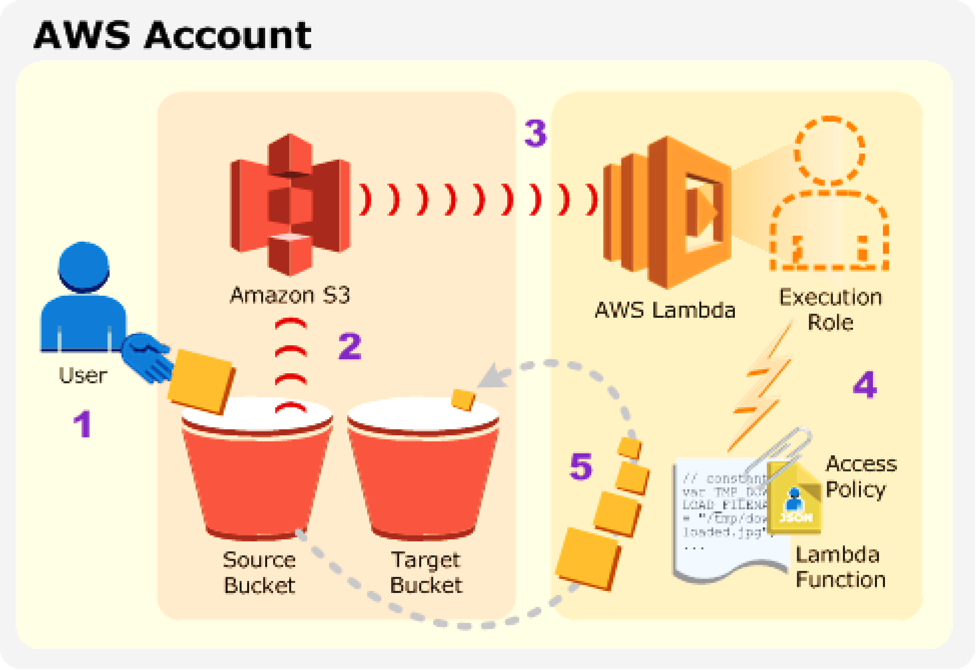
Please use your free tier AWS account to complete this lab

Lab - Using AWS Lambda with Amazon S3

Introduction: - In this lab we will create a sample Lambda function that creates thumbnails of the pictures (.jpg and .png) uploaded in an s3 bucket and saves them in another s3 bucket.

The following diagram illustrates the application flow:



1. A user uploads an object to the source bucket in Amazon S3 (object-created event).

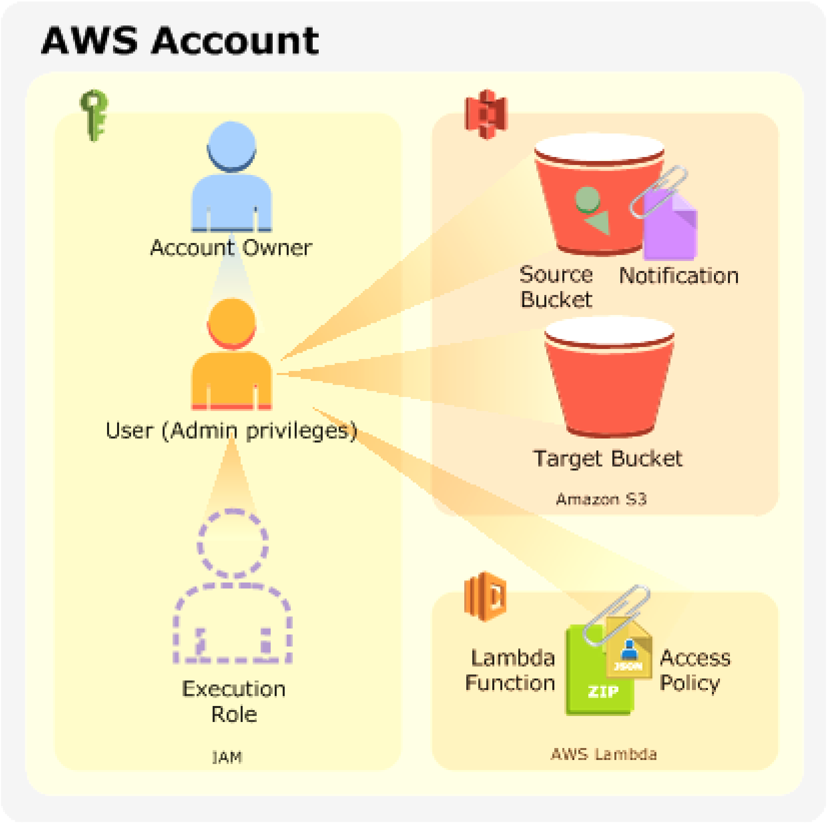
2. Amazon S3 detects the object-created event.

