DevOps Shack

Top 50 AWS DEVOPS Interview Questions And Answers

1. What is AWS DevOps?

Answer:

AWS DevOps is a combination of AWS services and DevOps practices that enable organizations to automate and streamline software development and deployment processes. It includes services like AWS CodePipeline, AWS CodeBuild, AWS CodeDeploy, AWS CloudFormation, AWS Lambda, and Amazon ECS to facilitate CI/CD, monitoring, and infrastructure automation.

2. How does AWS support DevOps?

Answer:

AWS provides:

- CI/CD: AWS CodePipeline, AWS CodeBuild, AWS CodeDeploy
- Infrastructure as Code (IaC): AWS CloudFormation, Terraform
- Monitoring & Logging: Amazon CloudWatch, AWS X-Ray, AWS Config
- Containerization & Orchestration: AWS ECS, EKS, Fargate
- Security & Compliance: AWS IAM, AWS Shield, AWS Inspector

3. What are the key AWS DevOps services?

Answer:

1. AWS CodePipeline – CI/CD automation
2. AWS CodeBuild – Compiles code, runs tests, and produces artifacts
3. AWS CodeDeploy – Automated deployments to EC2, Lambda, or on-premises
4. AWS CloudFormation – Infrastructure as Code (IaC) management
5. AWS Lambda – Serverless computing
6. Amazon ECS/EKS – Container orchestration
7. Amazon CloudWatch – Monitoring and logging
8. AWS IAM – Security and access management

4. Explain the AWS DevOps pipeline.

Answer:

A typical AWS DevOps pipeline consists of:

1. Source Code Repository (GitHub, CodeCommit)
2. Build & Test (AWS CodeBuild)
3. Artifact Storage (S3, ECR)
4. Deployment (AWS CodeDeploy)
5. Monitoring (CloudWatch, X-Ray)

5. What is CI/CD in AWS? How is it implemented?

Answer:

CI/CD is a practice that enables continuous integration and continuous deployment. It is implemented in AWS using:

- CodePipeline (for orchestrating CI/CD workflows)
- CodeBuild (for building & testing code)

- CodeDeploy (for deploying applications)

6. How do you automate deployments in AWS?

Answer:

Using AWS CodeDeploy with:

- Rolling Updates – Deploys updates incrementally
- Blue/Green Deployments – Creates two identical environments for seamless transition
- Canary Deployments – Deploys to a small subset before full rollout

7. What is AWS CodePipeline?

Answer:

AWS CodePipeline is a CI/CD service that automates software release processes by defining steps such as source, build, test, and deploy.

8. What is AWS CodeBuild?

Answer:

AWS CodeBuild is a fully managed build service that compiles source code, runs tests, and produces deployable artifacts.

9. What is AWS CodeDeploy?

Answer:

AWS CodeDeploy is a deployment service that automates software deployment to various AWS services like EC2, Lambda, and on-premises servers.

10. How does AWS CloudFormation work?

Answer:

AWS CloudFormation enables Infrastructure as Code (IaC) by allowing users to define AWS resources using YAML/JSON templates.

11. What is Terraform, and how does it differ from CloudFormation?

Answer:

Terraform is an open-source IaC tool that supports multiple cloud providers. Unlike CloudFormation (AWS-specific), Terraform is cloud-agnostic.

12. What is Amazon ECS?

Answer:

Amazon ECS (Elastic Container Service) is a managed container orchestration service for running Docker containers on AWS.

13. What is Amazon EKS?

Answer:

Amazon EKS (Elastic Kubernetes Service) is a managed Kubernetes service to deploy and scale containerized applications.

14. What is AWS Fargate?

Answer:

AWS Fargate is a serverless compute engine for containers that eliminates the need to manage EC2 instances.

15. How do you monitor AWS applications?

Answer:

Using AWS services like:

- **Amazon CloudWatch (logs, metrics, alarms)**
- **AWS X-Ray (distributed tracing)**
- **AWS Config (compliance monitoring)**

16. What is AWS Elastic Beanstalk?

Answer:

AWS Elastic Beanstalk is a PaaS service that automatically handles deployment, scaling, and monitoring of applications.

17. What is Amazon CloudWatch?

Answer:

Amazon CloudWatch is a monitoring and observability service for AWS resources and applications.

18. How does AWS Lambda work?

Answer:

AWS Lambda allows you to run code without provisioning servers. It executes functions in response to events from S3, DynamoDB, API Gateway, etc.

19. What is AWS IAM?

Answer:

AWS IAM (Identity and Access Management) controls user permissions, roles, policies, and authentication in AWS.

20. What is Auto Scaling in AWS?

Answer:

Auto Scaling dynamically adjusts EC2 instances based on demand using policies like Target Tracking, Scheduled Scaling, and Step Scaling.

21. What is Amazon Route 53?

Answer:

Amazon Route 53 is a scalable DNS service for routing traffic to AWS resources.

22. What are security best practices in AWS DevOps?

Answer:

- **Use IAM Roles instead of root users**

- Enable MFA for authentication
- Use AWS Secrets Manager for credentials
- Encrypt data using KMS
- Use AWS WAF & Shield for DDoS protection

23. What is AWS Config?

Answer:

AWS Config provides continuous monitoring and compliance auditing of AWS resources.

24. What is Blue-Green Deployment in AWS?

Answer:

It involves switching traffic from an old environment (blue) to a new one (green) to ensure zero-downtime deployment.

25. How do you ensure high availability in AWS?

Answer:

- Use Multi-AZ deployments
- Implement Auto Scaling
- Distribute traffic with Elastic Load Balancer (ELB)

26. What is AWS CodePipeline?

Answer:

AWS CodePipeline is a CI/CD service that automates the build, test, and deployment phases of application development.

How It Works:

1. Source Stage: Fetches code from GitHub, CodeCommit, or S3.
2. Build Stage: Uses AWS CodeBuild to compile the code and run tests.
3. Test Stage: Runs automated tests.
4. Deploy Stage: Deploys using AWS CodeDeploy, ECS, or Lambda.

27. What is Amazon ECR?

Answer:

Amazon Elastic Container Registry (ECR) is a fully managed Docker container registry that securely stores, manages, and deploys container images.

Key Features:

- Integrated with Amazon ECS & EKS
- Supports private & public repositories
- Provides image vulnerability scanning
- Uses AWS IAM for access control

28. How do you deploy a serverless application in AWS?

Answer:

You can deploy a serverless application using AWS Lambda and API Gateway.

Steps:

1. Write a Lambda function in Python, Node.js, or Java.
2. Package it using AWS CLI or SAM (Serverless Application Model).

3. Deploy it with AWS Lambda or AWS CloudFormation.
4. Use Amazon API Gateway to expose it as an API.
5. Store configuration in AWS Systems Manager Parameter Store.

Example:

```
Resources:
  MyFunction:
    Type: AWS::Lambda::Function
    Properties:
      Handler: index.handler
      Runtime: nodejs14.x
      Role: arn:aws:iam::123456789012:role/execution_role
```

29. What is AWS CloudTrail?

Answer:

AWS CloudTrail records all API calls & actions made in an AWS account, helping with audit, security, and compliance.

Features:

- Logs who did what and when
- Stores logs in S3 or CloudWatch
- Supports event filtering & alerts
- Helps detect unauthorized access

30. What is AWS Shield?

Answer:

AWS Shield is a DDoS protection service that safeguards AWS applications from attacks.

Types:

- AWS Shield Standard (Free) – Basic protection against common attacks.
- AWS Shield Advanced (Paid) – Advanced threat detection & response.

31. What is AWS Inspector?

Answer:

AWS Inspector is an automated security assessment tool that scans EC2 instances for vulnerabilities.

Features:

- Detects security loopholes & misconfigurations
- Supports CVE-based vulnerability scanning
- Integrates with AWS Security Hub

32. What is AWS App Mesh?

Answer:

AWS App Mesh is a service mesh that enables observability and networking for microservices.

Use Cases:

- Service-to-service communication control
- Traffic routing & load balancing
- Automatic retries & circuit breaking

33. Explain GitOps in AWS.

Answer:

GitOps is a DevOps practice where infrastructure and applications are deployed via Git repositories.

Implementation:

- Use AWS CodePipeline & CodeBuild for automation.
- Store IaC files in Git (Terraform, CloudFormation).
- Use FluxCD or ArgoCD for Kubernetes GitOps.

34. How do you set up Blue-Green Deployment in AWS?

Answer:

Blue-Green deployment reduces downtime and risk by running two environments:

- Blue (current production version)
- Green (new version to be tested and deployed)

Implementation in AWS:

1. Using AWS CodeDeploy:
   - Deploy the new version in Green environment.
   - Shift traffic using Route 53 DNS or Load Balancer.
2. Using Elastic Beanstalk:
   - Swap environment URLs between blue and green versions.
3. Using AWS Lambda & API Gateway:
   - Use Stage Variables and Traffic Shifting.

35. What is AWS Step Functions?

Answer:

AWS Step Functions is a serverless orchestration service that coordinates workflows using state machines.

Use Cases:

- Automating workflows (e.g., processing S3 uploads)
- Error handling & retries
- Chaining AWS services (Lambda, ECS, DynamoDB)

36. What is Canary Deployment in AWS?

Answer:

Canary deployment gradually shifts traffic to a new version of an application.

Example with AWS CodeDeploy:

1. Deploy 5% of traffic to the new version.
2. Monitor performance.
3. Gradually increase traffic to 100%.

37. What are the different types of Load Balancers in AWS?

Answer:

AWS offers 3 types of Elastic Load Balancers (ELB):

1. Application Load Balancer (ALB) – Layer 7, used for HTTP/HTTPS traffic.
2. Network Load Balancer (NLB) – Layer 4, used for high-performance TCP traffic.
3. Classic Load Balancer (CLB) – Legacy, supports both Layer 4 and Layer 7.

38. What is AWS Outposts?

Answer:

AWS Outposts extends AWS infrastructure on-premises for hybrid cloud solutions.

39. How do you implement logging and monitoring in AWS?

Answer:

Use Amazon CloudWatch for logs, metrics, and alarms.

