**Cloud-Based Digital Signage Solution with AWS Elemental Media-Live**

*A Course Project Report Submitted in partial fulfillment of the course requirements for the award of grades in the subject of*

# CLOUD BASED AIML SPECIALITY (22SDCS07A)

by

**Akasapu Monika**

**2210030430**

*Under the esteemed guidance of*

**Ms. P. Sree Lakshmi**

Assistant Professor,

Department of Computer Science and Engineering



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**K L Deemed to be UNIVERSITY**

*Aziznagar, Moinabad, Hyderabad,*

*Telangana, Pincode: 500075*

April 2025

# K L Deemed to be UNIVERSITY

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

## *Certificate* Picture 105

This is Certified that the project entitled **“Cloud-Based Digital Signage Solution with AWS Elemental MediaLive”** which is a experimental & theoretical & Simulation& hardware work carried out by Akasapu Monika in partial fulfillment of the course requirements for the award of grades in the subject of **CLOUD BASED AIML SPECIALITY**, during the year **2024-2025**. The project has been approved as it satisfies the academic requirements.

**Ms.P.Sree Lakshmi Dr. Arpita Gupta**

**Course Coordinator Head of the Department**

**Ms. P. Sree Lakshmi**

**Course Instructor**

# CONTENTS

Page No

1. Introduction 4
2. AWS Services Used as part of the project 5
3. Steps involved in solving project problem statement 7
4. Stepwise Screenshots with brief description 10
5. Learning Outcomes 18
6. Conclusion 19
7. References 20

## 1. INTRODUCTION

Digital signage has emerged as a transformative tool for delivering dynamic content across industries such as retail, transportation, education, and entertainment. With the advancement of pervasive display technologies, traditional static boards have evolved into intelligent, networked systems capable of real-time, context-aware communication [1]. These modern systems enhance user experience and operational efficiency by enabling remote content management and adaptive messaging based on environmental factors and audience behavior.

Interactive and contextual capabilities have become central to the evolution of digital signage. The development of backend service frameworks and integration of contextual data allow for greater personalization and responsiveness in content delivery [2][3]. In addition to conveying information, digital signage now plays a key role in customer engagement—reducing perceived wait times and improving overall satisfaction in public and commercial settings [4]. This progression has expanded the value of digital signage from simple displays to intelligent, interactive service platforms.

With the advent of cloud computing, platforms like AWS Elemental MediaLive offer powerful solutions for live video encoding and distribution in digital signage networks [5]. These cloud-based tools enable scalable, centralized control of live content, making it easier for organizations to manage broadcasts across multiple locations without complex infrastructure. By creating immersive and responsive experiences, digital signage has been shown to positively influence consumer behavior and emotional responses [6], highlighting its potential as a strategic communication and marketing asset.

## 2. AWS SERVICES USED AS PART OF THE PROJECT

### 1. AWS Elemental MediaLive

MediaLive is responsible for encoding live video streams into multiple bitrate outputs using adaptive bitrate technology. This is particularly crucial for digital signage solutions where real-time updates (e.g., event broadcasts, announcements) need to be pushed across a network of displays in malls, campuses, or transportation hubs. The service supports various input sources and offers integration with cloud storage, MediaConnect, and other AWS tools, making it ideal for scalable deployments [5].

### 2. AWS Elemental MediaPackage

MediaPackage complements MediaLive by converting the encoded streams into formats like HLS and MPEG-DASH, which are compatible with modern web browsers, smart

TVs, and signage devices. It also supports just-in-time packaging, encryption, and digital rights management (DRM), ensuring that content is secure and compliant while being distributed efficiently. This helps maintain broadcast integrity across public or private signage networks [5].

### 3. Amazon CloudFront

CloudFront distributes signage content globally with low latency by caching data at AWS edge locations. This minimizes buffering and ensures consistent playback quality regardless of the viewer's location. CloudFront also supports signed URLs and georestrictions, which is beneficial for targeted signage (e.g., region-specific promotions). It can serve both live streams and on-demand content, making it a versatile part of the digital signage delivery chain [5].

### 4. Amazon S3 (Simple Storage Service)

S3 acts as the primary storage backend for media assets, layout templates, pre-recorded videos, and static graphics. With lifecycle rules, you can automate archival or deletion of outdated content, and with versioning, rollback is easy. This ensures an organized, costefficient, and scalable storage solution for any digital signage deployment [2][3].

### 5. AWS Lambda

Lambda allows you to build intelligent content scheduling without maintaining servers. It can also handle user-triggered events—like playing a video when someone walks past a sensor-enabled sign—allowing for context-aware interactivity [3].

### 6. AWS Elemental MediaConvert

Used for file-based video processing, MediaConvert enables the transformation of raw or user-uploaded videos into high-quality outputs suitable for display. For instance, a retail brand can upload promotional footage that MediaConvert transcodes into different resolutions for landscape and portrait signage. It supports features like overlays, watermarks, and ad insertion, helping automate content refinement before it reaches the display endpoints [5].

### 7. AWS Step Functions

This service helps build and manage serverless workflows that coordinate services like

Lambda, MediaConvert, and S3. For example, a new content upload to S3 can trigger a Step Function workflow that converts the video, updates the metadata, sends an email confirmation, and schedules the new content for display. These visual workflows improve automation and reduce operational overhead [3][5].

### 8. Amazon CloudWatch

CloudWatch collects metrics, logs, and events from all the AWS services in the signage pipeline. You can set alarms for stream interruptions, storage overflows, or Lambda failures. Dashboards provide a real-time view of your signage health, and logs help in debugging issues quickly—ensuring maximum uptime and reliability [5].

### 9. AWS Identity and Access Management (IAM)

IAM governs who can access what in the signage ecosystem. For instance, designers might only upload media to S3, while admins have full control over streaming and monitoring services. You can use fine-grained policies to allow or deny access to specific APIs, automate secure workflows, and ensure compliance with organizational standards [5].

### 10. AWS Elemental MediaConnect

MediaConnect acts as a high-quality, reliable transport service for ingesting and distributing live video streams. It ensures secure and low-latency video transmission to MediaLive and other processing services. In digital signage, this service can be used to bring in third-party live content or remotely stream headquarters feeds to multiple branch locations without packet loss or jitter [5].

## 3. STEPS INVOLVED IN SOLVING PROJECT PROBLEM STATEMENT

### Creating an S3 Bucket

* + The Amazon S3 bucket was initialized as a centralized repository to store all media assets needed for digital signage. This included promotional videos, static images, layout templates, and backup files that are required for content rotation and fallback scenarios.
  + Using well-defined folder structures and naming conventions ensures efficient management, such as keeping video content separate from static files. Versioning was enabled in the bucket to track changes to assets over time and to ensure that old versions could be restored if necessary.
  + Bucket policies were created to ensure strict access control, so only authorized services or users could access or modify the assets. Additionally, lifecycle policies were set to automatically archive or delete old content, optimizing storage costs [2][3].

### 2. Uploading Media Files

* + After the S3 bucket was set up, the media files were uploaded using either the AWS CLI (Command Line Interface) for batch processing or the AWS Management Console for more manual uploads.
  + Videos, images, fallback content, and design assets were carefully categorized and tagged for easy identification. Metadata like file type, duration, and category were included to enable better management and retrieval of content.
  + AWS services like Lambda and MediaConvert were configured to automate workflows such as content transformation and scheduling. For instance, Lambda could automatically tag new uploads for specific signage zones or display times, ensuring content remains up-to-date [2].

### 3. Setting Up AWS Elemental MediaLive

* + AWS Elemental MediaLive was used to create a real-time live video channel capable of encoding multiple video inputs simultaneously. MediaLive allows the encoding of video streams into multiple adaptive bitrate outputs, ensuring content is delivered smoothly across varying network conditions.
  + Settings such as frame rate, resolution, and audio bitrate were customized to meet the specific requirements of signage displays, with higher resolutions (e.g., 1080p) chosen for large, high-definition screens.

