NAME***] \
--compute-cluster-configuration \
privateCluster=FALSE, \
kubeApiAuthorizedIpRanges=[***CIDR1***],[***CIDR2***]
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

For With private cluster

```
cdp environments initialize-aws-compute-cluster
--environment-name [***ENVIRONMENT NAME***] \
--compute-cluster-configuration \
privateCluster=TRUE
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

The environment will have COMPUTE_CLUSTER_CREATION_IN_PROGRESS status. You can use the following command to check the status of the environment creation, the statusReason field will contain the information about the process:

```
cdp environments describe-environment --env-name-or-crn [***ENVIRONMENT NAME
 OR CRN***]
```

For more detailed status information about the cluster creation, you can use the following command:

```
cdp compute list-clusters --env-name-or-crn [***ENVIRONMENT NAME OR CRN***]
```

Adding more Compute Clusters

You can add as many additional Compute Clusters as required beside the default Compute Cluster using Cloudera Management Console or CDP CLI.

Required role: EnvironmentAdmin

Using Cloudera Management Console

You can create additional Compute Clusters beside the default Compute Cluster using Cloudera Management Console.

1. Navigate to your environment.
2. Select Compute Clusters tab.

3. Click Add Compute Cluster.

The **Add Compute Cluster** wizard appears.

Add Compute Cluster

⊙ / liftie-v2 / Compute Clusters / Add Compute Cluster

Environment

liftie-v2

* Name

Description

Add Cluster   Cancel

4. Provide a Name to the cluster, and optionally a Description.

5. Click Add Cluster.

You will be redirected to the **Compute Clusters** tab, where you can track the creation process of the additional Compute Cluster.

Using CDP CLI

You can use the following CDP CLI command to create additional Compute Clusters after the default Compute Cluster is created:

```
cdp compute create-cluster
--environment --env-name-or-crn [***ENVIRONMENT NAME OR CRN***] \
--name [***CLUSTER NAME***]
```

After the command runs, you can verify if the additional Compute Cluster creation was successful using the following command:

```
cdp compute list-clusters --env-name-or-crn [***ENVIRONMENT NAME OR CRN***]
```

The additional Compute Cluster status should be in RUNNING state.

Managing Compute Clusters

After creation, you can view the Compute Cluster details, manage the access of the clusters, download the Kubeconfig file, and delete the created additional Compute Clusters.

Required resource role: EnvironmentAdmin or Owner

Required account role: IamViewer

Using Cloudera Management Console

1. Click on the Name of the Compute Cluster.

   You will be redirected to the **Cluster Details** page.

   On the **Cluster Details** page, you can view the **Status**, **Creation Date**, **Created By**, **Description** and **CRN** of the Compute Cluster.

2. Click Actions to open the drop-down menu.

   > **Note:** Ensure that you have one of the following roles to access the **Actions** menu
   >
   > - EnvironmentOwner or EnvironmentAdmin
   > - ClusterCreator or ClusterOwner

   a. Click Manage Access to update the list of users who have access to manage the Compute Cluster and use it for installing Data Services.

      1. Click Update roles to update the **Resource Role** of a user or group.

         You can assign the Owner resource role to a user or group to provide permission to manage the Compute Cluster and install services on it.

   b. Click Kubernetes Access to grant access for users to the Kubernetes API server.

      1. Enter the User ARN.
      2. Click Grant Access.

      You also have the option to Download the Kubeconfig from this page.

   c. Click Delete Compute Cluster, if you no longer need the additional Compute Cluster.

      > **Warning:** Ensure that the data services running on the Compute Cluster are deleted before deleting the Compute Cluster itself.

      1. Confirm the deletion of the additional Compute Cluster by clicking Remove.

3. Click Networking tab to view the subnet information of the Compute Cluster.
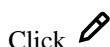
   a. Click ✏ to update the **Kubernetes API Server Authorized IP Ranges**.

      > **Note:** Ensure that you have one of the following roles to access the **Actions** menu
      >
      > - EnvironmentOwner or EnvironmentAdmin
      > - ClusterCreator or ClusterOwner

      1. Add or remove the CIDRs, and click Save.

4. Click **Encryption** tab to view the encryption key of the Compute Cluster.
5. Click **Node Groups** tab to have an illustrated overview of the resource utilization of the different services running on the Compute Cluster.
6. Click **Compute Cluster Version** to view the Kubernetes and Insfrastructure Service versions of the cluster.
7. Click **Logs** to check the different events of the Compute Cluster.

Suspending and resuming Compute Clusters

You can suspend and resume your default and additional Compute Clusters in an environment to manage your cloud costs using CDP CLI.

Before you begin

- Ensure that the environment is started and available.
- Ensure that there are no Data Service workload instances running in the Compute Cluster.

Required resource role: EnvironmentAdmin or Owner

Procedure

1. Run the following command to suspend the Compute Cluster:

```
cdp compute suspend-cluster --cluster-crn [***CLUSTER CRN***]
```

After running the command, the cluster status changes to SUSPENDING, and will progress to SUSPENDED once the operation completes. The status of the cluster can be polled using describe-cluster command:

```
cdp compute describe-cluster --cluster-crn [***CLUSTER CRN***]
```

2. Run the following command to resume the Compute Cluster with SUSPENDED status:

```
cdp compute resume-cluster --cluster-crn [***CLUSTER CRN***]
```

After running the command, the cluster status changes to RESUMING, and will progress to RUNNING status once the operation completes.

> **Note:** Only the following operations are supported on a cluster with SUSPENDED status:
> - DescribeCluster
> - ListClusters
> - DeleteCluster

Deleting Compute Clusters

You can delete additional Compute Clusters in an environment to manage your cloud costs using CDP CLI.

Before you begin:

- Ensure that the environment is started and available.
- Ensure that there are no Data Service workload instances running in the Compute Cluster.
- You can only delete additional Compute Clusters.
- You can only delete the default Cluster by deleting the environment.

Run the following command to delete the Compute Cluster:

```
cdp compute delete-cluster --cluster-crn [***CLUSTER CRN***]
```

Monitoring an environment

Once an environment exists, you can access it from the Cloudera Management Console.

Required role: EnvironmentUser, EnvironmentAdmin, or Owner

Monitoring an environment via UI

To access information related to your environment, perform the following steps:

1. To access an existing environment, navigate to Cloudera Management Console > Environments and click on your environment.
2. Click on the Summary tab to access environment details.
3. You can monitor the status of your environment from this page.

Monitoring an environment via CLI

You can view and monitor available environments from CDP CLI using the following commands:

* List available environments: cdp environments   list-environments

    Example:

```
cdp environments list-environments
{
    "environments": [
        {
            "environmentName": "cdp-demo",
            "crn": "crn:altus:environments:us-west-1:c8dbde4b-ccce-4f8d-
a581-830970ba4908:environment:d3361b40-39ab-4d87-bd5b-abc15f16b90c",
            "status": "DELETE_FAILED",
            "region": "us-east-2",
            "cloudPlatform": "AWS",
            "credentialName": "cdp-demo",
            "description": "Cdp demo"
        },
        {
            "environmentName": "cdp-new",
            "crn": "crn:altus:environments:us-west-1:c8dbde4b-ccce-4f8d-a5
81-830970ba4908:environment:1d2bacde-5c96-47c1-a597-9f276b824028",
            "status": "AVAILABLE",
            "region": "us-east-2",
            "cloudPlatform": "AWS",
            "credentialName": "cdp-demo",
            "description": ""
        }
}
```

* Get basic information about a specific environment: cdp environments   describe-environment --environment-name <value>
* Get id broker mappings: cdp environments get-id-broker-mappings   --environment-name <value>
* Get status of specified operation: cdp environments get-operation   --environment-name <value> [--operation-id <value>]

    To use the get-operation command to get the status of a specified event, you need to specify the operation id of the operation. Every operation that starts a process running in the background, like creating, stopping, or restarting an environment, returns an operationId field in the response.

    Example:

```
cdp environments stop-environment
{
    "operationId": "aaa52df6-9322-4a91-863b-98e459b65240"
}
```

The value of this operationId can be used as the value for the --operation-id option for the get-operation command.

Example:

```
cdp environments get-operation --environment-name foldikrskenv --operati
on-id aaa52df6-9322-4a91-863b-98e459b65240
```

Output format:

```
{
    "operationId": "identifier of the operation",
    "operationName": "Short name of the operation",
```

```
    "operationStatus": "UNKNOWN | RUNNING | FAILED | FINISHED | CANCELLE
D",
    "started": "Start time of the operation"
    "ended": "End time of the operation if it is completed"
}
```

Output example:

```
{
    "operationId": "aaa52df6-9322-4a91-863b-98e459b65240",
    "operationName": "EnvironmentStop",
    "operationStatus": "RUNNING",
    "started": "2025-03-05T13:59:32+00:00"
}
```

Unsuccessful operation statuses are stored for 2 weeks, while successful status operations are stored for 1 day.

The operation id is optional, and if it is omitted, the status of the last operation is returned.

• Get status of latest operation: cdp environments get-operation   --environment-name <value>

Example:

```
cdp environments get-operation --environment-name env1
{
    "operationId": "55cb5614-b53c-4f03-811f-459c4893c5f2",
    "operationName": "EnvironmentCreation",
    "operationStatus": "RUNNING",
    "started": "2025-01-15T08:46:48+00:00"
}
```

Related Information

Accessing Data Lake services

Understanding Data Lake details

Understanding Cloudera Data Hub cluster details

Managing FreeIPA

Environment status options

This topic lists all possible environment status options for the UI and CLI and explains what they mean.

Environment status	Description
Environment creation	
CREATION_INITIATED	Environment creation request was registered in the database and Cloudera is starting the environment creation flow.
ENVIRONMENT_INITIALIZATION_IN_PROGRESS	Setting up the region and network metadata (public/private and cidr).
ENVIRONMENT_VALIDATION_IN_PROGRESS	Setting up the region and network metadata (public/private and cidr).
NETWORK_CREATION_IN_PROGRESS	If the user chose the create new network option, then Cloudera creates the network on cloud provider side.
PUBLICKEY_CREATE_IN_PROGRESS	If the user choose the create new SSH key option, then Cloudera creates the SSH key on cloud provider side.
FREEIPA_CREATION_IN_PROGRESS	Creating the FreeIPA resources for an environment.
Environment update	
UPDATE_INITIATED	Environment update was requested and Cloudera is starting the update flow (network update, load balancer update, SSH key update).
Environment deletion	

Environment status	Description
DELETE_INITIATED	Environment deletion request was registered and Cloudera is starting the deletion flow.
NETWORK_DELETE_IN_PROGRESS	If the user chose the create new network option, then Cloudera deletes the network on cloud provider side.
PUBLICKEY_DELETE_IN_PROGRESS	If the user choosing the create new SSH key option, then Cloudera deletes the SSH key on cloud provider side.
FREEIPA_DELETE_IN_PROGRESS	Deleting the FreeIPA resources for an environment.
EXPERIENCE_DELETE_IN_PROGRESS	Deleting all the attached clusters (Cloudera Data Warehouse, Cloudera AI, and so on).
RDBMS_DELETE_IN_PROGRESS	Deleting all the provisioned RDS instances that are related to an environment.
CLUSTER_DEFINITION_DELETE_PROGRESS	Deleting all the cluster definitions that are created for an environment.
UMS_RESOURCE_DELETE_IN_PROGRESS	Deleting all the related UMS resources for an environment.
IDBROKER_MAPPINGS_DELETE_IN_PROGRESS	Deleting all the IBroker mapping for an environment.
S3GUARD_TABLE_DELETE_IN_PROGRESS	Deleting all the Dynamo DB tables for an environment.
DATAHUB_CLUSTERS_DELETE_IN_PROGRESS	Deleting all the attached Cloudera Data Hub clusters.
DATALAKE_CLUSTERS_DELETE_IN_PROGRESS	Deleting the attached Data Lake cluster.
ARCHIVED	Environment has been deleted (not shown on the UI).
Environment is running	
AVAILABLE	Environment is available (ready to use).
Environment process failed	
CREATE_FAILED	Environment creation failed (Detailed message in the statusReason).
DELETE_FAILED	Environment deletion failed (Detailed message in the statusReason).
UPDATE_FAILED	Environment update failed (Detailed message in the statusReason).
Environment stop	
STOP_DATAHUB_STARTED	Stopping all the Cloudera Data Hub clusters in an environment.
STOP_DATAHUB_FAILED	Stopping all the Cloudera Data Hub clusters in an environment failed (Detailed message in the statusReason).
STOP_DATALAKE_STARTED	Stopping the Data Lake cluster in an environment.
STOP_DATALAKE_FAILED	Stopping the Data Lake cluster in an environment failed (Detailed message in the statusReason).
STOP_FREEIPA_STARTED	Stopping the FreeIPA instances in an environment.
STOP_FREEIPA_FAILED	Stopping the FreeIPA instances in an environment failed (Detailed message in the statusReason).
ENV_STOPPED	Environment was successfully stopped.
Environment start	
START_DATAHUB_STARTED	Starting all the Cloudera Data Hub clusters in an environment.
START_DATAHUB_FAILED	Starting all the Cloudera Data Hub clusters in an environment failed (Detailed message in the statusReason).
START_DATALAKE_STARTED	Starting the Data Lake cluster in an environment.
START_DATALAKE_FAILED	Starting the Data Lake cluster in an environment failed (Detailed message in the statusReason).
START_FREEIPA_STARTED	Starting all the FreeIPA instances in an environment.

Environment status	Description
START_FREEIPA_FAILED	Starting all the FreeIPA instances failed in an environment (Detailed message in the statusReason).
START_SYNCHRONIZE_USERS_STARTED	Starting user sync for all the clusters in an environment.
START_SYNCHRONIZE_USERS_FAILED	Starting user sync for all the clusters in an environment failed (Detailed message in the statusReason).
FreeIPA instance deletion	
FREEIPA_DELETED_ON_PROVIDER_SIDE	The FreeIPA instance has been deleted on cloud provider side.
Load balancer	
LOAD_BALANCER_ENV_UPDATE_STARTED	Start updating the LoadBalancer on Data Lake in an environment.
LOAD_BALANCER_ENV_UPDATE_FAILED	Failed to update the LoadBalancer on Data Lake in an environment (Detailed message in the statusReason).
LOAD_BALANCER_STACK_UPDATE_STARTED	Start updating the LoadBalancer on Cloudera Data Hub clusters in an environment.
LOAD_BALANCER_STACK_UPDATE_FAILED	Failed to update the LoadBalancer on Cloudera Data Hub clusters in an environment (Detailed message in the statusReason).

Stopping and restarting an environment

You can stop an environment if you need to suspend but not terminate the resources within the environment. When you stop an environment, all of the resources within the environment are also stopped, including Data Lakes and Cloudera Data Hub clusters. You can also restart the environment.

> ⚠️ **Warning:**
> The Cloudera AI service does not support environment stop and restart. This means that if Cloudera AI workbenches are running or expected to be provisioned within an environment, then the environment should not be stopped. If done, this will disrupt running Cloudera AI workbenches and prevent successful provisioning of Cloudera AI workbenches in the environment.

Required role: EnvironmentAdmin or Owner

Steps

For Cloudera UI

1. Navigate to the environment in Cloudera Management Console > Environments.
2. Click Actions Stop Environment and confirm the action.
3. To restart the environment, click Actions Start Environment .

   > ⚠️ **Warning:** We have not tested or certified restarting the environment while Cloudera Data Engineering is running.

For CDP CLI

Use the following command to stop an environment:

```
cdp environments stop-environment --environment-name <ENVIRONMENT_NAME>
```

Use the following commands to start an environment:

```
cdp environments start-environment --environment-name <ENVIRONMENT_NAME>
  [--with-datahub-start]
```

Use the following commands to start an environment and all Cloudera Data Hub clusters running in it:

```
cdp environments start-environment --environment-name <ENVIRONMENT_NAME>
  --with-datahub-start
```

Deleting an environment

Deleting an environment terminates all resources within the environment including the Data Lake.

Before you begin

To delete an environment, you should first terminate all clusters running in that environment.

Required role: Owner or PowerUser

Steps

For Cloudera UI

1. In Cloudera Management Console, navigate to Environments.
2. Click on your environment.
3. Click Actions Delete .
4. Check the box next to "I would like to delete all connected resources" if you have Data Lake and Cloudera Data Hub clusters running within the environment. This will delete the Data Lake and Cloudera Data Hub clusters together with the rest of the environment.

   Note: The "I would like to delete all connected resources" option does not delete any Cloudera Data Warehouse or Cloudera AI clusters running within the environment, so these always need to be terminated prior to environment termination.
5. Check the box next to "I understand this action cannot be undone". This is required.
6. Click Delete.

For CDP CLI

When terminating an environment from the CDP CLI, you need to first terminate the Data Lake:

1. Terminate the Data Lake using the following command:

   ```
   cdp datalake delete-datalake --datalake-name <value>
   ```
2. Wait until the Data Lake terminates before proceeding. Use the following commands to check on the status of Data Lake:

   ```
   cdp datalake get-cluster-host-status --cluster-name <value>
   ```

   ```
   cdp datalake list-datalakes
   ```
3. Delete the environment using the following command:

   ```
   cdp environments delete-environment --environment-name <value> --cascadi
   ng
   ```

   The --cascading option deletes or Cloudera Data Hub clusters running in the environment.

If environment deletion fails, you can:

- Repeat the environment deletion steps, but also check "I would like to force delete the selected environment". Force deletion removes Cloudera resources from Cloudera, but leaves cloud provider resources running.

- Clean up cloud resources that were left on your cloud provider account. See Cleaning up a failed AWS environment.

Only the resources that were provisioned as part of the environment are deleted. For example, if a new network was created by Cloudera for the environment, the network will be deleted. But if you provided your existing network, it will not be deleted as part of environment deletion.

Cleaning up a failed environment

When environment creation fails, you should delete the environment. If environment termination fails, you should clean up any resources that might have already been created on your AWS account.

When environment creation fails, you should delete the environment by using the steps described in Delete an Environment.

If environment termination fails, you should clean up any resources that might have already been created on your AWS account. To do this:

1. Navigate to the AWS Management Console your region CloudFormation .
2. Use the search box to find the stacks corresponding to your Data Lake cluster and FreeIPA server.

   The CF stacks use the following naming convention:

   - The Data Lake cluster stack uses the name that you provided for your data lake
   - The FreeIPA server stack uses the name that you provided for your environment

3. Delete the CloudFormation stacks corresponding to your Data Lake and FreeIPA server instances:
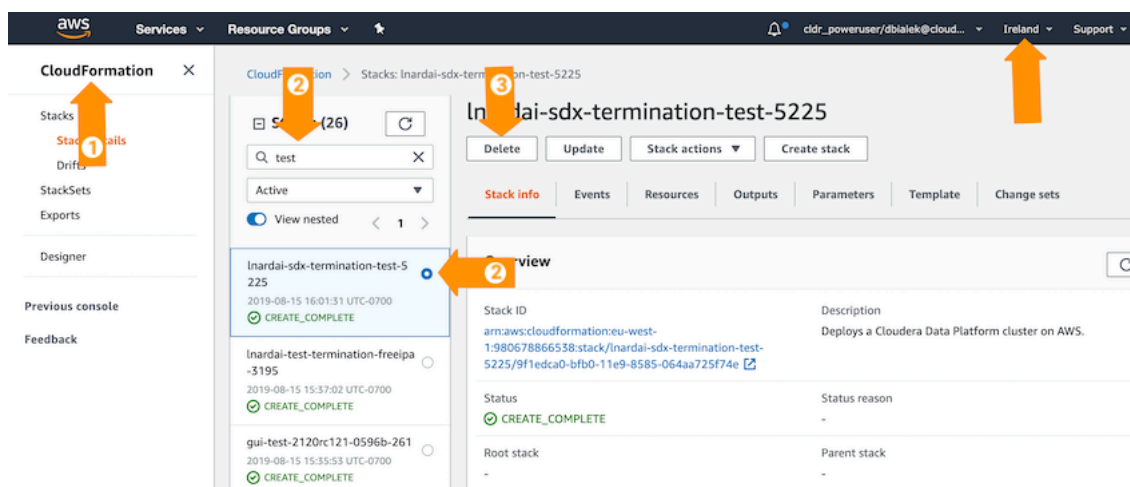
Adding subnets to an environment

You can add additional subnets to an existing environment. These subnets will only be used for all Cloudera Data Hub clusters created within the environment in the future.

Before you begin

These steps assume that you have already created the subnets that you want to add to the environment.

Required role: EnvironmentAdmin or Owner

Steps

For CDP UI

1. Navigate to Cloudera Management Console > Environments and click on the environment you want to modify.
2. Click the Summary tab.
3. Scroll down to the Network section.
4. Click the ☑ (edit) button, then click the Select Subnets pull down menu and select the subnet you want to add to the Environment.
5. Click the ▤ (save) button.

   You should see the new subnet listed in the Network section.

   > **Note:**
   >
   > The newly added subnets will not be used for any Cloudera services other than Cloudera Data Hub. The newly added subnets will only be used for the Cloudera Data Hub clusters created within the environment after the new subnets were added. All the existing environment resources such as the Data Lake, FreeIPA, and any existing Cloudera Data Hub clusters will remain within the subnets originally defined during environment creation.

For CDP_CLI

You can use the CDP CLI to add a subnet to an existing environment:

```
cdp environments update-subnet --environment <environment-name> --subnet-
ids <>
```

> **Note:**
>
> Even when adding a new subnet, you must include the existing subnets in the --subnet-ids <> attribute.

Adding security groups to an environment

You can add additional security groups to an existing environment. These security groups will be used for all Cloudera Data Hub clusters created within the environment in the future.

Before you begin

These steps assume that you have already created the security groups that you want to add to the environment.

Required role: EnvironmentAdmin or Owner

Steps

1. Navigate to Cloudera Management Console > Environments and click on the environment you want to modify.
2. Click the Summary tab.
3. Scroll down to the Security Access section.
4. Click the ☑ (edit) button, then under Select Security Access Type select the Provide Existing Security Groups option and select the security groups that you want to add to the Environment.
5. Click the ▤ (save) button.

   You should see the new security groups listed in the Security Access section.

   > **Note:**
   >
   > The newly added security groups will only be used for the Cloudera Data Hub clusters created within the environment after the new security groups were added. All the existing environment resources such as the Data Lake, FreeIPA, and any existing Cloudera Data Hub clusters will remain within the security groups originally defined during environment creation.

Adding a root SSH key to an environment

You can add an additional SSH public key to an existing environment. This SSH public key will be used for root access to the Cloudera Data Hub clusters created within the environment in the future.

Before you begin

These steps assume that you have already have the SSH key pair that you want to add.

Required role: EnvironmentAdmin or Owner

Steps

1. Navigate to Cloudera Management Console > Environments and click on the environment you want to modify.
2. Click the Summary tab.
3. Scroll down to the Advanced section.
4. Click the ☑ (edit) button, then under Root SSH select New SSH public key and paste the SSH public key that you want to add to the Environment. If your environment is on AWS, you can also select Existing SSH public key and then select the SSH public key as long as you have previously uploaded it on AWS.
5. Click the ▤ (save) button.

   Note:

   The newly added SSH key will only be used for root access to the Cloudera Data Hub clusters created within the environment after the new SSH key pair was added. All the existing environment resources such as the Data Lake, FreeIPA, and any existing Cloudera Data Hub clusters will remain accessible via the SSH key pair originally defined during environment creation.

Changing an environment's credential

You can change the credential attached to an environment as long as the new credential provides the required level of access to the same AWS account as the old credential.

Required roles:

- EnvironmentAdmin or Owner of the environment
- SharedResourceUser or Owner of the credential

Steps

> **For Cloudera UI**
> 1. Log in to the Cloudera web interface.
> 2. Navigate to the Cloudera Management Console.
> 3. Select Environments from the navigation pane.
> 4. Click on a specific environment.
> 5. Navigate to the Summary tab.
> 6. Scroll down to the Credential section.
> 7. Click 🖉 to access credential edit options.
> 8. Select the new credential that you would like to use.
> 9. Click ◉ to save the changes.

For CDP CLI

If you would like to delete a credential from the CDP CLI, use:

```
cdp environments update-environment-credential --environment-name <value>
  --credential-name <value>
```

Enabling environment telemetry

You can optionally enable workload analytics so that diagnostic information about job and query execution is sent to Cloudera Observability for Cloudera Data Hub clusters. Similarly, you can optionally enable logs collection so that logs generated during deployments will be automatically sent to Cloudera.

Required role:

- PowerUser can set environment telemetry settings for the whole tenant.
- EnvironmentCreator can set environment telemetry settings during environment registration.
- EnvironmentAdmin or Owner can set environment telemetry settings for an existing environment.

Enabling workload analytics

If you enable workload analytics, diagnostic information about job and query execution is sent to Cloudera Observability for Cloudera Data Hub clusters created within all environments. This is disabled by default and can be enabled:

- For the whole tenant:

  - From the Cloudera web interface by navigating to Cloudera Management Console>Global Settings>Telemetry, by turning on the Enable Workload Analytics.
  - Or from the CDP CLI using the following command:

    ```
    cdp environments set-account-telemetry --workload-analytics
    ```

- For a specific environment only:

  - During environment creation from the Cloudera web interface, by turning on the Enable Workload Analytics option under Logs Storage and Audits in the environment creation wizard.
  - For an existing environment, from environment details > Telemetry by turning on the Enable Workload Analytics.
  - For an existing environment, from the CDP CLI using the following command:

    ```
    cdp environments set-telemetry-features --environment-name <some-name> -
    -workload-analytics
    ```

The environment-level setting overrides the tenant-level setting.

> **Note:**
>
> Only Cloudera Data Hub clusters created after enabling workload analytics on an environment will send data to Cloudera Observability. Cloudera Data Hub clusters created before workload analytics was enabled will not start sending data to Cloudera Observability if workload analytics is enabled for their parent environment.

Enabling cluster logs collection

If you enable cluster logs collection, logs generated during deployments will be automatically sent to Cloudera. This is disabled by default and can be enabled:

- For the whole tenant:

  - From the Cloudera web interface by navigating to Cloudera Management Console>Global Settings>Telemetry, by turning on Enable Cluster Logs Collection.
  - Or from the CDP CLI using the following command:

    ```
    cdp environments set-account-telemetry --report-deployment-logs
    ```

- For a specific environment only:

  - During environment creation from the Cloudera web interface, by turning on the Enable Cluster Logs Collection option under Logs Storage and Audits in the environment creation wizard.
  - For an existing environment, from environment details > Summary >Telemetry by turning on the Enable Cluster Logs Collection.
  - For an existing environment, from the CDP CLI using the following command:

    ```
    cdp environments set-telemetry-features --environment-name <some-name> -
    -report-deployment-logs
    ```

The environment-level setting overrides the tenant-level setting.

Disable cloud storage logging for an existing environment

By default, Cloudera sends collected service logs from VM nodes to the cloud storage that you provided for logs during environment registration. In some cases, you may want to disable this for an existing environment.

You can disable this option from environment details > Summary > Telemetry by turning off Enable Cloud Storage Logging.

Note:
Disabling this option will affect only Cloudera Data Hub clusters created after the option was disabled.

Related Information
Configuring lifecycle management for logs on AWS

Defining anonymization rules for Cloudera logs

Cloudera includes a set of default anonymization rules and allows you to define custom anonymization rules in order to remove sensitive information from Cloudera logs.

Use PCRE convention for writing custom anonymization rule patterns.

Anonymization rules are applied to the following logs:

- Cluster logs collected during deployments and automatically sent to Cloudera. See Enabling environment telemetry.
- Diagnostics logs that can be collected for troubleshooting and sent to Cloudera Support. See Generating a VM-based diagnostic bundle.

Note: Anonymization rules are only applied to environments created after the rules were added in Cloudera.

Default anonymization rules

Cloudera includes a set of default anonymization rules that anonymize the following:

Anonymization rule (PCRE)	Replacement	Description
\b([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\-\._]*[A-Za-z0-9])@(([A-Za-z0-9]|[A-Za-z][A-Za-z0-9\-]*[A-Za-z0-9])\.)+([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\-]*[A-Za-z0-9])\b	email@redacted.host	Email addresses
\d{4}[^\w]\d{4}[^\w]\d{4}[^\w]\d{4}	XXXX-XXXX-XXXX-XXXX	Credit card numbers
\d{3}[^\w]\d{2}[^\w]\d{4}	XXX-XX-XXXX	SSN
FPW\:\s+[\w|\W].*	FPW: [REDACTED]	FreeIPA (workload) password
cdpHashedPassword=.*[']	[CDP PWD ATTRS REDACTED]	Hashed FreeIPA (workload) password.

Creating anonymization rule patterns

Use PCRE convention for writing anonymization rule patterns. For each pattern, come up with a replacement string.

Define custom anonymization rules

You can define custom anonymization rules in Cloudera. The anonymization rules are only applied to environments created after the rules were added in Cloudera.

Required role: PowerUser

Steps

For Cloudera UI

1. Once you have created the rules, navigate to Cloudera web interface > Cloudera Management Console > Global Settings > Telemetry > Anonymization rules.
2. Default rules are pre-populated.
3. Click on New rule and add a pattern and replacement string for your rule. Repeat for multiple rules.
4. Test the rules from the same page on the UI under Test rules:

   a. Under Input text paste an example text with sensitive content that should get anonymized by the rules that you added.
   b. Click Test all rules.
   c. The sensitive content should be removed nad replaced in the output printed in the Anonymized result text box.
5. Click Save Changes.

Note:

You can use the Set defaults button if you would like to revert to the default rules.

For CDP CLI

1. If you would like to add new rules, you should first prepare the patterns and replacement strings, and then test them with the following command:

```
cdp environments test-account-telemetry-rules --cli-input-json {
    "testInput": "Email: myemail@cloudera.com",
    "rules": [
        {
            "value": "\\b([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\\-\\._]*[A-
Za-z0-9])@(([A-Za-z0-9]|[A-Za-z][A-Za-z0-9\\-]*[A-Za-z0-9])\\.)+([A-Za-
z0-9]|[A-Za-z0-9][A-Za-z0-9\\-]*[A-Za-z0-9])\\b",
            "replacement": "email@redacted.host"
        }
    ]
}
```

2. Run the following command to get your current telemetry settings in JSON format:

```
cdp environments get-account-telemetry
```

3. Copy the JSON file that you obtained in the output of this command and paste it into a text editor.

4. Update the JSON file, updating the settings or adding new rules.

> **Note:**
>
> Make sure to preserve all the existing rules, or else they will be deleted. Also, make sure to pass the workloadAnalytics and cloudStorageLogging parameters. If you don't pass all of the parameters, the parameters that are not passed will get reset to their default values.

5. Once you have the JSON file updated, run the cdp environments set-account-telemetry command. For example:

```
cdp environments set-account-telemetry --cli-input-json {
    "workloadAnalytics": true,
    "cloudStorageLogging": true,
    "rules": [
        {
            "value": "\\b([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\\-\\._]*[A-
Za-z0-9])@(([A-Za-z0-9]|[A-Za-z][A-Za-z0-9\\-]*[A-Za-z0-9])\\.)+([A-Za-
z0-9]|[A-Za-z0-9][A-Za-z0-9\\-]*[A-Za-z0-9])\\b",
            "replacement": "email@redacted.host"
        }
    ]
}
```

Adding a customer managed encryption key to a Cloudera environment running on AWS

By default, Data Lake and FreeIPA's Amazon Elastic Block Store (EBS) volumes and Relational Database Service (RDS) are encrypted using a default key from Amazon's KMS, but you can optionally configure encryption using Customer Managed Keys (CMK). Cloudera Data Hub clusters inherit environment's encryption key by default but you have an option to specify a different CMK during Cloudera Data Hub creation.

Amazon offers the option to encrypt EBS volumes and RDS instances using a default key from Amazon's Key Management System (KMS) or using an external customer-managed KMS. By default, Data Lake and FreeIPA are encrypted using the default key from Amazon's KMS present in the region where the environment is running, but you can provide a customer-managed KMS key instead of the default key.

> **Note:** Only single-Region KMS keys are supported by Cloudera. Using a multi-Region KMS key will result in an error.

By default, Data Hubs use the same default key from Amazon's KMS or CMK as the parent environment but you have an option to pass a different CMK during Cloudera Data Hub creation.

Encryption is configured for block devices and root devices. When encryption is configured for a given cluster, it is automatically applied to all the disk devices of any new VM instances added as a result of cluster scaling or repair.

This documentation covers the following topics:

- AWS disk encryption options
- AWS prerequisites for using a CMK
- Registering a new AWS environment and specifying a CMK
- Adding a CMK to an existing AWS environment

Environment and Cloudera Data Hub encryption options

To learn about encryption options that Cloudera offers for Data Lake, FreeIPA, and Cloudera Data Hub clusters, refer to Environment and Data Hub encryption options.

AWS prerequisites for using a CMK

You can use your existing AWS CMK or create a new AWS CMK.

For detailed requirements, refer to AWS Requirements: Customer managed encryption keys.

Registering an AWS environment with a CMK

You can specify an existing customer managed key (CMK) during AWS environment registration and the encryption key will be used to encrypt the EBS volumes and RDS instances running in the environment.
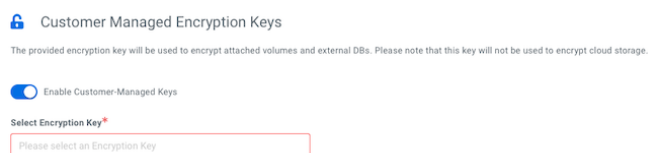
Steps

For Cloudera UI

You can register your environment as described in Register an AWS environment from CDP UI, just make sure that on the Data Access and Audit page you enable the following:

1. Under Customer-Managed Keys, click Enable Customer-Managed Keys.
2. In the same section, select the CMK:

The following screenshot shows the UI options:

🔓 Customer Managed Encryption Keys

The provided encryption key will be used to encrypt attached volumes and external DBs. Please note that this key will not be used to encrypt cloud storage.

⬤ Enable Customer-Managed Keys

Select Encryption Key*

Please select an Encryption Key

For CDP CLI

You can use your usual CDP CLI command to create an environment with a CMK, just add the --encryption-key-arn parameter and provide the encryption key created earlier.

Use the following CDP CLI command to create an environment with the encryption key created earlier. Replace the placeholders with actual values. For example <ENVIRONMENT-NAME> should be replaced with an actual name. The parameter important for this feature are highlighted:

```
cdp environments create-aws-environment \
--environment-name <ENVIRONMENT-NAME> \
--credential-name <EXISTING_CREDENTIAL> \
--region "<REGION>" \
--security-access cidr=<CIDR> \
--authentication publicKeyId="<SSH_KEY>" \
--log-storage storageLocationBase=<BUCKET_URL>,instanceProfile=<IDBROKE
R_IP> \
--vpc-id <VPC_ID> \
--subnet-ids <SUBNETS \
--encryption-key-arn <ENCRYPTION_KEY_ARN>
```

The ARN of the encryption key created earlier should be passed in the parameter --encryption-key-arn

If the customer-managed encryption key ARN is not passed, then the AWS region-specific default encryption key is used for encrypting EBS volumes and RDS instances.

You can obtain more complete commands using the instructions in Obtain CLI commands for registering an
environment.

Setting a CMK for an existing AWS environment

You can set a CMK for an existing environment. The CMK will be only used for encrypting disks of Cloudera Data
Hub clusters created after the CMK was added.

> **Note:**
>
> The CMK added to an existing environment will only be used for encrypting disks of Data Hubs created after
> the CMK was added. Data Lake and FreeIPA disks and databases are not encrypted with the CMK.

Steps

For Cloudera UI

You can add an encryption key for an existing environment that does not have an encryption key set by navigating
to the Summary page of the environment.

1. In the Cloudera Management Console, navigate to Environments and click on the environment for which you
   would like to set a CMK.
2. Click on the Summary tab.
3. Scroll down to the Customer Managed Encryption Key section.
4. If no CMK has been set, you will see a message stating that there is no customer-managed key enabled.
5. Click on the edit button in the top right corner of the tab.
6. Click on the toggle button next to Enable Customer-Managed Keys to enable adding a CMK.
7. Provide the encryption key ARN. You can copy and paste it from AWS KMS:

8. Click Save.

For CDP CLI

You can add an encryption key for an existing environment that does not yet have encryption enabled.

