1T33: Cloud Architecture

Lab- 7A: Building a Video Streaming Platform for Startup

Overview:

In this lab, you will design and deploy a basic yet scalable video streaming platform for a hypothetical startup. The project will involve setting up the required architecture to manage video uploads, storage, transcoding, and delivery to users, leveraging various AWS services to ensure scalability, high availability, and low latency.

Project Scenario:

You are part of a new startup tasked with creating a video streaming platform similar to YouTube or Netflix. Your platform needs to handle video uploads from users, transcode videos into multiple formats for different device types, and deliver those videos to users efficiently. Your goal is to build an MVP (Minimum Viable Product) that can later scale as your user base grows.

Objectives:

- Understand how to architect a video streaming platform using AWS services.
- Set up an AWS S3 bucket for video storage.
- Automate the transcoding process using AWS Elastic Transcoder or AWS MediaConvert.
- Deploy a content delivery network (CDN) for fast video streaming using AWS CloudFront.
- Ensure high availability and scalability using Auto Scaling Groups and Load Balancers.
- Understand video streaming protocols (HLS, DASH) and their implementation on AWS.

Expected Outcomes:

1. Video Upload & Storage: You will set up an infrastructure to allow users to upload videos, which will be stored securely in Amazon S3.

2. Video Transcoding: You will implement automatic transcoding of uploaded videos using AWS Elastic Transcoder or AWS MediaConvert to support multiple video formats and resolutions.

3. Content Delivery & Streaming: You will deploy AWS CloudFront as a CDN to ensure efficient video streaming with low latency across different geographical regions.

4. Scalable Architecture: You will deploy an architecture that automatically scales based on user demand, ensuring high availability during traffic spikes.

System Requirements:

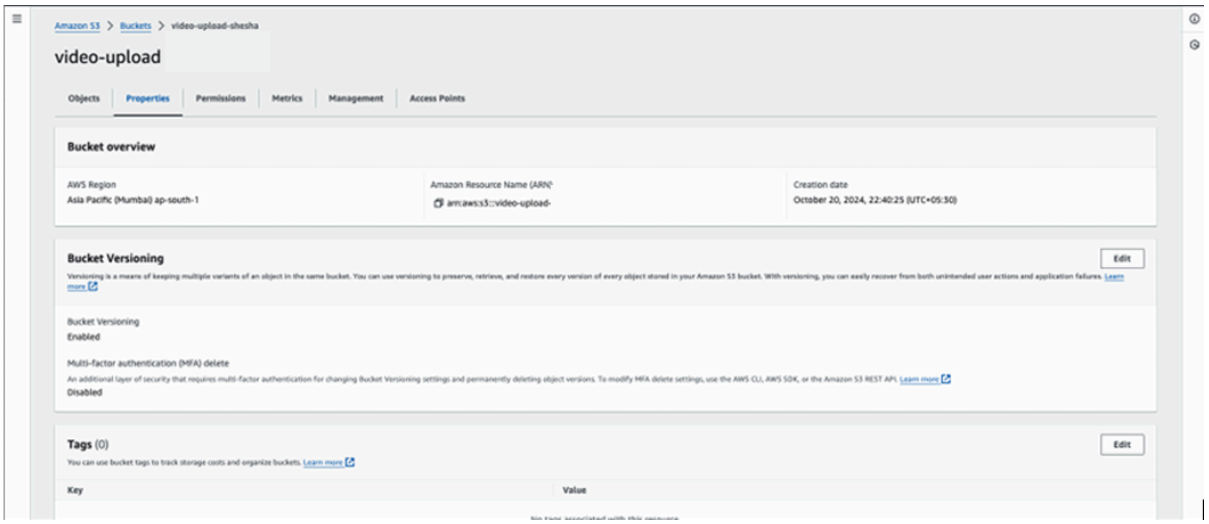
- AWS Account: Ensure access to an active AWS account.
- AWS Services: Permissions to access AWS S3, Elastic Transcoder, MediaConvert, CloudFront, EC2, Auto Scaling, RDS (optional), and IAM.
- Browser: A modern web browser for accessing the AWS Management Console.
- Internet Connection: Stable connection for AWS Console interaction and streaming tests.

Step-by-Step Laboratory Procedure:

Phase 1: Video Upload and Storage Setup

Step 1: Create an S3 Bucket for Video Storage

1. Sign in to the AWS Management Console and navigate to S3.
2. Click Create bucket.
3. Name the bucket (e.g., `startup-video-platform`), choose a region, and ensure public access is blocked for video uploads.
4. Click Create bucket to finalize.



