VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Belgavi-590 014

A MINI PROJECT REPORT
ON
“VIDEO STREAMING”

Submitted in partial fulfillment of the requirements for the **Mini Project (21CSMP67)** course of the 6th semester

Bachelor of Engineering
In
Computer Science & Engineering

Submitted By

Pratheek G Morabad (4AI21CS073)

Under the Guidance of:
Mrs. Mithuna B N., B.E., M.Tech.
Assistant Professor, Dept. Of CS&E,
AIT, Chikkamagaluru.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ADICHUNCHANAGIRI INSTITUTE OF TECHNOLOGY
CHIKKAMAGALURU -577102

(Affiliated to VTU, Belagavi and Approved by AICTE, New Delhi Chikkamagaluru- 577 102, Karnataka, India.)

ADICHUNCHANAGIRI INSTITUTE OF TECHNOLOGY

(Affiliated to VTU, Belagavi and Approved by AICTE, New Delhi) Chikkamagaluru- 577 102, Karnataka, India.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "VIDEO STREAMING" is a bonafide work carried out by PRATHEEK G MORABAD (4AI21CS073) in partial fulfillment of the requirements for Mini Project (21CSMP67) course of the 6th semester Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the academic year 2023-24. It is certified that all corrections and suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements in respect of Project Work prescribed for the said degree.

Signature of the Guide
Mrs. Mithuna B N.,B.E,M.Tech
Assistant Professor, Dept. of CS&E,
AIT, Chikkamagaluru.

Signature of the HOD
Dr. Pushpa Ravikumar.,B.E.,M.Tech.,Ph.D., MISTE
Professor and Head, Dept. Of CS&E,
AIT, Chikkamagaluru.

ACKNOWLEDGEMENTS

We express our humble Pranamas to his holiness **Parama Poojya Jagadguru Padmabushana Sri Sri Sri Dr. Balagangadharanatha Mahaswamiji** and **Parama Poojya Jagadguru Sri Sri Sri Dr. Nirmalanandanatha Mahaswamiji** and also to **Sri Sri Gunanatha Swamiji,** Sringeri Branch, Chikkamagaluru who have showered their blessings on us for framing our career successfully.

We are deeply indebted to our honorable Director **Dr. C K Subbaraya** for creating the right kind of care and ambience.

We are thankful to our beloved Principal **Dr. C T Jayadeva** for inspiring us to achieve greater endeavors in all aspects of learning.

We express our deepest gratitude to **Dr. Pushpa Ravikumar,** Professor& Head, Department of Computer Science & Engineering for her valuable guidance, suggestion and constant encouragement without which success of our project work would have been difficult.

I am thankful to our guide **Mrs. Mithuna B N,** Assistant Professor, Dept. of Computer Science & Engineering, AIT, Chikkamagaluru for his inspiration and lively correspondence for carrying our project work.

We would like to thank our beloved parents for their support, encouragement and blessings. And last but not least, we would like to express our heartfelt thanks to all teaching and nonteaching staff of Computer Science & Engineering Department and our friends who have rendered their help, motivation and support.

PRATHEEK G MORABAD	**(4AI21CS073)**
UDAY KEERTHI G R	**(4AI21CS111)**
SRUJAN A M	**(4AI21CS092)**
RAKSHITH C N	**(4AI21CS074)**

ABSTRACT

Efficient video streaming is essential for delivering high-quality digital content to users worldwide. Amazon Web Services (AWS) provides a powerful combination of S3 buckets and CloudFront to address the challenges associated with scalable and reliable video delivery. Amazon S3 (Simple Storage Service) offers a highly durable and scalable storage solution, allowing for the secure and cost-effective management of large volumes of video content. Its ability to automatically scale with demand ensures that videos are readily available, supporting varying levels of user traffic. Amazon CloudFront, a content delivery network (CDN), enhances video streaming performance by distributing content through a global network of edge locations.

Together, S3 and CloudFront offer a comprehensive solution for video streaming that combines scalability, performance, and cost-efficiency. S3’s pay-as-you-go storage model and CloudFront’s global edge network optimize data transfer speeds while controlling expenses. Furthermore, AWS's robust security features protect video content against unauthorized access and cyber threats. This integration not only enhances user satisfaction by ensuring reliable and fast video delivery but also supports the growing demand for online video content globally.

TABLE OF CONTENTS

Chapter	**Chapter Title**	**Page No.**
	Acknowledgment	i
	Abstract	ii
	Contents	iii
	List of Figures	iv
	List of Snapshots	v
Chapter 1	**Overview Of DevOps**	**1**
	1.1 Introduction to the DevOps	1
	1.2 DevOps Life Cycle	1
	1.3 Organization Of The Report	3
	1.4 Summary	4
Chapter 2	**About the DevOps**	**5**
	2.1 Overview of the DevOps	5
	2.2 How DevOps work	5
	2.3 Benefits of DevOps	5
	2.4 Functionality of DevOps	6
	2.5 Technologies and Skills in DevOps	7
	2.6 Github	7
	2.7 Summary	8
Chapter 3	**Reflection notes**	**9**
	3.1 Introduction	9
	3.2 S3 Bucket	10
	3.3 CloudFront	10
Chapter 4	**Task Performed**	**11**
	3.1 Aim	11
	3.2 Output	13
Chapter 5	**Results and discussions**	**14**
Chapter 6	**Conclusion**	**21**

LIST OF FIGURES

Figure No	Name of Figure	Page no
1.2.1	DEVOPS LIFE CYCLE	02

LIST OF SNAPSHOTS

Figure No	Name of Figure	Page no
5.1	CREATE BUCKET	14
5.2	NAME BUCKET	14
5.3	BLOCK PUBLIC	15
5.4	SERVER ENCRYPT	15
5.5	RETRIEVAL PAGE	16
5.6	UPLOAD PAGE	16
5.7	ADD VIDEO/FILES	17
5.8	CREATE DISTRIBUTION	17
5.9	LINKING DOMAIN NAME	18
5.10	PUBLIC ACCESS CONTROL	18
5.11	COPYING DOMAIN DISTRIBUTION	19
5.12	COPYING KEY S3 BUCKET	19
5.13	VIDEO PLAYED	20

CHAPTER 1

OVERVIEW OF DEVOPS

1.1 INTRODUCTION TO THE DEVOPS

In DevOps is a set of practices that combines software development (Dev) and IT operations (Ops) to improve the speed, quality, and reliability of software releases and deployments. It aims to bridge the gap between development and operations teams, promoting collaboration, automation, and continuous improvement.

Implementing DevOps is not just about tools; it is also about the way people work and the processes they use. DevOps breaks the historical silo between the engineering teams creating an application or service and those responsible for running that service in production. Processes and work align to the entire lifecycle of the products and services and all that is needed for their delivery and operation.

Ideally, one team manages all aspects of the service, including security and testing functions. In larger organizations there may still be some functional specialization, but it remains critical that the process and communication are focused on the end-to-end delivery of the entire service. This product-centric view can be based on something as simple as a microservice or a more complex set of deliverables that comprise a release (the circumstances and the end customer often make that determination). Over time the goal is to continue to make smaller changes and to iterate more rapidly.

The new processes and teams leverage as much automation as possible alongside technologies that facilitate the end-to-end connection of the product lifecycle, including the all-important feedback loop frothe customer to the team.

1.2 DEVOPS LIFE CYCLE

The DevOps lifecycle represents a continuous and collaborative approach to software development and operations, with the aim of delivering high-quality software efficiently. This lifecycle is often depicted using either a circle or an infinity loop, with steps flowing into one another without any clear beginning or end. This symbolizes the continuous nature of DevOps and the need for constant collaboration and iterative improvement throughout the entire process.