### 4. Creating a MediaLive Input

* + A critical part of the process was defining a secure and stable input source for the MediaLive channel. Input could be live video feeds from cameras for events or looping pre-recorded content stored in S3.
  + The security of these input sources was ensured by configuring proper input security groups, which defined which IP addresses were authorized to send content to the MediaLive channel.
  + This step was critical to ensuring that the video content being streamed was from trusted sources, and proper safeguards were in place to prevent unauthorized access to the video feed [5].

### 5. Configuring AWS Elemental MediaPackage

* + After encoding the video, AWS Elemental MediaPackage was integrated with MediaLive to transcode the live video streams into multiple HTTP-based adaptive formats like HLS (HTTP Live Streaming) and MPEG-DASH.
  + These formats allow the video content to be delivered smoothly across various devices, from smart TVs to mobile phones. MediaPackage also provided additional features like content encryption using SPEKE DRM to ensure secure distribution [5].

### 6. Generating Endpoints in MediaPackage

* + MediaPackage generated endpoints (URLs) for each stream format, making it easy to distribute content to end devices.

* + By generating distinct endpoints for different formats (e.g., HLS, DASH), the solution was able to cater to various screen sizes and network conditions.

### 7. Integrating AWS CloudFront

* + To distribute the content efficiently to geographically distributed displays, AWS CloudFront was used as a Content Delivery Network (CDN). A CloudFront distribution was created, with MediaPackage endpoints as the origin for the content.
  + CloudFront ensures faster delivery of content by caching media closer to end-users at edge locations across the globe.

### 8. Setting Up IAM Roles and Permissions

* + IAM roles and policies were carefully configured to enforce secure communication between services. For instance, MediaLive was granted permission to read from S3, and MediaPackage was allowed to access the MediaLive output.
  + Users were assigned fine-grained roles based on their responsibilities—content editors, system admins, and analytics viewers—all while following the principle of least privilege [5].

### 9. Automating Content Processing with MediaConvert (Optional Pre-recorded Content)

* + For the handling of pre-recorded videos, AWS Elemental MediaConvert was used to transcode video files into broadcast-ready formats suitable for display in digital signage systems.
  + MediaConvert also allowed for the addition of visual overlays, intro/outro sequences, and watermarks to videos, enhancing branding and ensuring that content was compliant with legal requirements.

### 10. Testing, Monitoring, and Optimization with CloudWatch

* + Finally, the system was thoroughly tested to simulate both live and on-demand playback scenarios. AWS CloudWatch was configured to track real-time metrics such as the health of MediaLive channels, input frame drops, CDN errors from CloudFront, and the performance of Lambda functions handling content rotation.
  + CloudWatch Alarms were set up to notify administrators of any critical issues, ensuring rapid responses to potential problems.

## 4. STEPWISE SCREENSHOTS WITH BRIEF DESCRIPTION

**Step 1:** Create General Purpose S3 Bucket. Upload the video files and index.html into the S3 bucket. The listed files include MOV video files, an M3U8 playlist file, an HTML file, and an output folder. These files are essential for media processing and streaming using AWS services like MediaLive and MediaPackage.

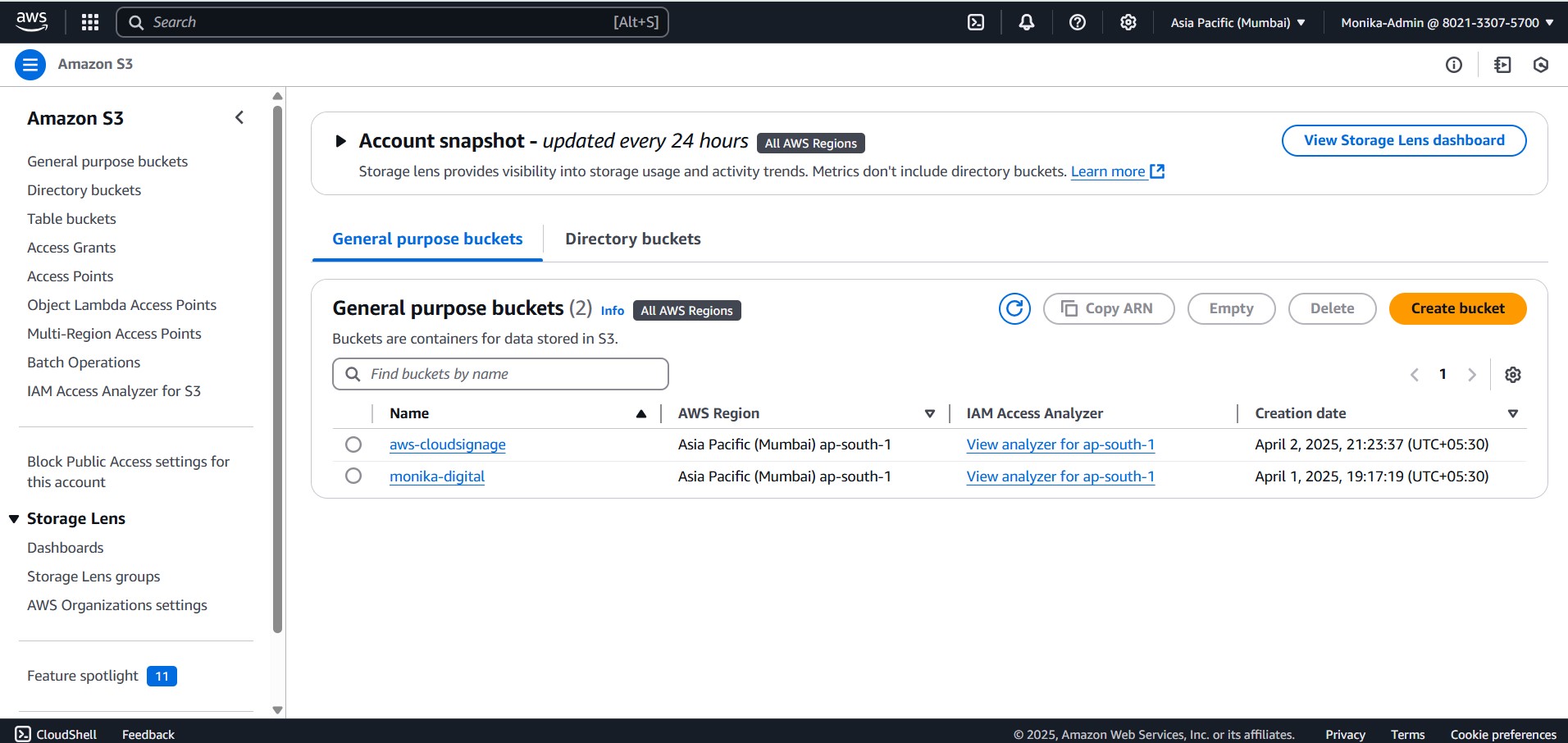


Fig 4.1: Creating General Purpose Bucket.