```
cdp environments update-aws-disk-encryption-parameters \
   --environment-name <ENVIRONMENT_NAME>
   --encryption-key-arn <CMK_ARN>
```

# Deploying Cloudera in multiple AWS availability zones

By default, Cloudera provisions Data Lake, FreeIPA and Cloudera Data Hub clusters in a single AWS availability zone (AZ), but you can optionally choose to deploy them across multiple availability zones (multi-AZ). It is possible to enable it either for all or some of these components.

Single-AZ vs multi-AZ deployment

Each AWS region has multiple availability zones, which act as failure domains, preventing small outages from affecting entire regions. By default, a Cloudera environment (FreeIPA and Data Lake) and Cloudera Data Hub clusters running in it are deployed across one subnet, meaning that each of them has VMs spread across a single availability zone (as on AWS every subnet is related to a single availability zone). This is illustrated in the following diagram, where all of the resources run in the eu-central region, and each of them is in a different subnet and availability zone (eu-central-1a for the FreeIPA, eu-central-1b for the Data Lake, and eu-central-1c for the Cloudera Data Hub clusters):
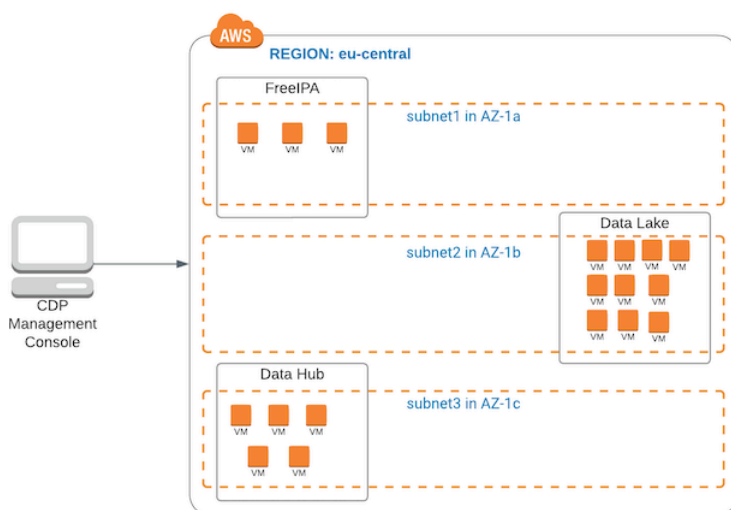
If you choose to deploy your Cloudera environment (FreeIPA and Data Lake) and Cloudera Data Hub clusters across multiple subnets and availability zones, each of these components is spread across three or more availability zones, providing high availability and fault tolerance. This is illustrated in the following diagram, where each of the components (FreeIPA, Data Lake, and Cloudera Data Hub clusters) is deployed across multiple subnets and multiple availability zones: eu-central-1a, eu-central-1b, and eu-central-1c:
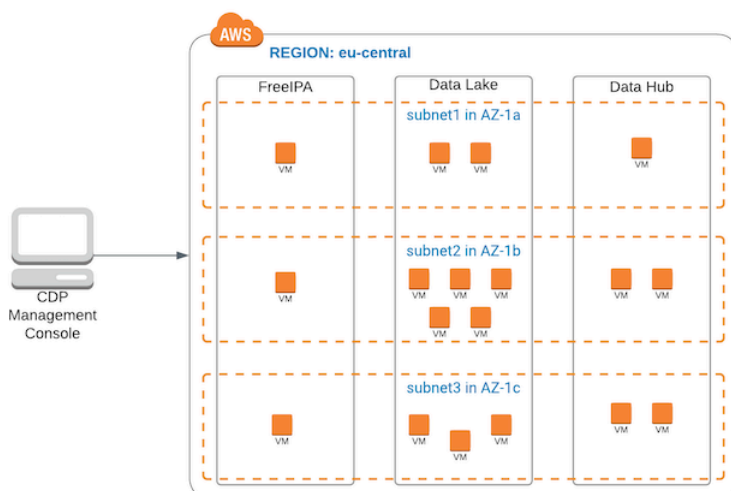
The Data Lake deployed in this setup is the enterprise Data Lake, which has enough nodes to utilize multiple availability zones for providing high availability and fault tolerance.

Multi-AZ can be enabled for Data Lake, FreeIPA, and Cloudera Data Hub clusters. It is possible to enable it either for all or some of these components.

Instances within a host group are distributed across availability zones equally, with the least used availability zone(s) preferred. For example, if there is a group of instances across three availability zones with 100 instances in AZ-1, 30 in AZ-2 and 30 in AZ-3, when an upscale of 10 nodes is requested, Cloudera provisions 5 instances in AZ-2 and 5 in AZ-3.

When a failure happens and a cluster needs to be repaired, the replacement VMs are always provisioned in the same subnet and availability zone as the old ones since the detached disks can only be reattached to a VM in the same availability zone. This means that if there is an availability zone outage, cluster repair is not possible.

Use cases

A multi-AZ Data Lake and FreeIPA constitute a resilient environment that provides a solid basis for multi-AZ Data Hubs and Cloudera data services. Data Hubs and Cloudera data services depend on the FreeIPA instance in the Data Lake to provide DNS resolution. Deploying FreeIPA across multiple availability zones ensures that critical DNS resolution is available in the event of an availability zone outage. Furthermore, an enterprise Data Lake provides high availability, and additional compute and memory resources for key SDX services and is recommended for production workloads.

Deploying your Cloudera Data Hub clusters across multiple availability zones is key if your mission-critical applications depend on HBase and Kafka. Multiple availability zone deployment for operational workloads is considered best practice by the cloud vendors. It ensures that your applications can continue to run in the event of a single availability zone outage.

When an entire availability zone fails, HBase utilized in Cloudera Operational Database automatically rebalances regions among the remaining instances in the cluster to maintain availability. The write-ahead log (WAL), which is replicated across the three availability zones is automatically replayed by the newly assigned region servers in other availability zones to ensure writes to the database are not lost.

When using the multi availability zone feature, Cloudera ensures that Kafka replicates partitions across brokers in different availability zones. During an availability zone failure this ensures that no data is lost and applications can continue to access the data they need. Cruise Control, which is deployed alongside every Kafka cluster in Cloudera on cloud, detects that topics need to be rebalanced to the remaining brokers. Once the availability zone is back online, you can repair your Kafka cluster, restoring the initial broker distribution across availability zones. Afterwards Cruise Control kicks in and ensures that all topic partitions are balanced across the cluster.

Limitations

The following limitations apply when deploying a multi-AZ Cloudera:

- When an AZ is down, you cannot create a new Cloudera Data Hub, and create or activate Cloudera data services within the environment. Existing workloads will continue to work.
- When an AZ is down, you cannot resize, stop, or restart Cloudera Data Hub clusters.

- When an AZ is down, existing running workloads will continue to run. Some Data Lake services will still be functioning (because they run across more than one AZ) but others may be down if the AZ where they are running is down.

  The following diagram shows how Data Lake services are distributed across AZs (assuming that there are 3 AZs per region). The Data Lake services marked in red run in a single AZ only and will be down if that AZ is down:

AWS REGION: eu-central

AZ-1a	AZ-1b	AZ-1c

Master
HBase Gateway
HDFS Fallover Controller
HDFS Gateway
Hive Gateway
Hive Metastore
Kafka Gateway
Namenode
Ranger Admin
Solr Gateway

Master
HBase Gateway
HDFS Fallover Controller
HDFS Gateway
Hive Gateway
Hive Metastore
Kafka Gateway
Namenode
Ranger Admin
Solr Gateway

Gateway
Atlas Server
Cloudera Manager
HBase Gateway
HDFS Gateway
Kafka Gateway
Knox Gateway
Ranger Admin
Solr Gateway
Zookeeper Server

Gateway
Atlas Server
HBase Gateway
HDFS Gateway
Kafka Gateway
Knox Gateway
Ranger Admin
Solr Gateway
Zookeeper Server

Core
Datanode
HBase Gateway
HBase Master
HBase Region Server
HDFS Gateway
HDFS Journal Node
Hive Gateway
Kafka Broker
Kafka Gateway
Solr Server
Solr Gateway

Core
Datanode
HBase Gateway
HBase Master
HBase Region Server
HDFS Gateway
HDFS Journal Node
Hive Gateway
Kafka Broker
Kafka Gateway
Solr Server
Solr Gateway

Core
Datanode
HBase Gateway
HBase Master
HBase Region Server
HDFS Gateway
HDFS Journal Node
Hive Gateway
Kafka Broker
Kafka Gateway
Solr Server
Solr Gateway

Auxiliary
Ranger UserSync
Ranger TagSync
Zookeeper Server

ID Broker
IDBroker

ID Broker
IDBroker

The following table shows which services aren't affected by an AZ outage (column 1 and 2) and those that may potentially be affected (column 3):

Services running in all available AZs	Services running in two AZs	Services running in a single AZ

Datanode	Atlas Server	Cloudera Manager
HBase Gateway	HDFS Failover Controller	Ranger UserSync
HBase Master	Hive Metastore	Ranger TagSync
HBase Region Server	IDBroker	
HDFS Gateway	Knox Gateway	
HDFS Journal Node	Namenode	
Hive Gateway	Ranger Admin	
Kafka Broker	Solr Gateway	
Kafka Gateway	Zookeeper Server	
Solr Server		

As shown in the diagram and the table, the following three services will be down if the AZ where they are running is down:

- Cloudera Manager - When Cloudera Manager is down, you cannot perform administrative tasks that you would normally perform via Cloudera Manager.
- Ranger UserSync - When Ranger UserSync is down, existing users are not impacted, but synchronizing new users to Ranger is disrupted. For example, if you add a new user and sync it to FreeIPA while Ranger UserSync is down, the newly added user will not be synced to Ranger and as a result the user will not be able to access workload data.
- Ranger TagSync - When Ranger TagSync is down, existing tags and policies are not impacted, but synchronizing new, updated, or removed tags with an external metadata service such as Apache Atlas and creating Ranger policies with these tags is disrupted.
- Single AZ environments or clusters cannot be converted to multi-AZ.
- While CDP CLI allows you to specify multiple subnets per Cloudera Data Hub cluster's host group, due to load balancer requirements only one subnet per AZ can be configured for the Cloudera Manager node group of the Cloudera Data Hub cluster.
- While by default Cloudera deploys Data Lake, FreeIPA, and Cloudera Data Hub clusters using CloudFormation, in case of the multi-AZ setup no CloudFormation is used. This affects vertical scaling: After vertical scaling (for example during cluster repair or upgrade) Cloudera always provisions the original instance types. For example, if you are using vertical scaling runbooks, repair and upgrade operations will use the previous instance type and root volume settings.

Enabling multi-AZ

Multi-AZ can be enabled for Data Lake, FreeIPA, and Cloudera Data Hub clusters running on AWS. It is possible to enable it either for all or some of these components.

To enable multi-AZ, you should register a multi-AZ Cloudera environment with an enterprise Data Lake. You can also create multi-AZ Cloudera Data Hub clusters within any existing environment. Detailed steps are provided below.

Register a multi-AZ environment

You can register a multi-AZ AWS environment via Cloudera UI or CDP CLI. You may choose to enable multi-AZ for Data Lake only or for FreeIPA only. There is no requirement to enable both.

Steps

For Cloudera UI

Register your environment as usual, just make sure to do the following:

1. On the Data Access and Data Lake Scaling page, select the enterprise Data Lake.
2. On the same page, scroll down and in the bottom of the page enable the Advanced Options.

3. In the Network and Availability section enable the Enable Multiple Availability Zones for Data Lake toggle button in order to enable multi-AZ for Data Lake. The option is disabled by default. The option only appears when an enterprise Data Lake is selected.

4. On the Region, Networking, and Security page, scroll down and in the bottom of the page enable the Advanced Options.

5. In the Network and Availability section enable the Enable Multiple Availability Zones for Data Lake toggle button in order to enable multi-AZ for FreeIPA. The option is disabled by default.

For CDP CLI

Register your AWS environment via CDP CLI as usual, but when creating a template for the environment registration commands, add the following to enable multi-AZ:

- In the environment creation command, add the multiAz=true to the --free-ipa option.
- In the Data Lake creation command, add the --multi-az option.
- In the Data Lake creation command, make sure to specify the enterprise Data Lake.

The following example illustrates these necessary additions, which are emphasized using bold typeface:

```
cdp environments create-aws-environment \
   --environment-name tb-multiaz-env \
   --credential-name tb-cldr-acc \
   --region "eu-central-1" \
   --security-access cidr=0.0.0.0/0 \
   --no-enable-tunnel \
   --authentication publicKeyId="test" \
   --log-storage storageLocationBase=s3a://cb-group/test,instanceProfile=ar
n:aws:iam::152413716728:instance-profile/mock-idbroker-admin-role \
   --vpc-id vpc-03e505817e3619238 \
   --subnet-ids subnet-013855b2gc32c2cd8 subnet-02b9054ef829374fe subnet-0
85c9ff36b38c0b35 \
   --free-ipa instanceCountByGroup=3,multiAz=true
```

```
cdp environments set-id-broker-mappings \
   --environment-name tb-multiaz-env \
   --data-access-role arn:aws:iam::152413716728:role/mock-idbroker-admin-
role \
   --ranger-audit-role arn:aws:iam::152413716728:role/mock-idbroker-admin-
role \
   --set-empty-mappings
```

```
cdp datalake create-aws-datalake \
   --datalake-name tb-multiaz-dl \
   --environment-name tb-multiaz-env \
   --cloud-provider-configuration instanceProfile=arn:aws:iam::15241371672
8:instance-profile/mock-idbroker-assumer,storageBucketLocation=s3a://cb-
group/test \
   --scale ENTERPRISE_HA \
   --runtime 7.2.12 \
   --no-enable-ranger-raz \
   --profile mowdev \
   --multi-az
```

Create multi-AZ Cloudera Data Hub clusters

If you would like to create multi-AZ Cloudera Data Hub clusters, see Creating a multi-AZ Cloudera Data Hub on AWS.

Defining custom tags

In the Cloudera Management Console clusters user interface, you can define tenant-level or environment-level custom tags across all instances and resources provisioned in your organization's cloud provider account.

Resource tagging

When you create an environment or other resources shared across your cloud provider account, Cloudera clusters automatically adds default tags to the Cloudera-created resources in your cloud provider account. You can also define additional custom tags that Cloudera applies to the cluster-related resources in your account.

You can use tags to protect the cloud resources of your Cloudera environment. Using the tags, you can exclude the resources that should not be removed during housekeeping or security deletion activities that can result in data corruption and data loss.

Default tags

By default, Cloudera applies certain tags to cloud provider resources whenever you create the resource, for example, an environment.

Cloudera applies the following tags by default:

- Cloudera-Resource-Name: the workload-appropriate Cloudera resource name. For example, an IPA CRN for an IPA, a data lake CRN for a data lake, or a Cloudera Data Hub CRN for a Cloudera Data Hub cluster. This CRN serves as a unique identifier for the resource over time.
- Cloudera-Creator-Resource-Name: Cloudera resource name of the Cloudera user that created the resource.
- Cloudera-Environment-Resource-Name: name of the environment with which the resource is associated.

Custom tags

There are two types of custom tags that you can define in the Cloudera Management Console: tenant-level tags that apply to Cloudera-created resources across your organization's entire cloud provider account, and environment-level tags.

In the Cloudera Management Console user interface, you can define tenant-level tags across all instances and resources provisioned in your organization's cloud provider account. These resources include not only provisioned instances, but disks, networks, and other resources as well. In your cloud provider account you can search or filter on either the tag key or value. Tenant-level tags cannot be overridden during environment creation.

In addition to tenant-level tags, you can also define environment-level tags. Environment-level tags are inherited by the resources specific to an environment. For example, environment-level tags are inherited by the following resources:

- FreeIPA
- Data Lakes
- Cloudera Data Hub clusters
- Data Warehouses
- Cloudera Operational Databases

As with tenant-level tags, you can search or filter on the key tag or key value in your cloud provider account.

> **Note:** Cloudera applies custom tags during creation of the resources. For example, you can only define environment-level tags during environment registration. If you want to add or update cloud provider resource tags, you must do so through the cloud provider API.

For more information about using tags on cloud provider resources, consult AWS, Azure, or Google Cloud documentation. It is your responsibility to ensure that your tags meet your cloud provider requirements.

Supported services

While some Cloudera services such as Cloudera Data Hub inherit environment-level tags, others require that you add tags when provisioning or enabling the data service. The following table shows how tags can be added for various Cloudera services:

Cloudera service	How to add tags
Data Lake	Inherits tenant or environment level tags.
FreeIPA	Inherits tenant or environment level tags.
Cloudera Data Engineering	Does not inherit tenant or environment level tags but you can define tags when enabling Cloudera Data Engineering.
Cloudera Data Hub	Inherits tenant or environment level tags and you can add more tags when creating a Cloudera Data Hub.
Cloudera Data Warehouse	Inherits tenant or environment level tags.
Cloudera DataFlow	Inherits tenant level tags and you can add more tags when enabling Cloudera DataFlow.
Cloudera AI	Does not inherit tenant or environment level tags but you can define tags when creating a Cloudera AI workbench.
Cloudera Operational Database	Inherits tenant or environment level tags and you can add more tags when creating a Cloudera Operational Database database via CLI.

Tenant-level tag requisites

Defining tenant-level tags depends on the provider. The following table summarizes the available characters for property keys and values for AWS, Azure, and GCP.

Provider/Property	AWS	Azure	GCP
Maximum number of tags per resource	50	50	64
Key minimum length (characters)	1	1	1
Key maximum length (characters)	127	512	63
Allowed characters	• letters (a-z, A-Z) • numbers (0-9) • spaces representable in UTF-8 • the following characters: + - = . _ : / @ • cannot start with aws)	• letters (a-z) • numbers (0-9) • can not start with microsoft, azure, or windows	• letters (a-z) • numbers (0-9) • spaces representable in UTF-8 • the following characters: _
Value minimum length (characters)	1	1	1
Value maximum length (characters)	255	255	63
Allowed characters	• letters (a-z, A-Z) • numbers (0-9) • spaces representable in UTF-8 • the following characters: + - = . _ : / @	• letters (a-z) • numbers (0-9)	• letters (a-z) • numbers (0-9) • spaces representable in UTF-8 • the following characters: _

Defining tenant-level tags

Required roles: PowerUser can define tags for the whole tenant.

• EnvironmentAdmin or Owner can set environment telemetry settings for a specific environment.

Steps

1. In the Cloudera Management Console, click Global Settings Tags .
2. Click Add.
3. Define both a key and a value for the tag. Both the key and the value must be between 4- 255 characters, with the following restrictions:

   Key

   > Allowed characters are hyphens (-), underscores (_), lowercase letters, and numbers. Keys must not start with 'aws'. Keys must start with a lowercase letter and must not start or end with spaces.

   Value

   > Allowed characters are hyphens (-), underscores (_), lowercase letters, and numbers. Values must not start or end with spaces. You can configure variables in the *{{{VARIABLENAME}}}* format. The following variables are supported for tenant-level tags:
   >
   > - *{{{CLOUDPLATFORM}}}* = AWS, AZURE or GCP
   > - *{{{USERNAME}}}* = Cloudera username
   > - *{{{USERCRN}}}* = Customer Resource Number (CRN) of Cloudera user
   > - *{{{CREATORCRN}}}* = CRN of Cloudera resource creator
   > - *{{{TIME}}}* = Actual time
   > - *{{{ACCOUNTID}}}* = Cloudera account ID
   > - *{{{RESOURCECRN}}}* = Generated string of Cloudera resource CRN

   Note: The policies that determine the allowed characters in custom tags in CDP are based on the policies of all three supported cloud providers (AWS, Azure, GCP). It is not possible to disable or override these policies.

4. Click Add, and if necessary repeat the process for additional tags.

   Note: Tenant-level tags are applied only to resources created after you define the tag. Any changes to the tag value do not propagate to existing resources that have the tag.

Defining environment-level tags

You define environment-level tags during environment registration.

Required roles: EnvironmentCreator can set tags for a specific environment during environment registration.

Steps

1. In the Cloudera Management Console, click Environments Register Environment .
2. Proceed through the environment registration steps.
3. After you define data access, add any environment-level tags by clicking Add and defining the tag key and value.

Related Information

Tagging your Amazon EC2 resources (AWS)

Updating instance metadata to IMDSv2

Cloudera can use IMDSv2 or IMDSv1 for accessing EC2 instance metadata from a running instance.

Cloudera currently uses IMDSv2 for accessing EC2 instance metadata on all newly created Data Lakes, FreeIPA clusters, and Cloudera Data Hub clusters, as long as an IMDSv2-compatible image is used. Prior to Cloudera supporting IMDSv2, Data Lakes, FreeIPA clusters, and Cloudera Data Hub clusters used IMDSv1; These clusters created with IMDSv1 can now be updated to IMDSv2 as long as an IMDSv2-compatible image was used to create the cluster.

Update an existing cluster to IMDSv2

You can update an existing Data Lake, FreeIPA cluster and Cloudera Data Hub cluster that is currently using IMDSv1 to IMDSv2. This is a zero downtime operation and does not disrupt any existing processes or jobs

Prerequisites

The following prerequisites needs to be met:

- Image must be compatible with IMDSv2

  You can only update to IMDSv2 if the image used for creating the cluster is compatible with IMDSv2. If the image is not compatible or if a cluster is already using IMDSv2, the Update to IMDSv2 button is grayed out:
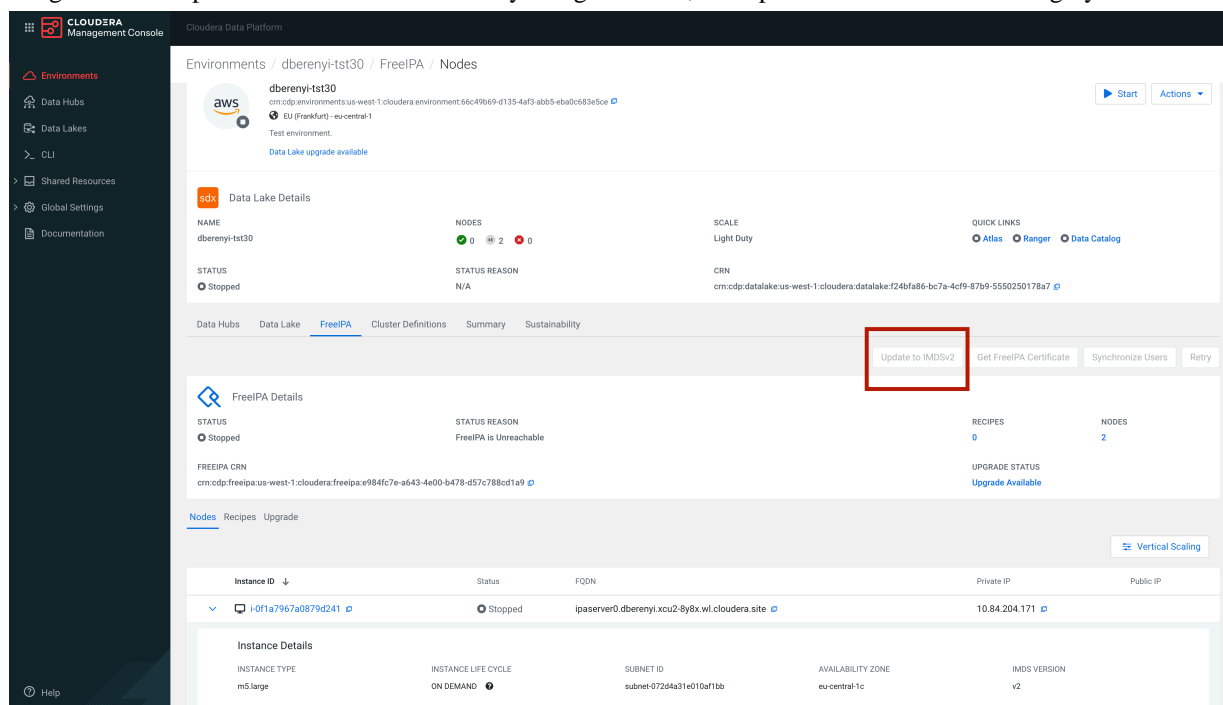
  If you would like to check image compatibility manually, see Checking if cluster image is compatible with IMDSv2 on page 65.

- You may need to update recipes.

  If you are using recipes, you need to update them first to ensure that they are compatible with IMDSv2. See:

  - Update FreeIPA recipes
  - Update Data Lake recipes
  - Update Data Hub recipes

  For a quick check, you can search for usage of "169.254.169.254" IP in the recipe content (as this is the IP pointing to the AWS IMDS). You can find an example in the Retrieve instance metadata documentation. You may also want to review How Instance Metadata Service Version 2 works.

- The ec2:ModifyInstanceMetadataOptions permission must be granted to the cross-account IAM role associated with the CDP environment.
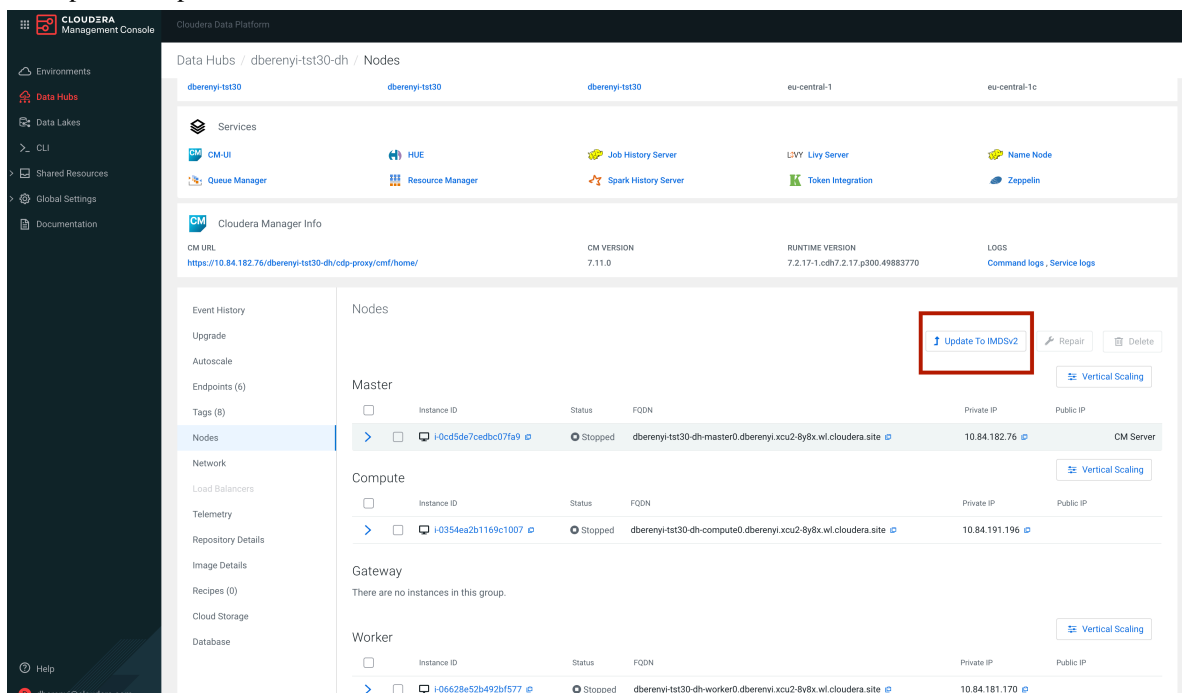
Required roles: EnvironmentAdmin or Owner of the environment

Steps

For Cloudera UI

1. In the Cloudera Management Console, navigate to Data Lake, FreeIPA or Cloudera Data Hub details.
2. Navigate to Nodes.

3. The option to Update to IMDSv2 is available as follows:

4. The update should happen within a few seconds. You can track the status in event history.

If you would like to verify that the update happened correctly, navigate to one of the EC2 instances in your AWS console.

Once the update is complete, the Update to IMDSv2 button is grayed out.

For CDP CLI

Steps CLI

Use the following commands to update a cluster from IMDSv1 ro IMDSv2:

Data Lake:

