Cloudera Management Console

Azure Environments

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

https://docs.cloudera.com/

Legal Notice

Contents

Working with Azure environments

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

Related Information

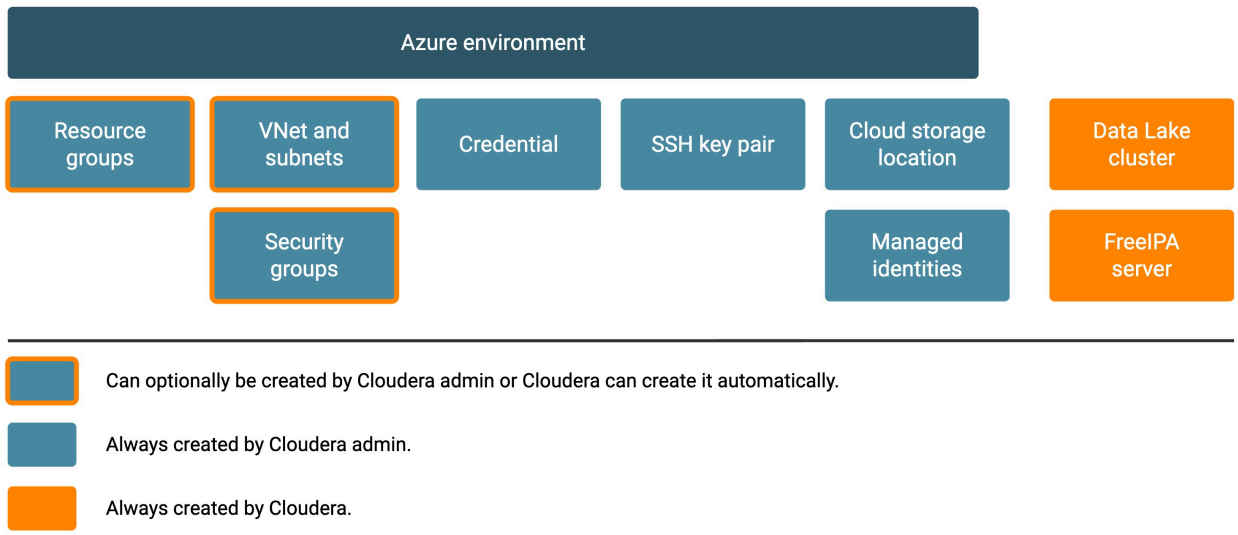
Managing provisioning credentials for Azure
Managing Data Lakes
Managing FreeIPA

Introduction to Azure 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 Azure data center.

The following diagram enumerates the components of an environment:



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 Azure account, or you can optionally pre-create them and specify 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 specified when registering an environment.
- The items in orange boxes are automatically provisioned on Azure 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 (called VNet on Azure) and subnets in which Cloudera resources are provisioned.
Security groups	Security groups (called NSGs on Azure) 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.
Credential	Cloudera uses the credential for authorization to provision resources (such as compute instances) within your cloud provider account. On Azure, credential creation involves creating an application and a service principal within the Azure Active Directory manually (app-based credential).
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 managed identities	When registering an environment, you must provide an ADLS Gen2 location for storing: • All workload cluster data • Cluster service logs and Ranger audits Furthermore, you must create and assign managed identities so that Cloudera can access the storage location.
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.
Resource groups	Cloudera creates multiple new resource groups for resources that it deploys on your Azure account.

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 Azure environment from the Cloudera UI

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

Before you begin

This assumes that you have already fulfilled the environment prerequisites described in Azure 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 Azure.
Microsoft Azure Credential (Required)	
Select Credential	Select an existing credential or select Create new credential. For instructions on how to create a credential, refer to Create an app-based credential.

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, -.
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 ADLS Gen2	
Enable Ranger authorization for ADLS Gen2 Identity	If you would like to use Fine-grained access control, enable this option and then select the Ranger RAZ managed identity created in the Minimal setup for cloud storage.
Data Access and Audit	
Assumer Identity (Required)	Select the Assumer managed identity created in Minimal setup for cloud storage.
Storage Location Base (Required)	Provide the ADLS Gen2 location created for data storage in Minimal setup for cloud storage.
Data Access Identity (Required)	Select the Data Lake Admin managed identity created in Minimal setup for cloud storage.
Ranger Audit Identity (Required)	Select the Ranger Audit managed identity 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.

Parameter	Description
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
Hardware and Storage	For each host group you can specify an instance type. For more information on instance types, see Sizes for virtual machines in Azure.
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 accessing and provisioning resources from Cloudera. If you would like to use a specific existing virtual network, the virtual network must be located in the selected region.
Resource Group	
Select Resource Group (Required)	You have two options: • Select one existing resource group. If you select this, all Cloudera resources will be provisioned into that resource group. • Select Create new resource groups to have Cloudera create multiple resource groups.
Customer Managed Encryption 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. For more information, refer to Customer managed encryption keys.
Select Encryption Key Resource Group	Select the resource group where the CMK is located.
Encryption key URL	Provide the URL of the key value where the CMK resides. This is the same as the key identifier that you can copy directly from Azure Portal.
Managed identity for encryption	If using Azure Database for PostgreSQL Flexible Server, you can optionally select a managed identity created for encrypting it. For more information, refer to Managed identity for encrypting Azure Database for PostgreSQL Flexible Server.
Network	
Select Network (Required)	You have two options: • Select the existing virtual network where you would like to provision all Cloudera resources. Refer to VNet and subnets. • Select Create new network to have a new network with three subnets created. **Warning:** On 30 September 2025, Microsoft will retire the default outbound access connectivity for virtual machines in Azure. You must configure an explicit outbound connectivity method. For more information, see Cloudera Customer Advisory-866 and Transitioning from Azure default outbound access documentation.
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.

Parameter	Description
Network CIDR (Required)	This option is only available if you select to create a new network. If you selected to create a new network, provide Network CIDR that determines the range of private IPs that VMs will use. This must be a valid private IP CIDR IP in IPv4 range. For example 10.10.0.0/16 are valid IPs. /16 is required to allow for enough IP addresses.
Create Private Subnets	This option is only available if you select to have a new network and subnets created. Is is turned on by default so that private subnets are created in addition to public subnets. If you disable it, only public subnets will be created. **Important:** For production deployments, Cloudera recommends that you use private subnets. Work with your internal IT teams to ensure that users can access the browser interfaces for cluster services.
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. If you are using your existing VPC, 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.
Create Private Endpoints	By default, the PostgreSQL Azure database provisioned for your Data Lake is reachable via a service endpoint (public IP address). To increase security, you can optionally select to have it reachable via a private endpoint instead of a service endpoint. **Note:** This option is only available if an existing resource group is selected. **Note:** Only the subnets that have Azure private endpoint network policies turned off are eligible for private endpoint creation. At least one such subnet is required. If you select to create a private endpoint and you are using your own VNet, you have two options: • Select “Create new private DNS zone” and Cloudera creates and manages a private DNS zone for you in the provided existing resource group. • Select your existing private DNS zone. If you select to create a private endpoint and you would like for Cloudera to create a new VNet, Cloudera creates a private DNS zone for you. For more information, refer to Private endpoint for Azure Postgres.
Create Public IPs	This option is disabled by default when Cluster Connectivity Manager is enabled and enabled by default when Cluster Connectivity Manager is disabled.
Flexible Server	During environment registration in Cloudera, the Flexible Server in public service mode is used by default, but you can specify to use the Flexible Server in private service mode (“Flexible Server with Private Link” or "Flexible Server with Delegated Subnet (deprecated)"). For more information, refer to Using Azure Database for PostgreSQL Flexible Server.
Proxies	
Select Proxy Configuration	Select a proxy configuration if previously registered. For more information refer to Setting up a proxy server.
Security Access Settings	

Parameter	Description
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, 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	
Enable **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.
Enable **User Defined Routing**	Enable User Defined Routing (UDR) in case public IPs are blocked for egress.In case you enable UDR, you must select the specific worker node subnet where the UDR is configured.
AKS Private DNS Zone ID	When selecting **Private Kubernetes Cluster**, you also need to select an existing private DNS zone or select creating a new private DNS zone by Cloudera on your Azure account for the database.
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 SSH public key (Required)	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 Azure. Refer to Defining custom tags.

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

Parameter	Description
Hardware and Storage	For each host group you can specify an instance type. For more information on instance types, see Sizes for virtual machines in Azure.
Cluster Extensions	
Recipes	You can optionally select and attach previously registered recipes to run on FreeIPA nodes. For more information, see Recipes.

9. Click Next.

10. On the Storage page, provide the following information:

Parameter	Description
Logs	
Logger Identity (Required)	Select the Logger managed identity created in Minimal setup for cloud storage.
Logs Location Base (Required)	Provide the ADLS Gen2 location created for log storage in Minimal setup for cloud storage.
Backup Location Base	Provide the ADLS Gen2 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 Azure environment from the CDP CLI

Once you’ve met the Azure cloud provider requirements, register your Azure environment.

Before you begin

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

Required role: EnvironmentCreator

Steps

Unlike in the Cloudera web interface, in CDP CLI environment creation is a three-step process with environment creation, setting IDBroker mappings and Data Lake creation being three separate steps. The easiest way to obtain the correct commands is to provide all parameters in the Cloudera web interface and then generate the CDP CLI commands on the last page of the wizard. For detailed steps, refer to Obtain CLI commands for registering an environment.

To learn more about how to create Compute Cluster enabled environments with CLI, see Enabling default Compute Cluster for new environments.

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.

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

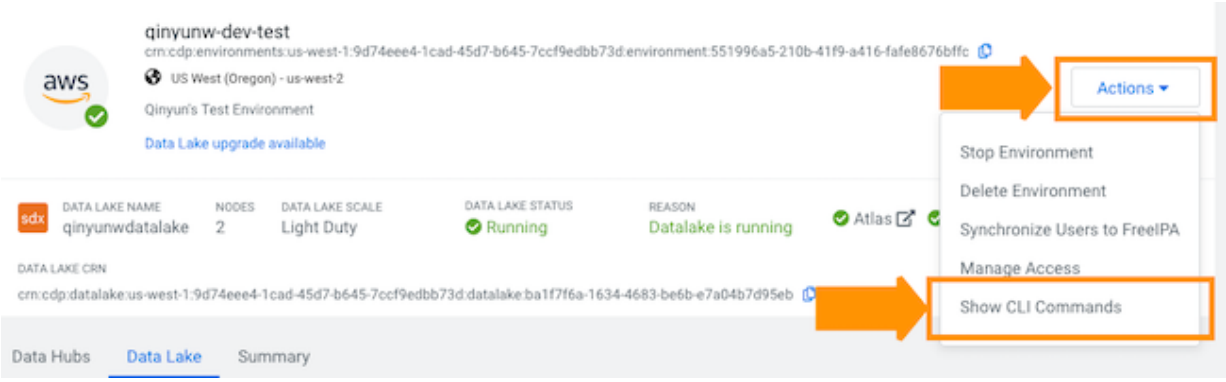
Obtain an environment template from an existing environment to create an environment with the exact same settings.

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

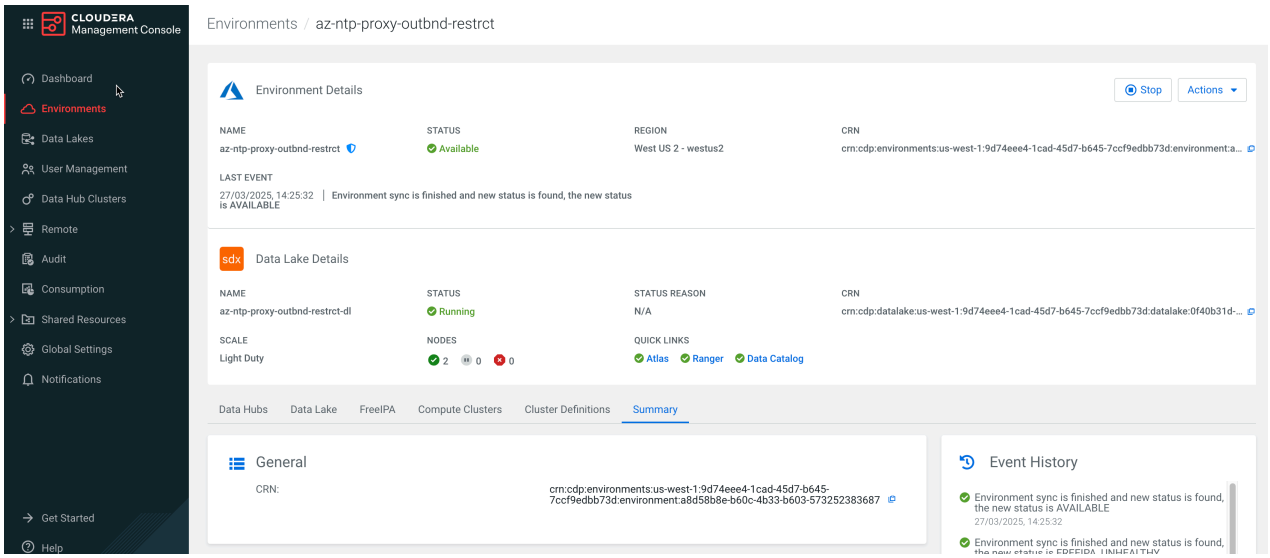
Steps

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.
Resource group	This lists the resource group(s) that your environment resources are associated with. During environment registration, you either provide an existing resource group or Cloudera creates multiple resource groups; There is no way to change this once an environment is running.
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.

Option	Description
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.
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.

Related Information

Understanding Cloudera Data Hub details

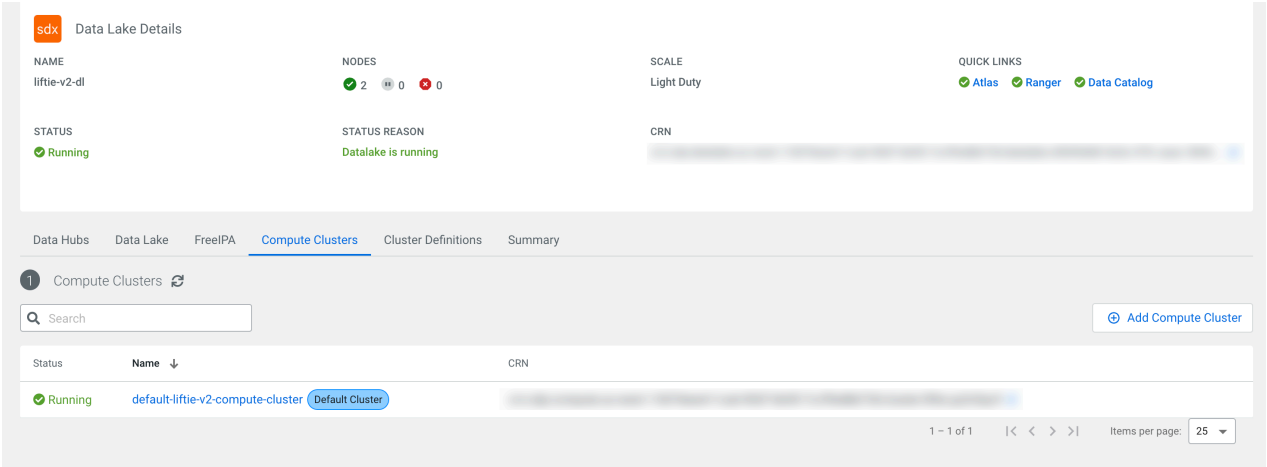
Understanding Data Lake details

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.

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

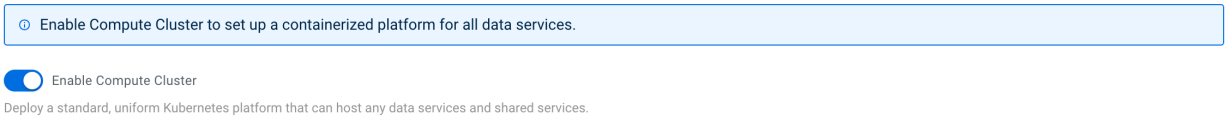
Before you begin

- Ensure that your Azure account has all the resources required by Cloudera.