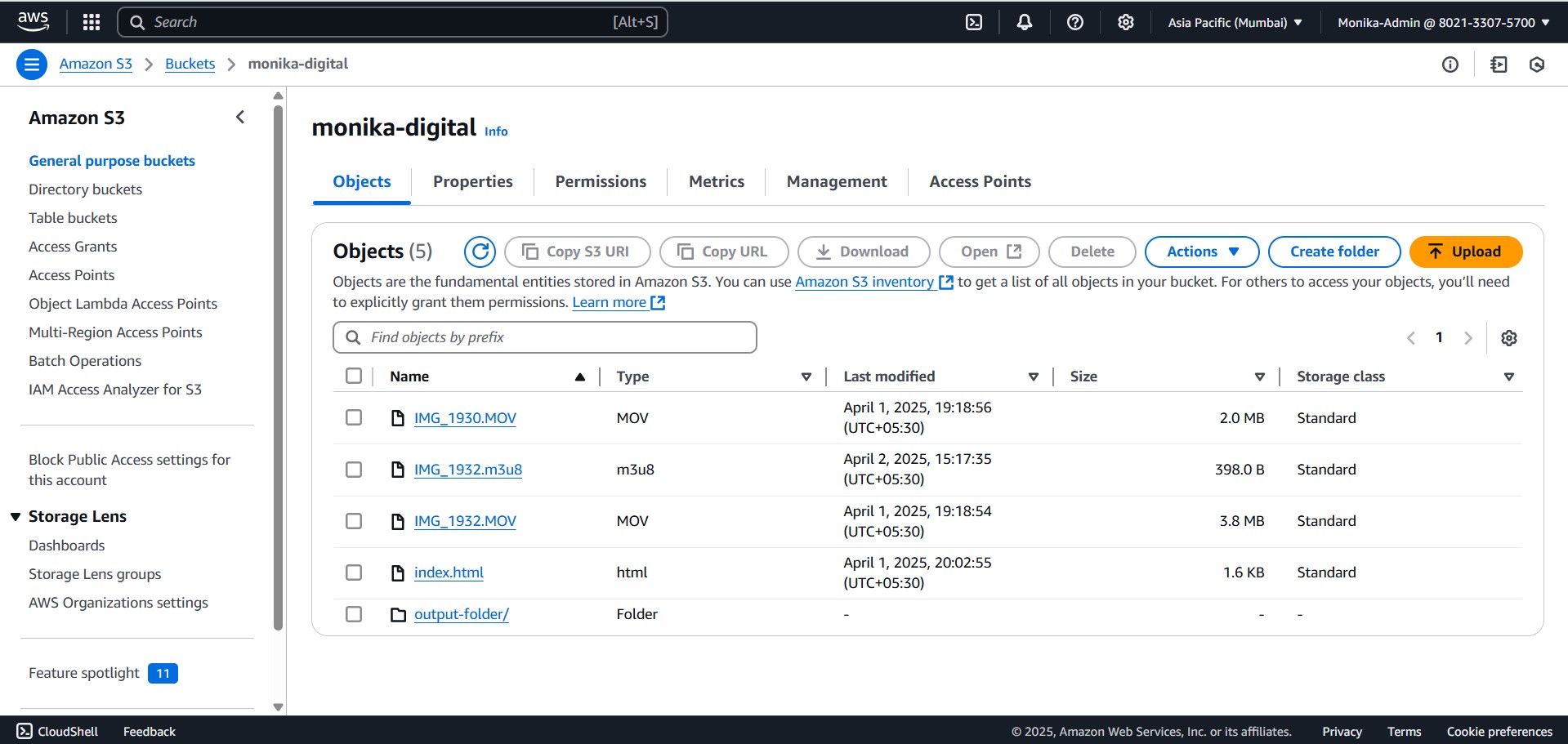


Fig 4.2: Add files into the bucket

Step 2: Update the bucket policy

"Sid": "PublicReadGetObject",

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:GetObject",

"Resource": “arn:aws:s3:::aws-cloudsignage/\*”

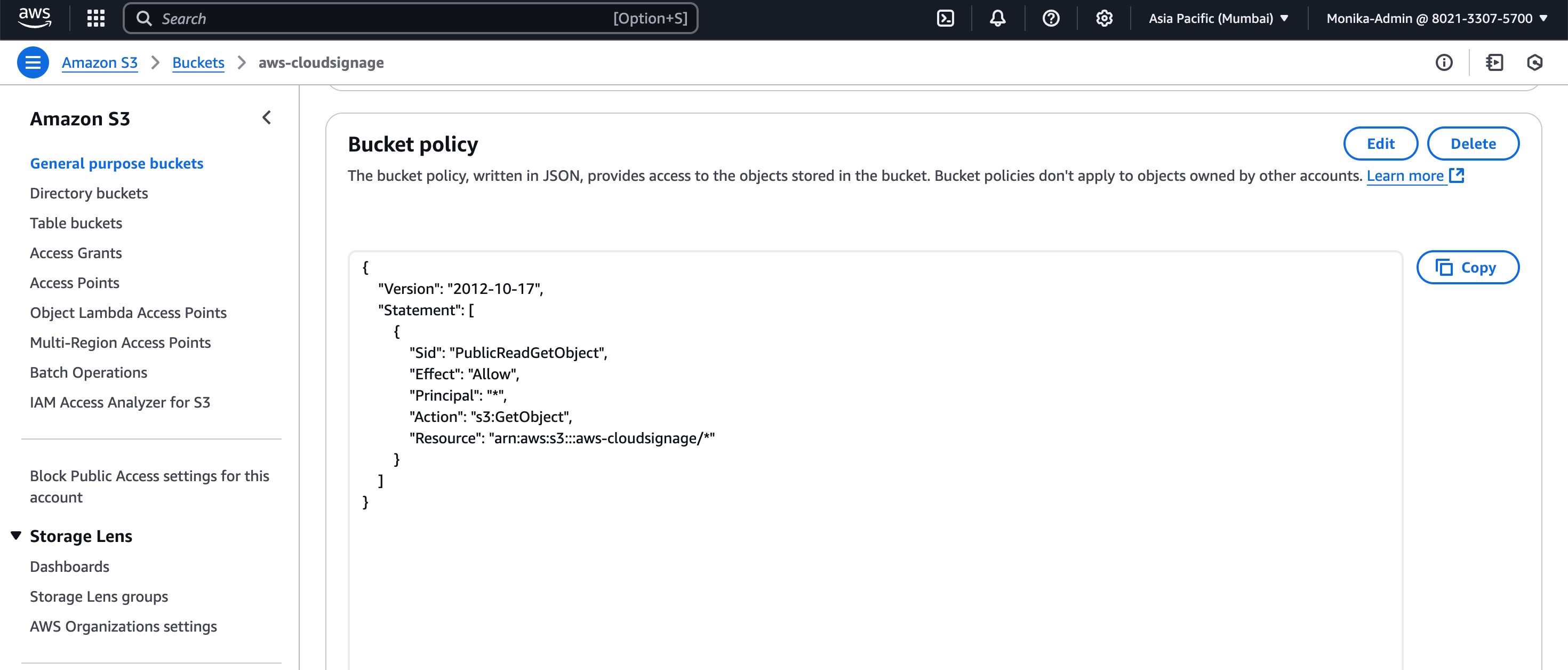


Fig 4.3: Updated Bucket Policy (IAM ).

Step 3: To create the job, AWS MediaConvert was configured to specify the MOV video files as input, and HLS as the output format. The conversion settings were optimized for the best video quality, ensuring adaptive bitrate streaming to support different devices and network conditions, with the output stored in the designated S3 output folder for easy access and distribution.

Step 4: Set up HTTPS Output Groups In MediaConvert, the output group configuration must be set to generate files that are compatible with HTTPS protocols. You define an HLS (HTTP Live Streaming) or DASH (Dynamic Adaptive Streaming over HTTP) output format, both of which can be delivered securely over HTTPS.

