

Here are 30 AWS DevOps scenario-based interview questions with real-time examples and answers, covering key topics like CI/CD, automation, monitoring, infrastructure as code (IaC), containers, and cloud best practices.

## 1. CI/CD &

## Automation

# 1. How would you implement a blue-green deployment strategy on AWS?

**Scenario:** Your team needs to deploy a new version of an application without downtime.

**Answer:**

- Use AWS Elastic Beanstalk or AWS

# CodeDeploy with blue (current) and green (new) environments.

- Deploy the new version (green) and test it.
- Switch traffic using Elastic Load Balancer (ELB) or Route 53 weighted routing.
- Rollback by switching back to the blue environment if issues

arise.

## 2. How do you set up a canary deployment on AWS?

**Scenario:** You want to release a new feature to 10% of users before a full rollout.

**Answer:**

- Use AWS CodeDeploy

with Linear or Canary deployment strategy.

- Route a small percentage (10%) of traffic to the new version.
- Use Amazon CloudWatch and AWS X-Ray for monitoring.
- If successful, gradually increase traffic to 100%.

### 3. How do you

# automate infrastructure deployment on AWS?

**Scenario:** You need to deploy EC2 instances, VPCs, and databases automatically.

**Answer:**

- Use AWS CloudFormation or

# Terraform to define infrastructure as code (IaC).

- Example Terraform script to create an EC2 instance: resource "aws\_instance" "web" {  
ami = "ami-12345678"  
instance\_type =  
"t2.micro" }
- Use AWS CodePipeline to automate

deployments.

# 4. How would you implement CI/CD for a microservices-based application on AWS?

**Scenario:** You have multiple microservices and want to automate builds and deployments.

**Answer:**

- Use AWS CodePipeline with CodeBuild (for building), CodeDeploy (for deployment), and CodeCommit (as a Git repository).
- Use Amazon ECR for container storage and Amazon ECS/EKS for orchestration.
- Example CodePipeline YAML file: version: 0.2

```
phases: build:  
commands: - docker  
build -t myapp . -  
docker tag  
myapp:latest  
123456789.dkr.ecr.us-  
east-1.amazonaws.com  
/myapp:latest
```

## 5. How do you handle rollbacks in AWS CodeDeploy? Scenario: A deployment

failed and you need to revert to a previous version.

## Answer:

- Use CodeDeploy's automatic rollback feature.
- Configure CloudWatch alarms to trigger rollbacks on failure.
- Example rollback strategy:

```
"rollbackConfiguration":  
{ "rollbackEnabled":  
true }
```

## 2. Monitoring &

### Logging

6. How do you set

up centralized

logging in AWS?

Scenario: You need to collect logs from multiple EC2 instances.

# Answer:

- Use Amazon CloudWatch Logs with the CloudWatch agent on EC2.
- Use AWS Kinesis Firehose to stream logs to S3 for analysis.
- Set up AWS OpenSearch (formerly Elasticsearch) for log visualization.

# 7. How do you monitor an AWS Lambda function?

**Scenario:** You need to track execution time and failures of a Lambda function.

**Answer:**

- Use AWS CloudWatch Logs for logs.
- Use AWS X-Ray for distributed tracing.

- Set up CloudWatch Alarms on error rate.

8. How do you set up alerts for high CPU utilization on an EC2 instance?

**Scenario:** Your application experiences CPU spikes.

**Answer:**

- Use CloudWatch

# Alarms with an SNS topic.

- Example alarm setup: {  
  "AlarmName":  
  "HighCPU",  
  "MetricName":  
  "CPUUtilization",  
  "Threshold": 80,  
  "ComparisonOperator":  
  "GreaterThanOrEqualToThreshold"  
}

## 3. Infrastructure

# as Code (IaC)

9. How would you  
manage multiple  
AWS environments  
(dev, staging,  
production)?

Scenario: You need  
different configurations  
for each environment.

Answer:

- Use Terraform

# Workspaces or separate CloudFormation stacks.

- Use **parameterized variables** in Terraform:

```
variable "env" {}  
resource  
"aws_s3_bucket"  
"example" { bucket =  
"myapp-${var.env}" }
```

## 10. How do you enforce security

# best practices in Terraform?

**Scenario:** You need to ensure IAM policies follow least privilege.

**Answer:**

- Use AWS IAM Policy Analyzer.
- Use AWS Config to audit Terraform-deployed resources.
- Example restricted IAM

```
policy: { "Effect":  
  "Allow", "Action":  
  "s3:GetObject",  
  "Resource":  
  "arn:aws:s3:::my-  
  bucket/*" }
```

## 4. Containers & Orchestration

# 11. How do you run a containerized application in AWS?

**Scenario:** You need to deploy Docker containers on AWS.

**Answer:**

- Use Amazon ECS with Fargate (serverless) or Amazon EKS (Kubernetes).
- Store images in Amazon ECR.
- Deploy using AWS App Runner or AWS Copilot

CLI.

# 12. How do you manage secrets in AWS for a Kubernetes application?

**Scenario:** You need to store API keys securely for EKS.

**Answer:**

- Use AWS Secrets

Manager and inject  
secrets using  
Kubernetes external  
secrets operator.

## 5. Security &

## Networking

13. How do you  
restrict SSH access  
to EC2 instances?

Scenario: You want to  
allow only specific IPs.

# Answer:

- Use Security Groups to allow SSH from trusted IPs only.

## 14. How do you implement multi-region disaster recovery in AWS?

Scenario: You need a backup site in another region.

# Answer:

- Use AWS Route 53 Failover Routing.
- Replicate data using AWS S3 Cross-Region Replication or RDS Multi-AZ.

15. How do you automate security patching in AWS?

Scenario: You need to ensure EC2 instances are

updated.

