Week 1 : Course Introduction and Cloud Lab Setup in AWS Academy

5 Essential Characteristics of Cloud Computing

1. On-Demand / Self Service
2. Broad Network Access
3. Resource Pooling
4. Elasticity
5. Measured Service

Keywords

- → Scalable
- → Elastic
- → Access via Internet
- → Pay-per-use
- → On-Demand
- → Shared resources
- → Rapidly Provisioned and Release

Essential Characteristics of Cloud Computing (Customer's Consideration)

→ On-Demand (Pay-per-use & On-Demand (self service)
- Choose the requirement without human interaction with Cloud Provider.
- Pay only how much is used
- Ex. How many machine & How long?

→ Rapid Elasticity, Scalable (Elastic & Rapidly Provisioned and Release
- Can scale up or down automatically based on demand.
- Effective capability that allow users to scale their system

Essential Characteristics of Cloud Computing (Provider's Consideration)

→ Measured Service
- Resource usage can be monitored, controlled, and reported
- Providing transparency for both provider and customer sides

→ Broad Network Access (Access via Internet)
- User can access through multiplatforms (PC, Mobile, Tablets)
- Require stable Internet connection

→ Resource Pooling (Shared Resources)
- Resources of Cloud are pooled for multiple customers or users
- Each user is isolated (no access to other user resource)
- High-Availability (HA)

Week 2: Cloud Computing Concepts and Public Cloud Infrastructure

Section 1: AWS Global Infrastructure

AWS Global Infrastructure

→ Built to deliver a flexible, reliable, scalable, and secure cloud computing environment with high-quality global network performance

AWS Region *

→ Geographical Area
→ Data Replication - controlled by user
→ Communication - AWS backbone network infrastructure between regions
→ Each region provides full redundancy and connectivity to the network
→ A region typically consists of two or more Availability Zones

Availability Zones *

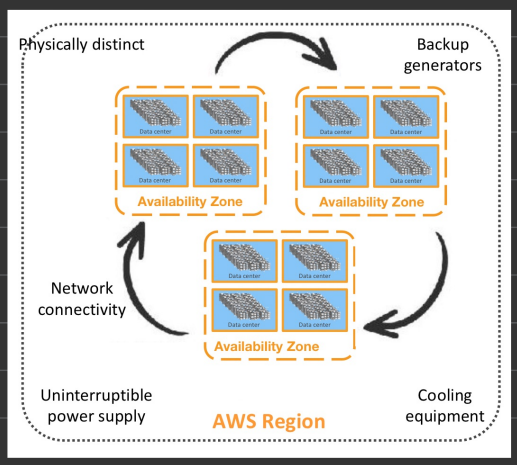
→ Each Availability Zone is a fully isolated partition of the AWS infrastructure
  → 69 zones worldwide currently
  → consists of discrete data centers
  → designed for fault isolation
  → interconnected with other Availability Zones by using high-speed private networking
  → AWS recommends replicating data and resources accross Availability Zones for resiliency

AWS Data Centers

→ Designed for security
→ Where data resides and data processing occurs
→ Each data center has redundant power, networking, and connectivity, and is housed in a separate facility
→ A center has 50,000 to 80,000 physical servers

AWS Infrasture Features

→ Elasticity and Scalability
  → Elastic Infrastructure: dynamic adaption of capacity
  → Scalable Infrastructure: adapts to accommodate growth
→ Fault-Tolerance
  → Continues operating properly in the presence of a failure
  → Built-in redundancy of components
→ High Availabity
  → High level of operational performance
  → Minimized downtime
  → No human intervention



Section 1 conclusion:

- → AWS Global Infrastructure consists of Regions and Availability Zones
- → Region choice is based on compliance requirements or to reduce latency
- → Each Availability Zone is physically separate from other zones and has redundant power, networking, and connectivity
- → Edge locations and regional edge caches improve performance by caching content

Section 3: Moving to the AWS Cloud - The AWS Cloud Adoption Framework (AWS CAF)

AWS Cloud Adoption Framework (AWF CAF)

- → Provides guidance and best practices to help organizations build a comprehensive approach to cloud computing. It helps accelerate successful cloud adoption
- → AWS CAF is organized into six perspectives
- → Perspective consist of sets of capabilities

6 core perspectives

For Business Capabilities
- → Business
- → People
- → Governance

For Technical Capabilities
- → Platform
- → Security
- → Operations

Business Perspective Capabilities

- → IT Finance
- → IT Strategy
- → Benefits Realization
- → Business Risk Management