```
cdp datalake update-to-aws-imds-v2 --crn <SPECIFY_CRN>
```

FreeIPA:

```
cdp environments update-freeipa-to-aws-imds-v2 --environment-crn <SPECIFY_CRN>
```

Cloudera Data Hub:

```
cdp datahub update-to-aws-imds-v2 --crn <SPECIFY_CRN>
```

The update should happen within a few seconds.

Checking if cluster image is compatible with IMDSv2

To check if the image is compatible, follow these steps:

Steps

For Cloudera UI

1. In the Cloudera Management Console, navigate to your cluster.
2. Navigate to the Image details tab.
3. Click on the image ID.
4. Scroll down to Package Versions details of the image:

Image Catalogs / cdp-default / 85e571ef-725e-4b3e-a6ce-…

Details of 85e571ef-725e-4b3e-a6ce-2c2878ff207c image

UUID	85e571ef-725e-4b3e-a6ce-2c2878ff207c	Description	7.2.18.0 OS Update Release
OS Type	RHEL8	Stack Name	Cloudera Runtime
Cluster Manager Version	7.12.0.0	Stack Version	7.2.18
Created On	Apr 26, 2024	Published On	May 2, 2024

Package Versions of 85e571ef-725e-4b3e-a6ce-2c2878ff207c image

blackbox-exporter	0.19.0	cdh-build-number	51297892
cdp-logging-agent	1.3.2_b1	cdp-minifi-agent	1.22.07
cdp-prometheus	2.36.2	cdp-request-signer	0.2.4
cdp-telemetry	1.3.2_b1	cem	2.0.99.0-9
cem_gbn	49762114	cfm	2.2.8.0-487
cfm_gbn	51325206	cloudbreak_images	3972f928a2d08658b38feee0bf449c1bb43e7327
cm	7.12.0.0	cm-build-number	51300666
composite_gbn	51474779	csa-dh	1.12.0.0-199
csa-dh_gbn	51461452	imds	v2
inverting-proxy-agent	3.0.7-b1	inverting-proxy-agent_gbn	41924420
java	8	java11	11.0.23
java17	17.0.11	java21	21.0.3
java8	1.8.0_412	metering_agent	2.0.0
node-exporter	1.3.1	psql	14
psql11	11.22	psql14	14.11
python36	3.6.8-38.module+el8.5.0+12207+5c5719bc	python38	3.8.16-1.module+el8.8.0+18967+20d359ae.1
python39	3.9.16-1.module+el8.8.0+20025+f2100191.2	salt	3001.8
salt-bootstrap	0.13.6-2022-05-20T08:57:17	source-image	ami-039ce2eddc1949546
stack	7.2.18		

5. Package versions include a variable called "imds". Find this variable and ensure that its value is "v2

For CDP CLI

1. You can use the describe CDP CLI command of the given cluster (FreeIPA, Data Lake, Cloudera Data Hub. The commands are as follows:

- cdp datahub describe-cluster
- cdp datalake describe-datalake
- cdp environment describe-environment

2. The response should contain image and image catalog JSON details.

Here is an example from image catalog where an image is compatible (note the "package-versions" section, which includes "imds" version. If the value is "v2", the image is compatible. For example, see the highlighted "imds" section in the following image catalog file:

```
{
        "created": 1709942743,
        "published": 1709951597,
        "date": "2024-03-09",
        "description": "Official Cloudbreak image",
        "images": {
          "aws": {
            "eu-central-1": "ami-0faa3e25092764091",
            "us-west-1": "ami-043b94655fb95edf1",
            "us-west-2": "ami-0f9ed7da8f775d00e"
          }
        },
        "os": "redhat8",
        "os_type": "redhat8",
        "uuid": "9b84a914-a15a-4856-94a9-3cfda722b0b8",
        "package-versions": {
          "blackbox-exporter": "0.19.0",
```

