SERVERLESS SPEECH-TO-TEXT WITH AWS TRANSCRIBE AND S3 EVENT TRIGGER USING LAMBDA AND CLOUDWATCH

M.LOHITH

2100032244

SEC-31

INTRODUCTION:

- Speech-to-text technology, also known as speech recognition or automatic speech recognition (ASR), is a technology that enables machines to convert spoken words into written text. This technology uses algorithms to analyze audio signals and identify the words and phrases being spoken.
- Speech-to-text technology has become increasingly important in modern technology because it enables people to communicate with computers and other devices in a more natural and efficient way. It allows users to interact with their devices using spoken commands, which can be especially useful for people who have difficulty typing or using a traditional keyboard.

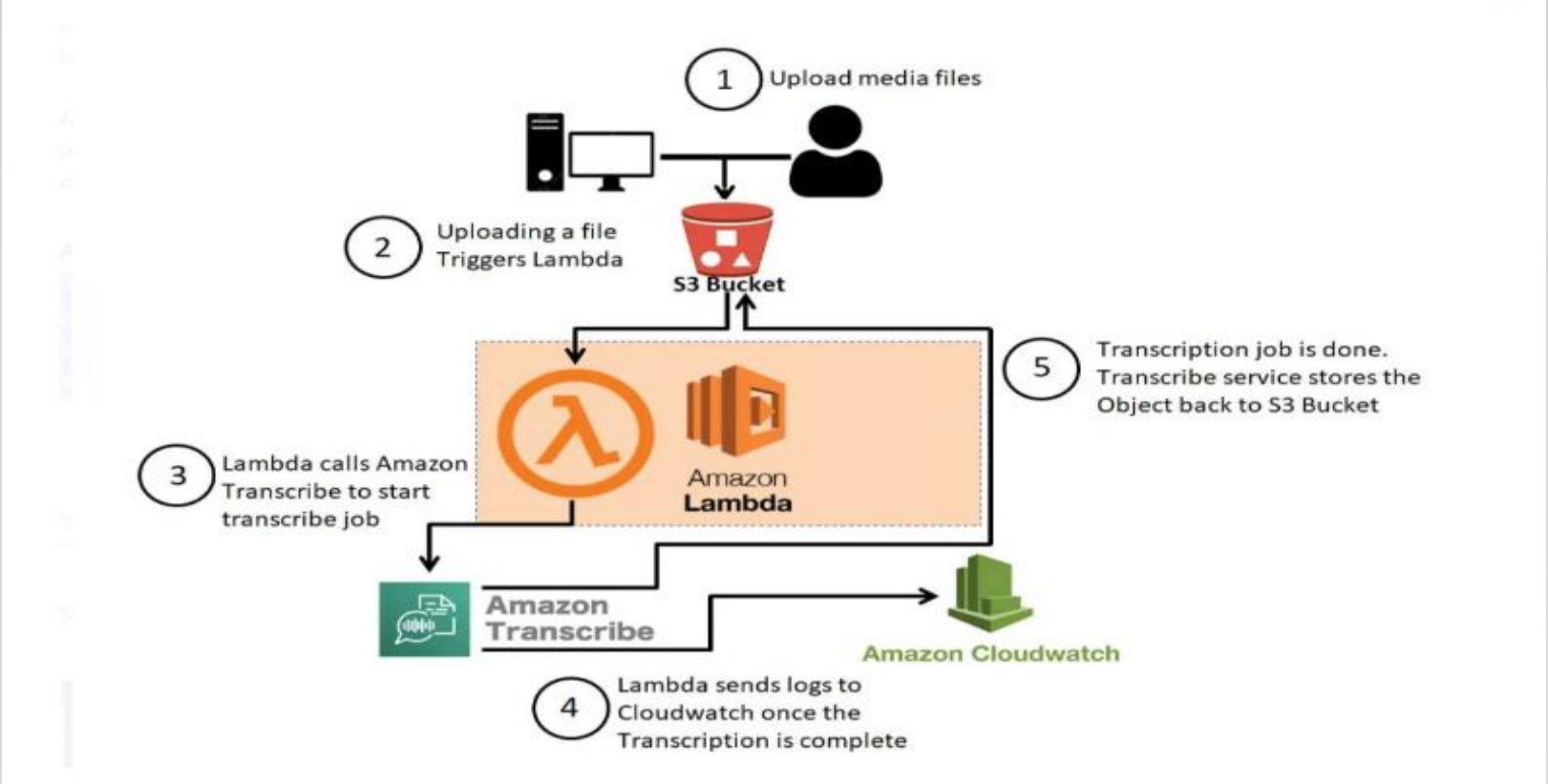
OBJECTIVE

The objective of this experiment is to showcase how AWS services can be utilized to automatically transcribe speech from an audio file placed into an S3 bucket by triggering a Lambda function. This provides a serverless and automated solution for speech-to-text conversion.

AGENDA

1. Introduction to AWS services used in the experiment
2. Setting up an S3 bucket to store audio files
3. Creating a Lambda function to trigger a Transcribe job
4. Configuring CloudWatch to monitor and log the entire process
5. Demonstration of the complete serverless speech-to-text solution.

ARCHITECTURE



STEPS

- Create a s3 bucket to upload audio file
- Create a lambda function to write the code for transcribe service
- Integrate S3 service with Lambda: As soon as some event occurs in S3 we want the lambda function will run.
- Add a trigger to our lambda function
- Chekcing the logs in CloudWatch
- Create a IAM role for our lambda function
- Upload the file in s3 and automatically job will created in transcribe.
- One folder is added to our s3 bucket after uploading the audio file
- Now click on that folder in that folder one json document is there in that text extracted from audio will be stored.

EXPERIMENT SETUP:

- The experiment setup involves the following steps:

I.Create an S3 bucket: The first step is to create an S3 bucket where the audio files will be stored.