Note: IT must aligned with business needs and IT investment can be traced to demonstrable business results

People Perspective Capabilities

- → Resource management
- → Incentive management
- → Career management
- → Training management
- → Organizational change management

Note: Prioritize training, staffing, and organizational changes to build an agaile organization

Governance Perspective Capabilities

- → Portfolio management
- → Program and Project Management
- → Business Performance Measurement
- → License Management

Note: Skills and processes align IT strategy and goals with business strategy and goals, so the organization can maximize the business value of its IT investment and minimize business risks

Platform Perspective Capabilities

- → Compute Provisioning
- → Network Provisioning
- → Storage Provisioning
- → Database Provisioning
- → Systems and Solution Architecture
- → Application Development

Note: Understand and communicate the nature of IT systems and their relationships. Must be able to describe the architecture of the target state enviroment in detail

Security Perspective Capabilities

- → Identity and Access Management
- → Detective Control
- → Infrastructure Security
- → Data Protection
- → Incident Responce

Note: The organization must meets its security objectives

Operations Perspective Capabilities

- → Service Monitoring
- → Application Performance Monitoring
- → Resource Inventory Management
- → Release Management / Change Management
- → Reporting and Analytics
- → Business Continuity / Disaster Recovery
- → IT Service Catalog

Note: Align with and support the operations of the business, and define how day-to-day, quarter-to-quarter, and year-to-year business will be conducted

Cloud Delivery and Deployment Models *

IaaS (Infrastructure as a service)	PaaS (Platform as a service)	SaaS (Software as a service)

→

Cost Less
Customer has more control over resources

Cost More
Customer has less control over cloud resources

IaaS *

No Control { Application, Middleware, Guest OS } Total Control

Admin Control ← Hypervisor → Make Requests

Total Control ← Hardware → No Control

Provider — Consumer

Pros

- → Customers can run almost any software or custom OS
- → Can quickly setup large networks of VMs without up-front investment for hardware
- → High level of compability between legacy application

Cons

- → Customers are responsible for patch and update

PaaS *

→ Ready-made environment that establishes a set of pre-packaged products and tools used to support the entire delivery life cycle of custom applications

No Control ← Application → Admin Control

Admin Control ← Middleware → Program to Interfaces

Total Control { OS, Hardware } No Control

Provider — Consumer

Pros

- → Customers do not have to perform administrative task (setup and maintain Infrastructure)

Cons

- → Network speed and reliability
- → Uncertain inefficiencies of scale (lock in with few cloud provider)
- → Portability of new application development

SaaS *

→ Software program as a shared cloud service. Can be leased and used
→ Stack Control: granted very limited administrative control over SaaS

Admin Control ← Application → Limited Admin Control / User Level Control

Total Control { Middleware, OS, Hardware } No Control

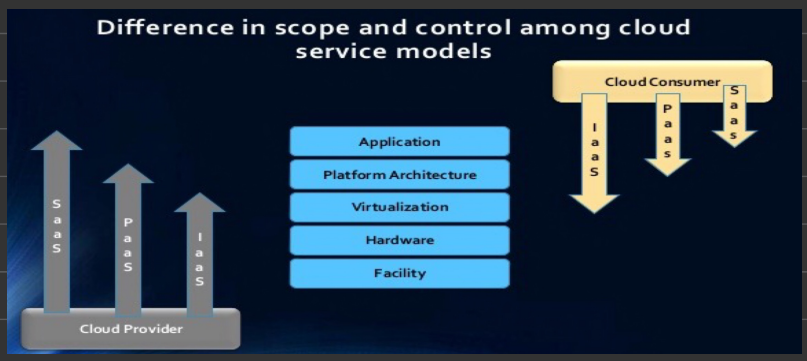
Provider — Consumer

Pros
- → Easier installation
- → Reduce overhead of license management
- → Good for backup and recovery
  (Provider responsible for storing data)
  (Data can be access from any location)
- → Reduce IT supports
- → Lower up-front expenses

Cons
- → Network reliability cannot be guranteed
- → Portability for transitioning workloads from one SaaS to another SaaS provider

