Module 1 - Cloud Concepts Overview

Slides

Objectives / Topics

- Define different types of cloud computing models
- Describe six advantages of cloud computing
- Recognize the main AWS service categories and core services
- Review the AWS Cloud Adoption Framework (AWS CAF)

Labs / Activities

- Knowledge Check

Section 1: Intro to Cloud Computing

Cloud Computing: The on-demand delivery of compute power, database, storage, applications, and other IT resources via the internet with pay-as-you-go pricing. Cloud computing enables you to stop thinking of infrastructure as hardware, and instead think of (and use) it as software.

Traditional Computing Model

- Infrastructure as hardware
- Hardware solutions: Require space, staff, physical security, planning, capital expenditure
- Have a long hardware procurement cycle
- Require you to provision capacity by guessing theoretical maximum peaks

Cloud Computing Model

- Infrastructure as software
- Software solutions:
- Are flexible
- Can change more quickly, easily, and cost-effectively than hardware solutions
- Eliminate the undifferentiated heavy-lifting tasks

Cloud service models vary on how much control you have over IT resources.

- Infrastructure as a Service (IaaS) - Most control
- Platform as a Service (PaaS)
- Software as a Service (SaaS) - Least control

Cloud computing deployment models

1. Cloud
2. Hybrid
3. On-premse (Private Cloud)

Cloud computing can do almost anything the traditional IT can do.

Section 2: Advantages of Cloud Computing

- Pay only for the resources you consume (variable cost vs upfront capital expenditure)
- Economies of scale achieved by aggregate of all users
- Scaling on demand
- Speed and flexibility - changes are software level, not hardware like traditional computing

- Lower overhead due to not maintaining hardware and data centers
- Data centers are global, like a company's customer base

Section 3: Introduction to Amazon Web Services

Web Service: Any piece of software that makes itself available over the internet and uses a standardized formatâ€”such as Extensible Markup Language (XML) or JavaScript Object Notation (JSON) â€” for the request and the response of an application programming interface (API) interaction.

What is AWS?

- AWS is a secure cloud platform that offers a broad set of global cloud-based products called services that are designed to work together.
- There are many categories of AWS services, and each category has many services to choose from.
- Choose a service based on your business goals and technology requirements.
- There are three ways to interact with AWS services:
- AWS Management Console - Graphical interface
- Command Line Interface (CLI) - Access via discrete commands or scripts
- Software Development Kits (SDK) - Access directly from code

Section 4: The AWS Cloud Adoption Framework (AWS CAF)

- AWS CAF provides guidance and best practices to help organizations build a comprehensive approach to cloud computing across the organization and throughout the IT lifecycle to accelerate successful cloud adoption.
- AWS CAF is organized into six perspectives and perspectives consist of sets of capabilities.

Focused on Business Capabilities 1. Business - IT is aligned with business needs - IT Finance - IT Strategy - Benefits Realization - Business Risk Management 2. People - training, staffing, and organizational changes - Resource Management - Incentive Management - Career Management - Training Management - Organizational Change Management 3. Governance - skills and processes align IT and business strategies and goals - Portfolio Management - Program Project Management - Business Performance Measurement - License Management

Focused on Technical Capabilities

1. Platform - describe the architecture of the target state environment in detail
    - Compute Provisioning
    - Network Provisioning
    - Storage Provisioning
    - Database Provisioning
    - Systems and Solution Architechture
    - Application Development
2. Security - the organization meets its security objectives
    - Identity and Access Management
    - Detective Control
    - Infrastructure Security
    - Data Protection
    - Incident Response
3. Operations - define how daily, quarterly, and yearly business will be conducted
    - Service Monitoring
    - Application Performance Monitoring
    - Resource Inventory Management
    - Release Management / Change Management
    - Reporting and Analytics
    - Business Continuity / Disaster Recovery
    - IT Service Catalog

Knowledge Check

Module 2 - Cloud Economics and Billing

Slides

Objectives / Topics

- Explain the AWS pricing philosophy
- Recognize fundamental pricing characteristics
- Indicate the elements of total cost of ownership
- Discuss the results of the Simple Monthly Calculator
- Identify how to set up an organizational structure that simplifies billing and account visibility to review cost data.
- Identify the functionality in the AWS Billing Dashboard
- Describe how to use AWS Bills, AWS Cost Explorer, AWS Budgets, and AWS Cost and Usage Reports
- Identify the various AWS technical support plans and features

Labs / Activities

- Knowledge Check
- Cost Calculator Activity -- Simple Monthly Calculator
- Support Plans Scavenger Hunt

Section 1: Fundamentals of Pricing

Three Fundamental Cost Drivers with AWS

1. Compute - charged by use time, varies by instance
2. Storage - charged per GB
3. Data Transfer - outbound transfers are aggregated and charged per GB, inbound transfers and data transfers between services in the same AWS Region typically have no charge

Paying for AWS

- Pay for what you use
- Reserve and save up to 75% versus On-Demand
- All Upfront Reserved Instance (AURI) -> Large Discount
- Partial Upfront Reserved Instance (PURI) -> Lower Discount
- No Upfront Payments Reserved Instance (NURI) -> Smallest Discount
- Scale and save as usage increases
- Tiered pricing for services like S3, EBS, EFS
- Save as AWS grows
- Custom Pricing
- Meet varying needs through custom pricing.
- Available for high-volume projects with unique requirements.

The services below are free but there might be charges associated with other AWS services that are used alongside these services.

- Amazon VPC
- Elastic Beanstalk
- Auto Scaling
- AWS CloudFormation
- AWS Identity and Access Management

Section 2: Total Cost of Ownership

Total Cost of Ownership (TCO): The financial estimate to help identify direct and indirect costs of a system

- Compare the costs of running an entire infrastructure environment or specific workload on-premises versus on AWS
- Budget and build the business case for moving to the cloud

TCO Considerations

1. Server Costs
2. Hardware: Server, rack chassis power distribution units (PDUs), top-of-rack (TOR) switches, and maintenance
3. Software: Operating system (OS), virtualization licenses, and maintenance
4. Facilities: Space, power, and cooling
5. Storage Costs
6. Hardware: Storage disks, storage area network (SAN) or Fibre Channel (FC) switches
7. Storage administration costs
8. Facilities: Space, power, and cooling
9. Network Costs
10. Network Hardware: Local area network (LAN) switches, load balancer bandwidth costs
11. Network administration costs
12. Facilities: Space, power, and cooling
13. IT Labor Costs
14. Server administration costs

Cost Calculator Activity -- Simple Monthly Calculator

Section 3: Billing

AWS Organizations

AWS Organizations: An account management service that enables you to consolidate multiple AWS accounts into an organization that you create and centrally manage. AWS Organizations includes account management and consolidated billing capabilities that enable you to better meet the budgetary, security, and compliance needs of a business.

Key Features and Benefits

- Policy based account management
- Group based account management
- Application programming interfaces (APIs) that automate account management
- Consolidated billing

Security

- Control access with AWS Identity and Access Management (IAM)
- IAM policiesenable you to allow or deny access to AWS services for users, groups, and roles.
- Service control policies (SCPs) enable you to allow or deny access to AWS services for individuals or group accounts in an organizational unit (OU).

Setup

1. Create organization
2. Create organizational units
3. Create service control policies
4. Test restrictions

Accessing AWS Organizations

- AWS Management Console
- AWS Command Line Interface (AWS CLI) tools
- Software Development Kits (SDKs)
- HTTPS Query Application Programming Interfaces (API)

AWS Billing and Cost Management

AWS Billing and Cost Management: The service that you use to pay your AWS bill, monitor your usage, and analyze and control your costs.

Tools

- **AWS Budgets:** It gives you the ability to set custom budgets that alert you when your costs or usage exceed (or are forecasted to exceed) your budgeted amount. You can also use it to set reservation utilization or coverage targets and receive alerts when your utilization drops below the threshold you define.
- **AWS Cost and Usage Report:** Tracks your AWS usage and provides estimated charges associated with your account.
- **AWS Cost Explorer:** Visualize, understand, and manage AWS costs and usage over time.