2.Configure AWS Transcribe: AWS Transcribe is a service that converts speech to text. We need to configure AWS Transcribe to process the audio files.

3.Create a Lambda function: We will create a Lambda function that will trigger whenever an object is placed into the S3 bucket.Specifically, we will trigger a job with AWS Transcribe to transcribe the speech from an audio file placed into S3.

4. Configure S3 bucket event trigger: We will configure an S3 bucket event trigger to invoke the Lambda function whenever an object is created in the S3 bucket.

5. Configure CloudWatch logging: We will configure CloudWatch to log the output of the Lambda function, which will contain the transcribed text.

6. Test the setup: We will upload an audio file to the S3 bucket and verify that the Lambda function is triggered, AWS Transcribe processes the audio file, and the transcribed text is logged to CloudWatch.

SOCIAL MEDIA LINKS

Github Link:

https://github.com/lohithmaturi/Cloud-Sereverless-Project

YouTube Link:

https://youtu.be/3jKug-mxHBU?si=dxEryRomJPCjWNDZ

LinkedIn Article Link:

https://www.linkedin.com/pulse/aws-serverless-speech-to-text-transcribe-s3-event-trigger-maturi-hyrkc/?trackingId=tSVyUuhgReGHuf4lbLTxmw%3D%3D

RESULTS AND CONCLUSION

- The experiment results will demonstrate the successful conversion of speech to text using AWS Transcribe and the other AWS services. The benefits of using a serverless architecture for speech-to-text conversion include cost savings, scalability, and ease of deployment. However, the limitations of the experiment include the possibility of delays due to the asynchronous nature of AWS Transcribe and the potential for increased latency due to the use of multiple services.
- In conclusion, this experiment provides a practical example of how to use AWS services to perform speech- to-text conversion in a serverless manner. Further, the use of serverless architecture reduces the cost and complexity of implementing speech-to-text functionality in real-world applications. The future scope of this experiment includes exploring other AWS services and integrating with other platforms to enhance the functionality of speech-to-text conversion.