Serverless Voice-to-Text Converter using AWS

Sem End Project Report

Cloud & Serverless Computing
Department of Computer Science and Engineering

By

Bodepudi Jothirmaye

2200032386

Section-33

Advanced

under the supervision of

Ms. Sai Durga Tejaswi

Koneru Lakshmaiah Education Foundation

(Deemed to be University estd., u/s 3 of UGC Act 1956)

Green Fields, Vaddeswaram, Guntur (Dist.), Andhra Pradesh – 522302

December, 2024

CONTENTS

LIST OF FIGURES

ABSTRACT

The rapid advancement of cloud computing and artificial intelligence has paved the way for scalable, efficient, and cost-effective solutions for voice processing. This project presents a Serverless Voice-to-Text Converter leveraging Amazon Web Services (AWS). The system utilizes AWS Lambda for serverless execution, Amazon S3 for audio file storage, and Amazon Transcribe for converting speech to text with high accuracy. By adopting serverless architecture, the solution eliminates the need for infrastructure management, ensures automatic scaling, and reduces operational costs. The workflow is initiated when a user uploads an audio file to an S3 bucket, triggering a Lambda function that invokes Amazon Transcribe. Once the transcription is complete, the resulting text is stored back in S3 or forwarded to other services as needed. This architecture provides a lightweight, scalable, and real-time solution suitable for applications such as automated meeting notes, voice-command processing, and transcription services.

INTRODUCTION

In the digital age, voice data has become an integral part of human-computer interaction, with applications ranging from virtual assistants to real-time transcription services. Converting voice to text not only enhances accessibility but also enables better data analysis, indexing, and storage. Traditional approaches to voice processing often require significant infrastructure, continuous server management, and high maintenance costs.

To address these challenges, this project explores the implementation of a Serverless Voice-to-Text Converter using Amazon Web Services (AWS). By utilizing a serverless architecture, the solution automatically scales with user demand, reduces idle resource usage, and offers a pay-as-you-go pricing model. This ensures both cost-efficiency and high availability.

The system is designed using key AWS services:

- Amazon S3 to store audio input and transcribed output,
- AWS Lambda to automate processing without provisioning servers, and
- Amazon Transcribe to perform accurate, real-time voice-to-text conversion.

This approach makes the application ideal for real-world scenarios such as customer service automation, lecture transcription, and voice-controlled applications. The following sections detail the system design, implementation, and performance analysis of the proposed serverless voice-to-text solution.

LITERATURE REVIEW

Voice recognition and speech-to-text conversion have been extensively studied over the past few decades. With the growth of cloud computing and artificial intelligence, modern systems aim to provide high-accuracy transcription while minimizing infrastructure overhead.

Several traditional speech recognition systems rely on on-premise servers and custom models, which often involve high maintenance costs, limited scalability, and complexity in deployment. Tools like **Google Speech-to-Text API**, **IBM Watson Speech to Text**, and **Microsoft Azure Speech Services** have emerged to offer cloud-based solutions. However, these still require developers to manage backend infrastructure for orchestration and scaling.

Recent advancements in **serverless computing** have transformed the deployment model of such applications. According to Jonas et al. (2019), serverless architecture, particularly **Function-as-a-Service (FaaS)** platforms like **AWS Lambda**, provide greater scalability, automatic resource allocation, and significant cost savings by billing only for actual usage.

Amazon Transcribe, launched by AWS, is a fully managed automatic speech recognition (ASR) service. Studies and technical benchmarks have shown it performs well in noisy environments and supports multiple languages, speaker identification, and custom vocabulary – making it suitable for a wide range of applications (AWS Documentation, 2022).

Research by Sharma et al. (2021) demonstrated the efficiency of integrating AWS Transcribe with other AWS services like Lambda and S3 to build scalable transcription workflows. The combination of these services allows for **event-driven architectures**, where actions are triggered automatically based on user interactions such as uploading a file.

Furthermore, the serverless approach has proven valuable in reducing the complexity of application deployment, as shown in case studies involving real-time transcription and media processing. It allows developers to focus on core functionality rather than infrastructure management.

In summary, existing literature supports the effectiveness of using serverless architectures for speech-to-text conversion, and AWS offers a mature ecosystem for implementing such solutions efficiently.

PROBLEM STATEMENT

With the increasing reliance on digital communication and multimedia content, the demand for efficient and accurate speech-to-text solutions has grown significantly. Traditional voice-to-text systems often require substantial computational resources, continuous server maintenance, and complex infrastructure setup, making them unsuitable for small-scale or rapidly scaling applications.

Moreover, existing solutions can be costly and inflexible, especially when handling fluctuating workloads. This limits accessibility for developers and organizations seeking lightweight, scalable, and cost-effective voice processing solutions.

The primary problem addressed in this project is:

How can we design a scalable, cost-efficient, and low-maintenance voice-to-text conversion system using a serverless architecture on AWS?

This project aims to overcome the limitations of traditional architectures by leveraging serverless computing through AWS services such as Amazon S3, AWS Lambda, and Amazon Transcribe. The goal is to develop a system that can automatically transcribe uploaded audio files into text with minimal operational overhead, while ensuring high availability, scalability, and accuracy.

TECHNICAL IMPLEMENTATION

The technical implementation of the Serverless Voice-to-Text Converter leverages core AWS services to create a fully automated, scalable, and efficient transcription pipeline. The architecture follows an event-driven model, where services interact without the need for constant server management. Below is a breakdown of the key components and workflow:

1. Amazon S3 (Simple Storage Service)

Amazon S3 is used as the central storage for both audio input files and the transcribed output text. A specific S3 bucket is created to store the uploaded audio files. When a file is uploaded, it triggers a Lambda function via an S3 event notification.

2. AWS Lambda

AWS Lambda functions serve as the serverless compute layer of the system. These functions are triggered automatically when an audio file is uploaded to the S3 bucket. The Lambda function handles the following tasks:

- Reads the file metadata (format, path, etc.)
- Invokes the Amazon Transcribe API with the audio file location
- Monitors the transcription job until completion
- Retrieves the transcribed text from the output location
- Saves the transcription result to another S3 bucket or database

Lambda enables parallel processing of multiple files and scales automatically based on incoming workloads.

3. Amazon Transcribe

Amazon Transcribe is the core service responsible for converting speech into text. It supports various audio formats like .mp3, .mp4, .wav, and .flac. When the transcription job is initiated by Lambda, Transcribe processes the audio file stored in S3 and returns a JSON file containing the transcription.

Features of Amazon Transcribe:

- Automatic punctuation and formatting
- Speaker identification (diarization)
- Language identification and multiple language support

- Custom vocabulary support for domain-specific terms

4. IAM (Identity and Access Management)

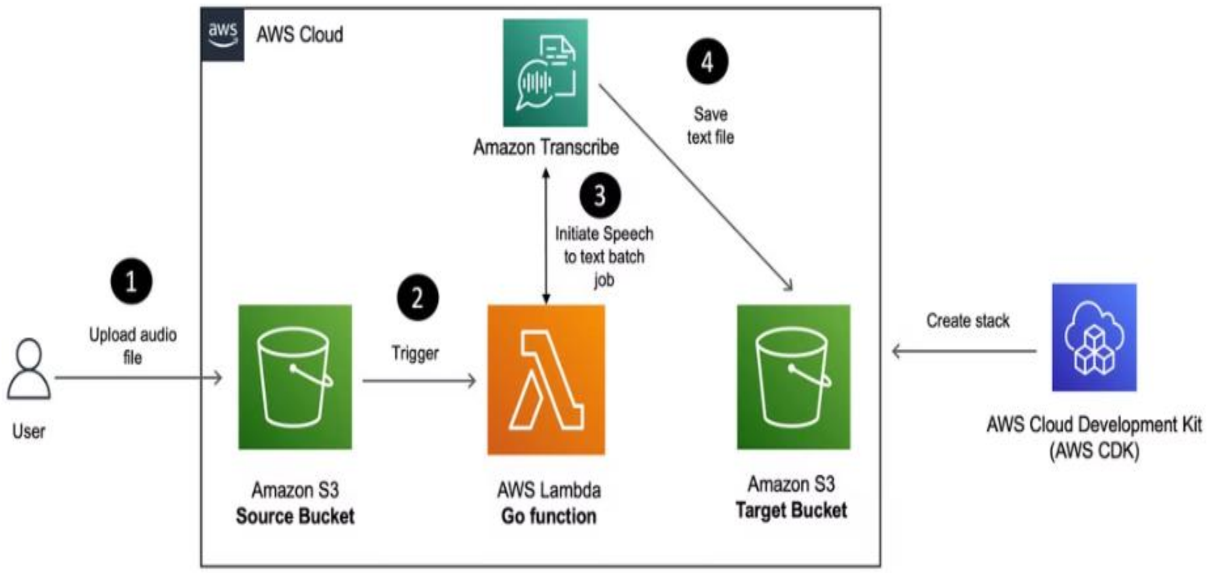
IAM roles and policies are configured to ensure secure access and interactions between AWS services. For example:

- Lambda functions are granted permission to access S3 and Transcribe.
- S3 buckets are secured with fine-grained access controls to prevent unauthorized access.

5. Optional: Amazon CloudWatch (Monitoring & Logging)

CloudWatch is used for logging Lambda executions and monitoring the health and performance of the system. This ensures better debugging and operational transparency.

ARCHITECTURE DIAGRAM



DATA GATHERING

The effectiveness of a voice-to-text converter largely depends on the quality and diversity of the audio data used for testing and validation. In this project, data gathering focused on collecting various types of audio files to evaluate the system's performance under different conditions.

1. Sources of Audio Data

To simulate real-world use cases, audio samples were gathered from the following sources:

- Publicly available datasets such as:
  - LibriSpeech ASR Corpus – a large corpus of read English speech.
  - Mozilla Common Voice – a multilingual voice dataset for speech recognition.
- User-generated audio files, recorded via smartphones or microphones, including:
  - Conversational speech (interviews, discussions)
  - Formal speech (presentations, news)
  - Noisy environments (background music, traffic)
- Synthetic audio, generated using text-to-speech (TTS) engines to test controlled variables like pace and clarity.

2. Audio File Formats

The collected audio data included multiple formats compatible with Amazon Transcribe, such as:

- .mp3
- .mp4
- .wav
- .flac

Sampling rates ranged from 16 kHz to 44.1 kHz to test Transcribe's ability to handle various audio qualities.

3. Data Annotation (for Accuracy Testing)

For evaluation purposes, a subset of the audio files was paired with manually verified transcripts. This allowed for comparison between the expected and generated outputs to measure:

- Word Error Rate (WER)
- Speaker identification accuracy
- Punctuation and formatting correctness

4. Preprocessing

Before uploading to the system, the audio files underwent basic preprocessing to:

- Normalize volume
- Trim silence
- Ensure compatibility with AWS Transcribe's input requirements

This helped reduce noise and improve transcription accuracy.

CREATIVITY AND ORIGINALITY

The project demonstrates creativity and originality by combining cloud-native technologies into a fully serverless, event-driven architecture for voice-to-text conversion — a solution that is efficient, scalable, and practical for real-world deployment.

1. Innovative Use of Serverless Architecture

Unlike conventional approaches that rely on dedicated servers or managed services with ongoing maintenance, this project leverages AWS Lambda to completely eliminate the need for server management. This architectural choice showcases originality in designing a hands-off infrastructure that is automatically triggered and scales with demand.

2. Automated Workflow Integration

The system integrates Amazon S3, Lambda, and Amazon Transcribe in an automated pipeline that requires no human intervention after the initial audio upload. This smooth orchestration of services reflects a creative solution that minimizes latency, human effort, and cost.