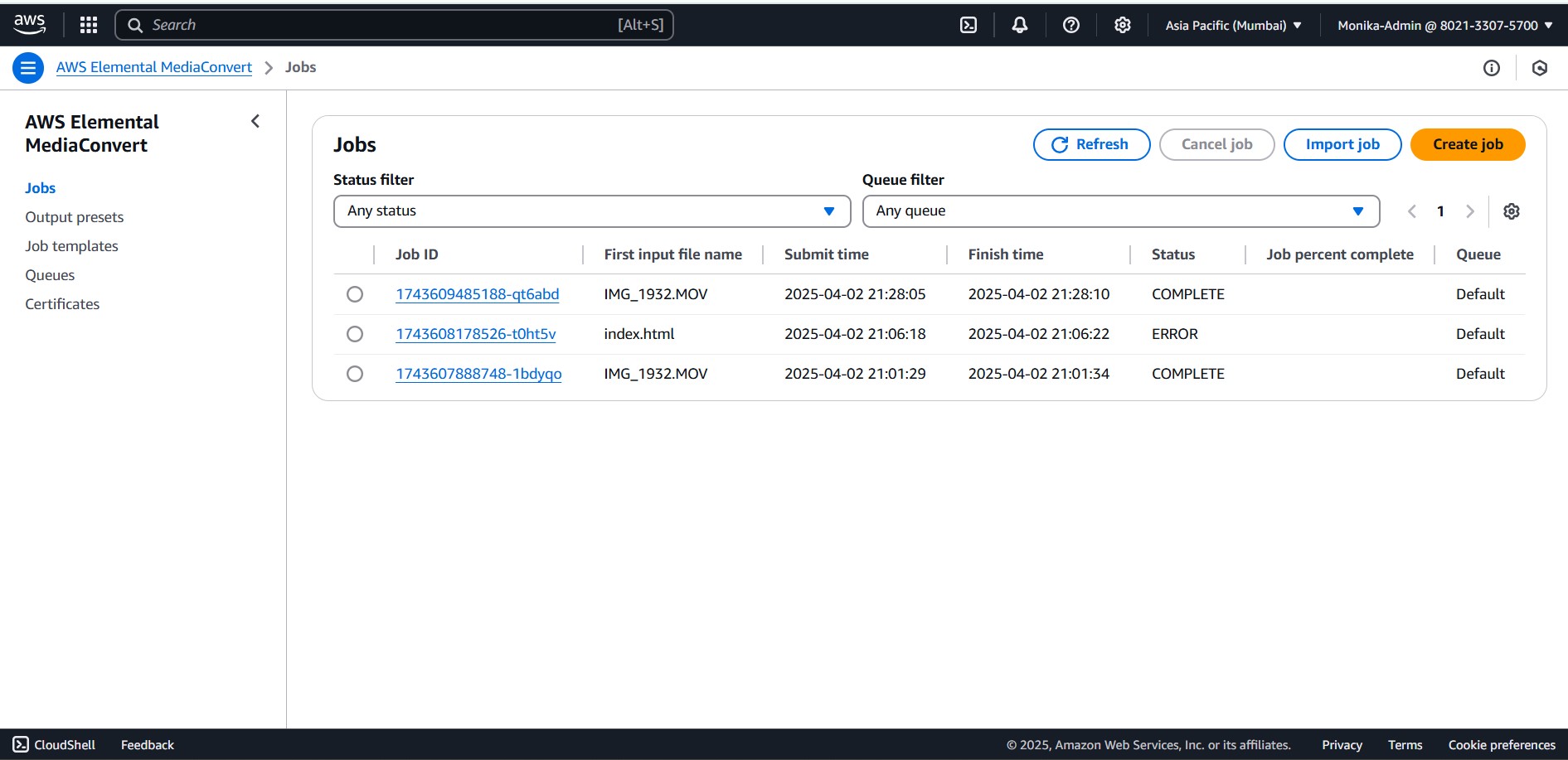


Fig 4.4: Create jobs in MediaConvert to create https protocols.

Step 5:Secure S3 Bucket Configuration

The S3 bucket designated as the output destination must have HTTPS endpoints enabled. This ensures that all files delivered from S3 (via MediaConvert) will use HTTPS, providing encrypted data transmission over the internet. You can enable S3 Transfer

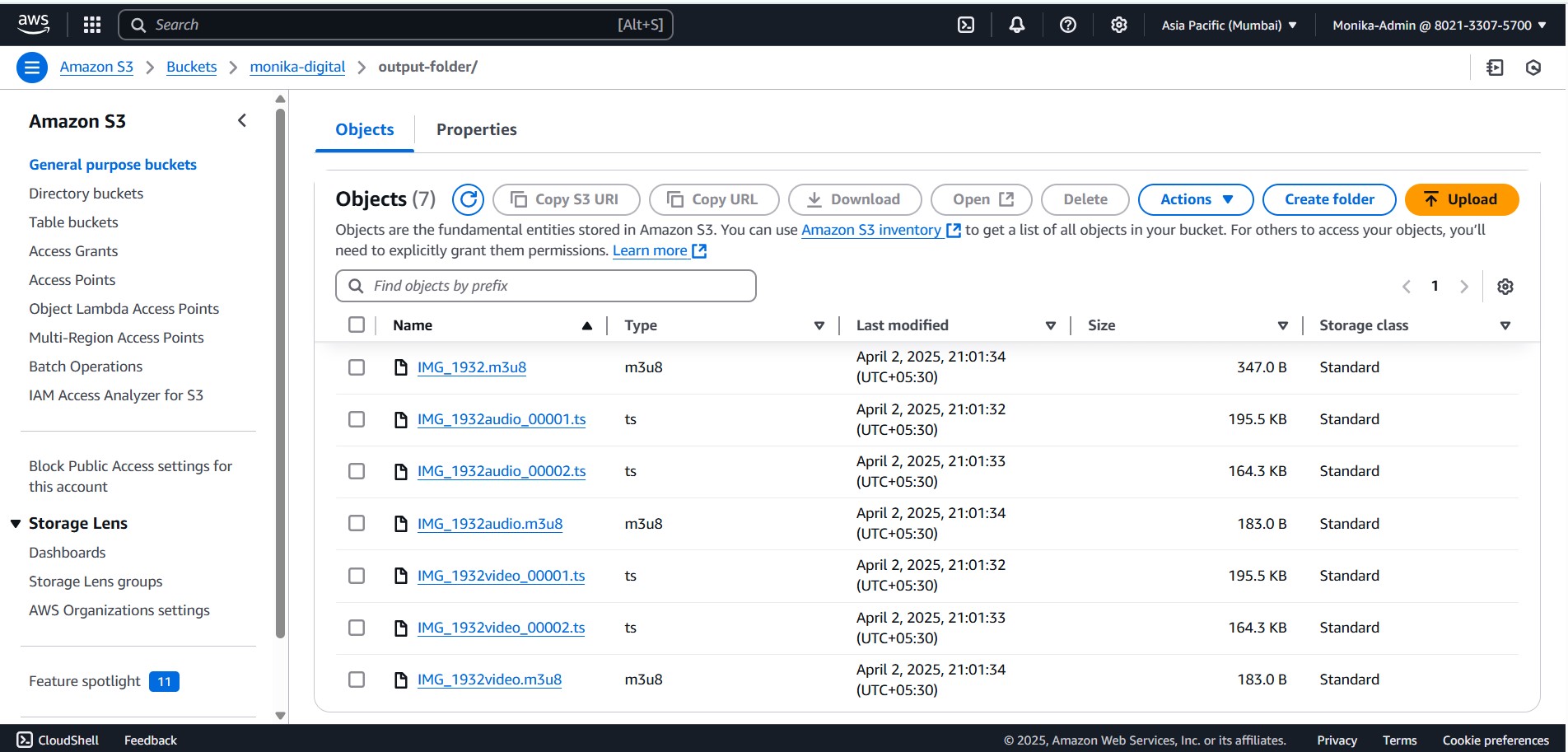


Fig 4.5: Output Folders are added into the S3 bucket.

Step 6:M3U8 Playlist Creation

When a MediaConvert job is executed, the M3U8 playlist file is generated as part of the output. This file contains references to the audio or video segments, allowing the player to request specific segments based on network conditions or device capabilities. In the case of audio-only output, the M3U8 file will list the individual audio chunks (typically .aac or .mp3), ensuring smooth playback across devices with varying bandwidths.