*

	IaaS	PaaS	SaaS
Level of Control (Consumer)	Full Administrative	Limited Administrative	Usage and Usage related configuration
Functionality Available	Full access to virtualized infrastructure of IT resources	Moderate level of control over resources	Access to Front-end User Interface
Consumer Activities	Setup and config bare infrastructure, install, manage, monitor	Develops, tests, deploys, and manage cloud services	Uses and config the cloud services
Provider Activities	Provisions and manage physical storage / Monitors usage of consumer	Pre-config the platform and other IT resources / Monitors usage of consumer	Implements, manage, maintain cloud services Monitors usage of consumer



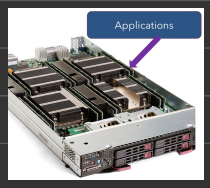
Cloud Computing Development Models

- → Public Cloud
- → On-premises (Private Cloud)
- → Hybrid

Creating a system in Cloud

Traditional System

- → Install required application in a physical server



System in Cloud

- → Cloud Infrastructure is a large amount of physical system
- → Can view cloud system as a single computer

→

Main Components of a Computer

1. CPU (Compute Resources)
   - → Unit of computing resources
     - VM
     - Container
     - Serverless Computing
2. Storage
   - → Block Storage - Hard Disk
   - → Object Storage - Dropbox, Onedrive
   - → Network File Sharing
3. Interfaces (Network Service)
   - → Connect all components inside the same cloud system
   - → Virtual Networks - AWS GCP (Virtual Private Cloud, VPC)
     - Ms Azure (Virtual Network)
4. Other cloud services
   - → Database
   - → Security

More detail later?

Week 3: Foundation Cloud Service: Storage

Module 2-1: Block Storage Concept and Amazon EBS

Block Storage *

- → Persistent Storage - retain data after power down (Non-volatile Storage)
- → Provide consistent and low latency
- → Data is broken up into blocks with ID
- → Read and Write through protocols
- → Change in data result in creating a new object, suitable for static file

Advantage:
1. Typically fast
2. Requires less bandwidth
3. Most programming language can easily read or write files
4. Bootable

Disadvantage:
1. Cost more (pay for all reserved space)
2. Storage can tie to 1 machine at a time
3. Only can access through a running machine
4. Require time to setup

Amazon EBS

- → Durable and Detachable Block-level Storage
- → Volumes are automatically replicated within the same AZ
- → General usage is to attach it to an
- → Can be backed up (to S3)

Volumn Type	Solid State Drive(SSD) General Purpose	Solid State Drive(SSD) Provisioned IOPS	Magnetic Drives (HDD) Throughput-Optimised	Magnetic Drives (HDD) Cold
Use cases	Low latency interactive apps, Dev and Test environment	Workload that require sub-millisecond latency, I/O intensive DB workloads	Big data, Log processing, Data warehouses	Lowest cost is important, for data that is infrequently accessed
Max IOPS	16,000	64,000	500	250
Max Throughput	250 MiB/s	1,000 MiB/s	500 MiB/s	250 MiB/s

Use cases:
1. Boot volumes for EC2
2. Data storage
3. Database Hosts

Features:
1. Snapshots: Recreate a new a new volume at any time and between multiple regions
2. Encryption: Increase security
3. Elasticity: Increase capacity or change volume type
   : Can increase volume size, change volume type, adjust performance

- → Cost per GB for each month.

Module 2-2: File Storage and Amazon EFS

File Storage *

- → Required another protocols to provide abstraction of files and folders
- → FAT/NTFS or NFS and SMB in Network storage
- → Each file can be accessed with specific path

Use cases:
1. File sharing in the network
2. Replication
3. Storage for important files

Advantages:
1. Concurrently access by multiple machine
2. Easy to name, delete, and manage permission across group of files

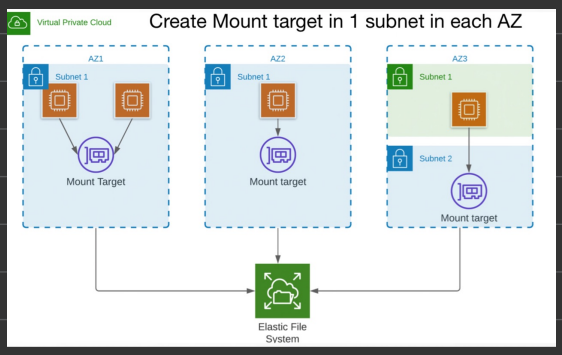
Disadvantages:
1. Difficult to scale
2. Not effective when the stored data becomes very large
   (Go through large amount of folders)

Amazon EFS

→ Multiple VMs can access at the same time
→ Fully managed service: can scale from GB to PB level
→ Support Network File system
→ Compatible with Linux
→ Concurrent access from multiple EC2s in the VPC