Section 4: Technical Support

AWS Support offers a range of plans that provide access to tools and expertise that support the success and operational health of your AWS solutions. All support plans provide 24/7 access to customer service, AWS documentation, whitepapers, and support forums.

- Proactive Guidance: Technical Account Manager (TAM)
- Best Practices: AWS Trusted Advisor
- Account Assistance: AWS Support Concierge

Support Plans

- **Basic Support:** Resource Center access, Service Health Dashboard, product FAQs, discussion forums, and support for health checks
- **Developer Support:** Support for early development on AWS
- **Business Support:** Customers that run production workloads
- **Enterprise Support:** Customers that run business and mission-critical workloads

Support response times vary based on plan and case severity. Basic offers no case support, all other support ranges from 24 hours to 15 minutes or less.

Support Scavenger Hunt

Knowledge Check

Module 3 - AWS Global Infrastructure Overview

Slides

Objectives / Topics

- Identify the difference between AWS Regions, Availability Zones, and edge locations
- Identify AWS service and service categories

Labs / Activities

- Knowledge Check
- Lab: AWS Management Console (Use Sandbox) -- Labs Instructions
- AWS Global Infrastructure Activity

Section 1: AWS Global Infrastructure

The AWS Global Infrastructure is designed and built to deliver a flexible, reliable, scalable, and secure cloud computing environment with high-quality global network performance.

AWS Regions

An AWS Region is a geographical area.

- Data replication across Regions is controlled by you.
- Communication between Regions uses AWS backbone network infrastructure.
- Each Region provides full redundancy and connectivity to the network.
- A Region typically consists of two or more Availability Zones

When selecting a region consider the following:

- Laws - Data governance and legal requirements
- Proximity - Select regions close to your customers for reduced latency
- Availability - Some services are region locked
- Cost - Cost varies by region

Each Availability Zone is a fully isolated partition of the AWS infrastructure. There are currently 69 Availability Zones worldwide.

- Availability Zones consist of discrete data centers
- They are interconnected with other Availability Zones by using high-speed private networking
- They are designed for fault isolation
- You choose your Availability Zones but AWS recommends replicating data and resources across Availability Zones for resiliency.

AWS Data Centers

- AWS data centers are designed for security. Each data center has redundant power, networking, and connectivity, and is housed in a separate facility.
- Data centers are where the data resides and data processing occurs. A data center typically has 50,000 to 80,000 physical servers

AWS provides a global network of 187 Points of Presence locations:

- Consists of 176 edge locations - where end users access services located at AWS
- 11 Regional edge caches - cache copies of your infrequent content close to your users
- Used with **Amazon CloudFront** - a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

AWS Infrastructure Features:

1. Elasticity and scalability - dynamically adapts to capacity and growth needs
2. Fault-tolerance - Continues operating properly in the presence of a failure due to built-in redundancy of components
3. High availability - High operational performance with minimized downtime and no human intervention

Section 2: AWS Services and Service Category Overview

Foundation Services

- Compute
- Networking
- Storage

Service Categories

23 different product or service categories, and each category consists of one or more services. The course covers the most common categories and the ones likely to be on the foundations exam.

AWS Storage Services

- Amazon Simple Storage Services (S3) - Object storage service that offers industry-leading scalability, data availability, security, and performance.
- Amazon Elastic Block Storage (EBS) - An easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale.
- Amazon Elastic File System (EFS) - A simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources.
- Amazon Simple Storage Service Glacier - A secure, durable, and extremely low-cost Amazon S3 cloud storage classes for data archiving and long-term backup.

AWS Compute Services

- Amazon EC2 - A web service that provides secure, resizable compute capacity in the cloud.
- Amazon EC2 Auto Scaling - Helps to maintain application availability and allows you to automatically add or remove EC2 instances according to conditions you define.
- Amazon Elastic Container Services (ECS) - A fully managed container orchestration service.
- Amazon EC2 Container Registry (ECR) - A fully-managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images.
- AWS Elastic Beanstalk - An easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
- AWS Lambda - Lets you run code without provisioning or managing servers.
- Amazon Elastic Kubernetes Services (EKS) - A fully managed Kubernetes service.
- AWS Fargate - A serverless compute engine for containers that works with both Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS).

AWS Database Services

- Amazon Relational Database Service (RDS) - Makes it easy to set up, operate, and scale a relational database in the cloud.
- Amazon Aurora - A MySQL and PostgreSQL compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases.
- Amazon Redshift - A fully managed, petabyte-scale data warehouse service in the cloud.
- Amazon DynamoDB - A key value and document database that delivers single-digit millisecond performance at any scale.

Networking and Content Delivery Services

- Amazon VPC - Lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define.
- Elastic Load Balancing - Automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions.
- Amazon CloudFront - A fast content delivery network (CDN) service that securely delivers data, videos, applications, and

APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

- AWS Transit Gateway - A service that enables customers to connect their Amazon Virtual Private Clouds (VPCs) and their on-premises networks to a single gateway.
- Amazon Route 53 - A highly available and scalable cloud Domain Name System (DNS) web service.
- AWS Direct Connect - A cloud service solution that makes it easy to establish a dedicated network connection from your premises to AWS.
- AWS VPN - Lets you establish a secure and private encrypted tunnel from your network or device to the AWS global network.

Security, Identity, and Compliance Services

- AWS Identity and Access Management (IAM) - Enables you to manage access to AWS services and resources securely. You can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources. IAM is a feature of your AWS account offered at no additional charge.
- AWS Organizations - helps you centrally govern your environment as you grow and scale your workloads on AWS.
- Amazon Cognito - Lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily.
- AWS Artifact - A central resource for compliance related information that matters to you.
- AWS Key Management Service (KMS) - Makes it easy for you to create and manage cryptographic keys and control their use across a wide range of AWS services and in your applications.
- AWS Shield - A managed Distributed Denial of Service (DDoS) protection service that safeguards applications running on AWS.

Cost Management Services

- AWS Cost and Usage Report - contains the most comprehensive set of AWS cost and usage data available, including additional metadata about AWS services, pricing, and reservations (e.g., Amazon EC2 Reserved Instances (RIs)).
- AWS Budgets - gives you the ability to set custom budgets that alert you when your costs or usage exceed (or are forecasted to exceed) your budgeted amount.
- AWS Cost Explorer - an easy-to-use interface that lets you visualize, understand, and manage your AWS costs and usage over time.

AWS Management and Governance Services

- AWS Management Console - A secure, easy-to-access, web-based portal for AWS.
- AWS Config - A service that enables you to assess, audit, and evaluate the configurations of your AWS resources.
- Amazon CloudWatch - A monitoring and observability service that provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. You can use it to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights.
- AWS Auto Scaling - monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost.
- AWS Command Line Interface (CLI) - a unified tool to manage your AWS services. With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.
- AWS Trusted Advisor - An online tool that provides you real time guidance to help you provision your resources following AWS best practices.
- AWS Well-Architected Tool - Helps you review the state of your workloads and compares them to the latest AWS architectural best practices.
- AWS CloudTrail - A service that enables governance, compliance, operational auditing, and risk auditing of your AWS account.

Knowledge Check

Lab: AWS Management Console (Use Sandbox) -- Labs Instructions

AWS Global Infrastructure Activity

Module 4 - AWS Cloud Security

Slides

Objectives / Topics

- Recognize the shared responsibility model
- Identify the responsibility of the customer and AWS
- Recognize IAM users, groups, and roles
- Describe different types of security credentials in IAM
- Identify the steps to securing a new AWS accountâ€¢Explore IAM users and groups
- Recognize how to secure AWS data
- Recognize AWS compliance programs

Labs / Activities

- Knowledge Check
- Lab: Introduction to IAM --- Lab Instructions

Section 1: AWS Shared Responsibility Model

AWS security is divided by part of the cloud: customers are responsible for security **in** the cloud, AWS is responsible for security **of** the cloud.