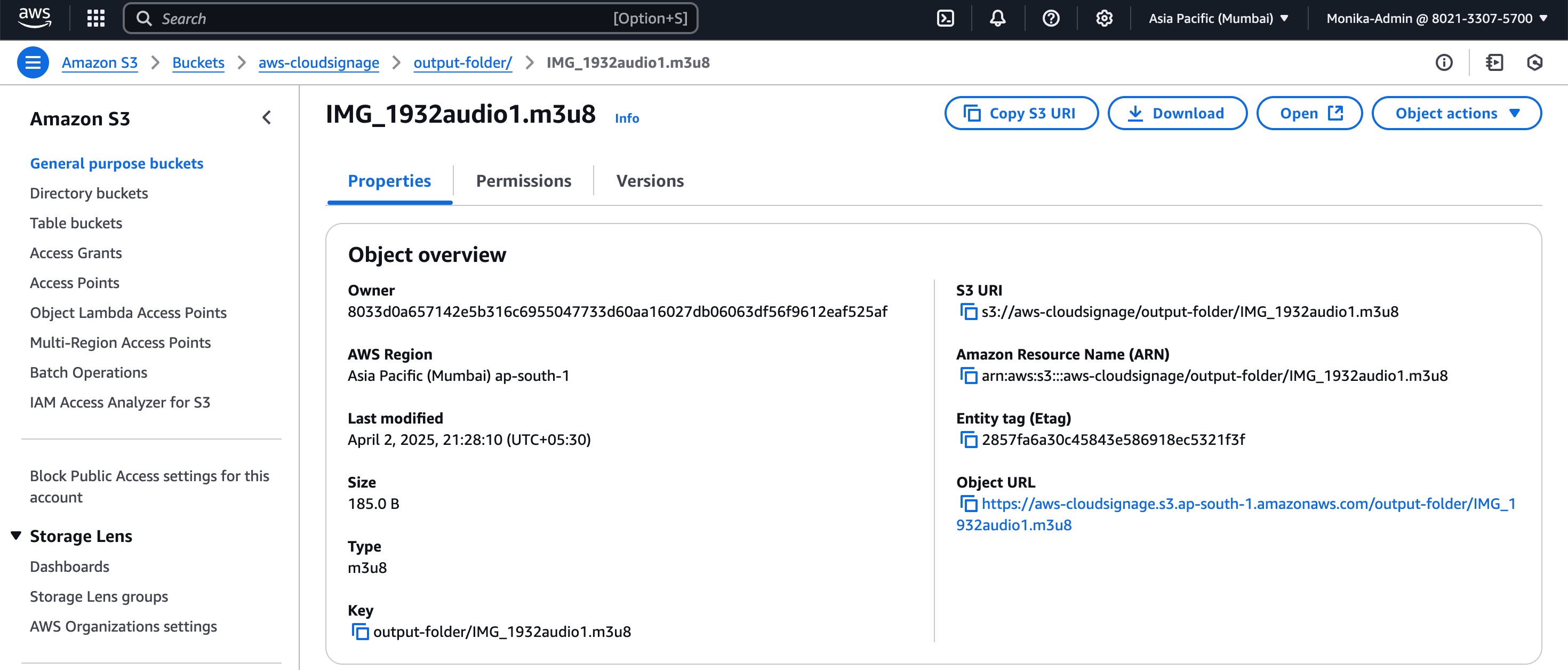


Fig 4.6: The Audio Output m3u8 is created.

Step 7: The MOV video is converted to HLS and stored in S3 bucket, later the object URL is used for board-cast.

[https://aws-cloudsignage.s3.ap-south-1.amazonaws.com/output-folder/ IMG\_1932audio1.m3u8](https://aws-cloudsignage.s3.ap-south-1.amazonaws.com/output-folder/IMG_1932audio1.m3u8).

The URL provided uses HTTPS, which ensures that all content is securely delivered to the end-user devices. Whether it's live or on-demand, the use of HTTPS ensures encrypted communication between the S3 bucket (or CloudFront) and the display devices, safeguarding the media content from interception.

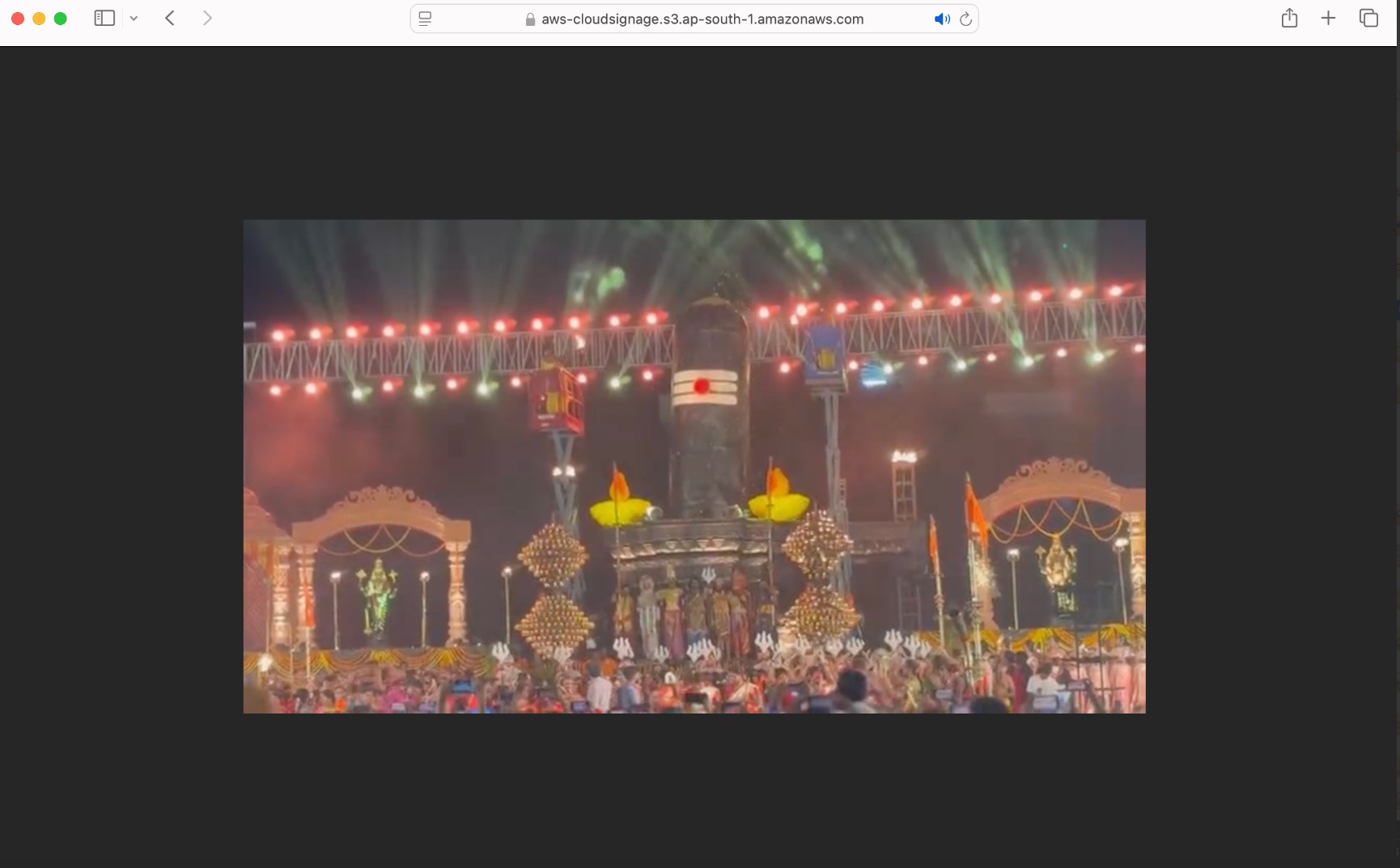


Fig 4.7: HTTPS video is generated in S3 bucket.

Step 8: CloudFront is used for the Content Distribution among different users.

CloudFront is a Content Delivery Network (CDN) that caches content at edge locations around the world, reducing latency by serving the content from the nearest location to the user. In the case of digital signage, this ensures that video streams, including the HLS playlist (.m3u8), are delivered with minimal delay, regardless of the geographic location of the viewers.

