

Azure environments

Date published: 2024-01-01

Date modified: 2024-08-15

CLOUDERA

Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

Azure environments overview.....	5
Azure environments requirements checklist.....	5
Activating an Azure environment from CDW.....	7
Retaining PostgreSQL backups in Azure Environments.....	9
Setting up minimum permissions.....	9
Viewing and editing Azure environment details.....	11
Enabling a private CDW environment.....	12
Setting up the environment for private cluster deployment.....	13
Creating and registering the Azure environment.....	14
Configuring a custom private DNS zone.....	16
Configuration options for a private CDW deployment.....	16
Overlay networking.....	17
Enabling CNI overlay networking.....	18
Using AKS monitoring with Cloudera Data Warehouse in Azure environments.....	18
Deactivating environments on Azure.....	19
Custom tags in Azure environments.....	19
Azure load balancers in Cloudera Data Warehouse.....	20
Enabling internal load balancer.....	22
Granting remote access to Kubernetes clusters on Azure Kubernetes Service.....	23
Revoking remote access to Kubernetes clusters on Azure Kubernetes Service.....	26
Managed storage access.....	26
Setting up managed storage access.....	28

Creating the CDP environment.....	28
Creating a UMS group and machine users.....	29
Creating a new Database Catalog.....	31
Creating a tenant-specific Virtual Warehouse.....	31

Azure Kubernetes Service upgrade.....	32
--	-----------

Setting scratch space limit for spilling Impala queries in Azure environments.....	32
---	-----------

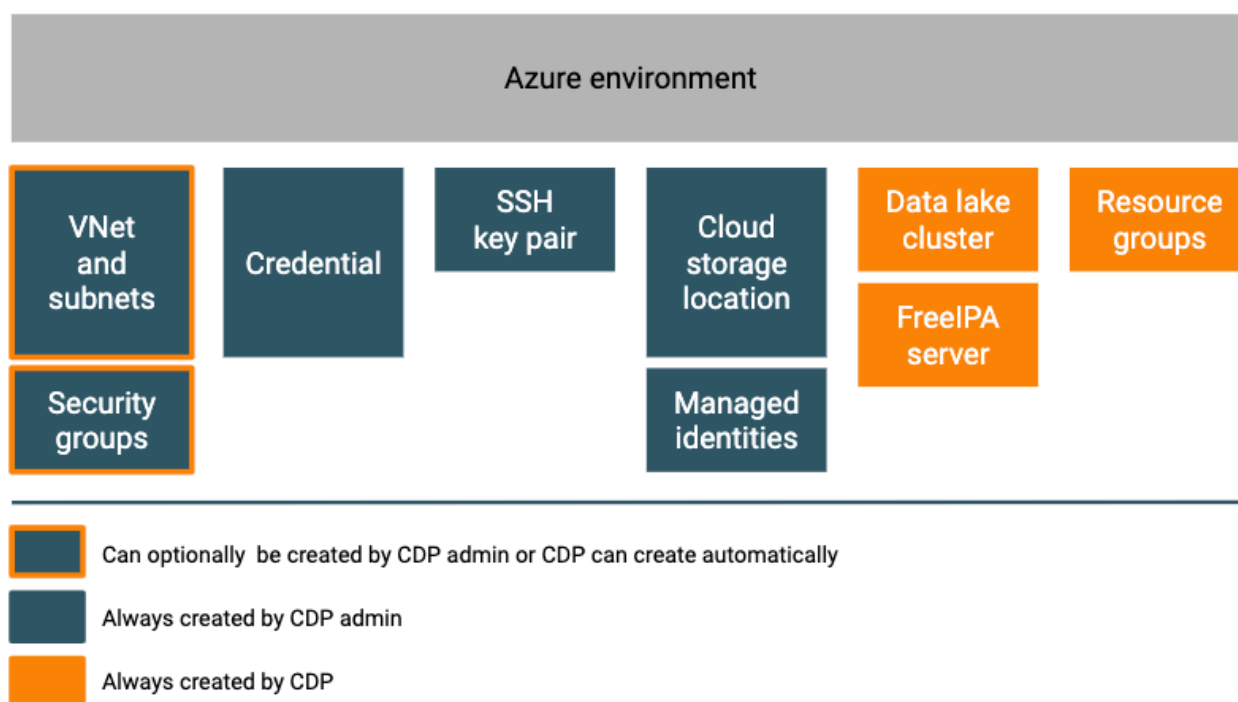
Azure environments overview

Learn about environments on Azure in CDP Public Cloud, including requirements you must meet before activating your environment in Cloudera Data Warehouse (CDW).

The CDP environment is closely related to the virtual private network in your cloud provider account. Registering an environment with Management Console provides CDP with access to your cloud provider account and identifies resources in your account that CDP services can access, including Cloudera Data Warehouse. A single environment is contained within a single cloud provider region, so all resources deployed by CDP are deployed within that region within one specific virtual network. After you have registered an environment with Management Console, you can activate the environment in CDW. Next, you create Database Catalogs, which enables CDW to access the associated Data Lake. Finally, you can create Virtual Warehouses in CDW that use the Database Catalog and its underlying environment.

Ranger Authorization is not enabled by default in Azure environment. You must enable RAZ as described in the [Introduction to RAZ on Azure environments](#).

The following diagram shows the components of an Azure environment:



For more information about Azure environments in CDP, see the links in the "Related information" section at the bottom of this page.

Azure environments requirements checklist

To successfully activate Azure environments with Cloudera Data Warehouse (CDW) service, make sure your environment meets the requirements listed in this topic.

1. Specific networking requirements



Important: Cloudera recommends activating CDW service and Cloudera Machine Learning (CML) service workloads on different subnets to maximize performance.

- Make sure Azure VNet subnets are large enough to support the CDW load

When an Azure environment is activated for CDW service, an Azure Kubernetes Service (AKS) cluster is provisioned in your subscription. The AKS cluster uses the [Azure Container Networking Interface \(CNI\)](#) plug-in for Kubernetes. This plug-in assigns IP addresses for every pod running inside the Kubernetes cluster. By default, the maximum number of pods per node is 30. This means that you need approximately 3,200 IP addresses for a 99-node cluster. If you activate an environment for CDW service, make sure that the subnets are large enough on the Azure VNet for the CDW load. Cloudera recommends using a CIDR/21 subnet or larger.

The following IP address ranges are defined as part of the AKS cluster creation process:

- 10.20.0.0/16
- 172.17.0.1/16

Make sure these ranges do not overlap with your VNET address ranges.

If you want to activate CDW on a very small subnet, overlay networking is recommended.

- Configure service endpoints on CDW subnets

You must configure service endpoints on the subnets used for the CDW service. This ensures that the network traffic between CDW components and Azure services remain on the Microsoft Azure backbone network. Microsoft.Storage and Microsoft.SQL must be registered; otherwise, the CDW service does not start on existing Azure VNets.



Note: If you have used only Data Hub in your CDP environment, you must check and possibly configure connections to additional outbound destinations for Cloudera Data Warehouse (CDW).

For more information, see [Azure Outbound Network Access Destinations](#) and [Virtual Network service endpoints](#) in the Azure documentation.

- Firewall exceptions for Azure AKS

If you need to restrict egress traffic in Azure, reserve a limited number of ports and addresses for cluster maintenance tasks including cluster provisioning. See [Control egress traffic for cluster nodes in Azure Kubernetes Service \(AKS\)](#) to prepare your Azure environment for AKS deployment.

For maintenance, Cloudera recommends putting the Azure portal URLs on the allowlist of your firewall or proxy server. For more information, see [Allow the Azure portal URLs on your firewall or proxy server](#).

2. Use only app-based credentials

For the Data Warehouse service, you must use only an app-based credential, which requires the Contributor role to create a new service principal. For more information about creating an app-based credential for the environment you want to use for the Data Warehouse service, see [Create an app-based credential](#). If you need to change your environment credential, see [Change environment's credential](#).

3. Obtain permissions

For environments that you plan to use for the Data Warehouse service, generally you need to ensure that the application you create in Azure has the built-in [Contributor](#) Azure role at the resource group level. For more information, see the description of app-based credentials in [Credential options on Azure](#) in the Management Console documentation.



Important: Alternatively, you can use [minimum permissions](#).

4. Created Azure app must have access to the storage account used during environment registration

Ensure that the application, which the Azure app-based credentials are attached to, have access to the ADLS Gen2 storage location you specify in Step 6 when you [register an Azure environment](#) topic. For information about storage accounts for Azure environments, see [ADLS Gen2 and managed identities](#) and [Minimal setup for cloud storage](#).

5. AKS cluster credential

During the environment activation process, you must provide the resource ID of a user-assigned managed identity to create the Azure Kubernetes cluster resource. This crucial identity allows for impersonation of the AKS cluster. To ensure this functionality, you must assign the appropriate [permissions](#) to the identity resource. Cloudera strongly recommends assigning the same permissions as those assigned to the app-based credentials. Additionally, you must grant the [Storage Blob Data Owner](#) role to the AKS identity on your Data Lake storage account. This permission is required for activating an environment in CDW.

To affirm that your resources, such as managed identities, virtual networks, and so on, are accessible, verify that you have assigned the appropriate permissions to any separate resource groups where they reside.

For more information about managed identities, see ["AKS managed identities"](#).

6. List of required resources for Azure environments

There is no cross-regional support for Cloudera Data Warehouse service. Azure environments used for the Data Warehouse service must have the following resources available in the specific Azure region where the environment is registered:

- [Azure Kubernetes Service \(AKS\)](#)
- [Azure Database for PostgreSQL](#)
- [Azure Data Lake Storage Gen2](#)
- [Virtual machine scale sets](#)
- [Ev3 series VM](#)
- [Dsv3-series VM](#)
- [Availability zones for AKS](#) (optional)



Important: To enable the Availability zones for AKS option when activating the environment for the Cloudera Data Warehouse service, make sure your region supports [Azure Availability](#).