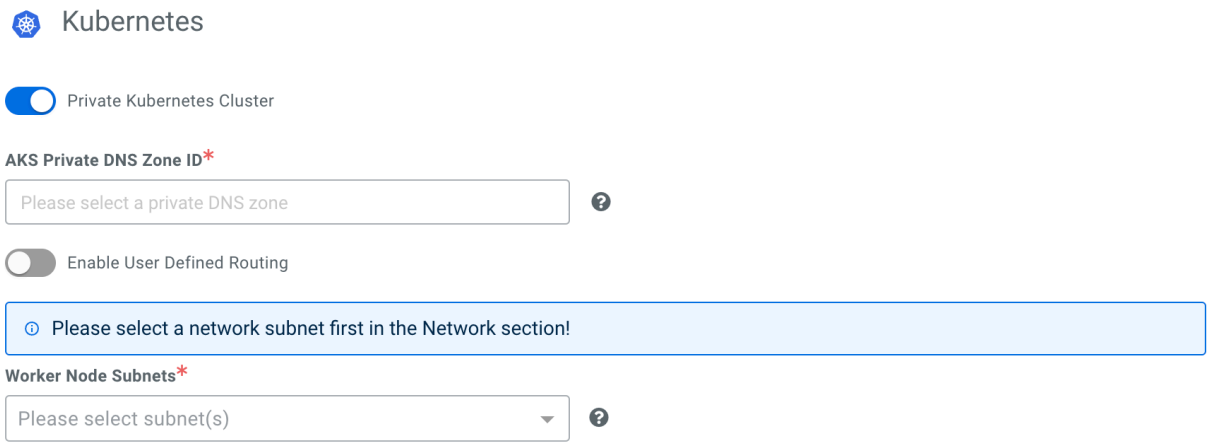
  For more information, see Azure subscription requirements.

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.



After completing the step for **Data Access and Data Lake Scaling**, configure the networking settings for Kubernetes with enabling **User Defined Routing**, and selecting **Private Kubernetes Cluster** or providing **Authorized IP Ranges** on the **Region, Networking and Security** page. When selecting **Private Kubernetes Cluster**, you also need to select an existing private DNS zone or select creating a new private DNS zone by Cloudera on your Azure account for the database. **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.



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

Using CDP CLI

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

For Without private cluster

```
cdp environments create-azure-environment \
--environment-name [***ENVIRONMENT NAME***] \
--credential-name [***CREDENTIAL NAME***] \
```

```
--region [***REGION***] \
--public-key [***PUBLIC SSH KEY***] \
--security-access [***SECURITY ACCESS CONFIGURATION***] \
--use-public-ip | --no-use-public-ip \
--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-azure-environment \
--environment-name [***ENVIRONMENT NAME***] \
--credential-name [***CREDENTIAL NAME***] \
--region [***REGION***] \
--public-key [***PUBLIC SSH KEY***] \
--security-access [***SECURITY ACCESS CONFIGURATION***] \
--use-public-ip | --no-use-public-ip \
--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***]

...
        "azureComputeClusterConfiguration": {
            "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-azure-compute-cluster
--environment-name [***ENVIRONMENT NAME***] \
--compute-cluster-configuration \
privateCluster=FALSE, \
kubeApiAuthorizedIpRanges=[***CIDR1***],[***CIDR2***]
```

```
workerNodeSubnets=[***SUBNET1***],[***SUBNET2***]
```

For With private cluster

```
cdp environments initialize-azure-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 role: EnvironmentAdmin

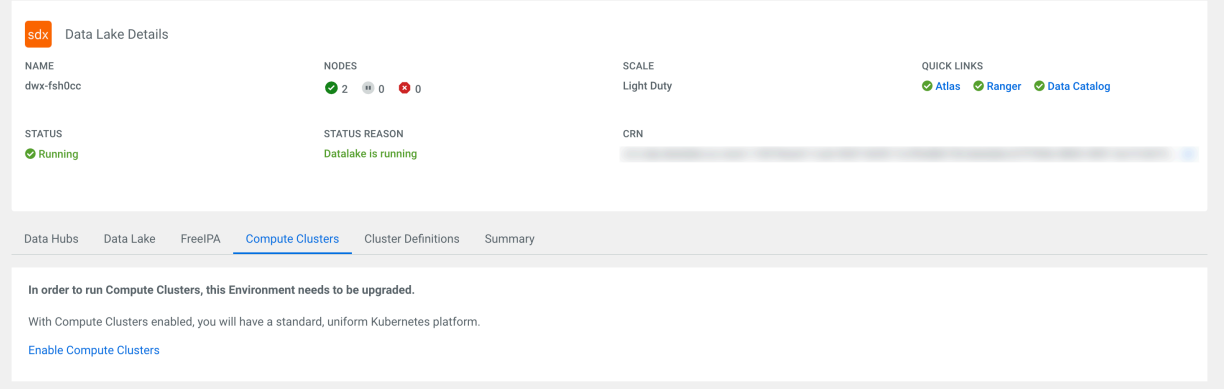
Before you begin

- Ensure that the environment has no default Compute Cluster provisioned.
- Ensure that the environment is started and available.

Using Cloudera Management Console

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

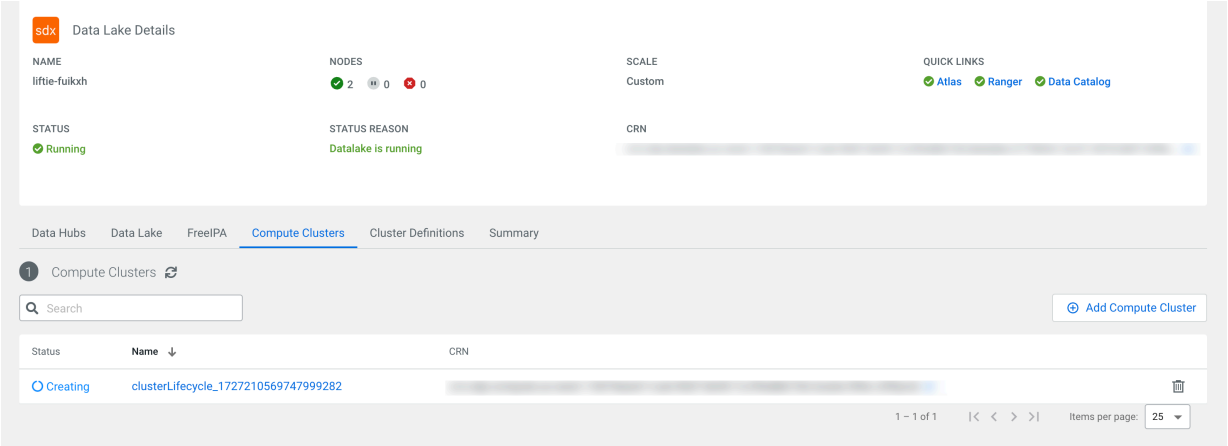
   The following message will indicate that Compute Cluster are not enabled for your environment:



3. Click Enable Compute Clusters.
4. Provide the necessary networking information for the **Kubernetes** cluster.
   a. Enable User Defined Routing (UDR) in case public IPs are blocked for egress.In case you enable UDR, you must select the specific worker node subnet where the UDR is configured.
   b. 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.
   c. 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.

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

For With private cluster

```
cdp environments initialize-azure-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***]
```

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***]
```

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.

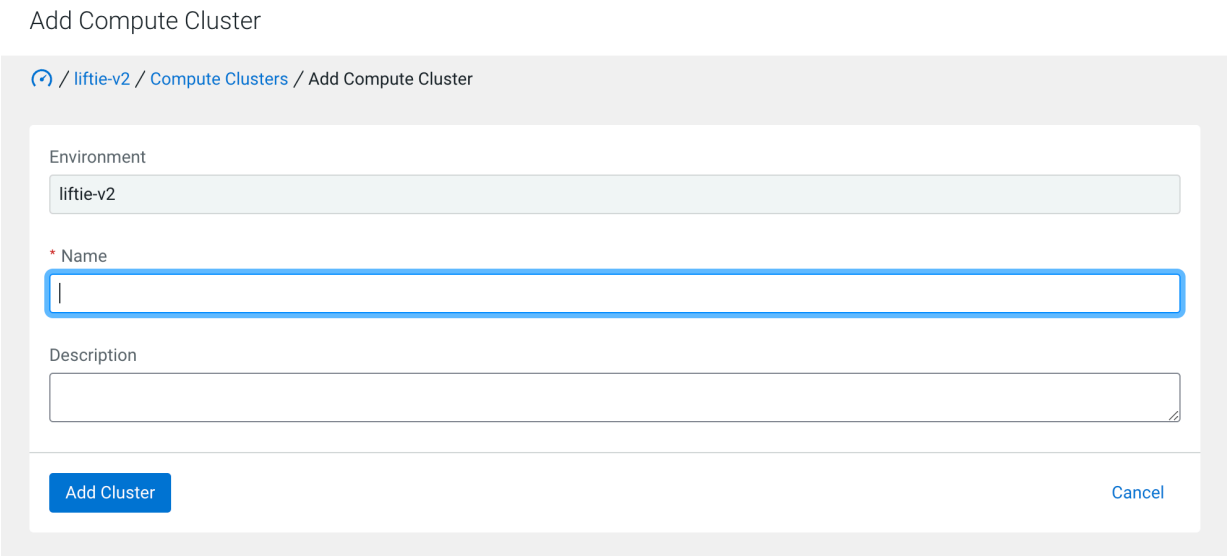
Required role: EnvironmentAdmin

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.



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 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 Kubeconfig to download the Kubeconfig file.

      1. Click Download Kubeconfig.

   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	
DELETE_INITIATED	Environment deletion request was registered and Cloudera is starting the deletion flow.

Environment status	Description
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.
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.

Environment status	Description
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.

   Note: To speed up future environment deployment, the cloudbreak-images resource group, which stores VHDs used for VM deployment, and its content are not deleted during environment deletion.
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 Azure 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 Azure 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 Azure 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 Azure account. To do this, log in to your Azure Portal and do one of the following:

- If Cloudera created all resource groups for your Cloudera resources, find it and delete these resource groups. The names of these resource groups will include the name of the environment.
- If you used your own existing resource group, navigate to it and delete all the resources that were created by Cloudera. The names of these resources will include the name of the environment.

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.

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 Azure subscription 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>
```

Updating AKS Private Zone ID for existing environments

You can add the AKS Private Zone ID for existing environments using CDP CLI.

Required role: EnvironmentAdmin or Owner

The AKS Private Zone ID can be configured during environment creation when an environment is going to be used for a Data Service. In case the AKS Private Zone ID is not configured for an existing environment, you can use the following CDP CLI command to update the details of the environment:

```
cdp environments update-data-service-resources
--environment [***ENVIRONMENT NAME***]
--data-services
"{"azure":
{"sharedManagedIdentity": [***USER-ASSIGNED MANAGED IDENTITY***],
"aksPrivateDnsZoneId": [***PRIVATE DNS ZONE ID***]}"
```

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

Configure lifecycle management for logs on Azure

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 Azure

By default, local Data Lake, FreeIPA, and Cloudera Data Hub disks attached to Azure VMs and the PostgreSQL server instance used by the Data Lake and Cloudera Data Hub clusters are encrypted with server-side encryption (SSE) using Platform Managed Keys (PMK), but you can optionally configure SSE with Customer Managed Keys (CMK).

The CMK can be specified during environment registration and, if present, is used for encrypting Data Lake, FreeIPA, and Cloudera Data Hub disks and PostgreSQL server instances. Alternatively, it is possible to specify it once the environment is running, in which case the CMK will only be used for Cloudera Data Hub clusters created after the CMK was added.

The disks that are attached to the VMs of the Data Lake, FreeIPA, and Cloudera Data Hub clusters will be associated with a Disk Encryption Set (DES) that is created with the key URL as the underlying encryption key version. The DES dedicated to the Cloudera environment will be created in the resource group of the environment before the FreeIPA launch at the beginning of the environment creation process.

This documentation covers the following topics:

- Azure prerequisites for using a CMK
- Registering a new Azure environment and specifying a CMK
- Creating a Cloudera Data Hub with the same CMK
- Checking whether a CMK exists for an Azure environment
- Adding a CMK to an existing Azure environment

Azure prerequisites for using a CMK

You can use your existing Azure vault and vault key or create a new Azure vault and vault key.

For detailed requirements, refer to Azure Prerequisites: Customer managed encryption keys.

Register an Azure environment with a CMK

You can specify an existing customer managed key (CMK) during Azure environment registration and the encryption key will be used to encrypt the VMs and databases running in the environment.

Steps

For Cloudera UI

You can register your environment as described in Register an Azure environment from the Cloudera UI, just make sure that on the Region, Networking and Security page you enable the following:

1. Under Customer-Managed Keys, click Enable Customer-Managed Keys.
2. In the same section, under Select Resource group select the resource group where the CMK is located.
3. Provide the URL of the key value where the CMK resides. This is the same as the key identifier that you can copy directly from Azure Portal.

For CDP CLI

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 parameters important for this feature are highlighted:

```
cdp environments create-azure-environment \
--environment-name <ENVIRONMENT_NAME> \
--credential-name <CREDENTIAL_NAME> \
--region <REGION> \
--security-access cidr=<YOUR-ORGANIZATION’S_CIDR> \
--public-key '<SSH_PUBLIC_KEY>' \
--log-storage <STORAGE_LOCATION> \
--description '<DESCRIPTION>' \
--resource-group-name <ENV_RESOUCE_GROUP_NAME> \
--encryption-key-resouce-group-name <CMK_RESOUCE_GROUP_NAME> \
--encryption-key-url <KEY_RESOUCE_ID>

cdp environments set-id-broker-mappings \
--environment-name <ENVIRONMENT_NAME> \
--data-access-role <DATA_ACCESS_IDENTITY>\
--ranger-audit-role <RANGER_AUDIT_IDENTITY> \
--set-empty-mappings

cdp datalake create-azure-datalake \
--datalake-name <DATALAKE_NAME> \
--environment-name <ENVIRONMENT_NAME> \
--cloud-provider-configuration <STORAGE_LOCATION_BASE_CONFIGURATION>
```

The encryption key can be present in a separate resource group than in which the environment is being created. The resource group in which encryption key is present can be provided using --encryption-key-resouce-group-name. Either --encryption-key-resouce-group-name or --resource-group-name must be present. Depending on the resource group where you environment and encryption key reside, there are three possible use cases:

- If your encryption key is in the same resource group as you wish to create the environment in, you can provide a common resource group with --resource-group-name parameter. In this case, you do not need to provide the --encryption-key-resouce-group-name parameter.
- If your encryption key is in a different resource group than you wish to create the environment in and you wish to use some other existing resource group for the environment, you can provide encryption key’s resource group with --encryption-key-resouce-group-name and the environment’s resource group with --resource-group -name.

If you don’t want to specify an existing resource group for the environment, you just need to provide the encryption key’s resource group using --encryption-key-resouce-group-name and the environment’s resource groups will be created by Cloudera.

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

If you are using Azure Database for PostgreSQL Flexible Server in Cloudera, you can also specify a managed identity and thus use the same CMK for encrypting the Azure Flexible Server database instance used by Cloudera. For more information, see Configuring a CMK for data encryption in Azure Database for PostgreSQL Flexible Server .

Create a Cloudera Data Hub on Azure with a CMK

Use Cloudera web interface or CDP CLI to create a Cloudera Data Hub cluster. Note that this doesn’t require any extra steps, so you can refer to the Cloudera Data Hub documentation.

The disks that are attached to the VMs of the Cloudera Data Hub cluster will be associated with a Disk Encryption Set (DES) that is created with the key URL as the underlying encryption key version. The DES dedicated to the Cloudera environment will be created in the resource group of the environment at the beginning of the environment creation process (before the FreeIPA launch, specifically).

Check Azure environment's CMK

You can check in an Azure environment's summary whether a CMK is used to encrypt a given environment.

Steps

For Cloudera UI

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.

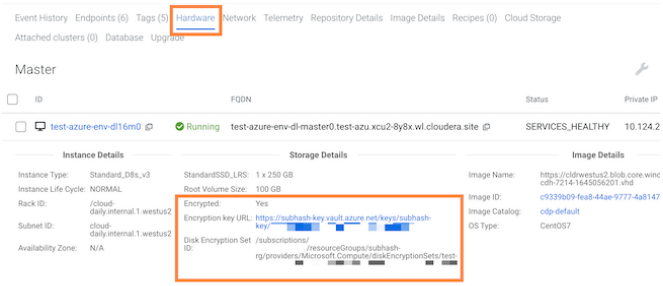
   If a CMK exists, the entry will look similar to:

4. You can also find the CMK information by clicking on the Data Lake tab and then navigating to the Hardware tab.
5. Click on >> on the right side of the screen next to any of the nodes to expand the node related information.
6. Under Storage Details, the following information is included about the CMK:

   - Whether the node has been encrypted with a CMK
   - CMK encryption key URL
   - Disk Encryption Set ID

   For example:

For CDP CLI

You can use the cdp environments update-azure-encryption-resources command to check environment's CMK encryption status.

If encryption key is set, the cdp environments update-azure-encryption-resources command will return the disk encryption set, encryption key, and encryption key's resource group details.

Check Cloudera Data Hub's CMK

You can check in a Cloudera Data Hub's hardware tab whether a CMK is used to encrypt it.

Note: A Cloudera Data Hub's CMK is the same as the one used for the environment in which the Cloudera Data Hub is running, so you can also find this information in the environment's details.

Steps

1. Navigate to the Cloudera Data Hub details > Hardware tab.
2. Click on >> on the right side of the screen next to any of the nodes to expand the node related information.
3. Under Storage Details, Under Storage Details, the following information is included about the CMK:
   - Whether the node has been encrypted with a CMK
   - CMK encryption key URL
   - Disk Encryption Set ID

Set a CMK for an existing Azure 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 Cloudera Data Hub clusters 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 the encryption key to 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 Encryption 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 following:
   a. Provide the encryption key URL. This is the same as the key identifier that you can copy directly from Azure Portal.
   b. If your encryption key resource group is different from the environment's resource group, provide the name of the resource group.
   c. Click Save.

For CDP CLI

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

If your encryption key resource group is same as environment resource group, use:

```
cdp environments update-azure-encryption-resources \
  --environment-name <ENVIRONMENT_NAME> \
  --encryption-key-url <CMK_URL> \
```

If your encryption key resource group is different from environment resource group, use:

```
cdp environments update-azure-encryption-resources \
  --environment-name <ENVIRONMENT_NAME> \
  --encryption-key-url <CMK_URL> \
```

```
  --encryption-key-resource-group-name <CMK_RESOURCE_GROUP>
```

If you are using Azure Database for PostgreSQL Flexible Server in Cloudera, you can also specify a managed identity and thus use the same CMK for encrypting the Azure Flexible Server database instance used by Cloudera. For more information, see Configuring a CMK for data encryption in Azure Database for PostgreSQL Flexible Server .

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

Provider/Property	AWS	Azure	GCP
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 the following words: 'azure', 'microsoft', and 'windows'. 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

Use Tags to Organize Azure Resources (Azure)

Using Azure Database for PostgreSQL Flexible Server

Cloudera uses Azure Database for PostgreSQL Flexible Server. The Flexible Server allows a highly available database to be deployed for Data Lake and Data Hub clusters. You can create Flexible Server instances with public access, where the Azure Database for PostgreSQL server is accessed through a public endpoint, or with private access, where the flexible server has no public endpoint accessible through the internet. The latter option requires a private DNS zone to be specified, and is either possible through the use of Azure Private Link for the Azure Database for PostgreSQL service, or, when Private Link is not used, it requires a delegated subnet to be created and added to your Cloudera Azure environment beforehand.

Using the Flexible Server offers the following benefits to Cloudera customers:

- Flexible Server allows you to deploy PostgreSQL version 14 and above. See *Supported PostgreSQL major versions in Azure Database for PostgreSQL - Flexible Server*.
- Unlike the previously used Single Server database instances, Flexible Server database instances can be stopped and restarted during Data Lake and Cloudera Data Hub cluster stop and restart. This offers a great cost-saving opportunity.

  Note: After 7 days of stopping the Flexible Server database instance, Azure automatically restarts the database. See *Limits in Azure Database for PostgreSQL - Flexible Server*.

- Flexible Server is multi-AZ capable and offers zone-redundant High Availability. With the Flexible Server, Data Lakes are backed with a highly-available PostgreSQL configuration of two instances. When using a multi-AZ deployment, the Flexible Server instances are deployed in multiple availability zones for additional fault tolerance.

For a detailed comparison of Single Server and Flexible Server offerings, refer to the *Comparison chart: Azure Database for PostgreSQL - Flexible Server vs. Single Server* in the Azure documentation.

Database server options

You can create Flexible Server instances with public access, where the Azure Database for PostgreSQL server is accessed through a public endpoint, or with private access , where the Flexible Server has no public endpoint accessible through the internet. There are two private flexible server options: Flexible Server using Azure Private Link for the Azure Database for PostgreSQL service or Flexible Server using a Delegated Subnet that needs to be created and added to your Cloudera Azure environment beforehand. Flexible Server with Private Link is the more advanced and recommended option as it offers easy connectivity to other Azure services that are utilizing Private Link for networking, the ability for a server to be reachable from public and private networks via both private and public addressing at the same time, and does not require delegated subnets to be created. For details, see *Private Link with Azure Database for PostgreSQL - Flexible Server*. Both private access options require a private DNS zone to be specified.

The Flexible Server with public access is used by default (as it does not require any special networking setup), but during environment creation you can specify to use the Flexible Server with private access, either with Private Link or with delegated subnet. It is also possible to specify to use the Single Server. However, the Flexible Server with Delegated Subnet and the Single Server options are marked as deprecated. It is not recommended to use these as they will be phased out. Azure Single Server only supports Postgres versions up to 11. Postgres versions newer than 11 require the use of Azure Flexible Server. For details, see *What happens to Azure Database for PostgreSQL - Single Server after the retirement announcement?*

New Cloudera environments on Azure automatically use Flexible Server with public endpoints and Cloudera Data Hub clusters automatically inherit the settings from the environment they run in, but you can also enable Flexible Server when creating a Cloudera Data Hub cluster.

In general, when registering an Azure environment and creating a Cloudera Data Hub cluster, you can choose to use Flexible Server, Flexible Server with Private Link, Flexible Server with Delegated Subnet (deprecated), Single Server (deprecated), or Single Server with Private Link (deprecated), but the exact options vary depending on the environment settings. You have the following options when Data Lake or Cloudera Data Hub cluster creation is initiated:

- If the parent environment has been configured for private access, and the database type is set as Flexible Server, the Cloudera Data Hub or Data Lake cluster is launched with Flexible Server with Private Link if no delegated subnets are specified.
- If the parent environment has been configured for private access, and the database type is set as Flexible Server, the Cloudera Data Hub or Data Lake cluster is launched with Flexible Server with delegated subnet if there is a delegated subnet specified.
- If the parent environment has been configured for private access, and the database type is not set, a cluster with a Flexible Server is launched. You must specify the database type as Single Server or Single Server with Private Link to launch the Cloudera Data Hub or Data Lake with a Single Server database.
- If the environment has been configured for public access, and the database type is not set specifically as Single Server, the Cloudera Data Hub or Data Lake cluster is launched with public Flexible Server.
- If the environment has been configured for public access, and the database type is set as Single Server, the Cloudera Data Hub or Data Lake cluster is launched with public Single Server.
- If an environment is configured with Azure Single Server and Service Endpoints, by default new Cloudera Data Hub clusters provisioned in this environment will be created with Public Flexible Servers. If in this case you would still like to use a Single Server instead, then the --database-type=SINGLE_SERVER CLI parameter should be used, or you can achieve the same on the UI under the Advanced Cloudera Data Hub options.

Limitations

The following limitations currently apply:

- If the server-side encryption (SSE) that the Data Lake or Cloudera Data Hub clusters are encrypted with is configured with Customer Managed Keys (CMK), an upgrade from Azure Single Server to Azure Flexible Server is currently not available. This capability is expected to be released soon.

In order to set up this feature, you should review Azure prerequisites and then you can enable Flexible Server during Azure environment registration in Cloudera or upgrade existing Azure Databse for PostreSQL Single Servers to Flexible servers as described in the linked documentation:

Related Information

Supported PostgreSQL major versions in Azure Database for PostgreSQL - Flexible Server
Limits in Azure Database for PostgreSQL - Flexible Server
Comparison chart: Azure Database for PostgreSQL - Flexible Server vs. Single Server
Private Link with Azure Database for PostgreSQL - Flexible Server
What happens to Azure Database for PostgreSQL - Single Server after the retirement announcement?
Upgrading Azure Single Server to Flexible Server

Azure prerequisites for Flexible Server

The following Azure prerequisites must be in place in order to use the Flexible Server with Cloudera:

- If you would like to use zone redundant-HA, select a region that supports it.
- The policy provided for the Azure credential must have the required permissions mentioned below.
- When Flexible Server with Delegated Subnets is selected from the Azure database type options, Flexible Server instances need to be deployed in a delegated subnet within the VNet. This is not required either when public endpoints are used or when Azure Private Link is used to achieve the “private service” mode.
- When deployed in “private service” mode, either using Azure Private Link or delegated subnets, providing the private DNS zone information is mandatory in order to be able to perform DNS resolution. This is not required when public endpoints are used.

Read on to learn how to meet these prerequisites.

Supported regions

Ensure that your selected region has compute availability for Flexible Server. See Flexible Server Azure Regions.

Credential permissions

The Prerequisites for the provisioning credential documentation lists role definitions that includes sufficient permissions for Flexible Server. The role definitions include the following additional permissions that are required for Flexible Server:

Permission	Description
Microsoft.DBforPostgreSQL/flexibleServers/write	Creates a server with the specified parameters or updates the properties or tags for the server.
Microsoft.DBforPostgreSQL/flexibleServers/delete	Deletes an existing server.
Microsoft.DBforPostgreSQL/flexibleServers/start/action	Starts an existing server.
Microsoft.DBforPostgreSQL/flexibleServers/stop/action	Stops an existing server.
Microsoft.DBforPostgreSQL/flexibleServers/firewallRules/write	Restrict which networks can access the database instance.
Microsoft.DBforPostgreSQL/flexibleServers/configurations/write	This permission is required to change Flexible Server configuration.

Delegated subnet

You have two options to deploy Flexible Server instances in private service mode (without public endpoints): using Private Link or without Private Link. Flexible Server with Private Link is the recommended option. If Private Link is not used , Flexible Server instances need to be deployed in a “delegated subnet” within the VNet. The Flexible Server with Delegated Subnet option, although still available on the UI, is deprecated and Cloudera recommends that you use the Azure Private Link option instead.

As mentioned in Azure documentation, to be able to utilize private access with VNet integration, when private endpoints are not used, it is a prerequisite to delegate a subnet to Microsoft.DBforPostgreSQL/flexibleServers. This delegation means that only Azure Database for PostgreSQL Flexible Servers can use that subnet. No other Azure resource types can be in the delegated subnet.

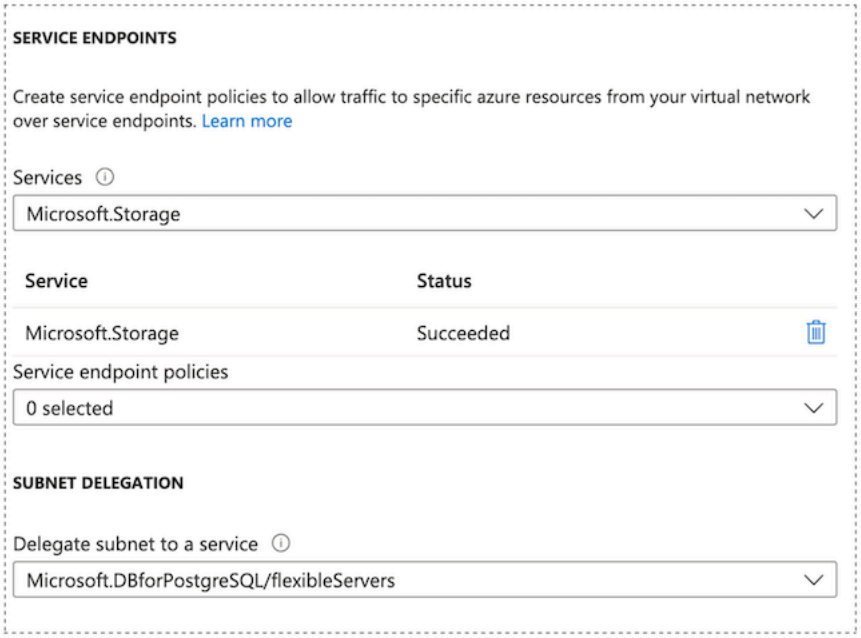
You need to create such a delegated subnet and provide it to Cloudera during environment registration. This delegated subnet will be used by Azure Database for PostgreSQL instances. The delegated subnet provided during environment registration will be used by default for all Azure Database for PostgreSQL instances used in Cloudera.

Note: Although you can currently select multiple subnets, the larger subnet is always used by Cloudera. That is, if there are two subnets provided, one with 128 available IPs and another with 256 available IPs, the second one will be picked. Even though you can select multiple delegated subnets, we recommend that you provide only one.

Creating a delegated subnet

For a step-by-step official guide on how to perform the delegation, see Delegate a subnet to an Azure service. For considerations on the delegated subnet’s sizing, see Virtual network concepts.

Here is a screenshot from Azure Portal showing the desired setting:



The Microsoft.Storage service endpoint is set automatically during deployment by Azure.

After the subnet has successfully been delegated, don't forget to record the full subnet ID or the name of the subnet for later use as an input (referred to as <delegated-subnet-id>). For example: /subscriptions/3ddda1c7-d1f5-4e7b-ac81-abcdefgh/resourceGroups/rg/providers/Microsoft.Network/virtualNetworks/vnet/subnets/subnet

Private DNS Zone

When using "private service" mode with Azure virtual network, a private DNS zone is mandatory in order to be able to perform DNS resolution.

As mentioned in Using a Private DNS Zone, a private DNS zone is used for creating a new Azure Database for PostgreSQL Flexible Server using private network access. Specifically, in case of Flexible Server with Private Link, the private DNS zone must be named privatelink.postgres.database.azure.com. In case of Flexible Server with a delegated subnet, a private DNS zone that ends with .postgres.database.azure.com should be created.

If you do not provide the private DNS zone, Cloudera creates a new private DNS zone in your resource group with the naming convention <resource-group-name>.flexible.postgres.database.azure.com. In case of Flexible Server with Private Link, the naming convention for the Cloudera-created private DNS zone is <resource-group-name>.privatelink.postgres.database.azure.com.

Note: If you are editing an existing environment to enable Flexible Server on it, you must provide a Private DNS Zone (unlike during environment creation where Cloudera can create it for you).

Note: Only one Private DNS Zone is created per resource group.

Creating a Private DNS Zone

When selecting a name for the Private DNS zone, in case of Flexible Server with Private Link, use privatelink.postgres.database.azure.com. Use one of the following forms for Flexible Server with Delegated Subnet:

[name1].[name2].postgres.database.azure.com

or

[name].postgres.database.azure.com

After the Private DNS zone has been created, don't forget to record the full Private DNS Zone ID, referred as <dns-zone-id>, for later use as an input. For example: /subscriptions/3ddda1c7-d1f5-4e7b-ac81-abcdefgh/resourceGroups/drg/providers/Microsoft.Network/privateDnsZones/flexible.private.postgres.database.azure.com

For instructions on how to create a Private DNS zone, see Quickstart: Create an Azure private DNS zone using the Azure portal.

Virtual Network Link

After the Private DNS Zone has been created in Azure, you need to link the VNet that you would like to use for your Cloudera environment to the Private DNS Zone. Once linked, resources hosted in that VNet can access the Private DNS Zone.

If you let Cloudera create the Private DNS Zone, then Cloudera creates the required Virtual Network Link connecting the VNet of your deployment with the created Private DNS Zone.

To learn more about virtual network links, see What is a virtual network link?.

Linking the virtual network

For instructions on how to link the VNet to the previously created Private DNS Zone, see Link the virtual network.

Enabling Flexible Server during Azure environment creation

During environment registration in Cloudera, the Flexible Server in public service mode is used by default. You also have two options to use the Flexible Server in private service mode: Flexible Server with Private Link and Flexible Server with Delegated Subnet. Flexible Server with Private Link is the recommended option, while Flexible Server with Delegated Subnet is still available but deprecated

When Cloudera is deployed in "private service" mode (without public endpoints), during environment creation you need to select the private DNS zone in case of using a Flexible Server with Private Link, or provide the following information when using a Flexible Server with delegated subnet:

- An ID of the delegated subnet
- A private DNS zone ID (optional parameter).

The virtual network link does not need to be specified as input. If you do not provide the private DNS zone, Cloudera creates it for you.

The steps below show you how to enable a private Flexible Server in Flexible Server with Private Link mode or Flexible Server with Delegated Subnet mode. If you would like to enable Flexible Server instances with public access, you do not need to do anything special, as this option is used by default.

Prerequisites

See Azure prerequisites for Flexible Server.

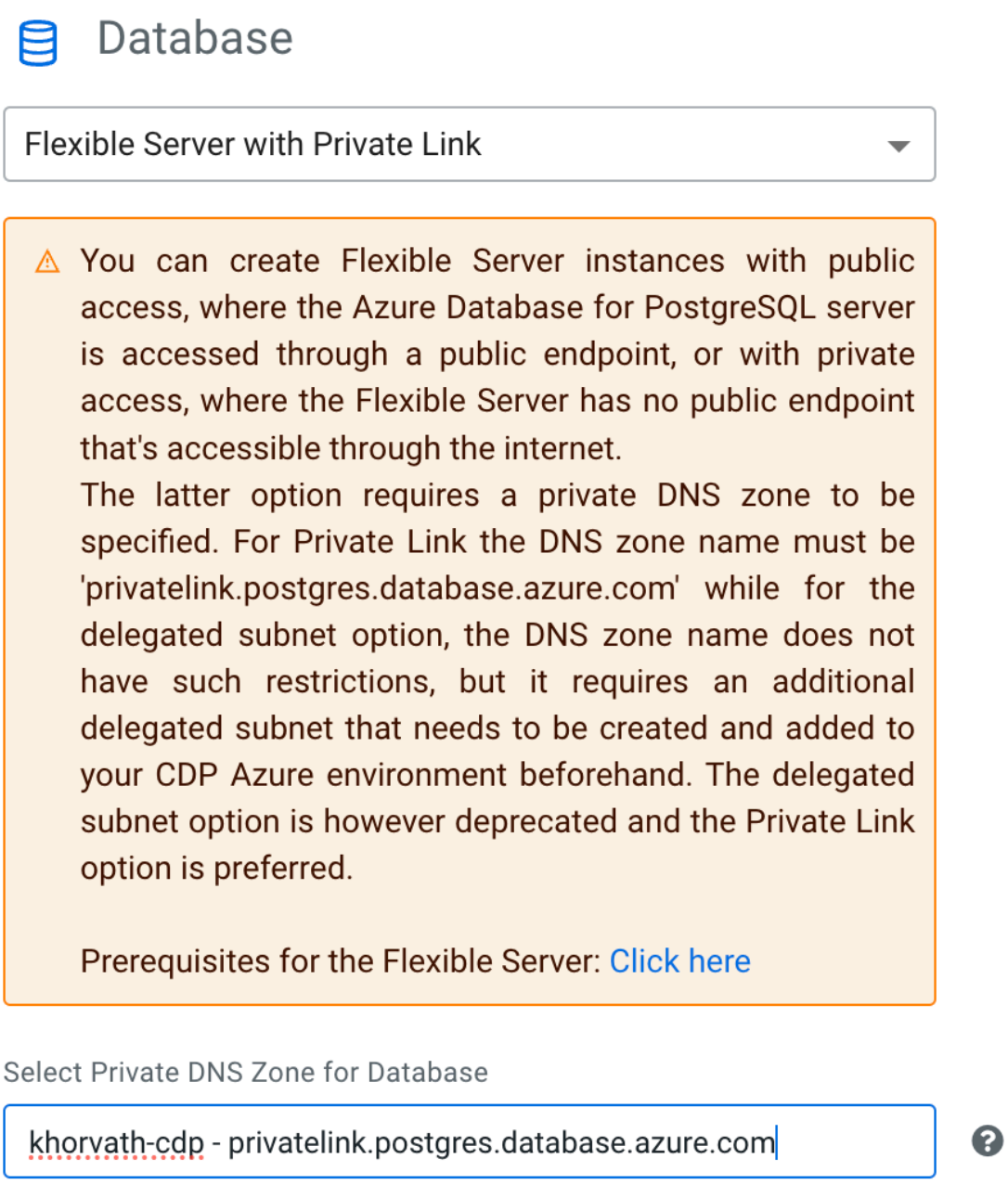
Steps

For Cloudera UI

Flexible Server with Private Link (recommended)

1. In the Database section, select Flexible Server with Private Link from the dropdown.

2. Select a private DNS zone for the private Flexible Server. If you do not select one, it will be created automatically. The private DNS zone must be named privatelink.postgres.database.azure.com.



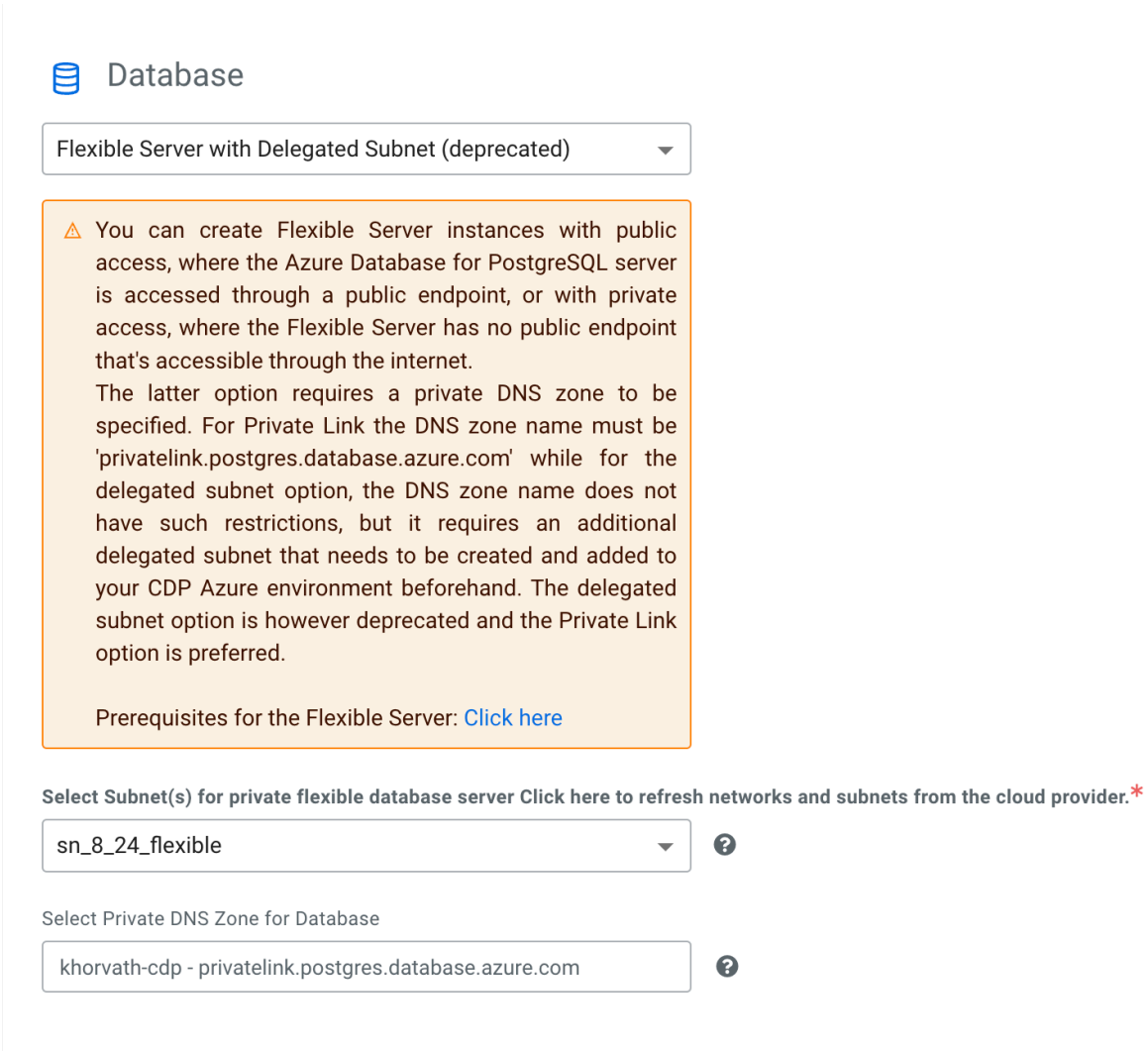
3. Finish registering your Azure environment in Cloudera.

Flexible Server with Delegated Subnet (deprecated)

1. In the Database section, select Flexible Server with Delegated Subnet from the dropdown.

2. Select a delegated subnet for the private Flexible Server.

   Note: Although you can currently select multiple subnets, the larger subnet is always used by Cloudera. That is, if there are two subnets provided, one with 128 available IPs and another with 256 available IPs, the second one will be picked. Even though you can select multiple delegated subnets, we recommend that you provide only one.
3. Select a private DNS zone for the Private Flexible Server. If you do not select one, it will be created automatically.



4. Finish registering your Azure environment in Cloudera.

For CDP CLI

1. Register an Azure environment using the cdp create-azure-environment CDP CLI command including --existing-network-params with a reference to the Private DNS Zone ID and, in case of Flexible Server with

Delegated Subnet, --flexible-server-subnet-ids with a reference to the delegated subnet ID. The virtual network link does not need to be specified as input.

Note: Although you can currently provide multiple subnets, the larger subnet is always used by Cloudera. That is, if there are two subnets provided, one with 128 available IPs and another with 256 available IPs, the second one will be picked. Even though you can select multiple delegated subnets, we recommend that you provide only one.

For example:

```
cdp environments create-azure-environment
  --environment-name <env-name>
...
  --existing-network-params networkId=dp-rg-test-vnet,resourceGroupName=
dp-rg,subnetIds=/subscriptions/3ddda1c7-d1f5-4e7b-ac81-0523f483b3b3/reso
urceGroups/dp-rg/providers/Microsoft.Network/virtualNetworks/dp-rg-vnet/
subnets/a,/subscriptions/3ddda1c7-d1f5-4e7b-ac81-0523f483b3b3/resourceGr
oups/dp-rg/providers/Microsoft.Network/virtualNetworks/dp-rg-vnet/subnet
s/2,default,databasePrivateDnsZoneId=<dns-zone-id>
  –-flexible-server-subnet-ids <delegated-subnet-id>
```

The following table explains the required parameters:

Parameter name	Description	Possible values
existing-network-params.databasePrivateDnsZoneId (string)	The ID of an existing private DNS zone used for the database.	Full resource reference
flexible-server-subnet-ids (array)	Required only for Flexible Server with Delegated Subnet option. If not specified, a Flexible Server with Private Link is launched. Comma separated list of the subnet names or full resource IDs delegated for flexible server. This can be specified in two formats: subnet1,subnet2 or /subscriptions/3ddda1c7-d1f5-4e7b-ac81-0523f483b3b3/resourceGroups/dp-rg/providers/Microsoft.Network/virtualNetworks/dp-rg-vnet/subnets/1,/subscriptions/3ddda1c7-d1f5-4e7b-ac81-0523f483b3b3/resourceGroups/dp-rg/providers/Microsoft.Network/virtualNetworks/dp-rg-vnet/subnets/2 The parameter takes a list of subnet IDs (or creates the IDs in case subnet names are provided), validates if they are indeed delegated, and takes the subnet with the largest CIDR range.	List of full resource reference(s). Although this parameter accepts a comma-separated list of subnets, the subnet with the largest CIDR range is always used. Therefore, you should provide only one subnet.

2. Set IDBroker mappings as usual using the cdp environments set-id-broker-mappings command.
3. Create a Data Lake as usual using the cdp create-azure-datalake CDP CLI command, including a reference to the database HA type and the database PostgreSQL engine version.

Related Information

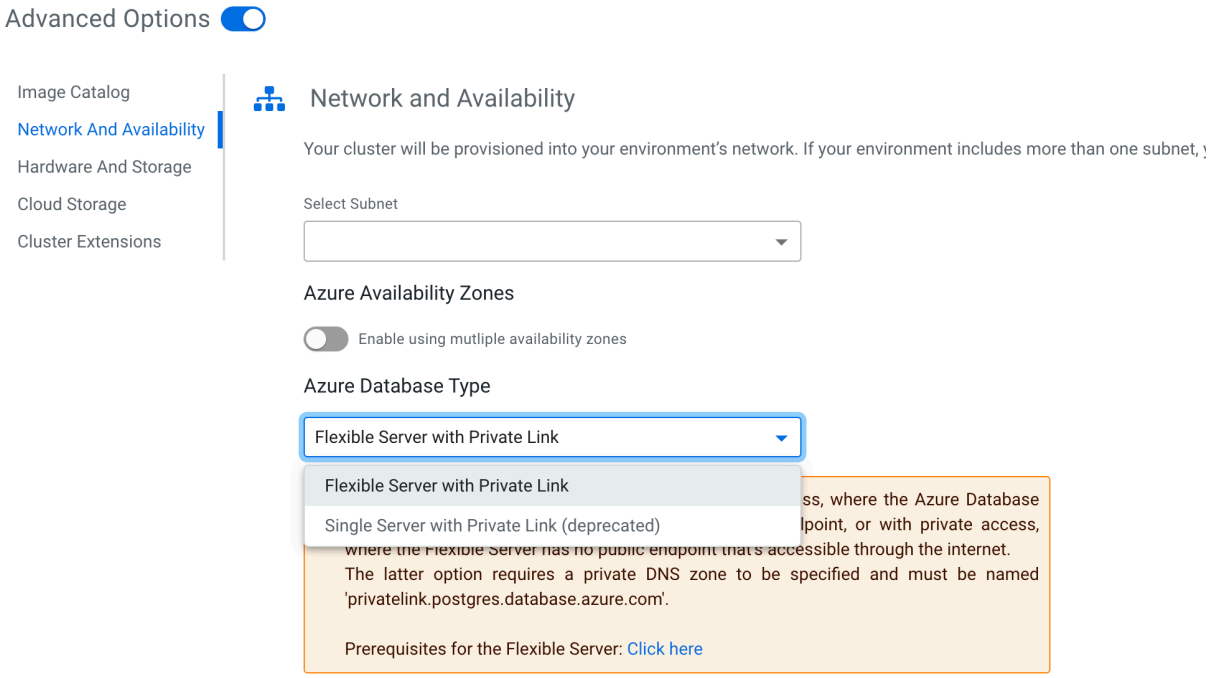
Registering an Azure environment from the Cloudera UI
Registering an Azure environment from the CDP CLI
Enabling admin and user access to environments

Enabling Flexible Server during Cloudera Data Hub cluster creation

A Cloudera Data Hub cluster uses the same Flexible Server or Single Server settings as the environment in which it runs, but you can choose to enable a Flexible Server on Cloudera Data Hub clusters running in an environment that uses a Single Server. This option can be specified during Cloudera Data Hub cluster creation in the Advanced Options under Advanced options Network and Availability Azure Database Type .

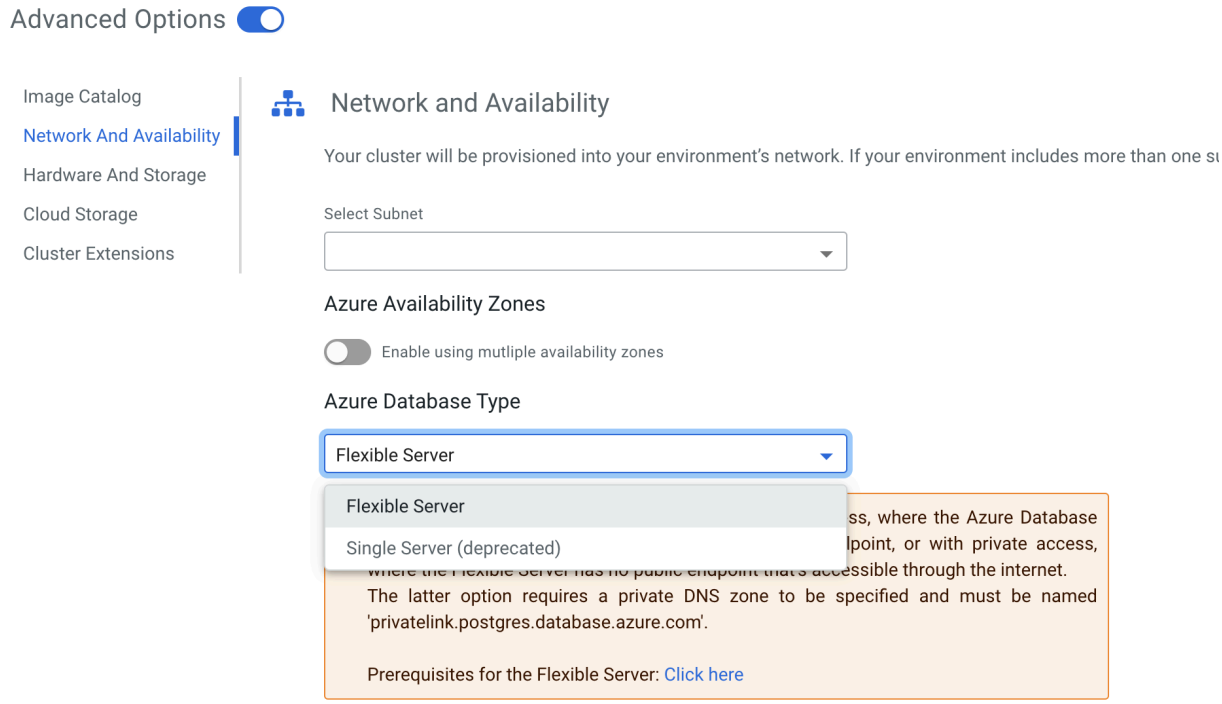
There are the following possible scenarios:

- If your environment uses Flexible Server with Private Link, and you have specified a private DNS zone on the environment level (either during environment creation or later by editing the environment), then from the Azure Database Type dropdown you can select to use Flexible Server with Private Link or Single Server with Private Link (deprecated). You cannot select the public Flexible Server or the public Single Server option, or Flexible Server with Delegated Subnet in this case. The Flexible Server with Private Link used for your Cloudera Data Hub cluster will use the private DNS zone specified on the environment level.
- If your environment uses Single Server with Private Link, then from the Azure Database Type dropdown you can select to use Flexible Server with Private Link or Single Server with Private Link (deprecated) during Cloudera Data Hub cluster creation.



- If your environment uses Private Flexible Server with Delegated Subnet, and you have specified a delegated subnet and a private DNS zone on the environment level (either during environment creation or later by editing the environment), then from the Azure Database Type dropdown you can select to use Flexible Server with Delegated Subnet (deprecated) or Single Server (deprecated) for your Cloudera Data Hub cluster. You cannot select to use the public Flexible Server in this case. If you choose to use the Flexible Server with Delegated Subnet option for your Data Hub, it will use the delegated subnet and private DNS zone specified on the environment level.
- If you choose to use the public Flexible Server option and do not specify a delegated subnet and a private DNS zone on the environment level, then from the Azure Database Type dropdown you can select to use Flexible Server or Single Server for your Cloudera Data Hub cluster. You cannot select to use either private Flexible Server options in this case.

- If you choose to use the public Single Server option on the environment level, then from the Azure Database Type dropdown you can select to use Flexible Server or Single Server for your Cloudera Data Hub cluster. You cannot select to use either private Flexible Server options in this case.



- If an environment is configured with Azure Single Server and Service Endpoints, by default new Cloudera Data Hub clusters provisioned in this environment will be created with Public Flexible Servers. If in this case you would still like to use a Single Server instead, then the --database-type=SINGLE_SERVER CLI parameter should be used, or you can achieve the same on the UI under the Advanced Data Hub options.

If you do not explicitly specify any Azure Database Server option, Cloudera follows the logic described in Private Flexible Server setup > Database Server Options.

Steps

For Cloudera UI

This option can be specified during Cloudera Data Hub cluster creation in the Advanced Options under Network and Availability > Select Azure Database Server.

For CDP CLI

Create a Cloudera Data Hub cluster as usual using the cdp create-azure-cluster CDP CLI command and specifying the —-database-type.

For example:

```
create-azure-cluster —-database-type FLEXIBLE_SERVER
```

The following table explains the required parameters:

Parameter	Description	Possible values

database-type (string)	The type of the azure database. FLEXIBLE_SERVER is the next generation managed PostgreSQL service in Azure that provides maximum flexibility over your database, built-in cost-optimizations. SINGLE_SERVER is a fully managed database service with minimal requirements for customizations of the database.	• FLEXIBLE_SERVER • SINGLE_SERVER

Related Information

Network and availability

Enabling Flexible Server with Private Link on an existing environment

You can change an existing environment that uses either a public Flexible Server or Flexible Server with Delegated Subnet to launch the new Data Hubs with a Flexible Server with Private Link. You can achieve this by editing the Network settings of your environment.

Note: Once you have enabled Flexible Server with Private Link by performing the following steps, you can no longer change the Azure Database Server type back to any other as this is the most advanced database server type.

Prerequisites

See Azure prerequisites for Flexible Server.

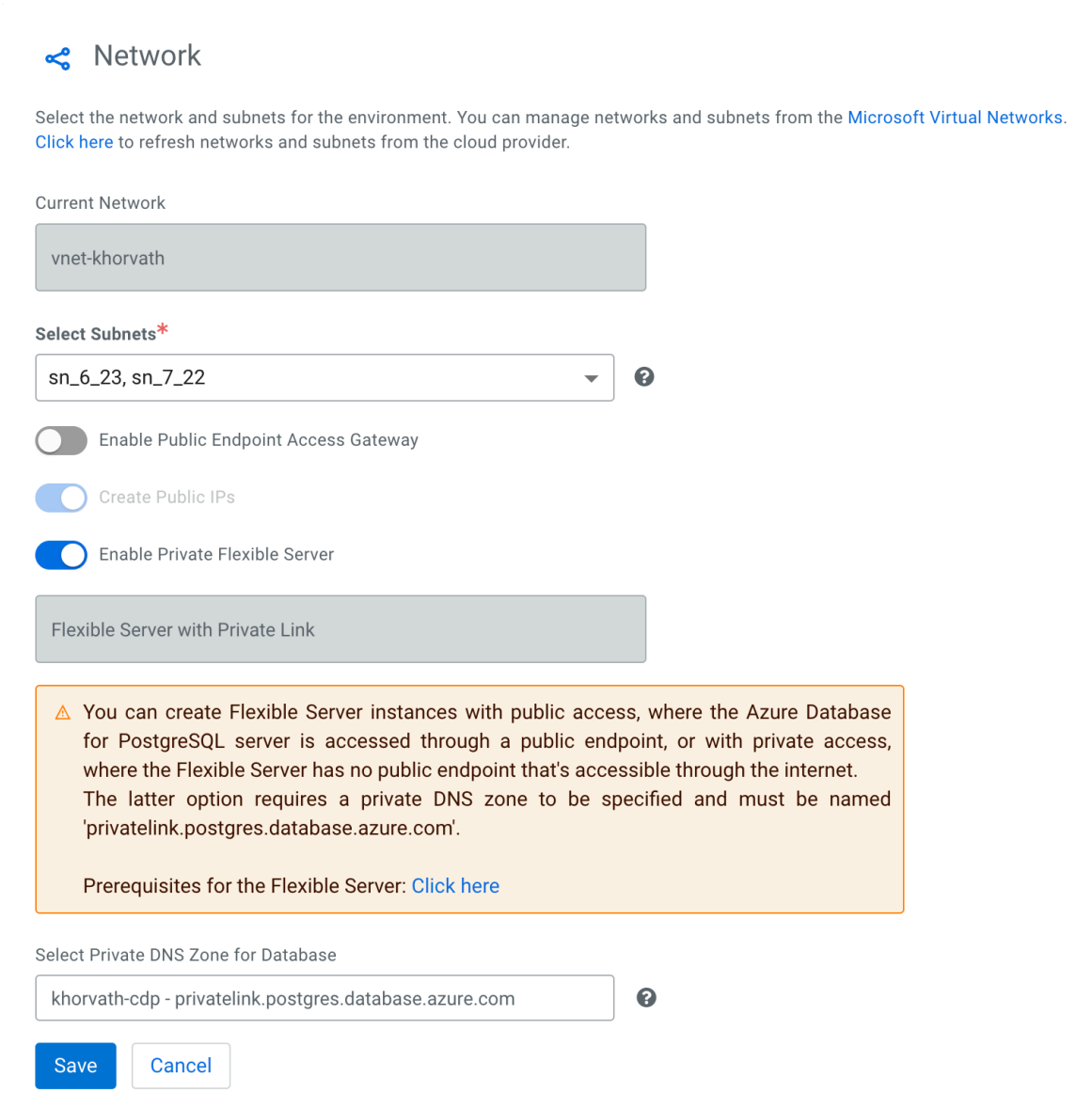
Steps

For Cloudera UI

1. In the Cloudera Data Hub cluster, navigate to Environments and then navigate to a specific environment.
2. Click the Summary tab.
3. Scroll down to Network and click the (edit) button.
4. Click the toggle button next to Enable Private Flexible Server to enable the feature.
5. Select Flexible Server with Private Link.

6. Select a Private DNS Zone for the Private Flexible Server. This is required.

Note: If you are editing an existing environment to enable a Flexible Server on it, you must provide a Private DNS Zone (unlike during environment creation where Cloudera cluster can create it for you).



7. Click Save.

For CDP CLI

Use the following command:

```
cdp environments update-azure-database-resources
 --environment <ENVIRONMENT_NAME_OR_CRN>
 --database-private-dns-zone-id <SPECIFY-DNS-ZONE-ID>
```

For example:

```
cdp environments update-azure-database-resources
```

```
  --environment eng-ml-dev-env-azure
  --database-private-dns-zone-id /subscriptions/1e0c1142-6256-61b9-
b80a-c5888d6e1e22/resourceGroups/b3x-at50645-weu-bde/providers/
Microsoft.Network/privateDnsZones/b3xeng.postgres.database.azure.com
```

Upgrading Azure Single Server to Flexible Server

PostgreSQL version 11 databases used by Data Lakes and Cloudera Data Hub clusters need to be upgraded to PostgreSQL 14. During this process, the Azure Single Server database will be upgraded to Azure Flexible Server. If there is a Private DNS Zone added to the environment, a Private Link will be created for the database and the Flexible Server will be private. Otherwise, a public Flexible Server will be created.

This process describes how to upgrade a database used by a Data Lake or Cloudera Data Hub clusters to Azure Flexible Server. For a general description of Data Lake and Cloudera Data Hub database upgrades, see Upgrading Data Lake/Data Hub database.

During the upgrade process, a backup is created from the Azure Single Server database, the server is deleted, and a new Azure Flexible Server is created with the same name. This way there is no need to change hostnames. However, the username format has changed between Single Server and Flexible Server. For details, see the Comparison table in the Microsoft Azure documentation.

Feature / Capability	Azure Database for PostgreSQL Single Server	Azure Database for PostgreSQL Flexible Server
Username in connection string	<user_name>@server_name. For example, pgadmusr@mypgServer.	Just username. For example, pgadmusr.

This change is automatically handled by Cloudera, however, as a result, Cloudera Runtime services using the database will be reconfigured during the upgrade process to use the new username format. Any third-party applications using the database would need to reflect this change by updating the username.

Limitations

- If you are using a public Single Server and want to upgrade to a private Flexible Server, you need to add a Private DNS Zone to your environment first. For instructions on adding a Private DNS Zone, see Private DNS Zone.
- If you would like to upgrade from Azure Single Server to a private Flexible Server, the upgrade to Flexible Server with delegated subnet is not available, as the Flexible Server with delegated subnet option is deprecated. It is recommended to use Flexible Server with Private Link instead.
- If you are already using a private Flexible Server with a delegated subnet, it is not yet supported to migrate to a private Flexible Server with Private Link.
- If you are performing a Data Lake resize because you want to upgrade to Cloudera Runtime 7.2.18 or later version, you must perform the upgrade of the database first.

Prerequisites

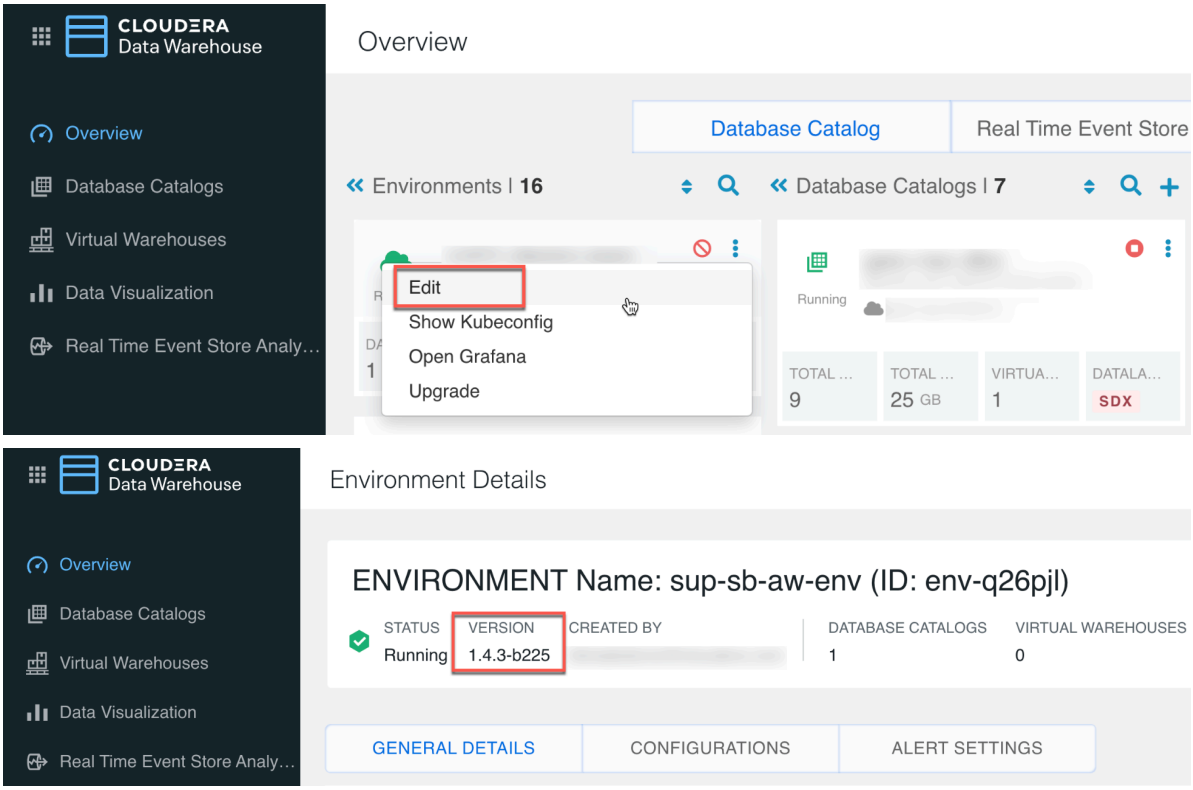
- When you are upgrading from Azure Single Server to Azure Flexible Server, new permissions are required. These permissions are being validated during the first phase of the upgrade. For a detailed list of prerequisites for Azure Flexible Server, see Azure prerequisites for Flexible Server.

  Important: There is a permission which is not required for enabling Flexible Server on a new environment but is specific to upgrading to Flexible Server from Single Server. Check the Azure prerequisites for Flexible Server even if you have enabled Flexible Server on your environment earlier.

  If you have missed adding any of the required permissions and the upgrade to Flexible Server fails, you need to fix the permission issues on Azure side and then perform a retry through the Cloudera Management Console.

- If you are upgrading from a public Single Server, make sure that port 5432 is open for outbound traffic. Otherwise the database upgrade might fail with a connection error. Public Azure Single Server access works with Service Endpoints, but these are not available for Flexible Server.
- If you are using Cloudera Data Warehouse, perform the following steps:
  - Upgrade to Cloudera Data Warehouse version 1.4.1 or higher before you can upgrade to Azure Flexible Server. Determine the Cloudera Data Warehouse version you are on by clicking edit on the environment:



  - Stop the virtual warehouses and data catalogs associated with the environment.
  - The validation process will not proceed until all Data Hubs are stopped.
- If you are using a Customer Managed Key (CMK) for data encryption, a managed identity is required to be able to upgrade from Azure Database for PostgreSQL Single Server to Flexible Server.

  For details on managed identity, see Managed identity for encrypting Azure Database for PostgreSQL Flexible Server.

  For information on how to add a managed identity to an environment already configured with CMK, see Updating an existing environment to add the managed identity Flexible Server.

Upgrading databases using the Cloudera UI

It is possible to upgrade from Azure Single Server to Azure Flexible Server using the Cloudera UI. The process can differ depending on PostgreSQL version that you are upgrading from, and depending on whether you are upgrading from private or public Single Server.

Note:

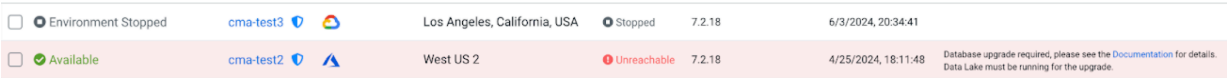
Because COD clusters are Data Hub clusters on the Control panel side, you can use the same Data Hub rules/process.

Upgrading a PostgreSQL 11 public Single Server to a PostgreSQL 14 public Flexible Server

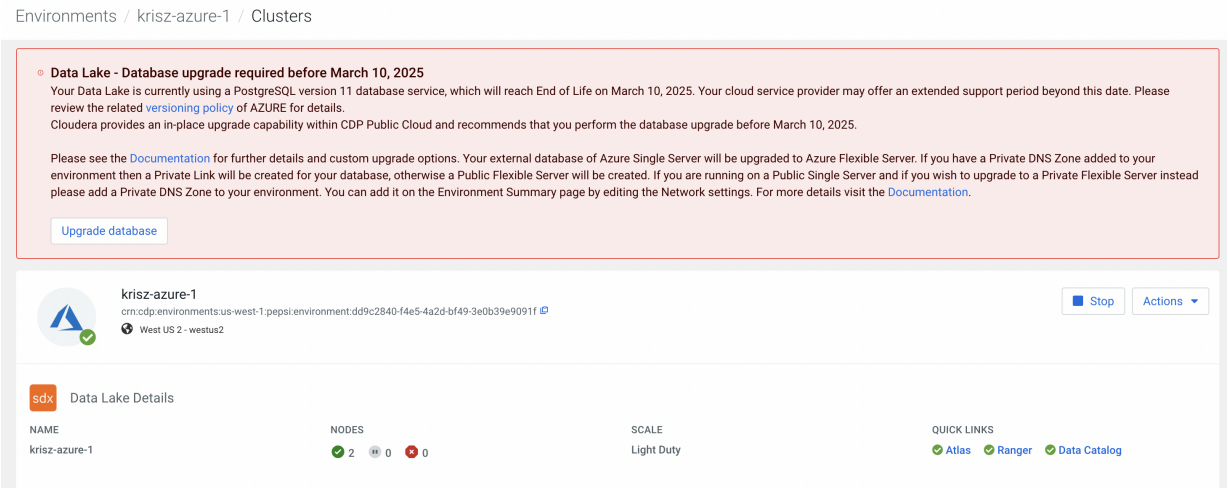
Before you begin, make sure the prerequisites listed in Upgrading Azure Single Server to Flexible Server are met.

1. In the Cloudera Management Console, navigate to Environments and then navigate to a specific environment.

   Environments that require an upgrade are highlighted with a red background.



2. Click the Upgrade database button inside the warning message at the top of the screen.



3. Once the Data Lake database is updated, check for the Cloudera Data Hub clusters of that Data Lake, if there is any database upgrade notification and perform the database upgrade as described above.

   Note: The database upgrade needs to be performed on every Data Lake and Cloudera Data Hub cluster separately, one by one.

Upgrading a PostgreSQL 10 or PostgreSQL 11 private Single Server to PostgreSQL 14 private Flexible Server with Private Link

Before you begin, make sure the prerequisites listed in Upgrading Azure Single Server to Flexible Server are met.

1. In the Cloudera Management Console, navigate to Environments and then navigate to a specific environment.
2. Click the Upgrade database button inside the warning message at the top of the screen.
3. Once the Data Lake database is updated, check for the Cloudera Data Hub clusters of that Data Lake, if there is any database upgrade notification and perform the database upgrade as described above.

   Note: The database upgrade needs to be performed on every Data Lake and Cloudera Data Hub cluster separately, one by one.

Upgrading a PostgreSQL 11 public Single Server to PostgreSQL 14 private Flexible Server with Private Link

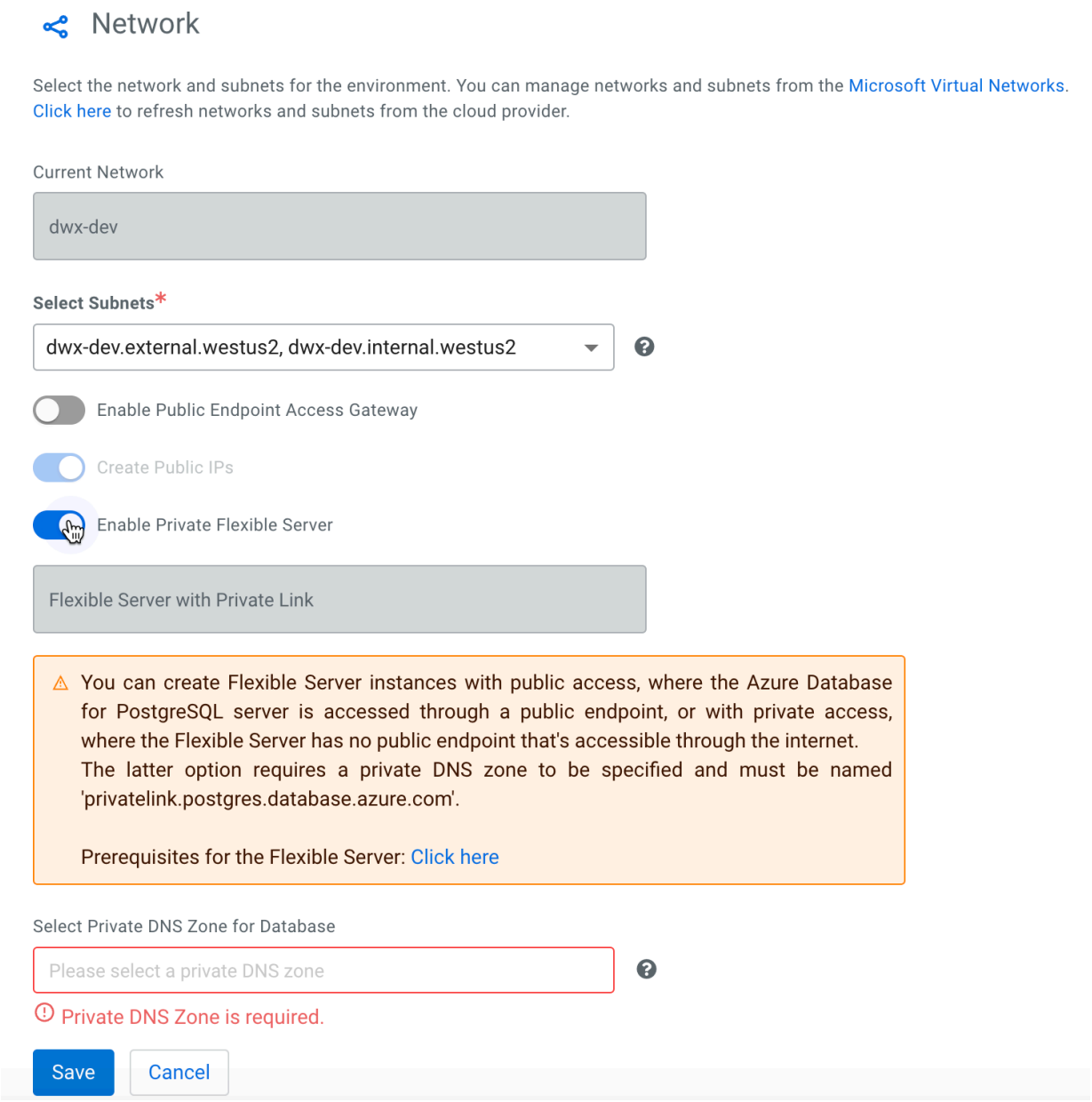
Before you begin, make sure the prerequisites listed in Upgrading Azure Single Server to Flexible Server are met.

If you would like to upgrade from a public to a Private Flexible Server, you need to enable Private Flexible Server first, and then upgrade.

1. In the Cloudera Management Console, navigate to Environments.

   Environments that require a PostgreSQL database upgrade are listed with a red background.
2. Select a specific environment.
3. Click the Summary tab.
4. Scroll down to Network and click the (edit) button.

5. Click the toggle button next to Enable Private Flexible Server.



6. Select a Private DNS Zone.

   The name of the Private DNS Zone must be privatelink.postgres.database.azure.com. You can only select from already existing Private DNS Zones. The Create new Private DNS Zone option is not available here.
7. Click Save.

   The Private Endpoint becomes enabled and the Private DNS Zone is displayed on the Network card of the Summary tab.

   Note: This does not affect the existing clusters of the environment, only the newly created clusters will utilize the Private Endpoint.
8. Click the Upgrade database button inside the warning message at the top of the screen.

9. Once the Data Lake database is updated, check for the Cloudera Data Hub clusters for that Data Lake, if there is any database upgrade notification and perform the database upgrade as described above.

Note: The database upgrade needs to be performed on every Data Lake and Cloudera Data Hub cluster separately, one by one.

Upgrading databases using the CDP CLI

It is possible to upgrade from Azure Single Server to Azure Flexible Server using the CDP CLI. You can use the regular Cloudera Data Hub and Data Lake database upgrade commands to do this. The upgade to PostgreSQL 14 automatically upgrades the database from Single Server to Flexible Server.

Data Lake database upgrade

Before you begin, make sure the prerequisites listed in Upgrading Azure Single Server to Flexible Server are met.

You can upgrade Data Lake database using the cdp datalake start-database-upgrade CLI command.

The --target-version parameter is optional. If you do not provide it, the database will be upgraded to PostgreSQL 14.

```
cdp datalake start-database-upgrade --help --form-factor public
NAME
       start-database-upgrade  -  Upgrades the database of the Data Lake cl
uster.

DESCRIPTION
       This command initiates the upgrade of the database  of  the  Data  La
ke cluster.
SYNOPSIS
            start-database-upgrade
          --datalake <value>
          [--target-version <value>]
          [--cli-input-json <value>]
          [--generate-cli-skeleton]
OPTIONS
       --datalake (string)
          The name or CRN of the Data Lake.
       --target-version (string)
          The database engine major version to upgrade to.

          Possible values:

          o VERSION_14
```

Cloudera Data Hub database upgrade

Before you begin, make sure the prerequisites listed in the *Prerequisites* section of Upgrading Azure Single Server to Flexible Server are met.

You can perform Cloudera Data Hub database upgrade using the cdp datahub start-database-upgrade CLI command.

The --target-version parameter is optional. If you do not provide it, the database will be upgraded to PostgreSQL 14

```
cdp datahub start-database-upgrade --help --form-factor public
NAME
       start-datahub-upgrade  -  Upgrades the database of the Data Hub clust
er.
DESCRIPTION
       This command initiates the upgrade of the database  of  the  Data  H
ub cluster.
SYNOPSIS
```

```
            start-database-upgrade
          --datahub <value>
          [--target-version <value>]
          [--cli-input-json <value>]
          [--generate-cli-skeleton]

OPTIONS
       --datahub (string)
          The name or CRN of the Data Hub.

       --target-version (string)
          The database engine major version to upgrade to.

          Possible values:

          o VERSION_14
```

Post-requisites after upgrading databases

Complete the following tasks after upgrading from Azure Single Server to Azure Flexible Server.

- Start the Cloudera Data Warehouse Database Catalogs and Virtual Warehouses. For each virtual warehouse, Cloudera recommends that you start, stop, and start again. This will completely refresh the Data Lake details for the virtual warehouse.

  Warning: You must be aware of an issue where the Database Catalog goes into an error state after upgrading to Azure Fexible Server. For more information about this known issue, see the *Cloudera Data Warehouse 1.9.6-b2 (released March 11, 2024) release notes*.

Related Information

Cloudera Data Warehouse Release Notes

Configuring a CMK for data encryption in Azure Database for PostgreSQL Flexible Server

You can optionally use a Customer Managed Key (CMK) for encrypting data in the Azure Flexible Server database instance used by Cloudera.

As described in Adding a customer managed encryption key to a Cloudera environment running on Azure, by default Cloudera clusters are encrypted with server-side encryption (SSE) using Platform Managed Keys (PMK) and Cloudera allows you to provide an existing CMK for encrypting Cloudera clusters.

When using the CMK for encrypting Cloudera clusters, you can also use that same CMK for encrypting the Azure Flexible Server database instance used by Cloudera. If you would like to do this, in addition to the typical CMK prerequisites, you should create a managed identity with specific permissions and then after providing the other CMK-related parameters (CMK resource group and URL) during Cloudera environment registration on Azure provide that managed identity during Azure environment registration in Cloudera.

Azure prerequisites

You should first meet the CMK-related prerequisites described in Azure Requirements: Customer managed encryption keys (add additional credential permissions and create a key vault and vault key).

In addition to that, you should create a managed identity as described in Managed identity for encrypting Azure Database for PostgreSQL Flexible Server.

Creating an environment with a CMK for encrypting Flexible Server

Steps

For Cloudera UI

Follow the usual steps for creating a Cloudera environment on Azure and make sure to do the following:

1. In the Register Environment wizard, on the Region, Networking and Security page find the Customer-Managed Keys section.
2. Click Enable Customer-Managed Keys.
3. In the same section, under Select Resource group select the resource group where your CMK is located.
4. Provide the URL of the key value where the CMK resides. This is the same as the key identifier that you can copy directly from Azure Portal.
5. Under Managed identity for encryption, select the managed identity created as a prerequisite.

For CDP CLI

Add the following CDP CLI parameters to the cdp environments create-azure-environment command:

```
--encryption-key-resource-group-name <CMK_RESOUCE_GROUP_NAME>
--encryption-key-url <KEY_RESOUCE_ID>
--encryption-user-managed-identity <EXISTING_MANAGED_IDENTITY>
```

Note: The former --user-managed-identity parameter has been deprecated. Use the --encryption-user-managed-identity instead, as shown above.

The --encryption-key-resource-group-name is not needed in some cases, as described in CDP CLI steps listed in Adding a customer managed encryption key to a Cloudera environment running on Azure.

While the first two parameters are required for using CMK for Cloudera in general the third parameter must be added in addition for encrypting a Flexible Server. It should be specified as in the following example:

```
--encryption-user-managed-identity /subscriptions/3ddda1c7-d1f5-4e4b-ac81-0523f483b3b1/resourcegroups/test-daily-rg/providers/Microsoft.ManagedIdentity/userAssignedIdentities/test-adminIdentity
```

Updating an existing environment to add the managed identity

Steps

For Cloudera UI

The steps for adding the managed identity are similar as those described in Set a CMK for an existing Azure environment, just in the Managed identity for encryption field, a managed identity should be provided in addition to the Encryption Key Resource Group and the Encryption Key URL.

The managed identity can be either added when you are adding the CMK, or can be added to an environment already configure with CMK. In the latter case, it is enough to populate the Managed identity for encryption field, the other fields are already populated.

For CDP CLI

The steps for adding the managed identity are similar as those described in Set a CMK for an existing Azure environment, just the --encryption-user-managed-identity should be specified in addition.

The managed identity can be either added when you are adding the CMK, or can be added to an already existing CMK setup. In both cases, use the full command with all the listed parameters.

Use the following CDP CLI command:

```
cdp environments update-azure-encryption-resources \
 --environment <ENVIRONMENT_NAME> \
 --encryption-key-url <VAUL_KEY_URL> \
```

