AWS Solution Architect Associate

Version : C03

Domain 4

Task 2

Design cost-optimized compute solutions

AWS Cost Management Services and Features

- ➔ **Billing and Payments**
  - ◆ View and pay AWS bills.
  - ◆ Set up and manage payment methods and preferences
- ➔ **Cost Analysis**
  - ◆ AWS Cost & Usage Report: Provides detailed cost and usage data.
  - ◆ AWS Cost Explorer: Analyzes cost and usage trends to identify cost drivers
- ➔ **Cost Organization**
  - ◆ Tagging: Organize resources by project, team, or cost center.
  - ◆ Cost Allocation Reports: Track and allocate costs to various departments or projects
- ➔ **Budgeting and Planning**
  - ◆ AWS Budgets: Create custom budgets, set alerts for threshold breaches and monitor budget
- ➔ **Savings and Commitments**
  - ◆ AWS Savings Plans and Reserved Instances: Save on predictable workloads by committing to usage levels over a set period
- ➔ **Consolidated Billing**
  - ◆ Consolidated billing for multiple AWS accounts, providing a single invoice to simplify cost management
- ➔ **Tools and Dashboards**
  - ◆ Billing Dashboard: Centralized place to view bills and manage payments
  - ◆ Bills Page: Detailed information about monthly bills and charges

AWS Instance Purchasing Options

Options	Description
On-Demand	Pay by the second, for the instances that you launch
Reserved Instances	Reduce your Amazon EC2 costs by making a commitment to a consistent instance configuration including instance type and Region for a term of 1 or 3 years
Spot Instances	Request unused EC2 instances which can reduce your Amazon EC2 costs significantly
Savings Plan	Reduce your Amazon EC2 costs by making a commitment to a consistent amount of usage in USD per hour, for a term of 1 or 3 years
Dedicated Hosts	Pay for a physical host that is fully dedicated to running your instances and bring your existing per-socket, per-core or per-VM software licenses to reduce costs
Dedicated instances	Pay by the hour for instances that run on single-tenant hardware
Capacity Reservations	Reserve capacity for your EC2 instances in a specific Availability Zone

Hybrid Compute Options

AWS offers a variety of hybrid compute options that integrate on-premises environments with AWS cloud services. These options include :

- ➔ **AWS Outposts** : Extends AWS infrastructure, services, APIs, and tools to any data center, co-location space or on-premises facility for a truly consistent hybrid experience
- ➔ **VMware Cloud on AWS** : Allows users to run VMware workloads on AWS with seamless integration and operational consistency
- ➔ **AWS Local Zones** : Extends AWS compute, storage, database, and other services closer to large populations and IT centers to support latency-sensitive applications
- ➔ **AWS Wavelength** : Integrates AWS compute and storage services within telecommunications providers' data centers at the edge of the 5G network to minimize latency for mobile and connected device applications
- ➔ **AWS Storage Gateway** : Hybrid storage service that integrates on-premises environments with cloud storage, including File Gateway, Tape Gateway and Volume Gateway for different use cases

Optimization of Compute Utilization

Optimizing compute utilization in AWS can be achieved through several strategies involving containers, serverless computing, and microservices :

- ➔ **AWS Compute Optimizer**
- ➔ **Containers**
  - ◆ Using containers such as those managed by AWS Fargate or Amazon ECS allows for efficient resource utilization by packaging applications and their dependencies together
  - ◆ This approach ensures consistent performance and scalability while isolating applications for better security and resource management
- ➔ **Serverless Computing**
  - ◆ AWS Lambda provides a serverless computing model where code execution is fully managed by AWS
  - ◆ This eliminates the need to provision and manage servers, automatically scales based on the load and you only pay for the compute time you consume
- ➔ **Microservices Architecture**
  - ◆ Decomposing applications into smaller, independent services that can be deployed and scaled individually improves efficiency
  - ◆ AWS services like Amazon ECS and AWS Lambda are well-suited for microservices, enabling better resource allocation and fault isolation

Optimization of Compute Utilization

AWS Compute Optimizer is a service designed to help optimize your AWS compute resources. It analyzes the configuration and utilization metrics of your AWS resources and provides recommendations for improving efficiency and reducing costs. Key aspects :

- ➔ **Rightsizing Recommendations**
  - ◆ AWS Compute Optimizer uses ML to analyze historical utilization data and recommend optimal AWS resource configurations, such as EC2 instance types, Auto Scaling groups, EBS volumes, and Lambda function memory sizes.
  - ◆ Help reduce costs by up to 25%
- ➔ **Performance Analysis**
  - ◆ Service examines CPU, memory and other utilization metrics to predict future performance needs
  - ◆ By identifying underutilized resources, it helps in making informed decisions about downsizing or consolidating resources without compromising performance
- ➔ **Actionable Insights**
  - ◆ AWS Compute Optimizer delivers intuitive recommendations that are easy to implement, ensuring that you can quickly adjust your resources to achieve optimal efficiency
- ➔ **Cost Reduction and Efficiency**
  - ◆ By following the recommendations, users can achieve significant cost savings and improve the overall efficiency of their AWS environment

Scaling Strategies

AWS offers several strategies for auto scaling to efficiently manage resources and optimize performance. Key strategies :

➔ **Dynamic Scaling**
  - ◆ This adjusts the number of instances based on real-time demand
  - ◆ Rules can be set to add or remove EC2 instances based on CPU utilization or network traffic

➔ **Predictive Scaling**
  - ◆ Uses ML to predict future traffic and scales resources in advance to handle the anticipated load, optimizing both cost and performance

➔ **Scheduled Scaling**
  - ◆ Allows to scale resources based on a schedule
  - ◆ For example - Capacity can be increased and decreases during known peak\off-peak times

➔ **Manual Scaling**
  - ◆ Manually adjust the number of instances based on requirements, offering full control over the scaling process

Different classes of workloads

Determining the required availability for different classes of workloads involves understanding the business needs, criticality of the workloads and the impact of downtime.Some guidelines -

Production Workloads

- ➔ **High Availability** : Production workloads, critical to business operations, typically require HA.This often means deploying resources across multiple AZs to ensure redundancy and fault tolerance
- ➔ **Multi-AZ Deployments** : Implementing multi-AZ deployments can protect against failures in a single AZ, improving the resilience of the application
- ➔ **Service Level Agreements** : Clear SLAs for uptime\architecture meets these requirements

Non-Production Workloads

- ➔ **Lower Availability** : Non-production workloads such as development, testing or staging environments generally have lower availability requirements. Single AZ deployments may be sufficient for these environments, reducing costs while still providing necessary functionality
- ➔ **Cost Optimization** : Cost-effective solutions that meet the necessary performance criteria

General Considerations

- ➔ **Workload Assessment** : Regular criticality assessment of workloads and adjustment of the availability strategy accordingly. Decision-making process should be driven by business needs & workload criticality
- ➔ **Region Selection**: Appropriate AWS region based on latency, regulatory requirements and proximity

AWS Solution Architect Associate

Version : C03

Domain 4

Task 2

The END