7. Azure subscription should be in a similar region as the resources

Ensure that your Azure subscription is in close proximity to the region where your resources are deployed and are governed by the same regulatory laws. The [Azure region requirements](#) says, "CDP requires that the ADLS Gen2 storage location provided during environment registration must be in the same region as the region selected for the environment." For more information, see [Azure geographies](#).

8. Support for third party k8s components

To minimize the IP address usage of the AKS clusters, Cloudera currently creates the Kubernetes (k8s) compute node pools with a "maxpods" settings of 15. This setting specifies the maximum number of pods that can run on a node. By default, Azure deploys several key components to every k8s cluster to provide infrastructure services. The number of components can vary based on the enabled services in your Azure subscription. Generally it takes 6-8 pods to host these infrastructure services inside the AKS clusters. Additionally Cloudera deploys 1-2 other pods to host Cloudera compute resources.

If you want to deploy services to the AKS cluster, make sure you do not exceed the maxpods limits of the cluster.

Activating an Azure environment from CDW

To use an Azure environment for Cloudera Data Warehouse (CDW) Public Cloud you must first activate it.

About this task

When you activate an environment, CDP creates an AKS (Azure Kubernetes Services) cluster to host Kubernetes-based resources. The underlying compute, network resources are managed by Azure, including the following ones:

- Resource group
- Compute instances, which are virtual machine scale sets
- Load balancer(s)
- Public IP address(es)
- Network security group
- Disk(s)

CDP supports the following Azure virtual machines and their corresponding compute instance types (Hive and Impala executors), which you select during environment activation:

Table 1: Compute Instance Types

Azure VM	Processor Type	Usage	Virtual Warehouse Support
Standard_E16pds_v5	ARM	Compute	Impala
Standard_E16_v3	Intel	Compute	Hive and Impala
Standard_E16ds_v4	Intel	Compute (default)	Hive and Impala
Standard_E16ads_v5	AMD	Compute	Hive and Impala
Standard_E16ds_v5	Intel	Compute	Hive and Impala
Standard_D8s_v4	Intel	Shared services (default)	Hive and Impala
Standard_D8as_v5	Intel	Shared services, used with AMD compute instance Standard_E16ads_v5	Hive and Impala

Three instances are added to the cluster as needed for shared services (always on components). Three shared nodes are dStandard_E2s_v3 MemoryOptimized using flexserver, for the Amazon Relational Database Service (RDS). These shared nodes are used for Hue and Data Visualization user metadata. For more information, see [Always active, shared services](#).

Deploying a private AKS

If you want to deploy a Private AKS cluster, you must use the CDP CLI as described in [AKS deployment configuration options](#).

Before you begin

- Obtain the DWAdmin role.
- Review the [Azure environments requirements checklist](#).

Procedure

1. In the CDW service, in Environments, locate the environment that you want to activate.
2. Click Activate.

3. In Activate Environment, configure the environment or accept the defaults:

- Select the Compute VM Size based on your workload.
- Select a Subnet inside the virtual network (VNet) that you want to use for CDW.

The VNet that you select must have a sufficient number of free IP addresses.

- Select Enable internal load balancer (ingress) to distribute traffic inside a virtual network.

For information about load balancers, see ["Azure load balancers"](#).

- Optionally, select Enable availability zones for AKS.
- [Specify a user-assigned, managed identity](#) for the AKS cluster.
- Select Enable AKS monitoring and then select the workspace from the adjacent drop-down list.
- Add trusted endpoint IP CIDRs for your load balancer in a comma-separated list to the Enable IP CIDR for the load balancer text box.
- Accept the default Use CNI overlay networking if IP address exhaustion is a concern for your deployment.

4. To use custom repository, select the Use Custom repository option.

Depending in your custom repository type, input ecr, acr, or docker in the Registry Type field.

Specify the URL of your custom repository. Fields for your user name and password appear. You do not need to enter these credentials.

5. Click ACTIVATE.

Related Information

[Azure Standard_E16ds_v4 instances](#)

[Azure Standard_E16_v3 instances](#)

[Bring your own control plane managed identity](#)

[Overlay networking](#)

Retaining PostgreSQL backups in Azure Environments

When you create a Cloudera Data Warehouse cluster using the CDP CLI create-cluster command, any PostgreSQL backup retention period you set on your Cloud Provider side, is observed by CDP.

Procedure

1. In Azure, configure backupRetentionDays.
2. Create a DW cluster using the CDP CLI create-cluster command.
The DW cluster will retain the PostgreSQL backups according to your configuration.

Setting up minimum permissions

You learn how to configure the minimum permissions required for Cloudera Data Warehouse (CDW) on Azure environments. The minimum permissions govern access control between CDW, Azure resources, and the Azure storage account.

Before you begin

Obtain the permissions and roles as described in ["Azure permissions"](#).

Procedure

1. Write code for a custom role that contains the minimum permissions required for CDW on Azure.

2. In the properties section of your subscription code, change variables into actual values.
For example, make the following substitutions to the sample code shown below:
 - [YOUR-SUBSCRIPTION-ID]: Your subscription ID in use.
 - [YOUR-RESTRICTED-ROLE-NAME]: The custom role name which is assigned to the application.
 - [YOUR-SINGLE-RESOURCE-GROUP]: The original resource group name.
3. Insert the code for the custom role into the actions section of your subscription.
The following sample code shows the properties and actions sections you need to add to your subscription:

```
{
  "properties": {
    "roleName": "[YOUR-RESTRICTED-ROLE-NAME]",
    "description": "Additional permissions to activate CDW clusters.",
    "assignableScopes": [
      "/subscriptions/[YOUR-SUBSCRIPTION-ID]/resourceGroups/[YOUR-SINGLE-RESOURCE-GROUP]"
    ],
    "permissions": [
      {
        "actions": [
          "Microsoft.Resources/deployments/cancel/action",
          "Microsoft.Resources/deployments/validate/action",
          "Microsoft.ContainerService/managedClusters/write",
          "Microsoft.ContainerService/managedClusters/agentPools/write",
          "Microsoft.ContainerService/managedClusters/read",
          "Microsoft.ContainerService/managedClusters/agentPools/read",
          "Microsoft.ContainerService/managedClusters/accessProfiles/listCredential/action",
          "Microsoft.ContainerService/managedClusters/delete",
          "Microsoft.ContainerService/managedClusters/rotateClusterCertificates/action",
          "Microsoft.DBforPostgreSQL/flexibleServers/read",
          "Microsoft.DBforPostgreSQL/flexibleServers/write",
          "Microsoft.DBforPostgreSQL/flexibleServers/delete",
          "Microsoft.DBforPostgreSQL/flexibleServers/firewallRules/write",
          "Microsoft.DBforPostgreSQL/flexibleServers/firewallRules/read",
          "Microsoft.DBforPostgreSQL/flexibleServers/firewallRules/delete",
          "Microsoft.DBforPostgreSQL/flexibleServers/configurations/read",
          "Microsoft.DBforPostgreSQL/flexibleServers/configurations/write",
          "Microsoft.DBforPostgreSQL/flexibleServers/databases/read",
          "Microsoft.DBforPostgreSQL/flexibleServers/databases/write",
          "Microsoft.DBforPostgreSQL/flexibleServers/databases/delete",
          "Microsoft.DBforPostgreSQL/servers/virtualNetworkRules/write",
          "Microsoft.DBforPostgreSQL/servers/databases/write",
          "Microsoft.Network/privateDnsZones/A/read",
          "Microsoft.Network/privateDnsZones/A/write",
          "Microsoft.Network/privateDnsZones/A/delete",
          "Microsoft.Network/privateDnsZones/virtualNetworkLinks/read",
          "Microsoft.Network/virtualNetworks/subnets/joinViaServiceEndpoint/action",

```

```

        "Microsoft.Network/routeTables/read",
        "Microsoft.Network/routeTables/write",
        "Microsoft.Network/routeTables/routes/read",
        "Microsoft.Network/routeTables/routes/write",
        "Microsoft.Network/routeTables/join/action",
        "Microsoft.Network/natGateways/join/action",
        "Microsoft.Network/virtualNetworks/subnets/joinLoadBal
ancer/action",
        "Microsoft.Network/privateDnsZones/write",
        "Microsoft.Network/privateDnsZones/read",
        "Microsoft.Network/privateDnsZones/virtualNetworkLinks/
write",
        "Microsoft.Network/privateEndpoints/write",
        "Microsoft.Network/privateEndpoints/read",
        "Microsoft.Network/privateEndpoints/privateDnsZoneGr
oups/read",
        "Microsoft.Network/privateEndpoints/privateDnsZoneGro
ups/write",
        "Microsoft.Network/privateEndpoints/privateDnsZoneGro
ups/delete",
        "Microsoft.Network/privateDnsZones/join/action"
    ],
    "notActions": [],
    "dataActions": [],
    "notDataActions": []
  }
}
]
}
}

```

Viewing and editing Azure environment details

After activating an Azure environment for Cloudera Data Warehouse (CDW) Public Cloud, you can change or view its details.

About this task

You can view CDP Azure environment details without leaving the CDW service UI. You can make the following changes:

- Add a description for the environment, which makes it easier to identify.
- Add or edit the list of IP CIDR(s) for Kubernetes cluster, which enable access from your internal network to the Kubernetes endpoints.
- Add or edit the list of IP CIDR(s) for the load balancer, which enable access from your internal network to the load balancer endpoints of services such as Hive, Impala, or Hue.
- Renew the certificate for this environment.

This option renews the encrypt certificates. Renewal happens automatically 30 days before the expiry date of the certificate.

- Refresh kubeconfig

After making changes that alter the Kubernetes configuration, such as [rotating certificates](#), as recommended by Microsoft, you must click Refresh kubeconfig.


Required role for viewing environment details: DWUser

Required role for editing environment details: DWAdmin

Before you begin

You must activate an environment before you can view or edit its details.

