

Riyaz Sayyad  
AWS Cloud Expert

# INTERMEDIATE



Interview

Q/A



Riyaz Sayyad  
AWS Cloud Expert

**Q-1**

## What is the role of an AWS Solution Architect?

**Ans**

AWS Solution Architects design and manage cloud infrastructure on AWS, ensuring optimal performance, security, and scalability. They work with developers and businesses to implement AWS best practices and align solutions with business needs. They also translate complex cloud concepts for both technical and non-technical stakeholders.

**Q-2**

## Explain how you would choose btw Amazon RDS, Amazon DynamoDB,& Amazon Redshift for a data-driven application.

**Ans**

- Amazon RDS: Best for structured, relational databases with SQL support and ACID compliance (e.g., MySQL, PostgreSQL).
- Amazon DynamoDB: Ideal for NoSQL applications requiring high scalability and low-latency performance (e.g., gaming leaderboards, IoT).
- Amazon Redshift: Suited for analytical workloads that require fast querying over large datasets (e.g., data warehouses, BI tools).



Riyaz Sayyad  
AWS Cloud Expert

Q-3

**What are some strategies for blue/green deployments on AWS?**

Ans

1. Using ALB with Two Target Groups
  - Route live traffic to the Blue (old) environment.
  - Deploy new version to the Green environment.
  - Switch traffic gradually using ALB listener rules.
2. AWS CodeDeploy for EC2 & ECS
  - Automate blue/green deployment with rollback capabilities.
  - Supports controlled traffic shifting.
3. Route 53 Weighted Routing
  - Distribute traffic across versions using weighted DNS records.

Q-4

**How would you manage and secure secrets for a CI/CD pipeline in AWS?**

Ans

- AWS Secrets Manager – Securely stores API keys and credentials.
- AWS Systems Manager Parameter Store – Manages sensitive parameters.
- IAM Roles & Policies – Enforce least privilege access.
- Encrypt Secrets – Use AWS KMS for encryption.



Riyaz Sayyad  
AWS Cloud Expert

Q-5

**What is your approach to handling continuous integration and deployment in AWS DevOps?**

Ans

**Continuous Integration (CI) Strategy**

1. Source Code Management – Store code in AWS CodeCommit, GitHub, or GitLab.
2. Automated Builds & Tests – Use AWS CodeBuild for compiling and testing.
3. Artifact Storage – Store build artifacts in Amazon S3 or AWS CodeArtifact.

**Continuous Deployment (CD) Strategy**

1. AWS CodePipeline – Automates CI/CD workflow.
2. AWS CodeDeploy – Deploys applications to EC2, ECS, Lambda.
3. Deployment Strategies – Use Blue/Green, Rolling, or Canary.

**Monitoring & Security**

- AWS CloudWatch & AWS X-Ray for logging & monitoring.
- IAM roles & AWS Secrets Manager for security.



**Message me and ask for your free roadmap today.**

Riyaz Sayyad  
AWS Cloud Expert

Q-6

Ans

## What are the key pillars of the AWS Well-Architected Framework?

- Operational Excellence: Automate operations, monitor performance, and improve continuously.
- Security: Implement IAM, encryption, and continuous threat monitoring.
- Reliability: Design for failure recovery, use backups, and ensure fault tolerance.
- Performance Efficiency: Optimize compute, storage, and database configurations for efficiency.
- Cost Optimization: Use cost-effective resources, auto-scaling, and rightsizing strategies.
- Sustainability: Minimize resource consumption and follow green cloud computing practices.



Riyaz Sayyad  
AWS Cloud Expert

Q-7

**What is Infrastructure as Code (IaC)? Describe in your own words.**

Ans

Infrastructure as Code (IaC) is the practice of automating the provisioning and management of IT infrastructure using declarative or imperative code instead of manual processes. It allows organizations to define, deploy, and manage infrastructure using machine-readable configuration files, ensuring consistency and scalability.

#### How It Works

- Infrastructure is defined in code (JSON, YAML, or HCL).
- Version-controlled using Git, Bitbucket, or AWS CodeCommit.
- Provisioned using tools like AWS CloudFormation, Terraform, or Pulumi.



Riyaz Sayyad  
AWS Cloud Expert

**Q-8**

**What is AWS CloudFormation, and how does it facilitate DevOps practices?**

**Ans**

AWS CloudFormation is an Infrastructure as Code (IaC) tool that automates resource provisioning using YAML/JSON templates.

