

Quickstart

Date published: 2021-04-06

Date modified: 2024-06-03

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

Cloudera DataFlow Quickstart.....	4
Verify cloud infrastructure prerequisites.....	6
AWS quickstart.....	6
Verify AWS prerequisites.....	6
Create a CDP credential.....	7
Register a CDP environment.....	8
Azure quickstart.....	15
Verify Azure prerequisites.....	16
Create an Azure AD app.....	17
Deploy the template.....	18
Assign roles.....	19
Create or locate an SSH Key.....	20
Create a CDP credential.....	20
Register a CDP environment.....	21
Give administrators access.....	25
Enable Cloudera DataFlow for your environment.....	26
Give users access.....	27
Add the Hello World ReadyFlow definition to the Catalog.....	28
Deploy the Hello World ReadyFlow using the deployment wizard.....	30

Cloudera DataFlow Quickstart

Get started with Cloudera DataFlow quickly by walking through a few simple steps. Ensure that administrators have access to Cloudera Public Cloud, enable Cloudera DataFlow for an environment, give users access to Cloudera DataFlow, and then add to the Catalog and deploy the Hello World ReadyFlow.

Cloudera DataFlow

Quickstart

1



[Verify infrastructure prerequisites](#)

2



[Provide access for administrators](#)

3



[Enable CDF for your environment](#)

4



[Provide access for users](#)

5



[Add to Catalog](#)

6

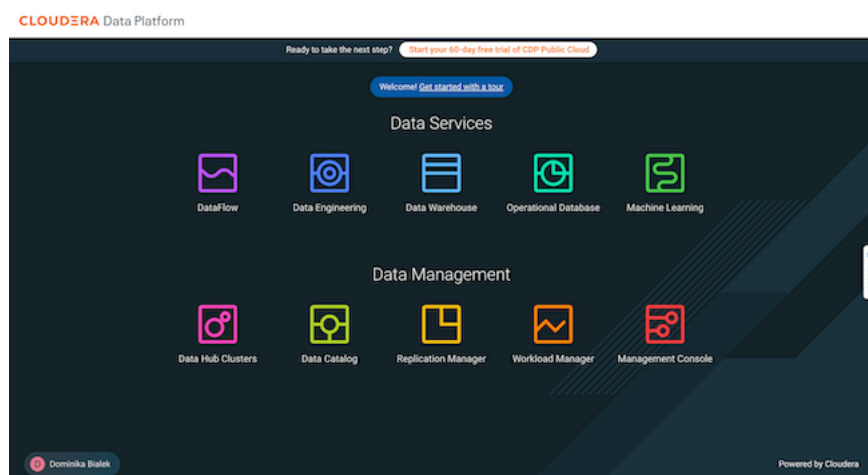
[Deploy a flow definition](#)

Verify cloud infrastructure prerequisites

As the administrator for your AWS environment, ensure that the environment meets the requirements for Cloudera Public Cloud and Cloudera DataFlow.

AWS quickstart

If you've reached the CDP landing page for the first time, you've come to the right place! In this quickstart, we'll show you step-by-step how to connect CDP to your AWS account, so that you can begin to provision clusters and workloads.



To complete this quickstart, you'll need access to two things:

- The CDP console pictured above
- The AWS console



Note: This AWS onboarding quickstart is intended for simple CDP evaluation deployments only. It may not work for scenarios where AWS resources such as VPC, security group, storage accounts, and so on, are pre-created or AWS accounts have restrictions in place.

The steps that we will perform are:

Step 0: Verify the AWS prerequisites

Step 1: Create a provisioning credential

Step 2: Register an AWS environment in CDP

Verify AWS cloud platform prerequisites

Before getting started with the AWS onboarding quickstart, review and acknowledge the following:

- This AWS onboarding quickstart is intended for simple CDP evaluation deployments only. It may not work for scenarios where AWS resources such as VPC, security group, storage accounts, and so on, are pre-created or AWS accounts have restrictions in place.
- Users running the AWS onboarding quickstart should have:
 - AWS Administrator permissions on the AWS account that you would like to use for CDP.
 - Rights to create AWS resources required by CDP. See list of [AWS resources used by CDP](#).
 - CDP Admin role or Power User role in CDP subscription.
- This AWS onboarding quickstart uses a CloudFormation template that automatically creates the required resources such as buckets, IAM roles and policies, and so on.

- CDP Public Cloud relies on several AWS services that should be available and enabled in your region of choice. Verify if you have enough quota for each AWS service to set up CDP in your AWS account. See list of [AWS resources used by CDP](#).

If you have more complex requirements than those listed here, contact Cloudera Sales Team to help you with the CDP onboarding.

Create a CDP credential

In the CDP console, the first step is to create a CDP credential. The CDP credential is the mechanism that allows CDP to create resources inside of your cloud account.

Procedure

1. Log in to the CDP web interface.
2. From the CDP home screen, click the Management Console icon.
3. In the Management Console, select Shared Resources > Credentials from the navigation pane.
4. Click Create Credential.
5. Click the Copy icon to the right of the **Create Cross-account Access Policy** text box.

Create Credential

aws

Name *

Enter credential name

Description

Enter description

☐ Enable Permission Verification

Create Cross-account Access Policy

Copy the following JSON to create an AWS IAM policy

Default Minimal

The default role allows for the default set of operations including everything that the minimal role allows for.

```
{
  "Statement": [
    {
      "Sid": "CloudFormationFull",
      "Action": [
        "cloudformation:*"
      ],
      "Effect": "Allow",
      "Resource": [
        "*"
      ]
    }
  ]
}
```

6. In a second browser tab, open the **AWS Console** and navigate to Identity and Access Management Policies. Click Create Policy.

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Account settings

IAM > Policies

Policies (952)

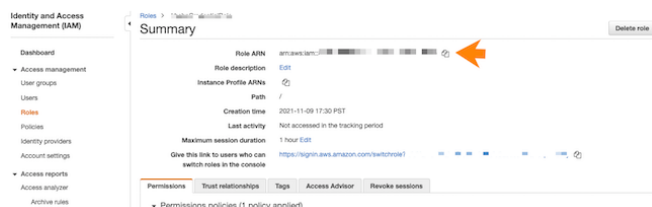
A policy is an object in AWS that defines permissions.

Filter policies by property or policy name and press enter

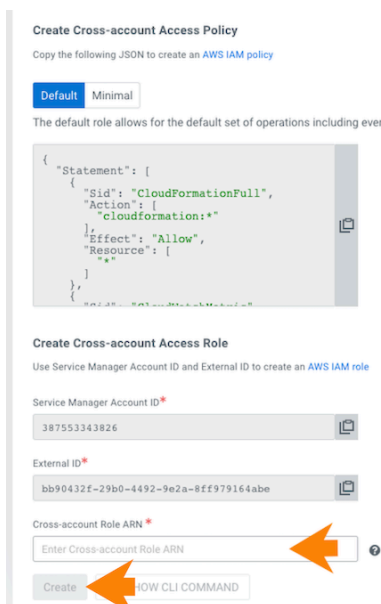
Policy name	Type	Used as	Description
AmazonSageMaker-ExecutionPolicy-20200218T124050	Customer managed	Permissions policy (1)	
aws-logs-policy	Customer managed	Permissions policy (1)	AWS - CDP Standardize
cdp_permissions	Customer managed	Permissions policy (1)	Incremental Permissions
demo-aws-gm-bucket-policy-s3access	Customer managed	None	

7. Click on the JSON tab and paste the access policy in the text box.
You may get a warning related to using wildcards. You may ignore it and proceed to the next step.
8. Click Next: Tags.
9. Click Review Policy.
10. Give the policy a unique name and a description.
11. Click Create Policy.
Next, you create the required cross-account role.
12. In the AWS console, navigate back to Identity and Access Management.
13. Click Roles>Create Role.

14. Under **Select type of trusted entity**, select Another AWS account.
15. Return to the CDP Management Console and copy the contents of the Service Manager Account ID field on the **Credentials** page.
16. In the AWS console, paste the Service Manager Account ID into the Account ID field.
17. Return to the CDP Management Console and copy the contents of the External ID field on the **Credentials** page.
18. In the AWS console, check the "Require external ID" options box, and then paste the External ID copied from CDP into the External ID field.
19. Click Permissions and select the checkbox next to the name of the policy that you created in Step 8.
20. Click Next: Tags.
21. Click Next: Review.
22. Give the role a unique name and description, then click Create Role.
23. Still in the role page of the AWS console, search for the role you just created, and click on it.
24. Copy the Role ARN at the top of the **Summary** page.



25. Return to the **Credentials** page in the CDP Management Console.
26. Give the CDP credential a name and description. The name can be any valid name.
27. Paste the Role ARN that you copied from the AWS console into the Cross-account Role ARN field, then click Create.



Now that you've created a cross-account role, proceed to creating a CDP environment.

Register a CDP environment

Before you register an environment, you'll want to create specific IAM roles and policies so that CDP can operate in a secure manner.

About this task

For background information, a description of what we're building and why can found [here](#). For this quickstart, we'll use CloudFormation to set all of this up for you.

Procedure

1. Download the CloudFormation provided template [here](#).

2. In the AWS console, deploy the CloudFormation template:
 - a) In **AWS Services**, search for CloudFormation.
 - b) Click Create Stack and select With new resources (standard).
 - c) Select Template is ready and then Upload a template file.

The screenshot shows the 'Create stack' wizard in the AWS console, specifically the 'Specify template' step. The 'Prerequisite - Prepare template' section is at the top, with three radio buttons: 'Template is ready' (selected), 'Use a sample template', and 'Create template in Designer'. Below this, the 'Specify template' section explains that a template is a JSON or YAML file. It has two radio buttons: 'Amazon S3 URL' and 'Upload a template file' (selected). Under 'Upload a template file', there is a 'Choose file' button and a text input field containing 'setup.json'. Below the input field, it says 'JSON or YAML formatted file'. At the bottom, an 'S3 URL' is displayed: 'https://s3.us-east-2.amazonaws.com/cf-templates-11mg5w1r6abx5-us-east-2/2020104Hxs-setup.json'. There is a 'View in Designer' button next to the URL. At the very bottom of the wizard, there are 'Cancel' and 'Next' buttons.

- d) Click Choose file and select the CloudFormation template that you downloaded.
- e) Click Next.
- f) Under Stack name, enter a stack name. The name can be any valid name.
- g) Under **Parameters**, complete the following fields:
 - BackupLocationBase: Choose an unused bucket name and path for the FreeIPA backups. CDP will be creating the bucket for you. The same bucket can be used for BackupLocationBase, LogsLocationBase, and StorageLocationBase. By default this is set to my-bucket/my-backups.
 - CrossAccountARN: Do not change the default value. This parameter is only required when encryption is enabled, but since in this quickstart we do not enable encryption, you should leave this value as is.
 - LogsLocationBase: Choose an unused bucket name and path for the logs. CDP will be creating the bucket for you. The same bucket can be used for BackupLocationBase, LogsLocationBase, and StorageLocationBase. By default this is set to my-bucket/my-logs.
 - StorageLocationBase: Choose an unused bucket name and path for the data. CDP will be creating the bucket for you. The same bucket can be used for BackupLocationBase, LogsLocationBase, and StorageLocationBase. By default this is set to my-bucket/my-data.
 - Prefix: A short prefix of your choosing, which will be added to the names of the IAM resources CDP will be creating. We chose "cloudera" as an example.
 - s3KmsEncryption: Encryption will be disabled for the created bucket. You don't need to change this value.

For example:

Specify stack details

Stack name

Stack name

mc-cdp-stack

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

BackupLocationBase

The storage base path to create an S3 bucket with default encryption for CDP. By default CDP will create the optional subdirectory in the bucket. It is possible to either use the same bucket or different buckets for StorageLocationBase and LogsLocationBase.

my-bucket/my-backups

CrossAccountARN

Required if s3 KMS Encryption is selected

arn:aws:iam::<ACCT_ID>:role/<ROLE_NAME>

LogsLocationBase

The storage base path to create an S3 bucket with default encryption for CDP. By default CDP will create the optional subdirectory in the bucket. It is possible to either use the same bucket or different buckets for StorageLocationBase and LogsLocationBase.

my-bucket/my-logs

StorageLocationBase

The logging base path to create an S3 bucket with default encryption for CDP. By default CDP will create the optional subdirectory in the bucket. It is possible to either use the same bucket or different buckets for StorageLocationBase and LogsLocationBase.

my-bucket/my-data

prefix

prefix for IAM objects, separated by a dash.

cloudera

s3KmsEncryption

If set to True S3 will be configured with AWS managed KMS server side encryption

false

Make a note of the BackupLocationBase, LogsLocationBase, StorageLocationBase, and Prefix that you define. You will need them later.

- h) Click Next.
- i) At the **Configure Stack Options** page, click Next.
- j) At the bottom of the **Review** page, under Capabilities, click the checkbox next to I acknowledge that AWS CloudFormation might create IAM resources with custom names, as that is exactly what we will be doing.

Capabilities

The following resource(s) require capabilities: [AWS::IAM::ManagedPolicy]

This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. [Learn more](#)

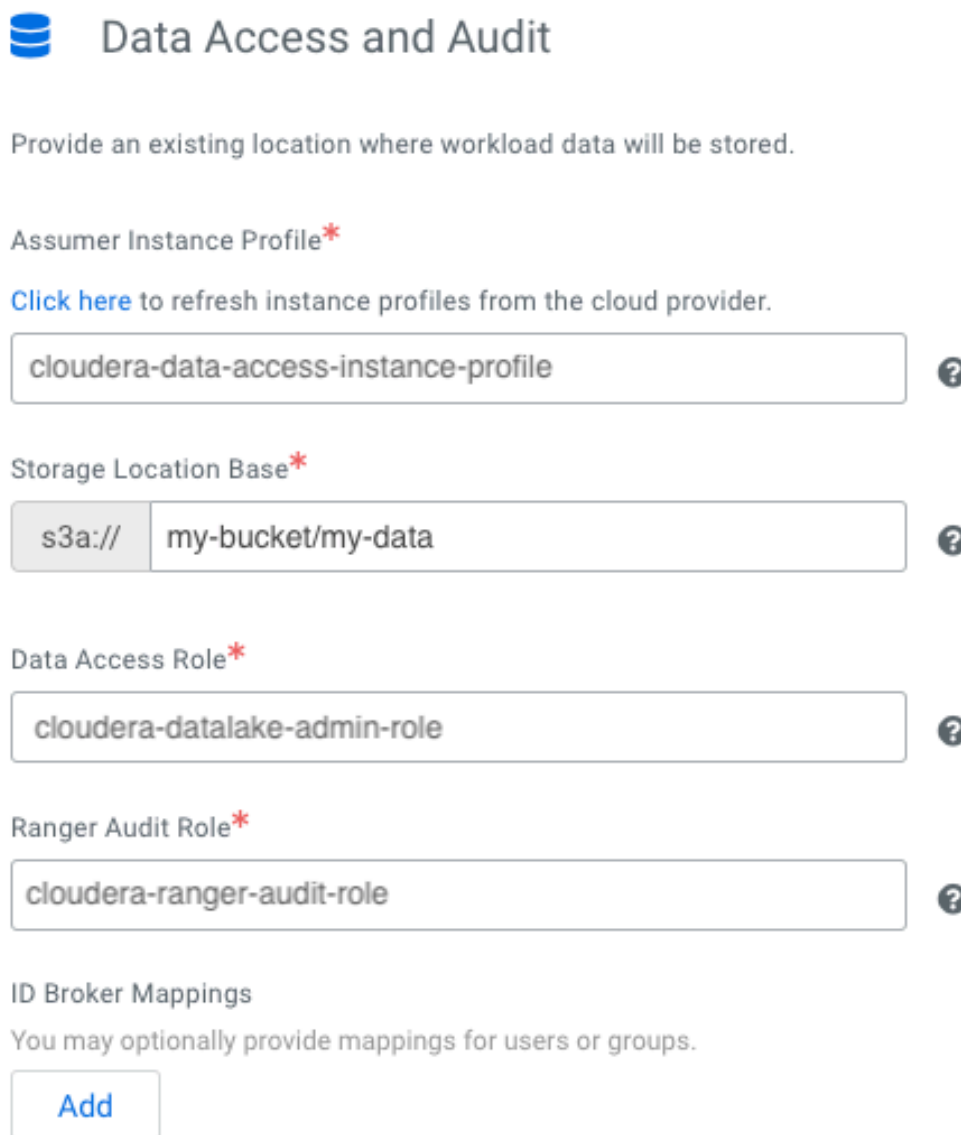
☒ I acknowledge that AWS CloudFormation might create IAM resources with custom names.

Cancel Previous Create change set **Create stack**

- k) Click Create Stack.
3. Still in the AWS console, create an SSH key in the region of your choice. If there is already an SSH key in your preferred region that you'd like to use, you can skip these steps.
 - a) In **AWS Services**, search for EC2.
 - b) In the top right corner, verify that you are in your preferred region.
 - c) On the left hand navigation bar, choose Key Pairs.
 - d) On the top right of the screen, select Create Key Pair.
 - e) Provide a name. The name can be any valid name.
 - f) Choose RSA type, and then choose the pem format.
 - g) Click Create key pair.
4. Return to the CDP Management Console and navigate to EnvironmentsRegister Environments.
5. Provide an environment name and description. The name can be any valid name.
6. Choose Amazon as the cloud provider.

7. Under **Amazon Web Services Credential**, choose the credential that you created earlier.
8. Click Next.
9. Under **Data Lake Settings**, give your new data lake a name. The name can be any valid name. Choose the latest data lake version.
10. Under **Data Access and Audit**:
 - Choose prefix-data-access-instance-profile>
 - For Storage Location Base, choose the StorageLocationBase from the cloud formation template.
 - For Data Access Role, choose prefix-datalake-admin-role.
 - For Ranger Audit Role, choose prefix-ranger-audit-role, where "prefix" is the prefix you defined in the **Parameters** section of the stack details in AWS.

For example:



Data Access and Audit

Provide an existing location where workload data will be stored.

Assumer Instance Profile*

[Click here](#) to refresh instance profiles from the cloud provider.

cloudera-data-access-instance-profile ?

Storage Location Base*

s3a:// my-bucket/my-data ?

Data Access Role*

cloudera-datalake-admin-role ?

Ranger Audit Role*

cloudera-ranger-audit-role ?

ID Broker Mappings

You may optionally provide mappings for users or groups.

Add

11. For Data Lake **Scale**, choose Light Duty.
12. Click Next.
13. Under Select Region, choose your desired region. This should be the same region you created an SSH key in previously.

14. Under **Select Network**, choose Create New Network.

15. Create private subnets should be enabled by default. If it isn't, enable it.



Note:

By enabling private subnets you will not have SSH access to cluster nodes unless you have access to the VPC.

16. Click the toggle button to enable Enable Public Endpoint Access Gateway.

For example:

Region, Location

Select Region

US West (Oregon) - us-west-2

Network

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

Select Network

Create new network

Network CIDR*

10.10.0.0/16

Create private subnets

Create Private Endpoints

⚠ Typical NAT gateway charges will be applied on your account, see AWS pricing for [more details](#)

Enable Public Endpoint Access Gateway

17. Under **Security Access Settings**, choose Create New Security Groups.

18. Under **SSH Settings**, choose the SSH key you created earlier.

For example:

The screenshot shows the AWS IAM console interface. The top section is titled "Security Access Settings" with a shield icon. It contains a dropdown menu labeled "Select Security Access Type" with the option "Create New Security Groups" selected, and a text input field labeled "Access CIDR*" with the value "0.0.0.0/0". Below this is the "SSH Settings" section with a key icon. It includes instructions to paste an SSH public key or select an existing one from the EC2 console. Two radio buttons are present: "New SSH public key" (unselected) and "Existing SSH public key" (selected). Under the "Existing SSH public key" option, there is a text input field labeled "Name of an existing SSH key pair to use for accessing cluster node instances." with the value "docs-test" entered.

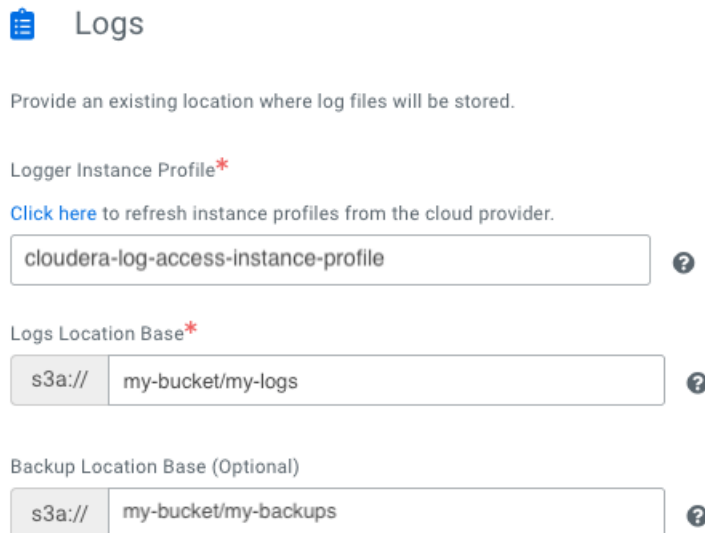
19. Optionally, under **Add Tags**, provide any tags that you'd like the resources to be tagged with in your AWS account.

20. Click Next.

21. Under Logs:

- Choose the Instance Profile titled prefix-log-access-instance-profile, where "prefix" is the prefix you defined in the **Parameters** section of the stack details in AWS.
- For Logs Location Base, choose the LogsLocationBase from the CloudFormation template.
- For Backup Location Base, choose the BackupLocationBase from the CloudFormation template.

For example, using the parameters we defined earlier:

 **Logs**

Provide an existing location where log files will be stored.

Logger Instance Profile*

[Click here](#) to refresh instance profiles from the cloud provider.

?

Logs Location Base*

?

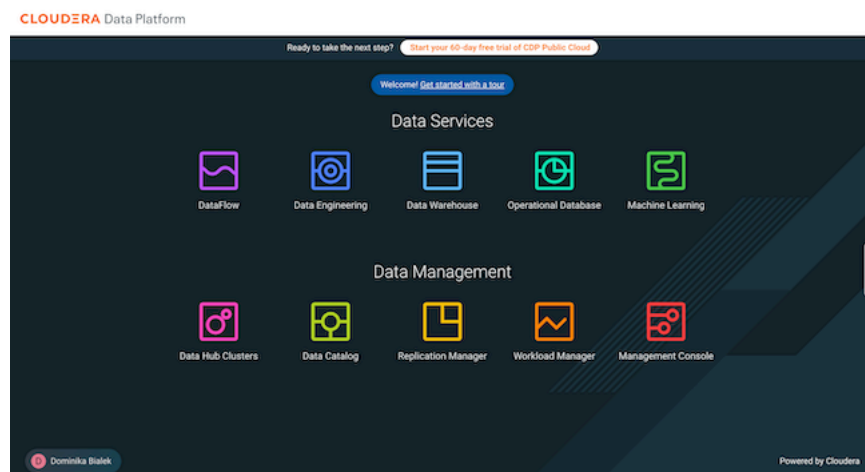
Backup Location Base (Optional)

?

22. Click Register Environment.

Azure quickstart

If you've reached the CDP landing page for the first time, you've come to the right place! In this quickstart, we'll show you step-by-step how to connect CDP to your Azure subscription, so that you can begin to provision clusters and workloads.