Customer Security

- Amazon Elastic Compute Cloud (Amazon EC2) instance operating system - Including patching, maintenance
- Applications - Passwords, role-based access, etc.
- Security group configuration
- OS or host-based firewalls - Including intrusion detection or prevention systems
- Network configurations
- Account management - Login and permission settings for each user

AWS Security

- Physical security of data centers - Controlled, need-based access
- Hardware and software infrastructure - Storage decommissioning, host operating system (OS) access logging, and auditing
- Network infrastructure - Intrusion detection
- Virtualization infrastructure - Instance isolation

Service Characteristics and Security

Infrastructure as a service (IaaS)

- Customer has more flexibility over configuring networking and storage settings
- Customer is responsible for managing more aspects of the security
- Customer configures the access controls

Platform as a service (PaaS)

- Customer does not need to manage the underlying infrastructure
- AWS handles the operating system, database patching, firewall configuration, and disaster recovery
- Customer can focus on managing code or data

Software as a service (SaaS)

- Software is centrally hosted
- Licensed on a subscription model or pay-as-you-go basis.
- Services are typically accessed via web browser, mobile app, or application programming interface (API)

- Customers do not need to manage the infrastructure that supports the service

Section 2: Identity and Access Management (IAM)

AWS Identity and Access Management (IAM) is a web service that enables Amazon Web Services (AWS) customers to manage users and user permissions in AWS. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which AWS resources users can access.

AWS Identity and Access Management (IAM) can be used to:

- **Manage IAM Users and their access:** You can create Users and assign them individual security credentials (access keys, passwords, and multi-factor authentication devices). You can manage permissions to control which operations a User can perform.
- **Manage IAM Roles and their permissions:** An IAM Role is similar to a User, in that it is an AWS identity with permission policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a Role is intended to be assumableby anyone who needs it.
- **Manage federated users and their permissions:** You can enable identity federationto allow existing users in your enterprise to access the AWS Management Console, to call AWS APIs and to access resources, without the need to create an IAM User for each identity.

Essential Components

IAM User: A person or application that can authenticate with an AWS account. When you define an IAM user, you select what types of access the user is permitted to use.

- Programmatic Access
- Authenticate using Access Key ID and Secret Access Key
- Provides AWS CLI and AWS SDK access
- AWS Management Console Access
- Authenticate using Account ID or Alias, or Username and Password
- Can enable Multi-Factor Authentication (MFA)

IAM Group: A collection of IAM users that are granted identical authorization.

- A user can belong to multiple groups
- There is no default group
- Groups cannot be nested

IAM Policy: The document that defines which resources can be accessed and the level of access to each resource.

- Permissions determine which resources and operations are allowed.
- All permissions are implicitly denied by default.
- If something is explicitly denied, it is never allowed.
- Two Types of Policies:
- Identity Based
  - Attach a policy to any IAM entity - user, group, or role
  - Policies specify actions that may or may not be performed by the entity
  - Policies and entities have a many-to-many relationship
  - When the policy is updated, the changes to the policy are immediately apply against all Users and Groups that are attached to the policy.
- Resource Based
  - Attached to a resource (such as an S3 bucket)
  - Specifies who has access to the resource and what actions they can perform on it
  - The policies are inline only, not managed. An inline policy is assigned to just one User or Group. Inline Policies are typically used to apply permissions for one-off situations.
  - Resource-based policies are supported only by some AWS services

IAM Role: Useful mechanism to grant a set of permissions for making AWS service requests.

- Similar to an IAM user: Attach permissions policies to it
- Different from an IAM user: Not uniquely associated with one person, intended to be assumable by a person, application, or service
- Role provides temporary security credentials

Principle of Least Privilege: The practice of limiting access rights for users to the bare minimum permissions they need to perform their work. Under POLP, users are granted permission to read, write or execute only the files or resources they need to do their jobs

Section 3: Securing a New AWS Account

Best practice: Do not use the AWS account root user except when necessary. Access to the account root user requires logging in with the email address and password that you used to create the account.

1. Stop using the account root user as soon as possible
2. While you are logged in as the account root user, create an IAM user for yourself. Save the access keys if needed.
3. Create an IAM group, give it full administrator permissions, and add the IAM user to the group.
4. Disable and remove your account root user access keys, if they exist.
5. Enable a password policy for users.
6. Sign in with your new IAM user credentials.
7. Enable multi-factor authentication (MFA)
8. Use AWS CloudTrail
9. CloudTrail tracks user activity on your account. It logs all API requests to resources in all supported services your account.
10. Basic AWS CloudTrail event history is enabled by defaultand is free. It contains all management event data for the last 90 days of account activity (this can be extended beyon 90 days)
11. Enable a billing report, such as the *AWS Cost and Usage Report*

Lab: Introduction to IAM --- Lab Instructions

Section 4: Securing Accounts

AWS Organizations

Organizations enable you to consolidate multiple AWS accounts so that you centrally manage them.

Security Features of AWS Organizations:

- Group AWS accounts into organizational units (OUs) and attach different access policies to each OU.
- Integration and support for IAM. Permissions to a user are the intersection of what is allowed by AWS Organizations and what is granted by IAM in that account.
- Use service control policies to establish control over the AWS services and API actions that each AWS account can access

Service Control Policies

- Service control policies (SCPs) offer centralized control over accounts. Limit permissions that are available in an account that is part of an organization.
- Ensures that accounts comply with access control guidelines.
- SCPs are similar to IAM permissions policies â€“ They use similar syntax, but an SCP never grants permissions. Instead, SCPs specify the maximum permissions for an organization.

Key Management Service

- Enables you to create and manage encryption keys
- Enables you to control the use of encryption across AWS services and in your applications.
- Integrates with AWS CloudTrail to log all key usage.
- Uses hardware security modules (HSMs) that are validated by Federal Information Processing Standards (FIPS) 140-2 to protect keys

Cognito

- Adds user sign-up, sign-in, and access control to your web and mobile applications. Scales to millions of users.
- Supports sign-in with social identity providers, such as Facebook, Google, and Amazon; and enterprise identity providers, such as Microsoft Active Directory via Security Assertion Markup Language (SAML) 2.0.

Shield

- Is a managed distributed denial of service (DDoS) protection service
- Provides always-on detection and automatic inline mitigations
- AWS Shield Standard enabled for at no additional cost. AWS Shield Advanced is an optional paid service.
- Use it to minimize application downtime and latency

Section 5: Securing Data

Data at Rest

- Data at rest = Data stored physically (on disk or on tape)
- Encryption encodes data with a secret key, which makes it unreadable. Only those who have the secret key can decode the data.
- You can encrypt data stored in any service that is supported by AWS KMS

Data in Transit

- Data in transit = Data moving across a network
- Transport Layer Security (TLS) â€” formerly SSL, is an open standard protocol. AWS Certificate Manager provides a way to manage, deploy, and renew TLS or SSL certificates
- Secure HTTP (HTTPS) creates a secure tunnel.

Section 6: Working to Ensure Compliance

Customers are subject to many different security and compliance regulations and requirements. AWS engages with certifying bodies and independent auditors to provide customers with detailed information about the policies, processes, and controls that are established and operated by AWS.

Compliance programs can be broadly categorized:

- Certifications and attestations
- Laws, regulations, and privacy
- Alignments and frameworks - Industry or function-specific security or compliance requirements

AWS Config

- Assess, audit, and evaluate the configurations of AWS resources.
- Automatically evaluate recorded configurations versus desired configurations.
- Review configuration changes and view detailed configuration histories.

AWS Artifact

- A resource for compliance-related information
- Provide access to security and compliance reports, and select online agreements
- Access AWS Artifact directly from the AWS Management Console

---

Knowledge Check

Module 5 - Networking and Content Delivery

Slides

Objectives / Topics

- Recognize the basics of networking
- Describe virtual networking in the cloud with Amazon VPC
- Label a network diagram
- Design a basic VPC architecture
- Indicate the steps to build a VPC
- Identify security groups
- Create your own VPC and add additional components to it to produce a customized network
- Identify the fundamentals of Amazon Route 53
- Recognize the benefits of Amazon CloudFront

Labs / Activities

- Knowledge Check
- Activity: Label a Network Diagram
- Activity: Design a VPC
- Lab: Build Your VPC and Launch a Web Server -- Lab Instructions

Section 1: Networking Basics

Network: Two or more machines that are connected together in order to communicate. A network can be divided into subnets and networking requires a networking device such as a router or a switch.