Features: 1. 11 9's durability (99.999999%)
2. EFS file system object is redundantly stored across AZs (default)
3. Elasticity: Growing and Shrinking automatically as the files are added/removed

Use cases: 1. Persistent file storages for VMs, Containers, Serverless service
2. Database backups
3. Content Management



Module 3: Adding a Storage Layer

Section 2: Using Amazon S3 Bucket

Object Storage Service
→ stores massive (unlimited) amounts of unstructured data
→ Data stored in as objects in a bucket
→ 5 TB is maximum size for each object
→ All objects have REST-accessible globally URL
→ Object has key, version ID, value, meta data, and subresources

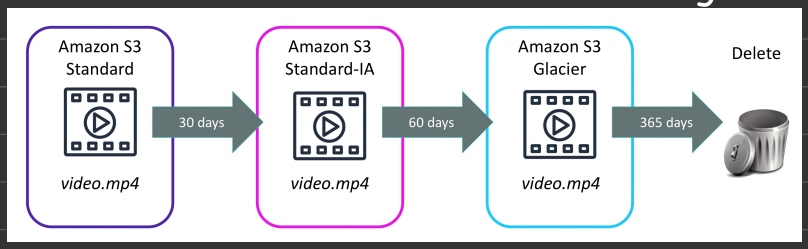
Benefits
→ Durability - Ensure data is not lost
- 11 9's durability
→ Scalability - Virtually unlimited capacity
- Any object of 5 TB or less
→ Availability - Access data when needed
- S3 Standard storage class is designed for four 9's availability
→ Security - Fine grain access control
→ Performance - Supported by many design pattern

Use case:
1. Store and distribute web content and media
2. Host static websites
3. Data store for computation and analytics
4. Backup and archieve critical data

Section 3: Storing data in Amazon S3

Glazier Storage Classes:
1. S3 Standard - Frequently accessed data
2. S3 Standard IA - Long lived, infrequent accessed data
3. S3 One Zone IA - Long Lived, infrequent accessd, non-crit
4. Amazon S3 Glacier (Deep Archieve) - Rarely accessed data

Lifecycle Policies:
→ Move based on age



Amazon S3 cost:

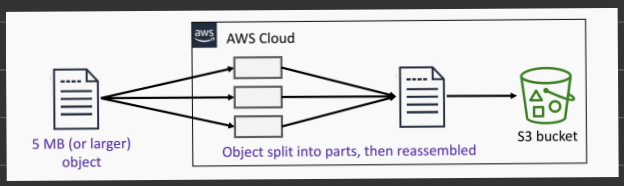
Pay:
- → GB of data stored per month
- → Transfer out to other regions
- → Put, copy, post, list, get, select, life cycle transition, data retrieval requests

No charge:
- → Data transfer in from internet
- → Delete and cancel requests
- → Transfer out to Amazon Cloudfront
- → Transfer between S3 buckets or to AWS services in same region

Section 4: Moving data to and from Amazon S3

- → AWS Management Console
- → AWS Command Line Interface
- → AWS Tools and SDKs

Multipart Uploads



Advantage:
1. Quick recovery from network issues
2. Ability to pause and resume uploads
3. Improved throughput

Amazon S3 Transfer Acceleration: Cloudfront / Protocol

AWS Snowball:

AWS Snow mobile:

Section 5: Choosing Regions for your achitecture

- → Data residency and regulatory compliance
- → Proximity of users to data
- → Service and feature availability
- → Cost-effectiveness

Week 4: Foundation Cloud Service : Compute *

Section 2: Adding compute with Amazon EC2

Amazon EC2

→ Resizable compute capacity in cloud
- Provides VM servers
- Provisions servers in minutes
- Automatically scale capacity up or down
- Pay only for capacity used

EC2 Instances

→ Virtual machine that runs on a physical host
→ Can configurate CPU and memory capacity
→ Support different storage options : 1. Instance store
2. Amazon Elastic Block Store (EBS)
→ Provides network connectivity

Use cases : 1. Control of computing resources
2. Options for cost optimization
3. Run any type of workload

Section 3: Choosing an AMI to launch an EC2 instance

Amazon Machine Image (AMI)

Provide information :
1. Template for root volume
2. Launch permission
3. Block device mappings