Step 2: Enable Object Versioning and Encryption

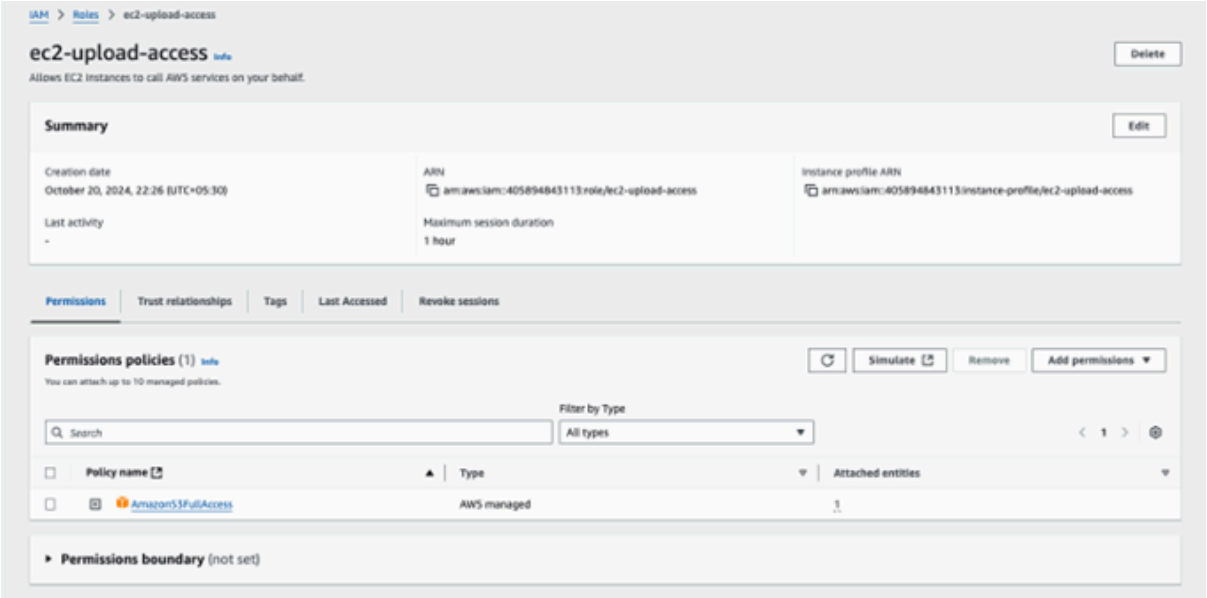
1. In the Properties tab of your S3 bucket, enable Versioning to keep previous versions of videos.

2. Enable Server-Side Encryption using AES-256 for security compliance.

Step 3: Configure Upload Permissions

1. Navigate to the Permissions tab and create an IAM role that allows users to upload videos securely to your S3 bucket.

2. Attach this role to an EC2 instance or Lambda function that handles video uploads.



Phase 2: Automating Video Transcoding

Step 4: Set Up AWS Elastic Transcoder or AWS MediaConvert

1. In the AWS Management Console, search for Elastic Transcoder or MediaConvert.

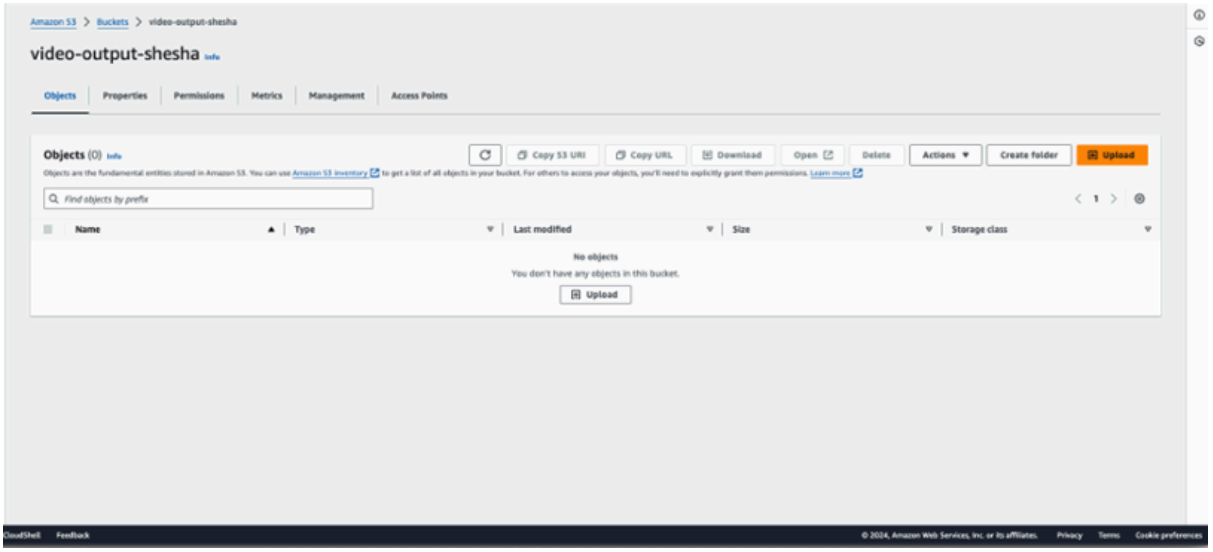
2. Click Create New Pipeline.