IP Address: A unique numerical label assigned to each device connected to a computer network. IPv4 defines an IP address as a 32-bit number, but because of the growth of the Internet IPv6 was created, using 128 bits for the IP address.

Classless Inter-Domain Routing (CIDR): A method for allocating IP addresses and IP routing. CIDR notation is a compact representation of an IP address and its associated routing prefix. The notation is constructed from an IP address, a slash ('/') character, and an integer. The integer is the count of leading 1 bits in the subnet mask. Larger values here indicate smaller networks. The maximum size of the network is given by the number of addresses that are possible with the remaining, least-significant bits below the prefix.

- Example: The IPv4 block 192.168.100.0/22 represents the 1024 IPv4 addresses from 192.168.100.0 to 192.168.103.255.

Open Systems Interconnection (OSI) Model: A conceptual model that characterises and standardises the communication functions of a computing system without regard to its underlying internal structure and technology. Its goal is the interoperability of diverse communication systems with standard communication protocols. The model partitions a communication system into abstraction layers.

| Layer | Number | Function | Protocol/Address | |-------------|--------|--|--------------------------| | Application | 7 | Means for an application to access a computer network | HTTP(S), FTP, DHCP, LDAP | | Presentation | 6 | - Ensures that the application layer can read the data - Encryption | ASCI, ICA | | Session | 5 | Enables orderly exchange of data | NetBIOS, RPC | | Transport | 4 | Provides protocols to support host-to-host communication | TCP, UDP | | Network | 3 | Routing and packet forwarding (routers) | IP | | Data Link | 2 | Transfer data in the same LAN network (hubs and switches) | MAC | | Physical | 1 | Transmission and reception of raw bit streams over a physical medium | Signals (1s and 0s) |

Section 2: Amazon VPC

- Enables you to provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define

- Gives you control over your virtual networking resources, including: Selection of IP address range, creation of subnets, and configuration of route tables and network gateways
- Closely resembles a traditional network that you would operate in your own data center, with the benefits of using the scalable infrastructure of AWS
- Enables you to customize the network configurationfor your VPC
- Enables you to use multiple layers of security
- You can create a VPC that spans multiple Availability Zones

VPCs

- Logically isolated from other VPCs
- Dedicated to your AWS account
- Belong to a single AWS Region and can span multiple Availability Zones
- When you create a VPC, you assign it to an IPv4 or IPv6 CIDR block. You cannot change the address range after creation.

Subnets

- Range of IP addresses that divide a VPC
- Belong to a single Availability Zone
- Classified as public (has route to the internet) or private (no internet)
- CIDR blocks of subnets cannot overlap
- Each CIDR block has 5 reserved addresses for: network address, internal communication, DNS resolution, future use, and network broadcast address

Elastic Network Interface

- An Elastic IP address is a static IPv4 address, associated with your AWS account, designed for dynamic cloud computing. You can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.
- A virtual network interface that you can attach to an instance.You can detach from the instance, and attach to another instance to redirect network traffic.
- Its attributes follow when it is reattached to a new instance.
- Each instance in your VPC has a default network interface that is assigned a private IPv4 address from the IPv4 address range of your VPC

Routes and Route Tables

- The route table controls routing for the subnet
- A route table contains a set of rules (or routes) that you can configure to direct network traffic from your subnet
- Each route specifies a destination and a target
- By default, every route table contains a local route for communication within the VPC
- Each subnet must be associated with a single route table

Section 3: VPC Networking

Internet Gateway

A scalable, redundant, and highly available VPC component that allows communication between instances in your VPC and the public internet. An internet gateway serves two purposes:

1. Provide a target in your VPC route tables for internet traffic
2. Perform network address translation for instances that were assigned public IPv4 addresses.

To make a subnet public, you attach an internet gateway to your VPC and add a route entry to the route table associated with the subnet.

Network Address Translation (NAT) Gateway

Enables instances in a private subnet to connect to the internet or other AWS services, but it prevents the public internet from initiating a connection with those instances. To create a NAT Gateway you must:

1. Specify the public subnet in which the gateway should live.
2. Specify an elastic IP address to associate with the NAT Gateway when you create it.

After you create a NAT Gateway, you must update the route table that is associated with one or more of your private subnets to point internet-bound traffic to the NAT gateway. This allows instances in your private subnets to communicate with the internet.

VPC Sharing

Enables customers to share subnets with other AWS accounts in the same organization. VPC Sharing enables multiple AWS accounts to create their application resources in a shared, centrally managed VPC.

The account that owns the VPC shares one or more subnets with other accounts, called participants, that belong to the same organization. After a subnet is shared, participants can view, create, modify, and delete their application resources in the subnets that are shared with them.

VPC Peering

Enables you to privately route traffic between two VPCs. Instances in either VPC can communicate with each other as if they were on the same network. You can create VPC peering connection between your own VPCs with a VPC in another AWS accounts, or between regions.

When you set up the VPC peering connection, you create rules in your route table to allow the VPCs to communicate with each other. VPC peering has some restrictions:

- IP Spaces cannot overlap
- Transitive peering (chaining VPC peering) is not supported
- You can only have one peering resource between the same two VPCs

AWS Direct Connect

Enables your to establish a dedicated private connection between your network and one of the direct connect locations. The private connection can increase bandwidth, throughput, and provide a more consistent network experience than internet-based or VPN connections.

VPC Endpoints

A virtual device that enables you to privately connect a VPC to supported AWS services. There are two types of endpoints:

1. Gateway endpoints that you specify as a target for a route in your route table to either S3 or DynamoDB
2. Interface endpoints are powered by AWS PrivateLink. PrivateLink provides private connectivity between VPCs, AWS services, and on-premises applications.

AWS Transit Gateway

A network transit hub that is used to interconnect virtual private clouds, on-premises networks, VPCs, Direct Connect gateways, and VPN connections to a transit gateway.

The topology of a Transit Gateway is a hub and spoke which reduces the number of connections required, and the complexity to implement and maintain it.

Activity: Label a Network Diagram

Section 4: VPC Security

Security Groups

- Act at the instance level
- Security groups have rules that control inbound and outbound instance traffic.
- Default security groups deny all inbound traffic and allow all outbound traffic.
- Security groups are stateful - return traffic is automatically allowed, regardless of rules
- You can specify allow rules, but not deny rules.
- All rules are evaluated before the decision to allow traffic.

Network Access Control Lists (ACLs)

- Act at the subnet level

- A network ACL has separate inbound and outbound rules, and each rule can either allow or deny traffic.
- Default network ACLs allow all inbound and outbound IPv4 traffic.
- Network ACLs are stateless - return traffic must be explicitly allowed by rules
- Customn network ACLs deny all inbound and outbound traffic until you add rules.
- You can specify both allow and deny rules.
- Rules are evaluated in number order, starting with the lowest number

Activity: Design a VPC

Lab: Build Your VPC and Launch a Web Server -- Lab Instructions

Section 5: Amazon Route 53

- A highly available and scalable Domain Name System (DNS) web service
- Used to route end users to internet applications by translating names (like www.example.com) into numeric IP addresses (like 192.0.2.1) that computers use to connect to each other
- Fully compliant with IPv4 and IPv6
- Connects user requests to infrastructure running in AWS and also outside of AWS
- Is used to check the health of your resources
- Features traffic flow
- Enables you to register domain names

Supported Routing

- Simple routing â€“ Use in single-server environments
- Weighted round robin routing â€“ Assign weights to resource record sets to specify the frequency
- Latency routing â€“ Help improve your global applications
- Geolocation routing â€“ Route traffic based on location of your users
- Geoproximity routing â€“ Route traffic based on location of your resources
- Failover routing â€“ Fail over to a backup site if your primary site becomes unreachable. Improve the availability of your applications that run on AWS by:
- Configuring backup and failover scenarios for your own applications
- Enabling highly available multi-region architectures on AWS
- Creating health checks
- Multivalue answer routing â€“ Respond to DNS queries with up to eight healthy records selected at random

Section 6: Amazon CloudFront

- Fast, global, and secure CDN service
- Global network of edge locations and Regional edge caches
- Self-service model
- Pay-as-you-go pricing

CloudFront Infrastructure

Edge locations â€“ A network of data centers that CloudFront uses to serve popular content quickly to customers.

