Assignment-1

05/05

Q.1. Use S3 bucket and host video streaming.

=> steps to host video on AWS S3 bucket.

(1) Download any sample video from internet

(2) Now break the whole video into smaller segments so that it can be easily transmitted over network

(3) To make smaller chunks we would ffmpeg tool, the general syntax is ffmpeg -i <VIDEO_NAME> -profile:v baseline -level 3.0 -start_number 0 -hls_time 3.5 -hls_list_size 0 -f hls output.m3u8.

(4) Ensure you download the ffmpeg before executing above command

(5) After that create a new S3 bucket, name it and allow public access

(6) Now, change the bucket policy so that anyone can access bucket objects.

(7) We also need to setup CORS policy so that any endpoint can request for resources of bucket.

(8) After setting up all necessary configurations, we need to upload video segments that we had created previously.

(9) Create a new folder named 'hls' inside bucket and upload all the video segments in it.

(10) Now we would create a simple HTML document that would be hosted on S3 bucket so that video can be played.

(11) The HTML file would contain the link of the main playlist of the video segments

(12) Open the link provided inside object properties

(13) The video will start streaming.

Q.2. Discuss BMW and Hotstar case studies using AWS.

=>

BMW Case Study:

Challenges:

(1) Data Volume: connected vehicles generate vast amounts of data, necessitating efficient storage and analysis.

(2) Real-Time Analytics: The need for real-time processing of data to enhance decision-making in areas such as predictive maintenance & customer service.

(3) Scalability: As the number of connected vehicles grew, BMW required a scalable solution that could adapt to varying workloads without compromising performance.

Solutions:

(1) Cloud infrastructure: BMW migrated its data and applications to AWS to leverage a flexible, scalable infrastructure. This included services such as Amazon EC2 for compute capacity and Amazon S3 for storage.

(2) Data Analytics and Machine Learning: BMW used SageMaker and Kinesis to build, train and deploy machine learning models and to employ real-time data processing.

(3) IoT integration: BMW utilised AWS IOT services to connect their vehicles, enabling features like over-the-air updates, real-time navigation, and safety alerts.

In summary, BMW's adoption of AWS transformed its operational processes, resulting in enhanced efficiency, reduced costs, and faster deployment

of new features. The cloud infrastructure facilitated innovative connected vehicle services, ultimately leading to improved customer satisfaction and loyalty. This strategic shift positions BMW as a leader in the automative industry, ready to meet evolving consumer expectations and market demands.

Hotstar Case Study.

Challenges:

(1) Traffic spikes: During events like cricket tournaments, Hotstar faced unprecedented traffic that often led to service disruptions and poor user experience

(2) Content Delivery: With a vast library of content, ensuring smooth and fast delivery of streams was crucial, particularly during high-demand periods.

(3) Data Insights: understanding user behavior and preferences was essential for content recommendations and targeted marketing.

Solutions:

(1) Auto Scaling Groups: Hotstar implemented AWS auto-scaling to automatically adjust capacity based on real-time demand, ensuring the platform could handle millions of users simultaneously without degradation in service.

② Amazon Cloudfront: This content Delivery Network (CDN) distributed content globally, reducing latency and ensuring high-quality video streaming regardless of user location.

③ Amazon S3: Hotstar utilized S3 for storing video content and user data efficiently. The durability and scalability of S3 allowed them to grow their content library without worrying about infrastructure limitations.

④ Amazon Redshift: This data warehousing service enabled Hotstar to perform complex queries and analytics on user data, leading to insights that informed content strategy and marketing efforts.

⑤ AWS Lambda: By using Lambda, Hotstar developed a highly responsive application architecture that allowed for quick processing of user requests, enhancing overall performance.

⑥ Amazon DynamoDB: This NoSQL database was employed for storing user session data, allowing for rapid read and write operations essential for a streaming service.

In conclusion, Hotstar's use of AWS enabled it to successfully handle high-volume streaming, reaching over 25 million concurrent users during peak events without interruptions. The insights gained from data analytics allowed for a tailored content strategy, boosting user engagement and subscriptions. Additionally, the flexible infrastructure provided cost efficiency, ensuring effective expenditure management during critical usage periods.

Q.3. Why Kubernetes and Advantages and disadvantages of Kubernetes. Explain How adidas uses Kubernetes

=>

Kubernetes is an open-source container orchestration platform designed to automate the deployment, scaling, and management of containerized applications. It provides a framework for running distributed systems resiliently, offering features such as load balancing, service discovery, automated rollouts and rollbacks, and resource monitoring.

Advantages of Kubernetes.

1. Scalability : Kubernetes can scale applications up or down automatically based on demand, ensuring optimal resource utilization.
2. High Availability: It automatically manages the availability of applications, restarting failed containers and rescheduling them.
3. Declarative configuration: Kubernetes uses a declarative approach, allowing users to define the desired state of the system.
4. Load Balancing: Built-in load balancing distributes network traffic efficiently across containers, improving performance.
5. Resource Management: Kubernetes can manage resources efficiently by allocating CPU and memory based on the needs of the containers.

Disadvantages of Kubernetes

1. Complexity: Kubernetes has a steep learning curve and can be complex to set up and manage.
2. Overhead: The resource overhead of running Kubernetes itself can be significant, especially for smaller applications.
3. Debugging challenges: Troubleshooting issues in a distributed system can be difficult, requiring advanced monitoring and logging setups.
4. Security: Kubernetes introduces new security challenges, such as securing the API server, managing access controls, and ensuring container security.

Adidas' Use of Kubernetes

Adidas has adopted Kubernetes to enhance its digital infrastructure, enabling better scalability and faster deployment.

Key uses:

1. Scalability for E-commerce: During peak shopping seasons or promotional events, Adidas uses Kubernetes to dynamically scale its applications.
2. Microservices Architecture: Adidas employs a microservices architecture that allows different parts of their platform to be developed and deployed.
3. Continuous Deployment: With Kubernetes, Adidas can implement continuous Integration/Deployment (CI/CD) pipelines.
4. Resource optimization: Kubernetes helps Adidas efficiently manage its cloud resources, ensuring that applications are running optimally while controlling costs.

⑤ Improved Reliability: The platform enhances the reliability of Adidas's services, automatically recovering from failures and maintaining application performance.

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

⇒ Nagios is an open-source monitoring system designed to monitor systems, networks, and infrastructure. It helps IT professionals ensure that critical services are running smoothly and provides alerts when issues arise. Key features include:

① Monitoring of various resources such as servers, network devices, applications and services
② Alerting via SMS, email when problems occur.
③ Reporting historical data and trends on the monitored infrastructure which can aid in troubleshooting.
④ Extensibility: Supports plugins, allowing for custom checks and integration with other tools.

Nagios's use in E-services.

In context of E-services, which involve online platforms and applications, Nagios plays a crucial role in:

① Server Monitoring: Nagios continuously monitors server health, including CPU usage, memory utilization, etc. It also monitors critical services (e.g. web servers & database) to ensure they are up and running.

(2) Network Monitoring: Monitors routes, switches, and firewalls for connectivity and performance issues. Tracks bandwidth usage and identifies potential bottlenecks.

(3) Application monitoring: checks the response time and availability of web applications. Monitors database performance metrics and availability, helping prevent downtime due to database issues.

(4) Alerting and Notifications: sends alerts when thresholds are exceeded (e.g. high CPU usage). It also integrates with ticketing systems to create incidents automatically.

(5) performance and availability Reporting: provides reports on uptime and performance trends, assisting in SLA compliance and capacity planning.

(6) Scalability: Can monitor multiple environments and remote locations, making it suitable for large-scale E-service infrastructures.