3. Amazon S3 publishes the s3:ObjectCreated:* event to AWS Lambda by invoking the Lambda function and passing event data as a function parameter.

4. AWS Lambda executes the Lambda function by assuming the execution role that you specified at the time you created the Lambda function.

5. From the event data it receives, the Lambda function knows the source bucket name and object key name. The Lambda function reads the object and creates a thumbnail using graphics libraries, and saves it to the target bucket.

AWS Resources Required: - You will have to create the following Amazon S3, Lambda, and IAM resources in your account to accomplish this lab.



1. IAM role

2. Two S3 buckets (one for holding original picture and anther to hold the thumbnails)

3. Lambda function.

***** Please ensure to use us-west-2 (Oregon) region to do this lab. *****

We will complete the scenario in below steps

Step 1 – Creating a Role that will allow your Lambda function to talk to s3 buckets.

Step 2 – Creating a bucket to host the original pictures and another to hold the thumbnails.

Step 3 – Create Lambda function.

Step 4 – Verifying the function.

Step 5 – Clean up post lab completion.

Let us get started.

Creating Lab:

Step 1 – Creating a Role that will allow your Lambda function to talk to s3 buckets

Go to IAM and create a role as below.

- Select type of trusted entity – Lambda
- Attach permissions policies - AmazonS3FullAccess
- Role name – lambda-execution-role (or any name you want)

Step 2 – Creating a bucket to host the original pictures and another to hold the thumbnails

Ensure both buckets are created in Oregon region and with default options.

- Create a bucket with any name you want "*bucketname*".
- Create another bucket with name "*bucketname*-resized".

Step 3 – Create a Lambda function

Step 3.1 – creating lambda function

Click on Lambda > Create a function > Author from scratch

- Name – "my-lambda-function" (or any name)
- Runtime – Python 3.6
- Role – Choose an existing role, (choose the one you created in the Step 01)

Click on create function, you should see a congratulations message on the next screen.

We have just created a name/placeholder for our function and selected desired platform, now let us supply the code.

Scroll to the **Function Code** section

- Code entry type – Upload a file from Amazon S3.
- Runtime – Python 3.6
- Handler – CreateThumbnail.handler
- S3 link URL – https://s3-us-west-2.amazonaws.com/us-west-2-aws-training/awsu-spl/spl-88/scripts/CreateThumbnail.zip

Click on Save on the top of the screen with rest all fields left as default. You should go through the other fields for educating yourselves later.

Step 3.2 – Creating a trigger

On the left side of the Designer section you will see the Add Triggers option, scroll down to select S3.

Now in the Configure triggers section.

- Bucket – select your bucket (not the one with resized in the name)
- Event type – Object Created (All)

Leave other fields as default and click on Add at the bottom of the screen and then save on top of the screen.

Step 4 – Verifying the function

- Upload a .jpg or .png objects to the source bucket using the Amazon S3 console.
- Verify that the thumbnail was created in the "*bucketname*-resized" bucket.
- Check the size difference between the original and resized object.

Congratulations, you have just created, a server less application as lambda function that gets triggered by an event in an s3 bucket.

You may now go to the monitoring tab and try to understand the various graphs shown on the console page.

Take two snapshots of your browser -

1- Of your S3 page showing both your buckets.
2- Of your "*-resized" bucket showing the thumbnail created.

You can upload these in git and attach them while submitting this lab completion on TopGear.

Step 5 – Clean up post lab completion.

- Delete both the buckets along with their objects.
- Delete the role you created.
- Delete the lambda function.

Disclaimer – this lab document is a simplified version of a tutorial published by AWS on their portal. We encourage you to visit the following link to know more in details when you get time.

https://docs.aws.amazon.com/lambda/latest/dg/with-s3-example.html?shortFooter=true

Lab Complete.