Regional edge cache â€“ CloudFront location that caches content that is not popular enough to stay at an edge location. It is located between the origin server and the global edge location.

Content Delivery Network

- A globally distributed system of caching servers
- Caches copies of commonly requested files (static content)
- Delivers a local copy of the requested content from a nearby cache edge or Point of Presence
- Accelerates delivery of dynamic content
- Improves application performance and scaling

CloudFront Pricing

- Charged for the volume of data transferred out from Amazon CloudFront edge location to the internet or to your origin.
- Charged for number of HTTP(S) requests.
- No additional charge for the first 1,000 paths that are requested for invalidation each month. Thereafter, $0.005 per path that is requested for invalidation.
- $600 per month for each custom SSL certificate that is associated with one or more CloudFront distributions that use the Dedicated IP version of custom SSL certificate support.

Knowledge Check

Module 6 - Compute

Slides

Objectives / Topics

- Provide an overview of different AWS compute services in the cloud
- Demonstrate why to use Amazon Elastic Compute Cloud (Amazon EC2)
- Identify the functionality in the EC2 consoleâ€¢Perform basic functions in Amazon EC2 to build a virtual computing environment
- Identify Amazon EC2 cost optimization elements
- Demonstrate when to use AWS Elastic Beanstalk
- Demonstrate when to use AWS Lambda
- Identify how to run containerized applications in a cluster of managed servers

Labs / Activities

- Knowledge Check
- Lab: Introduction to Amazon EC2 --- Lab Instructions
- Lab: AWS Lambda --- Lab Instructions
- Lab: AWS Elastic Beanstalk --- Lab Instructions

Section 1: Compute Services Overview

AWS offers many compute services

- EC2
- EC2 Auto Scaling
- Elastic Container Registry (ECR)
- Elastic Container Service
- VMware Cloud on AWS
- Elastic Beanstalk
- Lambda
- Elastic Kubernetes Service (EKS)
- Lightsail
- Batch
- Fargate
- Outposts
- Serverless Application Repository

The optimal compute service or services that you use will depend on your use case, some aspects to consider:

- What is your application design?
- What are your usage patterns?
- Which configuration settings will you want to manage?

Section 2: Amazon Elastic Compute Cloud (EC2)

Overview

- Provides virtual machines â€” referred to as EC2 instances â€” in the cloud
- Gives you full control over the guest operating system (Windows or Linux) on each instance
- You can launch instances of any size into an Availability Zone anywhere in the world with just a few clicks or a line of code, and they are ready in minutes
- Resizable compute capacity
- You can launch instances from Amazon Machine Images (AMIs)

- You can control traffic to and from instances
- Provides tools to build failure resilient applications and isolate them from common failure scenarios

Launching an EC2 Instance

These are the nine key decisions to make when you create an EC2 instance by using the AWS Management Console Launch Instance Wizard.

1. Select Amazon Machine Image (AMI)
2. Is a template that is used to create an EC2 instance (a virtual machine)
3. Contains a Windows or Linux operating system and often has some software pre-installed
4. AMI choices:
    - Quick Start â€“ Linux and Windows AMIs that are provided by AWS
    - My AMIs â€“ Any AMIs that you created
    - AWS Marketplace â€“ Pre-configured templates from third parties
    - Community AMIs â€“ AMIs shared by others; use at your own risk
5. Select an Instance Type
6. Optimized to fit different use cases
7. The instance type that you choose determines
    - Memory (RAM)
    - Processing power (CPU)
    - Disk space and disk type (Storage)
    - Network performance
8. Instance type categories
    - General purpose
    - Compute optimized
    - Memory optimized
    - Storage optimized
    - Accelerated computing
9. Instance types offer family, generation, and size (Example - `t3.large`: Family - `T`, Generation - `3`, Size - `large`)
10. Networking Features
    - The network bandwidth (Gbps) varies by instance type.
    - To maximize networking and bandwidth performance of your instance type enable enhanced networking and if you have interdependent instances, launch them into a cluster placement group.
    - Enhanced networking types are supported on most instance types. Enhanced networking types:
    - Elastic Network Adapter (ENA): Supports network speeds of up to 100 Gbps.
    - Intel 82599 Virtual Function interface: Supports network speeds of up to 10 Gbps.
11. Specify Network Settings
12. Where should the instance be deployed? Identify the VPC and optionally the subnet
13. Should a public IP addressbe automatically assigned?
14. You can have multiple networks, such as different ones for development, testing and production
15. Attach IAM Role (optional)
16. Will software on the EC2 instance need to interact with other AWS services? If yes, attach an appropriate IAM Role. IAM Roles can be attached at any time, not just launch.
17. An AWS Identity and Access Management (IAM) role that is attached to an EC2 instance is kept in an instance profile.
18. User Data Script (optional)
19. Specify a user data script at instance launch. Use user data scripts to customize the runtime environment of your instance
20. Script executes the first time the instance starts
21. Specify Storage
22. Configure the root volume
23. Attach additional storage volumes(optional). For each volume, specify:
    - Disk Size (in GB)
    - Volume type (SSDs or HDDs)
    - If the volume will be deleted when the instance is terminated
    - If encryption should be used
24. Storage Options:
    - Amazon Elastic Block Store (Amazon EBS) â€“
    - Durable, block-level storage volumes.
    - You can stop the instance and start it again, and the data will still be there.
    - Amazon EC2 Instance Store
    - Storage is provided on disks that are attached to the host computer where the EC2 instance is running.
    - If the instance stops, data stored here is deleted.

  - Other options for storage (not for the root volume)
  - Mount an Amazon Elastic File System (EFS) file system.
  - Connect to Amazon Simple Storage Service (S3).

25. Add Tags
26. Consists of a key and an optional value.
27. Tagging is how you can attach metadata to an EC2 instance
28. Potential benefits: Filtering, automation, cost allocation, and access control.
29. Security Group Settings
30. A security group is a set of firewall rules that control traffic to the instance
31. When you launch an instance, you associate one or more security groups with it
32. Create rules that specify the source, which ports that network communications can use and the protocol (TPC, UDP, ICMP)
33. Modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group
34. Identify or Create the Key Pair
35. At instance launch, you specify an existing key pair or create a new key pair
36. A key pair consists of a public key that AWS stores and a private key file that you store
37. It enables secure connections to the instance
38. For Windows AMIs â€“ Use the private key to obtain the administrator password that you need to log in to your instance
39. For Linux AMIs â€“ Use the private key to use SSH to securely connect to your instance

Miscellaneous

Consider using an Elastic IP Address if you require a persistent public IP address.

Instance metadata can be viewed in browser or a terminal window, and can be used to configure or manage a running instance

Amazon CloudWatch can be used to monitor an EC2 instance to provide near-real-time metrics, charts, and 15 months of historical data. CloudWatch has basic monitoring (no additional cost) or detailed monitoring.

Lab: Introduction to Amazon EC2 --- Lab Instructions

Section 3: Amazon EC2 Cost Optimization

Amazon Pricing Models

On-Demand Instances

- Pay by the hour
- No long-term commitments.
- Eligible for the AWS Free Tier.

Dedicated Hosts

- A physical server with EC2 instance capacity fully dedicated to your use.

Dedicated Instances

- Instances that run in a VPC on hardware that is dedicated to a single customer.

Spot Instances

- Instances run as long as they are available and your bid is above the Spot Instance price.
- They can be interrupted by AWS with a 2-minute notification
- Interruption options include terminated, stopped or hibernated
- Prices can be significantly less expensive compared to On-Demand Instances
- Good choice when you have flexibility in when your applications can run.