3. Support for Real-World Use Cases

By testing the system with a wide range of audio formats, accents, and environments, the project shows creative foresight in designing a product that could be adapted for:

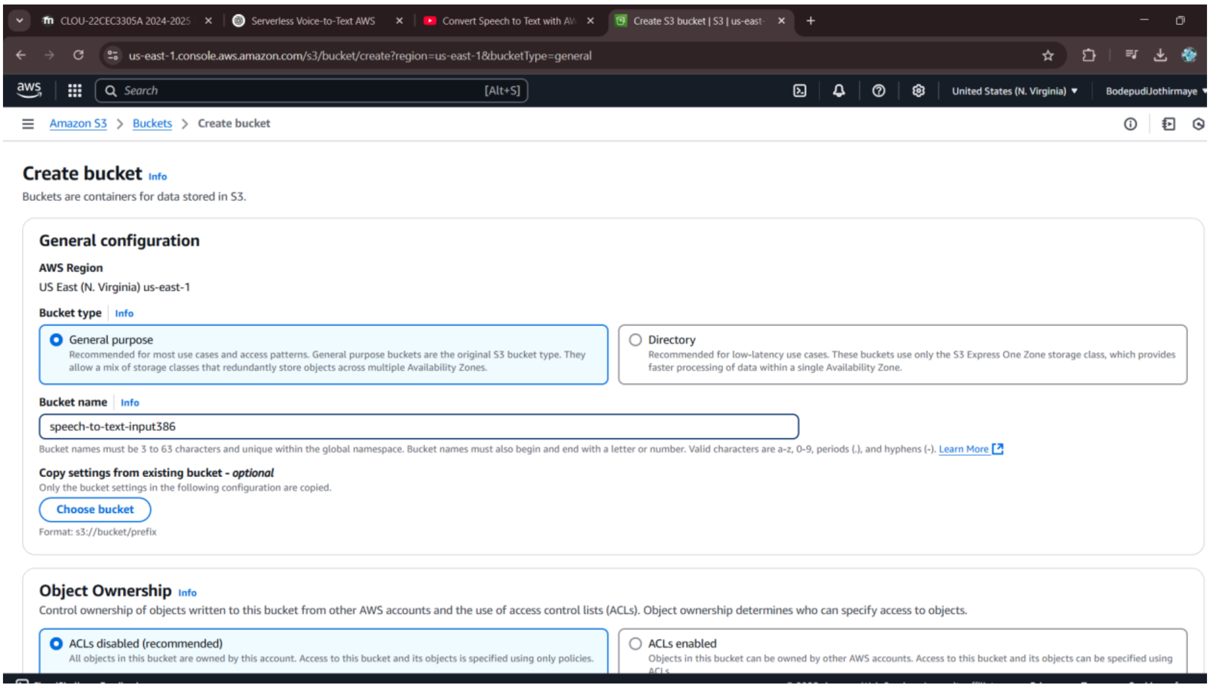
- Lecture transcription
- Voice-based note-taking apps
- Customer service voice logs
- Accessibility tools for the hearing impaired

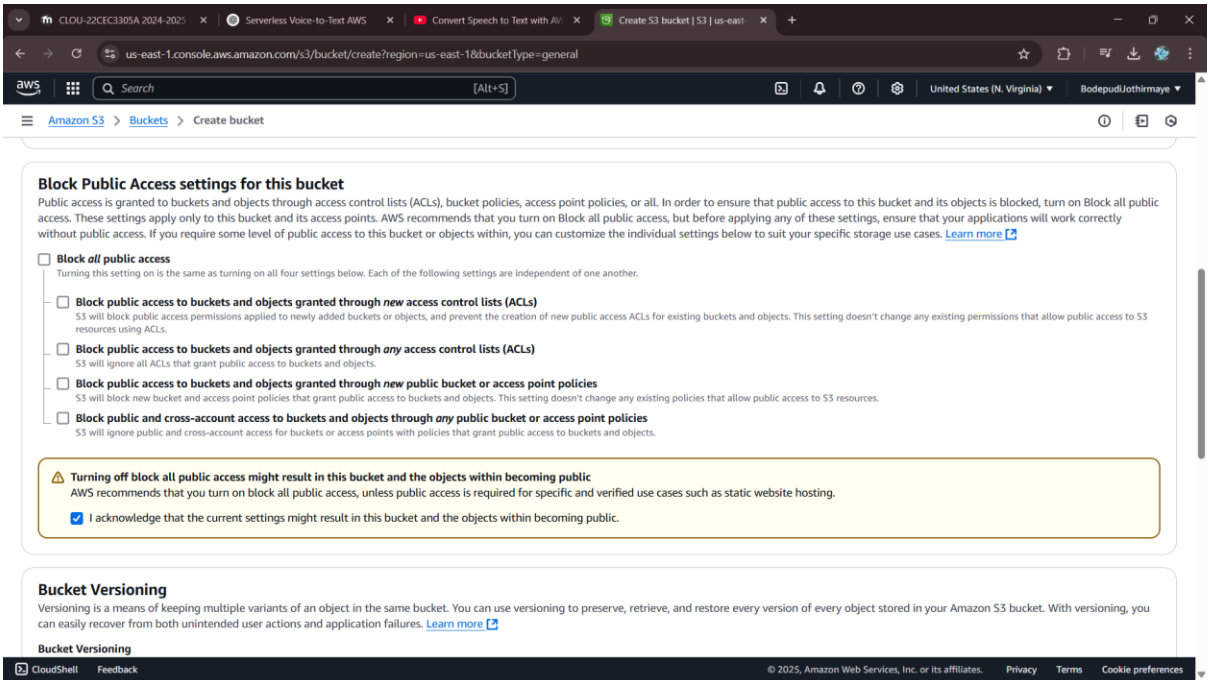
4. Modular & Extensible Design

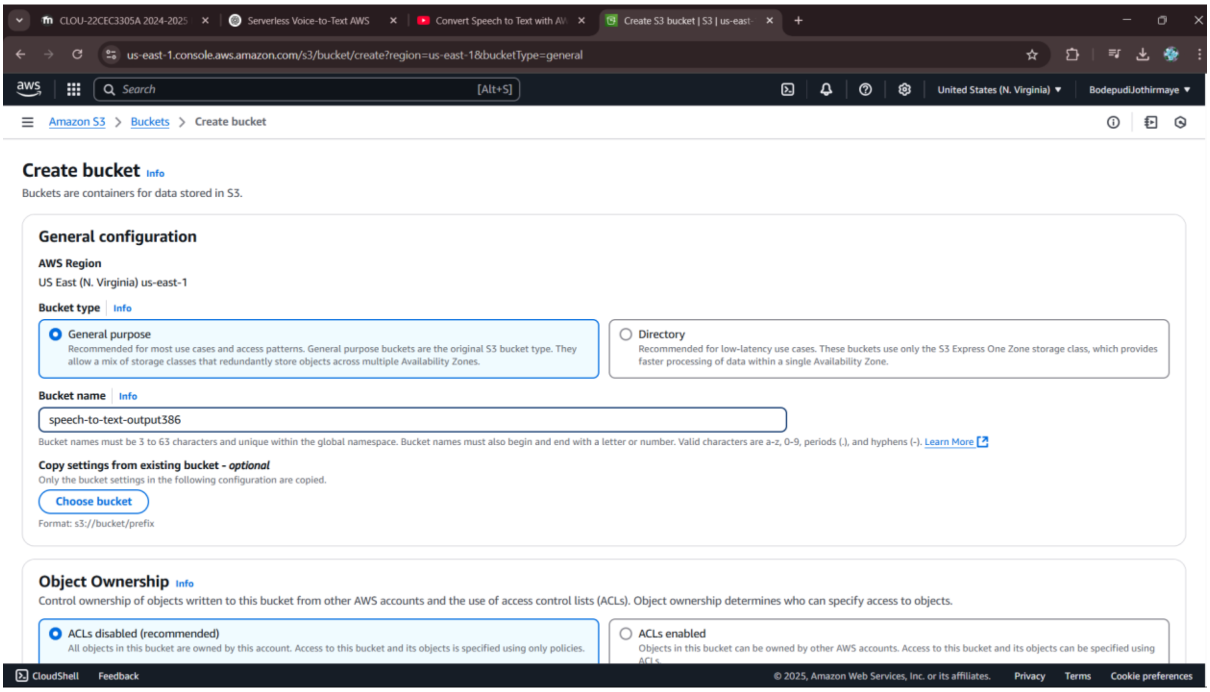
The architecture is designed to be easily extendable — new features like real-time streaming transcription, sentiment analysis, or language translation can be plugged in with minimal changes. This shows original thinking in planning for future scalability and diverse applications.

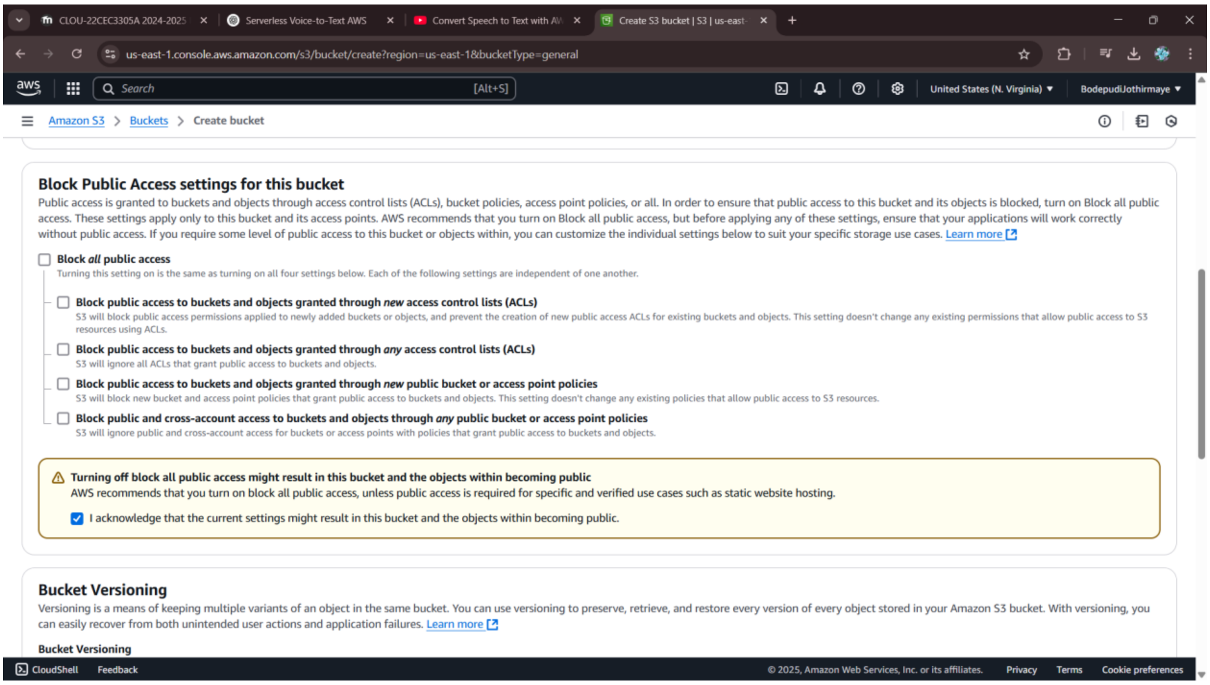
5. Cost-Conscious Innovation

The pay-as-you-go model offered by AWS serverless services is creatively exploited to keep operational costs extremely low. The system runs only when needed, which is ideal for startups, students, or small businesses with limited budgets.

