Building Data Lakes on AWS –

Lab 3: Working with Data as a Product

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at *AWS Training and Certification*.

Lab overview

You are a data engineer at Any Company. Your company is developing a new application that uses movie data. As part of the application development process, you were asked to set up permissions for two different types of customers: standard subscription and enterprise subscription. You set up these permissions using AWS Lake Formation tags (LF-tags) and then confirm if you grant the right access using Lake Formation tag-based access control (LF-TBAC) permissions.

In this lab, you view an AWS Glue job that maintains a processed dataset. You also configure access to support data discovery using LF-tags and then set up custom access for two different consumers. You use Amazon Athena to access the curated data product.

Objectives

By the end of this lab, you will be able to do the following:

View an AWS Glue job that maintains a dataset.

Define LF-tags and apply them to resources.

Grant LF-TBAC permissions to data consumers.

Verify consumer-specific data views using Athena.

ICON KEY

Various icons are used throughout this lab to call attention to different types of instructions and notes. The following list explains the purpose for each icon:

Note: A hint, tip, or important guidance.

Learn more: Where to find more information.

Consider: A moment to pause to consider how you might apply a concept in your own environment or to initiate a conversation about the topic at hand.

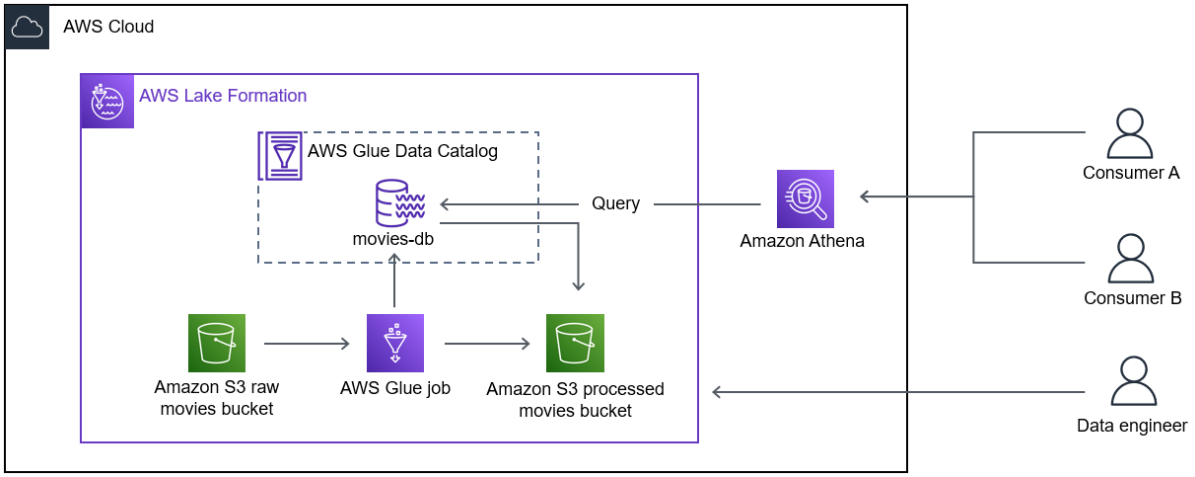
Refresh: A time when you might need to refresh a web browser page or list to show new information.

Hint: A hint to a question or challenge.

Answer: An answer to a question or challenge.

ENVIRONMENT OVERVIEW

The following diagram shows the basic architecture of the lab environment:



In the preceding diagram, an Amazon Simple Storage Service (Amazon S3) bucket containing raw movie data is registered to Lake Formation to create a data lake. The data is loaded in AWS Glue and is processed using an AWS Glue job. The AWS Glue job adds the data to the database. The database is connected to the Amazon S3 bucket. Two consumers query the Lake Formation database with Athena. Athena uses the AWS Glue Data Catalog and reads data from the database. A data engineer accesses and works with LF-tags to control the consumers' access.

Start lab

To launch the lab, at the top of the page, choose Start lab.

Caution: You must wait for the provisioned AWS services to be ready before you can continue.

To open the lab, choose Open Console.

You are automatically signed in to the AWS Management Console in a new web browser tab.

WARNING: Do not change the Region unless instructed.

COMMON SIGN-IN ERRORS

Error: You must first sign out

If you see the message, **You must first log out before logging into a different AWS account:**

Choose the **click here** link.

Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.

Choose Open Console again.

Error: Choosing Start Lab has no effect

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

Add the lab domain name to your pop-up or script blocker's allow list or turn it off.

Refresh the page and try again.

Task 1: Test and validate an AWS Glue job

You have an AWS Glue job that updates and appends new movie data in a data lake when any new movies are added to the underlying raw data. The AnyCompany application developers want you to view the job and validate the processed data before including the processed data in their application.

In this task, you view an AWS Glue job that maintains processed data in a data lake. You view the job definition in AWS Glue and then query the data using Athena.

TASK 1.1: TEST AN AWS GLUE JOB

An AWS Glue database, table, job, and trigger are already defined for the movies data. Test the AWS Glue job and review how the job transforms the data.

At the top of the AWS Management Console, in the search bar, search for and choose

AWS Glue.

In the left navigation pane, in the **Data Integration and ETL** section, choose **ETL Jobs**.

In the **Your jobs** section, choose **transform-movies**.

The **transform-movies** AWS Glue job appears.

Consider: Take a moment to read through the AWS Glue job script. How many nodes are contained in the job?

Each node completes a task. In this lab, there are eight nodes in the AWS Glue job. The nodes complete the following tasks:

Node 1: Inputs data from the **data/movies_csv/movies.csv** file in the S3 bucket.

Node 2: Inputs data from the AWS Glue Data Catalog.

Node 3: Fills missing data in the **rating** column and creates a **rating_filled** column using FillMissingValues from the **awsglueml.transforms** package.

Nodes 4 and 5: Applies mapping to the tables.

Node 6: Uses a custom PySpark transformation to append new rows to the movies data if there are any new rows added to the raw movies data.

Node 7: Selects the new rows dataframe from the collection returned by the custom transformation.

Node 8: Outputs data to the AWS Glue Data Catalog.

Choose the **Schedules** tab.

This AWS Glue job is scheduled using an AWS Glue trigger called **transform-movies-trigger**. The job is scheduled to run every 10 minutes and is defined by a cron expression.

Learn more: Refer to *AWS Glue triggers* in the *Additional resources* section for more information about scheduling AWS Glue jobs.

Choose the **Runs** tab.

The **Runs** tab shows all the recent job runs. You can run jobs automatically using a trigger, or manually when viewing an AWS Glue job. This job is set to run automatically based on a schedule.

Refresh: If a job does not appear or you want to see the current **Execution time**, choose the refresh icon.

Wait until at least one AWS Glue job has a **Run status** of **Succeeded**.

Note: Each **transform-movies** AWS Glue job takes 4-5 minutes to run.

Note: If any of the AWS Glue jobs errors with a max concurrency error, view the full list of AWS Glue jobs until you find the job that succeeded. This AWS Glue job only runs one job at a time.

You viewed the AWS Glue job and tested it by letting the trigger launch the AWS Glue job automatically.

TASK 1.2: VALIDATE AN AWS GLUE JOB

Now that you viewed your AWS Glue job, query the processed movie data using Athena.

At the top of the AWS Management Console, in the search bar, search for and choose

Athena.

Note: If the Athena **Get Started** menu appears, choose **Query your data with Trino SQL** option and then choose **Launch query editor**.

In the **Workgroup primary settings** window, choose **Acknowledge**.

In the **Data** section, the **AWSDataCatalog** data source and **transform-movies-db** database are automatically selected. In the **Tables** section, you see one table listed.

In the query editor, enter the following:

```
SELECT * FROM "transform-movies-db"."movies" limit 10;
```

Choose **Run**.

Ten records appear. There are 13 columns shown. As the engineer, you should see all the columns and records in the dataset.

Note: If records do not appear, the AWS Glue job has not finished yet. Wait until the AWS Glue job has finished and run the query again.

Next, count the number of rows in the dataset.

Choose the plus sign **+** to create a new query.

In the query editor, enter the following:

```
SELECT COUNT(*) AS number_of_movies FROM "movies";
```

Choose **Run**.

In the **Results** section, you see **4609** returned as the count for **number_of_movies**.

Congratulations! You successfully tested and validated an AWS Glue job.

Task 2: Define a set of LF-tags

AnyCompany wants the data tagged with several different keys, including the following keys with their corresponding values:

Environment: Development, Production

Customer: Regular, Enterprise

Confidential: True, False

You will use these keys and values later in the lab to control access to the movie database, table, and columns.

In this task, you define LF-tags in a Lake Formation data lake using your Lake Formation administrator permissions.