Reserved Instances

- Full, partial, or no upfront payment for instance you reserve
- Discount on hourly charge for that instance
- 1-year or 3-year term

Scheduled Reserved Instances

- Purchase a capacity reservation that is always available on a recurring schedule you specify
- 1-year term

Per second billing available for On-Demand Instances, Reserved Instances, and Spot Instances that run Amazon Linux or Ubuntu

Four Pillars of Cost Optimization

Right Size

- Provision instances to match the need - CPU, memory, storage, and network throughput
- Use Amazon CloudWatch metrics to dowsize as needed - How idle are instances? When?
- Best practice: Right size, then reserve

Increase Elasticity

- Stop or hibernate Amazon EBS-backed instances that are not actively in use
- Use automatic scaling to match needs based on usage

Optimal Pricing Model

- Leverage the right pricing model for your use case
- Optimize and combine purchase types
- Examples: Use On-Demand Instance and Spot Instances for variable workloads, use Reserved Instances for predictable workloads
- Consider serverless solutions (AWS Lambda)

Optimize Storage Choices

- Reduce costs while maintaining storage performance and availability
- Resize EBS volumes and change EBS volume types
- Delete EBS snapshots that are no longer needed
- Identify the most appropriate destination for specific types of data

Wrap-up

Cost optimization is an ongoing process.

- Define and enforce cost allocation tagging
- Define metrics, set targets, and review regularly.
- Encourage teams to architect for cost
- Assign the responsibility of optimization to an individual or to a team

Section 4: Container Services

Container Basics

Containers are a method of operating system virtualization.

- Repeatable
- Self-contained execution environments
- Software runs the same in different environments
- Faster to launch and stop or terminate than virtual machines

Docker is a software platform that enables you to build, test, and deploy applications quickly. Containers are created from a template called an image.

Amazon Elastic Container Service (ECS)

- A highly scalable, fast, container management service
- Key benefits
- Orchestrates the execution of Docker containers
- Maintains and scales the fleet of nodes that run your containers
- Removes the complexity of standing up the infrastructure
- Integrated with features that are familiar to Amazon EC2 service users

- Elastic Load Balancing
- Amazon EC2 security groups
- Amazon EBS volumes
- IAM roles

Do you want to manage the Amazon ECS cluster that runs the containers?

- If yes, create an Amazon ECS cluster backed by Amazon EC2 (provides more granular control over infrastructure)
- If no, create an Amazon ECS cluster backed by AWS Fargate (easier to maintain, focus on your applications)

Amazon Elastic Container Registry (ECR) is a fully managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images.

Amazon Elastic Kubernetes Service (EKS)

What is Kubernetes?

- Kubernetes is open source software for container orchestration
- Deploy and manage containerized applicationsat scale
- The same toolset can be used on premises and in the cloud
- Complements Docker
- Docker enables you to run multiple containers on a single OS host
- Kubernetes orchestrates multiple Docker hosts (nodes)
- Automates container provisioning, networking, load distribution and scaling

Amazon Elastic Kubernetes Service (Amazon EKS)

- Enables you to run Kubernetes on AWS
- Certified Kubernetes conformant (supports easy migration)
- Supports Linux and Windows containers
- Compatible with Kubernetes community tools and supports popular Kubernetes add-ons
- Use Amazon EKS to manage clusters of Amazon EC2 compute instances and run containers that are orchestrated by Kubernetes on those instances

Section 5: Introduction to AWS Lambda

- Serverless computing enables you to build and run applications and services without provisioning or managing servers.
- Supports multiple programming languages.
- Provides built-in fault tolerance and automatic scaling.
- An event source is an AWS service or developer-created application that triggers a Lambda function to run.
- Pay-per-use pricing
- The maximum memory allocation for a single Lambda function is 3,008 MB.
- The maximum execution time for a Lambda function is 15 minutes
- Deployment package size = 250 MB unzipped, including layers

Lab: AWS Lambda --- Lab Instructions

Section 6: Introduction to Elastic Beanstalk

- An easy way to get web applications up and running
- A managed service that automatically handles
- Infrastructure provisioning and configuration
- Deployment
- Load balancing
- Automatic scaling
- Health monitoring
- Analysis and debugging
- Logging
- No additional charge for Elastic Beanstalk, pay only for the underlying resources that are used
- It supports web applications written for common platforms

- You upload your code and Elastic Beanstalk automatically handles the deployment

Lab: AWS Elastic Beanstalk --- Lab Instructions

Knowledge Check

AWS Lambda Functions and Autoscaling Video --- Walkthrough Instructions

Build a Password-Protected Website with Lambda and CloudFront --- Accompanying Blog

Build, Train, and Deploy a ML Model to SageMaker --- Supporting Notebook

SageMaker Technical Deep Dive Playlist

Deploy a Python App with Plotly Dash and Elastic Beanstalk --- Accompanying Blog

Module 7 - Storage

Slides

Objectives / Topics

- Identify the different types of storage
- Explain Amazon S3
- Identify the functionality in Amazon S3
- Explain Amazon EBS
- Identify the functionality in Amazon EBS
- Perform functions in Amazon EBS to build an Amazon EC2 storage solution
- Explain Amazon EFS
- Identify the functionality in Amazon EFS
- Explain Amazon S3 Glacier
- Identify the functionality in Amazon S3 Glacier
- Differentiate between Amazon EBS, Amazon S3, Amazon EFS, and Amazon S3 Glacier

Labs / Activities

- Knowledge Check
- Lab: Storage --- Lab Instructions

Section 1: Amazon Elastic Block Store (EBS)

Amazon Elastic Block Store (EBS) is an easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale.

With **block storage**, files are split into evenly sized blocks of data, each with its own address but with no additional information (metadata) to provide more context for what that block of data is. **Object storage**, by contrast, doesnâ€™t split files up into raw blocks of data. Instead, entire clumps of data are stored in, yes, an object that contains the data, metadata, and the unique identifier. With block storage you can update a single block without having to update the entire file like in object storage.

Amazon EBS enables you to create individual storage volumes and attach them to an Amazon EC2 instance:

- Offers block-level storage
- HDD and SSD available
- Volumes are automatically replicated within its Availability Zone
- It can be backed up automatically to Amazon S3 through snapshots
- Designed for resiliency - Annual Failure Rate (AFR) is between 0.1% and 1%
- Uses include:
- Boot volumes and storage for (Amazon EC2) instances
- Data storage with a file system
- Database hosts
- Enterprise applications

EBS Features and Charges

Snapshots

- Point-in-time images
- Recreate a new volume at any time
- Added cost of Amazon EBS snapshots to Amazon S3 is per GB-month of data stored

Encryption

- Encrypted Amazon EBS volumes
- No additional cost

Elasticity

- Increase capacity
- Change to different types

Volumes

- Amazon EBS volumes persist independently from the instance
- All volume types are charged by the amount that is provisioned per month.

IOPS (Input/Output Operations Per Second)

- General Purpose SSD: Charged by the amount that you provision in GB per month until storage is released
- Magnetic: Charged by the number of requests to the volum
- Provisioned IOPS SSD: Charged by the amount that you provision in IOPS (multiplied by the percentage of days that you provision for the month).

Data transfer

- Inbound data transfer is free
- Outbound data transfer across Regions incurs charges

Lab: Storage --- Lab Instructions

Section 2: Amazon Simple Storage Service (S3)

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers scalability, data availability, security, and performance. Amazon S3 offers a range of object-level storage classes that are designed for different use cases.

- Data is stored as objects in buckets
- Virtually unlimited storage but a single object is limited to 5 TB
- Designed for 11 9s of durability
- Granular access to bucket and objects
- Data is redundantly stored in the Region

Data can be accessed via AWS Management Console, AWS Command Line Interface, or the SDK.

Common Use Cases and Scenarios

- Storing application assets
- Static web hosting
- Backup and disaster recovery(DR)
- Staging area for big data
- Application hosting
- Media hosting
- Software delivery

S3 Pricing

Pay only for what you use

- GBs per month
- Transfer OUT to other Regions
- PUT, COPY, POST, LIST, and GET requests

You do not pay for

- Transfers IN to Amazon S3
- Transfers OUT from Amazon S3 to Amazon CloudFront or Amazon EC2 in the same Region

Estimating S3 Pricing

1. Storage class type
2. Standard storage is designed for: 11 9s of durability and four 9s of availability
3. S3 Standard-Infrequent Access (S-IA) is designed for 11 9s of durability and three 9s of availability

4. Amount of storage
5. Requests
6. The number and type of requests (GET, PUT, COPY)
7. Type of requests: Different rates for GET requests than other requests.
8. Data transfer
9. Pricing is based on the amount of data that is transferred out of the Amazon S3 Region

Section 3: Amazon Elastic File System

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources. It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth.

EFS Features

- File storage in the AWS Cloud
- Works well for big data and analytics, media processing workflows, content management, web serving, and home directories
- Petabyte-scale, low-latency file system
- Shared storage
- Elastic capacity
- Supports Network File System (NFS) versions 4.0 and 4.1 (NFSv4)
- Compatible with all Linux-based AMIs for Amazon EC2

EFS Implementation

1. Create your Amazon EC2 resources and launch your Amazon EC2 instance
2. Create your Amazon EFS file system.Create your mount targets in the appropriate subnets
3. Connect your Amazon EC2 instances to the mount targets
4. Verify the resources and protection of your AWS account

Section 4: Amazon S3 Glacier

Amazon S3 Glacier is a data archiving service that is designed for security, durability, and an extremely low cost.

- Amazon S3 Glacier is designed to provide 11 9s of durability for objects
- It supports the encryption of data in transit and at rest through Secure Sockets Layer (SSL) or Transport Layer Security (TLS)
- The Vault Lock feature enforces compliance through a policy
- Extremely low-cost design works well for long-term archiving. Pricing is varied on region, and where the data is being sent.
- You can configure lifecycle archiving of Amazon S3 content to Amazon S3 Glacier. Lifecycle policies enable you to delete or move objects based on age.
- Retrieval options:
- Standard: 3â€“5 hours
- Bulk: 5â€“12 hours
- Expedited: 1â€“5 minutes
- Secure Storage
- Server-side encryption with AES-256
- Control access with IAM
- Manages your keys

Knowledge Check

Setting Up a Static Website with an S3 Bucket --- Accompanying Blog

Module 8 - Databases

Slides

Objectives / Topics

- Explain Amazon Relational Database Service (Amazon RDS)
- Identify the functionality in Amazon RDS
- Explain Amazon DynamoDB
- Identify the functionality in Amazon DynamoDB
- Explain Amazon Redshift
- Explain Amazon Aurora
- Perform tasks in an RDS database, such as launching, configuring, and interacting

Labs / Activities

- Knowledge Check
- Lab: Build a Database Server --- Lab Instructions

Section 1: Amazon Relational Database Service (RDS)

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. RDS provides you with six familiar database engines to choose from: Amazon Aurora, Oracle, Microsoft SQL Server, PostgreSQL, MySQL and MariaDB.

Managed vs Unmanaged Service

Unmanaged Challenges

- Server maintenance and energy footprint
- Software installation and patches
- Database backups and high availability
- Limits on scalability
- Data security
- Operating system (OS) installation and patches

RDS is a managed service that sets up and operates a relational database in the cloud. AWS Manages:

- OS installation and patches
- Database software installation and patches
- Database backups
- High availability
- Scaling
- Power and racking and stacking servers
- Server maintenance

RDS Features

The database instance is the basic building block of Amazon RDS. The DB instance is an isolated database environment that can contain multiple user created databases. When setting up the database, you pick an instance class and the type of storage you need for your database. You also need to specify which database engine to run: MySQL, Amazon Aurora, Microsoft SQL Server, PostgreSQL, MariaDB, or Oracle

Amazon RDS allows you to configure your DB instance for high availability with a Multi-AZ (availability zone) deployment. When you configure a Multi-AZ deployment, RDS automatically generates a standby copy of the database instance in another availability zone within the same VPC. After seeding the database copy, transactions are synchronously replicated to the standby copy. If the main database instance fails in a Multi-AZ deployment, RDS automatically brings the standby database instance online as the new main instance.

RDS also supports the creation of read replicas. Updates that are made to the source database instance are asynchronously copied to the read replica instance. You can reduce the load on your source DB instance by routing read queries from your applications to the read replica.

Cost and Usage of RDS

When to use RDS

Use when you require:

- Complex transactions or complex queries
- A medium to high query or write rate â€“ Up to 30,000 IOPS (15,000 reads + 15,000 writes)
- No more than a single worker node or shard
- High durability

Do not use when:

- Massive read/write rates (for example, 150,000 write/second)
- Sharding due to high data size or throughput demands
- Simple GET or PUT requests and queries that a NoSQL database can handle
- Relational database management system (RDBMS) customization

Billing

Multiple factors influence the cost of RDS

- Clock hour billing - resources incur charges when running
- Database characteristics effect cost
- DB Purchase Type
- On-Demand Instances - Compute capacity by the hour
- Reserved Instances - Low, one-time, upfront payment for database instances that are reserved with a 1-year or 3-year term
- Number of DB instances
- Provisioned storage
- No charge - Backup storage of up to 100 percent of database storage for an active database
- Charge (GB/month) - Backup storage for terminated DB instances
- Additional Storage - Charge (GB/Month) for backup storage in addition to the provisioned storage
- Number of Requests
- Deployment type â€” Storage and I/0 charges vary, depending on whether you deploy to a single availability zone or multiple
- Data transfer â€“ No charge for inbound, tiered charges for outbound

Lab: Build a Database Server --- Lab Instructions

Section 2: Amazon DynamoDB

- Fast and flexible NoSQL database service for any scale.
- NoSQL database tables with no limits
- Virtually unlimited storage
- Items can have differing attributes
- Low-latency queries
- Scalable read/write throughput with no limits
- Supports document and key-value store models.
- Replicates your tables automatically across your choice of AWS Regions
- Works well for mobile, web, gaming, adtech, and Internet of Things (IoT) applications
- Provides consistent, single-digit millisecond latency at any scale

Section 3: Amazon Redshift

Amazon Redshift is a fully managed, petabyte-scale data warehouse service in the cloud. The Redshift service manages all of the work of setting up, operating, and scaling a data warehouse. These tasks include provisioning capacity, monitoring and backing up the cluster, and applying patches and upgrades to the Amazon Redshift engine.

- Columnar storage and parallel processing architectures
- Automatically and continuously monitors cluster
- Encryption is built in

Section 4: Amazon Aurora

Amazon Aurora is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases. Aurora features a distributed, fault-tolerant, self-healing storage system that auto-scales up to 64TB per database instance. It delivers high performance and availability with up to 15 low-latency read replicas, point-in-time recovery, continuous backup to Amazon S3, and replication across three Availability Zones (AZs). It also automates time-consuming tasks such as provisioning, patching, backup, recovery, failure detection, and repair.

Knowledge Check

Module 9 - Cloud Architecture

Slides

Objectives / Topics

- Describe the AWS Well-Architected Framework, including the five pillars
- Identify the design principles of the AWS Well-Architected Framework
- Explain the importance of reliability and high availability
- Identify how AWS Trusted Advisor helps customers
- Interpret AWS Trusted Advisor recommendations

Labs / Activities

- Knowledge Check

Section 1: AWS Well-Architected Framework

- A guide for designing infrastructures that are, secure, high-performing, resilient, and efficient
- A consistent approach to evaluating and implementing cloud architectures
- A way to provide best practices that were developed through lessons learned by reviewing customer architectures
- There are 5 pillars to the Well-Architected Framework: Operational Excellence, Security, Reliability, Performance Efficiency, and Cost Optimization
- The AWS Well-Architected Tool helps you to implement the Well-Architected Framework

Operational Excellence

Focus: Run and monitor systems to deliver business value, and to continually improve supporting processes and procedures.

Key Topics

- Managing and automating changes
- Responding to events
- Defining standards to successfully manage daily operations

Design Principles

- Perform operations as code
- Annotate documentation
- Make frequent, small, reversible changes
- Refine operations procedures frequently
- Anticipate failure
- Learn from all operational events and failures

Operational Excellence Questions

Prepare

- How do you determine what your priorities are?
- How do you design your workload so that you can understand its state?
- How do you reduce defects, ease remediation, and improve flow into production?
- How do you mitigate deployment risks?
- How do you know that you are ready to support a workload?