```
  --encryption-key-resource-group-name <RG_NAME> \
  --encryption-user-managed-identity <EXISTING_MANAGED_IDENTITY>
```

Enabling Single Server

During environment registration in Cloudera, the Flexible Server in public service mode is used by default, but you can specify to use the Single Server (deprecated) or Single Server with Private Link (deprecated) options.

Steps

For Cloudera UI

During environment registration in Cloudera, in the Database section, enable one of the Single Server options by selecting Single Server (deprecated) or Single Server with Private Link (deprecated) from the dropdown. The Flexible Server option is pre-selected by default.

For CDP CLI

1. Register an Azure environment using the cdp create-azure-environment CDP CLI command as usual.
2. Set IDBroker mappings as usual using the cdp environments set-id-broker-mappings command.
3. Create a Data Lake using the cdp create-azure-datalake CDP CLI command, including a reference to the database HA type. For example:

```
cdp datalake create-azure-datalake
  --datalake-name <datalake-name>
...
  --database-type SINGLE_SERVER
```

The following table explains the required parameters:

Parameter	Description	Possible values
database-type (string)	The type of the azure database. FLEXIBLE_SERVER (recommended) is the next generation managed PostgreSQL service in Azure that provides maximum flexibility over your database, built-in cost-optimizations. SINGLE_SERVER is a fully managed database service with minimal requirements for customizations of the database.	• FLEXIBLE_SERVER • SINGLE_SERVER

4. Create a Cloudera Data Hub using the create-azure-cluster CDP CLI command, including a reference to the database HA type and the database PostgreSQL engine version. For example:

```
cdp datahub create-azure-cluster
 --cluster-name <marketplace-data-hub>
...
 --datahub-database HA
 --database-type SINGLE_SERVER
```

The following table explains the required parameters:

Parameter	Description	Possible values
datahub-database (string)	Represents the database availability type.	• HA (This means, same zone HA) • NON_HA

database-type (string)	The type of the azure database. FLEXIBLE_SERVER (default value) is the next generation managed PostgreSQL service in Azure that provides maximum flexibility over your database, built-in cost-optimizations. SINGLE_SERVER is a fully managed database service with minimal requirements for customizations of the database. If you do not specify this parameter, FLEXIBLE_SERVER is used by default.	• FLEXIBLE_SERVER • SINGLE_SERVER

Troubleshooting Flexible Server

Refer to this documentation for troubleshooting Azure environments using Flexible Server.

Auto-migrated PostgreSQL 11 Single to Flexible Server upgrade failure to PostgreSQL 14

Problem:

The following warning error message is displayed during the upgrade to PostgreSQL 14 Azure Flexible Server when trying to upgrade a database that was auto-migrated by Microsoft from PostgreSQL 11 Single Server to PostgreSQL 11 Flexible Server:

```
The major version upgrade failed precheck. Upgrading with password authentic
ation mode enabled is not allowed from source version MajorVersion11. Please
 enable SCRAM and reset the passwords prior to retrying the upgrade.
```

Solution:

Perform the following steps. The process does not cause any downtime.

1. Find the azure.accepted_password_auth_method server parameter from Settings Server parameters on the Azure portal.
2. Enable SHA-256 authentication by selecting SCRAM-SHA-256.
3. Save your settings at the top of the page.
4. Retry the upgrade in the Cloudera Control Plane.

Capacity limitation related warning during Flexible Server upgrade

Problem:

A capacity related validation warning is shown during the upgrade from Azure Single Server to Flexible Server.

Solution:

Perform the following steps:

1. File an Azure Support Ticket, specifying the Azure subscription and region that is used for the Cloudera setup. Microsoft Azure support will ensure that the necessary capacity is allocated to unblock provisioning Flexible Server same-zone HA database instances.
2. Once the capacity has been increased, retry the Single Server to Flexible Server upgrade operation.

Missing default outbound access

Problem:

Data Lake provisioning failed due to missing outbound access.

Solution:

By default, Single Server utilizes service endpoints (Azure Database for PostgreSQL server - Microsoft.Sql) that provide secure and direct connectivity to DB service over an optimized route over the Azure backbone network from the Virtual Machines, but that is not the case with Flexible Server. This means that there are certain cases when you have to explicitly ensure that outbound (i.e. egress) network connectivity from the selected Virtual Network is set up using NAT-Gateway, UDR, or similar. If you are using a public Flexible Server, make sure that port 5432 is open for outbound traffic.

This requirement is applicable in the following conditions:

- An existing network is being used.
- Create Public IP is disabled.
- Public Endpoint Gateway is disabled.

The lack of default outbound access can be mitigated by using private setup or using public IPs instead.

For more information, see Default outbound access in Azure.

Enabling a private endpoint for Azure Postgres

By default Cloudera uses service endpoints, but you can select to use private endpoints instead. During environment registration you can optionally select the “Create Private Endpoint” option to use private endpoints instead of using a service endpoint. Currently, only one service or private endpoint is used, for Azure Postgres.

The endpoint type - either service or private endpoints - and whether to use your own private DNS zones or have Cloudera create them are decided at environment creation. All services capable of that endpoint type will use the selected endpoint type - this currently means only the Postgres server.

Note:

The endpoint type cannot be changed after the environment was created.

By default service endpoints are used, and in order to use private endpoints the user has to explicitly enable the “Create Private Endpoints“ option from either the Cloudera UI or CDP CLI. There is no option to not use either. Once you have turned on private endpoints then you can specify your own private DNS zone if you wish.

Prerequisites

You should also decide if you would like to use your own Azure private DNS zone or have Cloudera create an Azure private DNS zone for you, and meet the related prerequisites. Review the requirements described in Private endpoint for Azure Postgres.

Enabling private endpoints

You can enable private endpoints and select to use a Cloudera-managed Azure private DNS zone or your own Azure private DNS zone during Azure environment registration in the Cloudera Management Console.

Required role: EnvironmentCreator

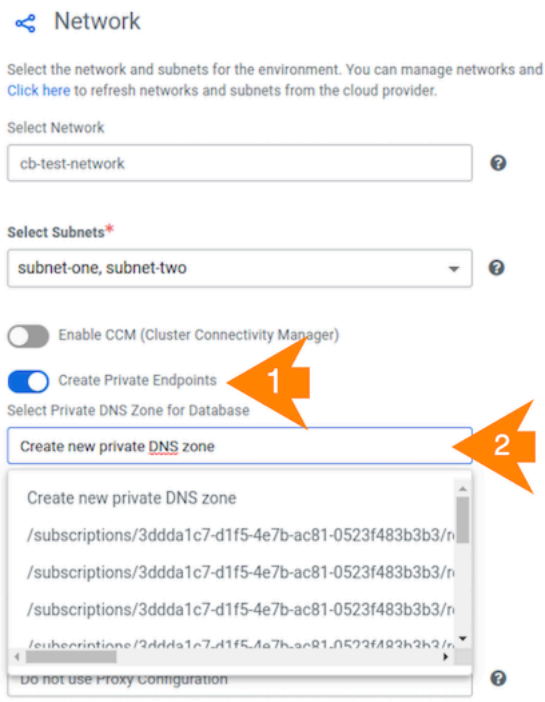
Steps

For Cloudera UI

When registering an environment specify the following:

1. In the Resource group section, select to use your pre-created resource group.
2. In the Network section of the register environment wizard, enable the "Create Private Endpoints" option.

3. A new drop down option appears. You have two options:
   a. Select an existing private DNS zone.
   b. If you would like Cloudera to create the private DNS zone, select “Create new private DNS zone”.



4. Provide other Azure environment parameters as usual and complete environment registration.

For CDP CLI

The parameter createPrivateEndpoints defines that Azure Postgres will be configured with a Private Endpoint and a Private DNS Zone. When this option is disabled, Azure Service Endpoints are created. This option is disabled by default, so Azure Service Endpoints are used by default.

This is how to specify this via CDP CLI when running the cdp environments create-azure-environment command:

```
--create-private-endpoints
--no-create-private-endpoints
```

If you would like to specify your own Azure private DNS zone, you should additionally include the databasePrivateDnsZoneId as part of existing-network-params:

```
--existing-network-params networkId=<NETWORK_ID>,resourc
eGroupName=<NETWORK_RG_NAME>,subnetIds=<SUBNET_1>,<SUBNE
T_2>,databasePrivateDnsZoneId=<PRIVATE_DNS_ZONE_RESOURCE_ID>
```

The <PRIVATE_DNS_ZONE_RESOURCE_ID> that you need to provide should look similar to:

```
/subscriptions/<SUBSCRIPTION_ID>/resourceGroups/<RESOURCE_GROUP_NAME>/pr
oviders/Microsoft.Network/privateDnsZones/privatelink.postgres.database.
azure.com
```

For example:

```
/subscriptions/a9a10f9c-5323-4fd2-8s14-747b2d68784c/resourceGroups/my-
resource-group/providers/Microsoft.Network/privateDnsZones/contoso.com
```

Related Information

Registering an Azure environment from the Cloudera UI

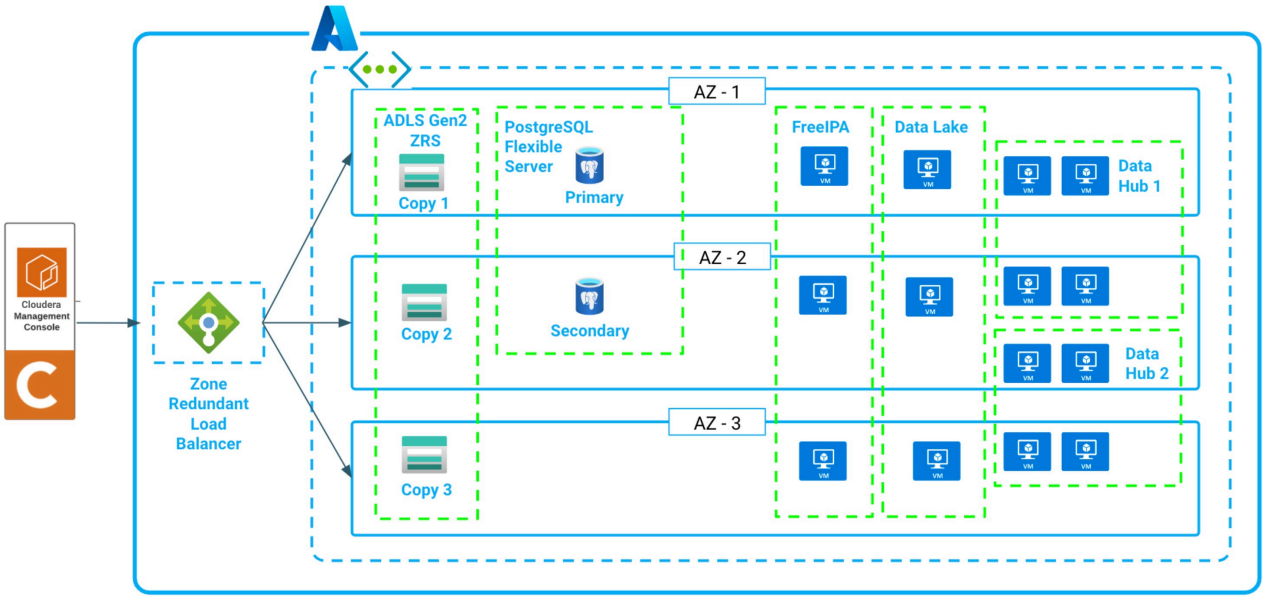
Registering an Azure environment from the CDP CLI

Deploying Cloudera in multiple Azure availability zones

You can optionally choose to deploy Data Lake, FreeIPA, and Cloudera Data Hub clusters across multiple availability zones (multi-AZ). With multi-AZ support, newly created Azure environments, enterprise Data Lakes and Cloudera Data Hub clusters using HA templates can be deployed across multiple availability zones of the selected Azure region. This provides fault tolerance during the extreme event of an availability zone outage.

Note: Only Enterprise Data Lakes can use multi-AZ; other Data Lake shapes do not support it. Enterprise Data Lake is only available in Cloudera Runtime 7.2.17 and newer; therefore, multi-AZ for Cloudera on Azure is only available in Cloudera Runtime 7.2.17 and newer.

Each Azure region has multiple availability zones, which act as failure domains, preventing small outages from affecting entire regions. If you choose to deploy your Cloudera environment (FreeIPA and Data Lake) and Cloudera Data Hub clusters across multiple availability zones, each of these components is spread across three availability zones, providing high availability and fault tolerance. This is illustrated in the following diagram:



With the multi-AZ option enabled, your services are deployed in the following way:

- Azure environments are always created with three FreeIPA servers, deployed on virtual machines spread across three available zones.
- In an Azure Enterprise Data Lake each host group is configured so that virtual machines of all critical services are spread across three available zones.
- In HA Cloudera Data Hub clusters, virtual machines of each host group are evenly spread across three availability zones, following a round-robin logic.

When a zone 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.

By default, If you do not enable multi-AZ, Cloudera and Cloudera customers do not have visibility into how Azure distributes VMs across availability zones, because the Azure Portal or CLI do not provide this information.

When creating Cloudera Data Hub clusters via CDP CLI, you have the option to specify the Availability Zone, which, in addition to allowing you to select the Availability Zones that should be used, allows you to set up Availability Zone targeting, where all nodes of the cluster are placed on the same Availability Zone. This enables creating disaster recovery scenarios, where a primary and secondary cluster are running in different Availability Zones. If an Availability Zone outage occurs and the primary cluster is lost, it is guaranteed that the secondary cluster is not impacted.

Use cases

A multi-AZ Data Lake and FreeIPA constitute a resilient environment that provides a solid basis for multi-AZ Cloudera Data Hub clusters and Cloudera data services. Cloudera Data Hub clusters 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, a medium duty or 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 an availability zone outage.

When an entire availability zone fails, HBase 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 Availability Zone is down, you cannot create a new Cloudera Data Hub cluster, and create or activate Cloudera data services within the environment. Existing workloads will continue to work.
- When an Availability Zone is down, you cannot resize, stop, or restart Cloudera Data Hub clusters.
- Non-AZ environments or clusters cannot be converted to multi-AZ.

Azure requirements

In order to use multi-AZ, you should meet the following Azure requirements:

1. The Azure region that you select should support setting up Azure PostgreSQL Flexible Server in Zone-Redundant HA mode and also the instance types to be used. See Flexible Server Azure Regions.
2. The ADLS Gen2 storage account should be created as zone-redundant storage (ZRS). To specify ZRS via Azure CLI during storage account creation, the --sku option should be set to Standard_ZRS. Below is a sample Azure CLI command:

```
azure % az storage account create \
 --name test-storage \
 --resource-group rg-test-rg \
 --access-tier Cool \
 --allow-blob-public-access false \
```

```
--allow-cross-tenant-replication false \
--allow-shared-key-access true \
--enable-hierarchical-namespace true \
--sku Standard_ZRS
```

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:
   a. Select to use the Enterprise Data Lake.
   b. On the same page, scroll down and in the bottom of the page enable the Advanced Options.
   c. 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 the Enterprise Data Lake is selected.
2. On the Region, Networking, and Security page:
   a. Scroll down and in the bottom of the page enable the Advanced Options.
   b. 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.
3. Finish registering your environment as usual.

For CDP CLI

Use the following CDP CLI commands to register an environment with a multi-AZ Data Lake and FreeIPA:

1. Register an Azure environment using thecdp environments create-azure-environment command and include multiAz=true in the--free-ipa parameter as shown in this example:

   ```
   cdp environments create-azure-environment \
   --environment-name test-env \
   ...
   --free-ipa instanceCountByGroup=3,multiAz=true \
   ```

   If you do not include the multiAz=true, the default Availability Zone distribution will be used.

   You can also optionally include the--availability-zonesparameter to select the specific availability zones that should be used. Valid values for availability zones are 1,2 and 3. If this parameter is not provided, all Availability Zones are used. For example:

   ```
   cdp environments create-azure-environment \
   --environment-name test-env \
   ...
   --free-ipa instanceCountByGroup=3,multiAz=true \
   --availability-zones 1 2
   ```

2. Set IDBroker mappings as usual using thecdp environments set-id-broker-mappingscommand.
3. Create a Data Lake using the cdp datalake create-azure-datalake command and adding the --multi-az parameter. For example:

   ```
   cdp datalake create-azure-datalake \
   --datalake-name test-dl \
   --environment-name test-env \
   ...
   ```