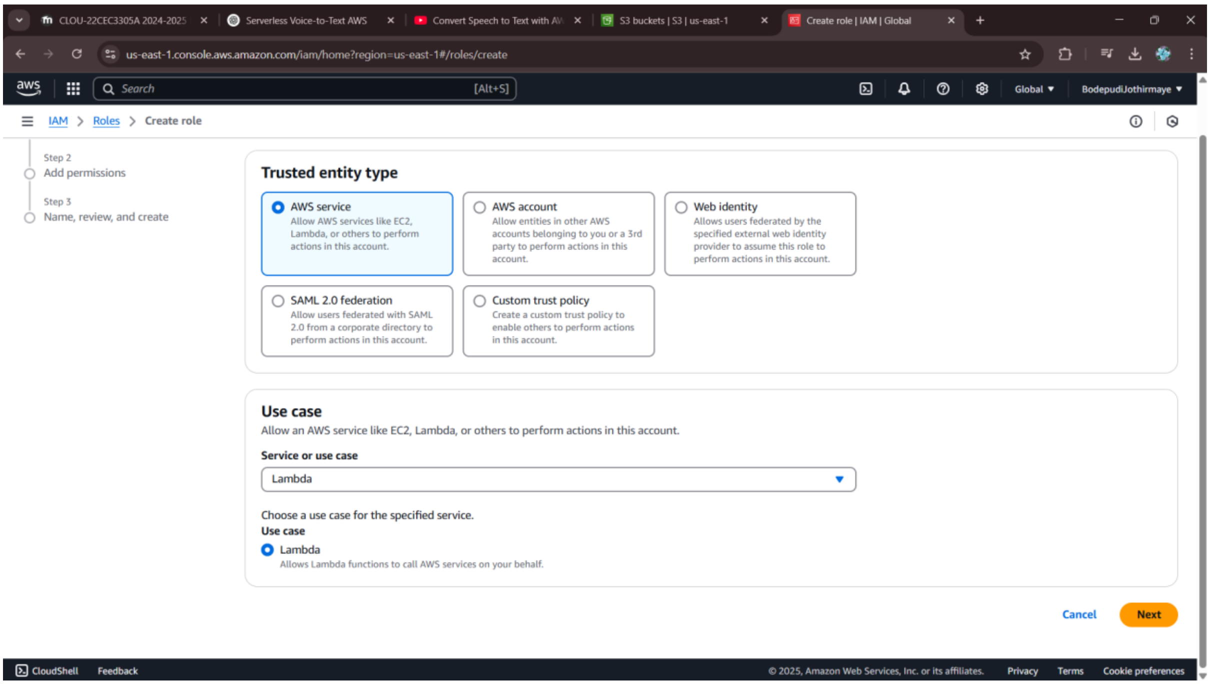



CLOU-22CEC3305A 2024-2025
Serverless Voice-to-Text AWS
Convert Speech to Text with AW
Create S3 bucket | S3 | us-east-
us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general
Search
[Alt+S]
United States (N. Virginia)
BodepudiJothirmaye
Amazon S3 > Buckets > Create bucket
Block Public Access settings for this bucket
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.
Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
Block public and cross-account access to buckets and objects through any public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.
Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.
I acknowledge that the current settings might result in this bucket and the objects within becoming public.
Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more
Bucket Versioning
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

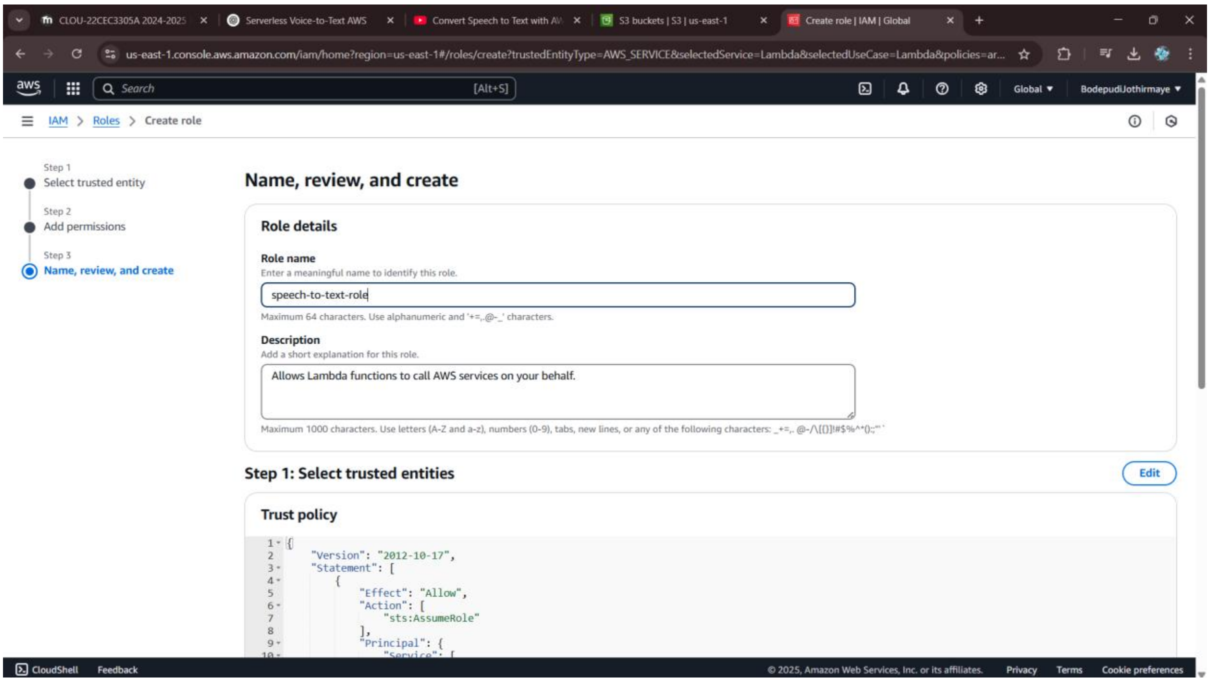

CLOU-22CEC3305A 2024-2025
Serverless Voice-to-Text AWS
Convert Speech to Text with AW
Create S3 bucket | S3 | us-east-
us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general
Search
[Alt+S]
United States (N. Virginia)
BodepudiJothirmaye
Amazon S3 > Buckets > Create bucket
Create bucket Info
Buckets are containers for data stored in S3.
General configuration
AWS Region
US East (N. Virginia) us-east-1
Bucket type Info
General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.
Directory
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.
Bucket name Info
speech-to-text-output386
Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). Learn More
Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.
Choose bucket
Format: s3://bucket/prefix
Object Ownership Info
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.
ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

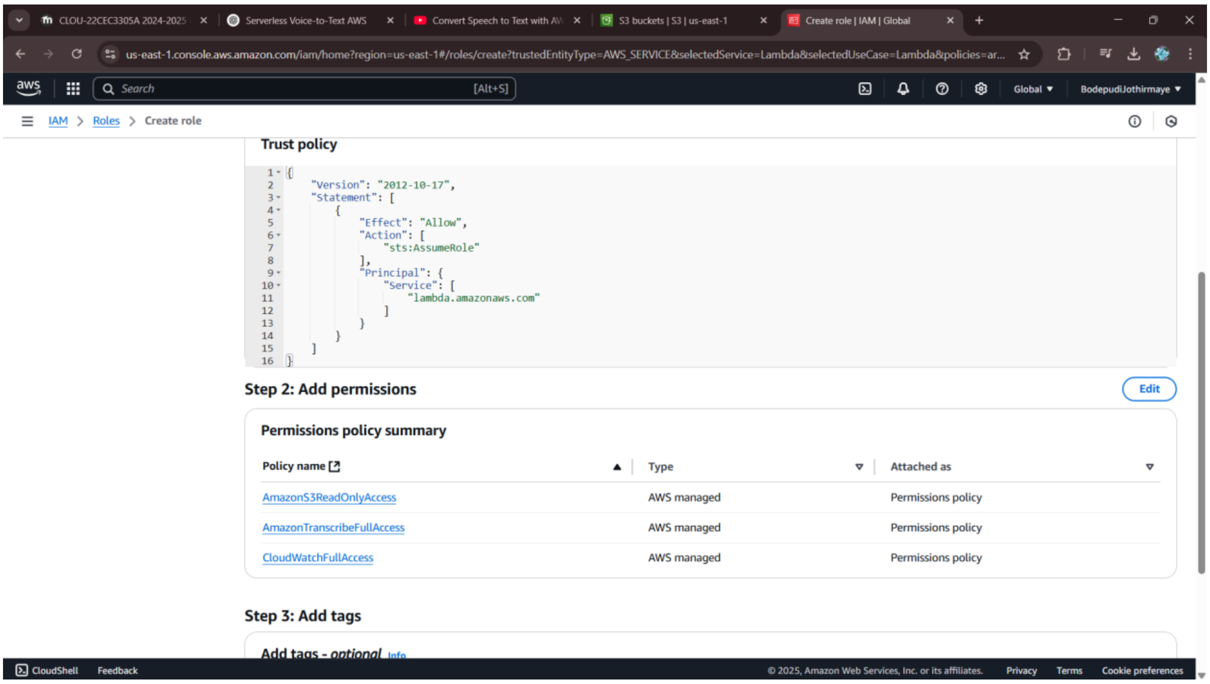


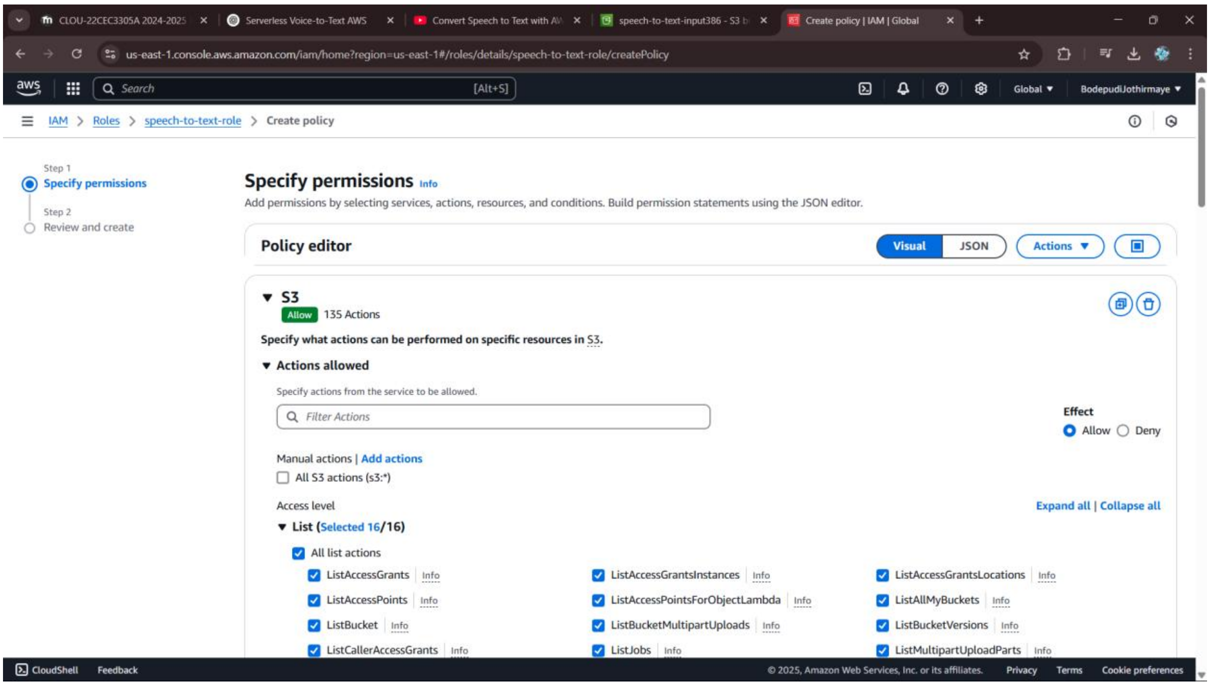
Create S3 bucket | S3 | us-east-1
us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general
Amazon S3 > Buckets > Create bucket
Block Public Access settings for this bucket
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.
Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
Block public and cross-account access to buckets and objects through any public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.
Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.
I acknowledge that the current settings might result in this bucket and the objects within becoming public.
Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more
Bucket Versioning
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

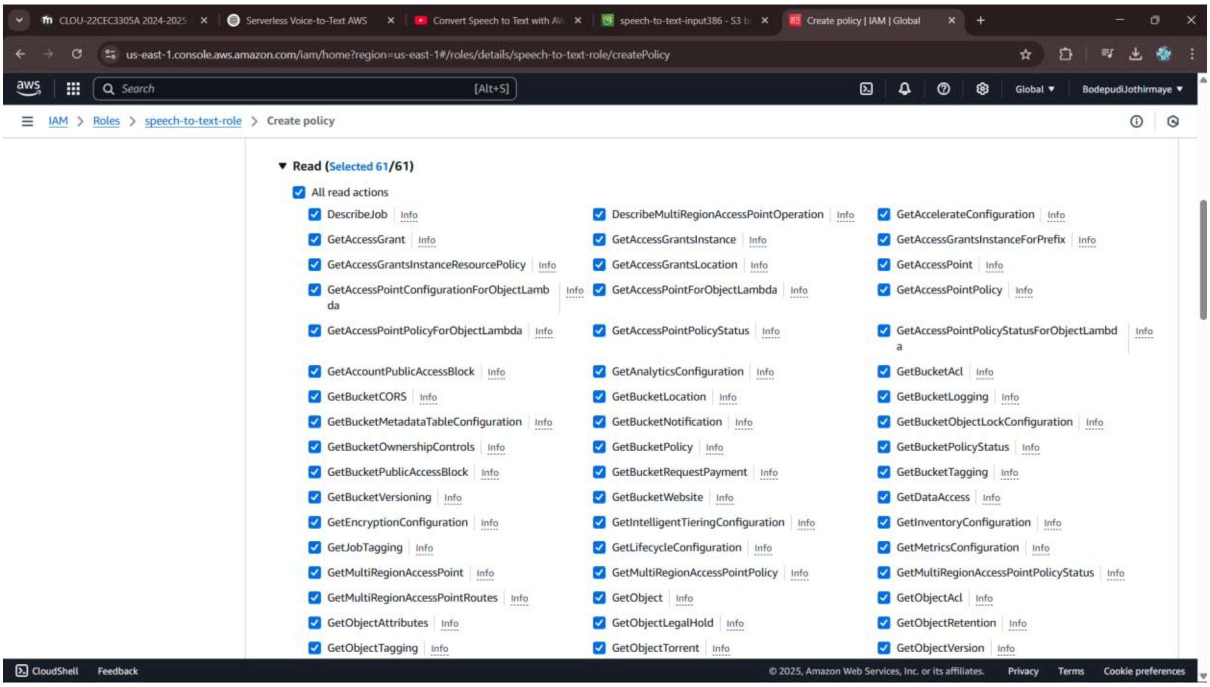


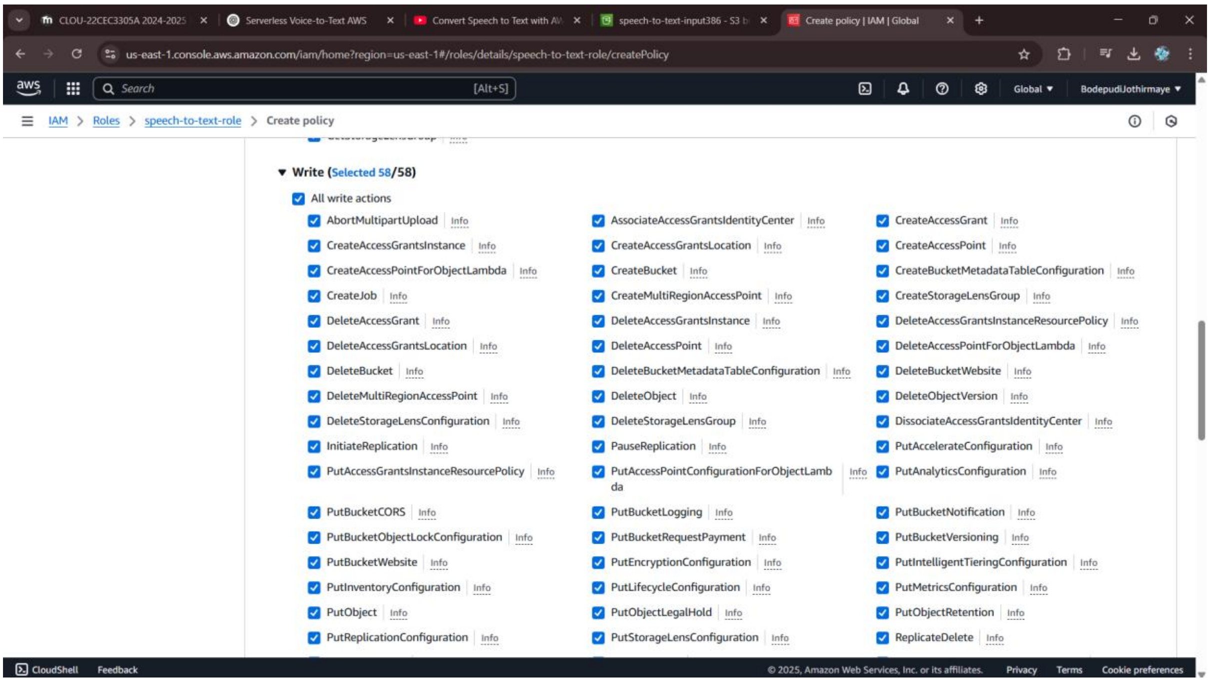
Create role | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/create
IAM > Roles > Create role
Step 2
Add permissions
Step 3
Name, review, and create
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Service or use case
Lambda
Choose a use case for the specified service.
Use case
Lambda
Allows Lambda functions to call AWS services on your behalf.
Cancel
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