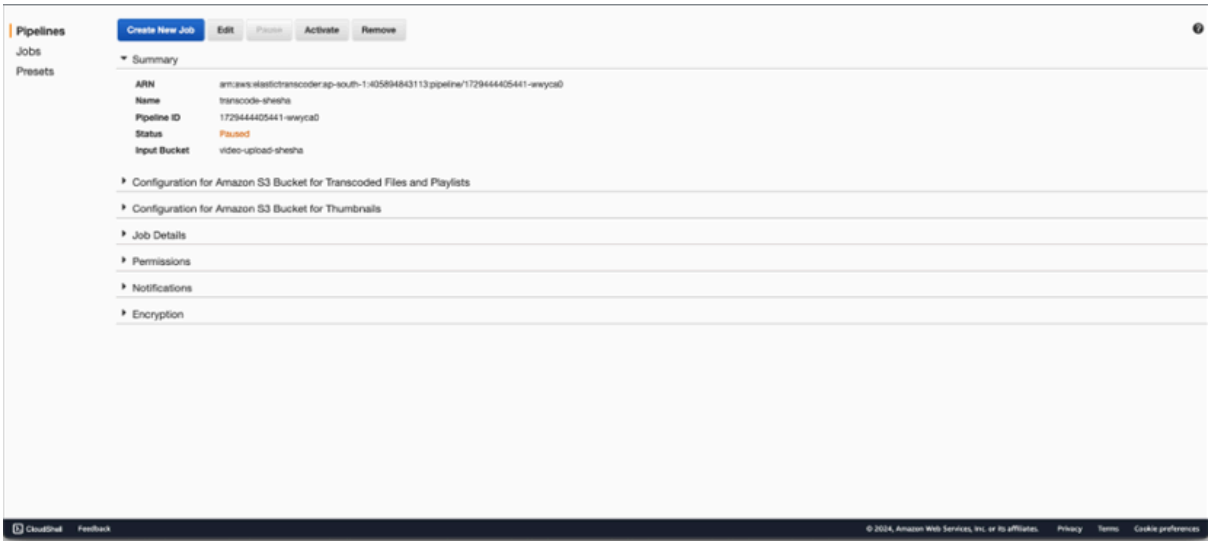
   - Choose your S3 bucket as the input source.

   - Create a new S3 bucket for storing transcoded outputs (e.g., `startup-video-output`).

   - Configure the transcoder to output multiple formats (H.264, MPEG-4, etc.).