```
                    "cdp-logging-agent": "1.3.2_b1",
                    "cdp-minifi-agent": "1.22.07",
                    "cdp-prometheus": "2.36.2",
                    "cdp-request-signer": "0.2.4",
                    "cdp-telemetry": "1.3.2_b1",
                    "cloudbreak_images": "8cf7cd165b58343011091c4908f9f796d4dceb
92",
            "imds": "v2",
                    "inverting-proxy-agent": "3.0.7-b1",
                    "inverting-proxy-agent_gbn": "41924420",
                    "java": "8",
                    "java11": "11.0.21",
                    "java17": "17.0.9",
                    "java21": "21.0.2",
                    "java8": "1.8.0_392",
                    "metering_agent": "2.0.0",
                    "node-exporter": "1.3.1",
                    "psql": "14",
                    "psql11": "11.22",
                    "psql14": "14.11",
                    "python36": "3.6.8-38.module+el8.5.0+12207+5c5719bc",
                    "python38": "3.8.16-1.module+el8.8.0+18967+20d359ae.1",
                    "python39": "3.9.16-1.module+el8.8.0+20025+f2100191.2",
                    "salt": "3001.8",
                    "salt-bootstrap": "0.13.6-2022-05-20T08:57:17",
                    "source-image": "ami-039ce2eddc1949546"
                },
                "tags": {
                    "fips-mode": "disabled",
                    "hardening": "cis_server_l1"
                }
            },
```

Restricting access for Cloudera services that create their own security groups on AWS

The security groups that you select to use during environment registration are only used for the Data Lake, FreeIPA, Cloudera Data Hub clusters, and Cloudera Operational Databases running in that environment. The Kubernetes-based Cloudera services (Cloudera Data Engineering, Cloudera DataFlow, Cloudera Data Warehouse, and Cloudera AI) create their own security groups with rules that should be restricted separately.

The following table explains where and when you can restrict these rules:

Note: If you do not restrict these endpoints, Cloudera defaults to opening access to all (0.0.0.0/0).

Cloudera service	Type of access that can be restricted	When and where to restrict	Link to related documentation
Cloudera DataFlow	Admin access to the Kubenetes API Server endpoint can be restricted. End user access can be restricted.	Restrict admin access to Kubernetes endpoints during or after enabling Cloudera DataFlow via the Kubernetes API Server Endpoint Access setting. Restrict end user access to the the Cloudera DataFlow endpoints during or after enabling Cloudera DataFlow via the Load Balancer Endpoint Access setting. .	Enabling Cloudera DataFlow for an environment Managing Kubernetes API Server user access
Cloudera Data Engineering	Admin access to Kubenetes endpoints can be restricted. End user access can only be restricted manually from the AWS management console.	Restrict admin access to Kubernetes endpoints during enabling Cloudera Data Engineering via the Whitelist IPs parameter. Restrict end user access manually from the AWS management console.	Enabling Cloudera Data Engineering and Limiting Incoming Endpoint Traffic for Data Engineering Services
Cloudera Data Warehouse	Both admin access to Kubernetes endpoints and end user access are always set to the same range that can be set in environment activation settings. While the access to the Kubernetes endpoints is a combination of the Cloudera Control Plane's CIDR and your CIDR provided in environment activation settings, the access to the end user access points (JDBC, UI) is only your CIDR provided in environment activation settings.	In Cloudera Data Warehouse environment's activation settings.	Restricting access to endpoints in AWS environments and Editing the IP CIDRs in the trusted list for endpoints in AWS environments
Cloudera AI	There are two separate options, one for admin access to Kubernetes endpoints and another for end user access.	During Cloudera AI workbench provisioning, under Network Settings: • The Load Balancer Source Ranges parameter can be used to restrict end user access. • Selecting the checkmark Restrict access to Kubernetes API server to authorized IP ranges allows you to restrict admin access to Kubernetes endpoints.	Provisioning Cloudera AI workbenches

Configuring lifecycle management for logs on AWS

To avoid unnecessary costs related to Amazon S3 cloud storage, you should create lifecycle management rules for your cloud storage bucket used by Cloudera for storing logs so that these logs get deleted once they are no longer useful.

Some examples of Cloudera logs stored in cloud storage are: cloudera server logs, cloudera agent logs, autossh logs, freeipa logs, ranger audit logs, datahub services logs, datalake logs, cm management services logs, and so on. These logs are mostly useful for troubleshooting, so they can be periodically deleted.

AWS allows you to set up lifecycle management rules for your S3 buckets. For example, you can set a specified expiration period for a cloud storage location so that the files in that location get deleted automatically on a scheduled basis. Cloudera recommends that you do this for the cloud storage location that you provided to Cloudera for log storage.

Consider the following when setting up lifecycle management rules:

- As logs and data locations may overlap with each other (in case the same bucket or container is used for both), ensure to use the correct path prefixes in order to delete only the logs. The prefixes are listed below.
- When setting an expiration period, consider how long you would like to keep the logs to allow enough time for troubleshooting. For example, in case your Data Lake, FreeIPA or Cloudera Data Hub cluster is ever down, you should be able to access the logs for troubleshooting.

Prefixes based on AWS environment's logs location base

Prior to creating lifecycle management rules in S3, review this information to ensure that you use the correct path.

Note:

Path logic changed in February 2021. Starting in February 2021, the path automatically contains the cluster-logs folder as a peer of the cluster-backups folder, creating a better structural separation between logs and backups.

	The "cluster-logs" prefix is automatically generated if a bucket name without any subdirectories is used as logs location	The "cluster-logs" prefix is automatically generated if subdirectories are provided	If your environment was registered prior to February 2021: If you defined a sub-directory, then that subdirectory is used instead of "cluster-logs"
Logs location provided during environment registration	**s3a://my-bucket/**	**s3a://my-bucket/my-dl**	**s3a://my-bucket/my-dl**
FreeIPA prefix for lifecycle rule	cluster-logs/freeipa	my-dl/cluster-logs/freeipa	my-dl/freeipa
DataLake prefix for lifecycle rule	cluster-logs/datalake	my-dl/cluster-logs/datalake	my-dl/datalake
DataHub prefix for lifecycle rule	cluster-logs/datahub	my-dl/cluster-logs/datahub	my-dl/datahub

Creating expiration rules for an S3 bucket via AWS CLI

If you would like to create expiration rules via AWS CLI, refer to the following examples of JSON files in order to create your own JSON file.

If the log base location is s3://mybucket, create a JSON file similar to the following:

```
{
  "Rules": [
    {
      "Filter": {
          "Prefix": "cluster-logs/"
       },
      "Status": "Enabled",
      "AbortIncompleteMultipartUpload": {
          "DaysAfterInitiation": 7
       },
      "NoncurrentVersionExpiration": {
          "NoncurrentDays": 7
       },
      "Expiration": {
          "Days": 7
       },
      "ID": "cleanup cloud storage data after 7 days"

    }]
```

```
}
```

If the log base location is s3://mybucket/mypath and your environment was registered in February 2021 or later,
create a JSON file similar to the following:

```
{
 "Rules": [
  {
     "Filter": {
         "Prefix": "mypath/cluster-logs/"
     },
     "Status": "Enabled",
     "AbortIncompleteMultipartUpload": {
         "DaysAfterInitiation": 7
     },
    "NoncurrentVersionExpiration": {
        "NoncurrentDays": 7
     },
     "Expiration": {
         "Days": 7
     },
     "ID": "cleanup cloud storage data after 7 days"
  }]
}
```

If the log base location is s3://mybucket/mypath and your environment was registered prior to February 2021, create a
JSON file similar to the following:

> **Note:** Multiple rules are needed in this case to ensure that the cluster-backups folder (which is also located
> in "mypath") does not get deleted. Starting in February 2021, this is no longer needed because the cluster-
> backups folder is always a peer of the cluster-logs folder.

```
{
 "Rules": [
  {
     "Filter": {
         "Prefix": "mypath/freeipa"
     },
     "Status": "Enabled",
     "AbortIncompleteMultipartUpload": {
         "DaysAfterInitiation": 7
     },
    "NoncurrentVersionExpiration": {
        "NoncurrentDays": 7
     },
     "Expiration": {
         "Days": 7
     },
     "ID": "cleanup cloud storage data after 7 days (freeipa)"
  },
  {
     "Filter": {
         "Prefix": "mypath/datalake"
     },
     "Status": "Enabled",
     "AbortIncompleteMultipartUpload": {
         "DaysAfterInitiation": 7
     },
    "NoncurrentVersionExpiration": {
        "NoncurrentDays": 7
     },
     "Expiration": {
         "Days": 7
```

```
        },
        "ID": "cleanup cloud storage data after 7 days (datalake)"

    },
{

        "Filter": {
            "Prefix": "mypath/datahub"
        },
        "Status": "Enabled",
        "AbortIncompleteMultipartUpload": {
            "DaysAfterInitiation": 7
        },
        "NoncurrentVersionExpiration": {
            "NoncurrentDays": 7
        },
        "Expiration": {
            "Days": 7
        },
        "ID": "cleanup cloud storage data after 7 days (datahub)"
    }
]
}
```

Once you have the JSON file ready, post it with an AWS CLI command similar to:

```
aws s3api put-bucket-lifecycle-configuration /
--bucket mybucket /
--lifecycle-configuration file:///my/path/rules.json
```

If you have existing rules on the nodes, you can use the get-bucket-lifecycle-configuration command first, edit that JSON (adding the new rules there), and then post the updated JSON.

Creating expiration rules for an S3 bucket via S3 console

If you would like to create expiration rules via S3 console, navigate to your S3 bucket > Management tab > Create lifecycle rule.

The following screenshots show an example setup for a single prefix:

Related Information
Managing your storage lifecycle
Amazon S3 – Object Expiration
Enabling environment telemetry

Troubleshooting for RAZ-enabled AWS environment

This section includes common errors that might occur while using a RAZ-enabled AWS environment and the steps to resolve the issues.