**Answer:**

- Use AWS Systems Manager Patch Manager to automate OS patching.

**6. Miscellaneous**

**16. How do you set up cost optimization in AWS?**

**Scenario:** Your AWS bill is increasing.

**Answer:**

- Use AWS Cost Explorer to analyze usage.
- Set up Auto Scaling Groups for EC2 instances.
- Use Spot Instances for non-critical workloads.

**17. How do you implement AWS**

# IAM best practices?

**Scenario:** You need to secure user access.

**Answer:**

- Enable MFA for all users.
- Use IAM roles instead of IAM users.
- Enforce least privilege access.

This list covers 17 key AWS DevOps scenarios.

Let me know if you want  
the remaining 13  
questions!

Here are the remaining 13  
AWS DevOps scenario-  
based interview  
questions and answers,  
covering more real-time  
use cases in automation,  
scalability, networking,  
security, and  
troubleshooting.

# 7. High Availability & Scalability

18. How do you handle high availability for an application running on EC2?

Scenario: Your application needs 99.99%

uptime.

## Answer:

- Deploy EC2 instances across multiple Availability Zones (AZs) in an Auto Scaling Group.
- Use an Elastic Load Balancer (ELB) to distribute traffic.
- Store data in Amazon RDS Multi-AZ or

# DynamoDB Global Tables.

- Example Auto Scaling policy: {  
  "AutoScalingGroupName": "my-app-asg",  
  "MinSize": 2, "MaxSize": 10 }

19. How do you  
scale an  
application

# dynamically in

## AWS?

**Scenario:** Your website traffic varies throughout the day.

**Answer:**

- Use Auto Scaling Groups with policies based on CPU, memory, or request count.
- Use AWS Lambda for event-driven workloads.

- Implement DynamoDB on-demand capacity mode for unpredictable traffic.

20. How would you manage a sudden traffic spike on your application?

Scenario: Your application gets a sudden surge in users.

# **Answer:**

- Use AWS Auto Scaling to scale EC2 instances dynamically.
- Enable AWS Global Accelerator to route traffic efficiently.
- Implement CloudFront caching for static content.

## **8. Networking & Security**

# 21. How do you securely connect an on-premises data center to AWS?

**Scenario:** Your company needs a hybrid cloud setup.

**Answer:**

- Use AWS Site-to-Site VPN or AWS Direct Connect for secure

connectivity.

- Use **Transit Gateway** to connect multiple VPCs and on-premises networks.

**22. How do you restrict access to an S3 bucket to only a specific VPC?**

**Scenario:** Your S3 bucket should not be publicly

accessible.

## Answer:

- Use VPC Endpoint for S3 and configure a bucket policy: { "Effect": "Deny", "Principal": "\*", "Action": "s3:\*", "Resource": "arn:aws:s3:::my-secure-bucket/\*", "Condition": { "StringNotEquals": { "aws:SourceVpc":

```
"vpc-12345678" } } }
```

# 23. How do you detect and respond to security threats in AWS?

**Scenario:** You need a real-time security monitoring solution.

**Answer:**

- Use AWS GuardDuty to detect anomalies.

- Enable AWS Security Hub for centralized security monitoring.
- Set up AWS WAF to block malicious traffic.

**24. How would you set up private access to an RDS database?**

**Scenario:** Your database should not be exposed to

the public internet.

## Answer:

- Deploy RDS inside a private subnet of a VPC.
- Use AWS Secrets Manager to store database credentials securely.
- Configure security groups to allow access only from application servers.

# 9. Serverless & Event-Driven Architectures

25. How do you process large log files using AWS Lambda?

Scenario: You receive daily log files that need processing.

# Answer:

- Use Amazon S3 event triggers to invoke AWS Lambda when a new log file is uploaded.
- Use AWS Step Functions for managing stateful workflows.

26. How do you implement event-driven automation

# in AWS?

**Scenario:** You need to trigger actions based on AWS service events.

**Answer:**

- Use Amazon EventBridge (CloudWatch Events) to trigger Lambda functions or SNS notifications when specific events occur.

# 27. How do you reduce cold start issues in AWS Lambda?

**Scenario:** Your Lambda function has high latency.

**Answer:**

- Use Provisioned Concurrency for always-ready Lambda instances.

- Reduce package size and optimize dependencies.
- Use Amazon CloudFront with Lambda@Edge for global caching.

## 10. AWS DevOps

### Troubleshooting

28. How do you debug a failing

# AWS CodeBuild job?

Scenario: Your CI/CD pipeline fails during the build phase.

Answer:

- Check AWS CodeBuild logs in CloudWatch Logs.
- Enable debug mode in the buildspec file:  
version: 0.2 phases:  
build: commands: -

```
echo "Debugging  
build..." - env - cat /var/  
log/codebuild.log
```

- Use AWS X-Ray to trace API calls.

**29. Your ECS  
service is stuck in  
"Pending" state.  
How do you  
troubleshoot?**

**Scenario:** Your containers

are not starting.

## Answer:

- Check if ECS Task IAM Role has the correct permissions.
- Verify if there are enough EC2 instances in the ECS cluster.
- Check CloudWatch Logs for container startup errors.

30. You deployed

# an application on AWS, but it is not accessible. How do you troubleshoot?

**Scenario:** Users cannot reach your web application.

**Answer:**

- Check security groups and network ACLs to ensure inbound rules

allow traffic.

- Verify Route 53 DNS configuration.
- Use VPC Flow Logs and AWS CloudTrail to inspect network traffic.

## Final Thoughts

These 30 AWS DevOps scenario-based questions cover real-world challenges that DevOps engineers face. Let me

know if you need further explanation on any topic!