

Q.1 Use S3 bucket and host Video Streaming.

⇒ Steps:

1. Create a S3 bucket:

- Search for S3 under services & click on Create Bucket.
- Since you are using Academy account, your region would be most probably Virginia (you can change this)
- Set the video permissions to public to allow public access using IAM (Identity & Access Management) or private.

2. Upload Video to S3 bucket:

- After successful creation of bucket, click on it & then click on upload to upload your video file (or using AWS CLI).

3. Enable static website hosting:

- In bucket settings, in properties tab enable Static Website Hosting.
- Your video will have an endpoint URL which can be used to access it using a browser.

4. Use Cloudfront for Optimized Streaming:

- Cloudfront is AWS's Content Delivery Network (CDN) that catches your video in multiple edge locations worldwide, improving load times for users located far from your S3 bucket.
- Set S3 bucket as the origin for Cloudfront, distribute the video using the Cloudfront provided URL.

5. Test your Setup:

- After configuring CloudFront test the URL in a browser to (test) ensure it runs smoothly without buffering.
- You can also use performance testing tools to monitor it.



Q2. Discuss BMW and Hotstar Case study using AWS.

⇒ BMW Case Study: BMW leverages AWS cloud services to manage its connected vehicle platform. This platform gather and processes real-time data from millions of BMW cars globally, providing insights into vehicle health, driving patterns & user preferences.

• Key AWS Services:

- AWS IOT: BMW uses AWS IOT to connect millions of vehicles, allowing data to be gathered in real time.
- Amazon S3: The large volume of data is securely stored in Amazon S3, offering durability & scalability.
- Amazon EC2: BMW utilizes EC2 instances for real-time data analysis & processing.
- Amazon SageMaker: This machine learning service allows BMW to predict vehicle maintenance & optimize performance based on the data collected.
- AWS Lambda: Provides event-driven computing capabilities which helps BMW react to real-time data changes without the need to provision servers.

• Benefits to BMW:

- Scalability of AWS services enables BMW to manage the ever-increasing data from its growing fleet of vehicles.
- The platform enhances vehicle safety, navigation, infotainment services through predictive analytics, ultimately improving customer satisfaction.



### \* Hotstar Case Study:

- Hotstar is one of the world's largest streaming platforms particularly popular during live events like IPL.
- It faced the challenge of handling millions of concurrent viewers during high-traffic events, which they solved using AWS.

#### • Key AWS Services:

- Amazon CloudFront: Hotstar uses CloudFront, AWS's content delivery network, to distribute live streams efficiently to millions of users, ensuring low latency & high quality video streaming.
- AWS Lambda: Hotstar leverages AWS Lambda to automatically scale resources based on traffic demands, particularly during unpredictable surges in viewership.
- Amazon EC2 Auto Scaling: Hotstar employs EC2 Auto Scaling to automatically adjust the no. of instances as per demand, ensuring seamless streaming for millions of viewers.
- Amazon RDS & Amazon S3: RDS is used for database management, while S3 stores video content & related data.

#### • Benefits to Hotstar:

- AWS enables Hotstar to handle over 10 million concurrent viewers during live sports events.
- With AWS Auto Scaling, Hotstar ensures cost-effective operations, scaling up resources only when necessary.
- CloudFront allows Hotstar to deliver content quickly & without buffering, no matter where the viewer is located globally.



Q3. Why Kubernetes? Advantages & disadvantages of Kubernetes.  
Explain how Adidas uses Kubernetes?

⇒ \* Why Kubernetes?

- Kubernetes is the leading container orchestration platform used to manage & automate the deployment, scaling and operation of containerized applications.
- Kubernetes is highly popular in modern DevOps practices for managing microservices architectures.

\* Advantages of Kubernetes:

- Automatic Scaling: Kubernetes can scale applications horizontally by automatically adjusting the no. of running containers based on real-time demand.
- Self-Healing: If a container crashes, Kubernetes automatically restarts it. If a node goes down, Kubernetes reschedules the container onto another node.
- Rolling Updates & Rollbacks: Applications can be updated without downtime, also if something goes wrong, it can automatically roll back to the previous version.
- Portability & Flexibility: Kubernetes works across different environments, including on-premise, public clouds (AWS, GCP, Azure) & hybrid setups.
- Efficient Resource Utilization: It optimizes CPU & memory usage across all running containers.

\* Disadvantages of Kubernetes:

- Complexity: Setting up & maintaining Kubernetes can be complicated. Beginners face a steep learning curve due to its wide range of features & configurations.
- Resource Overhead: It can be overkill for small projects. Running Kubernetes clusters requires significant resources, even when managing simple workloads.



- Security Management: It offers powerful controls, but it requires a robust understanding of its security practices to avoid misconfigurations that could lead to vulnerabilities.

\* Adidas Use of Kubernetes: Adidas adopted Kubernetes on Google Cloud to build and manage their global e-commerce platform. By using Kubernetes:

- Reduced Deployment Time: The deployment time for new applications & services has dropped from hours to just minutes.
- Efficient Scaling: During high-traffic events like sales, Adidas can quickly scale their resources using Kubernetes auto-scaling features.
- Improved Developer Productivity: It adopted a micro-services-based architecture using Kubernetes, allowing development teams to work more independently, deploy faster & achieve continuous delivery.

Q4. What is Nagios & how is it used in E-services?

⇒ \* What is Nagios?

- Nagios is a powerful open-source tool that monitors systems, network & infrastructure.
- It is widely used to keep track of system metrics, detect failures and send alerts to administrators before outages or performance degradation occurs.

\* Key features of Nagios:

- Comprehensive Monitoring: Nagios monitors server resources, network devices, applications, services and even websites. It checks metrics such as CPU usage, disk space, memory consumption & application states.



- Custom Alerts: Administrators can set thresholds for alerts. For eg, Nagios can send notifications when CPU usage exceeds 90% or disk space falls below a certain level.
- Event Handler Integration: Nagios allows users to automate actions when certain event occurs. For eg, if a service fails, Nagios can attempt to restart it automatically.
- Scalability: It can monitor large infrastructures with 1000's of nodes through distributed monitoring setups.

\* How Nagios is used in E-services:

- Server Monitoring: E-service providers use Nagios to ensure their servers (whether on-premises or cloud based) are always operational. It monitors the uptime, resource usage, critical services running on the servers.
- Network Monitoring: It keeps a track of network devices like routers, switches & firewalls. It checks for downtime, packet loss & bandwidth issues, ensuring the seamless operation of networked services.
- Application Monitoring: It monitors essential applications like web servers (Apache, Nginx), database servers (MySQL, PostgreSQL) & email services. It alerts the technical team whenever critical services become unavailable.
- Incident Management: In large org., Nagios integrates with incident management tools (e.g., ServiceNow or Jira) to automatically create tickets for issues, speeding up the resolution process. This <sup>helps</sup> prevents downtime, ensuring e-services remain available to customers.

By deploying Nagios, businesses can proactively manage their IT infrastructure, ensuring smooth & reliable delivery of e-services which results in minimal downtime & higher user satisfaction.