**How It Helps DevOps**

- Automates infrastructure deployment
- Ensures consistency across environments
- Supports version control for infrastructure
- Integrates with AWS CodePipeline for automated deployments

**Q-9**

**What are the key security best practices for AWS EC2?**

**Ans**

- Use IAM roles instead of storing credentials on EC2
- Restrict SSH access using Security Groups
- Enable Amazon Inspector for vulnerability scans
- Encrypt EBS volumes using AWS KMS
- Use AWS Systems Manager for secure remote access



Riyaz Sayyad  
AWS Cloud Expert

**Q-10**

**Ans**

**Describe how you would use AWS services to implement a microservices architecture.**

- API-First Approach: Use Amazon API Gateway for secure service communication.
- Compute Choices: Deploy microservices using AWS Lambda (serverless) or ECS/EKS (containerized).
- Database Strategy: Choose DynamoDB for NoSQL, RDS for SQL-based services.
- Messaging & Event-Driven Architecture: Use SNS, SQS, and EventBridge for inter-service communication.
- Security & Authentication: Implement IAM, Cognito, and AWS Secrets Manager.
- Monitoring & Logging: Use AWS CloudWatch, X-Ray, and AWS Logging for visibility.
- CI/CD Pipeline: Automate deployments using AWS CodePipeline, CodeBuild, and CodeDeploy.



Riyaz Sayyad  
AWS Cloud Expert

**Q-11**

**How do you use AWS CodePipeline to automate a CI/CD pipeline for a multi-tier application?**

**Ans**

AWS CodePipeline is a fully managed CI/CD service that automates the build, test, and deployment phases of application development. For a multi-tier application, CodePipeline can manage deployment across different application layers (frontend, backend, and database) while integrating with AWS services like CodeCommit, CodeBuild, CodeDeploy, and CloudFormation.

#### Steps to Automate CI/CD for a Multi-Tier Application

1. **Source Stage:** Store code in CodeCommit/GitHub/S3, set auto triggers.
2. **Build Stage:** Use CodeBuild to compile, test, and create artifacts.
3. **Test Stage (Optional):** Deploy test environments, run security & integration tests.
4. **Deployment Stage:** Use CodeDeploy with rolling, blue/green, or canary strategies.
5. **Monitoring & Alerts:** Use CloudWatch, SNS for alerts, and X-Ray for debugging.



Riyaz Sayyad  
AWS Cloud Expert

Q-12

Ans

## What are the key security best practices for AWS EC2?

- IAM Roles & Policies: Use IAM for access control and enforce least privilege.
- Security Groups & Firewalls: Restrict access to trusted IPs and networks.
- Disable Password Logins: Use SSH key authentication instead of passwords.
- Enable Multi-Factor Authentication (MFA): Secure access with additional authentication layers.
- Regular Patching & Updates: Keep the OS and applications up to date.
- Encrypt Data: Use EBS volume encryption and TLS for data security.
- Monitor & Audit: Use AWS CloudTrail and GuardDuty for logging and monitoring threats.



Riyaz Sayyad  
AWS Cloud Expert

Q-13

Ans

## How do you use AWS Systems Manager in a Production Environment?

AWS Systems Manager automates infrastructure management, enhancing security, compliance, and efficiency in production environments.

It is commonly used for:

- Patch Management: Automates OS and application patching for EC2, on-premises, and hybrid systems.
- Remote Command Execution: Securely runs scripts and commands across multiple instances using Run Command.
- Inventory Collection: Tracks system configurations, installed applications, and compliance status.
- Secure Parameter Storage: Stores and manages secrets and configuration values with Parameter Store.
- AWS Integration: Works with EC2, RDS, Lambda, and AWS Organizations for centralized control.
- Monitoring & Compliance: Improves visibility using OpsCenter, CloudWatch, and AWS Config for governance.



Riyaz Sayyad  
AWS Cloud Expert

**Q-14**

**How does Amazon ECS benefit AWS DevOps?**

**Ans**

- Fully managed orchestration – No need to manage Kubernetes clusters.
- Deep AWS integration – Seamlessly works with IAM, CloudWatch, and ALB.
- Auto Scaling support – Dynamically adjusts containers.
- Supports AWS Fargate – Runs containers without managing EC2 instances.