Procedure

1. In the CDW service, go to the Environments tab.
2. Locate the environment that you want to view and click  Edit .
The **Environment Details** page is displayed.
3. Go to the Configurations tab.
4. Make changes described above, or just view information, such as when it was created and last updated, or how many Database Catalogs and Virtual Warehouses use the environment,
5. If you have made changes, click Apply Changes.

Enabling a private CDW environment

An introduction to the terms and components involved in creating a private CDW environment prepares you to successfully complete the required tasks. You see a network configuration diagram.

AKS simplifies container-based application deployment and management. When you create an AKS cluster, a control plane is automatically created and configured. The Azure platform configures the secure communication between the control plane and compute nodes. In a private cluster, the API server for the control plane has an internal IP address. The Azure nodes and the Kubernetes control plane components do not have publicly routable IP addresses. When you deploy a private cluster, the network traffic between your API server and your node pools remains only on the private network.

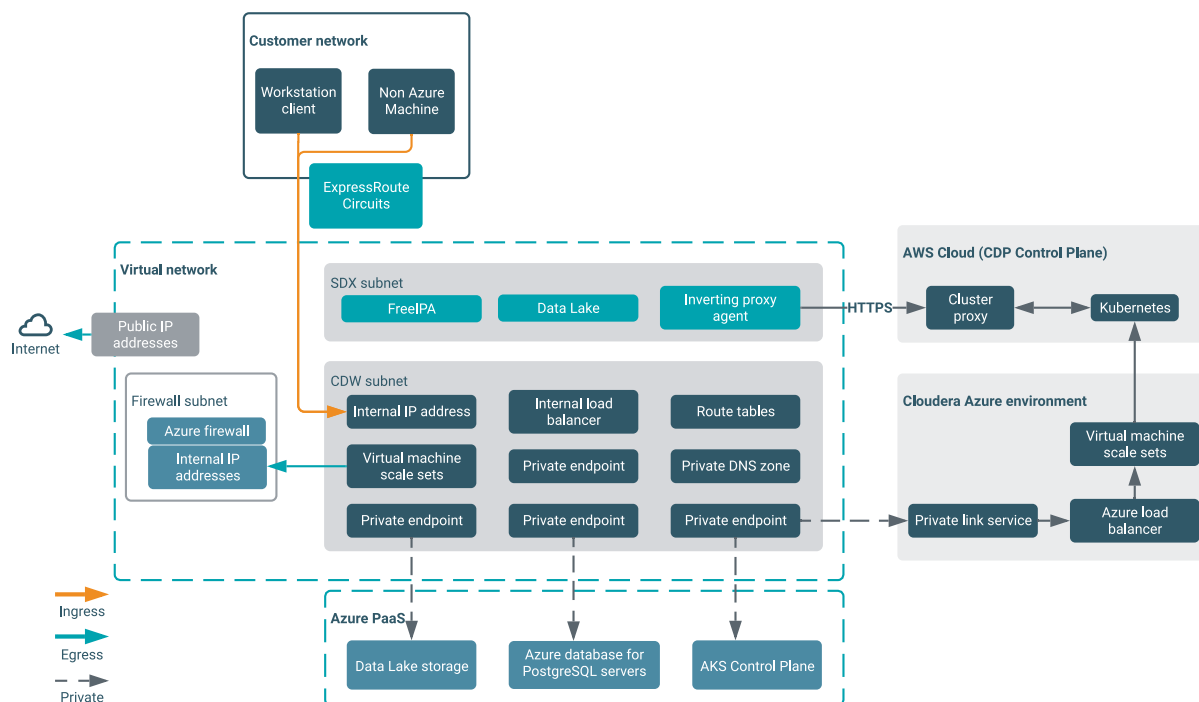
Cluster Connectivity Manager (CCM) {version 2} enables the CDP Control Plane to communicate with the Kubernetes control plane using an inverting proxy solution. Cloudera Data Warehouse (CDW) can communicate with the Kubernetes control plane and the other resources, such as virtual machines deployed in your network, by using a special established channel.

Large enterprises have a requirement to create private CDW environments in the public cloud similar to the on-premises one in terms of security and networking. Generally, public access to your cloud environment, resources, and deployed applications is not allowed.

A private environment has the following criteria:

- Kubernetes API server for the control plane must have private API endpoints
- Hue, DAS, JDBC, or ODBC applications can only be accessed through private endpoints
- Egress Kubernetes traffic must be controlled using a transparent proxy
- Storage, SQL database, and other resources can only be accessed through private endpoints
- CDP Control Plane can only be accessed through private endpoints using Azure Private Link Service
- Ability to configure a DNS zone in your environment

The following layout shows a typical network for private environments:



Setting up the environment for private cluster deployment

In Azure, you perform steps and follow links to Azure documentation to meet prerequisites for enabling a private CDW environment.

Procedure

1. In the Microsoft Azure portal, [create a resource group](#) for CDP.

The resource group provides a management layer that enables you to create, update, and delete resources in your Azure account.

2. (Optional) [Create a private storage account](#) and [network access rules](#) to block all internet traffic.

The private endpoints allow clients on a virtual network (VNET) to securely access data over a Private Link.

3. [Create a VNet](#) and [add a subnet](#).

VNet is the fundamental building block for your private network in Azure. It enables Azure resources, such as Azure Virtual Machines, to securely communicate with each other and with the internet.

4. Create a [delegated subnet](#) for Azure Database for PostgreSQL Flexible Server.
5. (Optional) Configure the CDP Control Plane Private Link service.

To ensure that all egress traffic including CDP Control Plane traffic travels only through private networks, you must configure Cluster Connectivity Manager (CCM) (v2) and CDP Control Plane private endpoints properly for the selected VNet.

Contact your Cloudera account representative to register the CDP Control Plane Private Link service endpoints in your Azure VNet.

6. Configure custom DNS on the VNet to resolve Azure Private DNS zones.

To resolve private endpoint DNS records, the VNet DNS servers must be capable of resolving Azure DNS records.

7. [Disable network endpoint policies for private endpoints](#) and [Azure Private Link Service](#).
8. Configure the following firewall exceptions for CDW and AKS on the egress firewall, and for storage account endpoints, if the account is not Private-Link enable: [Outbound network access destinations](#) and [Restrict egress traffic in AKS](#)
9. [Configure user-defined routing](#) (UDR) on the VNet to forward all traffic to an egress firewall and link it to the subnet.

Creating and registering the Azure environment

In CDP, you perform a step-by-step procedure to create the environment, and select options for the database, virtual machines, and CCM during registration.

About this task

Procedure

1. Create a CDP Azure environment in the VNET that you created earlier.
2. Select private environment options for the PostgreSQL database, virtual machines, and CCM (v2) while registering the Azure environment in CDP.


On the environment registration screen, the Enable Cluster Connectivity Manager option is enabled by default. It ensures that all traffic from Cloudera Control Plane to your cloud resources travels through a secured HTTPS tunnel. CDW Private AKS only works with CCM (v2).

Do not create public IPs so that Azure VMs have private IP addresses only.

3. Enable the Create Private Endpoints option.

By default, the PostgreSQL Azure database provisioned for your Data Lake is reachable through a service endpoint (public IP address). To increase security, you must select to have it reachable through a private endpoint

instead of a service endpoint. You can create a new private DNS zone for the database or you can use your own. CDW will also use the zone specified here.



Network

Select the network and subnets for the environment. You can manage networks and [Click here](#) to refresh networks and subnets from the cloud provider.

Select Network


?

Network CIDR*

?

☒ Create private subnets

☒ Enable CCM (Cluster Connectivity Manager)

 When using CCM without the Public Endpoint Access Gateway enable

☐ Enable Public Endpoint Access Gateway

☒ Create Private Endpoints

Select Private DNS Zone for Database

?

☐ Create Public IPs

Related Information

[Working with Azure environments](#)

[Register an Azure environment from CDP UI](#)

[Enabling private endpoint for PostgreSQL on Azure](#)

Configuring a custom private DNS zone

You learn the requirements for configuring a DNS zone for AKS and Azure Database for PostgreSQL - Flexible Server. From the syntax information, you see how to construct a CDP CLI command that configures the DNS zone.

About this task

When you set up the environment for private cluster deployment, you configure custom DNS on the VNET to resolve Azure Private DNS zones. If this default AKS Private DNS zone creation behavior is not suitable for your needs, you can use the CDP CLI to further customize the zones. Activating CDW clusters with the private CDW option enables multiple different AKS features at once to support private CDW deployments.

Before you begin

- Create the DNS zone `privatelink.<region>.azmk8s.io`.
- Grant at least the private dns zone contributor and vnet contributor roles to the service principal (Azure Enterprise Application) used for environment creation.
- If the Private DNS Zone is in a different subscription than the AKS cluster, you need to register `Microsoft.ContainerServices` in both the subscriptions.

Procedure

1. Configure a DNS zone for AKS using the CDP CLI.

Use the following syntax:

```
cdp dw create-cluster --<create_cluster_options> \
--environment-crn <CLOUDERA_RESOURCE_NAME_OF_ENVIRONMENT> --use-private-l
oad-balancer \
--azure-options <property>=<value>,<property>=<value> ... ,\
enablePrivateAks=true,\
privateDNSZoneAKS=<PRIVATE_DNS_ZONE_RESOURCE_ID>
```

where: Azure options might be `userAssignedManagedIdentity={managed-identity}`, `subnetId={subnet-id}`, `outboundType=UserDefinedRouting`, `enablePrivateSQL=true`, and the `PRIVATE_DNS_ZONE_RESOURCE_ID` looks something like this:

```
privateDNSZoneAKS=/subscriptions/{sub-id}/resourceGroups/{rg-name}/provi
ders/Microsoft.Network/privateDnsZones/privatelink.{region}.azmk8s.io
```

If you specify none for the `privateDNSZoneAKS` parameter, Azure defaults to public DNS which means AKS does not create a Private DNS Zone.