```
--scale ENTERPRISE \
--runtime 7.2.17 \
--multi-az
```

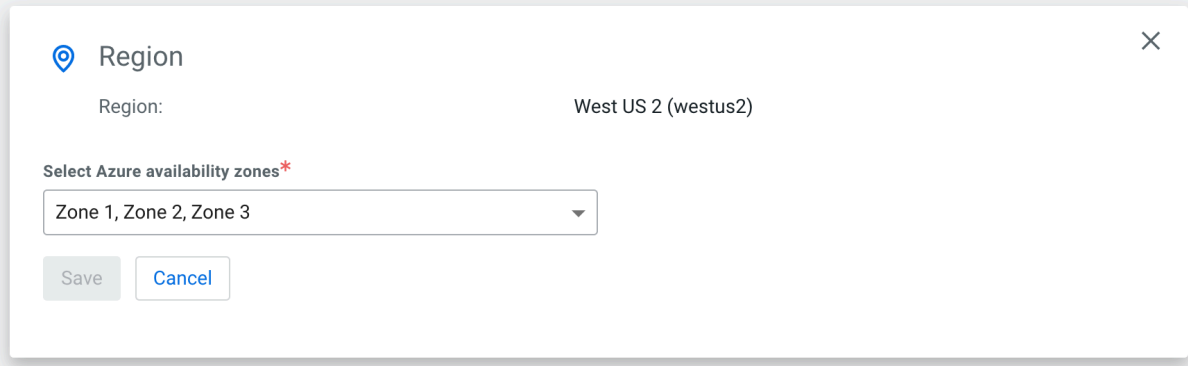
Modify environment's Availability Zones for new Cloudera Data Hub clusters

You can modify an environment's Availability Zones. In this case, the added Availability Zones will only be used for new Cloudera Data Hub clusters. The environment clusters and existing Cloudera Data Hub clusters running in it will continue to use the original Availability Zones.

Steps

For Cloudera UI

1. In the Cloudera Management Console, navigate to environment details > Summary.
2. Scroll down to the Region section.
3. Under Select Azure availability zones, select the availability zones:



4. Click Save.

For CDP CLI

Below is the CLI command for updating Availability Zones of an existing environment:

```
cdp environments update-azure-availability-zones \
--environment-name test-env \
--availability-zones12 3
```

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 Azure.

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

The security groups that you select to use during environment registration are only used for the Data Lake, FreeIPA, Cloudera Data Hub clusters, and Operational Databases running in that environment. The Kubernetes-based Cloudera services (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).

CDP service	Type of access that can be restricted	When and where to restrict	Link to related documentation
Cloudera Data Engineering	Admin access to Kubenetes endpoints can be restricted.	Restrict admin access to Kubernetes endpoints during enabling Data Engineering via the Whitelist IPs parameter.	Enabling Cloudera Data Engineering
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.	Editing environment details
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

Configure lifecycle management for logs on Azure

To avoid unnecessary costs related to ALS Gen2 cloud storage, you should create lifecycle management rules for your cloud storage container 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.

Azure allows you to set up lifecycle management rules for your ADLS cloud storage. For example, you can set a specified expiration period for a cloud storage location so that 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.

To set up lifecycle management:

- Review the prefixes listed below.
- Follow the instructions in Optimize costs by automating Azure Blob Storage access tiers.

Prefixes based on Azure environment's logs location base

Prior to creating lifecycle management rules, 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	**abfs://mycontainer@myaccount.dfs.core.windows.n**	**abfs://mycontainer/my-dl@myaccount.dfs.core.window**	**abfs://mycontainer/my-dl@myaccount.dfs.core.window**
FreeIPA prefix for lifecycle rule	mycontainer/cluster-logs/freeipa	mycontainer/my-dl/cluster-logs/freeipa	mycontainer/my-dl/freeipa
DataLake prefix for lifecycle rule	mycontainer/cluster-logs/datalake	mycontainer/my-dl/cluster-logs/datalake	mycontainer/my-dl/datalake
DataHub prefix for lifecycle rule	mycontainer/cluster-logs/datahub	mycontainer/my-dl/cluster-logs/datahub	mycontainer/my-dl/datahub

Related Information

Enabling environment telemetry

Migrating Azure Load Balancers and Public IP

Microsoft is planning to transition from Basic SKU to Standard SKU for Basic Load Balancers and Public IPs. Cloudera recommends to migrate to Standard SKUs as soon as possible.

As of today, the default SKU for "Microsoft.Network/publicIPAddresses" and "Microsoft.Network/loadBalancers" is Basic SKU, which takes effect when the SKU is not specified explicitly in the ARM template.

the following customer resources are currently created with Basic Public IP:

1. Non-multi-AZ FreeIPA instances with Public IP
2. Non-multi-AZ Data Lake and Cloudera Data Hub instances with Public IP, where Load Balancing is not explicitly defined as Standard SKU
3. Non-multi-AZ Basic SKU Load Balancers

Microsoft is planning to phase out the Basic SKU for both resources, according to the following timeline:

- March 31, 2025 - Last day to create new Basic Load Balancer and Public IP.
  - After this date, you will not be able to create new Basic Load Balancers and Public IPs. Those created before this date will continue to work.
  - The default SKU is transitioned to Standard SKU for any new Cloudera environments on Azure.

    The Standard SKU supports enhanced functionalities such as availability zones and a better security model.
  - Cloudera Data Hub and Data Lake creation requests explicitly specifying Basic SKU Load Balancer will be rejected.

- September 30, 2025 - Retirement Day of Basic Load Balancer and Public IP.
  - After this date, you will not be able to use Basic Load Balancers and Public IPs.
  - Clusters that have Basic SKU Public IP can become unreachable.
  - Knox and Oozie on clusters that have basic SKU Load Balancer can become unreachable.

Cloudera recommends to upgrade to Standard SKUs as soon as possible.

In the affected clusters, you will see a notification in Cloudera Management Console asking you to perform a migration to Standard SKU. To migrate from Basic SKU to Standard, you have the following options:

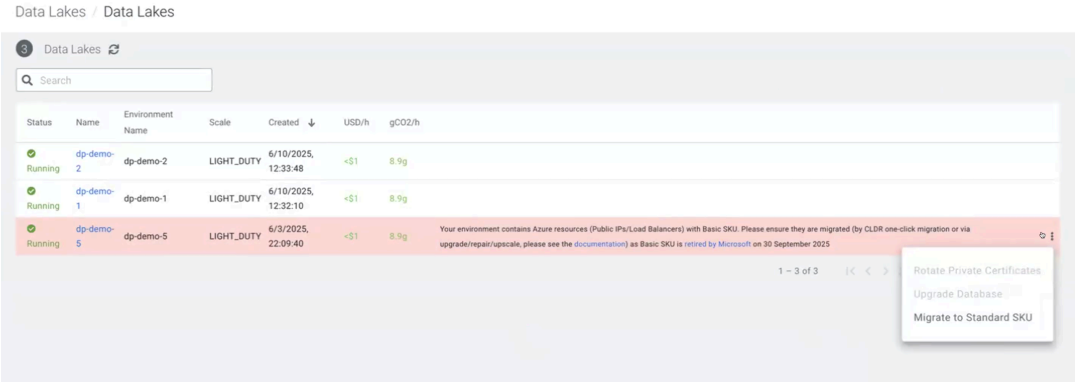
Automatic migration flow

You can complete the migration by performing an upgrade, repair, or upscale operation on your clusters with Basic SKU Load Balancers and Public IPs. During the upgrade, repair, or upscale operation, the Basic SKU Load Balancers and Public IPs will be migrated automatically to Standard SKU.

Note: While the operations are in progress, the cluster can be unreachable for a couple of minutes and client connections might need to be re-initiated.

On-demand one-click migration flow

You can complete the migration by navigating either to the **Environments**, **Data Lakes** or **Data Hubs** page in Cloudera Management Console, and select ⋮ where you see the warning message for the migration. Select Migrate to Standard SKU, and confirm the operation to start the migration.



Transitioning from Azure default outbound access

Microsoft is planning to retire the default outbound access connectivity for virtual machines in Azure. **You must transition your Cloudera environment on Azure to use an explicit method for outbound internet connectivity while creating any new virtual machines.** This ensures enhanced security, predictability, and reliability for your clusters.

In Microsoft Azure (Azure), virtual machines that are created in a virtual network without a defined explicit outbound method were assigned a default implicit outbound public IP address. This IP address enabled outbound connectivity from the resources to the internet. If you have deployed a virtual machine in Azure without explicit outbound connectivity, a default outbound access IP was assigned to the virtual machine.

As previously communicated in Cloudera Customer Advisory-866, the default outbound access connectivity for virtual machines in Azure is retired on 30 September, 2025. After this date, all new virtual machines (new clusters as well as repaired/upgraded cluster nodes) need to use explicit outbound connectivity methods, such as Azure NAT Gateway, Azure Load Balancer outbound rules, or a directly attached Azure public IP address.

The retirement of the default outbound access connectivity will have the following impact:

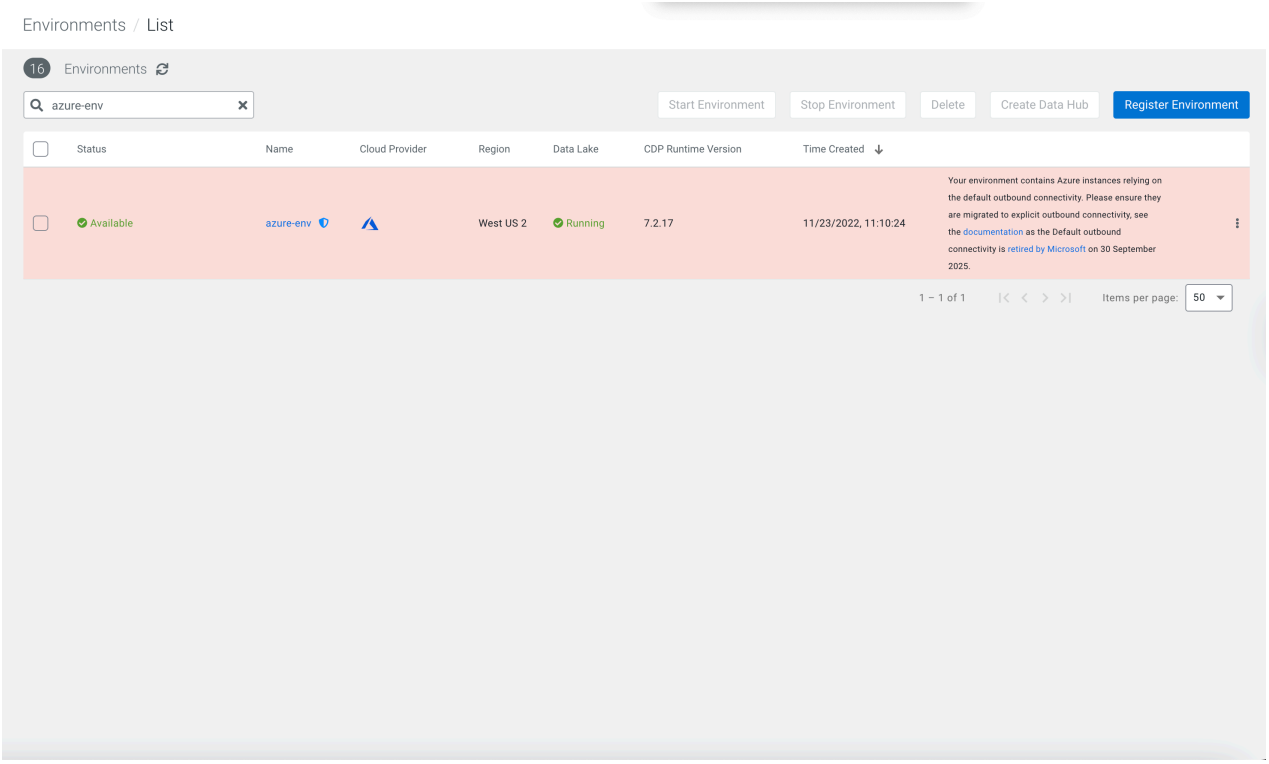
- Existing virtual machines that use the default outbound access will continue to work after the retirement date. However, certain operations, such as operating system upgrade, repair, and recover, will break your cluster.

- New virtual machines created after September 30, 2025, even if deployed into existing VNets and subnets, will not receive default outbound access. These virtual machines must be configured with an explicit outbound method to access the internet.

Cloudera recommends enabling explicit outbound connectivity for your virtual machines using one of the following Microsoft-recommended options, in order of preference:

- Associate a NAT Gateway with the subnet hosting your Cloudera environment. This is the preferred and most scalable option.
- Add a Firewall or Network Virtual Appliance (NVA) to your virtual network, and point traffic to the virtual machine using a User Defined Route (UDR).

To assist with this transition, the Cloudera Control Plane displays the following warning message for any cluster that does not use explicit outbound connectivity:

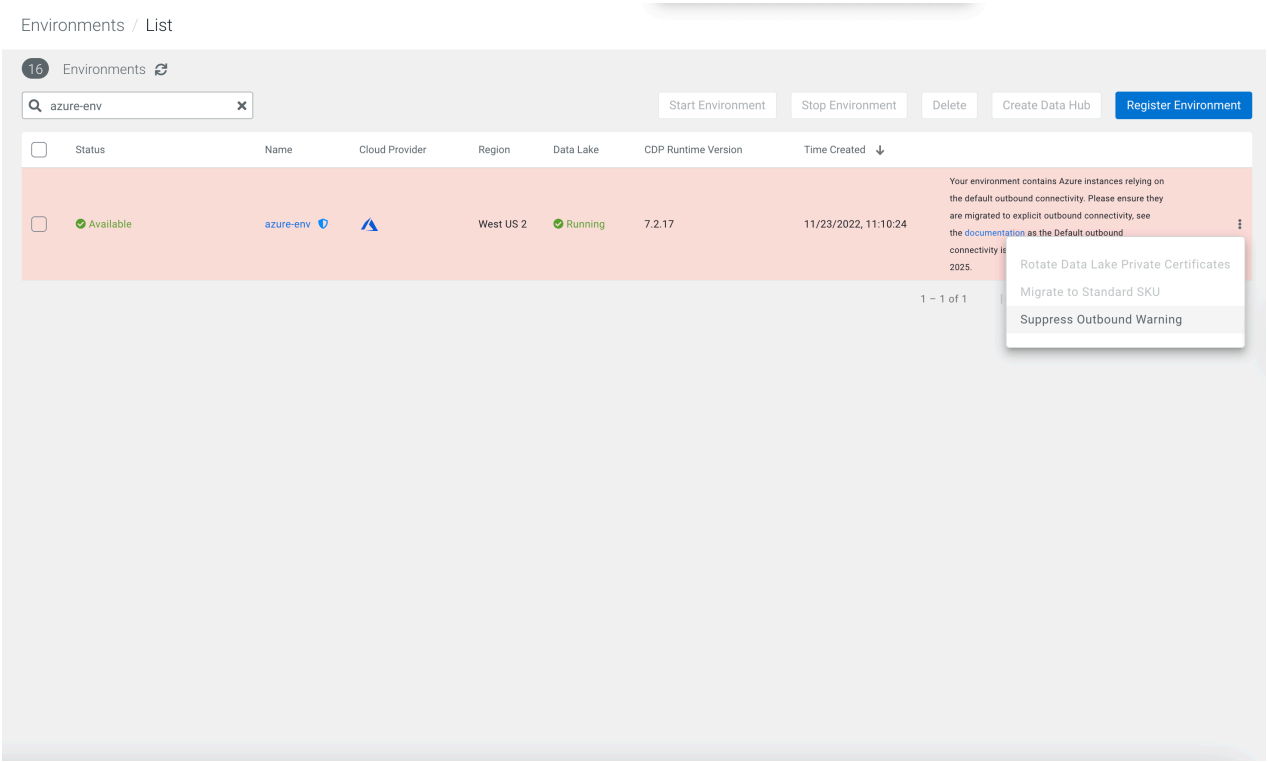


Note:

If you have already configured a Firewall or Network Virtual Appliance (NVA) to allow explicit outbound connectivity, you can safely disable the notification by selecting ⋮ Suppress Outbound Warning.

This step is necessary because the Cloudera Management Console lacks the credential permissions needed to automatically validate these outbound configurations on the Azure side.

The notification about the transition automatically disappears after one of the explicit outbound connectivity methods is detected for the cluster. As it is your responsibility to configure this connectivity to maintain optimal cluster performance and security, you can also turn off the notification for all affected resources in your account by clicking ⋮ Suppress Outbound Warning .



After acknowledging the responsibilities, the warning message will not appear again for the affected resources: