Configuring Cloud Data Access

Date of Publish: 2019-02-06

https://docs.hortonworks.com/

Contents

Configuring access to Amazon S3

Amazon S3 is not supported as default file system, but access to data in S3 is possible via the s3a connector. Use these steps to configure access from your cluster to Amazon S3.

These steps assume that you are using an HDP version that supports the s3a cloud storage connector (HDP 2.6.1 or newer).

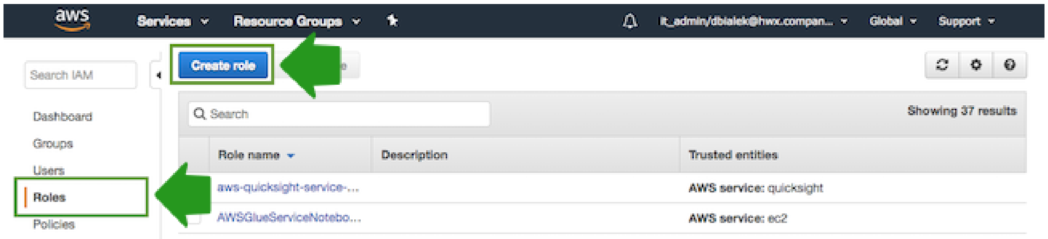
Create an IAM role for S3 access

In order to configure access from your cluster to Amazon S3, you must have an existing IAM role which determines what actions can be performed on which S3 buckets.

If you already have an IAM role, skip to the next step. If you do not have an existing IAM role, use the following instructions to create one.

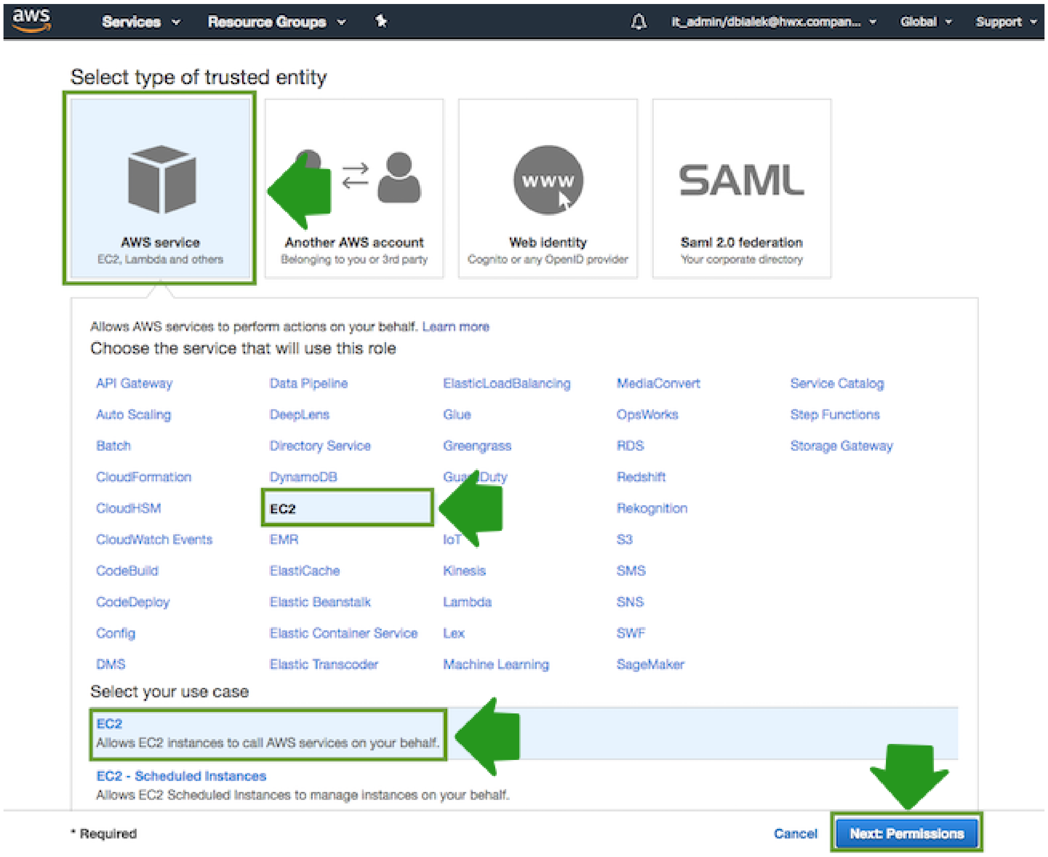
Steps

1. Navigate to the IAM console > Roles and click Create Role.



Role name	Description	Trusted entities
aws-quicksight-service-...		AWS service: quicksight
AWSGlueServiceNotebo...		AWS service: ec2

2. In the “Create Role” wizard, select AWS service role type and then select EC2 service and EC2 use case.



Select type of trusted entity

AWS service
EC2, Lambda and others

Another AWS account
Belonging to you or 3rd party

Web identity
Cognito or any OpenID provider

Saml 2.0 federation
Your corporate directory

Allows AWS services to perform actions on your behalf. Learn more

Choose the service that will use this role

API Gateway
Auto Scaling
Batch
CloudFormation
CloudHSM
CloudWatch Events
CodeBuild
CodeDeploy
Config
DMS

Data Pipeline
DeepLens
Directory Service
DynamoDB
EC2
EMR
ElastiCache
Elastic Beanstalk
Elastic Container Service
Elastic Transcoder