2. Create a DNS zone for Azure Database for PostgreSQL.

When you set up the environment for private cluster deployment, you need to specify a custom private DNS zone. See the [Use a Private DNS zone](#) topic in Azure documentation to prepare your DNS infrastructure.

Related Information

[Configure a private DNS Zone](#)

Configuration options for a private CDW deployment

The list of properties and explanations of how to use them, followed by an example, prepares you to configure a private CDW deployment.

To customize the CDW deployment, you can use the following properties and values after the `--azure-options` parameter:

outboundType

You can customize an AKS cluster with a unique `outboundType` depending on your network configuration for egress traffic.

Possible values: `LoadBalancer/UserDefinedRouting/UserAssignedNATGateway`

enablePrivateSQL=true/false

If you set this option to true, CDW creates an Azure Database for PostgreSQL - Flexible Server with Private access (virtual network integration) enabled to bring the server inside your Virtual Network (VNET).

privateSQLSubnetName

You can specify the delegated subnet for Azure Database for PostgreSQL - Flexible Server.

privateDNSZoneSQL

You can specify the private DNS zone for Azure Database for PostgreSQL - Flexible Server.

enablePrivateAks=true/false

By using a private cluster, you can ensure network traffic between your API server and your node pools remains on the private network. Enabling this option instructs CDW services to create an AKS cluster with a private API.

The `--create-cluster` options are:

- `--use-private-load-balancer`
- `--no-use-private-load-balancer`

An internal load balancer makes a Kubernetes service accessible only to applications running in the same virtual network as the Kubernetes cluster.

The following example shows how to configure a DNS zone for AKS:

```
cdp dw create-cluster --environment-crn --use-private-load-balancer \
--azure-options userAssignedManagedIdentity={managed-identity},subnetId={s
ubnet-id},enablePrivateSQL=true,\
enablePrivateAks=true, \
privateDNSZoneAKS=/subscriptions/subid/resourceGroups/rname/providers/Mi
crosoft.Network/privateDnsZones/privatelink.{region}.azmk8s.io
```

Related Information

[Customize user-defined routes \(UDR\) in Azure Kubernetes Service \(AKS\) - Azure Kubernetes Service](#)

[Private Link Azure Database for PostgreSQL Single server](#)

[Create a private Azure Kubernetes Service cluster - Azure Kubernetes Service](#)

[Create an internal load balancer - Azure Kubernetes Service](#)

[Create an AKS cluster with a user-assigned NAT Gateway](#)

Overlay networking

By default, CDW uses CNI overlay networking.

Default CDW networking

Using [Azure Container Networking Interface \(CNI\)](#), every pod gets an IP address from the node subnet and is accessed directly. Each IP address must be unique across your network space, and you must plan for them in advance of deploying your CDW cluster. Each node has a configuration parameter for the maximum number of pods that it can support. The equivalent number of IP addresses per executor node is reserved up front for it. This requires advanced planning and it can often lead to IP address exhaustion. As an alternative, you must rebuild the cluster in large subnets so your cluster can meet your applications' demands. You can configure the maximum pods that are

deployable to an executor node when you create the cluster or when you create new executor node pools. However, if you do not specify the maximum number of pods for the `maxPods` property when you create new executor node pools, by default each executor node gets 30 pods (with one IP address per pod).

About using overlay networking

To avoid IP address exhaustion, you can enable the overlay networking feature when you activate an Azure environment to use with CDW. For a full description of CNI Overlay networking in AKS, see the [Microsoft documentation](#).



Important: Third-party overlay networking solutions like Weaveworks Weave Net and Project Calico are not supported in Microsoft Azure. Cloudera strongly recommends that you do not deploy these third-party overlay networking solutions.

Enabling CNI overlay networking

If you are experiencing IP address exhaustion for your Azure environment in Cloudera Data Warehouse (CDW) Public Cloud, you can enable a CNI overlay network for the environment.

About this task

Required role: EnvironmentAdmin or PowerUser

After you have registered your environment with CDP, navigate to the CDW service and perform the following steps to configure your Azure environment to use overlay networking.

Procedure

1. In the Cloudera Data Warehouse, in Environments, search for and locate the environment that you want to configure.
2. Click Activate to activate the environment.
3. In Activation Settings, check Use CNI overlay networking.

Docker Bridge CIDR

172.17.0.1/16

☒ Use kubenet networking

Please check the [limitations](#) of kubenet networking before activation.

☐ Use Custom repository ⓘ

Custom Environment Subdomain

4. Click ACTIVATE.

Using AKS monitoring with Cloudera Data Warehouse in Azure environments

Cloudera Data Warehouse (CDW) provides an option to enable and leverage AKS monitoring in Azure environments. When you enable AKS monitoring from the CDW UI and select a Log Analytics workspace, CDW sends the log data to Azure Monitor Logs.

About this task

You can use the Log Analytics tool to edit and run log queries, and create dashboards and alerts from the log queries.

Required role: EnvironmentAdmin or PowerUser

Before you begin

Before you enable AKS monitoring on CDW Public Cloud, you must configure a log analytics workspace in your Azure account. See the link to Microsoft Azure documentation for this task at the bottom of this page.

Procedure

1. In the CDW UI, go to the Environments tab.
2. Locate the environment for which you want to configure the AKS monitoring workspace.
3. Click Activate to launch the Activation Settings dialog box where you can configure the environment to use an AKS monitoring workspace:
4. In the Activation Settings dialog box, select Enable AKS monitoring.
5. Click the drop-down list to select a workspace.
6. Click Activate:

Related Information

[Create a Log Analytics workspace in the Azure portal](#)


Deactivating environments on Azure

Learn how to deactivate an Azure environment for Cloudera Data Warehouse (CDW) Public Cloud.

About this task

Required role: EnvironmentAdmin or PowerUser

Procedure

1. In the CDW service, go to the Environments tab.
 2. Locate the environment that you want to deactivate and click Deactivate.
The **Action** modal is displayed.
 3. On the Action modal, you can select environment deactivation options:
 - Choose Drop Data to remove the managed buckets that were created by CDW during environment activation. The underlying data in the data lake is untouched. Only the default Database Catalog is retained because it resides in the data lake. Non-default Database Catalogs and Virtual Warehouses are deleted.
-  **Important:** All metadata and the Ranger policies for non-default Database Catalogs are deleted.
- Choose Force Delete to drop the data and to remove the Database Catalogs and Virtual Warehouses that are associated with the environment.
4. Click OK to deactivate the environment.

Custom tags in Azure environments

Adding tenant-level custom tags help you to search or filter resources the key tag or key value in your Azure account.

Tenant-level tags apply to Cloudera-created resources across your organization's entire cloud provider account. The tenant-level tags can only be applied while creating a new CDP environment. The CDP environment custom tags propagated to the compute instances within the Cloudera Data Warehouse (CDW) service. You can use these tenant-level tags to identify and monitor the following resources within the CDW service for Azure environments:

- Compute instances in Hive and Impala Virtual Warehouses
- Compute instances in Druid cluster
- Azure virtual machine agent pools



Note: If you are activating a private CDW environment, ensure you do not have more than 15 tags total; otherwise, use a custom DNS zone or public DNS for the AKS cluster. In Azure, the maximum number of tags for the private DNS zones are limited to 15.

Azure load balancers in Cloudera Data Warehouse

Azure provides a public and a private (internal) load balancer to evenly distribute the inbound traffic that arrives at the load balancer's front end to backend pool instances, such as Azure virtual machine scale sets. By default, Azure Kubernetes Service (AKS) cluster uses the Standard SKU for the load balancer.

Depending on your network configuration, you can choose to use the public or the private (internal) load balancer. If you have defined custom user-defined (static) routes (UDR) to route the traffic to a firewall appliance in Azure or on-premises resources, then you can enable the internal load balancer while activating an Azure environment for Cloudera Data Warehouse (CDW). Otherwise, CDW uses the Standard public load balancer that is enabled by default when you provision an AKS cluster.

You enable the internal load balancer from the CDW UI when you activate the environment.

