**Final Project: S3 Delete Trigger Lambda**

User Guide by: Jaired Lyons

# **Section 1: Review and Preparation**

For this project, as long as Labs # 17-18, you will not need to create or assign any new roles of permissions. The following will check that you have everything from those that is relevant to this project.

1. Log in to the AWS Console using your **Cloud9Administrator** account.
2. Go to the IAM service. (If you are not allowed access, log in with your **Root** account)
3. Click on User Groups then “Cloud9Administrators”
4. From there, click on the “Permissions” tab.
5. A screenshot of a web page

   Description automatically generated with medium confidenceReview your permissions, you should see “IAMFullAccess” and “AWSLambda\_FullAccess” (among others).
6. If you are missing one or more, use “Add permissions”-> Attach policies to attach them directly.
7. Next, to ensure you have a role that can interact with S3 and Lambda (which will be used later) click on “Roles” on the sidebar.
8. If you still have the role “S3ReadWriteLambdaLab17” then you can skip to the next section.
9. Click on “Create Role”
10. Choose Service “Lambda”
11. A screenshot of a computer

    Description automatically generated with medium confidencePress “Next”
12. A screenshot of a computer

    Description automatically generated with low confidenceSearch “S3” and select “AmazonS3FullAccess”.
13. Clear pervious search and search for “S3ObjectLambda”, scroll until you find “AmazonS3ObjectLambdaExecutionRolePolicy” and select it.
14. A screenshot of a computer

    Description automatically generated with medium confidencePress “Next”
15. Call the role “S3ReadWriteLambda”.
16. Scroll down and press “Create Role”.
17. Now, using the AWS Console, go to SES.
18. Click on “Verified Identities” on the left-hand side.
19. A screenshot of a phone

    Description automatically generated with low confidenceEnsure you have at least two verified identities for this project.

**Section 2: Create S3 Bucket**

1. Go to S3
2. Click on “Create bucket”
3. Call the bucket *s3-delete-[yourinitials]*. I would call mine *s3-delete-jl*. (Remember, bucket names need to be lower-cases, no-spaces, and all bucket names across AWS need to be distinct)

A screenshot of a computer

Description automatically generated with medium confidence

1. Leave the rest of the default values, then click “Create bucket”
2. Upload some files into this bucket. These files can be of any type and will be used to test the Lambda function later.

**Section 3: Create Lambda Function**

1. Go to Lambda
2. Click on the “Create Function” button.
3. A screenshot of a computer

   Description automatically generated with medium confidenceMake sure the “Author From Scratch” button is selected
4. Enter “fpDelete” for the name, and Python 3.7 for the Runtime.
5. Under “Change default execution role” select “Use an existing role”
6. A screenshot of a computer

   Description automatically generated with medium confidenceFor existing role, ensure your “S3ReadWriteLambda” is selected (Lab17 version is alright if you have it.
7. Press the “Create function” button.

**Section 4: Give Lambda Trigger**

1. In the Designer Window, select “+Add trigger”

A screenshot of a computer

Description automatically generated with medium confidence

1. Under “Trigger configuration” section, make sure you select “S3” from the dropdown and then bucket s3-delete-[yourinitials].
2. A screenshot of a computer

   Description automatically generated with medium confidenceUnder Event types, uncheck “All object create events” and check “All object delete events”
3. Check the box under Recursive invocation.
4. Note: S3 considers moving and renaming to fall under delete events

**Section 5: Setting up code**

1. Return to the code page of your function.
2. Click the Upload button, selecting “.zip file”, browse to the finaljl.zip file that is included in this project and click Save.
3. You should get a message that your code is being updated, click “OK”.
4. You can now edit the new lambda\_function.py file that was uploaded.
5. Looking at the code, the latter half may remind you of the joke Email Lab 18.
6. Most of the code from there carries over, but with a few new alterations.
7. First, each line now ends in a ‘\n’ which will create a new line in the automated email.
8. A picture containing text, font, screenshot, line

   Description automatically generatedThe second are a few new lines that will alert you of what file was deleted from what bucket.
9. The next big thing you’re likely to notice (and that is relevant to the previous statement) are the two event.get statements grabbing “Records”. These will both grab a large JSON string (a version of which is viewable in the .zip file the code is stored in). Most importantly, this JSON file contains the name of the file deleted and what bucket it was deleted from. This means you could add multiple bucket triggers to this function and know which specific one got something deleted from it.
10. A picture containing text, font, screenshot, white

    Description automatically generatedBut we have no use for the raw JSON besides getting the correct subscripts for the next step. The reason these two translating statements are walled behind an if-statement is to avoid the error “typeerror: 'nonetype' object is not iterable” when you try to test it in lieu of any actual problems.
11. Inside said if statement are the two functions that will parse the data to the right endpoint to give the desired results.
12. With all this automation, the only things you need to change are the name (to your name) and the two email addresses to two of your verified email addresses.
13. Now save, deploy, and test your function. After you receive the test email, try deleting something from your s3-delete-[yourinitials] bucket.

A screenshot of a computer

Description automatically generatedThe following are two example emails from this project. The first is from Testing the function after Deploying it. The second is from deleting something.

A screenshot of a computer program

Description automatically generated with medium confidence