Name - Pranav P. Titambe , 62/D15A.

DOMS | Page No. 1 | Date / /

05/05

Advance DevOps
Assignment - 1

Q2] Discuss BMW and Hotstar cas study using AWS.

→ a) BMW case study using AWS

BMW has leveraged AWS for various aspects of their operations, focusing on.

- Connected car - BMW uses AWS to power its connected car solutions, providing real-time data processing and analytics for vechile-to-cloud interactions. This allows BMW to gather insights into vehicle performace, driver behaviour and more, enhancing the overall during experience.
- Scalibility & flexibility - By using AWS services like Amazon S3 (for data storage) Amazon EC2 (for computing resources), and Aws lambda (for serverless computing), BMW can scale their applications globally without being limited by infrastructure.
- Security and compliance - AWS services help BMW maintain strict security and compliance regulations, including GDPR, ensuring that customer data is secure.

b) Hotstar case study using AWS

Hotstar, India's largest streaming platform, uses AWS for its live streaming and video-on-demand services. Here's how AWS played a crucial role:

- Handling massive scale for live event - One of Hotstar biggest chanallenges was scaling its infrastructu

to handle peak traffic during popular live-streamed events like the Indian Premier League (IPL) cricket matches. In 2019, Hotstar achieved a record of 25.3 million concurrent viewers for an IPL match, made possible by AWS's scalibility.

- Elastic Load Balancing and EC2 - Hotstar utilizes Amazon EC2 instances to scale compute resources dynamically. Elastic Load Balancing (ELB) helps distribute incoming traffic across multiple instances, allowing Hotstar to handle millons of users simultaneously during peak times.
- Serverless Architecture - Hotstar uses AWS Lambda to run code without provising or managing servers, which helps in scaling automatically for peak usage, particularly during live sporting events.

Q3] Why kubernetes and advantages and disadvantages of kubernetes. Explain How adidas uses kubernetes

→ Kubernetes (often abriviated as K8s) is an open-source container orchestration platform developed by google that automates the development, deployment, scaling, and management of containerized applications.

Why Kubernetes?

Kubernetes solves many challenges faced by developers

and operations teams when managing large-scale, complex containerized applications.

① Automated scaling: Kubernetes can automatically scale applications up and down based on traffic and resource needs, ensuring efficient use of infrastructure

② Self-healing: K8s monitors the health of containers and automatically replaces or restarts them if they fail.

③ Declarative configuration: Kubernetes allows for declarative infrastructure management, where users define the desired state of the applications (like how many containers should be running), and K8s ensures it is maintained.

④ DevOps Integration: Kubernetes ~~first~~ fits well into modern CI/CD pipelines and supports advanced use case like blue-green deployment and canary releases.

Advantages of kubernetes:

① Scalibility: Automatically scales applications horizontally based on load and resources needs, ensuring optimal performance under varying conditions.

④ High availability: Built-in load balancing and failover capabilities ensure that services remain available even if some instances go down.

⑤ Efficient Resource Management: K8s allocates resources dynamically based on workload requirements optimizing infrastructure usage.

Disadvantages of Kubernetes:-

① Complexicity: Kubernetes has a steep learning curve, and managing a kubernetes cluster can be complex, especially for smaller teams or less experienced users.

② Resource intensive: Running a kubernetes cluster requires significant computational resources, making it potentially overkill for ~~smarter~~ smaller applications or team

③ Networking challenges: Networking in kubernetes can be complex, especially when dealing with multi-cluster environments or ensuring network security between services.

How Adidas Uses Kubernetes:-

① Multi-cloud strategy - Adidas deploys kubernetes in a multi-cloud environment, utilizing both Google Cloud Platform (GCP) and AWS. This flexibility helps

Adidas avoid vendor lock-in while ensuring they can deploy services across different regions and data centers for better performance and availabilit

(2) CI/CD Integration : kubernetes is tightly integrated into Adidas CI/CD pipeline, allowing the company to automate deployments, run tests, and deliver new features faster.

(3) Resilience and high availability : Adidas relies on kubernetes for managing failover, replications and load balancing, which ensures that services remain available even during infrastructure failures or maintenance periods.

Q4] What are Nagios ~~and explain~~ how Nagios are used in E-services?

→ Nagios is an open-source monitoring tool designed for IT infrastructure monitoring. It helps organisations monitor network devices, servers, applications, and services to ensure they operate optimally and to detect potential issues before they lead to service disruptions.

How are Nagios is used in E-services.
E-services, which include online platforms offering electronic services such as e-commerce websites, online banking, government portals, and other

web-based applications, require reliable, continuous operations. Nagios is used in E-services to ensure high availability, service reliability, and performance.

① Server and applications monitoring - Nagios monitors server metrics like CPU load, memory usage, disk space, and network usage, alerting administrators where thresholds are exceeded.

② Networking Monitoring - E-services rely heavily on uninterrupted network connectivity. Nagios monitors network devices like routers, switches, firewalls, and load balancers, detecting potential problems that could lead to outages.

③ Database Monitoring - Databases are crucial for the functioning of e-services. Nagios ensures that databases (like MySQL, PostgreSQL) are running, and it checks for performance issues like slow query execution times.

Q1] Use S3 bucket and host video streaming

→ steps to implement / perform hosting video streaming on S3 bucket.

① Create an S3 bucket
- Login to your AWS Management Console
- Navigate to the S3 service and click on Create bucket.
- Configure the bucket
  - Choose the unique name for the bucket.
  - Select the region
  - Disable 'block all public access'. Since we want the video to be publicly accessible
  - Click Create bucket.

② Upload video files to the S3 bucket.
- Once the bucket is created, go into the bucket and click upload.
- Upload your video files, S3 supports various video formats like MP4, mov and MKV.

③ Set Permissions for Public Access.
- If your video is to be publicly accessible.
- Go to the permissions tab of the S3 bucket.
- Edit the bucket policy and add a policy to allow public access. Here's an example policy.

```
{
  "Version" : "2012-10-17",
  "statement" : [
    {
      "Effect" : "Allow",
      "Principal" : "*",
      "Action" : "s3:GetObject",
      "Resource" : "arn:aws:s3:::your-bucket-name/*"
    }
  ]
}
```

This will allow public read access to objects in your bucket.

④ Set up static Website Hosting
- Go to the 'Properties' tab of your S3 bucket.
- Scroll down to the 'Static web hosting' section.
- Enable 'Static website hosting' and specify an index document (if you have an HTML file for users to access the video).
- You will get a 'Bucket website endpoint' that can be used as the URL to access the video or your website

⑤ Now click on the bucket endpoint and your video will be streamed onto the webpage.

Now you have streamed video on S3 bucket.