ElasticLoadBalancing
Glue
Greengrass
GuardDuty
IoT
Kinesis
Lambda
Lex
Machine Learning

MediaConvert
OpsWorks
RDS
Redshift
Rekognition
S3
SMS
SNS
SWF
SageMaker

Service Catalog
Step Functions
Storage Gateway

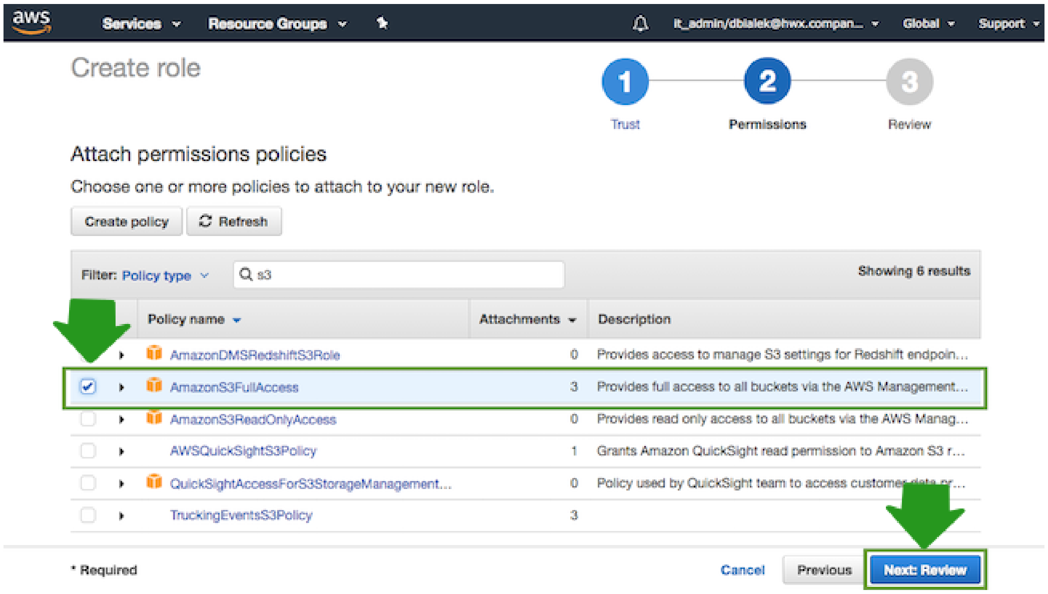
Select your use case

EC2
Allows EC2 instances to call AWS services on your behalf.

EC2 - Scheduled Instances
Allows EC2 Scheduled Instances to manage instances on your behalf.

* Required

3. When done, click Next: Permissions to navigate to the next page in the wizard.
4. Select an existing S3 access policy or click Create policy to define a new policy. If you are just getting started, you can select a built-in policy called “AmazonS3FullAccess”, which provides full access to S3 buckets that are part of your account:



Create role

Attach permissions policies

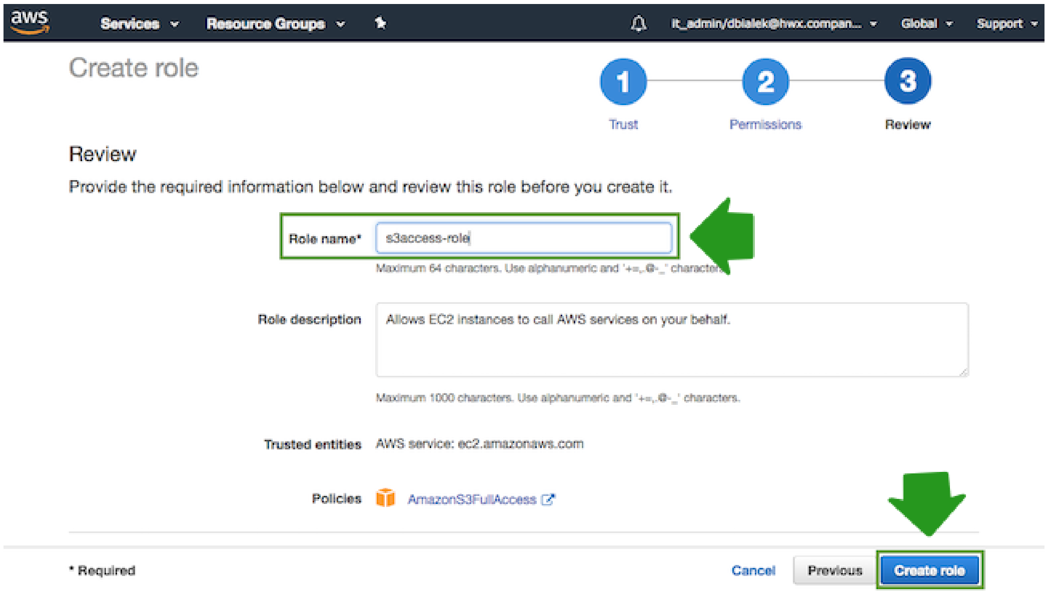
Choose one or more policies to attach to your new role.

Filter: Policy type   s3   Showing 6 results

	Policy name	Attachments	Description
□	AmazonDMSRedshiftS3Role	0	Provides access to manage S3 settings for Redshift endpoin...
☑	AmazonS3FullAccess	3	Provides full access to all buckets via the AWS Management...
□	AmazonS3ReadOnlyAccess	0	Provides read only access to all buckets via the AWS Manag...
□	AWSQuickSightS3Policy	1	Grants Amazon QuickSight read permission to Amazon S3 r...
□	QuickSightAccessForS3StorageManagement...	0	Policy used by QuickSight team to access customer data pr...
□	TruckingEventsS3Policy	3	

* Required

5. When done attaching the policy, click Next: Review.
6. In the Roles name field, enter a name for the role that you are creating:



Create role

Review

Provide the required information below and review this role before you create it.

Role name* s3access-role

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Role description Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

Trusted entities AWS service: ec2.amazonaws.com

Policies AmazonS3FullAccess

* Required

7. Click Create role to finish the role creation process.

Configure access to S3

Access to data in S3 from your cluster VMs can be automatically configured by attaching an existing instance profile allowing access to S3.

Prerequisites

You or your AWS admin must create an IAM role with an S3 access policy which can be used by cluster instances to access one or more S3 buckets. Refer to Create an IAM role for S3 access.

Steps

1. On the Cloud Storage page in the advanced cluster wizard view, select Use existing instance profile.
2. Select an existing IAM role created in step 1.
3. During the cluster creation process, Cloudbreak assigns the IAM role and its associated permissions to the EC2 instances that are part of the cluster so that applications running on these instances can use the role to access S3. Once your cluster is in the running state, you will be able to access the S3 bucket from the cluster nodes.

Related Information

Create an IAM role for S3 access

Using an IAM Role to Grant Permissions to Applications Running on Amazon EC2 Instances (AWS)

Creating a cluster

Test access from HDP to S3

To test access to S3 from HDP, SSH to a cluster node and run a few hadoop fs shell commands against your existing S3 bucket.

To test access, SSH to any cluster node and switch to the hdfs user by using sudo su hdfs.

Amazon S3 access path syntax is:

```
s3a://bucket/dir/file
```

For example, to access a file called “mytestfile” in a directory called “mytestdir”, which is stored in a bucket called “mytestbucket”, the URL is:

```
s3a://mytestbucket/mytestdir/mytestfile
```

The following FileSystem shell commands demonstrate access to a bucket named “mytestbucket”:

```
hadoop fs -ls s3a://mytestbucket/
hadoop fs -mkdir s3a://mytestbucket/testDir
hadoop fs -put testFile s3a://mytestbucket/testFile
hadoop fs -cat s3a://mytestbucket/testFile
test file content
```

For more information about configuring the S3 connector for HDP and working with data stored on S3, refer to Cloud Data Access HDP documentation.

Related Information

Cloud data access (HDP)

Configure S3 storage locations

After configuring access to S3 via instance profile, you can optionally use an S3 bucket as a base storage location; this storage location is mainly for the Hive Warehouse Directory (used for storing the table data for managed tables).

Prerequisites

- You must have an existing bucket. For instructions on how to create a bucket on S3, refer to AWS documentation.
- The instance profile that you configured must allow access to the bucket.

Steps

1. When creating a cluster, on the Cloud Storage page in the advanced cluster wizard view, select Use existing instance profile and select the instance profile to use, as described in the documentation for configuring access to S3.
2. Under Storage Locations, enable Configure Storage Locations by clicking the

   button.
3. Provide your existing bucket name under Base Storage Location.

   Note:

   Make sure that the bucket already exists within the account.
4. Under Path for Hive Warehouse Directory property (hive.metastore.warehouse.dir), Cloudbreak automatically suggests a location within the bucket. You may optionally update this path or select Do not configure.

   Note:

   This directory structure will be created in your specified bucket upon the first activity in Hive.

Related Information

Configure access to S3

Create a bucket (AWS)

Configuring access to ADLS Gen1

Azure Data Lake Store Gen1 (ADLS Gen1) is not supported as default file system, but access to data in ADLS Gen1 is possible via the adl connector.

Note:

After ADLS Gen2 was introduced, "ADLS" was renamed to "ADLS Gen1" on Azure Portal. After ADLS Gen2 related features were introduced in Cloudbreak, "ADLS" was renamed to "ADLS Gen2" in Cloudbreak web UI, CLI, and documentation.

These steps assume that you are using an HDP version that supports the adl cloud storage connector (HDP 2.6.1 or newer).

Related Information

Overview of Azure Data Lake Store (Azure)

Prerequisites

If you would like to use ADLS Gen1 to store your data, you must have an Azure subscription and a storage account.

For instructions on how to get started with ADLS Gen1, refer to Azure documentation.

Related Information

Get started with Azure Data Lake Store using the Azure portal (Azure)

Configure access to ADLS Gen1

To configure authentication with ADLS Gen1 using the client credential, you must register a new application with the Active Directory service and then give your application access to your storage account. After you’ve performed these steps, you can provide your application’s information when creating a cluster.

Prerequisites

Register an application and add it to the storage account, as described in Step 1 and Step 2 of the HCC article How to configure authentication with ADLS.

Note:

Do not perform the Step 3 described in this article. Cloudbreak automates this step.

Steps

1. In Cloudbreak web UI, on the advanced Cloud Storage page of the create a cluster wizard, select Use existing ADLS storage.
2. Provide the following parameters for your registered application:
   - ADLS Account Name: This is the storage account that your application was assigned to.
   - Application ID: You can find it in your application’s settings.
   - Key: This is the key that you generated for your application. If you did not copy the it, you must create a new key from the Keys page in your application’s settings.
3. Once your cluster is in the running state, you will be able to access the Azure blob storage account from the cluster nodes.

Related Information

How to configure authentication with ADLS (HCC)

Creating a cluster

Test access to ADLS Gen1

To test access to ADLS Gen1, SSH to a cluster node and run a few hadoop fs shell commands against your existing storage account.

To test access, SSH to any cluster node and switch to the hdfs user by using sudo su hdfs.

ADLS access path syntax is:

```
adl://account_name.azuredatalakestore.net/dir/file
```

For example, the following Hadoop FileSystem shell commands demonstrate access to a storage account named “myaccount”:

```
hadoop fs -mkdir adl://myaccount.azuredatalakestore.net/testdir
hadoop fs -put testfile adl://myaccount.azuredatalakestore.net/testdir/
testfile
```

To use DistCp against ADLS Gen1, use the following syntax:

```
hadoop distcp
    [-D hadoop.security.credential.provider.path=localjceks://file/home/
user/adls.jceks]
    hdfs://namenode_hostname:9001/user/foo/007020615
    adl://myaccount.azuredatalakestore.net/testDir/
```

For more information about configuring the adl connector and working with data stored in ADLS Gen1, refer to Cloud Data Access HDP documentation.

Related Information

Cloud data access (HDP)

Configure ADLS Gen1 storage locations

After configuring access to ADLS Gen1, you can optionally use that storage account as a base storage location; this storage location is mainly for the Hive Warehouse Directory (used for storing the table data for managed tables).

Steps

1. When creating a cluster, on the Cloud Storage page in the advanced cluster wizard view, select Use existing ADLS storage and enter information related to your storage account, as described in the instructions for configuring access to ADLS Gen1.
2. Under Storage Locations, enable Configure Storage Locations by clicking the

   button.
3. Provide your existing directory name under Base Storage Location.

   Note:

   Make sure that the container already exists within the account.
4. Under Path for Hive Warehouse Directory property (hive.metastore.warehouse.dir), Cloudbreak automatically suggests a location within the directory. You may optionally update this path or select Do not configure.

   Note:

   This directory structure will be created in your specified container upon the first activity in Hive.

Related Information

Configure access to ADLS Gen1

Configuring access to ADLS Gen2

Azure Data Lake Storage Gen2 (ADLS Gen2) is not supported as default file system, but access to data in Azure Data Lake Storage Gen2 is possible via the abfs connector.

Use the following steps to configure access from your cluster to ADLS Gen2. These steps assume that you are using an HDP version that supports the abfs cloud storage connector (HDP 3.0 or newer).

For basic information about ADLS Gen 2, refer to Azure documentation.

Note:

This feature is technical preview: It is not suitable for production.

Related Information

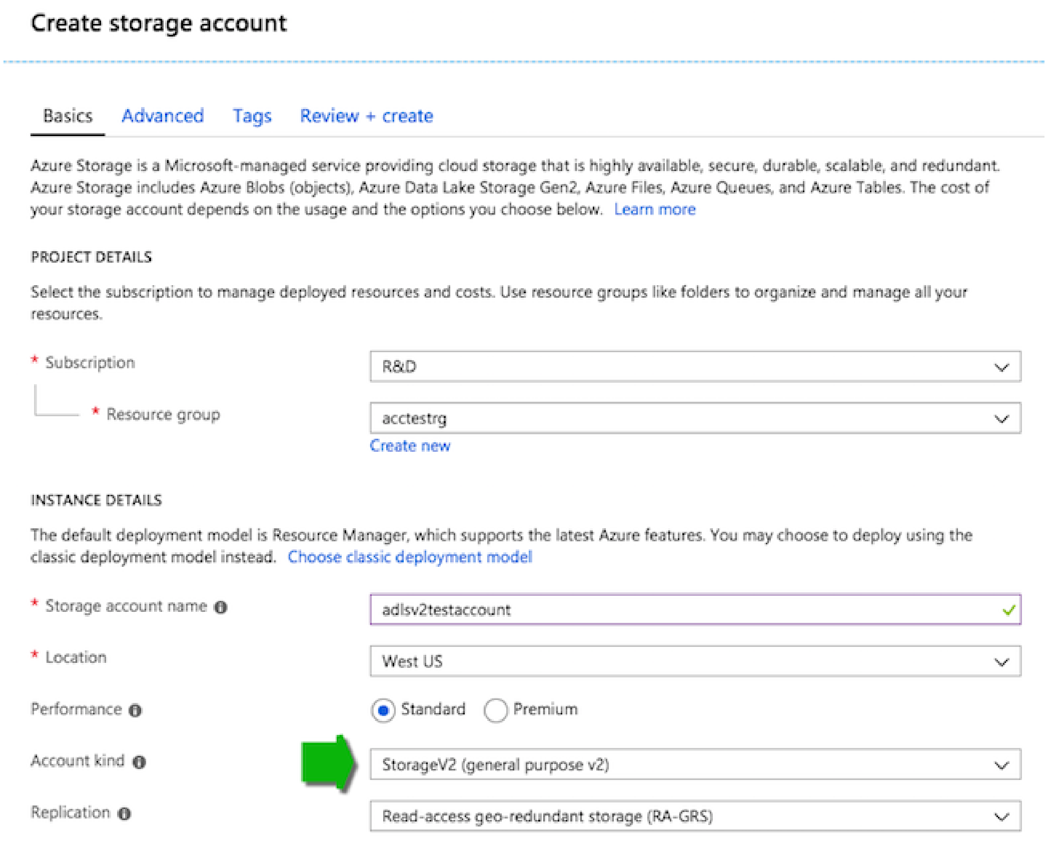
The Azure Blob Filesystem driver (Azure)

Introduction to Azure Data Lake Storage Gen2 Preview (Azure)

Prerequisites

If you would like to use ADLS Gen2 to store your data, you must have an Azure subscription and a storage account of type “StorageV2”.

To create an ADLS gen2 storage account, select "StorageV2":



Create storage account

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. Learn more

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription: R&D

* Resource group: acctestrg

Create new

INSTANCE DETAILS

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. Choose classic deployment model

* Storage account name: adlsv2testaccount ✓

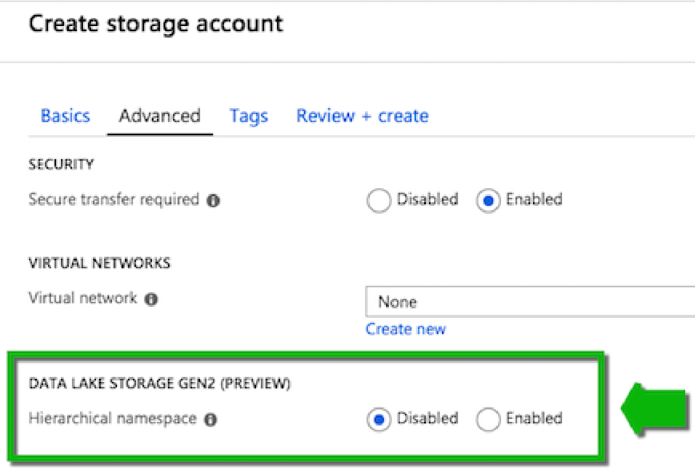
* Location: West US

Performance: ● Standard ○ Premium

Account kind: StorageV2 (general purpose v2)

Replication: Read-access geo-redundant storage (RA-GRS)

Optionally, you may enable the hierarchical namespace feature for the storage account:



Create storage account

SECURITY

Secure transfer required ○ Disabled ● Enabled

VIRTUAL NETWORKS

Virtual network: None

DATA LAKE STORAGE GEN2 (PREVIEW)

Hierarchical namespace ● Disabled ○ Enabled

For more information about the hierarchical namespace feature and for instructions for creating a storage account, refer to Azure documentation.

Related Information

Quickstart: Create an Azure Data Lake Storage Gen2 Preview storage account (Azure)

Azure Data Lake Storage Gen2 Preview hierarchical namespace (Azure)

Configure access to ADLS Gen2

When creating a cluster with Cloudbreak, you can configure access from the cluster a to ADLS Gen2.

Prerequisites

1. In order to access ADLS Gen2 from clusters, your account must have the following permissions:

```
"Actions": [
  "Microsoft.Storage/*/read",
  "Microsoft.Storage/storageAccounts/write",
  "Microsoft.Storage/storageAccounts/blobServices/write",
  "Microsoft.Storage/storageAccounts/blobServices/containers/*",
  "Microsoft.Storage/storageAccounts/listkeys/action",
  "Microsoft.Storage/storageAccounts/regeneratekey/action",
  "Microsoft.Storage/storageAccounts/delete",
  "Microsoft.Storage/locations/deleteVirtualNetworkOrSubnets/action",
  "Microsoft.DataLakeStore/*/read"
"DataActions": [
  "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/*"
],
```

   You can find more information about these permissions by using the following Azure CLI command:

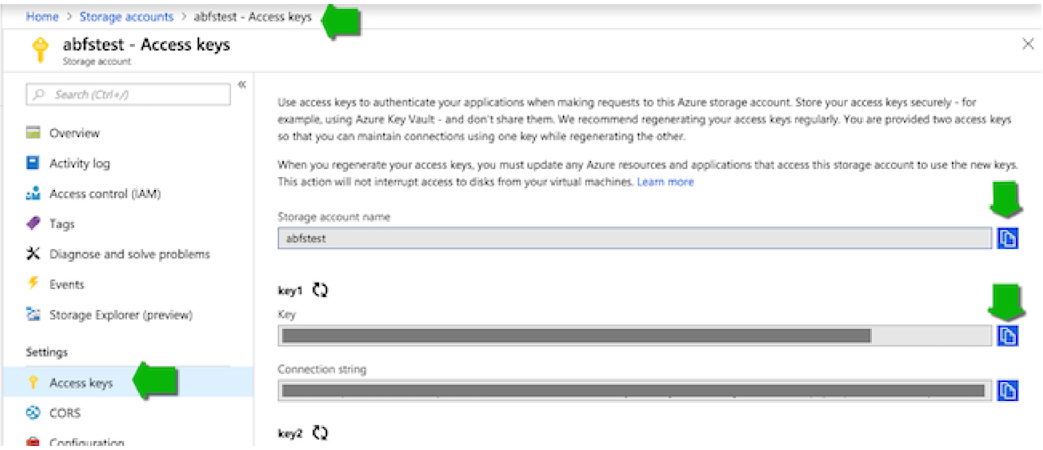
```
az provider operation show --namespace Microsoft.Storage
```

2. If you are NOT using hierarchical namespace, then you must set up the container that you want to use with your cluster.

Steps

1. In Cloudbreak web UI, on the advanced Cloud Storage page of the create a cluster wizard, select Use existing ABFS storage.
2. Provide the following parameters for your registered application:
   - Storage Account Name
   - Access Key

You can obtain the storage account name and the access key from the storage account’s Settings > Access keys:



Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more

Storage account name: abfstest

key1

Key: [redacted]

Connection string: [redacted]

key2

3. Once your cluster is in the running state, you will be able to access the ADLS Gen2 storage account from the cluster nodes.

Related Information

Creating a cluster

Test access to ADLS Gen2

To test access to ADLS, SSH to a cluster node, switch to the hdfs user (by using ‘sudo su hdfs’), and run a few hadoop fs shell commands against your existing storage account.

ADLS access path syntax is:

```
abfs://file_system@account_name.dfs.core.windows.net/dir/file
```

Use abfss instead of abfs to connect with a secure socket layer connection.

For more information about syntax refer to Azure documentation.

For example, the following Hadoop FileSystem shell commands demonstrate access to “test container” in a storage account named “myaccount”:

```
hadoop fs -mkdir abfs://test-container@myaccount.dfs.core.windows.net/
testdir
hadoop fs -put testfile abfs://test-
container@myaccount.dfs.core.windows.net/testdir/testfile
```

FilesystemNotFound error

When hierarchical namespace is enabled, you may experience the java.io.FileNotFoundException with an error code of 404 mentioning FilesystemNotFound and the error message “The specified filesystem does not exist”. The common reason for this error is that you are trying to access a blob container created through the Azure Portal, but when hierarchical namespace is enabled, you must not create containers through Azure Portal.

To solve this issue, do one of the following:

- Delete the blob container through Azure Portal, wait a few minutes, and then try accessing this container.
- Or use a different container that is not created through Azure Portal.

Related Information

URI Syntax (Azure)

Configure ADLS Gen2 storage locations

After configuring access to ADLS, you can optionally use that ADLS Gen2 storage account as a base storage location; this storage location is mainly for the Hive Warehouse Directory (used for storing the table data for managed tables).

Steps

1. When creating a cluster, on the Cloud Storage page in the advanced cluster wizard view, select Use existing ADLS storage and enter information related to your storage account, as described in the instructions for configuring access to ADLS Gen2.
2. Under Storage Locations, enable Configure Storage Locations by clicking the

   button.
3. Provide your existing directory name under Base Storage Location.

   Note:

   Make sure that the container already exists within the account.
4. Under Path for Hive Warehouse Directory property (hive.metastore.warehouse.dir), Cloudbreak automatically suggests a location within the directory. You may optionally update this path or select Do not configure.

   Note:

   This directory structure will be created in your specified container upon the first activity in Hive.

Related Information

Configure access to ADLS Gen2

Configuring access to WASB

Windows Azure Storage Blob (WASB) is an object store service available on Azure. WASB is not supported as default file system, but access to data in WASB is possible via the wasb connector.

These steps assume that you are using an HDP version that supports the wasb cloud storage connector (HDP 2.6.1 or newer).

Prerequisites

If you want to use Windows Azure Storage Blob to store your data, you must have an Azure subscription and a storage account.

For detailed instructions, refer to Azure documentation.

Related Information

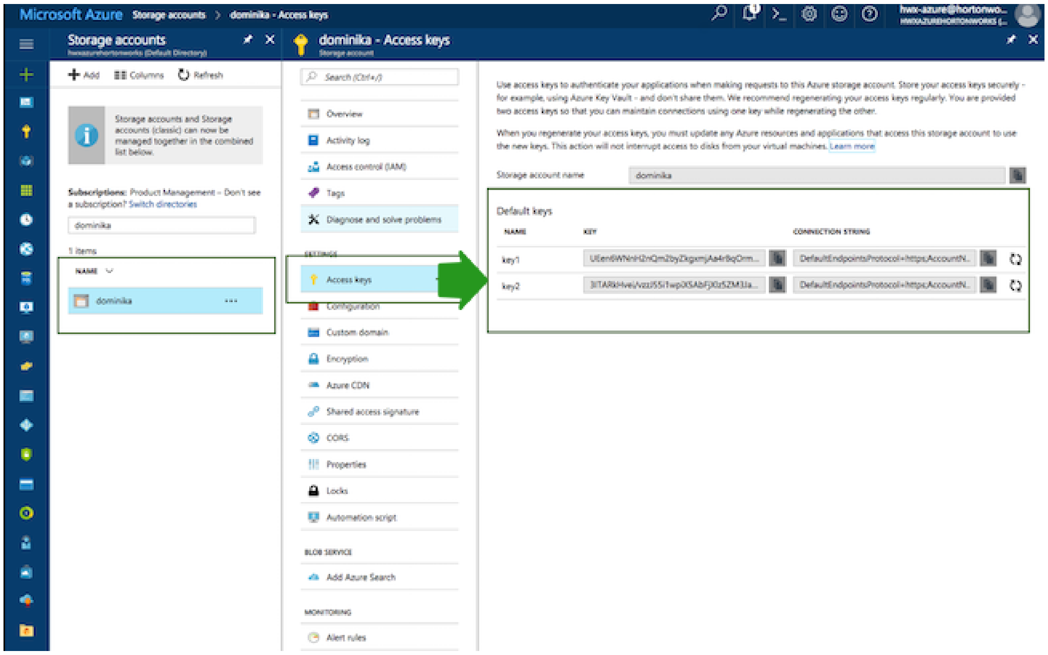
Create a storage account (Azure)

Configure access to WASB

In order to access data stored in your Azure blob storage account, you must obtain your access key from your storage account settings, and then provide the storage account name and its corresponding access key when creating a cluster.

Steps

1. Obtain your storage account name and storage access key from the Access keys page in your storage account settings:



Storage accounts and Storage accounts (classic) can now be managed together in the combined list below.

Subscriptions: Product Management – Don't see a subscription? Switch directories

dominika

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more

Storage account name: dominika

Default keys

NAME	KEY	CONNECTION STRING
key1	UEen6WNnH2nQm2byZkgxmjAa4r8qOrm...	DefaultEndpointsProtocol=https;AccountN...
key2	3ITARkHvei/vzzJ55i1wpiXSAbFjX0z5ZM3Ja...	DefaultEndpointsProtocol=https;AccountN...

2. In Cloudbreak web UI, on the advanced Cloud Storage page of the create a cluster wizard:
   a. Select Use existing WASB storage.
   b. Provide the Storage Account Name and Access Key parameters obtained in the previous step.
3. Once your cluster is in the running state, you will be able to access ADLS from the cluster nodes.

Related Information

Creating a cluster

Test access to WASB

To test access to WASB, SSH to a cluster node and run a few hadoop fs shell commands against your existing WASB account.

To test access, SSH to any cluster node and switch to the hdfs user by using sudo su hdfs.

WASB access path syntax is:

```
wasb://container_name@storage_account_name.blob.core.windows.net/dir/file
```

For example, to access a file called “testfile” located in a directory called “testdir”, stored in the container called “testcontainer” on the account called “hortonworks”, the URL is:

```
wasb://testcontainer@hortonworks.blob.core.windows.net/testdir/testfile
```

You can also use “wasbs” prefix to utilize SSL-encrypted HTTPS access:

```
wasbs://<container_name>@<storage_account_name>.blob.core.windows.net/dir/
file</pre>
```

The following Hadoop FileSystem shell commands demonstrate access to a storage account named "myaccount" and a container named "mycontainer":

```
hadoop fs -ls wasb://mycontainer@myaccount.blob.core.windows.net/
hadoop fs -mkdir wasb://mycontainer@myaccount.blob.core.windows.net/testDir

hadoop fs -put testFile wasb://mycontainer@myaccount.blob.core.windows.net/
testDir/testFile

hadoop fs -cat wasb://mycontainer@myaccount.blob.core.windows.net/testDir/
testFile
test file content
```

For more information about configuring the ADLS connector and working with data stored in ADLS, refer to Cloud Data Access HDP documentation.

Related Information

Cloud data access (HDP)

Configure WASB storage locations

After configuring access to WASB, you can optionally use that WASB storage account as a base storage location; this storage location is mainly for the Hive Warehouse Directory (used for storing the table data for managed tables).

1. When creating a cluster, on the Cloud Storage page in the advanced cluster wizard view, select Use existing WASB storage and enter information related to your WASB account, as described in the instructions for configuring access to WASB.
2. Under Storage Locations, enable Configure Storage Locations by clicking the

   button.
3. Provide your existing container name under Existing Base Storage Location.

   Note:

   Make sure that the container already exists within your storage account.
4. Under Path for Hive Warehouse Directory property (hive.metastore.warehouse.dir), Cloudbreak automatically suggests a location within the container. You may optionally update this path or select Do not configure.

   Note:

   This directory structure will be created in your specified container upon the first activity in Hive.

Related Information

Configure access to WASB

Configuring access to GCS

Google Cloud Storage (GCS) is not supported as default file system, but access to data in GCS is possible via the gs connector. Use these steps to configure access from your cluster to GCS.

These steps assume that you are using an HDP version that supports the gs cloud storage connector (HDP 2.6.5 introduced this feature as a TP).

Related Information

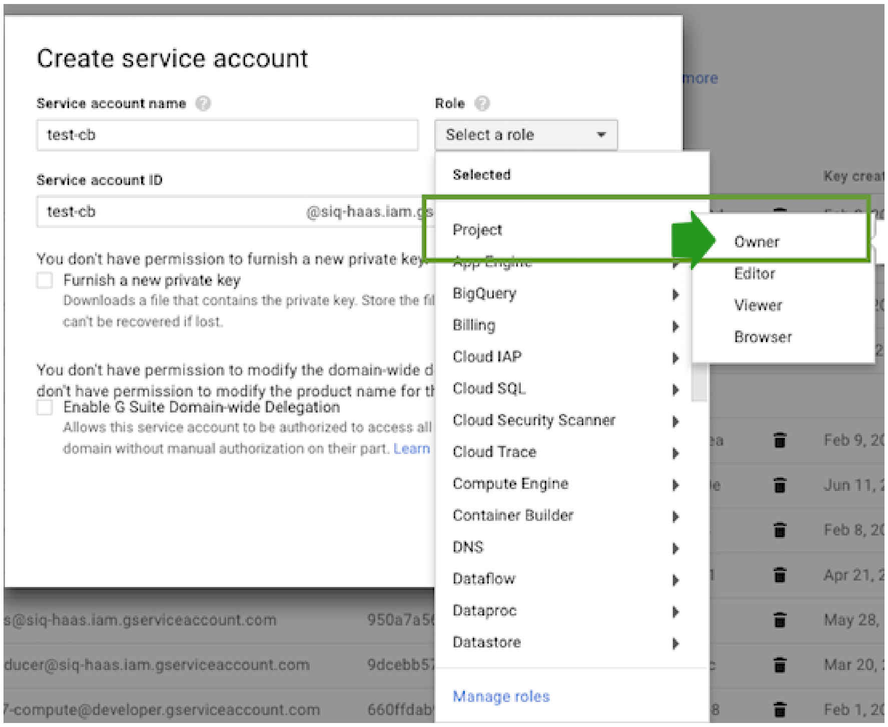
Cloud Storage Documentation (Google)

Prerequisites

Access to Google Cloud Storage is via a service account. The service account used for access to GCS must meet the minimum requirements.

The service account that you provide to Cloudbreak for GCS data access must have the following permissions:

- The service account must have the project-wide Owner role:



Create service account

Service account name: test-cb

Role: Select a role

Service account ID: test-cb @siq-haas.iam.gs

You don't have permission to furnish a new private key

□ Furnish a new private key

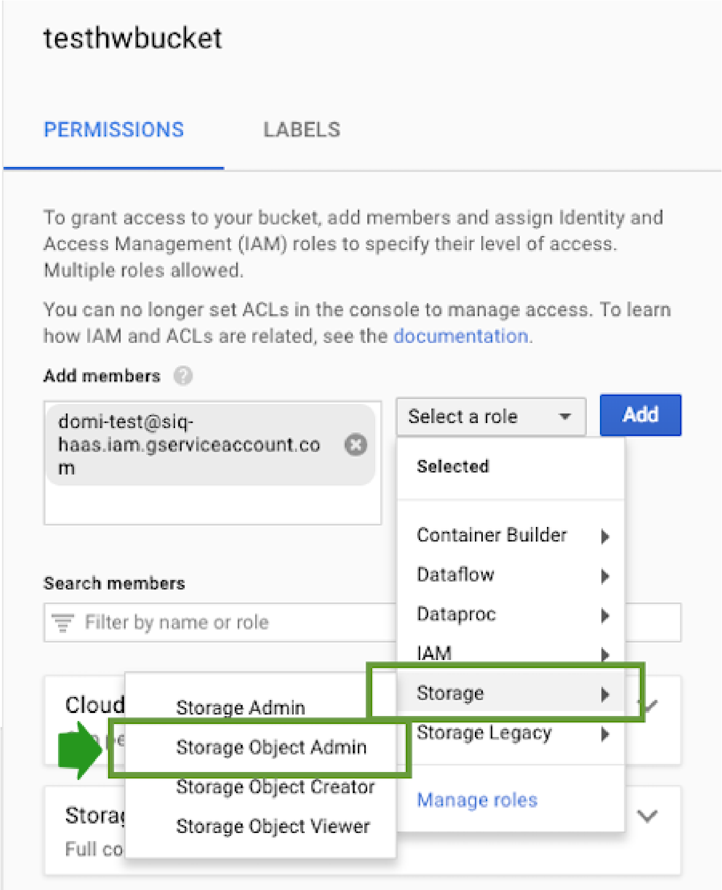
Downloads a file that contains the private key. Store the fil can't be recovered if lost.

You don't have permission to modify the domain-wide d don't have permission to modify the product name for th

□ Enable G Suite Domain-wide Delegation

Allows this service account to be authorized to access all domain without manual authorization on their part. Learn

- The service account must have the Storage Object Admin role for the bucket that you are planning to use. You can set this in the bucket's permissions settings:



testhwbucket

To grant access to your bucket, add members and assign Identity and Access Management (IAM) roles to specify their level of access. Multiple roles allowed.

You can no longer set ACLs in the console to manage access. To learn how IAM and ACLs are related, see the documentation.

Add members: domi-test@siq-haas.iam.gserviceaccount.com

For more information on service accounts and instructions how to create one, refer to Google documentation.

Related Information

Understanding service accounts (Google)

Creating and managing service accounts (Google)

Configure access to GCS

Access to Google Cloud Storage can be configured separately for each cluster by providing the service account email address.

Steps

1. On the Cloud Storage page in the advanced cluster wizard view, select Use existing GSC storage.
2. Under Service Account Email Address provide the email of the service account that you created as a prerequisite.
3. Once your cluster is in the running state, you should be able to access buckets that the configured storage account has access to.

Related Information

Prerequisites

Creating a cluster

Test access to GCS

Test access to the Google Cloud Storage bucket by running a few commands from any cluster node.

1. SSH to a cluster node and switch to the hdfs user by using sudo su hdfs.

2. Test access to your GCS bucket by running a few commands. For example, you can use the command listed below (replace “mytestbucket” with the name of your bucket):

   ```
   hadoop fs -ls gs://my-test-bucket/
   ```

For more information about configuring the gs connector for HDP and working with data stored on GCS, refer to Cloud Data Access HDP documentation.

Related Information

Cloud data access (HDP)

Configuring GCS storage locations

After configuring access to GCS via service account, you can optionally use a GCS bucket as a base storage location; this storage location is mainly for the Hive Warehouse Directory (used for storing the table data for managed tables).

Prerequisites

- You must have an existing bucket. For instructions on how to create a bucket on GCS, refer to GCP documentation.
- The service account that you configured must allow access to the bucket.

Steps

1. When creating a cluster, on the Cloud Storage page in the advanced cluster wizard view, select Use existing GCS storage and enter information related to your GCS account, as described in the instructions for configuring access to GCP.
2. Under Storage Locations, enable Configure Storage Locations by clicking the

   ○

   button.
3. Provide your existing directory name under Base Storage Location.

   Note:

   Make sure that the bucket already exists within the account.
4. Under Path for Hive Warehouse Directory property (hive.metastore.warehouse.dir), Cloudbreak automatically suggests a location within the bucket. You may optionally update this path or select Do not configure.

Note:

This directory structure will be created in your specified bucket upon the first activity in Hive.

Related Information

Configure access to GCS

Creating buckets (Google)