Connect Databricks to object storage platform

The object storage platform (OSP) is a high-performance object storage architecture designed with strong security. OSP is S3 API compatible, enabling easy data sharing with existing cloud services.

Note: Sample excerpt from cloud integration documentation (company details anonymized).

Why use Databricks to connect to OSP?

Databricks provides a simple compute framework for managing data workloads.

You can create clusters with AWS EC2 and use Databricks for data processing. The OSP is the data source and destination.

The advantages of using object storage with Databricks include:

- Lower costs compared to AWS S3
- Data (at rest) security in the private cloud
- Convenient data exchange with other cloud services
- API compatibility with AWS S3 for external data exchange
- AWS resources exchange data through Databricks without performance drops

Setting up Databricks

You can use an existing Databricks account to configure the connection to object storage.

Prerequisites

The requirements to connect object storage with Databricks include:

- A Databricks account on AWS with the permissions to create roles and S3 buckets.
- Access to private cloud region.
- An AWS IAM cross-account role ARN.

Creating a Databricks account

If you don’t have an account:

1. Follow the instructions on the Get Started with Databricks page in the Databricks documentation to create an account.
2. Select **Amazon Web Services** as the cloud service provider.
3. Click **Get Started.** Databricks sends a welcome message to your email inbox.

4. Verify your email address and create a Databricks password. On submit, the **Databricks console** opens.
5. Select the **Databricks on AWS** subscription plan, and then click **Continue**.

Databricks prompts you to create a **Workspace**.

Configure Workspace credentials

A workspace organizes Databricks objects such as notebooks, libraries, dashboards and provides access to data objects and computational resources.

Before you can create a workspace, you must first configure the following credentials:

- Define cross-account IAM role credentials to the private cloud region for application resources, such as a VPC.
- S3 bucket credentials in the private cloud region for workspace storage.

Important: The workspace that you configure must be in the same region as the S3 bucket.

Define cross-account IAM role credentials

If you don't have IAM roles already defined:

1. Follow the instructions in the Databricks documentation to Create a cross-account IAM role and to define an inline policy.
2. Take note of the **Role ARN**. You'll use it for new credential configuration.

If you do have IAM roles defined:

1. From **AWS IAM > Roles**, click a **Role name** in the list.
2. Copy the **Role ARN**. You'll use it when adding a new credential configuration.

Configure S3 bucket credentials

You must configure an S3 bucket for workspace storage. The S3 bucket is used as root storage for workplace objects, such as cluster logs and job results libraries.

To configure storage:

1. Follow the instructions to Define a storage configuration and generate a bucket policy in the Databricks documentation.
2. Ensure that you copy the storage policy from Databricks and add it to the S3 bucket policy.

Create the Workspace

Once you have completed creating credentials, you can create the workspace and then add the cross-account role ARN and S3 Bucket configurations

To create the workspace, from the **Databricks console**:

1. Select **Create a workspace**, and then select **Custom AWS configuration**. The **Create workspace** dialog displays.
3. Enter a **Workspace name.**
4. In **Region**, select the target AWS region.
5. In **Credential configuration**, select **Add a New Credential**.

Add Credential configuration

To add the cross-account role ARN, from the **Add Credential configuration** dialog:

1. Enter the **Credential configuration** name.
2. Type the **Role ARN**, and then click **Add**.
3. From the **Storage configuration** field, select **Add a Storage configuration**.

Add Storage configuration

In this step, you modify the AWS bucket policy by adding the Databricks policy.

1. Enter the **Storage configuration** name.
2. In **Bucket name**, enter the exact name of the S3 bucket.
3. Click **Generate** policy.
4. Copy the policy from Databricks and paste it into the AWS S3 bucket policy.
5. In the **Add Storage configuration** dialog, click **Add**.
6. Click **Save**.

The workspace takes several minutes to provision.

Create a Cluster

Create a cluster to maximize performance:

1. From the list of **Workspaces**, select a workspace.
2. Click **Open**, located next to the name of the workspace.
3. From the sidebar, select **Create** and then click **Cluster**.

Configure the Cluster

1. Enter the **Cluster Name**.
2. Select a **Databricks Runtime version**. As a rule, select the latest runtime version.
3. Select a **Worker type** according to your workload requirements.
4. Select the **number of Workers** according to your workload requirements.

5. Select the **Driver type**.

Configure advanced cluster options

To access AWS configurations and allow the underlying Spark software to interact with object storage:

1. Select **Advanced Options**.
2. In the **IAM role passthrough** section, click the **Spark** tab.
3. In the **Spark configuration** section, copy and paste the following code.

```
spark.hadoop.fs.s3a.bucket.access.key <object-storage-access-key>
spark.hadoop.fs.s3a.bucket.secret.key <object-storage-secret-key>
spark.hadoop.fs.s3a.bucket.endpoint <object-storage-endpoint>
```

4. Click **Create Cluster**.

Create a Notebook

Store job commands in a notebook. See Notebooks in the Databricks documentation for specific instructions on managing and using Notebooks.

To create a new notebook and attach it to your cluster:

1. From the sidebar, select **Create** and then click **Notebook**. The **Create Notebook** page displays.
2. Enter a **Name** for the new notebook.
3. Select a **Cluster** to attach to the notebook.
4. Click **Create**. The cluster resources are allocated to the notebook.

The cluster is now ready to interact with object storage.

Validate

To ensure that the cluster is correctly configured:

1. Write a small text file to object storage.
2. List the contents of the directory to validate that the file exists.