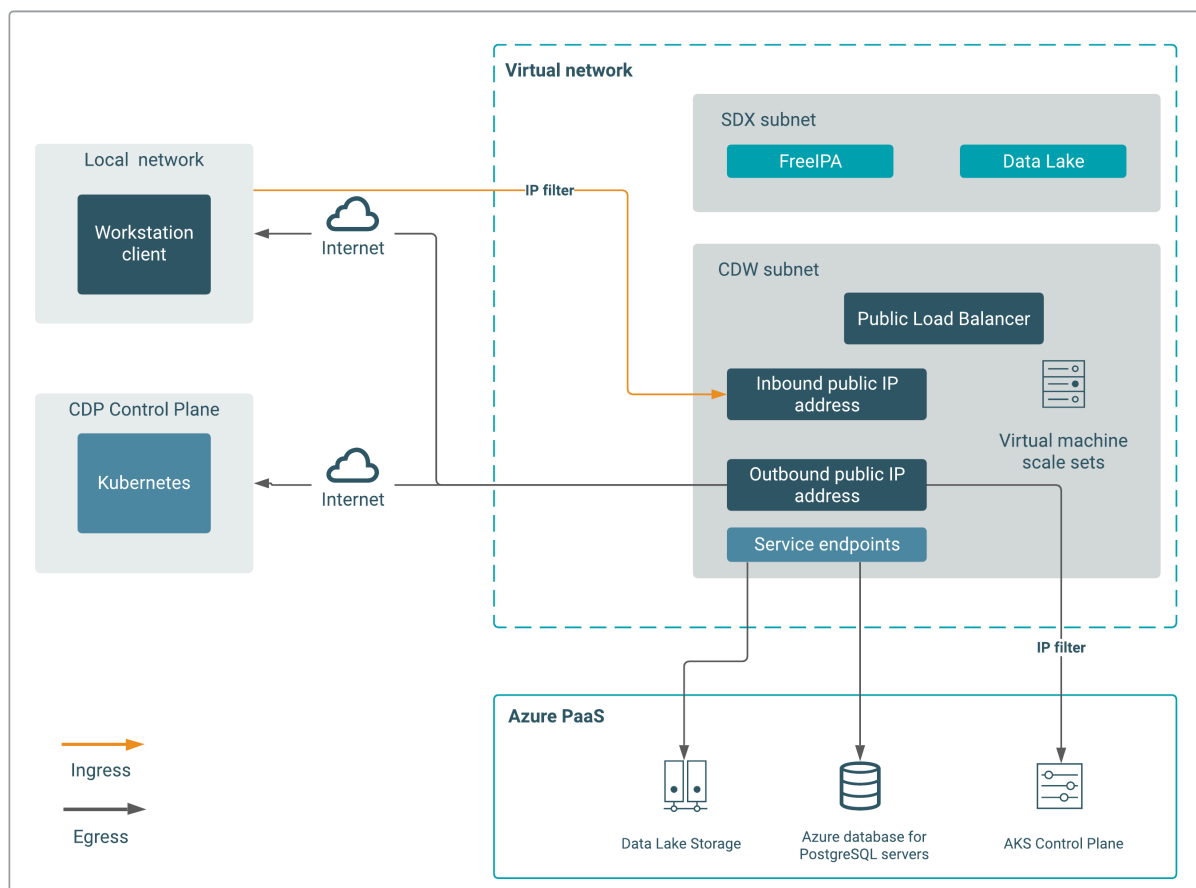
Using the public load balancer

When you activate an Azure environment in CDP, you provision Azure cloud resources such as the PostgreSQL database server and AKS cluster in your Microsoft Azure subscription. AKS contains a number of Azure infrastructure resources, including virtual machine scale sets, virtual networks, and managed disks. An AKS deployment contains a Standard public load balancer by default, that provides inbound and outbound connections to the cluster executor nodes.

The public load balancer has two public IP addresses to handle inbound and outbound connections. You can access Hue, JDBC, or the Impala shell using the AKS endpoints from the public internet. However, you can restrict access to the endpoints by defining an IP trusted list while or after activating the environment.

Also, the outbound connections towards CDP Control Plane and the image registry flows through the public load balancer using the outbound public IP address.

The following diagram shows the ingress and egress traffic when a public load balancer is used:

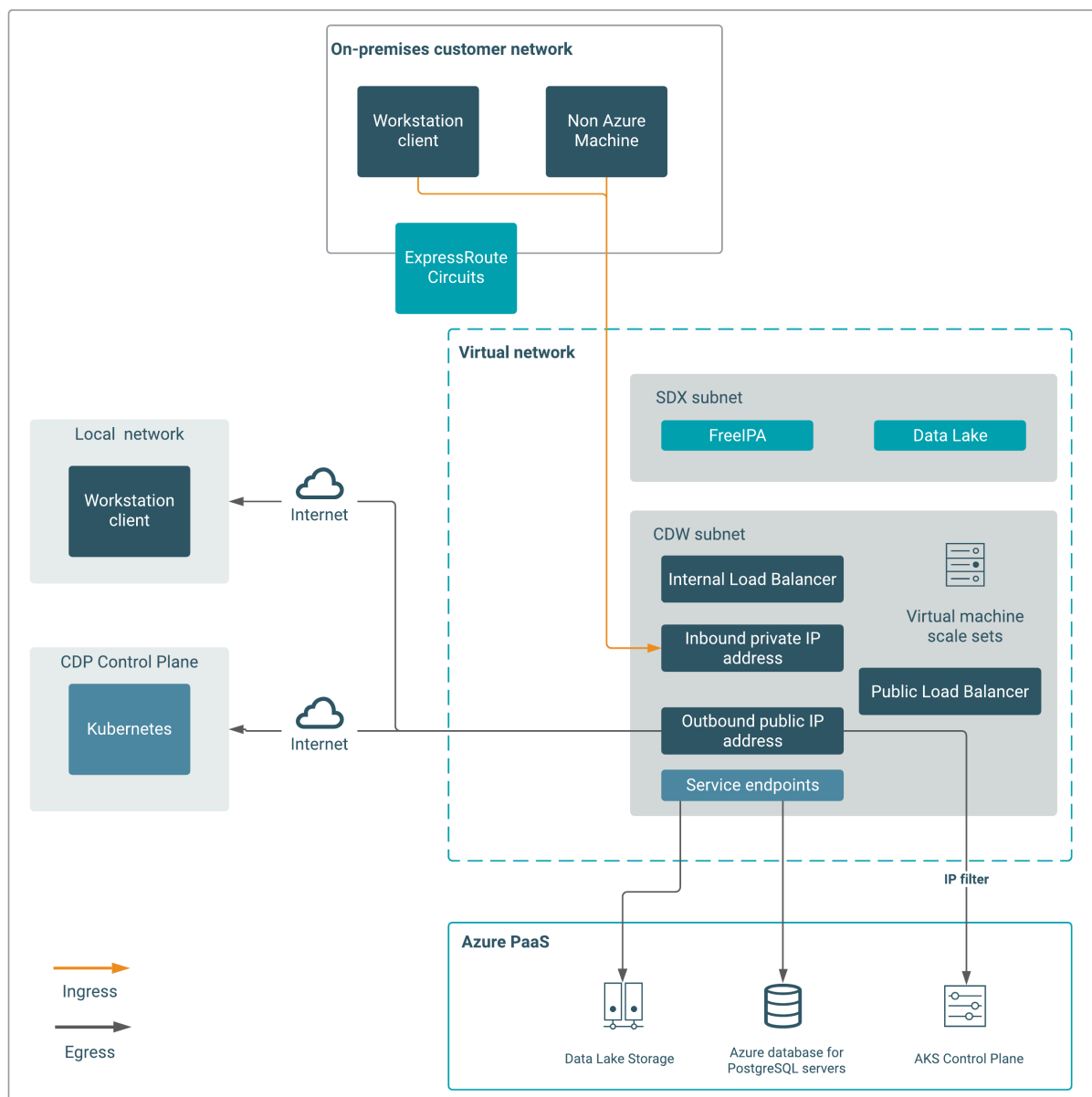


If you have defined custom UDRs to route the traffic to a firewall appliance in Azure or on-premises resources, then you cannot access the load balancer endpoints from the internet. Using a firewall with a UDR breaks the ingress setup due to asymmetric routing. In asymmetric routing, a packet takes one path to the destination and takes another path when returning to the source. The incoming load balancer traffic is received via its public IP address, but the return path goes through the firewall's private IP address. Using an internal load balancer solves the issue with asymmetric routing. The incoming packets arrive at the firewall's public IP address, get translated to the load balancer's private IP address, and then return to the firewall's private IP address using the same return path because the load balancer is deployed with a private frontend IP address.

Using the private (internal) load balancer

Internal load balancers distribute traffic inside a virtual network. Internal load balancers do not have a public IP address and cannot be reached from the public internet, even when you have configured UDRs. An internal load balancer makes a Kubernetes service accessible only to applications running in the same or peered Azure virtual network (VNET) as the Kubernetes cluster, or applications running in on-premises networks that are connected using Azure ExpressRoute circuits.

The following diagram shows ingress and egress traffic when a private (internal) load balancer is used:



For outbound connections, a public load balancer having a public IP address is used. For inbound connections, an internal load balancer having a private IP address is used. You cannot access the AKS endpoints from the public internet. Even with firewall publishing rules, public requests may fail due to certificate/DNS issues.

Related Information

[Enabling internal load balancer](#)

Enabling internal load balancer

To use an internal load balancer for Cloudera Data Warehouse (CDW), you must select the option to enable an internal load balancer while activating the Azure environment from the CDW UI. Otherwise, CDW uses the Standard public load balancer that is enabled by default when you provision an AKS cluster.

Procedure

1. In the CDW service, go to the Environments tab.
2. Locate the environment that you want to activate and click Activate.
The **Activation Settings** modal is displayed.
3. In the **Activation Settings** dialog box, select Enable internal load balancer (ingress).
4. Click Activate.

Granting remote access to Kubernetes clusters on Azure Kubernetes Service

You can remotely access Azure Kubernetes Service (AKS) clusters for troubleshooting, log collection, and maintenance purposes using SSH. To do that, you must add the Azure Active Directory (AD) object ID in the CDW environment Kubeconfig.

About this task

Required Role: DWAdmin

Before you begin

- You must have an active CDW environment to grant your users remote access to the Kubernetes cluster.
- Contact your Azure account administrator to obtain the Azure AD object ID.

Obtaining the Azure AD object ID using Azure CLI

You can obtain the object ID by running the following command:

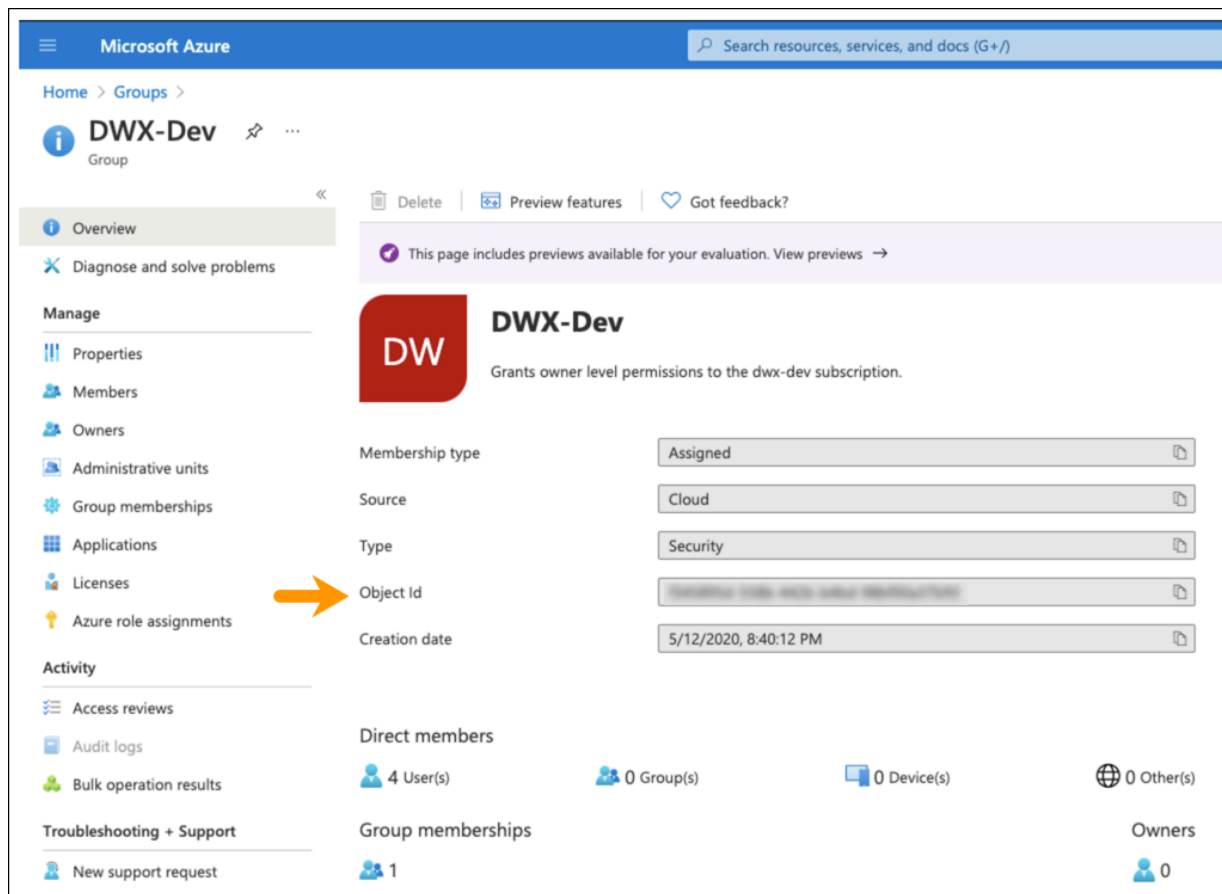
```
az ad group list --filter "displayname eq '[***GROUP-NAME***]'" -o table
```

This lists all the existing groups within the directory.

Obtaining the Azure AD object ID from the Azure portal

1. Sign in to the Azure portal using a Global administrator account for the directory.
2. Search for and select Azure Active Directory.
3. On the **Active Directory** page, select Groups and click Overview.

- Note the Object ID from the **Group Overview** page.



Procedure

- In the Data Warehouse service, go to the Environments tab.
- Locate the environment for which you want to grant access to AKS and click Edit GROUP ACCESS .
- Enter the Azure AD object ID in the Add new group text box.
- Click Grant Access to save your changes.

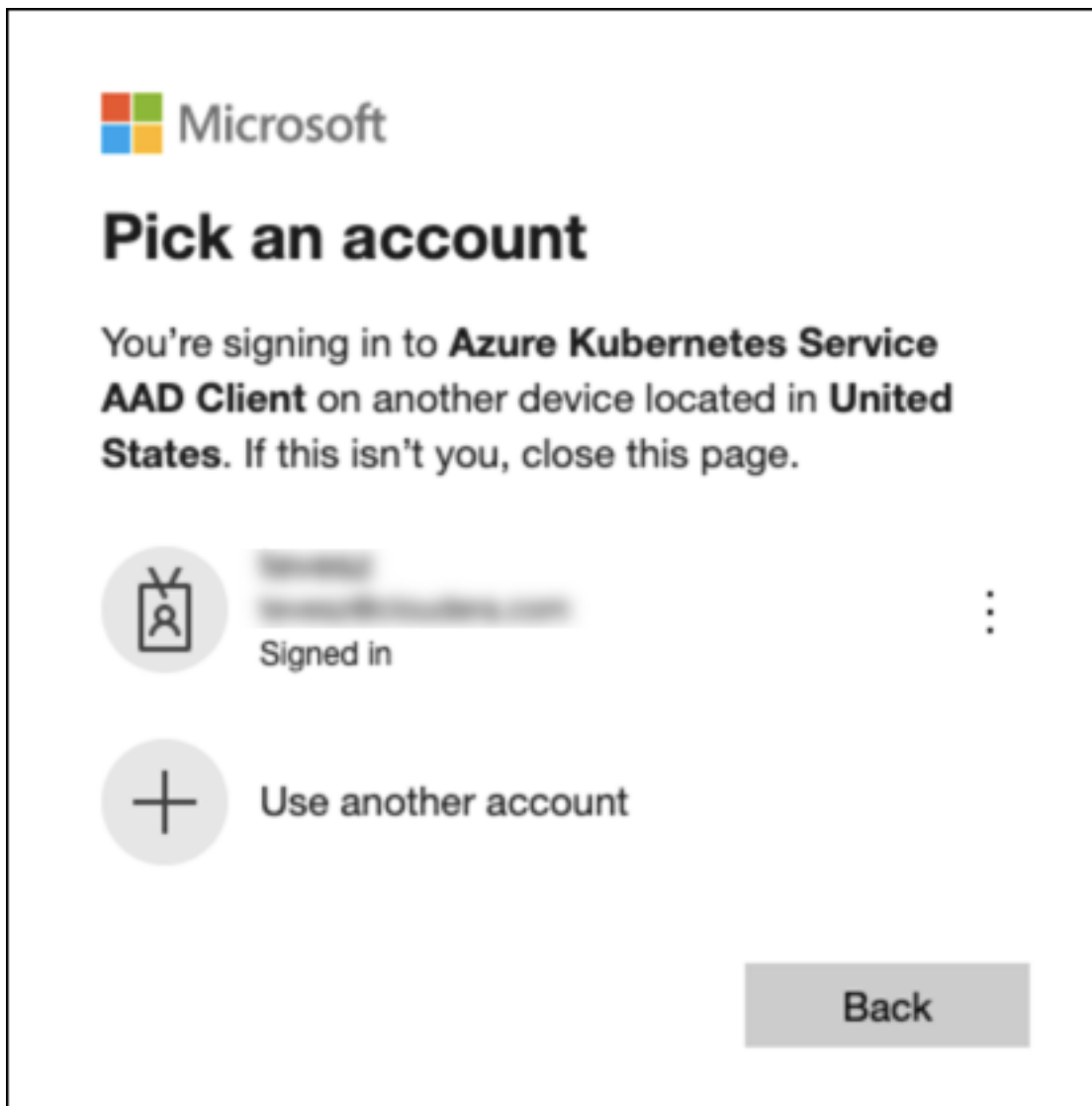
What to do next

Access the Virtual Machines remotely using Azure CLI. When you enter a kubectl command, such as, kubectl get pods Azure CLI, you see the following message:

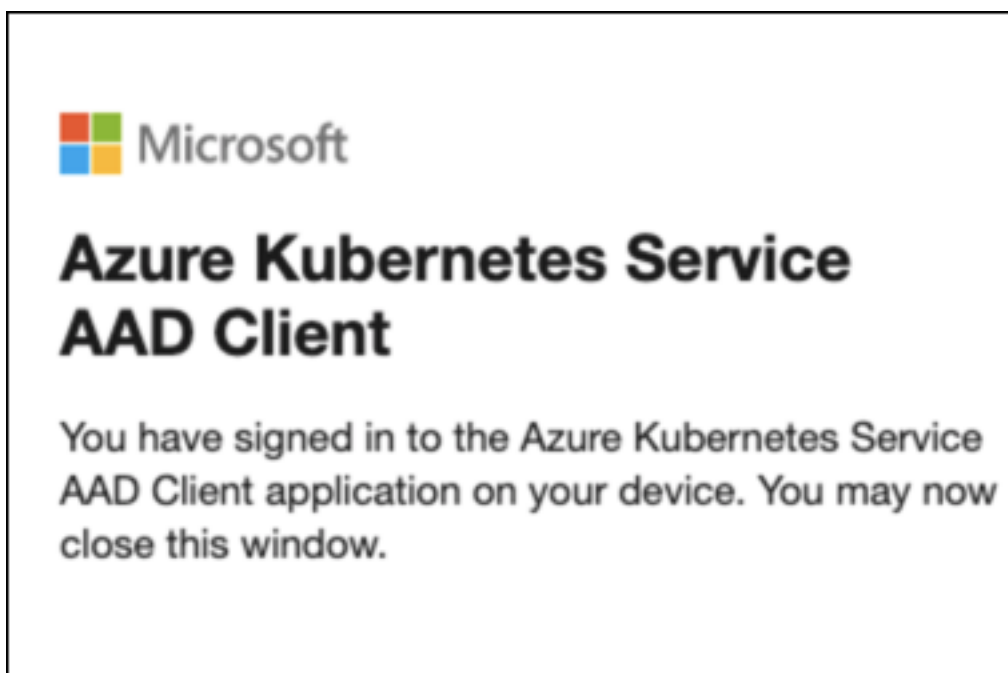
```
To sign in, use a web browser to open the page https://microsoft.com/
devicelogin and enter the code ***** to authenticate.
```

- Open a web browser and go to <https://microsoft.com/devicelogin>.
- Enter the code on the **Enter code** page and click Next.

3. On the **Pick an account** page, select the account for signing into the AKS AAD Client.



Upon successful login, you get the following message:



Revoking remote access to Kubernetes clusters on Azure Kubernetes Service

You can revoke remote access to Kubernetes clusters on Azure Kubernetes Service (AKS) by removing the Azure AD object ID from the CDW environment Kubeconfig.

About this task

Required Role: DWAdmin

Procedure

1. In the Data Warehouse service, go to the Environments tab.
2. Locate the environment for which you want to revoke access.
3. Click the options menu and select Show Kubeconfig.
4. Click the delete icon under Action to revoke access for the user.
5. Click Hide to close the dialog box.

Managed storage access

Understanding how Cloudera Data Warehouse (CDW) stores data for multiple tenants and a high-level overview of the configuration tasks prepares you as DWAdmin to set up a RAZ-controlled warehouse.

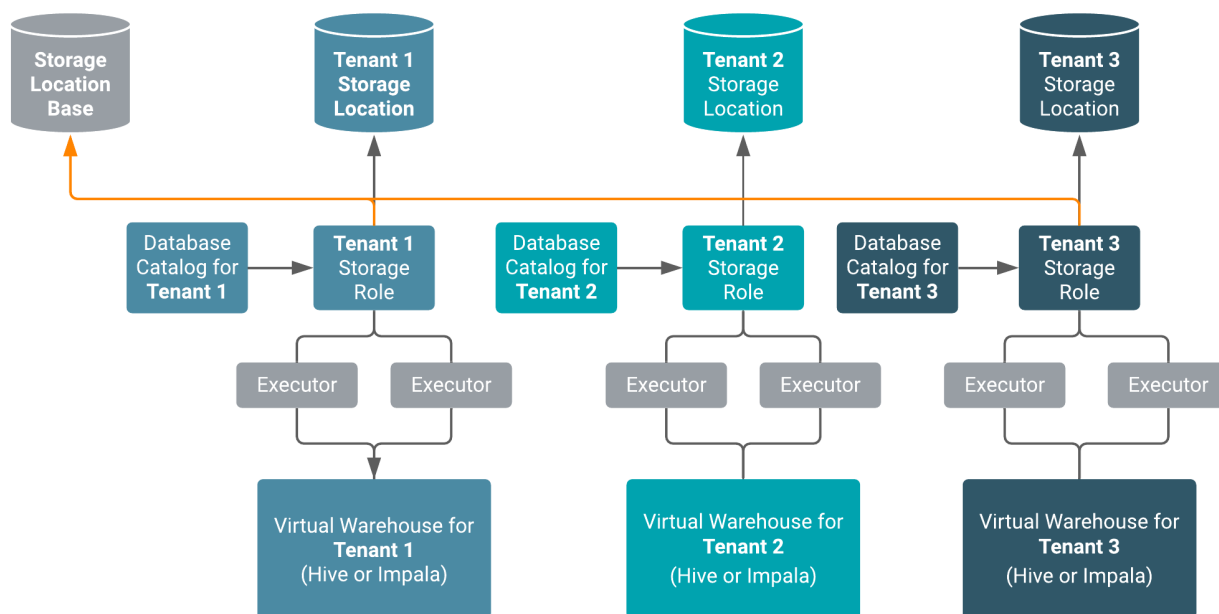
The multitenant storage technique in CDW requires a separate Database Catalog plus at least one Virtual Warehouse per tenant.

You configure a dedicated tenant storage role to give only the tenant-specific default database catalog instance and its associated virtual warehouse data access to both of the following buckets.

- The tenant storage location
- Potentially shared Storage Location Base

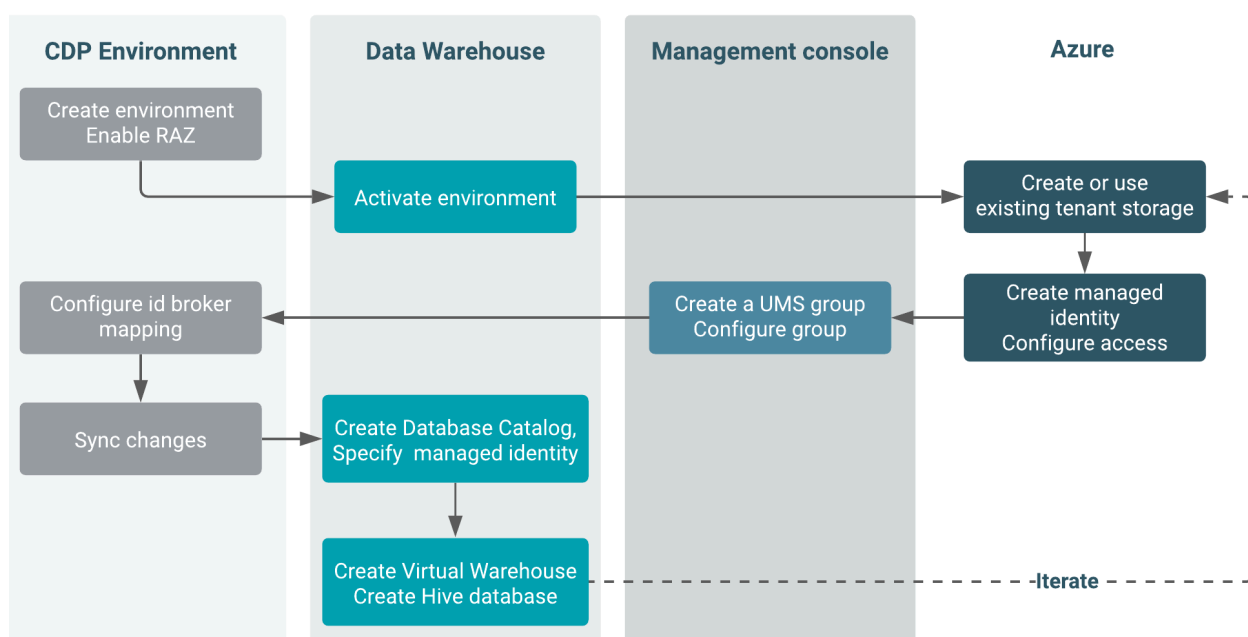
- The SDX Data Lake bucket

If the access token for a role is leaked, the data of others is not compromised; only one tenant's own, and shared, data is vulnerable. A managed identity IDBroker approach to accessing the Virtual Warehouse is shown in the following graphic:



Using a Ranger Remote Authorization (RAZ)-enabled environment, you can control access to data based on user roles and classifications. As a CDP Admin, you can apply Ranger fine-grained access control policies to cloud storage.

The following diagram shows the high-level steps for configuring RAZ-enabled storage:



After you obtain the entitlements required for this feature `CDW_ALLOW_MULTI_DEFAULT_DBC` and `CDW_STORAGE_ROLES`, you can configure storage as shown in this diagram:

- In CDP, register an environment that enables RAZ and uses the SDX Data Lake.
- Activate the environment in CDW.
- Manually create a tenant storage location(s), or use existing ones, for access by the tenant storage role.
- In Azure, create an Azure managed identity for the tenant.
- Assign the tenant specific Azure managed identity as a Storage Blob Data Owner role to the tenant-specific container; assign the tenant specific Azure managed identity as a Storage Blob Data Owner role to [Storage Location Base](#).
- In CDP Management Console, create an UMS group for this tenant.
- Add the UMS machine user (created when you activated the CDW environment) to the UMS group.
- Add the id broker mapping to the environment.
- Sync user group changes for the RAZ-enabled environment with FreeIPA.
- Create a separate Database Catalog with a unique managed identity created in the step above.
- Create a tenant-specific Virtual Warehouse based on the Database Catalog, SDX Data Lake, and RAZ-enabled environment.
- Create a tenant-specific Hive or Impala database that points to tenant storage locations.

As the metadata is shared across all tenants, Ranger grants access to tenant data via a group at the database level.

- For each tenant, repeat the actions above, starting from the step after activating the environment for CDW.

The managed identity accesses a tenant specific location and SDX Data Lake shared Storage Location Base. The Storage Location Base contains shared data, stores the Directed Acyclic Graph (DAG) data used by the Database Catalog, and provides integration with Cloudera Workload XM. WXM writes Impala query data to the shared Storage Location Base.

You need the UMS group to add the [IDBroker mapping](#), as a single UMS machine user cannot have multiple ID broker mappings to different managed identities.

The following topics describe step-by-step how to set up your Database Catalog, and Virtual Warehouse for storing RAZ-enabled data.

Setting up managed storage access

You learn the consequences of setting up RAZ control, which cannot be removed, and requirements you must meet to enable managed storage access.

About this task

The RAZ-controlled warehouse supports only multiple Shared Data Experience (SDX) Database Catalogs instead of CDW-specific, isolated database catalogs. In a RAZ-controlled environment, the Elastic Kubernetes Service (EKS) ec2 executor will not have read/write permissions to the storage location. Consequently, after activating the CDW environment, you cannot remove RAZ control. RAZ-control of CDW continues during upgrades of the Database Catalog and Virtual Warehouse.

Before you begin

- Request activation of the following entitlements:
 - CDW_ALLOW_MULTI_DEFAULT_DBC
 - CDW_STORAGE_ROLES
- You meet the requirements described in the [Azure requirements documentation](#).

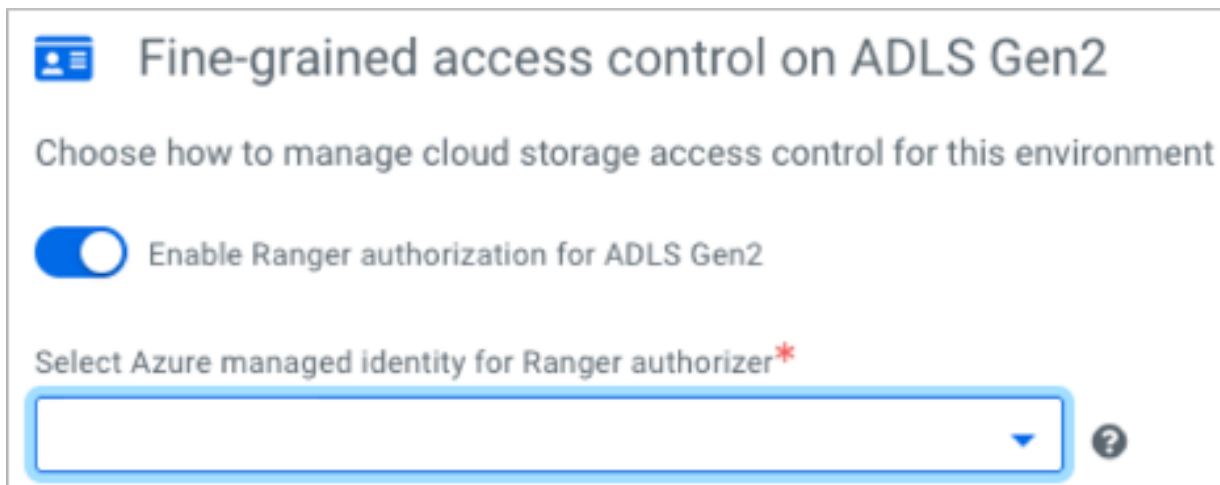
Required role: PowerUser

Creating the CDP environment

You follow steps to register and activate a new environment to enable RAZ in CDW.