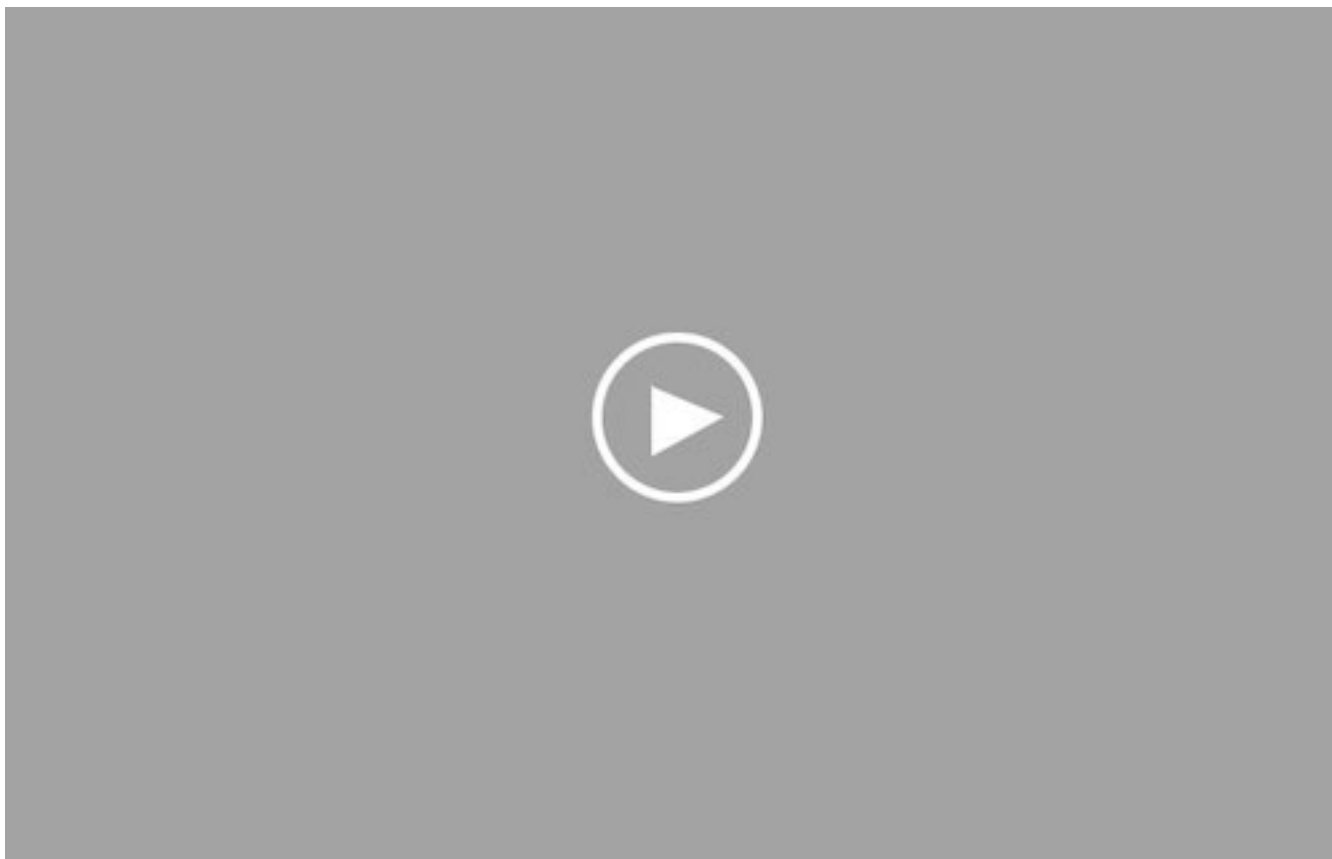
To complete this quickstart, you'll need access to three things:

- The CDP console pictured above
- The Azure console
- Azure Cloud shell



Note: This Azure onboarding quickstart is intended for simple CDP evaluation deployments only. It may not work for scenarios where Azure resources such as VNet, security group, storage accounts, and so on, are pre-created or Azure accounts have restrictions in place.

In addition to this documentation, you can refer to the following video:



The steps that we will perform are:

Step 0: Verify the Azure prerequisites

Step 1: Create an Azure AD app

Step 2: Deploy the Azure quickstart template

Step 3: Assign roles

Step 4: Create or locate an SSH key

Step 5: Create a CDP credential

Step 6: Register a CDP environment

Verify Azure cloud platform prerequisites

Before getting started with the Azure onboarding quickstart, review and acknowledge the following:

- This Azure onboarding quickstart is intended for simple CDP evaluation deployments only. It may not work for scenarios where Azure resources such as VNet, security group, storage accounts, and so on, are pre-created or Azure accounts have restrictions in place.
- User running the Azure onboarding quickstart should have:
 - Owner permissions on the Azure subscription that you would like to use for CDP.
 - Rights to create Azure resources required by CDP. See list of [Azure resources used by CDP](#).
 - Rights to create an Azure AD application (service principal) and assign Contributor role at subscription level.
 - CDP Admin role or Power User role in CDP subscription.

- This Azure onboarding quickstart uses an Azure ARM template that automatically creates the required resources such as storage accounts, containers, managed identities, resource groups, and so on.
- CDP Public Cloud relies on several Azure services that should be available and enabled in your region of choice. Verify if you have enough quota for each Azure service to set up CDP in your Azure account. See list of [Azure resources used by CDP](#).

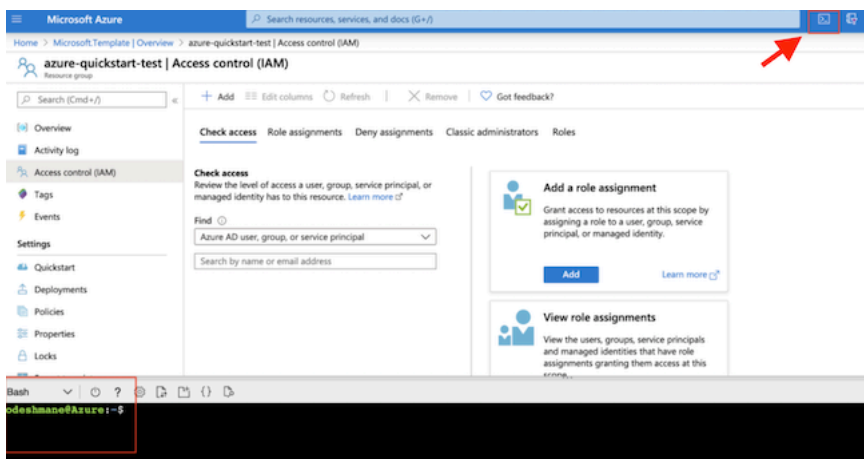
If you have more complex requirements than those listed here, contact Cloudera Sales Team to help you with CDP onboarding.

Create an Azure AD app

In the Azure portal, create an application in your Azure Active Directory tenant. This steps allows you to use the native Cloud Shell terminal and not have to set up Azure CLI.

Procedure

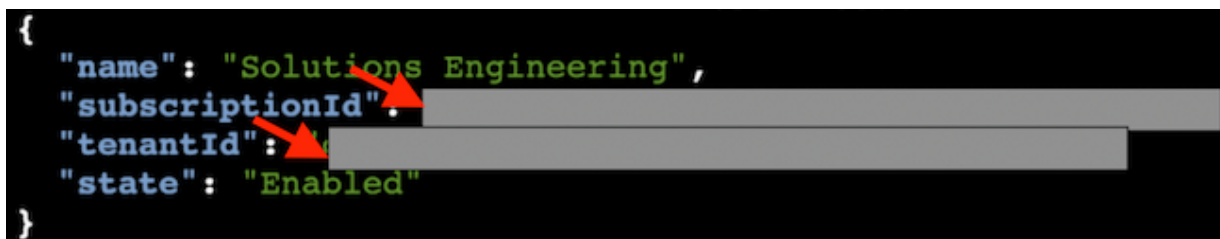
1. Log in to the Azure portal and launch Cloud Shell.



2. Run the following command to return the subscription ID and tenant ID:

```
az account list | jq '.[0]|{"name": .name, "subscriptionId": .id, "tenantId": .tenantId, "state": .state}'
```

The output of this command is shown below:



Make a note of the subscriptionId and tenantId values. You will need them later.



Note: In case you have more than one subscription, make sure to only make a note of the subscription that you would like to use for CDP.

- Run the command below to create an app in Azure AD and assign the "Contributor" role at the subscription.



Note: Replace {subscriptionId} in the command with the subscription ID value from the previous step.

```
az ad sp create-for-rbac --name http://cloudbreak-app --role Contributor
--scopes /subscriptions/{subscriptionId}
```

The output of this command is shown below:

```
msb@msb-azure:~$ az ad sp create-for-rbac --name http://cloudbreak-app --role Contributor --scopes /subscriptions/
Creating a role assignment under the scope of "/subscriptions/
{
  "appId": "11111111-1111-1111-1111-111111111111",
  "displayName": "cloudbreak-app",
  "name": "http://cloudbreak-app",
  "password": "11111111-1111-1111-1111-111111111111",
  "tenant": "11111111-1111-1111-1111-111111111111"
}
```

Deploy the Azure quickstart template

The Azure quickstart template is a customized ARM template that deploys essential Azure resources for the CDP environment.

Procedure

- Click [Deploy to Azure](#) to begin ARM template deployment of CDP prerequisites in your Azure subscription.
- Log in to Azure to create the resources in your subscription that are required for CDP deployment. These resources include VNet, ADLS Gen2, and 4 user managed identities.
- On the **Custom deployment** screen, click Create new under the Resource group field and then give the resource group a name (it should only contain letters, numbers, and hyphens).
- Under **Settings**, provide an Environment Name in the corresponding field.

Custom deployment
Deploy from a custom template

TEMPLATE

Customized template
10 resources

[Edit template](#) [Edit paramet...](#) [Learn more](#)

BASICS

Subscription *

Resource group * [Create new](#)

Location *

SETTINGS

Environment Name

Virtual Network Name

Storage Account Name

Data Access Identity Name

Logger Identity Name

Assumer Identity Name

Ranger Audit Identity Name

TERMS AND CONDITIONS

[Purchase](#)

5. Accept the terms and conditions, and click Purchase.

An ARM script begins to run in the background, creating the resources required for a CDP environment. This may take around 10 minutes.