3. Set transcoding presets (low, medium, and high quality) for various devices and streaming needs.





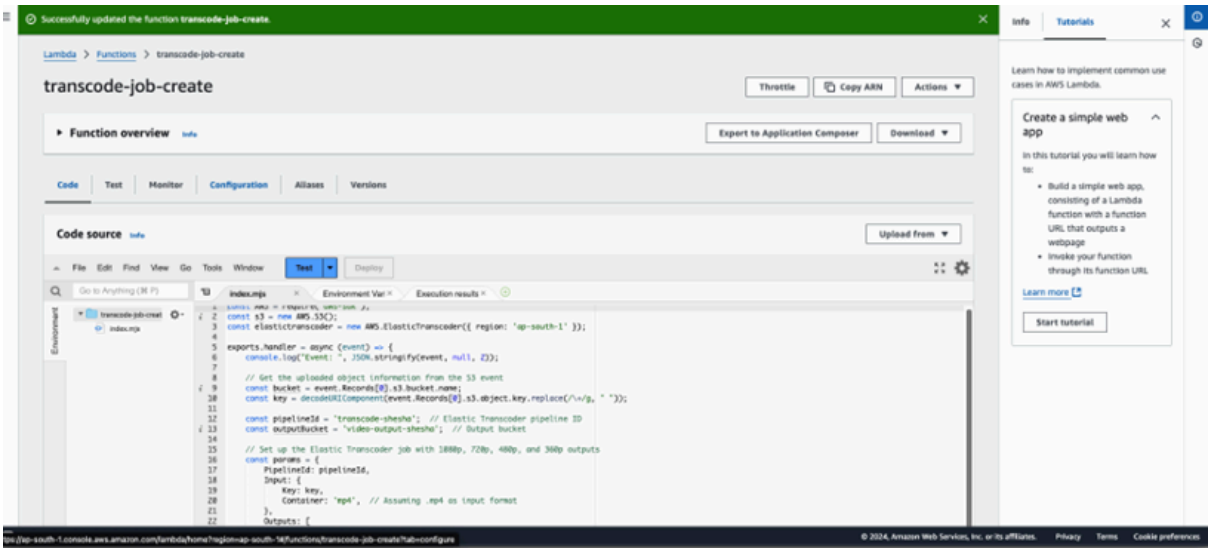
Step 5: Automate the Transcoding Process

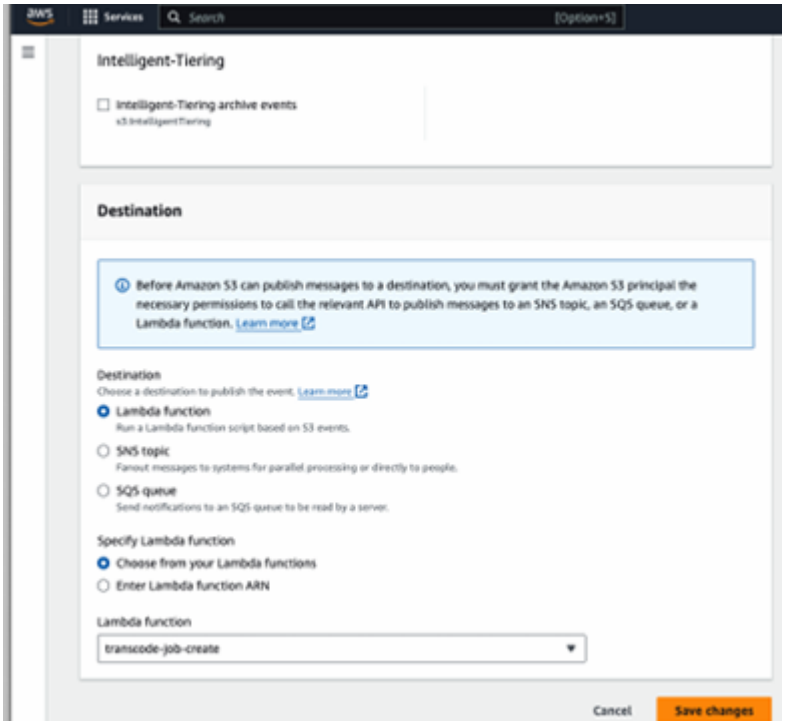
1. Set up an S3 Event Notification to trigger a Lambda function whenever a new video is uploaded.