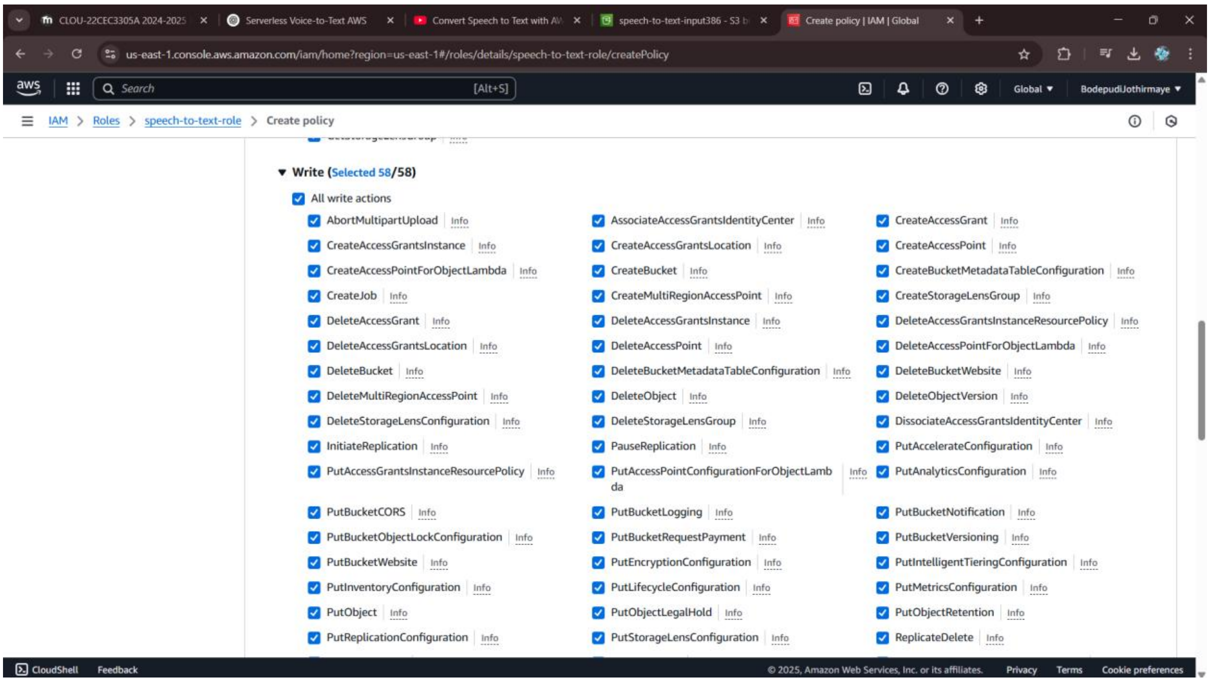


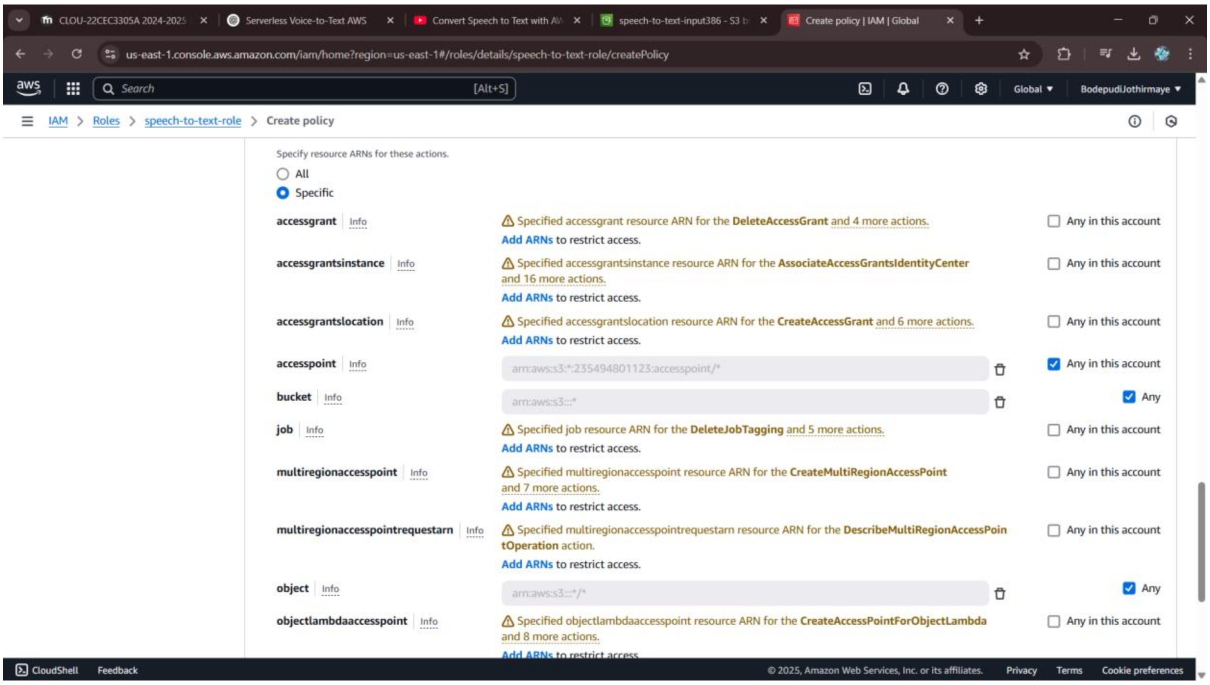




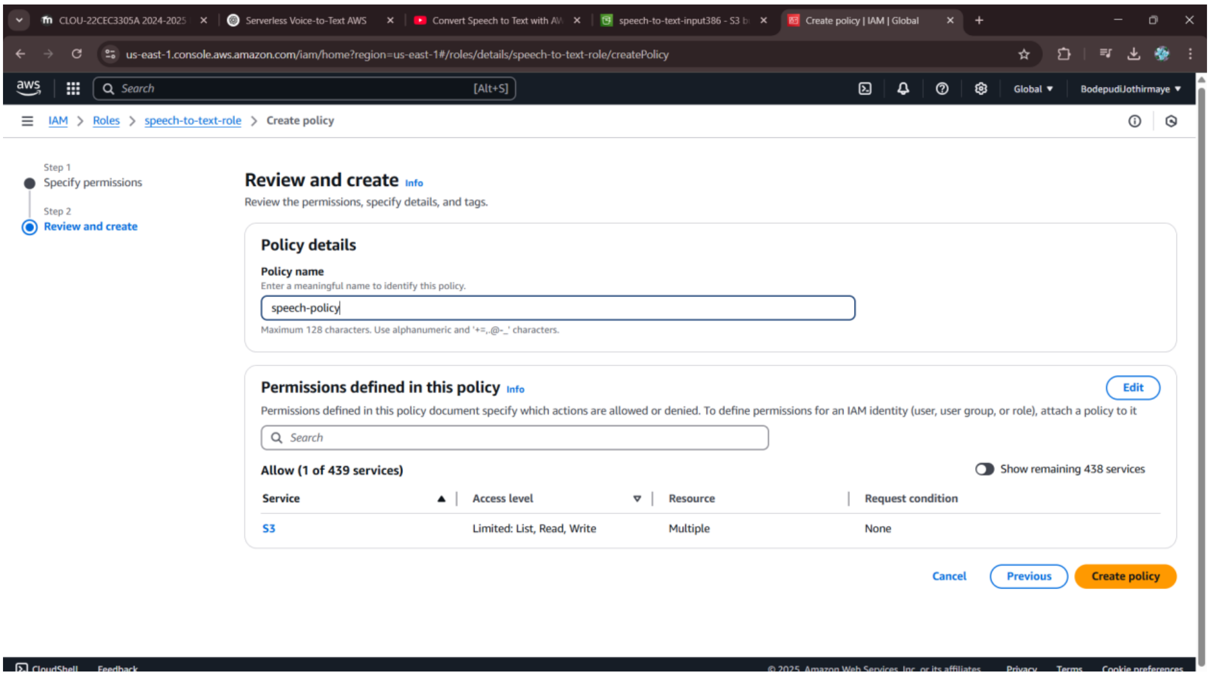



CLOU-22CEC3305A 2024-2025
Serverless Voice-to-Text AWS
Convert Speech to Text with AW
speech-to-text-input386 - S3 b
Create policy | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/details/speech-to-text-role/createPolicy
Search
[Alt+S]
Global
BodepudiJothirmaye
IAM > Roles > speech-to-text-role > Create policy
Write (Selected 58/58)
All write actions
AbortMultipartUpload Info
AssociateAccessGrantsIdentityCenter Info
CreateAccessGrant Info
CreateAccessGrantsInstance Info
CreateAccessGrantsLocation Info
CreateAccessPoint Info
CreateAccessPointForObjectLambda Info
CreateBucket Info
CreateBucketMetadataTableConfiguration Info
CreateJob Info
CreateMultiRegionAccessPoint Info
CreateStorageLensGroup Info
DeleteAccessGrant Info
DeleteAccessGrantsInstance Info
DeleteAccessGrantsInstanceResourcePolicy Info
DeleteAccessGrantsLocation Info
DeleteAccessPoint Info
DeleteAccessPointForObjectLambda Info
DeleteBucket Info
DeleteBucketMetadataTableConfiguration Info
DeleteBucketWebsite Info
DeleteMultiRegionAccessPoint Info
DeleteObject Info
DeleteObjectVersion Info
DeleteStorageLensConfiguration Info
DeleteStorageLensGroup Info
DissociateAccessGrantsIdentityCenter Info
InitiateReplication Info
PauseReplication Info
PutAccelerateConfiguration Info
PutAccessGrantsInstanceResourcePolicy Info
PutAccessPointConfigurationForObjectLambda Info
PutAnalyticsConfiguration Info
PutBucketCORS Info
PutBucketLogging Info
PutBucketNotification Info
PutBucketObjectLockConfiguration Info
PutBucketRequestPayment Info
PutBucketVersioning Info
PutBucketWebsite Info
PutEncryptionConfiguration Info
PutIntelligentTieringConfiguration Info
PutInventoryConfiguration Info
PutLifecycleConfiguration Info
PutMetricsConfiguration Info
PutObject Info
PutObjectLegalHold Info
PutObjectRetention Info
PutReplicationConfiguration Info
PutStorageLensConfiguration Info
ReplicateDelete Info
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

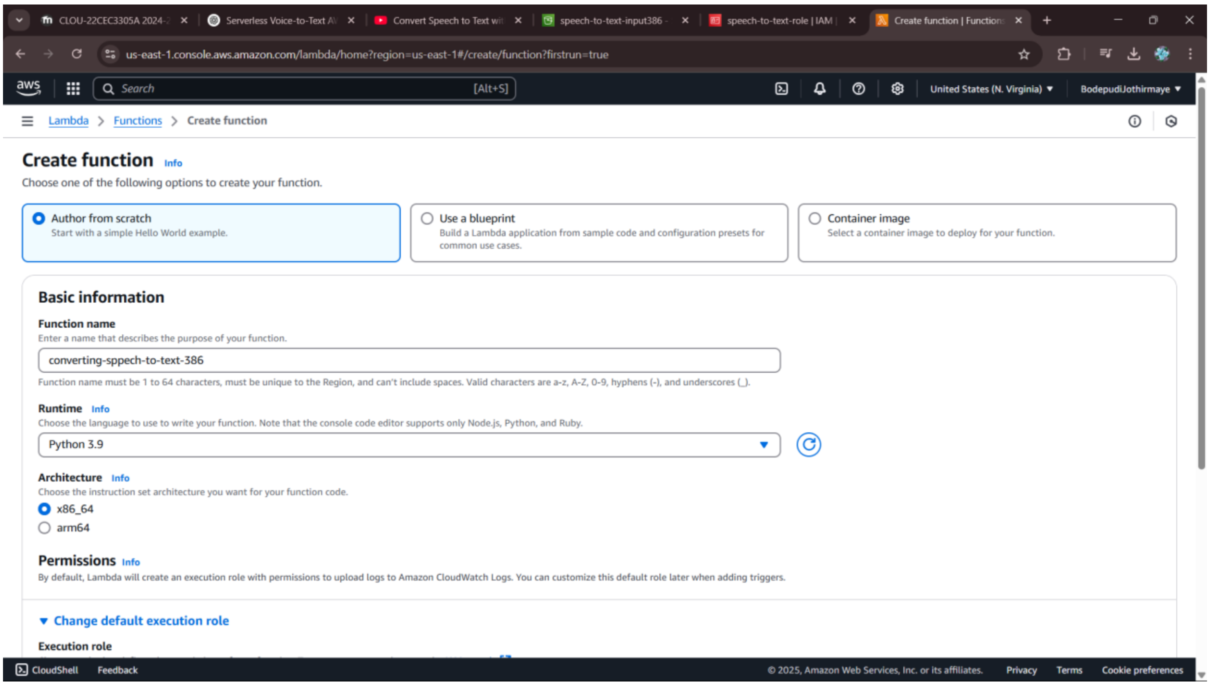

CLOU-22CEC3305A 2024-2025
Serverless Voice-to-Text AWS
Convert Speech to Text with AW
speech-to-text-input386 - S3 b
Create policy | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/details/speech-to-text-role/createPolicy
Search
[Alt+S]
Global
BodepudiJothirmaye
IAM > Roles > speech-to-text-role > Create policy
Write (Selected 58/58)
All write actions
AbortMultipartUpload Info
AssociateAccessGrantsIdentityCenter Info
CreateAccessGrant Info
CreateAccessGrantsInstance Info
CreateAccessGrantsLocation Info
CreateAccessPoint Info
CreateAccessPointForObjectLambda Info
CreateBucket Info
CreateBucketMetadataTableConfiguration Info
CreateJob Info
CreateMultiRegionAccessPoint Info
CreateStorageLensGroup Info
DeleteAccessGrant Info
DeleteAccessGrantsInstance Info
DeleteAccessGrantsInstanceResourcePolicy Info
DeleteAccessGrantsLocation Info
DeleteAccessPoint Info
DeleteAccessPointForObjectLambda Info
DeleteBucket Info
DeleteBucketMetadataTableConfiguration Info
DeleteBucketWebsite Info
DeleteMultiRegionAccessPoint Info
DeleteObject Info
DeleteObjectVersion Info
DeleteStorageLensConfiguration Info
DeleteStorageLensGroup Info
DissociateAccessGrantsIdentityCenter Info
InitiateReplication Info
PauseReplication Info
PutAccelerateConfiguration Info
PutAccessGrantsInstanceResourcePolicy Info
PutAccessPointConfigurationForObjectLambda Info
PutAnalyticsConfiguration Info
PutBucketCORS Info
PutBucketLogging Info
PutBucketNotification Info
PutBucketObjectLockConfiguration Info
PutBucketRequestPayment Info
PutBucketVersioning Info
PutBucketWebsite Info
PutEncryptionConfiguration Info
PutIntelligentTieringConfiguration Info
PutInventoryConfiguration Info
PutLifecycleConfiguration Info
PutMetricsConfiguration Info
PutObject Info
PutObjectLegalHold Info
PutObjectRetention Info
PutReplicationConfiguration Info
PutStorageLensConfiguration Info
ReplicateDelete Info
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

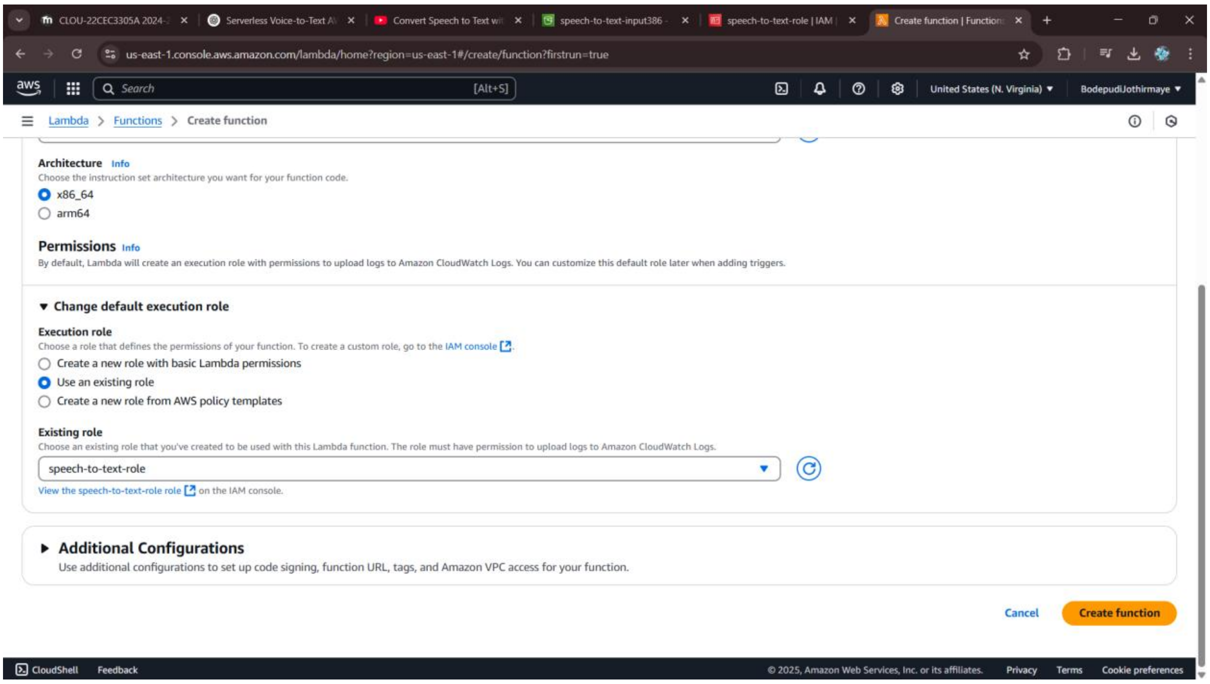


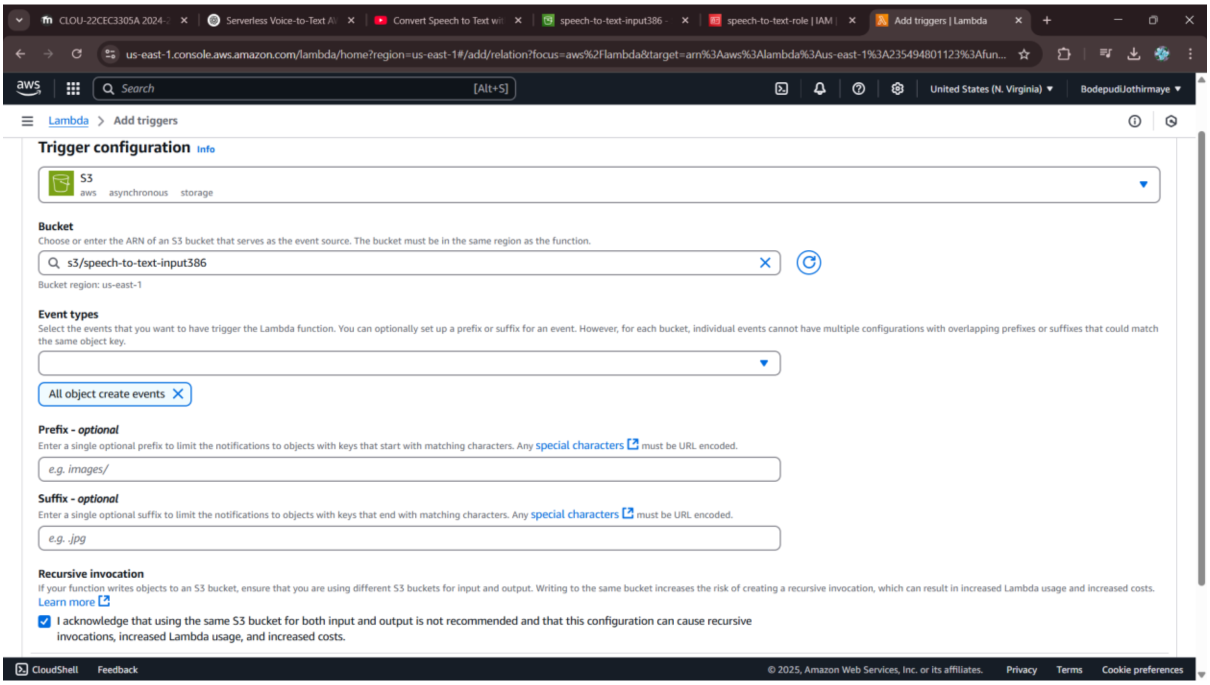
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/details/speech-to-text-role/createPolicy
IAM > Roles > speech-to-text-role > Create policy
Specify resource ARNs for these actions.
All
Specific
accessgrant
Specified accessgrant resource ARN for the DeleteAccessGrant and 4 more actions.
Add ARNs to restrict access.
Any in this account
accessgrantsinstance
Specified accessgrantsinstance resource ARN for the AssociateAccessGrantsIdentityCenter and 16 more actions.
accessgrantslocation
Specified accessgrantslocation resource ARN for the CreateAccessGrant and 6 more actions.
accesspoint
arn:aws:s3:*:235494801123:accesspoint/*
bucket
arn:aws:s3:::*
Any
job
Specified job resource ARN for the DeleteJobTagging and 5 more actions.
multiregionaccesspoint
Specified multiregionaccesspoint resource ARN for the CreateMultiRegionAccessPoint and 7 more actions.
multiregionaccesspointrequestarn
Specified multiregionaccesspointrequestarn resource ARN for the DescribeMultiRegionAccessPointOperation action.
object
arn:aws:s3:::*/*
objectlambdaaccesspoint
Specified objectlambdaaccesspoint resource ARN for the CreateAccessPointForObjectLambda and 8 more actions.