6. When your resource group is up, navigate to the **Overview** page of the resource group.

7. Copy and paste the following values into a note, as you will need them in the next task:

- Subscription ID: Your subscription ID is found at the top of the resource group **Overview** page.
- Resource group: The name of the resource group that you created.

Assign roles

Azure Resource Manager templates do not support role assignments at a scope other than resource groups. Perform the following role assignments through UI or CLI.

Before you begin

Make sure that you have your note from the previous step, where you copied values for the Subscription ID and resource group name.

Procedure

1. Once you have values for the subscription ID, resource group name, storage account, environment name, and all four managed identities, click [here](#) to download a script.
2. Create a new file in Cloud Shell with the same name, and copy the content of the script there.
3. Replace the following values in the script with the values you have collected thus far:

```
#!/bin/sh

export SUBSCRIPTIONID="<REPLACE WITH YOUR AZURE SUBSCRIPTION ID>"
export RESOURCEGROUPNAME="<REPLACE WITH EXISTING RESOURCE GROUP NAME>"
export STORAGEACCOUNTNAME=$(az storage account list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[0].name' | tr -d ' ')
export ASSUMER_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[0].principalId' | tr -d ' ')
export DATAACCESS_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[1].principalId' | tr -d ' ')
```

For example, your script should look similar to this:

```
#!/bin/sh

export SUBSCRIPTIONID="jfs85ls8-sik8-8329-fq0m-jqo7v06dk6sy"
export RESOURCEGROUPNAME="myCDPresourcegroup"
export STORAGEACCOUNTNAME=$(az storage account list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[0].name' | tr -d ' ')
export ASSUMER_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[0].principalId' | tr -d ' ')
export DATAACCESS_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[1].principalId' | tr -d ' ')
export ASSUMER_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[0].principalId' | tr -d ' ')
export DATAACCESS_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[1].principalId' | tr -d ' ')
export LOGGER_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[2].principalId' | tr -d ' ')
export RANGER_OBJECTID=$(az identity list -g $RESOURCEGROUPNAME --subscription $SUBSCRIPTIONID | jq '.[3].principalId' | tr -d ' ')
# Assign Managed Identity Operator role to the assumerIdentity principal at subscription scope
az role assignment create --assignee $ASSUMER_OBJECTID --role 'f1a07417-d97a-45cb-824c-7a7467783830' --scope "/subscriptions/$SUBSCRIPTIONID"
# Assign Virtual Machine Contributor role to the assumerIdentity principal at subscription scope
az role assignment create --assignee $ASSUMER_OBJECTID --role '9980e02c-c2be-4d73-94e8-173b1dc7cf3c' --scope "/subscriptions/$SUBSCRIPTIONID"
```

```
# Assign Storage Blob Data Contributor role to the assumerIdentity principal at logs filesystem scope
az role assignment create --assignee $ASSUMER_OBJECTID --role 'ba92f5b4-2d11-453d-a403-e96b0029c9fe' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/logs"
# Assign Storage Blob Data Contributor role to the loggerIdentity principal at logs/backup filesystem scope
az role assignment create --assignee $LOGGER_OBJECTID --role 'ba92f5b4-2d11-453d-a403-e96b0029c9fe' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/logs"
az role assignment create --assignee $LOGGER_OBJECTID --role 'ba92f5b4-2d11-453d-a403-e96b0029c9fe' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/backups"
# Assign Storage Blob Data Owner role to the dataAccessIdentity principal at logs/data/backup filesystem scope
az role assignment create --assignee $DATAACCESS_OBJECTID --role 'b7e6dc6d-f1e8-4753-8033-0f276bb0955b' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/data"
az role assignment create --assignee $DATAACCESS_OBJECTID --role 'b7e6dc6d-f1e8-4753-8033-0f276bb0955b' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/logs"
az role assignment create --assignee $DATAACCESS_OBJECTID --role 'b7e6dc6d-f1e8-4753-8033-0f276bb0955b' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/backups"
# Assign Storage Blob Data Contributor role to the rangerIdentity principal at data/backup filesystem scope
az role assignment create --assignee $RANGER_OBJECTID --role 'ba92f5b4-2d11-453d-a403-e96b0029c9fe' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/data"
az role assignment create --assignee $RANGER_OBJECTID --role 'ba92f5b4-2d11-453d-a403-e96b0029c9fe' --scope "/subscriptions/$SUBSCRIPTIONID/resourceGroups/$RESOURCEGROUPNAME/providers/Microsoft.Storage/storageAccounts/$STORAGEACCOUNTNAME/blobServices/default/containers/backups"
```

4. Run the Cloud Shell script: `sh azure_msi_role_assign.sh`

Create or locate an SSH Key

CDP requires that you provide a public SSH key for admin access to VM instances.

You can find more information on SSH key requirement in the topic [SSH key](#). If you need to create one, you can do so by running `ssh-keygen -t rsa`.

When you complete this step, you have created all of the Azure resources required for this quickstart.

Create a CDP credential

In the CDP Console, the first step is to create a CDP credential. The CDP credential is the mechanism that allows CDP to create resources inside of your cloud account.

Procedure

1. Log in to the CDP web interface.
2. From the CDP home screen, click the Management Console icon.
3. In the Management Console, select Shared Resources > Credentials from the navigation pane.

4. Select the Azure tab, name your credential, and enter the values you previously collected for subscription ID, app ID, and password.

The screenshot shows the Cloudera DataFlow interface with the Azure tab selected. The interface includes input fields for Name, Description, Subscription Id, Tenant Id, App Id, and Password. Red arrows point to the Name, Subscription Id, Tenant Id, App Id, Password, and Create buttons. The interface also includes a code block for the Azure CLI command to list accounts and a button to show the CLI command.

Name*
Enter credential name

Description
Enter description

Paste the following command into [Azure Shell](#) to identify your Subscription Id and your Tenant Id:

```
az account list | jq '.[] | {"name": .name, "subs"
```

Subscription Id*
Enter Azure Subscription Id

Tenant Id*
Enter Tenant Id (Directory Id)

In order to create an application, you could use following command in [Azure Shell](#) or you could create it on [Azure Portal](#).

```
az ad sp create-for-rbac \ --name http://{"
```

App Id*
Enter Application Id

Password*
Enter the key generated for your app

Create > SHOW CLI COMMAND

Register a CDP environment

When you register an environment, you set properties related to data lake scaling, networking, security, and storage. You will need your Azure environment name, resource group name, storage account name, and virtual network name from your resource group.

Procedure

1. In the CDP Management Console, navigate to Environments and click Register Environment.
2. Provide an Environment Name and description. The name can be any valid name.
3. Choose Azure as the cloud provider.

4. Under Microsoft Azure Credential, choose the credential you created in the previous task.

1 Register Environment

- Name your environment
- Select a cloud provider
- Provide a credential for your cloud provider account

2 Data Lake Scaling

- Provide Data Lake name
- Choose Data Lake scale

3 Region, Networking, Security and Storage

- Select a region
- Add your SSH settings
- Define Security Access
- Select an existing Network, subnet and ABFS storage account

4 Data Access, Audit and Storage

- Add Data Access
- Add Logs Storage

General Information

Environment Name*

Enter Environment Name

Description

Enter Description

Select Cloud Provider

azure

Microsoft Azure Credential

Select Credential

azure-qs-test

5. Click Next.
6. Under **Data Lake Settings**, give your new data lake a name. The name can be any valid name. Choose the latest data lake version.

7. Under **Data Access and Audit**, choose the following:

- Assumer Identity: <resourcegroup-name>-<envName>-AssumerIdentity
- Storage Location Base: data@<storageaccount-name>
- Data Access Identity: <resourcegroup-name>-<envName>-DataAccessIdentity
- Ranger Audit Role: <resourcegroup-name>-<envName>-RangerIdentity



Warning: Ensure that you have entered the correct location base. If the name entered does not match the actual location base created by the quickstart script, environment registration will fail.

For example:



Data Access and Audit

Provide an existing location where workload data will be stored.

Assumer Identity*

azure-quickstart-test - cdpazureqs-AssumerIdentity



Storage Location Base*

abfs://

data@cdpazureqs

.dfs.core.windows.net



Data Access Identity*

azure-quickstart-test - cdpazureqs-DataAccessIdentity



Ranger Audit Identity*

azure-quickstart-test - cdpazureqs-RangerIdentity



8. For Data Lake **Scale**, choose Light Duty.

Register Environment

- Name your environment
- Select a cloud provider
- Provide a credential for your cloud provider account

2 Data Lake Scaling

- Provide Data Lake name
- Choose Data Lake scale

3 Region, Networking, Security and Storage

- Select a region
- Add your SSH settings
- Define Security Access
- Select an existing Network, subnet and ABFS storage account

4 Data Access, Audit and Storage

- Add Data Access
- Add Logs Storage

Data Lake Settings

Data Lake Name*
azure-qs-test-dl

Data Lake version*
Runtime 7.1.0

Scale

Choose a scale and a purpose of this environment from a pre-defined Data Lake template

☒ Light Duty

☐ Secure Access

9. Click Next.

10. Under **Select Region**, choose your desired region. This should be the same region you created an SSH key in previously.

11. Under **Select Resource Group**, choose your resource group <resourcegroup-name>.

12. For the **Select Network** field, select the name of the "Virtual Network" resource that was created when you deployed the ARM template to create the resource group. The name of the Virtual Network should be the same as your environment name, but you can verify this in the Azure portal on the Overview page of your resource group.

13. Under **Security Access Settings**, select Create New Security Groups for the Security Access Type.

Register Environment

- Name your environment
- Select a cloud provider
- Provide a credential for your cloud provider account

2 Data Lake Scaling

- Provide Data Lake name
- Choose Data Lake scale

3 Region, Networking, Security and Storage

- Select a region
- Add your SSH settings
- Define Security Access
- Select an existing Network, subnet and ABFS storage account

4 Data Access, Audit and Storage

- Add Data Access
- Add Logs Storage

Region, Location

Select Region
Central US - Central US

Network

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

Select Network
cdpazureqs

Select Subnets*
default

☐ Enable Cluster Connectivity Manager

☐ Don't Create Public Ip

Security Access Settings

Select Security Access Type
Create New Security Groups

Access CIDR*
0.0.0.0/0

14. Under **SSH Settings**, paste the public SSH key that you created earlier.

15. Optionally, under **Add Tags**, provide any tags that you'd like the resources to be tagged with in your Azure account.

16. Click Next.

17. Under **Logs**, choose the following:

- Logger Identity: <resourcegroup-name>-<envName>-LoggerIdentity
- Logs Location Base: logs@<storageaccount-name>
- Backup Location Base: backups@<storageaccount-name>



Warning: Ensure that you have entered the correct location base. If the name entered does not match the actual location base created by the quickstart script, environment registration will fail.

For example:



Logs

Provide an existing location where log files will be stored.

Logger Identity*

azure-quickstart-test - cdpazurereqs-LoggerIdentity

Logs Location Base*

abfs:// logs@cdpazurereqs dfe.core.windows.net

Backup Location Base (Optional)

abfs:// backups@cdpazurereqs dfe.core.windows.net

18. Click Register Environment.

Give administrators access

To enable Cloudera DataFlow for an environment, users must have the DFAdmin role. Grant the DFAdmin role to a user or group that should be allowed enable CDF for an environment.

Before you begin

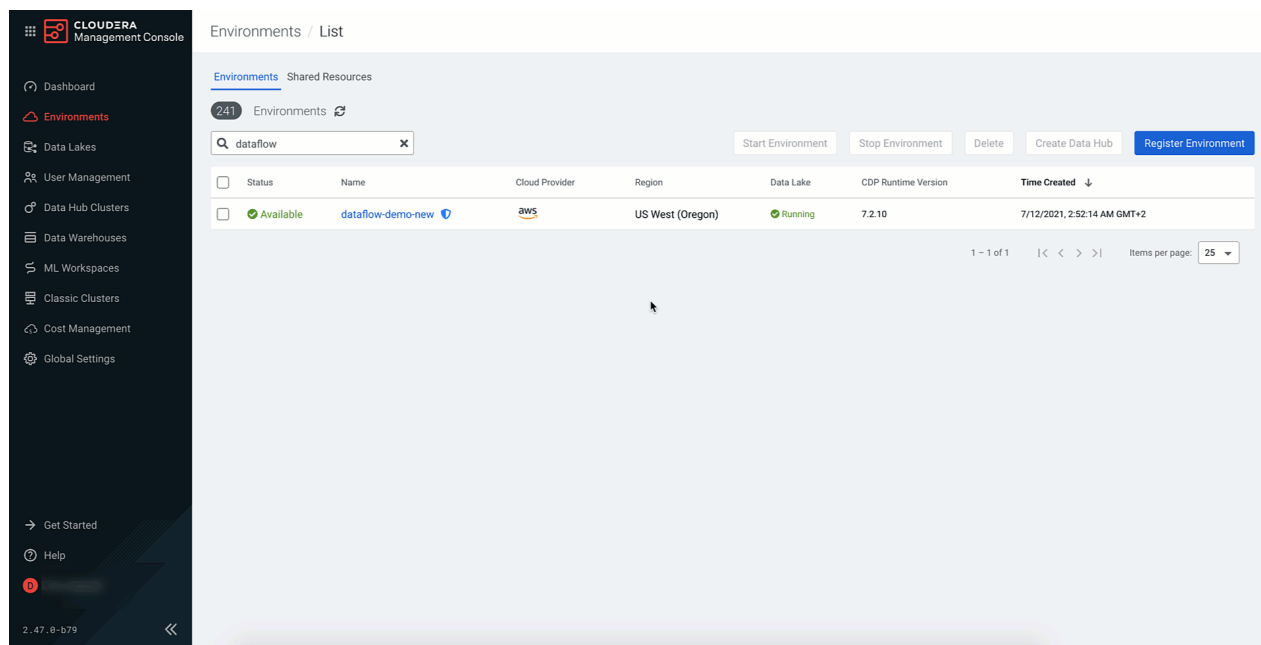
- You have the Cloudera PowerUser role.

Procedure

1. From the Cloudera Management Console, click Environments.
2. Use the search field to find and select the Cloudera Public Cloud environment for which you want to grant DFAdmin rights.
3. Click Actions | Manage Access to display the Environment Access page.
4. Find the user to whom you want to grant the DFAdmin role, and click Update Roles.

5. From the Update Roles page, select DFAdmin and click Update.

Example



What to do next

When you have finished granting a user or group administrator permissions, they can now proceed by enabling Cloudera DataFlow for an environment.

Related Information

[Cloudera DataFlow Security](#)

Enable Cloudera DataFlow for your environment

Before you can deploy flow definitions, you must enable Cloudera DataFlow for a Cloudera Public Cloud environment. Enabling Cloudera DataFlow for an environment means that you are preparing an active and healthy Cloudera Public Cloud environment for use with Cloudera DataFlow.

Before you begin

- You have a cloud provider account and meet the infrastructure and network requirements.
- You have a healthy Cloudera Public Cloud environment, with FreeIPA and the data lake running and healthy.
- You have the DFAdmin role for the Cloudera Public Cloud environment for which you want to enable Cloudera DataFlow.

Procedure

1. Navigate to Cloudera DataFlow, by selecting DataFlow from the Cloudera Public Cloud Home Page.
2. In Cloudera DataFlow, navigate to Environments, and click Enable to launch the Enable Environment dialog for the environment you want to enable.

3. From Enable Environment, provide the following information:

- DataFlow Capacity – Specifies Kubernetes cluster minimum and maximum size
- Networking
- Specify whether a public endpoint should be deployed to access CDF components via the internet.