Procedure

1. [Register an environment with RAZ](#) using the CDP web interface.
In the web interface, in Fine-grained access control, select Enable Ranger Authorization.



2. In Cloudera Data Warehouse Overview, locate the RAZ-enabled environment, and click Start to activate the environment..
This action disables the standard default Database Catalog that is automatically created after activation. The UMS machine user is created and attached to the environment when you activate the CDW environment. Later, you see how to add this same UMS machine user to a different UMS group for each tenant.
3. In Activate Environment, select Enable Storage Roles.
Unchecking Enable Storage Roles disables the metadata proxy associated with managed storage access.
Repeat the following steps for each tenant:
4. In Azure, manually create a one or more storage locations for the tenant, or use existing storage locations.
5. In Azure, [create a managed identity](#) that has the role Storage Blob Data Owner to access tenant specific container and across all tenants shared, Storage Location Base in the SDX Data Lake.
For example, create a managed identity named tenant-1 and a container called container-tenant-1.

Creating a UMS group and machine users

This procedure ensures that the [CDP machine user](#) gets permission to access the tenant bucket.

Procedure

Repeat the following steps for each tenant:

1. In Management Console, User Management Groups , click CREATE GROUP and [create a User Management Service \(UMS\) group](#), for example group-tenant-1.

2. In Groups Members , search for and select your `srv_machine_<env id>_storage_role` to [add this UMS machine user](#) to group-tenant-1.

3. In Management Console Environments , select an environment, and click Actions Manage Access IDBroker Mappings Edit .
4. Click + to add a mapping, select the Group-tenant-1 and Role-tenant-1, and specify the complete managed identity. For example: `/subscriptions/<subscription_id>/resourcegroups/<resource_group>providers/Microsoft.ManagedIdentity/userAssignedIdentities/<managed-identity-name>`
5. Synchronize your group changes with FreeIPA by [performing a user sync](#) per environment: In the RAZ-enabled environment, click Actions Synchronize Users to FreeIPA.
The UMS machine user gets the permission to access the tenant-specific container.

Creating a new Database Catalog

You repeat a step-by-step procedure to set up a new Database Catalog for each tenant. You must correctly enter your managed identity regular expression. You see how to recognize a discrepancy in your entry and your actual managed identity.

About this task

When you create a new database catalog, you specify the managed identity you created earlier. In the event of an invalid managed identity specification, the following regular expression appears in the UI.

```
'\subscriptions\/(.+?)\/resourcegroups\/(.+?)\/providers\/Microsoft.ManagedIdentity\/userAssignedIdentities\/(.+)', 'i'
```

An example that matches the regular expression is:

```
/subscriptions/<subscription_id>/resourcegroups/<resource_group>providers/Microsoft.ManagedIdentity/userAssignedIdentities/<managed-identity-name>
```

- subscriptions: Your subscription ID
- providers: Microsoft.ManagedIdentity (required value)
- resourcegroups: Your resource group
- userAssignedIdentities: Your managed identity name.

Procedure

Repeat the following steps for each tenant.

1. Click **Data Warehouse Database Catalog Create Database Catalog**.
2. In **New Database Catalog**, enter the complete description of the managed identity you created earlier.
3. Select the RAZ-enabled environment.

In Data Lake, SDX is the required value. The backend Data Lake and Database Catalog database must be the same.

The backend Data Lake and Database Catalog database must be the same.

4. In **Tenant Storage Role**, enter the complete managed identity you obtained earlier.
For example:

```
For example:  
/subscriptions/<subscription_id>/resourcegroups/<resource_group>providers/  
Microsoft.ManagedIdentity/userAssignedIdentities/<managed-identity-name>
```

CDW attempts to validate your managed identity. If successful, proceed. If validation fails, you see the error message described above. Correct the problem, and try again.

5. In **Tenant Storage Location**, enter the tenant-specific container container-tenant-1, for example, and click **CREATE**.

Creating a tenant-specific Virtual Warehouse

You follow a step-by-step procedure to create a Virtual Warehouse based on the Database Catalog and SDX Data Lake you created in the RAZ-enabled environment.

About this task

The Database Catalog and Data Lake point to the same backend database. In the tenant-specific Virtual Warehouse, you create a tenant-specific Hive or Impala database that points to tenant-specific buckets. As the metadata is shared across all tenants, Ranger grants access to tenant data at the table level. One or more tenant-specific databases alongside databases for shared data can run in the same HMS instance.

Procedure

Repeat the following steps for each tenant.

1. In the Data Warehouse service, click **Virtual Warehouses New Virtual Warehouse**.
2. Specify a name, select either the Hive or Impala type, and select the Database Catalog you created for the tenant.
3. In Overview, find your Virtual Warehouse, click Hue.
4. Create a tenant-specific Hive or Impala database where the location for external and managed tables are pointing to the tenant-specific buckets.

```
CREATE (DATABASE|SCHEMA) [IF NOT EXISTS] database_name
[COMMENT database_comment]
[LOCATION external_table_path]
[MANAGEDLOCATION managed_table_directory_path]
[WITH DBPROPERTIES (property_name=property_value, ...)];
```

Do not set LOCATION and MANAGEDLOCATION to the same path. For more information, see [Create a default directory for managed tables](#).

5. In Ranger, grant the tenant users access to the tenant-specific Hive or Impala database.

Azure Kubernetes Service upgrade

Azure Kubernetes Service (AKS) cluster requires regular updates to the Kubernetes versions. Using the latest AKS version supported by Cloudera avoids compatibility issues between Cloudera Data Warehouse (CDW) and Azure resources.

Using the Azure CLI or Azure portal to upgrade the AKS cluster is not supported. CDW automatically provisions the latest supported AKS version when you activate an environment in CDW. AKS 1.29 is provisioned automatically when you [activate your environment from CDW](#) using release 1.9.1-b233 (released July 26, 2024) and onwards.

To upgrade to the latest AKS, you must deactivate and reactivate your CDW environment. To restore the state of your CDW cluster after reactivation, Cloudera recommends using the backup-restore functionality. For more information, see [Backing up and restoring CDW](#).

Setting scratch space limit for spilling Impala queries in Azure environments

Certain memory-intensive SQL operations write temporary data to disk (known as spilling to disk) when Impala is close to exceeding its memory limit on a particular host. Consequently, a query can complete successfully instead of failing with an out-of-memory error. You must set the scratch space limit carefully because you cannot change the limit later.

About this task

The compute nodes on Azure environments use the STANDARD_E16DS_v4 or STANDARD_E16DS_V5 memory-optimized Virtual Machines by default. These nodes have a temporary SSD storage of 2 * 300 GiB (600 GiB). This space is utilized for both cached data and scratch space. The scratch space available is insufficient for the spilling Impala queries.

Spilling queries to disk require a scratch space in multiples of several 100 GiBs or TiBs, without which the queries may fail. To prevent failure, you can configure the scratch space between 300 GiB and 16 TiB per executor node while creating an Impala Virtual Warehouse. In addition to the local SSD, you can use managed disks (Standard SSD and Premium disks) for extra space.

In CDW, the range of options for scratch space using the combination of the local SSDs and managed disks has been tailored to optimize the cost-to-space ratio. The following table shows the pricing and throughput information for the available scratch space options:

Table 2:

Disks (SSD + managed disk)	Size (GiB)	Throughput (MB/s)	Price (\$/hour)
SSD only	300	968 cached / 384 uncached	0
SSD + 3 * P10	684	300 (3 * 100)	0.073 (3 * 0.024)
SSD + 2 * P15	812	250 (2 * 125)	0.094 (2 * 0.047)
SSD + 3 * P15	1068	375 (3 * 125)	0.141 (3 * 0.047)
SSD + 2 * P20	1324	300 (2 * 150)	0.182 (2 * 0.091)
SSD + 3 * P20	1836	450 (3 * 150)	0.273 (3 * 0.091)
SSD + P40	2348	250	0.322
SSD + 3 * P30	3372	600 (3 * 200)	0.504 (3 * 0.168)
SSD + P50	4396	250	0.617
SSD + E60	8492	400	0.841
SSD + E70	16684	600	1.683

You can select the scratch space limit only while creating a new Impala Virtual Warehouse.

Before you begin

- Obtain the CDW_IMPALA_EBS_SCRATCH_SPACE entitlement.
- Activate the environment in CDW using the E16ds_v4 or E16ds_v5.
- Disable Availability Zones for AKS during the environment activation; otherwise, you cannot set the scratch space limit.

Procedure

1. Log in to Cloudera Data Warehouse.
2. In the Data Warehouse service, click Virtual Warehouses in the left navigation panel.
3. On the **Virtual Warehouses** page, click New Virtual Warehouse.
4. Specify a Name.
5. Select Virtual Warehouse type as IMPALA.
6. Select the Database Catalog that it queries.
7. Select a Size.

Additional configuration options are displayed along with Scratch Space Limit per node (in GiBs).

8. Select the scratch space from the Scratch Space Limit per node (in GiBs) dropdown.
9. Click Create Virtual Warehouse.

Related Information

[Azure Standard_E16ds_v4 instances](#)

[Azure Standard_E16_v3 instances](#)

[Adding a new Virtual Warehouse](#)

[Activating Azure environments](#)