

Q1. What is Service Bus and what is its primary purpose in its cloud ecosystem?

A: Service Bus is a managed cloud service that handles a specific infrastructure concern (compute, storage, messaging, AI). Using it shifts operational burden to the cloud provider while you focus on application logic.

Q2. How does IAM work with Service Bus?

A: Access to Service Bus is controlled via IAM roles and policies. Attach the minimum required permissions to a service account or role. Avoid long-lived access keys; use instance roles or Workload Identity Federation instead.

Q3. How do you monitor Service Bus in production?

A: Service Bus emits metrics to the cloud provider's monitoring service (CloudWatch, Cloud Monitoring, Azure Monitor). Set alerts on key metrics (error rate, latency, capacity). Use distributed tracing to correlate across services.

Q4. How do you scale Service Bus to handle variable load?

A: Service Bus supports auto-scaling policies based on CPU, queue depth, or custom metrics. Set minimum and maximum capacity. Horizontal scaling (more instances) is generally preferred over vertical for cloud workloads.

Q5. How do you secure Service Bus in production?

A: Place Service Bus in a private subnet with no public endpoint where possible. Use VPC security groups or firewall rules to allow only required traffic. Encrypt data at rest and in transit. Rotate credentials regularly.

Q6. How do you manage configuration and secrets for Service Bus?

A: Store sensitive values in a managed secret store (AWS Secrets Manager, GCP Secret Manager, Azure Key Vault). Inject secrets as environment variables at runtime, never bake them into container images or source code.

Q7. What are the main cost drivers for Service Bus and how do you optimize them?

A: Cost in Service Bus typically comes from compute hours, data transfer, storage, and request counts. Right-size instances, use reserved capacity for steady-state workloads, and enable lifecycle policies to expire old data.

Q8. How do you implement high availability with Service Bus?

A: Deploy Service Bus across multiple availability zones. Use managed replicas or active-active configurations. Implement health checks and automatic failover. Test resilience regularly with chaos engineering or failover drills.

Q9. How does Service Bus integrate with a CI/CD pipeline?

A: CI/CD pipelines use the cloud CLI or SDK to deploy updates to Service Bus after tests pass. Infrastructure-as-code (Terraform, CDK) defines Service Bus configuration as version-controlled code deployed via the pipeline.

Q10. What are common pitfalls when first adopting Service Bus?

A: Common mistakes include misconfigured IAM policies (too permissive), no cost alerting, deploying only in one availability zone, and missing backups. Read the service SLA carefully to understand what the cloud provider guarantees.