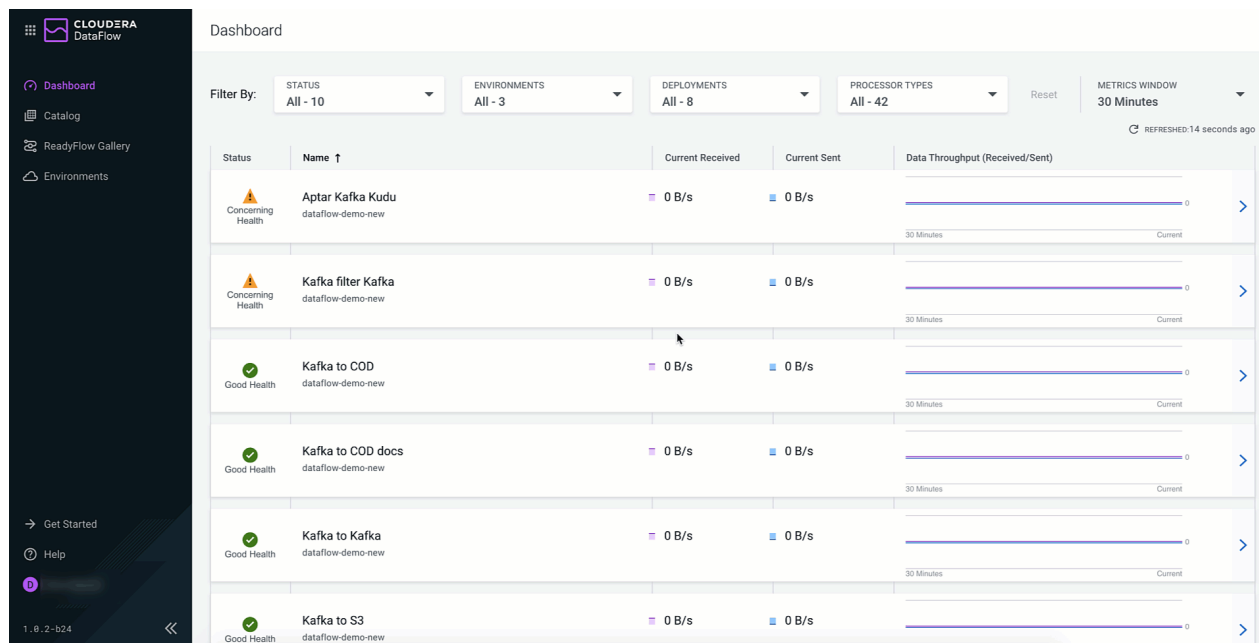
Note:

CDF only makes this option selectable if it detects that at least one subnet in your environment is considered a public subnet.

- A list of source IP address ranges which are allowed to connect to the Kubernetes API server.

4. Click Enable. Enabling CDF can take up to one hour.

Example



What to do next

When you have finished enabling Cloudera DataFlow for an environment, proceed by giving users permission to import and deploy flow definitions.

Related Information

[Enabling Cloudera DataFlow for an environment](#)

[Managing Cloudera DataFlow in an environment](#)

Give users access

Cloudera DataFlow restricts who can import flow definitions and deploy them. To get started with Cloudera DataFlow, you must grant users permissions to perform these tasks so that they can import and deploy flow definitions in Cloudera DataFlow.

About this task

To get started with importing and deploying flow definitions, a user needs to have the following Cloudera DataFlow roles, at a minimum:

- DFCatalogAdmin – To import flow definitions to the Cloudera DataFlow Catalog

- DFFlowAdmin – To deploy flow definitions in Cloudera DataFlow

Before you begin

- You have PowerUser role.
- You know the user or group name to which you want to grant Cloudera DataFlow user access roles.

Procedure

1. Give a user permission to import flow definitions.
 - a) From Cloudera Management Console, click User Management.
 - b) Enter the name of the user or group you wish to authorize in the Search field.
 - c) Select the user or group from the list that displays.
 - d) Click Roles, then Update Roles.
 - e) From Update Roles, select DFCatalogAdmin and click Update.
2. Give a user or group permission to deploy flow definitions.
 - a) From Cloudera Management Console, click Environments to display the Environment List page.
 - b) Select the environment to which you want a user or group to deploy flow definitions.
 - c) Click Actions Manage Access to display the Environment Access page.
 - d) Find the user or group and click Update Roles.
 - e) From Update Roles, select DFFlowAdmin.
 - f) Click Update Roles.

What to do next

When you have finished giving user or groups permission to import and deploy flow definitions, proceed by importing a flow definition.

Related Information

[Cloudera DataFlow Security](#)

Add the Hello World ReadyFlow definition to the Catalog

Hello World is an out of box flow definition, designed to make getting started with Cloudera DataFlow quick and easy. To use it, just add the ReadyFlow to the Catalog.

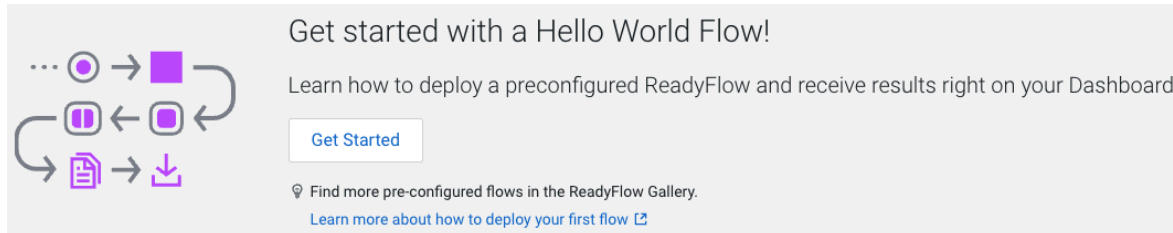
Before you begin

- You have an enabled and healthy Cloudera DataFlow environment.
- You have been assigned the DFCatalogAdmin role granting you access to the Catalog.
- You have been assigned the DFFlowAdmin role for the environment to which you want to deploy the flow definition.

About this task



Note: In a new Cloudera DataFlow deployment with an empty Catalog, you can deploy Hello World directly from the Dashboard.

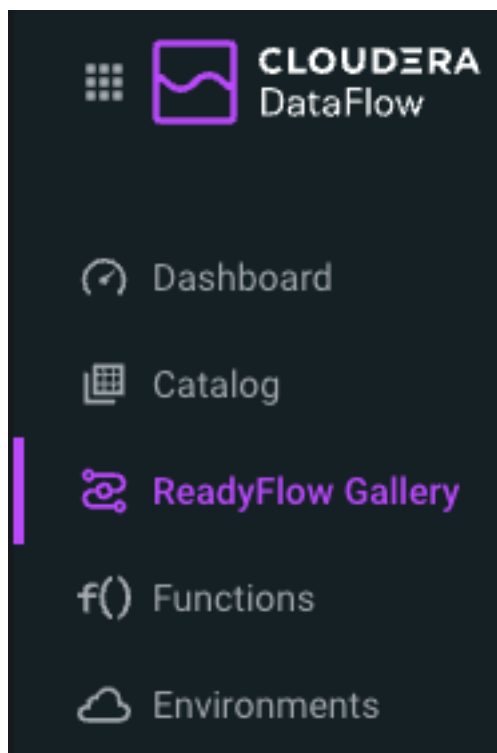


Just click Get Started on the main pane of the Dashboard. You will be forwarded to the Catalog, from where you can deploy the ReadyFlow.

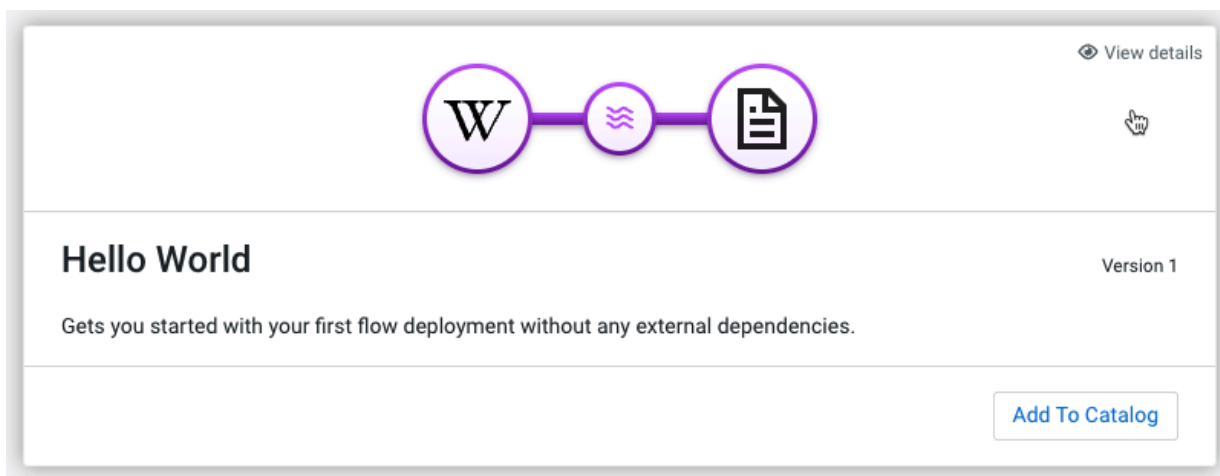
If the Catalog in your ReadyFlow deployment already contains at least one flow definition, You can add Hello World to the Catalog from the ReadyFlow Gallery.