TASK 2.1: DEFINE AN LF-TAG

At the top of the AWS Management Console, in the search bar, search for and choose

AWS Lake Formation.

In the **Welcome to Lake Formation** popup window, make sure that **Add myself** is selected and then choose **Get started**.

Note: If you see **Read Only Admin is not supported.**

Go to settings under **Data catalog** and choose **Get started**.

choose **Save**.

In the left navigation pane, in the **Permissions** section, choose **LF-Tags and Permissions**.

Choose **Add LF-tag**.

For **Key**, enter

Confidential.

For **Values**, enter

True,False.

Choose **Add**.

Choose **Add LF-tag**.

You added an LF-tag with a key of **Confidential** and values of **True** and **False**.

CHALLENGE A: CREATE MORE LF-TAGS

Add two more LF-tags for **Environment** (Development, Production) and **Customer** (Regular, Enterprise).

Hint: Repeat the steps in the previous task to add more LF-tags.

Answer: Navigate **here** for a solution.

Congratulations! You successfully defined a set of LF-tags.

Task 3: Apply LF-tags to resources

In this task, you use the LF-tags to tag the database, table, and columns in a way that is consistent with AnyCompany's data sharing plan.

TASK 3.1: APPLY LF-TAGS TO A DATABASE

The movies database is ready for production. Add an LF-tag that provides access to the database for production consumers.

In the left navigation pane, in the **Data catalog** section, choose **Databases**.

Choose **transform-movies-db**.

Choose **Actions.**

Choose **Edit LF-tags**.

Choose **Assign new LF-Tag**.

For **Assigned keys**, choose **Environment**.

For **Values**, choose **Production**.

Choose **Save**.

A tag appears in the **LF-Tags** section for the **transform-movies-db (database)** resource.

CHALLENGE B: APPLY LF-TAGS TO A TABLE

The movies table is ready for production and is not confidential. Add LF-tags that provide access to the database for production consumers who have access to data that is not confidential.

Hint: The **Environment** tag of **Production** is inherited from the database, so you only need to add a **Confidential** tag of **False** to the table.

Answer: Navigate **here** for a solution.

TASK 3.2: APPLY LF-TAGS TO A COLUMN

AnyCompany wants only their enterprise customers to have access to the **rank** and **rating_filled** columns in the **movies** table. Edit the **Customer** LF-tag for the **rank** and **rating_filled** columns to only include enterprise customers.

In the left navigation pane, in the **Data catalog** section, choose **Tables**.

Choose **movies**.

Choose **Actions**.

Choose **Edit schema**.

Select
the **year**, **title**, **directors_0**, **genres_0**, **genres_1**, **running_time_secs**, **actors_0**, **actors_1**, **actors_2**, **directors_1**, and **directors_2** columns.

Choose **Edit LF-Tags**.

Choose **Assign new LF-Tag**.

For **Assigned keys**, choose **Customer**.

For **Values**, choose **Regular**.

Choose **Save**.

Deselect all the columns.

Select the **rank** and **rating_filled** columns.

Choose **Edit LF-Tags**.

Choose **Assign new LF-Tag**.

For **Assigned keys**, choose **Customer**.

For **Values**, choose **Enterprise**.

Choose **Save**.

Choose **Save as new version**.

Only enterprise customers can view the **rank** and **rating_filled** columns.

Congratulations! You successfully added LF-tags to database, table, and column resources.

Task 4: Create LF-tag permissions

In this task, you establish the data sharing policy by creating LF-tag permissions.

There are two test consumers you will use to check if the regular and enterprise customers have the correct permissions. **Consumer_A** represents a regular customer and **Consumer_B** represents an enterprise customer.

When you are done, each user should have the following permissions:

Key	Data Engineer	Consumer_A	Consumer_B
Environment	Development, Production	Production	Production
Customer	Regular, Enterprise	Regular	Regular, Enterprise
Confidential	True, False	False	False

TASK 4.1: REVOKE IAM-BASED ACCESS TO AWS GLUE DATA CATALOG RESOURCES

Revoke AWS Identity and Access Management (IAM) based access to the database and table. Removing the IAM allowed principals from the data permissions restricts data access to the LF-tags.

In the left navigation pane, in the **Permissions** section, choose **Data lake permissions**.

Choose one of the **IAMAllowedPrincipals** lines.

Choose **Revoke**.

Choose **Revoke**.

Choose the other **IAMAllowedPrincipals** line.

Choose **Revoke**.

Choose **Revoke**.

You revoked the IAM permissions.

TASK 4.2: CREATE LF-TAG PERMISSIONS FOR A CONSUMER

Grant **Consumer_A** access to production databases.

Choose **Grant**.

For **IAM users and roles**, select **Consumer_A**.

Choose **Add LF-Tag key-value pair**.

For **Key**, choose **Environment**.

For **Values**, choose **Production**.

In the **Database permissions** section, for **Database permissions**, choose **Describe**.

Choose **Grant**.

Consumer_A has access to production databases.

Grant **Consumer_A** access to data that is not confidential and is for regular customers.

Choose **Grant**.

For **IAM users and roles**, select **Consumer_A**.

Choose **Add LF-Tag key-value pair**.

For **Key**, choose **Confidential**.

For **Values**, choose **False**.

Choose **Add LF-Tag key-value pair**.

For **Key**, choose **Customer**.

For **Values**, choose **Regular**.

In the **Table permissions** section, for **Table permissions**, choose **Select** and **Describe**.

Choose **Grant**.

Consumer_A has access to tables that are not confidential and are for regular customers.

When you add two LF-tags in one permission, those tags act as an *AND*, not an *OR*. **Consumer_A** only has access to tables

where **Confidential=False** *AND* **Customer=Regular**. This means **Consumer_A** is able to see 11 of the 13 columns in the movies table.

Learn more: Refer to *Lake Formation tag-based access control permissions model* in the *Additional resources* section for more information about describe and associate permissions.

CHALLENGE C: CREATE LF-TAG PERMISSIONS FOR ANOTHER CONSUMER

Grant describe permissions for **Consumer_B**, an enterprise customer in the production environment who does not have access to confidential data.

Hint: Set up **Consumer_B** the same way as **Consumer_A**, changing only the values chosen for the **Customer** key.

Hint: Choose the **Regular** and **Enterprise** values for the **Customer** key.

Answer: Navigate **here** for a solution.

Congratulations! You successfully created LF-tag permissions.

Task 5: Verify consumer-specific data views

The movies table, as part of the Lake Formation data lake, is a product available to all users and roles that have been granted permissions. You created different access permissions for two sets of consumers, represented in this lab by users **Consumer_A** and **Consumer_B**. The purpose of this task is to verify that both consumers have access to the data product as targeted for their consumer type.

In this task, you verify the consumer-specific data views for two test consumers using Athena. To test the LF-TBAC, check if **Consumer_A** can see the database and table but not the **rank** and **rating_filled** columns. Then, check if **Consumer_B** can see the database, table, and all columns.

TASK 5.1: VERIFY THE DATA VIEW FOR THE FIRST CONSUMER

Verify that **Consumer_A** can see the database and table but not the **rank** and **rating_filled** columns.

On the top-right of the screen, choose your username.

Choose **Sign out**.

Copy the **LoginURL** value listed to the left of these instructions.

Paste the **LoginURL** into your browser tab.

The AWS login page appears.

For **IAM user name**, enter

Consumer_A.

For **Password**, copy and paste the **AdministratorPassword** listed to the left of these instructions.

Choose **Sign in**.

You are now signed in as the **Consumer_A** user.

Note: Make sure the AWS Region listed in the console matches the **Region** value listed to the left of these instructions.

At the top of the AWS Management Console, in the search bar, search for and choose

`Athena`.

Note: If the Athena **Get Started** menu appears, choose **Query your data with Trino SQL** option and then choose **Launch query editor**.

In the **Workgroup primary settings** window, choose **Acknowledge**.

Choose the plus sign **+** to create a new query.

In the query editor, enter the following:

```
SELECT * FROM "transform-movies-db"."movies" limit 10;
```

Choose **Run**.

Ten records appear. There are 11 columns shown. As **Consumer_A**, a regular customer, you cannot see the **rank** and **rating_filled** columns. You have successfully tested the tag-based permissions applied to **Consumer_A**.

TASK 5.2: VERIFY THE DATA VIEW FOR SECOND CONSUMER

Verify that **Consumer_B** can see the database, table, and all columns.

On the top-right of the screen choose your username.

Choose **Sign out**.

Copy the **LoginURL** value listed to the left of these instructions.

Paste the **LoginURL** into your browser tab.

The AWS login page appears.

For **IAM user name**, enter

`Consumer_B`.