Benefits : 1. Repeatability - repeatly launch instances with efficiency/precision
2. Reusability - instances are identically configured
3. Recoverability - create AMI from configured instance as backup
- can replace failed instance by launching new instance from same AMI

Choosing : 1. Region
2. OS
3. Storage type of the root device
4. Architecture
5. Virtualization type

EC2 Image Builder

→ Automates the creation, management, and deployment of golden VM images
- Provides graphical interface to create image-building pipelines
- Creates and maintains EC2 AMI and on-premises VM images
- Produce secure, validated, and up-to-date images
- Enforces version control

Section 4: Selecting an EC2 instance type

→ CPU, memory, storage, and network performance

Instance Types:

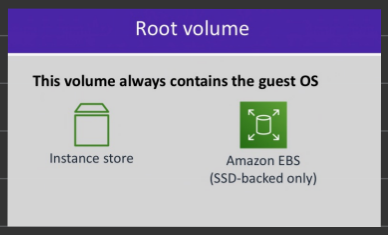
1. General purpose
2. Compute optimized
3. Memory optimized
4. Storage optimized
5. Accelerated Computing

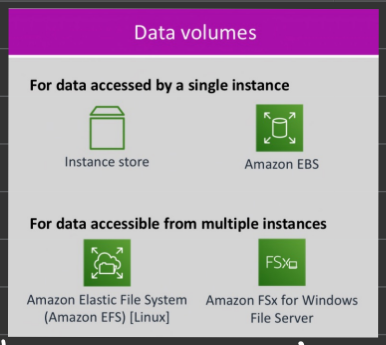
AWS Compute Optimizer

→ Recommends optimal instance type, instance size, and Auto Scaling group config

→ Recommends from work pattern

Section 6: Adding storage to an Amazon EC2 instance





Instance Store

→ Provides non-persistent storage to an instance

Characteristics:
1. Temporary block-level storage
2. Uses HDD or SSD
3. Instance store data is lost when the instance is stopped or terminated

Use cases:
1. Buffers
2. Cache
3. Scratch data

Amazon EBS

→ Network-attached persistent storage to EC2 instance

Characteristics:
1. Persistent block-level storage
2. Can attach to any instance in same Availability Zone
3. Use HDD or SSD
4. Can be encrypted
5. Supports snapshots persists to S3
6. Data persists independently from life of the instance

Use cases:
1. Stand-alone database
2. General application data storage

Amazon EFS

→ file system storage for Linux-based workloads

Amazon FSx

→ fully managed shared file system storage for Windows EC2

Week 5: Foundation Cloud Service: Database

Section 2: Database Layer Considerations

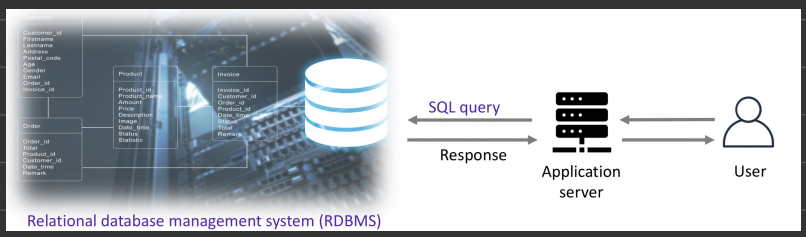
1. Scalability
   → How much throughput is needed? Will it be able to scale up later?
2. Total Storage Requirements
   → How large does the database have to be?
3. Object Size and Type
   → Do we need to store simple data structures, large data objects, or both?
4. Durability
   → Data durability, availability, recoverability level. Do regulatory obligations apply?

Database Types

1. Relational

Advantage:
1. Ease of use
2. Data integrity
3. Reduced data storage
4. Common language (SQL, structured query language?

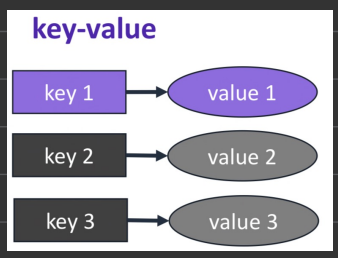
Use cases:
1. Need strict schema rules, ACID compliance, data quality
2. Do not need extreme read/write capacity
3. Do not need extreme performance
   - RDBMS can be the best, lowest effort solution