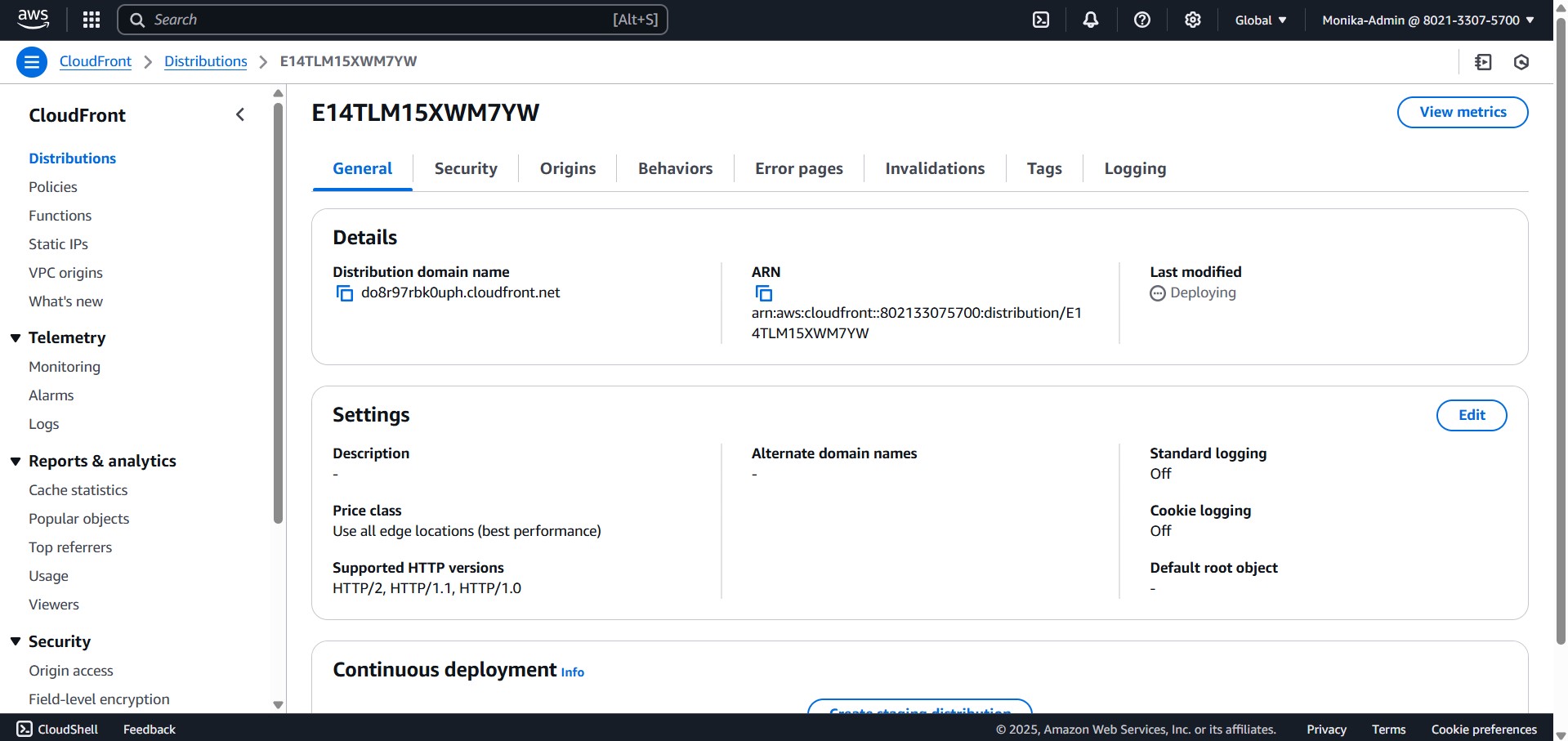


Fig 4.8: Create CloudFront for Distribution of content.

Step 9:<https://do8r97rbk0uph.cloudfront.net/index.html>,this link is used for content distribution.CloudFront automatically distributes the content, such as the index.html page or associated media files, across its global network of edge servers. When a user requests the content, CloudFront routes the request to the nearest edge location, reducing latency and ensuring faster load times, no matter where the viewer is located. This is essential for digital signage deployments that need to reach users across various regions with consistent performance.

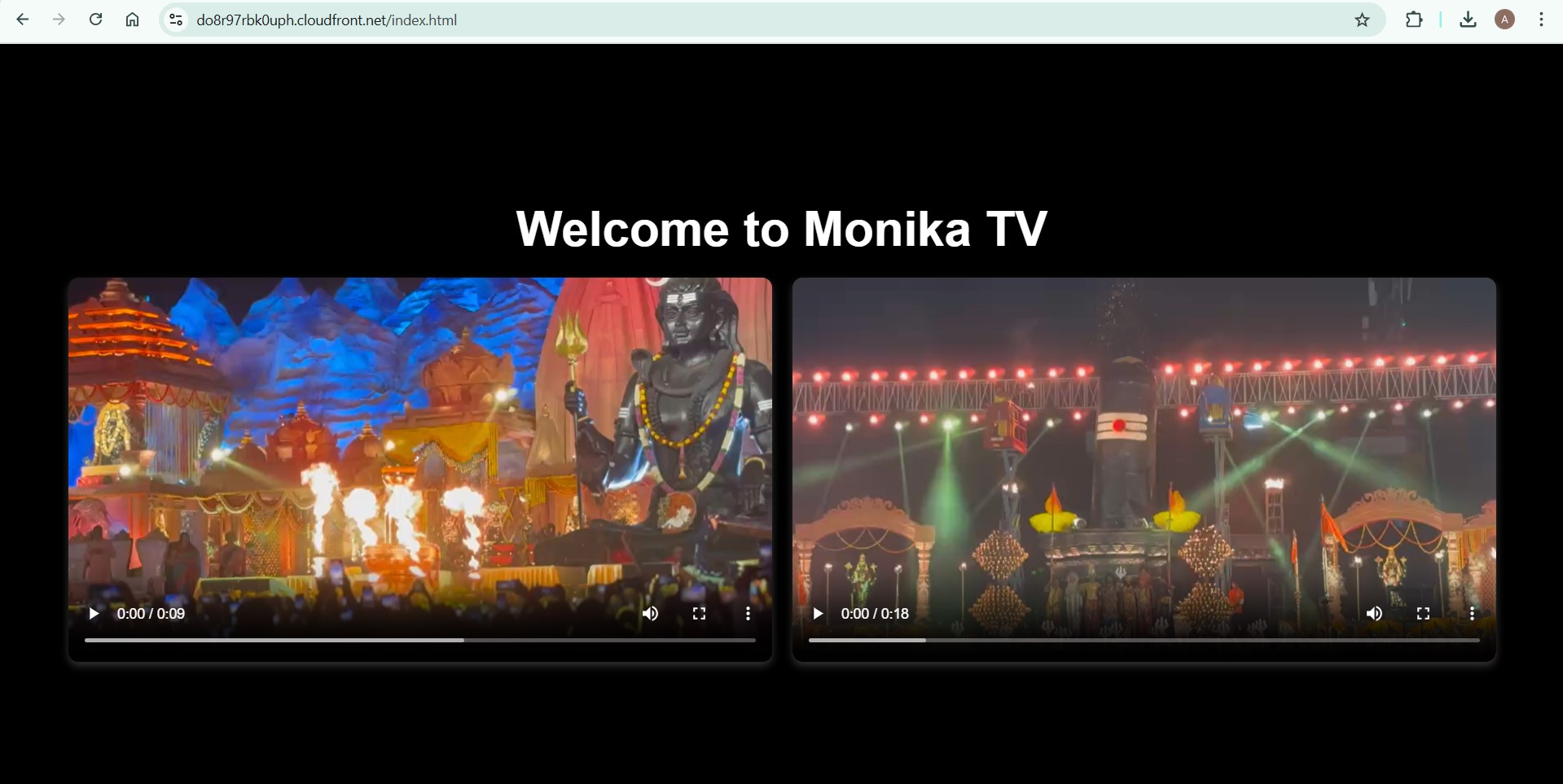


Fig 4.9: Content is distributed through the cloud-front.