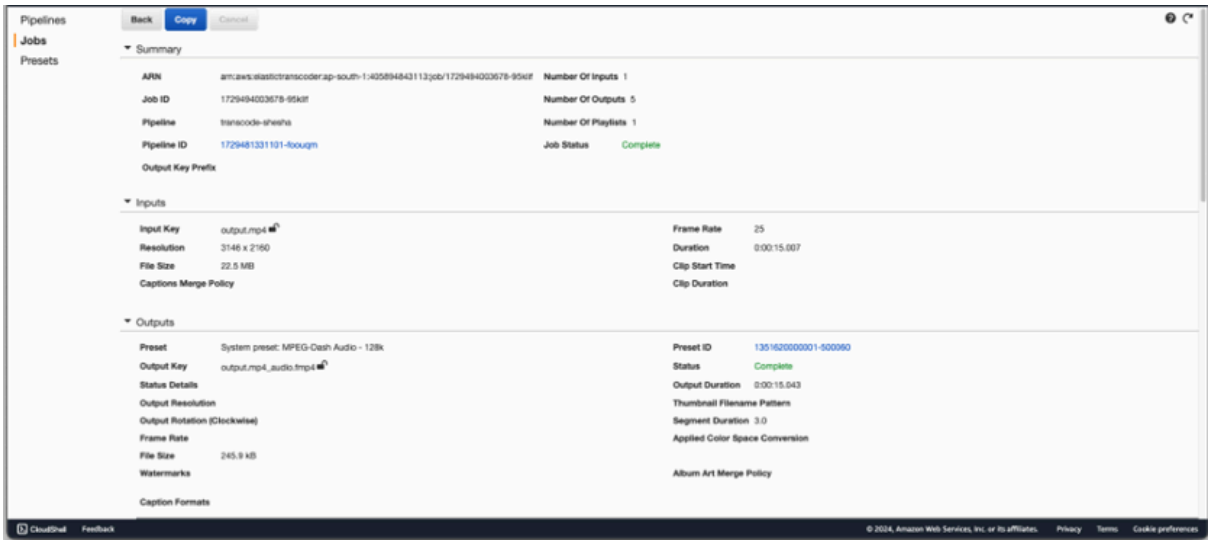
2. The Lambda function will initiate the transcoding process through AWS Elastic Transcoder or MediaConvert.

3. Ensure the output videos are stored in the designated S3 output bucket.

1T33: Cloud Architecture







Phase 3: Content Delivery and Streaming

Step 6: Set Up AWS CloudFront for Content Delivery

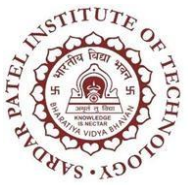
1. Go to CloudFront in the AWS Management Console.
2. Click Create Distribution and choose Web Distribution.
3. Set your S3 output bucket as the origin, which contains the transcoded video files.
4. Enable HTTP/2 and GZIP Compression for faster content delivery.
5. Set up CNAMEs and SSL certificates for custom domain access to your videos.

Step 7: Configure HLS or DASH for Streaming

1. In your S3 bucket, ensure that video formats are segmented into HLS or DASH for adaptive streaming.
2. In CloudFront, configure behaviors for serving these segments, ensuring that your platform supports adaptive bitrate streaming based on user device bandwidth.

Phase 4: Ensuring Scalability and Availability

Step 8: Set Up EC2 Auto Scaling and Load Balancing



1T33: Cloud Architecture

1. Launch an EC2 instance with your front-end video platform (e.g., a website for uploading and watching videos).
2. Create an Auto Scaling Group to ensure the platform can scale based on traffic demand.
   - Set a minimum and maximum number of instances based on expected traffic.
3. Configure an Elastic Load Balancer to distribute traffic across instances.

Step 9: Set Up a Database for User Information (Optional)

1. Deploy an RDS instance for storing user details, video metadata, and streaming preferences.
2. Ensure that the RDS instance is set to multi-AZ deployment for high availability.

Phase 5: Security and Monitoring

Step 10: Implement IAM Policies for Access Control

1. Create and attach IAM roles to restrict who can upload, transcode, or access specific resources.
2. Implement security best practices like least privilege and MFA (Multi-Factor Authentication) for access control.

Step 11: Set Up Monitoring and Logging

1. Enable CloudWatch Logs to monitor EC2 instance performance and CloudFront distribution analytics.
2. Set up alarms for key metrics like CPU usage, network throughput, and scaling activity.

Bonus Tasks (Optional):

- Add support for user authentication using AWS Cognito to allow users to sign up and log in securely.

- Set up a serverless architecture for video transcoding and delivery using AWS Lambda for greater scalability and reduced costs.

Summary

By completing this lab, you will gain hands-on experience in building and managing a scalable, secure, and efficient video streaming platform using AWS services. This setup will allow the hypothetical startup to grow from an MVP stage to a production-ready solution that can handle increased traffic and user demand, while adhering to cloud security and best practices.

Conclusion:

In conclusion, this lab provides a hands-on approach to building and deploying a scalable video streaming platform using AWS services. By utilizing Amazon S3 for video storage, AWS Elastic Transcoder or MediaConvert for automated video processing, and AWS CloudFront for efficient content delivery, you can streamline video uploads, processing, and playback across multiple devices.

The integration of Auto Scaling Groups and Load Balancers ensures high availability and scalability, allowing the platform to seamlessly handle growing traffic as the user base expands. Additionally, understanding video streaming protocols like HLS and DASH enhances performance and compatibility,