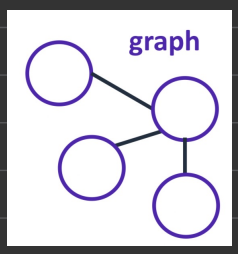


2. Non-relational

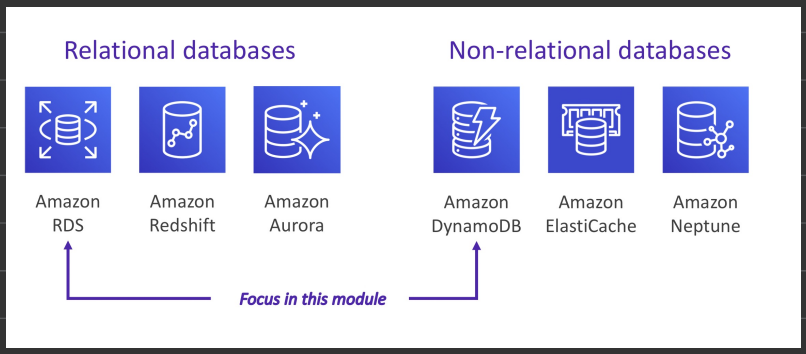
Advantage:
1. Flexibility
2. Scalability
3. High Performance
4. Highly functional APIs

Use cases:
1. Database must scale horizontally to handle massive data volume
2. Data does not lend itself well to traditional schemas
3. Read/write rates exceed what can be economically supported through traditional RDBMS





Amazon Database Options



Section 3: Amazon RDS

→ Fully maraged relational database service

Characteristics:
1. Access pattern: Transactional, Light analytics
2. Data size: Low-TB range
3. Performance: Mid to high throughput, Low latency
4. Business use cases: Transactional, OLAP

Use cases:
1. Complex data application
2. Application that need to combine and join datasets
3. Application that need enforced syntax rules

6 Database types supported:
1. Microsoft SQL Server
2. Oracle
3. MySQL
4. PostgreSQL
5. Aurora
6. MariaDB

Database instance sizing

	T family	M family	R family
Type	Burstable instances	General-purpose instances	Memory-optimized instances
Sizing	1 vCPU/1 GB RAM to 8 vCPU 32 GB RAM	2 vCPU/8 GB RAM to 96 vCPU 384 GB RAM	2 vCPU/16 GB RAM to 96 vCPU 768 GB RAM
Networking	Moderate performance	High performance	High performance
Ideal Workload	Smaller or variable	CPU-intensive	Query-intensive, high connection counts
Highlights	T3 can burst above baseline for extra charge	M5 offers up to 96 vCPU	R5 offers up to 96 vCPU 768 GiB RAM

Multi-AZ deployment for high availability

Advantages:
1. Enhanced durability
2. Increased availability
3. Fail over to standby occurs automatically

Automated failover condition:
- Loss of availability in primary Availability Zone
- Loss of network connectivity to primary
- Compute unit failure on primary
- Storage failure on primary

Read Replicas for Performance

Advantage: 1. Enhanced performance
2. Increased availability
3. Designed for security

Supported by: MySQL MariaDB PostgreSQL Oracle

Limits: 1. Five read replicas per primary
2. For strict read-after-write consistency, read from the primary

Amazon Aurora

→ Fully managed relational database engine

Amazon Redshift

→ Data warehousing service
→ Store very large datasets
→ used for online analytics processing (OLAP)

Section 4: Amazon DynamoDB

Amazon DynamoDB

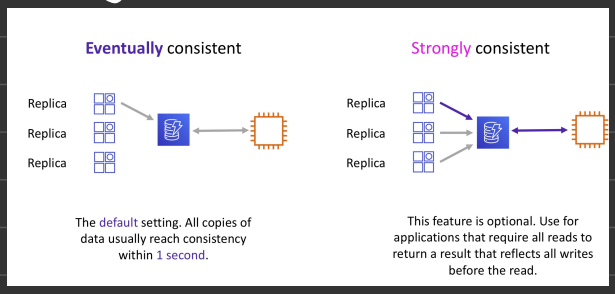
→ Fully managed non-relational key-value and document database service
→ Performance at any scale: horizontal scaling capability
→ Serverless: Event-driven programming
→ Enterprise-ready: Encryption, access controls, backups

Suitable for: 1. High volume data
2. Scale quickly
3. Do not need complex joins
4. Require ultra-high throughput and low latency

Key features: 1. No SQL Tables
2. In-memory caching
3. Item can have different attributes
4. Support >20m requests/sec at peak

Use cases: 1. Leaderboard / Scoring (Game Server)
2. Temporary Data (Online Cart)

Consistent Options:



Section 5: Database security controls

→ See slide 46,47 for recommendations

Section 6: Migrating data into AWS database

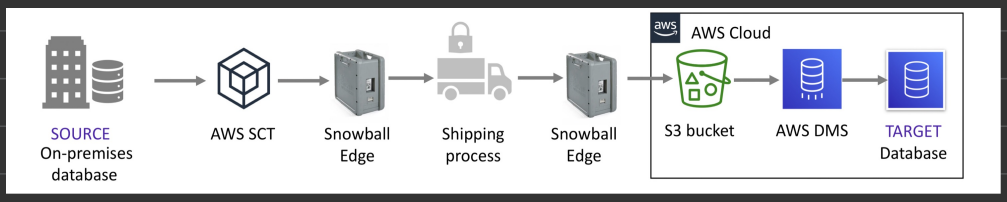
AWS Database Migration Service (DMS)

→ Migrate to and from most commercial and open source databases

→ Migrate between databases on EC2, RDS, S3, and on-premises

Key features:
1. One time migration
2. Continuous data replication
3. AWS Schema Conversion Tool (AWS SCT)
   - support changing the database engine between source and target
4. Steps:
   4.1. Create target database
   4.2 Migrate schema
   4.3. Set up replication process
   4.4. Initiate the transfer
   4.5. Switch production to new database

Note: Use AWS Snowball Edge with AWS DMS when:
1. Database is too large
2. Connection is too slow
3. Privacy and security concerns

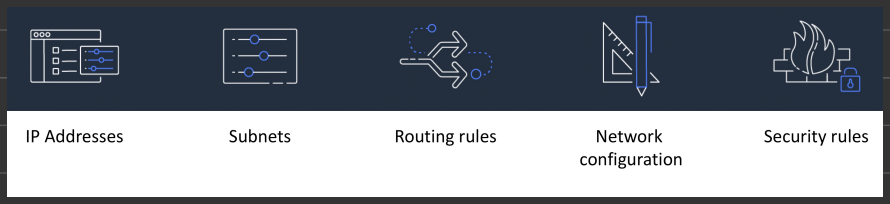


Week6: Foundation Cloud Service: Network

Section 2: Creating an AWS Networking Environment

Amazon VPC

→ Provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define



Subnets: Dividing your VPC

→ Subnet is a segment or partition of a VPC's IP address range where you can allocate a group of resources
→ Subnet are not isolation boundaries
→ Subnets are a subset of VPC CIDR block
→ Subnet CIDR cannot overlap
→ Each subnet resides entirely within one Availability Zone
→ Reserves five IP addresses in each Subnet

Multiple VPCs (One Account)

Best suited: 1. Single team / orgnization
2. Limited team, easier to maintain standards and manage access
Exception: Governance and compliance standards are complex

Multiple Accounts

Best suited: 1. Large organization / multiple IT teams
2. Medium-sized organization + rapid growth
Exception: Challenging to manage access and standards

Section 3: Connecting your AWS networking environment to the internet

Internet Gateways

→ Allow communication between resources in the VPC and the internet
→ Horizontally scaled, redundant, highly available
→ Provide target in subnet route tables for internet routable traffic

Directing Traffic between VPC resources (Route Table)

→ Use Route Table
→ Each VPC has main route table
→ All subnets must be associated with a route table
→ Best practice - use custom route tables for each subnet

Elastic IP Addresses (Remapping IP)

- → Static IP, public IPV4 of AWS account
- → Associated with an instance or elastic network interface
- → Can remap to another instance
- → Useful for redundancy when load balancers are not an option

NAT Gateways

- → Enable instances in a private subnet to initiate outbound traffic to internet or AWS
- → Prevent private instances from receiving inbound requests from internet

Subnet Use cases

- → Data Store Instances
- → Batch-processing instances
- → Backend Instances
- → Web Application Instances