The stages of the DevOps lifecycle can be categorized in the following way:

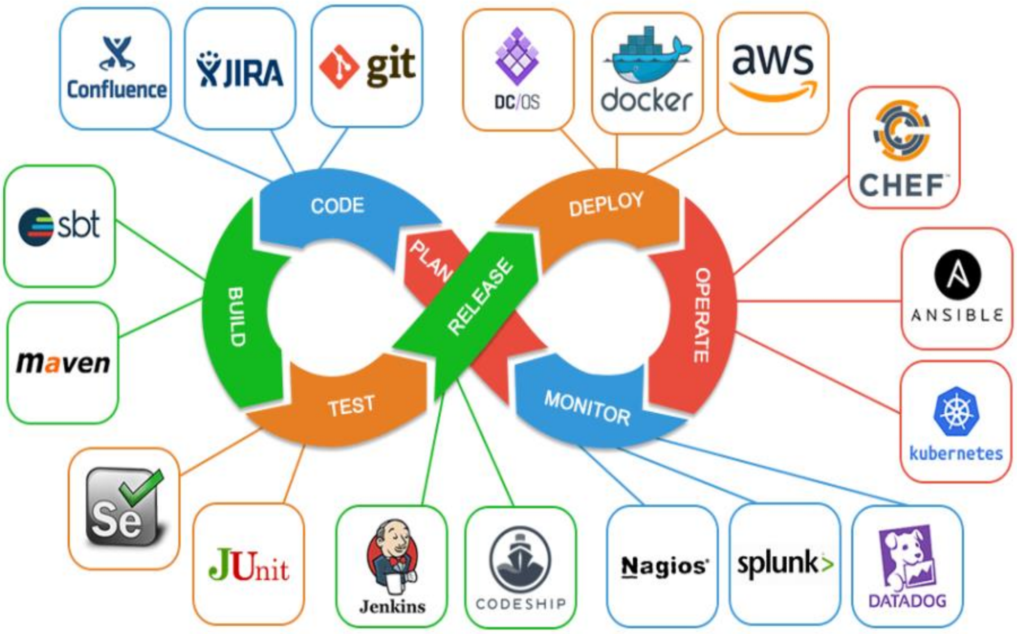


Fig 1.2.1: DevOps lifecycle

- **Discovery**

In the initial phase, DevOps teams engage in brainstorming and concept exploration. This stage is crucial for aligning development efforts with strategic goals and ensuring that proposed ideas will have a positive impact on users. By prioritizing ideas that deliver real value, organizations set a clear direction for subsequent phases of the DevOps lifecycle.

- **Planning**

The next phase encourages the adoption of agile practices, breaking work into smaller, manageable pieces to deliver incremental value. Agile planning empowers teams to work efficiently and deliver value cumulatively while adapting to changing requirements.

- **Building**

During the building phase, DevOps teams use version control systems to manage changes—compiling code, performing unit tests, packaging the application, and sending code changes to the code source. Effective code management is essential for maintaining code quality and ensuring that development progresses smoothly.

- **Testing**

Continuous integration is at the core of the testing phase. Multiple developers can contribute to a shared code repository, and automated tests are executed upon code changes to validate correctness before integration. CI serves as a quality assurance mechanism, instilling confidence in the reliability of the code.

- **Deployment**

With testing complete, features are ready to be deployed into production. This can be automated through continuous deployment so that manual intervention is not required, and changes are instead automatically and immediately delivered to the end-user. Alternatively, continuous delivery allows for more oversight, preparing changes so that they are ready for deployment at the push of a button.

- **Operation**

The operation phase encompasses the comprehensive management of IT services —from the design and implementation of infrastructure to its configuration, deployment, and ongoing maintenance—all aimed at delivering reliable and efficient services to end-users. This phase focuses on ensuring the smooth operation of the IT environment that underpins an organization's services, maintaining uptime, performance, and security while responding to any operational challenges that may arise in production environments.

- **Observation**

As any changes are deployed, teams monitor metrics and logs to swiftly identify and address issues affecting product uptime, speed, and functionality. Automatic notifications keep teams informed of changes and potential problems. Effective monitoring and observation practices are critical for maintaining service quality and addressing issues promptly.

- **Ongoing feedback**

DevOps teams continuously evaluate each release, generating reports to inform and improve future releases. Gathering feedback, including customer input, serves as a cornerstone for enhancing development processes. Ongoing feedback ensures that DevOps practices remain responsive to changing needs and that each release represents an improvement over the previous one.

1.3 ORGANIZATION OF THE REPORT

Project report is organized as follows:

Chapter 2: About the DevOps, it presents brief overview of the DevOps, how DevOps works and benefits of DevOps, functionality of DevOps, the different DevOps Practices, andintroduction about Git Hub and Microsoft Azure organizational structure, technologies and skills supported by DevOps.

Chapter 3: Task Performed, it includes motivation, problem statement, objectives, and scope of the project.

Chapter 4: Reflection notes, it includes introduction of S3 buckets, working and CloudFront introduction overview.

Chapter 5: Results and discussions, this section presents the snapshots of the S3 buckets created and the distributions added.

Chapter 6: Conclusion and future enhancements, this section presents the conclusion and future enhancements.

1.4 SUMMARY

This chapter describes about the DevOps overview. Section 1.1 presents the brief overview introduction about DevOps. Section 1.2 describes lifecycle of DevOps. 1.3 describes organization of the report. DevOps is a collaborative approach that integrates development and IT operations teams to streamline and automate the software development lifecycle. By fostering communication between these teams and leveraging practices like continuous integration and continuous deployment (CI/CD), infrastructure as code (IaC), and automated testing, DevOps aims to accelerate software delivery, enhance quality, and improve operational efficiency. The ultimate goal is to create a more agile and responsive development process that quickly adapts to changing requirements and user needs.

CHAPTER 2

ABOUT THE DEVOPS

2.1 OVERVIEW OF THE DEVOPS

DevOps is the combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes. This speed enables organizations to better serve their customers and compete more effectively in the market.

2.2 HOW DEVOPS WORK

Under a DevOps model, development and operations teams are no longer "siloed." Sometimes, these two teams are merged into a single team where the engineers work across the entire application lifecycle, from development and test to deployment to operations, and develop a range of skills not limited to a single function. In some DevOps models, quality assurance and security teams may also become more tightly integrated with development and operations and throughout the application lifecycle.

When security is the focus of everyone on a DevOps team, this is sometimes referred to as DevSecOps. These teams use practices to automate processes that historically have been manual and slow. They use a technology stack and tooling which help them operate and evolve applications quickly and reliably. These tools also help engineers independently accomplish tasks (for example, deploying code or provisioning infrastructure) that normally would have required help from other teams, and this further increases a team's velocity.

2.3 BENEFITS OF DEVOPS

1. **Speed:** Move at high velocity so you can innovate for customers faster, adapt to changing markets better, and grow more efficient at driving business results. The DevOps model enables your developers and operations teams to achieve these results. For example, micro services and continuous delivery let teams take ownership of services and then release updatesto them quicker.

2. **Rapid Delivery:** Increase the frequency and pace of releases so you can innovate and improve your productfaster. The quicker you can release new features and fix bugs, the faster you can respond to your

customers' needs and build competitive advantage. Continuous integration and continuous delivery are practices that automate the software released deploy.

3. Reliability: Ensure the quality of application updates and infrastructure changes so you can reliably deliver at a more rapid pace while maintaining a positive experience for end users. Use practices like continuous integration and continuous delivery to test that each change is functional and safe. Monitoring and logging practices help you stay informed of performancein real-time.

4. Scale: Operate and manage your infrastructure and development processes at scale. Automation and consistency help you manage complex or changing systems efficiently and with reduced risk. For example, infrastructure as code helps you manage your development, testing, and production environments in a repeatable and more efficient manner.

5. Improved Collaboration: Build more effective teams under a DevOps cultural model, which emphasizes values suchas ownership and accountability. Developers and operations teams collaborate closely, share many responsibilities, and combine their workflows. This reduces inefficiencies and saves time(e.g. reduced handover periods between developers and operations, writing code that take into account the environment in which it is run).

6. Security: Move quickly while retaining control and preserving compliance. You can adopt a DevOps model without sacrificing security by using automated compliance policies, fine- grained controls, and configuration management techniques. For example, using infrastructureas code and policy as code, you can define and then track compliance at scale.

2.4 FUNCTIONALITY OF DEVOPS

- **Collaboration:** DevOps emphasizes the need for collaboration between software development and IT operations teams. This collaboration ensures that all stakeholders are aligned on the goals of the project, and that communication channels are open throughout the development process.
- **Automation:** DevOps seeks to automate as much of the software development lifecycle as possible, from testing to deployment to monitoring. This automation ensures that processes are repeatable and consistent, reducing the risk of errors and enabling faster delivery of software applications.

- **Continuous delivery:** DevOps prioritizes the continuous delivery of software and applications, with frequent releases that provide value to end-users. This approach ensures that feedback is received early in the development process, enabling teams to make adjustments and improvements as needed.
- **Monitoring and feedback:** DevOps teams use a variety of tools and processes to monitor software applications in production, ensuring that any issues are quickly identified and resolved. This feedback loop enables continuous improvement and ensures that end-users receive a high-quality experience.

2.5 TECHNOLOGIES AND SKILLS IN DEVOPS

- Version Control Systems (VCS): Git, SVN, Mercurial
- Continuous Integration (CI) tools: Jenkins, Travis CI, CircleCI, GitLab CI
- Configuration Management (CM) tools: Ansible, Puppet, Chef, SaltStack
- Containerization technologies: Docker, Kubernetes
- Infrastructure as Code (IaC) tools: Terraform, CloudFormation, Ansible, Chef
- Monitoring and Logging tools: Nagios, Zabbix, ELK Stack (Elasticsearch,Logstash, Kibana)
- Cloud computing platforms: Amazon Web Services (AW Azure,Programming languages: Python, Ruby, Java, Go
- Google Cloud Platform (GCP)
- Scripting languages: Bash, PowerShell, Perl
- Agile development methodologies: Scrum, Kanban, Lean
- Soft skills: communication, collaboration, problem-solving, leadership, adaptability.

2.6 GITHUB

GitHub is a web-based platform that provides developers with a powerful suite of tools andservices for version control and collaborative software development. Launched in 2008, GitHub quickly became the go-to platform for millions of developers worldwide, offering an intuitive userinterface, robust features, and seamless integration with other development tools.

One of the key features of GitHub is its version control system, which enables developers to track changes to their code over time, collaborate with other developers, and manage complex software projects. This allows developers to work more efficiently and effectively, reducing the risk of errors and conflicts in the code. With GitHub's version control system, developers can createbranches of code, make changes, and merge those changes back into the main codebase, all whilekeeping a detailed record of every change made.

Another important feature of GitHub is its support for open-source software development. GitHub provides a centralized platform where developers can share their code with others, collaborate on projects, and contribute to open- source software projects. This has helped to fuela vibrant and active community of developers who work together to create new software solutions,fix bugs, and improve existing code. This allows developers to work more efficiently andeffectively, reducing the risk of errors and conflicts in the code.

Finally, GitHub provides a range of tools and services that streamline the software development process, and project management. With GitHub, developers can manage their code repositories, collaborate with other developers in real-time. This makes it easier for teams to work and deliver high quality software faster and with fewer errors**.**

Overall, GitHub has revolutionized the way developers work by providing a powerful suite of tools and services that enable version control, collaboration, and project management. With its intuitive user interface, robust features, and support for open-source development, GitHub has become an essential tool for developers around the world.

2.7 SUMMARY

This chapter describes about the DevOps. Section 2.1 presents the overview of the DevOps. Section 2.2 describes how DevOps work. Section 2.3 briefs out the benefits of the DevOps. Section 2.4 list outs the functionalities of DevOps. Section 2.5 points out thetechnologies and skills in DevOps. Section 2.6 says about github. GitHub plays a crucial role in DevOps by providing a platform for version control, collaboration, and automation. It allows development teams to manage code repositories, track changes, and collaborate on code through pull requests and code reviews. GitHub Actions, its built-in CI/CD service, enables teams to automate workflows for testing, building, and deploying code. By integrating with other DevOps tools and services, GitHub supports seamless automation of development processes, continuous integration, and continuous deployment, ultimately enhancing the efficiency and speed of delivering high-quality software.

CHAPTER 3

REFLECTION NOTES

3.1 INTRODUCTION

To stream video, use AWS S3 for scalable storage and AWS Elemental MediaStore for low-latency delivery. Upload your videos to S3, configure MediaStore for real-time streaming, and use a media player to access and playback content efficiently. This setup ensures reliable, high-quality video streaming.

Key Aspects of s3 buckets

When using Amazon S3 buckets for video streaming, several key aspects are crucial to consider:

1. Scalability: S3 provides virtually unlimited storage capacity, allowing you to handle large volumes of video content and scale easily as your needs grow. This is essential for managing and delivering high-quality video streams to a large audience.

2. Durability and Availability: 3 offers 99.999% durability and high availability, ensuring that your video files are securely stored and reliably accessible. This reduces the risk of data loss and ensures consistent availability of content.

3. Cost Efficiency: With S3's pay-as-you-go pricing model, you only pay for the storage and data transfer you use. This flexibility allows you to manage costs effectively while scaling your storage needs for video streaming.

4. **Access Control and Security**: S3 provides robust security features, including access control policies, encryption options, and versioning. These features help protect your video content from unauthorized access and ensure data privacy and integrity.

5. **Integration with Other AWS Services**: S3 integrates seamlessly with services like AWS Elemental Media Convert for video transcoding and AWS CloudFront for content delivery. This integration streamlines the video processing and distribution workflow, enhancing the overall streaming experience.

6. **Player Compatibility**: Ensuring compatibility with various devices and browsers.

7. Scalability: Handling varying numbers of viewers efficiently.

8. Security: Protecting content from unauthorized access and piracy.

9. Logging and Monitoring: Provides logging of access requests and integration with AWS CloudWatch for monitoring bucket activities and performance.

3.2 S3 BUCKETS

An Amazon S3 bucket is a scalable storage container within AWS S3 where you can store and manage any amount of data, such as files, documents, and backups. Each bucket is globally unique and can be configured with access controls, data lifecycle policies, and storage classes to meet various performance and compliance needs. Buckets are region-specific, allowing you to optimize for latency and regulatory requirements, and they support integration with other AWS services for data processing, delivery, and security.

Working of s3 buckets

- Creation: You create a bucket in a specified AWS region.Each bucket is uniquely named and can store an unlimited number of objects.
- Uploading Objects: You upload files (objects) to the bucket. Each object consists of the file data, metadata, and a unique key (name) within the bucket.
- Access Control: You set permissions on the bucket and its objects using policies, ACLs, or IAM roles, determining who can read, write, or manage the data.
- Storage Classes: Objects can be stored in various classes (e.g., Standard, Glacier) based on access needs and cost considerations.
- Data Management: Buckets support features like versioning (keeping multiple versions of objects), lifecycle policies (automatically transitioning or deleting objects), and cross-region replication (copying objects to buckets in other regions).

3.3 CLOUDFRONT

Amazon CloudFront is a Content Delivery Network (CDN) service that accelerates the delivery of websites, applications, and content by distributing it across a global network of edge locations. By caching content closer to users, CloudFront reduces latency and speeds up access to static and dynamic data. It integrates seamlessly with other AWS services, offers robust security features such as SSL/TLS encryption and AWS WAF, and provides detailed analytics for performance monitoring.

CHAPTER 4

TASK PERFORMED

4.1 AIM: video streaming using s3 buckets and Cloud Front

To set up video streaming using Amazon S3 and CloudFront, follow these steps:

1. Prepare Your S3 Bucket

1. Create an S3 Bucket:

   - Go to the AWS Management Console.
   - Navigate to S3 and click “Create bucket.”
   -Enter a unique name and choose the region.
   - Click “Create.”

2. Upload Your Videos:

   - Open your bucket.
   - Click “Upload” and select your video files.
   - Click “Upload” to start the upload process.

3. Set Permissions:

   - Go to the “Permissions” tab in your bucket settings.
   - Set the bucket policy to allow access to your videos.

   Example policy:

json

```
        {
                "Version": "2012-10-17",
                "Statement": [
        {
        "Effect": "Allow",
        "Principal": "*",
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::your-bucket-name/*"
        }
        ]}- Adjust the policy according to your access requirements.
```

2. Set up CloudFront

1. Create a CloudFront Distribution:
   - Go to the AWS Management Console.
   - Navigate to CloudFront and click "Create Distribution."
   - Choose "Web" for the delivery method.
2. Configure the Distribution:
   - Origin Settings:
     - For the "Origin Domain Name," select your S3 bucket from the dropdown.
     - Set "Origin Path" if you want to specify a subfolder in your bucket.
     - Choose "Match Viewer" for the "Viewer Protocol Policy" or set it to "Redirect HTTP to HTTPS" for secure delivery.
   - Default Cache Behavior Settings:
     - Set "Viewer Protocol Policy" to "Redirect HTTP to HTTPS" if you want to enforce secure connections.
     - Configure caching based on your requirements.
   - Distribution Settings:
     - Add a suitable "Alternate Domain Name (CNAME)" if you want to use a custom domain.
     - Select an SSL certificate if using HTTPS.
     - Adjust other settings such as logging and default root object.
3. Create and Deploy the Distribution:
   - Review your settings and click "Create Distribution."
   - It may take a few minutes for the distribution to be deployed.

3. Configure Your Player

1. Get the CloudFront URL:
   - Once the distribution is deployed, go to the CloudFront console and find your distribution.
   - Copy the "Domain Name" of your CloudFront distribution.
2. Update Your Video URLs:
   - Use the CloudFront domain to access your videos. Replace the S3 URL in your video player with the CloudFront URL. For example:

     https://your-cloudfront-domain/path-to-your-video.mp4

3. Embed the Video Player:

   - Integrate the video URL into your web application or website using an HTML5 video player or any media player that supports streaming.

4. Testing and Monitoring

1. Test Streaming:

   - Play your video using the CloudFront URL to ensure it streams correctly.

2. Monitor Performance:

   - Use CloudFront’s monitoring tools in the AWS Management Console to track metrics and performance.
   - Check S3 and CloudFront logs for access patterns and issues.

By following these steps, you will set up a scalable, efficient video streaming solution leveraging S3’s storage and CloudFront’s content delivery capabilities.

4.2 OUTPUT

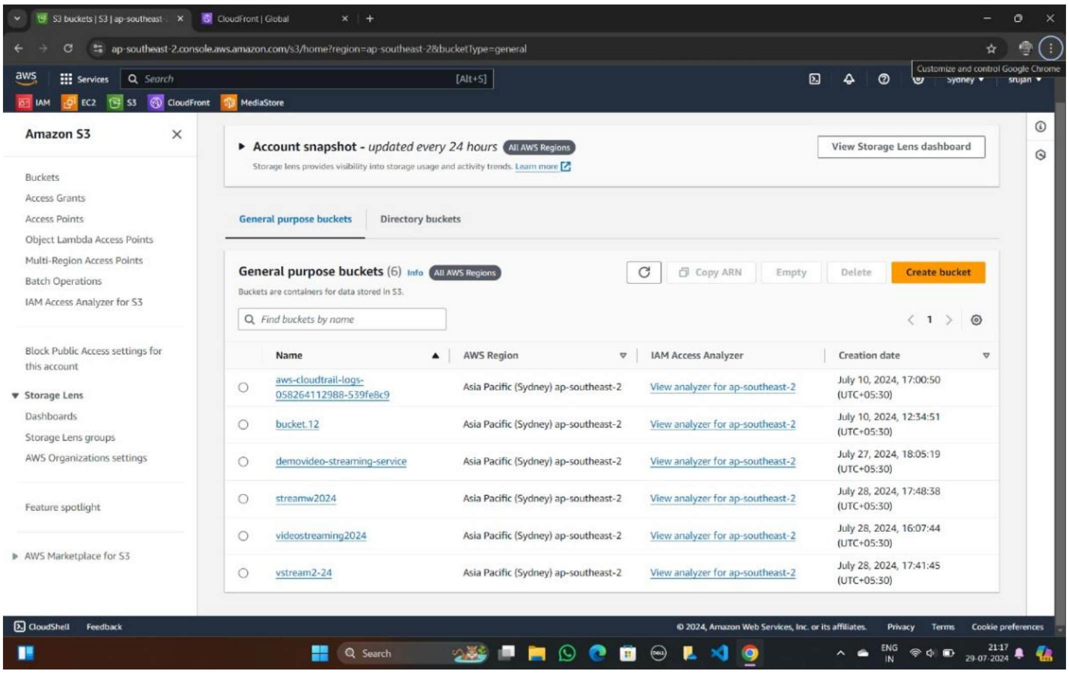
Using Amazon S3 and CloudFront for video streaming delivers a robust and efficient experience. Videos stored in S3 buckets benefit from CloudFront's global network of edge locations, which cache and deliver content closer to viewers, reducing latency and improving load times. This setup ensures high performance and scalability, handling varying numbers of concurrent viewers and traffic spikes effectively. The combination also supports adaptive bitrate streaming, adjusting video quality based on the viewer’s connection speed to provide a seamless viewing experience.

Additionally, this solution offers enhanced security through SSL/TLS encryption and access controls, safeguarding content against unauthorized access. CloudFront's detailed analytics and monitoring tools provide insights into viewer behavior and performance, allowing for continuous optimization. The result is a high-quality, reliable, and secure streaming experience, with content delivered quickly and efficiently to a global audience.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 S3 BUCKET CREATION



Account snapshot - *updated every 24 hours* All AWS Regions

Storage lens provides visibility into storage usage and activity trends. Learn more

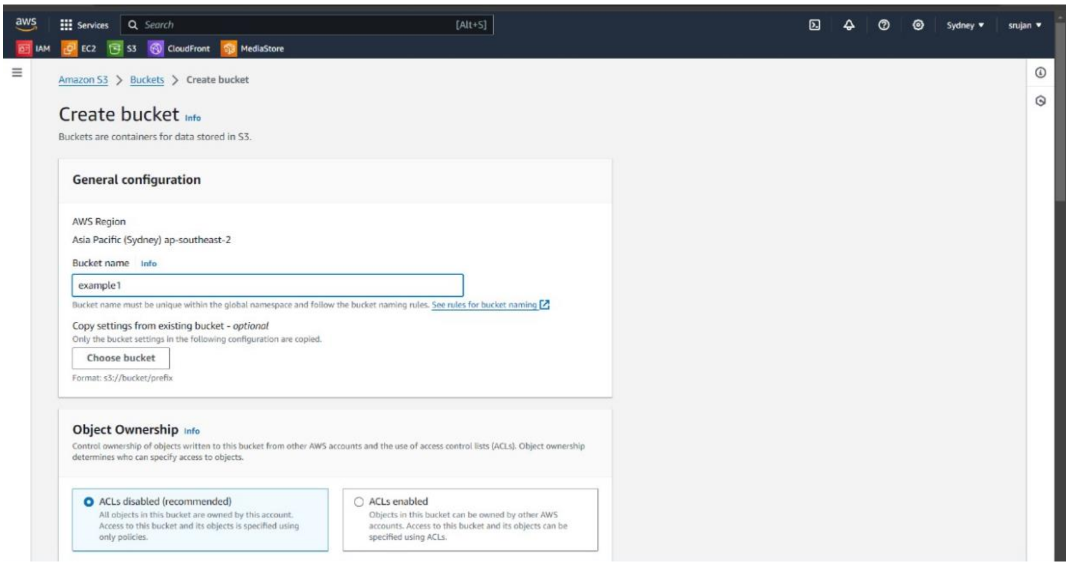
General purpose buckets (6) Info All AWS Regions

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
aws-cloudtrail-logs-058264112988-539fe8c9	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 10, 2024, 17:00:50 (UTC+05:30)
bucket.12	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 10, 2024, 12:34:51 (UTC+05:30)
demovideo-streaming-service	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 27, 2024, 18:05:19 (UTC+05:30)
streamw2024	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 28, 2024, 17:48:38 (UTC+05:30)
videostreaming2024	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 28, 2024, 16:07:44 (UTC+05:30)
vstream2-24	Asia Pacific (Sydney) ap-southeast-2	View analyzer for ap-southeast-2	July 28, 2024, 17:41:45 (UTC+05:30)

From all the services available, we select the Amazon S3 service. In the sub sections we select the buckets section and select create bucket.

5.2 S3 BUCKET NAMING



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region: Asia Pacific (Sydney) ap-southeast-2

Bucket name Info: example1

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

Object Ownership Info

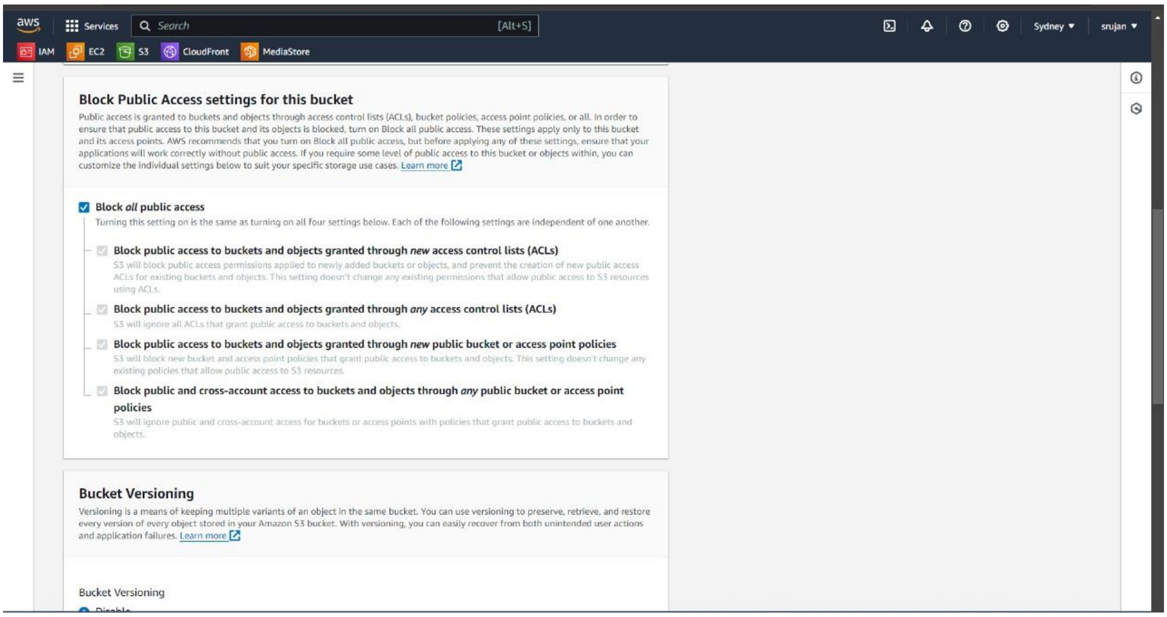
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● 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 ACLs.

To create a bucket, we need to provide the general configuration with AWS region and object ownership, the bucket name is given.

5.3 BLOCK PUBLIC ACCESS



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.

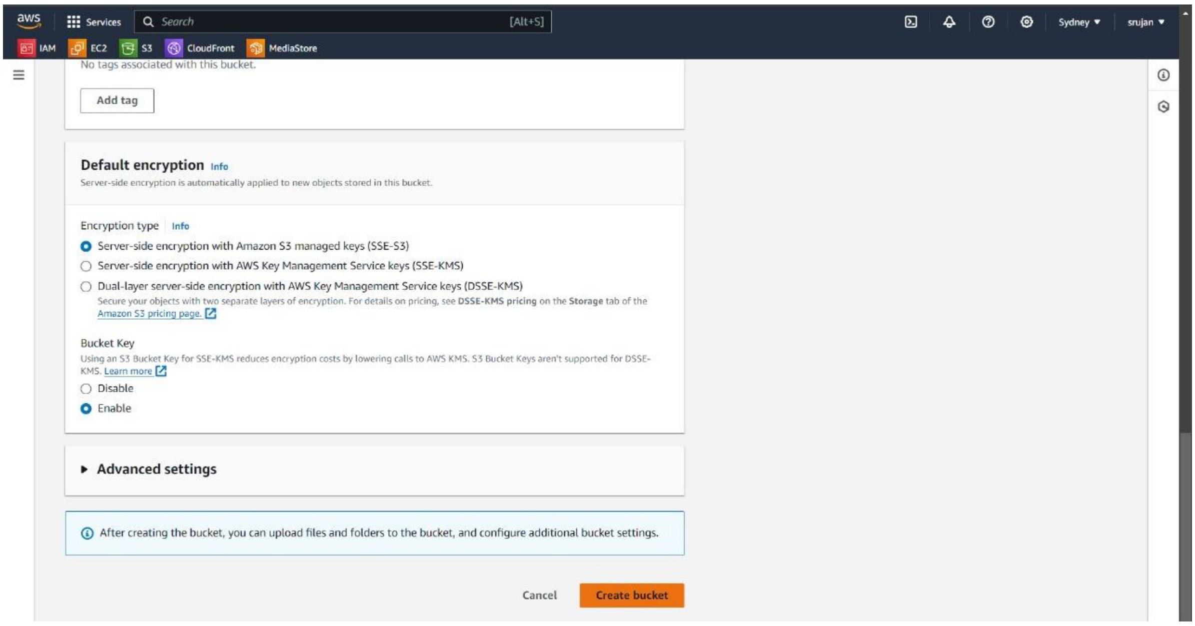
Bucket Versioning

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

Providing the access for the users to the bucket is specified. It allows public access settings for the present bucket.

5.4 SET SERVER ENCRYPTION



No tags associated with this bucket.

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type Info

● Server-side encryption with Amazon S3 managed keys (SSE-S3)
○ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
○ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

Bucket Key

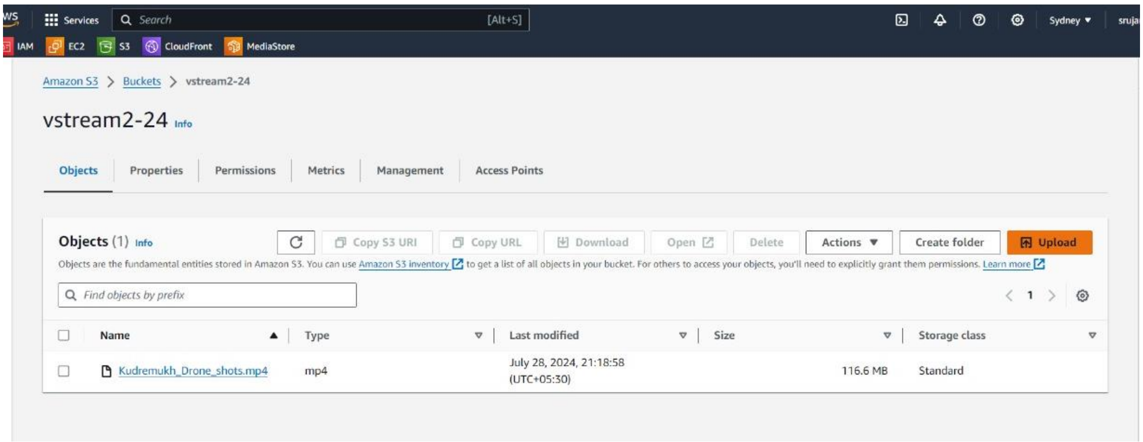
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

○ Disable
● Enable

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Server side encryption is automatically applied to new object stored in this bucket and bucket key for this is enabled.

5.5 RETRIEVAL PAGE



Amazon S3 > Buckets > vstream2-24

vstream2-24 Info

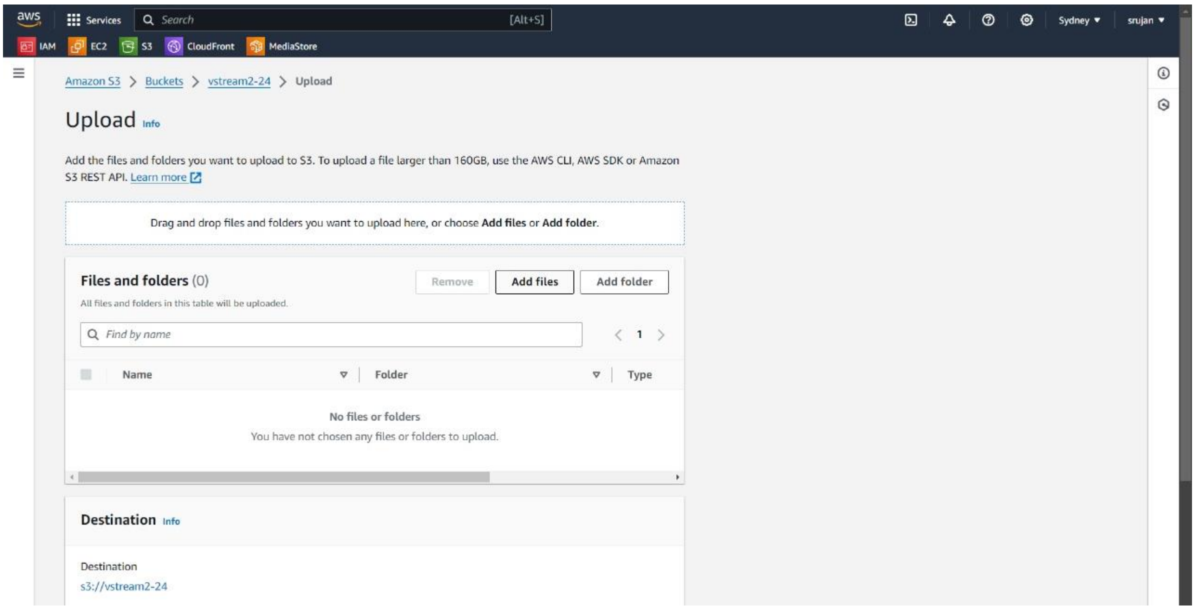
Objects (1) Info

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

Name	Type	Last modified	Size	Storage class
Kudremukh_Drone_shots.mp4	mp4	July 28, 2024, 21:18:58 (UTC+05:30)	116.6 MB	Standard

To view the uploaded objects to this bucket with respect to its properties, permissions, metrics, management and access points

5.6 UPLOAD PAGE



Amazon S3 > Buckets > vstream2-24 > Upload

Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK 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 (0)

All files and folders in this table will be uploaded.

Name	Folder	Type

No files or folders

You have not chosen any files or folders to upload.

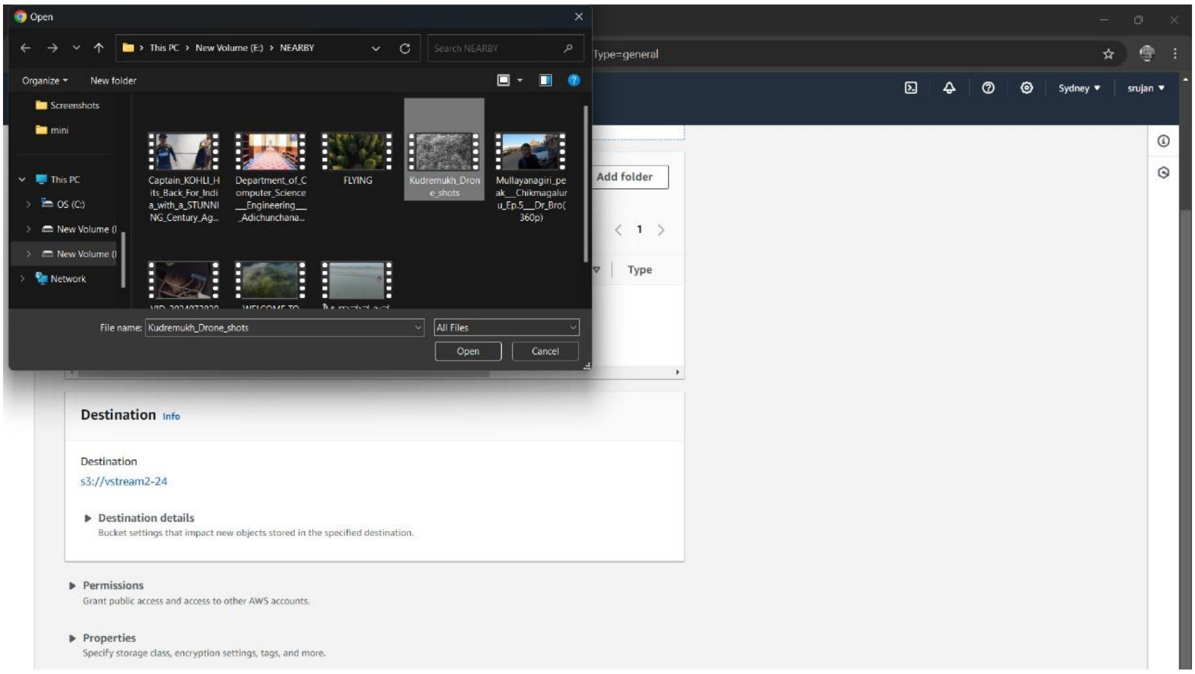
Destination Info

Destination

s3://vstream2-24

We can add files and folders which we want to upload to S3 bucket. The uploaded file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API

5.7 PROCESS OF ADDING FILES



Open
This PC > New Volume (E:) > NEARBY
Search NEARBY
Organize
New folder
Screenshots
mini
This PC
OS (C:)
New Volume (I
New Volume (I
Network
Captain_KOHLI_Hits_Back_For_India_with_a_STUNNING_Century_Ag...
Department_of_Computer_Science__Engineering___Adichunchana...
FLYING
Kudremukh_Drone_shots
Mullayanagiri_peak__Chikmagaluru_Ep.5__Dr_Bro(360p)
All Files
Open
Cancel
Type=general
Sydney
srujan
Add folder
Type

File name: Kudremukh_Drone_shots

Destination Info

Destination
s3://vstream2-24

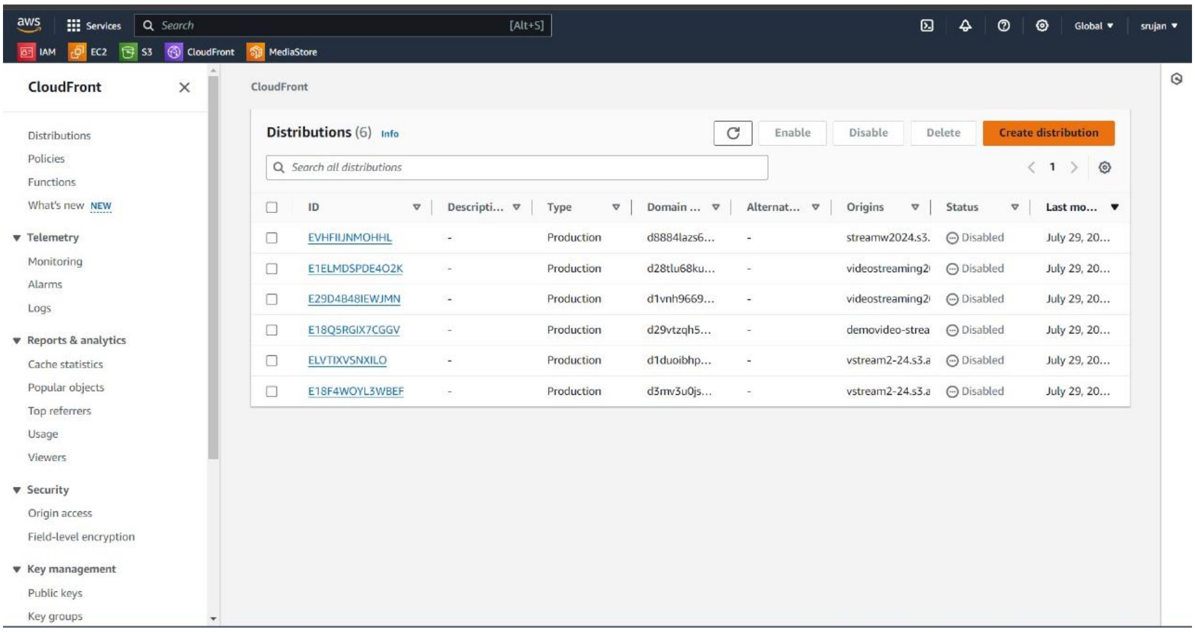
▸ Destination details
Bucket settings that impact new objects stored in the specified destination.

▸ Permissions
Grant public access and access to other AWS accounts.

▸ Properties
Specify storage class, encryption settings, tags, and more.

The files that are required to be uploaded to this bucket are browsed from the resources available in the system.

5.8 CREATING DISTRIBUTION

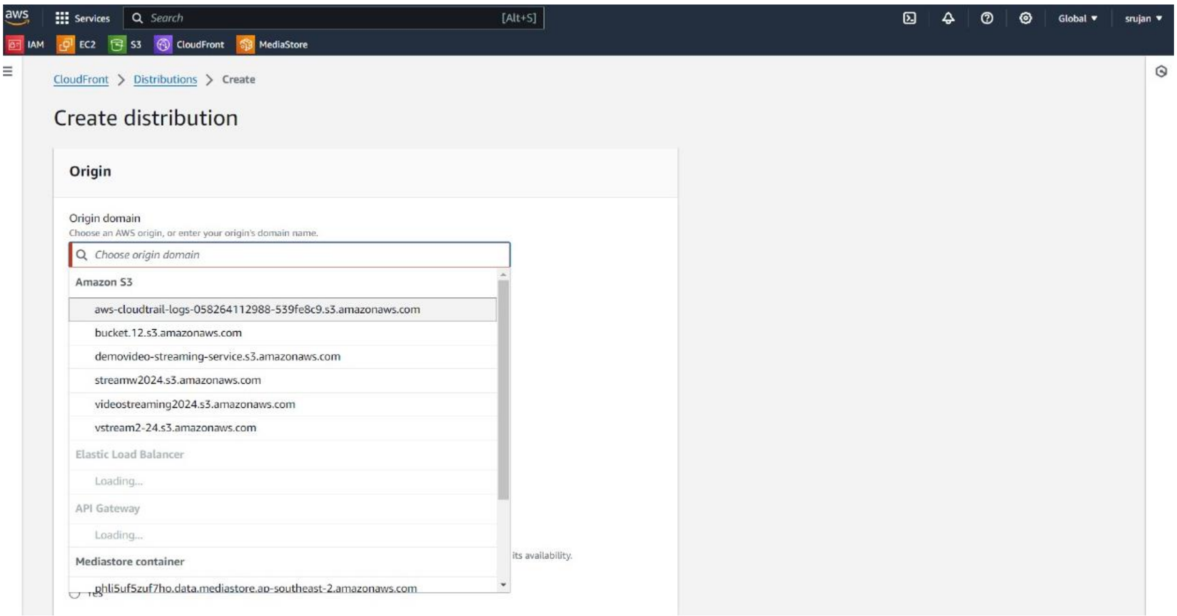


Distributions (6) Info

☐	ID	Descripti...	Type	Domain ...	Alternat...	Origins	Status	Last mo...
☐	EVHFIJNMOHHL	-	Production	d8884lazs6...	-	streamw2024.s3.	Disabled	July 29, 20...
☐	E1ELMDSPDE4O2K	-	Production	d28tlu68ku...	-	videostreaming2	Disabled	July 29, 20...
☐	E29D4B48IEWJMN	-	Production	d1vnh9669...	-	videostreaming2	Disabled	July 29, 20...
☐	E18Q5RGIX7CGGV	-	Production	d29vtzqh5...	-	demovideo-strea	Disabled	July 29, 20...
☐	ELVTIXVSNXILO	-	Production	d1duoibhp...	-	vstream2-24.s3.a	Disabled	July 29, 20...
☐	E18F4WOYL3WBEF	-	Production	d3mv3u0js...	-	vstream2-24.s3.a	Disabled	July 29, 20...

Created distributions and the available or already created distributions are displayed in this page.

5.9 LINKING DOMAIN TO S3 BUCKET



Create distribution

Origin

Origin domain

Choose an AWS origin, or enter your origin's domain name.

Choose origin domain

Amazon S3

- aws-cloudtrail-logs-058264112988-539fe8c9.s3.amazonaws.com
- bucket.12.s3.amazonaws.com
- demovideo-streaming-service.s3.amazonaws.com
- streamw2024.s3.amazonaws.com
- videostreaming2024.s3.amazonaws.com
- vstream2-24.s3.amazonaws.com

Elastic Load Balancer

Loading...

API Gateway

Loading...

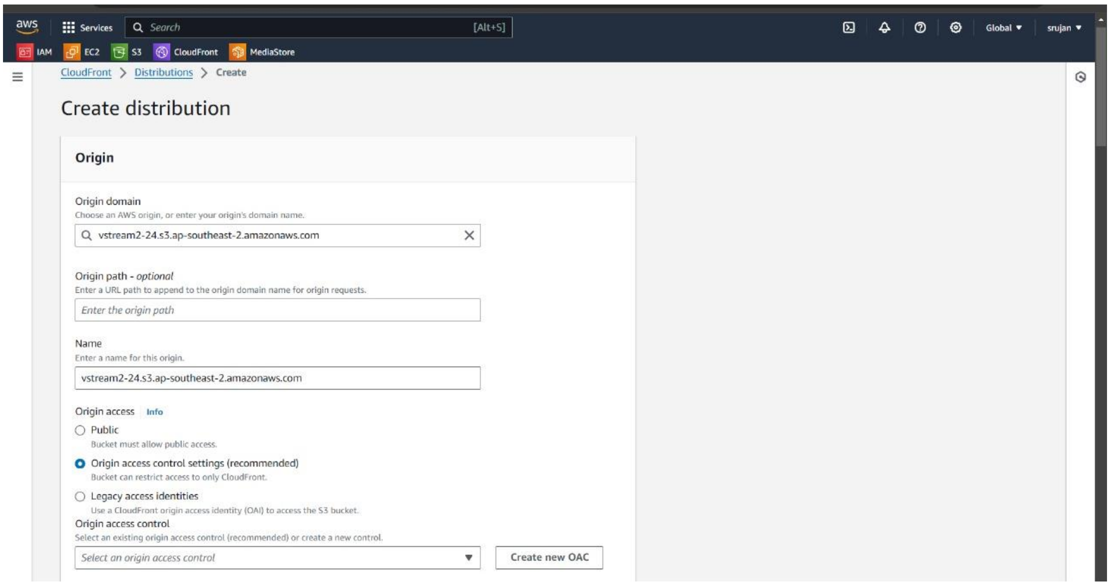
Mediastore container

ghli5uf5zuf7ho.data.mediastore.ap-southeast-2.amazonaws.com

its availability.

Here the list of created S3 buckets are displayed by which we can select the required bucket by its name and can be linked to the original domain.

5.10 PUBLIC ACCESS WITH CONTROL



Create distribution

Origin

Origin domain

Choose an AWS origin, or enter your origin's domain name.

vstream2-24.s3.ap-southeast-2.amazonaws.com

Origin path - *optional*

Enter a URL path to append to the origin domain name for origin requests.

Enter the origin path

Name

Enter a name for this origin.

vstream2-24.s3.ap-southeast-2.amazonaws.com

Origin access Info

○ Public

Bucket must allow public access.

● Origin access control settings (recommended)

Bucket can restrict access to only CloudFront.

○ Legacy access identities

Use a CloudFront origin access identity (OAI) to access the S3 bucket.

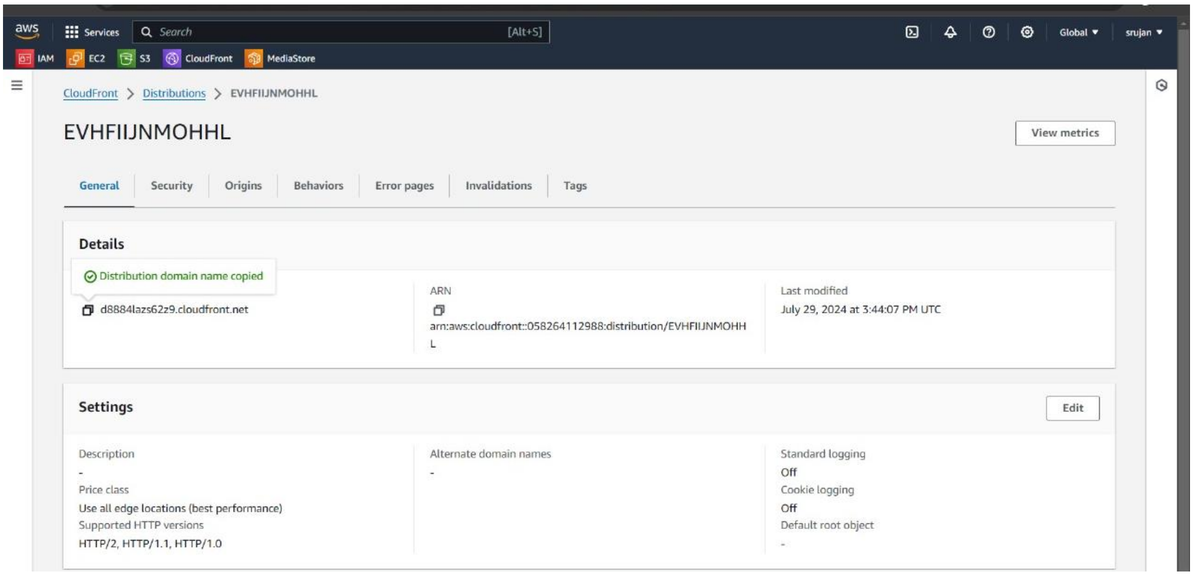
Origin access control

Select an existing origin access control (recommended) or create a new control.

Select an origin access control

Here we can restrict the access to only CloudFront else we can select an existing original access control or create a new control.