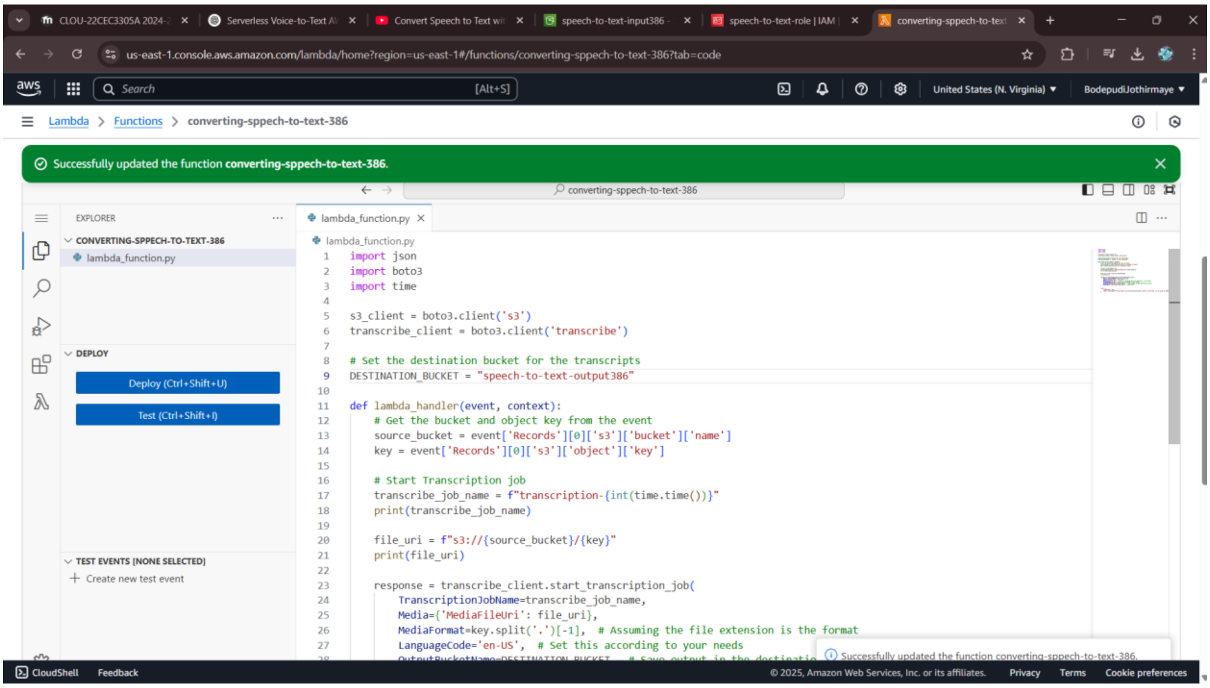


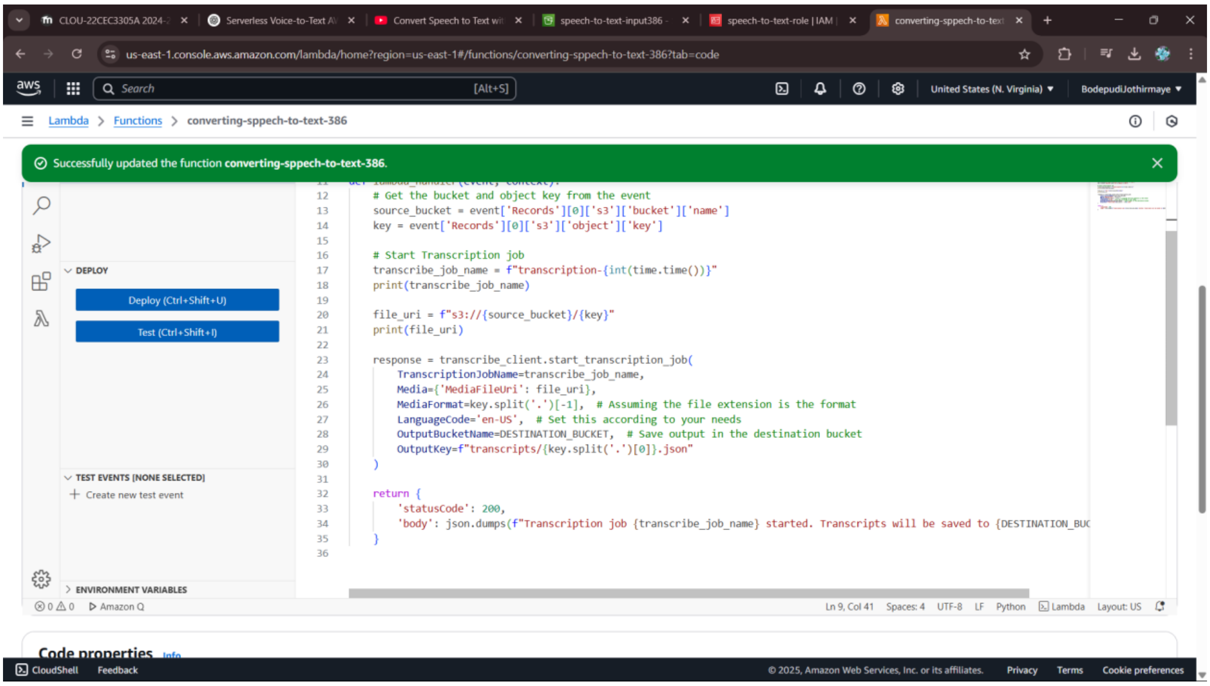
IAM > Roles > speech-to-text-role > Create policy
Step 1
Specify permissions
Step 2
Review and create
Review and create
Review the permissions, specify details, and tags.
Policy details
Policy name
Enter a meaningful name to identify this policy.
speech-policy
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.
Permissions defined in this policy
Edit
Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it
Allow (1 of 439 services)
Show remaining 438 services
Service
Access level
Resource
Request condition
S3
Limited: List, Read, Write
Multiple
None
Cancel
Previous
Create policy

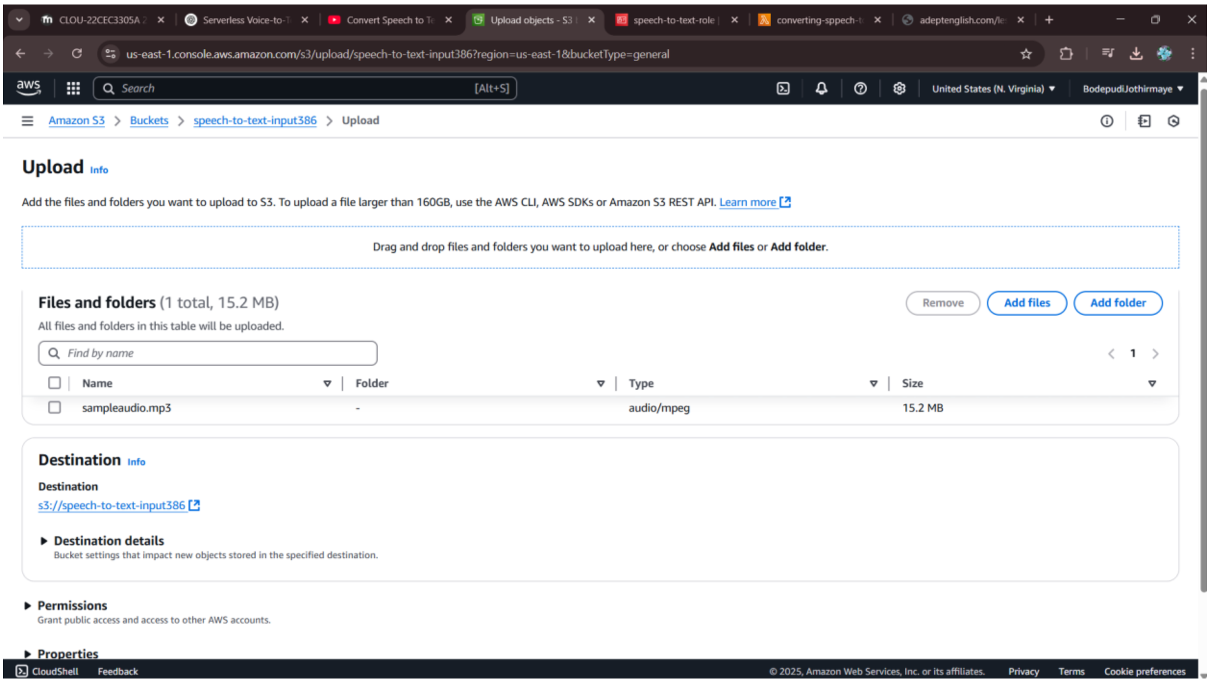

us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/create/function?firstrun=true
Lambda > Functions > Create function
Create function Info
Choose one of the following options to create your function.
Author from scratch
Start with a simple Hello World example.
Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.
Container image
Select a container image to deploy for your function.
Basic information
Function name
Enter a name that describes the purpose of your function.
converting-sppech-to-text-386
Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).
Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.9
Architecture Info
Choose the instruction set architecture you want for your function code.
x86_64
arm64
Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.
Change default execution role
Execution role
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/create/function?firstrun=true
Lambda > Functions > Create function
Architecture Info
Choose the instruction set architecture you want for your function code.
x86_64
arm64
Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.
Change default execution role
Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
Create a new role with basic Lambda permissions
Use an existing role
Create a new role from AWS policy templates
Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
speech-to-text-role
View the speech-to-text-role role on the IAM console.
Additional Configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.
Cancel
Create function
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


Lambda > Add triggers
Trigger configuration Info
S3
aws asynchronous storage
Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.
s3/speech-to-text-input386
Bucket region: us-east-1
Event types
Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.
All object create events
Prefix - optional
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any special characters must be URL encoded.
e.g. images/
Suffix - optional
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters. Any special characters must be URL encoded.
e.g. .jpg
Recursive invocation
If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs.
Learn more
I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


Lambda > Functions > converting-sppech-to-text-386
Successfully updated the function converting-sppech-to-text-386.
EXPLORER
CONVERTING-SPPECH-TO-TEXT-386
lambda_function.py
DEPLOY
Deploy (Ctrl+Shift+U)
Test (Ctrl+Shift+I)
TEST EVENTS [NONE SELECTED]
Create new test event
import json
import boto3
import time
s3_client = boto3.client('s3')
transcribe_client = boto3.client('transcribe')
# Set the destination bucket for the transcripts
DESTINATION_BUCKET = "speech-to-text-output386"
def lambda_handler(event, context):
# Get the bucket and object key from the event
source_bucket = event['Records'][0]['s3']['bucket']['name']
key = event['Records'][0]['s3']['object']['key']
# Start Transcription job
transcribe_job_name = f"transcription-{int(time.time())}"
print(transcribe_job_name)
file_uri = f"s3://{source_bucket}/{key}"
print(file_uri)
response = transcribe_client.start_transcription_job(
TranscriptionJobName=transcribe_job_name,
Media={'MediaFileUri': file_uri},
MediaFormat=key.split('.')[-1], # Assuming the file extension is the format
LanguageCode='en-US', # Set this according to your needs
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


Lambda > Functions > converting-sppech-to-text-386
Successfully updated the function converting-sppech-to-text-386.
DEPLOY
Deploy (Ctrl+Shift+U)
Test (Ctrl+Shift+I)
TEST EVENTS [NONE SELECTED]
Create new test event
ENVIRONMENT VARIABLES
# Get the bucket and object key from the event
source_bucket = event['Records'][0]['s3']['bucket']['name']
key = event['Records'][0]['s3']['object']['key']
# Start Transcription job
transcribe_job_name = f"transcription-{int(time.time())}"
print(transcribe_job_name)
file_uri = f"s3://{source_bucket}/{key}"
print(file_uri)
response = transcribe_client.start_transcription_job(
TranscriptionJobName=transcribe_job_name,
Media={'MediaFileUri': file_uri},
MediaFormat=key.split('.')[-1], # Assuming the file extension is the format
LanguageCode='en-US', # Set this according to your needs
OutputBucketName=DESTINATION_BUCKET, # Save output in the destination bucket
OutputKey=f"transcripts/{key.split('.')[0]}.json"
)
return {
'statusCode': 200,
'body': json.dumps(f"Transcription job {transcribe_job_name} started. Transcripts will be saved to {DESTINATION_BUC
}
Code properties
© 2025, Amazon Web Services, Inc. or its affiliates.