For **Password**, copy and paste the **AdministratorPassword** listed to the left of these instructions.

Choose **Sign in**.

You are now signed in as the **Consumer_B** user.

Note: Make sure that the AWS Region listed in the console matches the **Region** value listed to the left of these instructions.

At the top of the AWS Management Console, in the search bar, search for and choose

Athena.

Note: If the Athena **Get Started** menu appears, choose **Query your data with Trino SQL** option and then choose **Launch query editor**.

In the **Workgroup primary settings** window, choose **Acknowledge**.

Choose the plus sign + to create a new query.

In the query editor, enter the following:

```
SELECT * FROM "transform-movies-db"."movies" limit 10;
```

Choose **Run**.

Ten records appear. There are 13 columns shown. As **Consumer_B**, an enterprise customer, you can see all of the columns, including **rank** and **rating_filled**. You have successfully tested the tag-based permissions applied to **Consumer_B**.

Congratulations! You successfully verified the consumer-specific data views for both consumers.

Conclusion

Congratulations! You now have successfully:

Viewed an AWS Glue job that maintains a dataset.

Defined LF-tags and applied them to resources.

Granted LF-TBAC permissions to data consumers.

Verified consumer-specific data views using Athena.

End lab

Follow these steps to close the console and end your lab.

Return to the **AWS Management Console**.

At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.

Choose End lab and then confirm that you want to end your lab.

Additional resources

AWS Glue triggers

Assigning LF-Tags to Data Catalog resources

Overview of data filtering

Lake Formation tag-based access control permissions model

Appendix

CHALLENGE A SOLUTION

Add an LF-tag for **Environment** (Development, Production).

Choose **Add LF-tag**.

For **Key**, enter

`Environment`.

For **Values**, enter

`Development,Production`.

Choose **Add**.

Choose **Add LF-tag**.

Then, add an LF-tag for **Customer** (Regular, Enterprise).

Choose **Add LF-tag**.

For **Key**, enter

`Customer`.

For **Values**, enter

`Regular,Enterprise`.

Choose **Add**.

Choose **Add LF-tag**.

You successfully created two more LF-tags.

To continue this lab, go to Task 3.

CHALLENGE B SOLUTION

Add LF-tags that provide access to production consumers who have access to data that is not confidential.

In the left navigation pane, in the **Data catalog** section, choose **Tables**.

Choose **movies**.

Choose **Actions**.

Choose **Edit LF-tags**.

The **Environment** tag of **Production** is inherited from the database, so you only need to add a **Confidential** tag of **False**.

Choose **Assign new LF-Tag**.

For **Assigned keys**, choose **Confidential**.

For **Values**, choose **False**.

Choose **Save**.

Two tags appear in the **LF-Tags** section for the **movies (table)** resource.

Consider: Take a moment to think about how you can use inheritance with LF-tags in your own data lakes to accurately tag databases, tables, and columns.

Learn more: Refer to *Assigning LF-Tags to Data Catalog resources* in the *Additional resources* section for more information about assigning LF-tags to resources.

You successfully added LF-tags that provide access to production consumers who have access to data that is not confidential.

To continue this lab, go to Task 3.2.

CHALLENGE C SOLUTION

Grant **Consumer_B** access to production databases.

Choose **Grant**.

For **IAM users and roles**, select **Consumer_B**.

Choose **Add LF-Tag**.

For **Key**, choose **Environment**.

For **Values**, choose **Production**.

In the **Database permissions** section, for **Database permissions**, choose **Describe**.

Choose **Grant**.

Consumer_B has access to production databases.

Grant **Consumer_B** access to data that is not confidential and is for enterprise customers.

Choose **Grant**.

For **IAM users and roles**, select **Consumer_B**.

Choose **Add LF-Tag**.

For **Key**, choose **Confidential**.

For **Values**, choose **False**.

Choose **Add LF-Tag**.

For **Key**, choose **Customer**.

For **Values**, choose **Enterprise** and **Regular**.

Note: A principal that is granted permissions on a LF-tag with multiple values can access AWS Glue Data Catalog resources with either of those values

In the **Table permissions** section, for **Table permissions**, choose **Select** and **Describe**.

Choose **Grant**.

Consumer_B has access to tables where **Confidential=False** *AND* (**Customer=Enterprise** *OR* **Customer=Regular**). This means **Consumer_B** can see all the columns in the movies table.

You successfully granted permissions for **Consumer_B**.

To continue this lab, go to Task 5.