**Q-15**

**Why might ECS be preferred over Kubernetes?**

**Ans**

**Reasons to Prefer ECS**

- Simplified Management – No control plane or master nodes.
- Lower Cost – No additional infrastructure overhead.
- AWS-Native Integration – Built-in compatibility with IAM, CloudWatch, VPC, and Fargate
- Serverless Containers – Fargate removes EC2 management



Riyaz Sayyad  
AWS Cloud Expert

**Q-16**

**What is the relationship between AWS Glue and AWS Lake Formation?**

**Ans**

- AWS Glue: Primarily an ETL (Extract, Transform, Load) service for data integration.
- AWS Lake Formation: Builds on Glue, adding security, access controls, and governance for managing data lakes.
- Integration: Lake Formation uses Glue's ETL engine, Data Catalog, and schema inference

**Q-17**

**How do you optimize AWS costs for a high-traffic web application?**

**Ans**

- Use Reserved & Spot Instances: Optimize compute costs based on predictable vs. flexible workloads.
- Enable Auto-Scaling: Dynamically scale resources to match demand.
- Leverage Caching: Use CloudFront and ElastiCache to reduce backend loads.
- Optimize Storage: Implement S3 lifecycle policies and Glacier for archiving.
- Monitor & Analyze Costs: Use AWS Cost Explorer, Budgets, and Trusted Advisor for cost management.





Riyaz Sayyad  
AWS Cloud Expert

Q-18

Ans

**What considerations would you take into account when migrating an existing on-premises application to AWS?**

- Assess the Current Setup: Identify dependencies, database size, and performance requirements.
- Choose the Right AWS Services: Use EC2 for compute, RDS for databases, and S3 for storage.
- Migration Strategy: Lift-and-shift, re-platform, or full re-architecting.
- Data Migration Tools: Use AWS DMS, Snowball, or S3 Transfer.
- Security & Compliance: Encrypt data, implement IAM, and follow AWS security best practices.
- Testing & Validation: Run pilot migrations, test application performance, and optimize configurations.
- Cost Optimization: Use reserved instances, auto-scaling, and cost monitoring tools.

Ready for your 10X cloud career boost?



Riyaz Sayyad  
AWS Cloud Expert

Q-19

Ans

## What key factors should be considered in designing a deployment solution on AWS?

Designing an efficient AWS deployment requires selecting the right services based on your application's compute, storage, and database needs.

### Key Factors in Designing a Deployment Solution on AWS

- Provisioning: Use Infrastructure as Code (IaC) with AWS CloudFormation or Terraform.
- Configuring: Automate setup using AWS Systems Manager or Ansible.
- Deploying: Choose strategies like rolling, blue/green, or canary with AWS CodeDeploy.
- Scaling: Implement Auto Scaling Groups (ASG) and Elastic Load Balancing (ELB).
- Monitoring: Use CloudWatch, AWS X-Ray, and SNS for alerts and insights.





Riyaz Sayyad  
AWS Cloud Expert

Q-20

Ans

## How do you ensure multi-region redundancy in an AWS architecture?

- Deploy Resources Across Regions: Use multi-region EC2, RDS, and S3 for resilience.
- Route 53 for DNS Failover: Implement geo-routing and health checks.
- S3 Cross-Region Replication: Ensure data availability across multiple regions.
- Active-Active or Active-Passive Setup: Choose failover strategies based on workload.
- Monitor & Automate Recovery: Use CloudWatch, AWS Global Accelerator, and auto-healing mechanisms.



Riyaz Sayyad  
AWS Cloud Expert

Q-21

**What are the strategies to create a highly available and fault-tolerant AWS architecture for critical web applications?**

Ans

- Redundancy & Load Balancing: Distribute traffic using ELB across multiple AZs.
- Auto-Scaling: Scale resources dynamically based on demand.
- Fault Isolation & Disaster Recovery: Use backups, snapshots, and multi-region replication.
- Graceful Degradation: Ensure the system remains functional even during failures.
- Continuous Monitoring & Testing: Use AWS CloudWatch, X-Ray, and fault injection testing.

Message me and ask for your free roadmap today.



# Was it helpful?

follow me for more Cloud related content



Like



Comment



Share



Save

**For ACMP Check out: <https://needforcloud.com>**