

Assignment 1

04/05

i. Use S3 bucket and host video streaming.

→

a. Setup on Amazon S3 bucket.

1. Search for S3 on the services section. Click on it, then click on create bucket. This will direct you to the bucket creation page, now name the bucket.
2. Maintain other options as default, click on create bucket. After bucket has been created, now we need to add our video in this bucket. For that click on the name of the bucket, this will restrict you to the objects screen which shows the objects of your bucket. Click on upload. Select add files. An mp4 extension file is needed as we need to host a video.

b. Setup Cloudfront.

1. As the video is being uploaded, search for cloudfront on the services tab and open it in a new tab.
2. On the left pane under security, you will find origin access. Click on it, then click on identities (legacy). Click on create origin access identity. Give the identity a name and click on create.
3. Go back to distributions on the left pane and click on create cloudfront distribution.
4. Here in origin field, select the S3 bucket where the video is uploaded. Under origin, access, select legacy access, identities. Here select the identity you have created. Under bucket policy, select updating the bucket S3 policy.

In default cache behaviour, under viewer, select redirect http to https. Under web application firewall, select enable security protections to provide a layer of security.

5. keep remaining options as default and click on create distribution.

c. Accessing the hosted video.

1. Once the distribution is deployed, copy the domain name of your distribution.
2. Now go to S3 bucket and click on its name. Click on the name of your video you have uploaded.
3. Combine the domain name of the distribution & the key of the video to make your final link of the video that is streamed.

2. Discuss BMW and Hot Star case studies using AWS.

- BMW Group case study with AWS
- Overview: - BMW Group one of the world's leading premium automobile manufacturers uses AWS to drive innovation and efficiency in its IT infrastructure. The company leverages AWS services to build a seamless and highly scalable platform, which supports its connected car architecture and provides a seamless digital experience to customers.
- Challenges: BMW faced challenges in managing a global network of data centers that required significant maintenance and operational overhead. The need to analyze vast amounts of data generated by connected cars & deliver updates to millions of vehicles worldwide made scalability & real time processing crucial.
- Solutions with AWS:
 - Connected car platform: BMW builds its connected car platform on AWS using services like Amazon S3, Amazon EC2 and AWS Lambda. This platform connects and processes terabytes of vehicle sensors data, enabling real time analytics and enhancing predictive maintenance.
 - Data storage & analytics: BMW uses Amazon S3 for scalable data storage and Amazon Redshift for data analysis.
 - Machine learning: Used Amazon SageMaker to build models for improved safety & services.

Hotstar case study with AWS.

- Overview: Hotstar, one of India's largest streaming platforms, uses AWS to manage traffic peaks during high profile live events.
- Challenges: Hotstar needed to handle unpredictable traffic spikes especially during events like IPL, with millions of concurrent viewers.
- AWS solutions.
 - Scalable architecture: Hotstar implemented AWS services like Amazon CloudFront and Amazon S3 to manage traffic peaks effectively. This setup allowed Hotstar to scale dynamically and handle over 25 million concurrent viewers during IPL.
 - Serverless architecture: AWS Lambda and Amazon DynamoDB were used to build serverless components, reducing the operational overhead and ensuring that the infrastructure scaled automatically with increasing traffic.
 - Content delivery: Hotstar leverages Amazon CloudFront, a global content delivery network (CDN) to distribute video content to users worldwide with low latency and high transfer speeds.

3 why kubernetes and advantages and disadvantages of kubernetes? Explain how adidas uses kubernetes.

kubernetes is an open source container orchestration platform that automates the deployment, scaling and management of containerized applications. It allows for efficient management of clusters of containers, typically used in microservices architecture. Here is a breakdown of advantages & disadvantages.

Advantages.

- Scalability: kubernetes enables automatic scaling of applications based on the demand for resources.
- high availability: It ensures high availability of applications through container redundancy.
- self healing: It is a container, crashes kubernetes automatically, restarts it and if a node fails, it redistributes the containers.
- flexibility: It works across different environment, whether on premise or cloud.

Disadvantages.

- complexity: Setting up & managing kubernetes can be complicated, especially for small teams or organization.
- learning curve: It requires a deep understanding of containerization, networking and cloud native concepts.

- **Resource overhead:** Running kubernetes clusters consumes significant system resources which might be overall for small applications.
- **Operational overhead:** Maintaining kubernetes clusters can require ongoing maintenance and management, often needing specialized devops expertise.

Adidas leverages kubernetes to power its digital transformation and improve the scalability and reliability of its digital infrastructure.

- Microservices architecture
- scaling for peak demand.
- agile development.
- cloud native approach.
- improved CI/CD pipelines.

Adidas Packages its applications into containers using Docker or similar tools. Apps are deployed in kubernetes as pods, smallest deployable container unit. Configmaps and secrets are used to store sensitive information such as API keys. Namespaces are used to logically separate resources of different teams (like dev, prod, test)

4. What are Nagios and explain how Nagios are used in E-services?

→

Nagios is an open source monitoring system that helps organizations monitor systems, network and performance. It provides alerting services for servers, applications and network devices.

Key features of Nagios.

- Monitoring: Tracks the status of hosts and services.
- Alerting: Sends notifications via email or sms when issues are detected.
- Plugins: Supports various plugins to extend functionality for monitoring different services.
- Web interface: Offers a user friendly dashboard to visualize system health.
- Customizable: Highly configurable to fit specific monitoring needs.

How Nagios is used in E services.

- Uptime monitoring: Ensures that web applications and services are always available to users. Nagios can continuously check the availability of websites and alert administrators of any downtime.
- Performance monitoring: Tracks resource usage (cpu, memory)

(disk, network) to ensure that servers are operating efficiently. This helps in proactive resource management.

- Service monitoring: Monitors specific applications and services (like databases or email servers) to ensure they are running correctly and responding to requests.
- Alerting and Reporting: Provides real time alerts for system failures or performance degradation enabling quick responses to issues that could affect service delivery.
- Capacity planning: Gathers historical data that can help in forecasting future needs allowing organizations to scale their infrastructure accordingly.