.DOCKER EXAM.

Mar 25, 2025

1. How do you deploy Docker containers on AWS?

Deploying Docker containers on AWS can be done using Amazon ECS, EKS, or EC2 instances.

- **Push Image:** First, you push the container image to Amazon ECR or Docker Hub.
- **Define Task:** Then, you create a task definition in ECS or a pod definition in EKS.
- **Launch Container:** Finally, you launch the container in a cluster. For EC2, you manually install Docker and run the container using Docker commands.

2. What is the role of Amazon ECR (Elastic Container Registry)?

Amazon ECR is a fully managed container registry that stores, manages, and secures Docker images.

- **Storage & Security:** It provides a secure place to store container images.
- **Integration:** Works seamlessly with ECS, EKS, and Lambda for easier deployments.
- **Automation:** Can automatically scan and replicate images across AWS regions.

3. What is the difference between ECS and EKS in AWS?

Both ECS and EKS manage containers, but they differ in approach.

- **ECS (Elastic Container Service):** AWS-native service, easier to set up, and fully integrated with AWS.
- **EKS (Elastic Kubernetes Service):** Managed Kubernetes service, offering more flexibility for running Kubernetes-based applications but requiring more configuration.

4. How does Docker integrate with AWS CI/CD pipelines?

AWS services can automate Docker builds and deployments.

- **Build & Push:** AWS CodeBuild compiles and pushes Docker images to ECR.
- **Automated Deployment:** CodePipeline automates deployment to ECS or EKS using CodeDeploy.

- **Continuous Monitoring:** CloudWatch and X-Ray help track deployments and performance.

5. What is the role of AWS Fargate in Docker container deployment?

AWS Fargate provides a serverless approach to container management.

- **No Infrastructure Management:** Runs containers without provisioning EC2 instances.
- **Automatic Scaling:** Adjusts resources based on workload.
- **Security & Cost Efficiency:** Offers isolation and cost savings by charging for only the resources used.

6. How do you manage Docker container scaling on AWS?

Scaling ensures optimal performance and cost efficiency.

- **ECS Service Auto Scaling:** Adjusts container instances based on CPU/memory usage.
- **Kubernetes Horizontal Pod Autoscaler (HPA):** Dynamically scales pods in EKS.
- **Load Balancers:** AWS ALB/NLB distribute traffic effectively to maintain performance.

7. What is a Docker Compose file, and how can it be used with AWS?

A Docker Compose file defines multi-container applications and their configurations.

- **Simplifies Deployment:** Helps manage multiple containers with a single YAML file.
- **AWS Integration:** Works with AWS Copilot CLI or ECS CLI for seamless deployments.
- **Infrastructure as Code:** Enables easy setup and scaling of containerized applications.

8. How do you monitor Docker containers in AWS?

Monitoring ensures application reliability and performance.

- **AWS CloudWatch:** Collects logs and metrics from running containers.
- **AWS X-Ray:** Provides distributed tracing to analyze performance.
- **Third-Party Tools:** Prometheus and Grafana offer additional visualization and alerting.

9. How do you secure Docker containers on AWS? Security is crucial for protecting containerized applications.

- **IAM Roles:** Restrict access based on the principle of least privilege.
- **AWS Secrets Manager:** Manages sensitive credentials securely.
- **Vulnerability Scanning:** Amazon ECR scans container images for security issues.
- **Network Policies:** Use VPC security groups and private subnets for protection.

10. How do you optimize cost while running Docker containers on AWS?

Optimizing costs ensures efficient resource usage.

- **Use AWS Fargate Spot:** Run containers at lower costs with spot pricing.
- **Right-size EC2 Instances:** Choose appropriate instance types for workloads.
- **Auto-Scaling:** Adjust resources dynamically to match demand.
- **Monitor with AWS Cost Explorer:** Track usage and cut unnecessary expenses.