5.11 COPYING DOMAIN FROM DISTRIBUTION



CloudFront > Distributions > EVHFIIJNMOHHL

EVHFIIJNMOHHL

Details

Distribution domain name copied

d8884lazs62z9.cloudfront.net

ARN
arn:aws:cloudfront::058264112988:distribution/EVHFIIJNMOHHL

Last modified
July 29, 2024 at 3:44:07 PM UTC

Settings

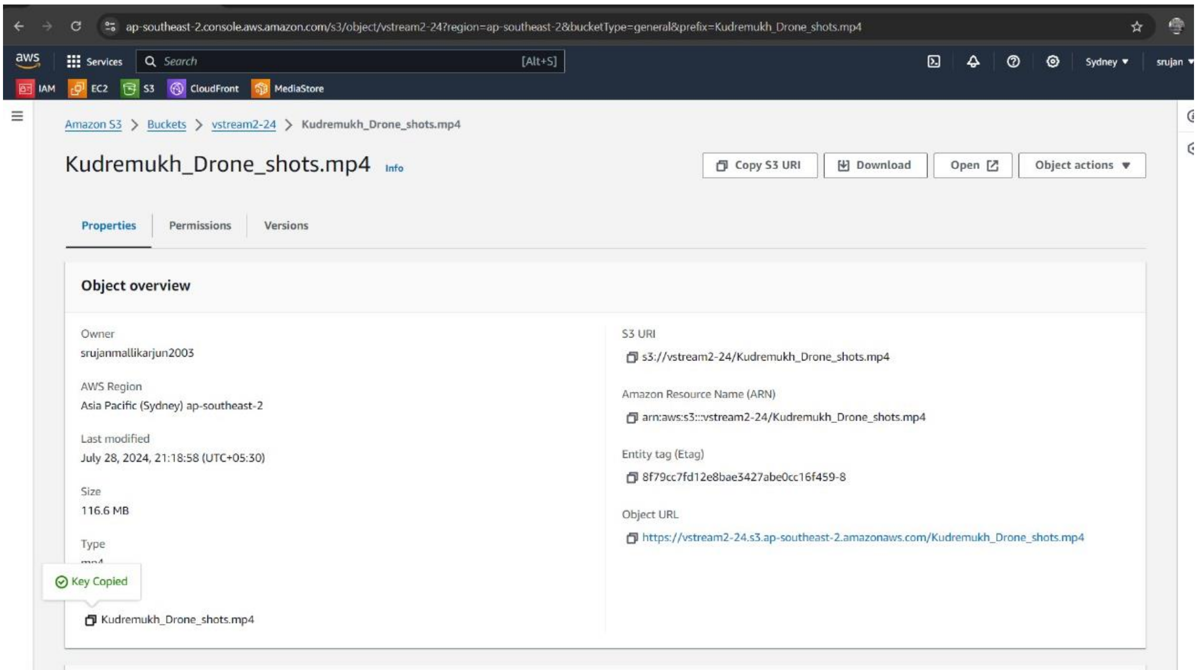
Description
-
Price class
Use all edge locations (best performance)
Supported HTTP versions
HTTP/2, HTTP/1.1, HTTP/1.0

Alternate domain names
-

Standard logging
Off
Cookie logging
Off
Default root object
-

From the created distribution, the required domain name is copied to create the required URL by pasting it as a suffix.

5.12 COPYING KEY FROM S3 BUCKET



ap-southeast-2.console.aws.amazon.com/s3/object/vstream2-24?region=ap-southeast-2&bucketType=general&prefix=Kudremukh_Drone_shots.mp4

Amazon S3 > Buckets > vstream2-24 > Kudremukh_Drone_shots.mp4

Kudremukh_Drone_shots.mp4 Info

Object overview

Owner
srujanmallikarjun2003

AWS Region
Asia Pacific (Sydney) ap-southeast-2

Last modified
July 28, 2024, 21:18:58 (UTC+05:30)

Size
116.6 MB

Type
mp4

Key Copied

Kudremukh_Drone_shots.mp4

S3 URI
s3://vstream2-24/Kudremukh_Drone_shots.mp4

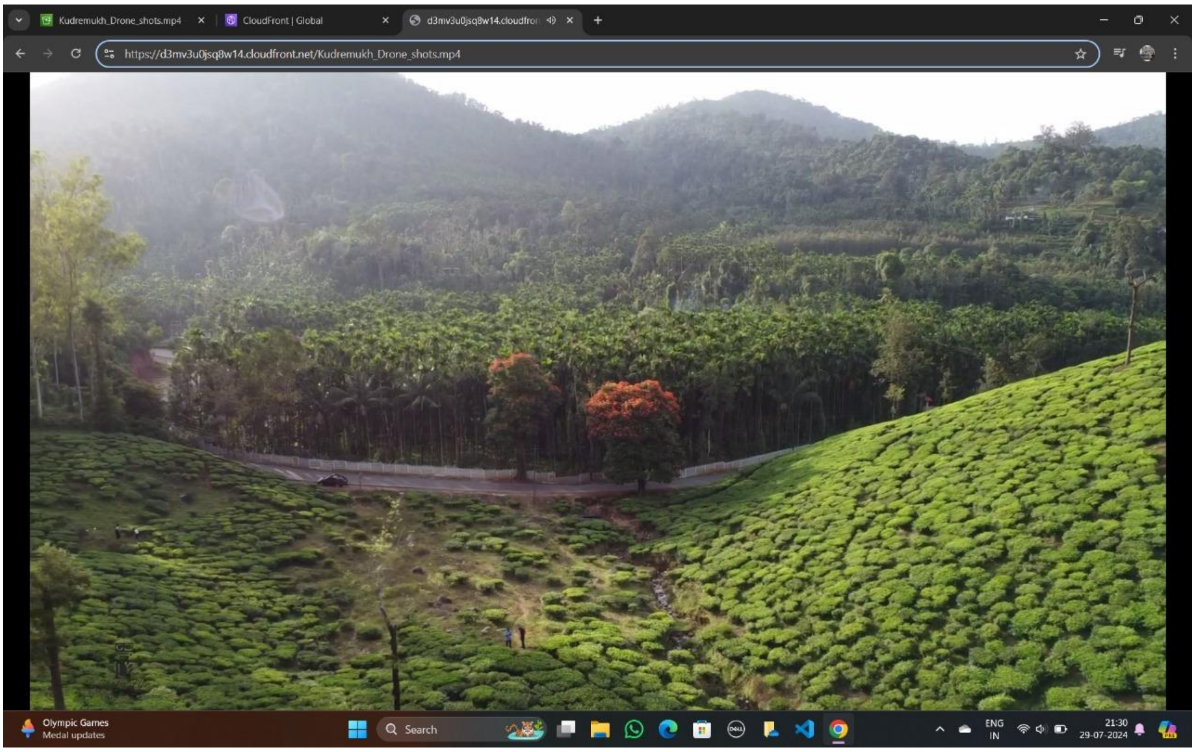
Amazon Resource Name (ARN)
arn:aws:s3:::vstream2-24/Kudremukh_Drone_shots.mp4

Entity tag (Etag)
8f79cc7fd12e8bae3427abe0cc16f459-8

Object URL
https://vstream2-24.s3.ap-southeast-2.amazonaws.com/Kudremukh_Drone_shots.mp4

From the created S3 bucket, the key is copied and pasted as the second part of the URL suffix to play the uploaded video.

5.13 VIDEO PLAYED



The uploaded video to the S3 bucket is played by above created URL. The URL can be searched in any search engine of your choice.

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

In summary, using Amazon S3 buckets for video storage provides a robust foundation for managing large volumes of media files. S3's scalability, durability, and cost-effectiveness make it an ideal choice for storing video content, whether it's for on-demand streaming or live broadcasts. Its integration with AWS's ecosystem ensures that your content is reliably preserved and easily accessible, streamlining the workflow for content management and distribution. On the other hand, Amazon CloudFront enhances the video streaming experience by delivering content with low latency and high transfer speeds. This results in faster load times and smoother playback, regardless of where your audience is located. The combination of S3 and CloudFront ensures that viewers experience minimal buffering and high-quality streaming.

Together, S3 and CloudFront provide a comprehensive and efficient solution for video streaming needs. By leveraging these AWS services, you benefit from a scalable and resilient infrastructure that supports a seamless viewing experience. This synergy not only optimizes content delivery but also reduces operational complexity, allowing you to focus on creating and delivering exceptional video content while AWS handles the technical challenges of storage and distribution.

FUTURE ENHANCEMENT

Future enhancements in video streaming using Amazon S3 buckets and CloudFront could significantly improve both performance and user experience. Integrating advanced transcoding services like AWS Elemental MediaConvert will enable automatic conversion of video into various formats and resolutions, ensuring compatibility across diverse devices and network conditions. Adaptive bitrate streaming, supported by AWS Elemental MediaPackage, will further enhance playback quality by dynamically adjusting video resolution based on the viewer's internet speed, minimizing buffering and interruptions. Additionally, incorporating AI-driven content analysis with services such as Amazon Rekognition can enable more intelligent content management, including automated tagging and enhanced search functionalities.

AWS CloudWatch will provide valuable insights into streaming performance and viewer engagement, allowing for proactive adjustments and optimizations. Integration with live streaming solutions like AWS Elemental MediaLive will enable seamless delivery of live events with minimal latency. Together, these advancements will provide a more robust, secure, and high-quality video streaming experience.

REFERENCES

- **https://aws.amazon.com**

- **https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html**

- **https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_create_for-service.html**

- **https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/permissions-reference-cw.html**

- **https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html**