Procedure

1. On the left hand navigation pane, click ReadyFlow Gallery.



2. From the ReadyFlow Gallery page, select the Hello World ReadyFlow.



3. If you hover over the card representing the flow definition, the View Details hint appears. Click anywhere on the card to review the ReadyFlow details.
4. Click Add to Catalog to add the ReadyFlow into the CDF Catalog and make it ready for deployment.
5. Click Add to confirm that you want to add Hello World to the Catalog.

Results

The Hello World flow definition is added to the Catalog and is ready for deployment.

What to do next

You can now proceed with deploying the ReadyFlow.

Deploy the Hello World ReadyFlow using the deployment wizard

Learn about the steps to deploy the Hello World Flow to get started with Cloudera DataFlow.

About this task

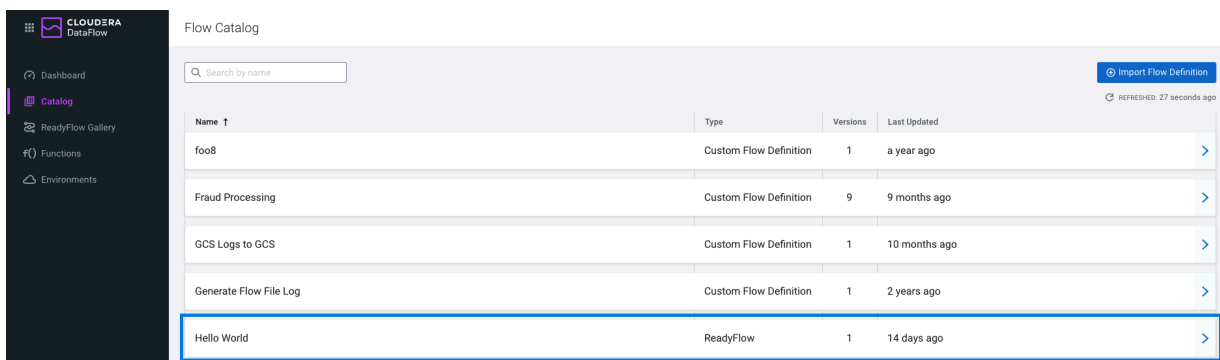
Once you have added the Hello World ReadyFlow into the Catalog, stay in the Catalog and use the Deployment wizard to deploy that flow definition.

Before you begin

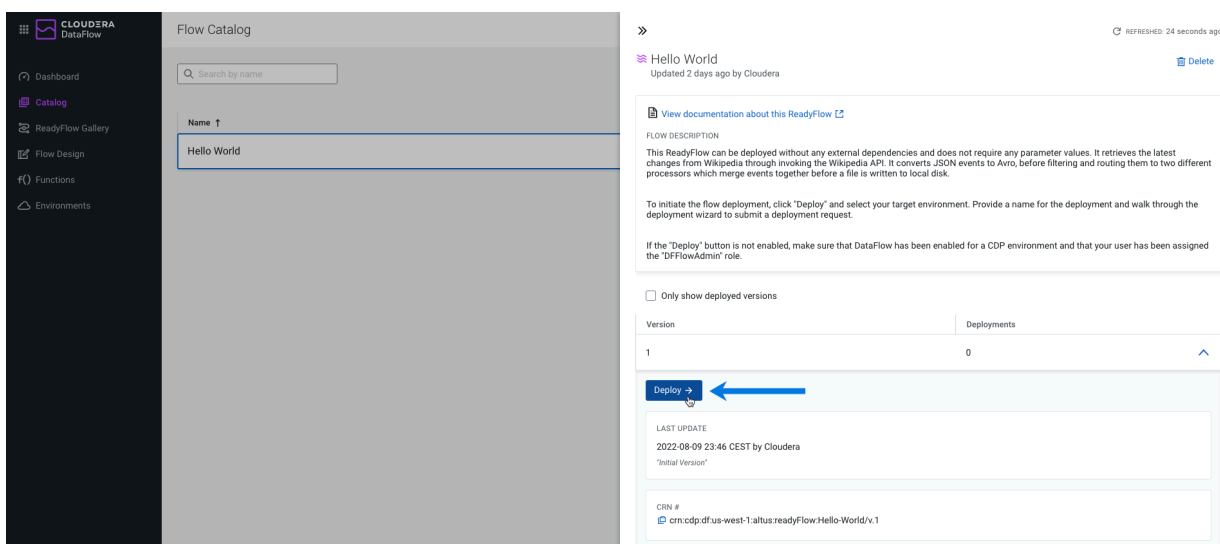
- You have an enabled and healthy Cloudera DataFlow environment.
- You have imported the Hello World Flow flow definition.
- You have been assigned the DFCatalogAdmin role granting you access to the Catalog.
- You have been assigned the DFFlowAdmin role for the environment to which you want to deploy the flow definition.

Procedure

1. To launch the Deployment wizard, select the Hello World flow in the Catalog to display the flow definition details and versions.



2. Select Version 2 and click Deploy to launch the Deployment wizard.



3. Select the Target Workspace where you want to deploy the ReadyFlow.

The Deployment wizard only displays workspaces that meet the following conditions:

- Cloudera DataFlow is enabled for the environment
- The environment is in healthy state
- You are allowed to access the environment


4. Click Continue.

5. In Overview, give your flow deployment a unique name.



Note: Flow Deployment names need to be unique. The Deployment Wizard indicates whether a name is valid by displaying a green check below the Deployment Name text box.

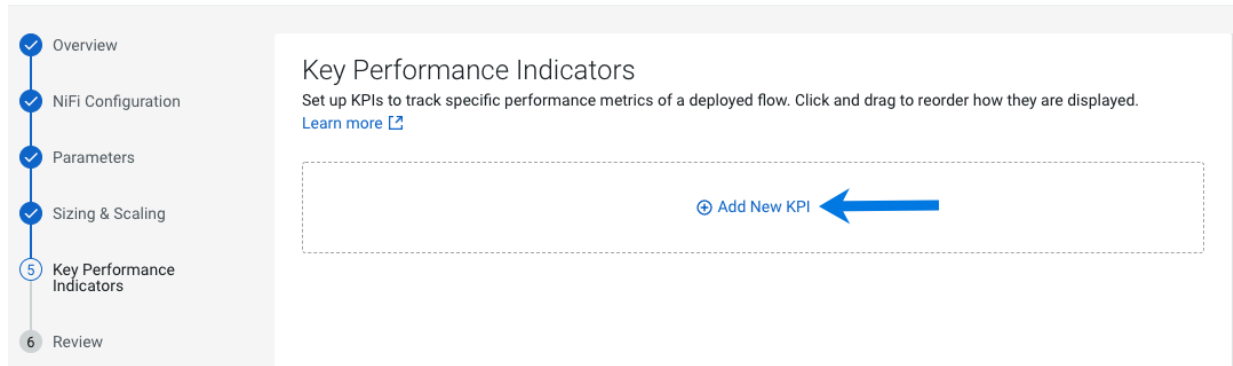


Note: As you have created no Projects for this quick start, you can leave your flow deployment  Unassigned.

6. Click Next.
7. In NiFi Configuration, click Next.
8. In Parameters, click Next.
9. In Sizing & Scaling, click Next.

- 10.** In Key Performance Indicators, click Add New KPI to add a Key Performance Indicator. It lets you monitor the performance of your Cloudera DataFlow deployment.

cdf-priv / New Deployment



- 11.** To learn about the use of Key Performance Indicators (KPIs), add a KPI that will track how much data the processor is writing to the destination files. The KPI will raise an alert whenever this value goes below 1MB.

Set the following properties:

KPI Scope

select Processor.

Processor Name

select Write "Added Content" Events to File.

Metric to Track

select Bytes Sent

Alerts

select Trigger alert when metric is less than and set value to 1 MBytes.

Alert will be triggered when metric is outside the boundary(s) for

set to 2 Minutes.

Add new KPI



Details

KPI Scope

Processor

Processor Name

Write "Added Content" Events To File

Metric to Track

Bytes Sent

METRIC DESCRIPTION:

Number of bytes sent to an external recipient

Alerts

☐ Trigger alert when metric is greater than

Value

MBytes

☒ Trigger alert when metric is less than

1

MBytes

Alert will be triggered when metric is outside the boundary(s) for

2

Minutes

Cancel

Add

12. Click Add, to create the KPI.

13. Click Next.

14. Review a summary of the information provided. When you are finished, complete flow deployment by clicking Deploy.

Results

Once you click Deploy, you are redirected to the Alerts tab in the detail view for the deployment where you can track its progress.

Related Information

[Deploying Flow Definitions](#)

[Monitoring and Managing Flow Deployments](#)

[KPI overview](#)

[Working with KPIs](#)