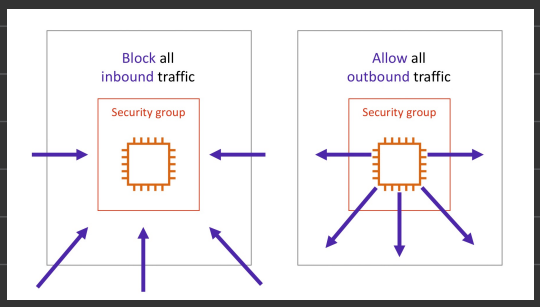
Bastion

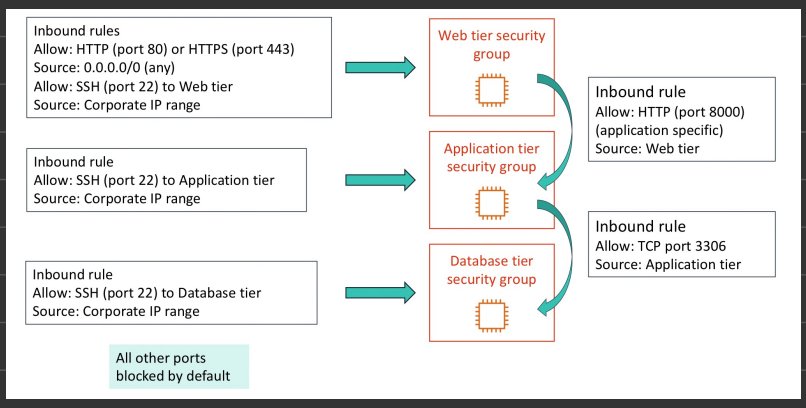
- → Server which provide access to a private network from an external network
- → Minimize chances of penetration

Section 4: Securing your AWS Networking Environment

Security Groups

- → Stateful firewalls that control inbound and outbound traffic
- → Act at the level of instance or network interface





Network Access Control

- → Act at subnet level
- → Allow all inbound and outbound by default
- → Stateless firewalls (explicit inbound and outbound)

Note: structure your infrastructure with multiple layers of defense

Week 7: Foundation Cloud Service: Basic Security

Section 2: Connecting to your remote Network with AWS Site-to-Site VPN

AWS Site-to-Site VPN

→ Connect on-premises network to VPC
- Uses internet protocol security (IPSec) communications to create encrypted virtual private network (VPN) tunnels
- Provides two encrypted tunnels per VPC connection
- Charged per VPC connection - hour

Static and Dynamic Routing

Static : 1. Requires to specify all routes
2. Specify static routing if gateway device does not support BGP

Dynamic : 1. Use Border Gateway Protocol (BGP) to advertise its routes to the virtual private gateway
2. Specify dynamic routing if gateway device supports BGP

Section 3: Connecting to your remote network with AWS Direct Connect

AWS Direct Connect (DX)

→ dedicated, private network connection

Use cases : 1. Hybrid Environment
2. Transferring Large Datasets
3. Network performance predictability
4. Security and compliance

Section 4: Connecting VPCs in AWS with VPC Peering

VPC Peering

→ one to one networking connection between two VPCs
→ No gateways, VPN connections, and separate network appliances needed
→ Highly available connections
→ No single point of failure or bandwidth bottleneck
→ Traffic always stays on the global AWS backbone

Restriction : 1. Use private IP addresses
2. Can establish between two AWS account
3. No overlap CIDR blocks
4. Can only have one peering resources
5. Do not support transitive peering relationships

Section 5: Scaling your VPC Network with AWS Transit Gateway

AWS Transit Gateway

→ Service that connect VPC and on-premises networks to a single gateway
- Fully managed, highly available, flexible routing service
- Acts as a hub for all traffic
- Connects up to 5,000 VPCs and on-premises networks

Step: 1. Create a transit gateway
2. Deploy elastic network interfaces
3. Update the VPC route table
4. Update the transit gateway route table

Section 6: Connecting your VPC to supported AWS services

VPC Endpoints

- → Enable you to privately connect VPC to supported AWS services
- → Enable traffic without leaving Amazon Network
- → Do not require internet gateway, VPN, NAT, or firewall proxies
- → Horizontally scaled, redundant, and highly available

Two Types: 1. Interface Endpoint - elastic network interface with a private IP address that serves as an entry point for traffic destined to a supported service (Powered by AWS PrivateLink)

Ex. Amazon CloudWatch
Amazon EC2 API
Elastic Load Balancing

2. Gateway Endpoint - gateway that specify as a target for a route in your route table for traffic destined to a supported AWS service

Ex. Amazon S3
Amazon DynamoDB

Set up Interface Endpoint

1. Specify AWS service, endpoint service, or AWS Market place service you want to connect
2. Choose VPC where you want to create the interface endpoint
3. Choose a subnet in your VPC that will use the interface endpoint
   Can specify more than one subnet in different Availability zones
4. (Optional) Enable private DNS for endpoint
5. Specify the security groups to associate with the network interface