Step 10: AWS Elemental MediaLive is a video service that lets you easily create live outputs for broadcast and streaming delivery.MediaLive allows you to configure video streams and set up different outputs, ensuring that live content is delivered with the appropriate encoding settings and formats. You can create live video channels, manage multiple video outputs for different devices, and even customize your stream with the required encoding parameters for adaptive bitrate streaming.

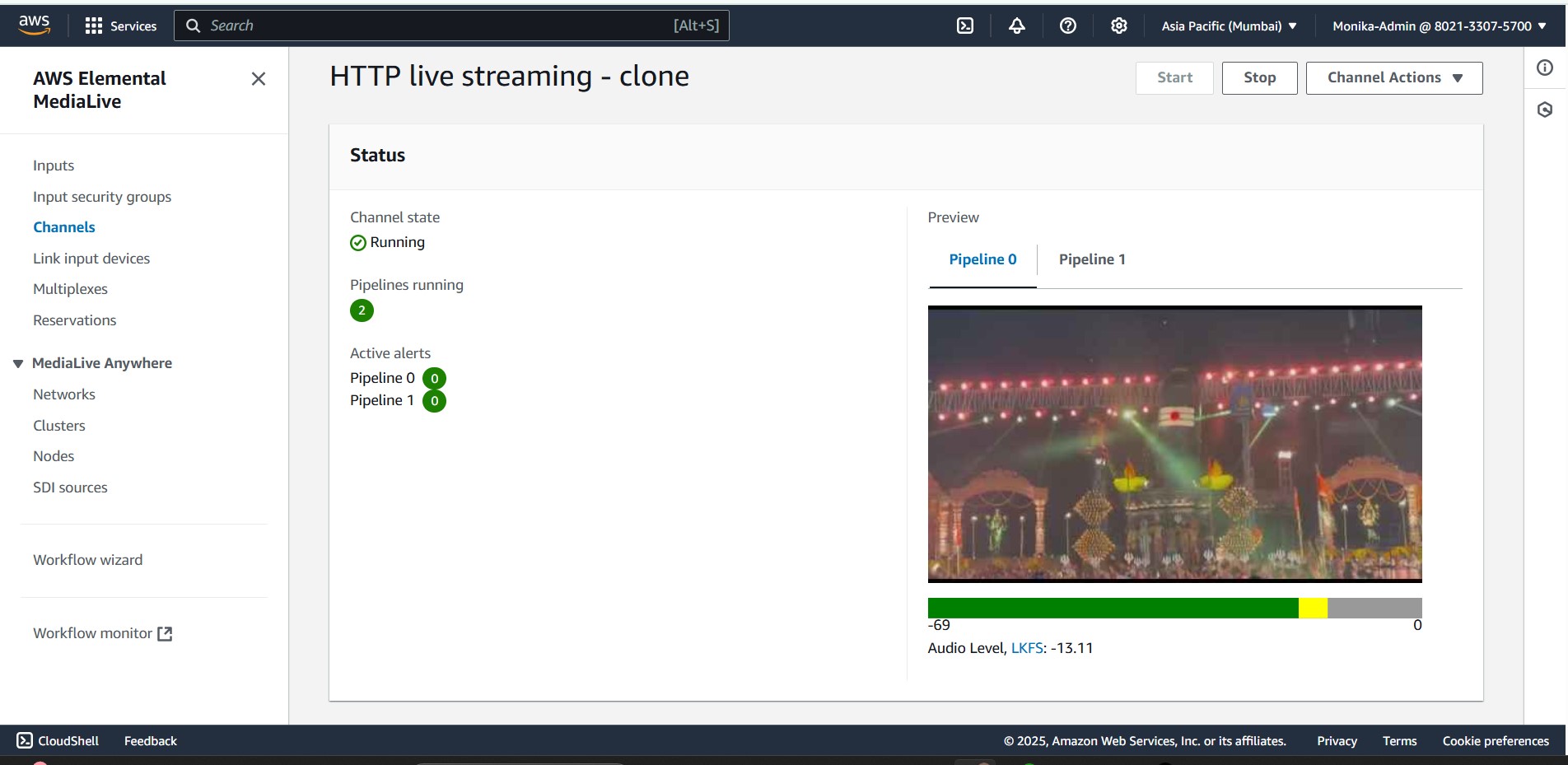


Fig 4.10: Channel is created for content scheduling.

Step 11: AWS CloudWatch is used to monitor the MediaLive Channels and the protocols status while delivering the content.These logs provide detailed information that can be used for quick diagnosis and resolution, reducing downtime and ensuring a reliable viewing experience.

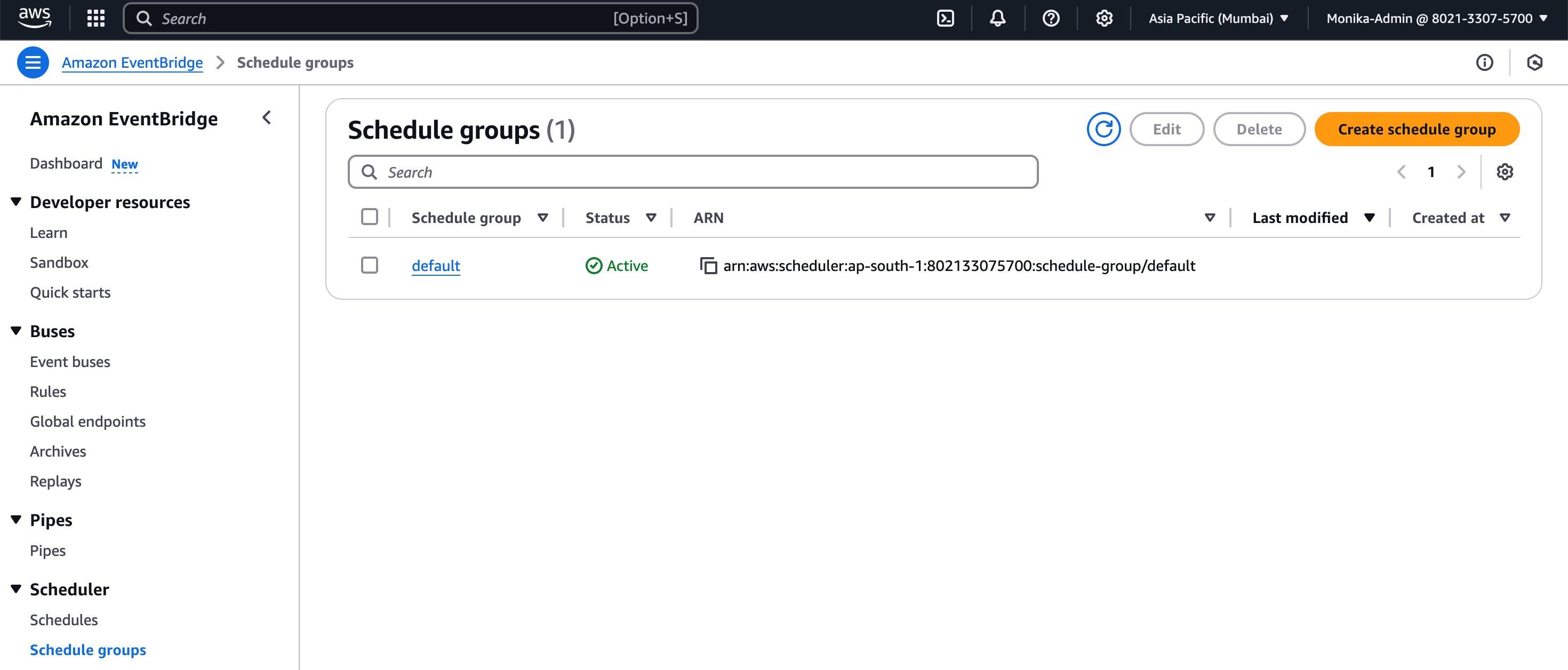


Fig 4.11: CloudWatch default is used for monitoring.