1. **Enable CloudWatch Logs for EC2, Lambda, ECS.**
2. **Use AWS X-Ray for tracing requests.**
3. **Configure AWS Config for compliance monitoring.**

40. Explain AWS Service Catalog.

Answer:

AWS Service Catalog manages and deploys approved cloud resources in an organization.

41. How do you optimize costs in AWS DevOps?

Answer:

- **Use Spot Instances & Savings Plans.**
- **Implement Auto Scaling & Right-Sizing.**
- **Use S3 Intelligent-Tiering for storage.**

42. What is AWS Control Tower?

Answer:

AWS Control Tower sets up and manages multi-account AWS environments.

43. How does AWS Systems Manager work?

Answer:

AWS Systems Manager manages EC2 instances, automation, patching, and logs.

44. What is the difference between AWS Organizations and AWS Landing Zone?

- **AWS Organizations – Multi-account governance.**
- **AWS Landing Zone – Pre-configured multi-account setup.**

45. Explain the AWS Well-Architected Framework.

Answer:

It consists of 6 pillars:

1. **Operational Excellence**
2. **Security**
3. **Reliability**
4. **Performance Efficiency**
5. **Cost Optimization**
6. **Sustainability**

46. What is AWS OpsWorks?

Answer:

AWS OpsWorks is a configuration management service that automates application deployment using Chef and Puppet. It helps manage infrastructure as code (IaC).

Key Components:

1. OpsWorks Stacks – Manages EC2 instances, RDS databases, and EBS volumes.
2. OpsWorks for Chef Automate – Automates infrastructure using Chef recipes.
3. OpsWorks for Puppet Enterprise – Automates system configuration with Puppet.

Use Cases:

- Automating server configuration and patching.
- Managing multi-tier applications (Web, App, DB servers).
- Enforcing security policies using Chef/Puppet.

47. How do you implement DevSecOps in AWS?

Answer:

DevSecOps integrates security practices into the DevOps pipeline to ensure secure application development and deployment.

AWS DevSecOps Best Practices:

1. Identity & Access Management (IAM)
    - Use IAM roles & policies for access control.
    - Enable multi-factor authentication (MFA).
2. Infrastructure Security
    - Use AWS WAF to protect web applications.
    - Enable AWS Shield for DDoS protection.

    - Implement Amazon GuardDuty for threat detection.

3. Automated Security Testing
    - Use AWS CodeBuild to integrate security scans (e.g., OWASP ZAP, Snyk).
    - Enable AWS Security Hub to centralize security alerts.
4. Compliance & Governance
    - Use AWS Config for compliance checks.
    - Implement AWS Audit Manager for security audits.
5. Secure CI/CD Pipelines
    - Use AWS CodePipeline with security checks.
    - Scan containers using Amazon ECR Image Scanning.
6. Data Encryption & Secrets Management
    - Encrypt data using AWS KMS.
    - Store secrets in AWS Secrets Manager or SSM Parameter Store.

48. What is AWS Artifact?

Answer:

AWS Artifact is a compliance and auditing tool that provides security reports and compliance documents from AWS.

Key Features:

- Access SOC, ISO, PCI DSS compliance reports.
- Download security-related audit reports.
- Helps meet regulatory requirements for GDPR, HIPAA, etc.

Use Case Example:

A financial company needs to prove AWS PCI DSS compliance to regulators. They can download AWS Artifact reports and present them as evidence.

49. Explain AWS Transit Gateway.

Answer:

AWS Transit Gateway is a network hub that connects multiple VPCs, on-premises networks, and AWS services in a scalable manner.

How It Works:

- **Acts as a centralized router for all VPCs.**
- **Eliminates VPC peering complexity.**
- **Supports multicast and inter-region peering.**

Use Case Example:

A company has 5 VPCs in different AWS regions and an on-premises data center. Instead of managing multiple VPNs & VPC peering, they use Transit Gateway for seamless connectivity.

50. How do you secure an S3 bucket?

Answer:

To secure an Amazon S3 bucket, follow these best practices:

1. Apply IAM Policies

- **Deny public access unless explicitly needed.**
- **Grant least privilege access using IAM roles.**

2. Enable Bucket Policies

- **Use bucket policies to restrict access based on IP, user, or conditions.**

Example policy to allow access only from a specific IP:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::example-bucket/*",
      "Condition": {
        "NotIpAddress": {
          "aws:SourceIp": "192.168.1.1/32"
        }
      }
    }
  ]
}
```

3. Enable Encryption

- Use S3 Default Encryption (SSE-S3, SSE-KMS).
- Encrypt objects using client-side encryption.

4. Enable S3 Block Public Access

- Prevent accidental public access by enabling Block Public Access settings.

5. Use MFA Delete

- Enable MFA Delete to prevent unauthorized deletion of objects.

6. Enable AWS CloudTrail Logging

- **Track API actions on the S3 bucket.**

7. Enable S3 Object Lock

- **Prevent accidental deletion of critical files.**

8. Use VPC Endpoint for Private Access

- **Restrict bucket access only to VPC resources, avoiding internet exposure.**