Operate

- How do you understand the health of your workload?
- How do you understand the health of your operations?

- How do you manage workload and operations events?

Evolve

- How do you evolve operations?

Security

Focus: Protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies.

Key Topics

- Identifying and managing who can do what
- Establishing controls to detect security events
- Protecting systems and services
- Protecting confidentiality and integrity of data

Design Principles

- Implement a strong identity foundation
- Enable traceability
- Apply security at all layers
- Automate security best practices
- Protect data in transit and at rest
- Keep people away from data
- Prepare for security events

Security Questions

Identity and access management

- How do you manage credentials and authentication?
- How do you control human access?
- How do you control programmatic access?

Detective Controls

- How do you detect and investigate security events?
- How do you defend against emerging security threats?

Infrastructure Protection

- How do you protect your networks?
- How do you protect your compute resources?

Data Protection

- How do you classify your data?
- How do you protect your data at rest?
- How do you protect your data in transit?

Incident Response

-How do you respond to an incident?

Reliability

Focus: Prevent and quickly recover from failures to meet business and customer demand.

Key Topics

- Setting up
- Cross-project requirements
- Recovery planning
- Handling change

Design Principles

- Test recovery procedures
- Automatically recover from failure
- Scale horizontally to increase aggregate system availability
- Stop guessing capacity
- Manage change in automation

Reliability Questions

Foundations

- How do you manage service limits?
- How do you manage your network topology?

Change Management

- How does your system adapt to changes in demand?
- How do you monitor your resources?
- How do you implement change?

Failure Management

- How do you back up data?
- How does your system withstand component failure?
- How do you test resilience?
- How do you plan for disaster recovery?

Performance Efficiency

Focus: Use IT and computing resources efficiently to meet system requirements and to maintain that efficiency as demand changes and technologies evolve.

Key Topics

- Selecting the right resource types and sizes based on workload requirements
- Monitoring performance
- Making informed decisions to maintain efficiency as business needs evolve

Design Principles

- Democratize advanced technologies
- Go global in minutes
- Use serverless architectures
- Experiment more often
- Have mechanical sympathy

Performance Efficiency Questions

Selection

- How do you select the best performing architecture?
- How do you select your compute solution?
- How do you select your storage solution?
- How do you select your database solution?
- How do you select your networking solution?

Review

- How do you evolve your workload to take advantage of new releases?

Monitoring

- How do you monitor your resources to ensure they are performing as expected?

Tradeoffs

- How do you use tradeoffs to improve performance?

Cost Optimization

Focus: Run systems to deliver business value at the lowest price point.

Key Topics

- Understanding and controlling when money is being spent
- Selecting the most appropriate and right number of resource types
- Analyzing spending over time
- Scaling to meeting business needs without overspending

Design Principles

- Adopt a consumption model
- Measure overall efficiency
- Stop spending money on data center operations
- Analyze and attribute expenditure
- Use managed and application-level services to reduce cost of ownership

Cost Optimization Questions

Expenditure Awareness

- How do you govern usage?
- How do you monitor usage and cost?
- How do you decommission resources?

Cost-Effective Resources

- How do you evaluate cost when you select services?
- How do you meet cost targets when you select resource type and size?
- How do you use pricing models to reduce cost?
- How do you plan for data transfer changes?

Matching Supply and Demand

- How do you match supply of resources with demand?

Optimizing Over Time

- How do you evaluate new services?

Section 2: Reliability and Availability

Reliability

- A measure of your systemâ€™s ability to provide functionality when desired by the user
- System includes all system components: hardware, firmware, and software
- Probability that your entire system will function as intended for a specified period
- Mean time between failures (MTBF) = total time in service/number of failures

Metrics

- Mean Time to Failure (MTTF)
- Mean Time to Repair (MTTR)
- Mean Time Between Failures (MTBF) = MTTF + MTTR

Availability

- Normal operation time / total time
- A percentage of uptime (for example, 99.9 percent) over time (for example, 1 year)

- Number of 9s â€“ Five 9s means 99.999 percent availability

High Availability

- System can withstand some measure of degradation while still remaining available
- Downtime is minimized
- Minimal human intervention is required

Availability Factors

- Fault tolerance: The built-in redundancy of an application's components and its ability to remain operational.
- Scalability: The ability of an application to accommodate increases in capacity needs without changing design.
- Recoverability: The process, policies, and procedures that are related to restoring service after a catastrophic event.

Section 3: AWS Trusted Advisor

AWS Trusted Advisor is an online tool that provides real-time guidance to help you provision your resources following AWS best practices. It looks at your entire AWS environment and gives you real-time recommendations in five categories: Cost Optimization, Performance, Security, Fault Tolerance, Service Limits. You can use AWS Trusted Advisor to help you optimize your AWS environment as soon as you start implementing your architecture designs.

Knowledge Check

Making Your Environment Highly Available --- Password --- Walkthrough Instructions

Module 10 - Automatic Scaling and Monitoring

Slides

Objectives / Topics

- Indicate how to distribute traffic across Amazon Elastic Compute Cloud (Amazon EC2) instances by using Elastic Load Balancing
- Identify how Amazon CloudWatch enables you to monitor AWS resources and applications in real time
- Explain how Amazon EC2 Auto Scaling launches and releases servers in response to workload changes
- Perform scaling and load balancing tasks to improve an architecture

Labs / Activities

- Knowledge Check
- Lab: Scale and Load Balance Your Architecture --- Lab Instructions

Section 1: Elastic Load Balancing

Elastic Load Balancing distributes incoming application or network traffic across multiple targets in a single Availability Zone or across multiple Availability Zones. It also scales your load balancer as traffic to your application changes over time. Monitoring is done via Amazon CloudWatch, access logs, and AWS CloudTrail logs

Types of Load Balancers

Application Load Balancer	Network Load Balancer	Classic Load Balancer (Previous Generation)
Load balancing of HTTP and HTTPS traffic	Load balancing of TCP, UDP, and TLS traffic where extreme performance is required	Load balancing of HTTP, HTTPS, TCP, and SSL traffic
- Routes traffic to targets based on content of request - Provides advanced request routing to targeted at the delivery of modern application architectures, including microservices and containers	- Routes traffic to targets based on IP protocol data - Can handle millions of requests per second while maintaining ultra-low latencies - Optimized to handle sudden and volatile traffic patterns	Load balancing across multiple EC2 instances
Operates at the application layer (OSI model layer 7)	Operate at the transport layer (OSI model layer 4)	Operates at both the application and transport layers

- With Application Load Balancers and Network Load Balancers, you register targets in target groups, and route traffic to the target groups.
- With Classic Load Balancers, you register instances with the load balancer.

Section 2: Amazon CloudWatch

Monitors: AWS resources and applications that run on AWS

Collects and tracks: Standard and custom metrics

Alarms: Send notifications to an Amazon SNS topic and perform Amazon EC2 Auto Scaling or Amazon EC2 actions

- Create alarms based on:
- Static threshold
- Anomaly detection
- Metric math expression
- Specify:
- Name space
- Metric
- Statistic

- Period
- Conditions
- Additional configuration
- Actions

Events: Define rules to match changes in AWS environment and route these events to one or more target functions or streams for processing

Section 3: Amazon EC2 Auto Scaling

- Monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost
- Provides a simple, powerful user interface that enables you to build scaling plans for resources
- Helps you maintain application availability
- Enables you to automatically add or remove EC2 instances according to conditions that you define
- Detects impaired EC2 instances and unhealthy applications, and replaces the instances without your intervention
- Provides several scaling options: Manual, scheduled, dynamic or on-demand, and predictive
- An Auto Scaling group is a collection of EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management.
- Scale out (launch instances), Scale in (terminate instances)

Lab: Scale and Load Balance Your Architecture --- Lab Instructions

Knowledge Check

AWS Lambda Functions and Autoscaling Video --- Walkthrough Instructions