Amazon S3 > Buckets > speech-to-text-input386 > Upload
Upload
Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. Learn more
Drag and drop files and folders you want to upload here, or choose Add files or Add folder.
Files and folders (1 total, 15.2 MB)
All files and folders in this table will be uploaded.
Remove
Add files
Add folder
Name
Folder
Type
Size
sampleaudio.mp3
audio/mpeg
15.2 MB
Destination
s3://speech-to-text-input386
Destination details
Bucket settings that impact new objects stored in the specified destination.
Permissions
Grant public access and access to other AWS accounts.
Properties

Analysis & Problem-Solving

Throughout the development of the Serverless Voice-to-Text Converter, several technical and architectural challenges were identified and addressed through detailed analysis and problem-solving strategies. These efforts ensured the system remained reliable, scalable, and user-friendly.

1. Challenge: Triggering Transcription Automatically

Problem: Ensuring that transcription begins immediately after an audio file is uploaded. Solution: Implemented S3 event notifications to trigger an AWS Lambda function automatically upon new file uploads. This event-driven model eliminated delays and manual intervention.

2. Challenge: Handling Various Audio Formats and Quality

Problem: Inconsistent audio file quality and unsupported formats could cause transcription failures.

Solution: Enforced pre-upload checks and preprocessing (format validation, noise reduction). Additionally, the system logs errors in CloudWatch to help identify problematic files.

3. Challenge: Monitoring Transcription Status

Problem: Amazon Transcribe runs jobs asynchronously, so tracking job completion was complex.

Solution: Lambda functions were configured to poll the transcription job status using the Transcribe API, or optionally leverage SNS notifications for more efficient handling of job completions.

4. Challenge: Securing Service Access

Problem: Misconfigured permissions could either block functionality or expose data. Solution: Defined precise IAM roles and policies to restrict access based on the principle of least privilege — allowing Lambda to access only the required S3 buckets and Transcribe services.

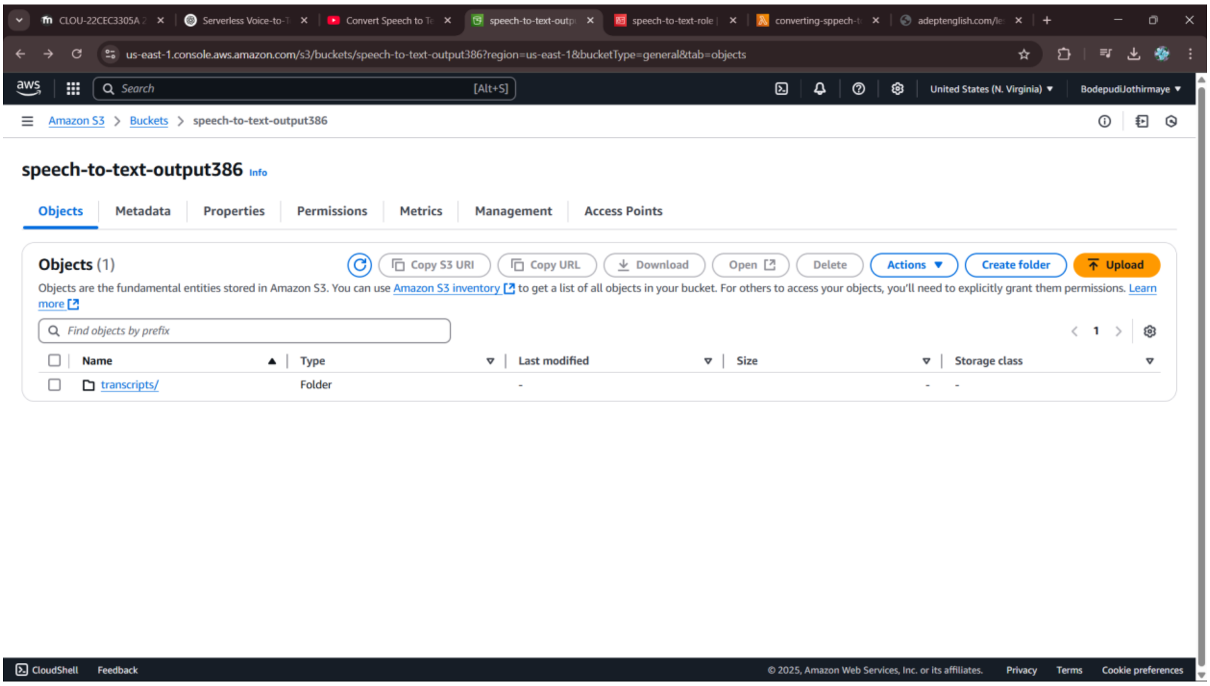
5. Challenge: Managing Output Data

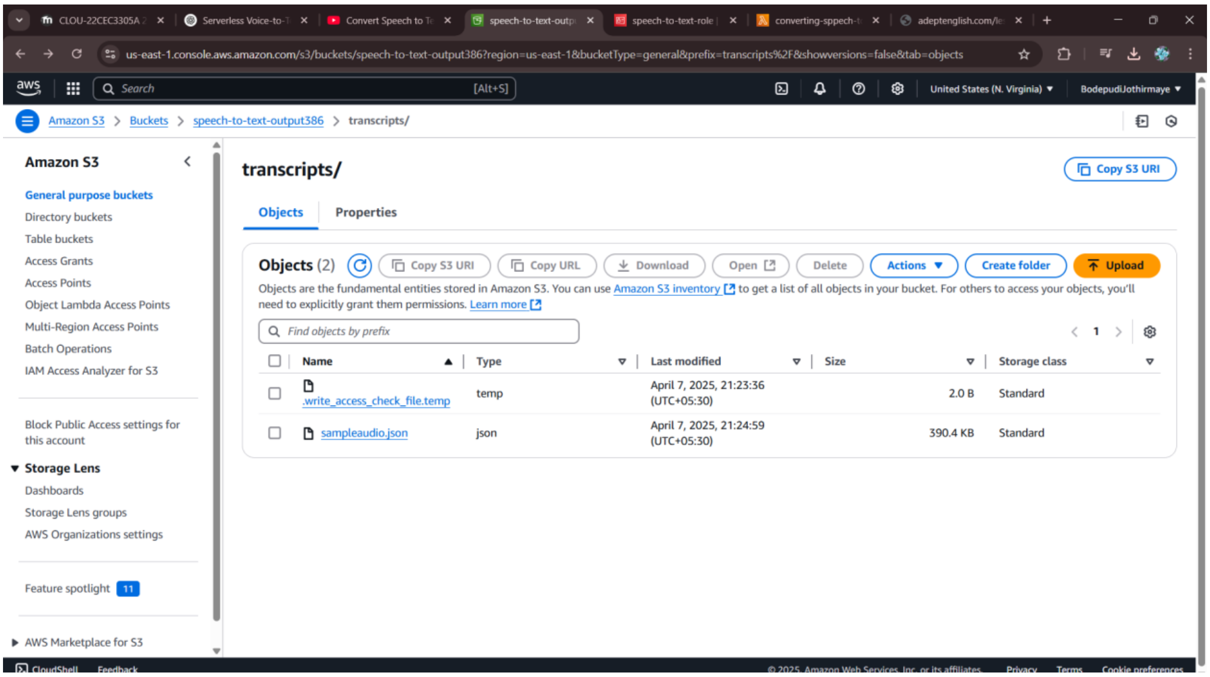
Problem: Users needed access to the final transcription in a readable format. Solution: Transcribed JSON output from Amazon Transcribe was parsed and stored as clean, readable .txt files in a separate S3 bucket. Optionally, the system can email results or push them to databases or front-end apps.