Step 12: Amazon EventBridge is a server-less service that uses events to connect application components together, making it easier for developers to build scalable eventdriven applications.To check the pipeline between channel and content it is used.

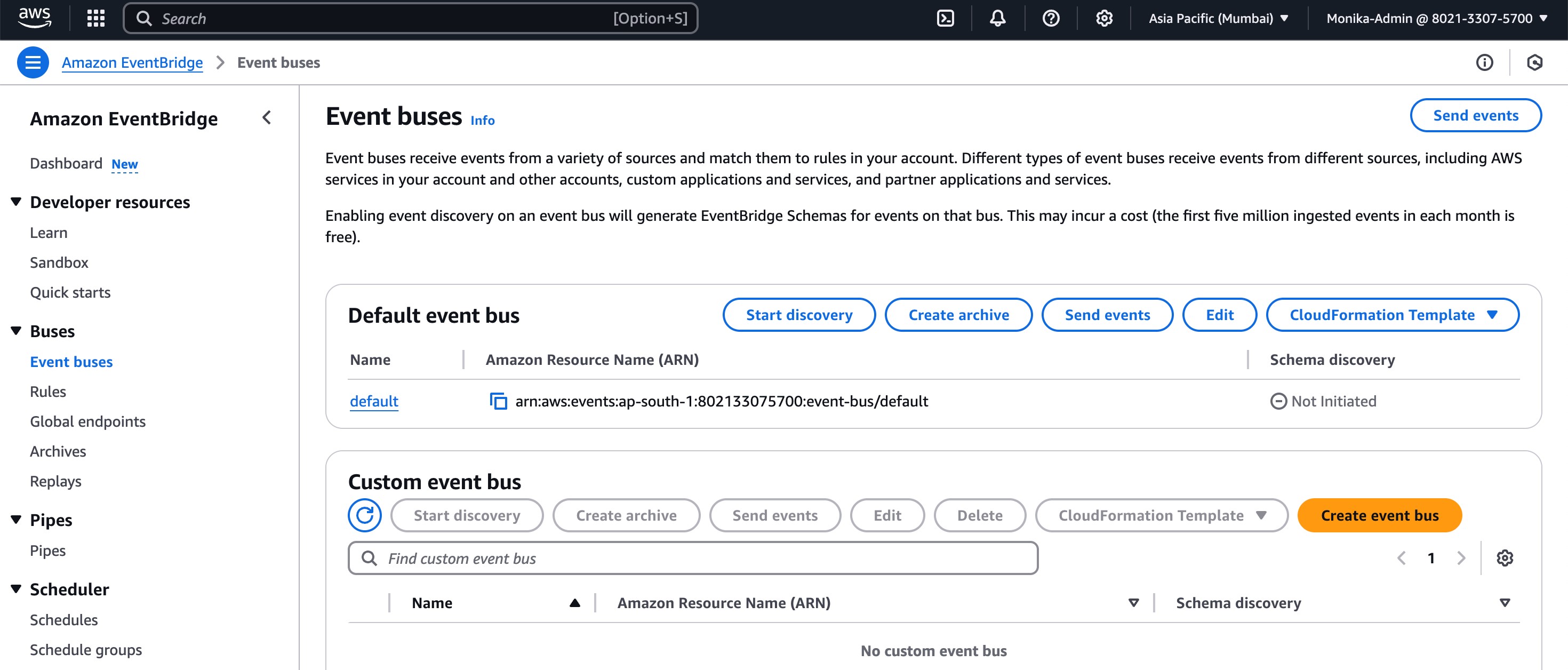


Fig4.12: AWS EventBridge default is used to monitoring.

## 5. LEARNING OUTCOMES

Through this project, I gained hands-on experience with AWS Media Services such as MediaLive, MediaPackage, and MediaConvert, which are essential for video processing and streaming. I also learned the importance of Amazon CloudFront in ensuring fast content delivery with minimal latency. Understanding how Amazon S3 functions as a scalable and reliable storage solution for media files was another key takeaway.

Additionally, working with IAM roles and permissions helped me grasp the significance of cloud security and access control in managing AWS resources effectively.

Moreover, I explored AWS CloudWatch for real-time monitoring, allowing me to track system performance and troubleshoot issues efficiently. The project also provided insights into workflow automation by integrating multiple AWS services for seamless media content delivery. Managing costs while utilizing AWS Free Tier services was another crucial aspect, reinforcing best practices for cloud resource optimization. Overall, this experience strengthened my ability to build, deploy, and manage a cloud-based digital signage system while improving scalability and performance efficiency.

Additionally, I learned how to optimize media streaming by leveraging AWS Elemental MediaConvert to transcode videos into different formats suitable for various devices. Understanding the role of MediaConvert in ensuring compatibility across different streaming platforms was a valuable aspect of the project. This enhanced my knowledge of adaptive bitrate streaming, which improves user experience by adjusting video quality based on network conditions.

Finally, this project reinforced my ability to work with cloud-based architectures and services, preparing me for real-world applications in media streaming and content delivery. It also strengthened my problem-solving skills, as I had to troubleshoot issues related to video playback, permissions, and distribution settings. By completing this project, I developed a deeper understanding of AWS services and their role in modern cloud computing solutions.

## 6. CONCLUSION

This project successfully demonstrated the potential of AWS cloud services in implementing an efficient and scalable digital signage system. By integrating MediaLive, MediaPackage, and CloudFront, I was able to create a seamless media streaming workflow. The ability to store and manage media assets using Amazon S3 further reinforced my understanding of cloud storage solutions. The experience of configuring IAM roles and security policies also highlighted the importance of access control and data security in cloud environments.

Furthermore, the implementation of AWS Elemental MediaConvert allowed me to optimize media content for diverse playback scenarios, ensuring compatibility with multiple devices and network conditions. The use of CloudWatch for monitoring system performance helped in identifying and troubleshooting potential issues efficiently. These insights are essential in real-world cloud-based applications, where performance, scalability, and security play a critical role.

The project also emphasized the significance of automation in cloud-based solutions. By integrating various AWS services, I streamlined content delivery without requiring extensive manual intervention. Understanding the principles of adaptive bitrate streaming and CDN integration through CloudFront further enriched my knowledge of optimizing digital media delivery across different geographies.

In conclusion, this project provided valuable practical exposure to AWS cloud services and their application in media streaming. It strengthened my problem-solving skills and deepened my technical expertise in cloud computing. Moving forward, I can leverage these learnings in future projects involving cloud-based content distribution and large-scale media processing.

## 7. REFERENCES

[1] Davies, Nigel, Sarah Clinch, and Florian Alt (2014). *Pervasive displays: understanding the future of digital signage.* Synthesis Lectures on Mobile and Pervasive Computing, 8(1): 11–28.

[2] Bauer, Christine, Paul Dohmen, and Christine Strauss (2011). *Interactive Digital Signage: an Innovative Service and its Future Strategies.* Proceedings of the 1st International Workshop on Frontiers in Service Transformations and Innovations (FSTI 2011), in conjunction with the 2nd International Conference on Emerging Intelligent Data and Web Technologies (EIDWT 2011), 137–142.

[3] Bauer, Christine, Paul Dohmen, and Christine Strauss (2012). *A Conceptual Framework for Backend Services of Contextual Digital Signage.* Journal of Service Science Research, 4(2): 271–297.

[4] Garaus, Marion and Udo Wagner (2018). *Let me entertain you – Reducing perceived waiting time through digital signage.* Proceedings of the 47th EMAC Annual Conference (EMAC 2018).

[5] Amazon Web Services (AWS). (2023). *AWS Elemental MediaLive Anywhere: Cloud-Controlled Live Video Encoding.*

[6] Dennis, Charles, J. Joško Brakus, Suraksha Gupta, and Eleftherios Alamanos (2014). *The Effect of Digital Signage on Shoppers' Behavior: The Role of the Evoked Experience.*