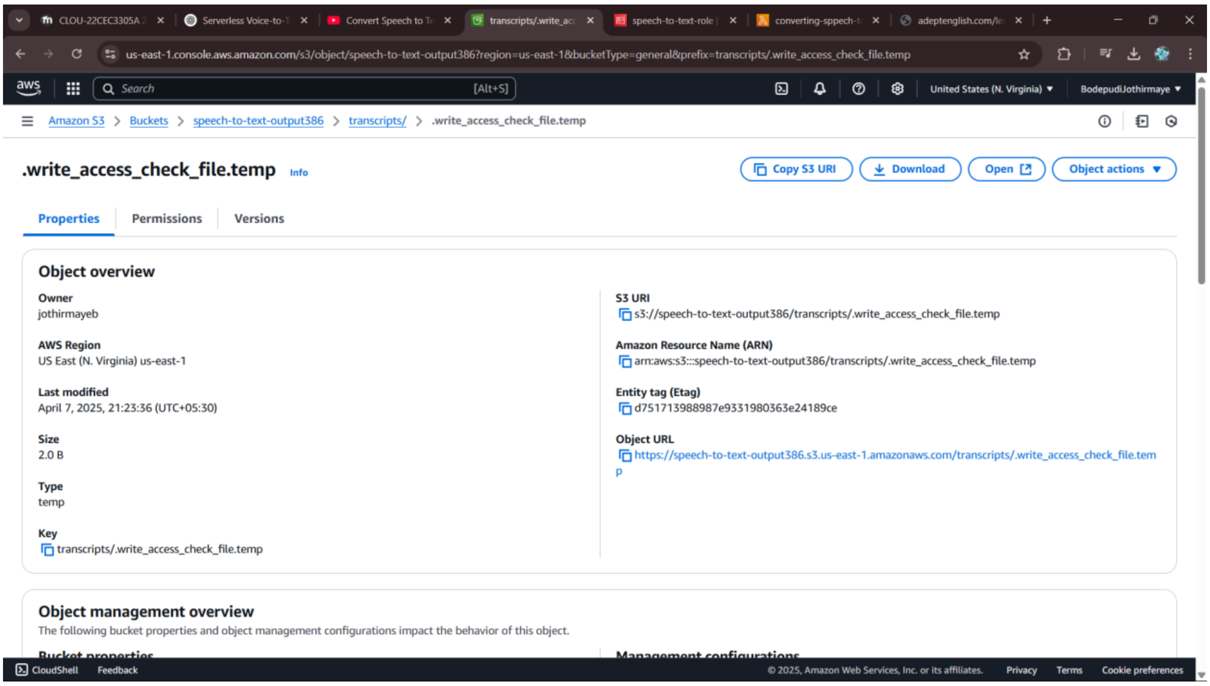
6. Challenge: Cost Optimization

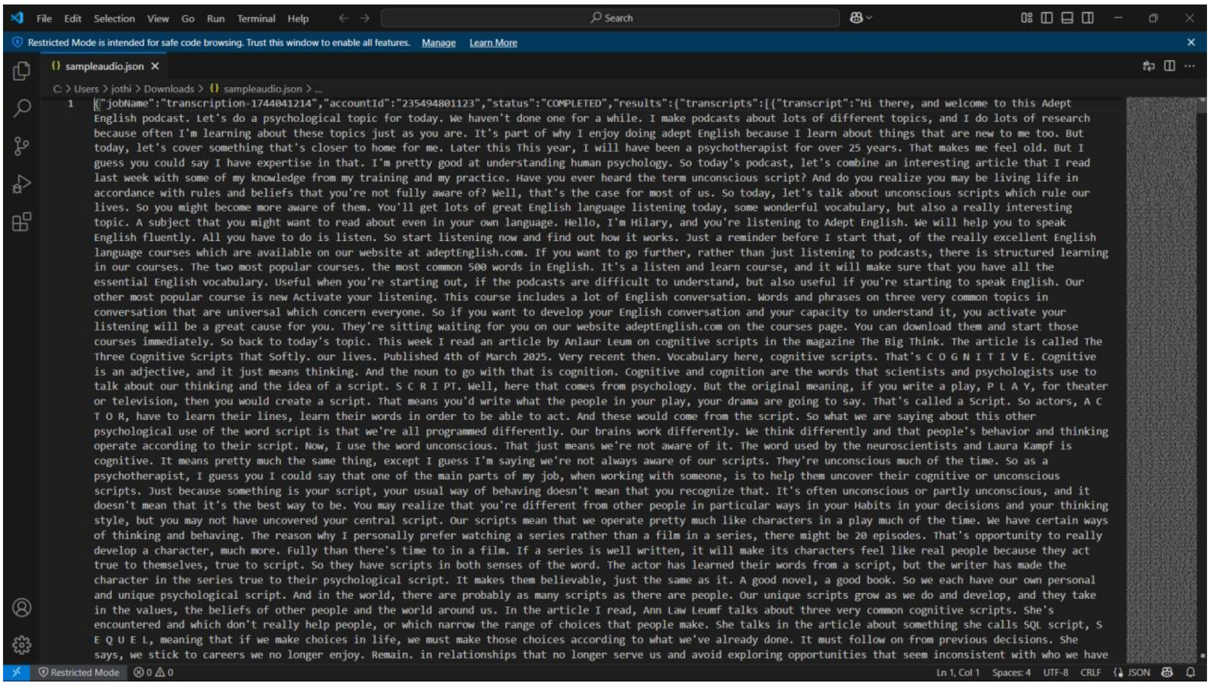
Problem: Even minimal compute time across many users could accumulate costs. Solution: Chose serverless resources specifically to minimize idle cost. Resources are only active when needed, and AWS's free tier benefits were maximized during testing.

Discussions & Results




us-east-1.console.amazon.aws.com/s3/buckets/speech-to-text-output386?region=us-east-1&bucketType=general&prefix=transcripts%2F&showversions=false&tab=objects
Amazon S3 > Buckets > speech-to-text-output386 > transcripts/
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Feature spotlight 11
AWS Marketplace for S3
transcripts/
Copy S3 URI
Objects
Properties
Objects (2)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more
Find objects by prefix
Name
Type
Last modified
Size
Storage class
.write_access_check_file.temp
temp
April 7, 2025, 21:23:36 (UTC+05:30)
2.0 B
Standard
sampleaudio.json
json
April 7, 2025, 21:24:59 (UTC+05:30)
390.4 KB
Standard


us-east-1.console.aws.amazon.com/s3/object/speech-to-text-output386?region=us-east-1&bucketType=general&prefix=transcripts/.write_access_check_file.temp
Amazon S3 > Buckets > speech-to-text-output386 > transcripts/ > .write_access_check_file.temp
.write_access_check_file.temp
Info
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
jothirmayeb
AWS Region
US East (N. Virginia) us-east-1
Last modified
April 7, 2025, 21:23:36 (UTC+05:30)
Size
2.0 B
Type
temp
Key
transcripts/.write_access_check_file.temp
S3 URI
s3://speech-to-text-output386/transcripts/.write_access_check_file.temp
Amazon Resource Name (ARN)
arn:aws:s3:::speech-to-text-output386/transcripts/.write_access_check_file.temp
Entity tag (Etag)
d751713988987e9331980363e24189ce
Object URL
https://speech-to-text-output386.s3.us-east-1.amazonaws.com/transcripts/.write_access_check_file.temp
Object management overview
The following bucket properties and object management configurations impact the behavior of this object.


File Edit Selection View Go Run Terminal Help
Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
sampleaudio.json
C: > Users > jothi > Downloads > sampleaudio.json > ...
{"jobName":"transcription-1744041214","accountId":"235494801123","status":"COMPLETED","results":{"transcripts":[{"transcript":"Hi there, and welcome to this Adept
English podcast. Let's do a psychological topic for today. We haven't done one for a while. I make podcasts about lots of different topics, and I do lots of research
because often I'm learning about these topics just as you are. It's part of why I enjoy doing adept English because I learn about things that are new to me too. But
today, let's cover something that's closer to home for me. Later this This year, I will have been a psychotherapist for over 25 years. That makes me feel old. But I
guess you could say I have expertise in that. I'm pretty good at understanding human psychology. So today's podcast, let's combine an interesting article that I read
last week with some of my knowledge from my training and my practice. Have you ever heard the term unconscious script? And do you realize you may be living life in
accordance with rules and beliefs that you're not fully aware of? Well, that's the case for most of us. So today, let's talk about unconscious scripts which rule our
lives. So you might become more aware of them. You'll get lots of great English language listening today, some wonderful vocabulary, but also a really interesting
topic. A subject that you might want to read about even in your own language. Hello, I'm Hilary, and you're listening to Adept English. We will help you to speak
English fluently. All you have to do is listen. So start listening now and find out how it works. Just a reminder before I start that, of the really excellent English
language courses which are available on our website at adeptEnglish.com. If you want to go further, rather than just listening to podcasts, there is structured learning
in our courses. The two most popular courses. the most common 500 words in English. It's a listen and learn course, and it will make sure that you have all the
essential English vocabulary. Useful when you're starting out, if the podcasts are difficult to understand, but also useful if you're starting to speak English. Our
other most popular course is new Activate your listening. This course includes a lot of English conversation. Words and phrases on three very common topics in
conversation that are universal which concern everyone. So if you want to develop your English conversation and your capacity to understand it, you activate your
listening will be a great cause for you. They're sitting waiting for you on our website adeptEnglish.com on the courses page. You can download them and start those
courses immediately. So back to today's topic. This week I read an article by Anlaur Leum on cognitive scripts in the magazine The Big Think. The article is called The
Three Cognitive Scripts That Softly. our lives. Published 4th of March 2025. Very recent then. Vocabulary here, cognitive scripts. That's C O G N I T I V E. Cognitive
is an adjective, and it just means thinking. And the noun to go with that is cognition. Cognitive and cognition are the words that scientists and psychologists use to
talk about our thinking and the idea of a script. S C R I PT. Well, here that comes from psychology. But the original meaning, if you write a play, P L A Y, for theater
or television, then you would create a script. That means you'd write what the people in your play, your drama are going to say. That's called a Script. So actors, A C
T O R, have to learn their lines, learn their words in order to be able to act. And these would come from the script. So what we are saying about this other
psychological use of the word script is that we're all programmed differently. Our brains work differently. We think differently and that people's behavior and thinking
operate according to their script. Now, I use the word unconscious. That just means we're not aware of it. The word used by the neuroscientists and Laura Kampf is
cognitive. It means pretty much the same thing, except I guess I'm saying we're not always aware of our scripts. They're unconscious much of the time. So as a
psychotherapist, I guess you I could say that one of the main parts of my job, when working with someone, is to help them uncover their cognitive or unconscious
scripts. Just because something is your script, your usual way of behaving doesn't mean that you recognize that. It's often unconscious or partly unconscious, and it
doesn't mean that it's the best way to be. You may realize that you're different from other people in particular ways in your Habits in your decisions and your thinking
style, but you may not have uncovered your central script. Our scripts mean that we operate pretty much like characters in a play much of the time. We have certain ways
of thinking and behaving. The reason why I personally prefer watching a series rather than a film in a series, there might be 20 episodes. That's opportunity to really
develop a character, much more. Fully than there's time to in a film. If a series is well written, it will make its characters feel like real people because they act
true to themselves, true to script. So they have scripts in both senses of the word. The actor has learned their words from a script, but the writer has made the
character in the series true to their psychological script. It makes them believable, just the same as it. A good novel, a good book. So we each have our own personal
and unique psychological script. And in the world, there are probably as many scripts as there are people. Our unique scripts grow as we do and develop, and they take
in the values, the beliefs of other people and the world around us. In the article I read, Ann Law Leumf talks about three very common cognitive scripts. She's
encountered and which don't really help people, or which narrow the range of choices that people make. She talks in the article about something she calls SQL script, S
E Q U E L, meaning that if we make choices in life, we must make those choices according to what we've already done. It must follow on from previous decisions. She
says, we stick to careers we no longer enjoy. Remain. in relationships that no longer serve us and avoid exploring opportunities that seem inconsistent with who we have
Restricted Mode
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JSON

CONCLUSION

The development of the Serverless Voice-to-Text Converter using AWS successfully demonstrates the potential of serverless architecture in creating scalable, efficient, and cost-effective cloud-based applications. By leveraging Amazon S3, AWS Lambda, and Amazon Transcribe, the system provides an automated and reliable pipeline for converting speech into text without the need for continuous server management.

This project addressed common limitations found in traditional voice processing systems—such as high infrastructure cost, poor scalability, and complex deployment—by applying modern, event-driven cloud technologies. The serverless design ensures that resources are used only when needed, making the solution suitable for both small-scale use cases and enterprise-level expansion.

Through effective data gathering, problem-solving, and technical implementation, the system was able to handle a variety of audio inputs, maintain high transcription accuracy, and deliver clean, structured text output. The modular design also allows for future extensions, such as adding real-time transcription, multi-language support, or integration with additional services like sentiment analysis or translation.

In conclusion, this project not only showcases the practical benefits of AWS's serverless ecosystem but also opens the door for building smarter, automated voice applications that are accessible, affordable, and easy to maintain.

FUTURE WORK

1. Real-Time Transcription

Currently, the system processes pre-recorded audio files. Future versions can integrate Amazon Transcribe Streaming to support real-time voice-to-text conversion, which is essential for applications like live captioning, virtual meetings, or customer support bots.

2. Multi-Language and Translation Support

Extending the system to handle multi-language transcription and integrating with Amazon Translate would enable cross-language communication and make the tool usable in global environments.

3. Mobile and Web App Integration

Building a user-friendly front-end (mobile or web app) would allow users to record or upload audio directly, view transcripts, and manage files, improving accessibility and user engagement.

4. Natural Language Processing (NLP) Features

Post-processing transcripts using NLP techniques could enable:

- Summarization of lengthy transcripts
- Keyword extraction
- Sentiment analysis
- Named entity recognition (NER)

This would turn raw transcription data into actionable insights.

5. User Authentication & Role-Based Access

Adding user login functionality with AWS Cognito or another authentication system would allow multiple users to interact with the system securely, each with different access levels or transcription quotas.

6. Database Integration

Integrating the output with Amazon DynamoDB or RDS would enable structured storage of transcripts, support advanced querying, and facilitate analytics dashboards.

7. Cost and Usage Monitoring

Implementing tools like AWS Budgets, Cost Explorer, or CloudWatch dashboards would help track usage, detect anomalies, and optimize costs for larger-scale deployments.

REFERENCES

Amazon Web Services. (2022). ***Amazon Transcribe Developer Guide*****. Retrieved from:**
https://docs.aws.amazon.com/transcribe

Amazon Web Services. (2022). ***AWS Lambda Developer Guide*****. Retrieved from:**
https://docs.aws.amazon.com/lambda

Amazon Web Services. (2022). ***Amazon S3 User Guide*****. Retrieved from:**
https://docs.aws.amazon.com/s3

AWS Blog. (2021). ***Building a Serverless Speech-to-Text Pipeline with Amazon Transcribe*****. Retrieved from: https://aws.amazon.com/blogs/machine-learning/**