AWS Certified

Cloud Practitioner

* Sony Shrestha

AWS allows you to share a single server with other AWS customers using virtualization.

Cloud computing services are delivered via the Internet and managed through web browser.

# Cloud Computing

Cloud computing is on demand delivery of compute power, database storage, applications, and other IT resources through a cloud services platform via Internet and pay-as-you-go pricing.

# Cloud Terminology

1. Availability
2. Agility
3. Durability
4. Elasticity

## Availability

* always up and running

## Agility

* high speed

## Durability

* data is not corrupted

## Elasticity

* auto scaling, scale up and scale down

# 6 advantages of Cloud Computing

1. Go global in minutes
2. Stop guessing capacity
3. Stop spending money running and maintaining data centers
4. Increase speed and agility
5. Benefit from massive economic of scale
6. Trade capital expense for variable expense
7. Go global in minutes

* You can deploy your applications around the world at the click of a button.

1. Stop guessing capacity

* Start with what you want then scale-up or scale-down

1. Stop spending money running and maintaining data centers

* No need to manage servers
* Focus on your applications instead of managing hardware

1. Increase speed and agility

* High speed

1. Benefit from massive economic of scale

* Huge discount
* Lower pay-as-you-go

1. Trade capital expense for variable expense

* Pay for what you use instead of making huge upfront investments
* Pay for what you consume

# Cloud Computing Models

1. IaaS
2. PaaS
3. SaaS

## IaaS

* Infrastructure as a Service
* It is the basic building block that you can rent like an EC2 instance.
* You have highest level of flexibility and management control over your IT resources.
* Eg: Web hosting

## PaaS

* Platform as a Service
* Mostly used by developers to build application.
* No need to manage underlying infrastructure like hardware and operating system
* Eg: Storefront website

## SaaS

* Software as a Service
* It is complete product or application that you can rent.
* Eg: Email Provider

# Cloud Deployment Models

1. Private Cloud
2. Public Cloud
3. Hybrid Cloud

## Private Cloud

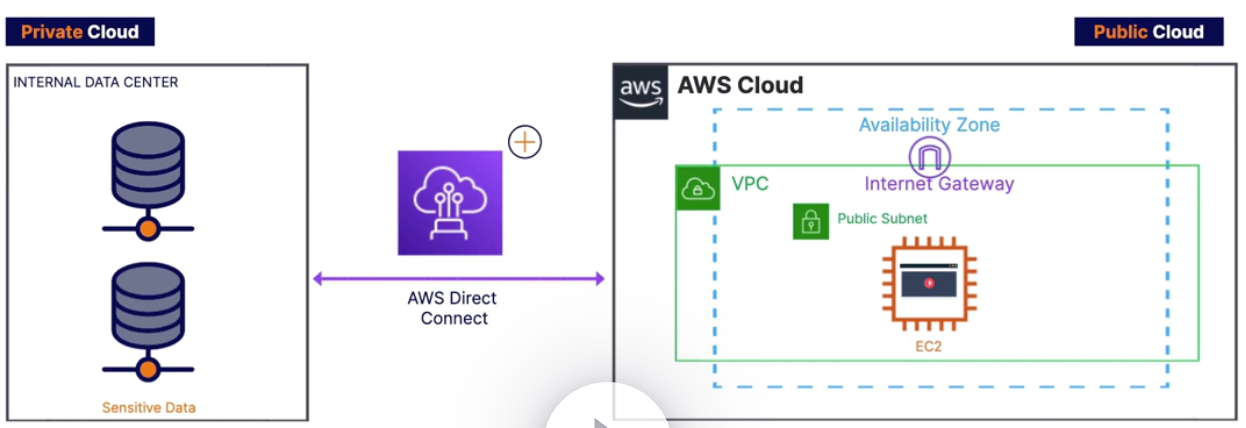
* Called on-premises
* Exists in your internal data center. Everything runs on internal data center
* Does not offer advantage of cloud computing
* There is increased level of security
* This deployment model is same as legacy IT infrastructure

## Public Cloud

* Offered by AWS
* Offers advantages of cloud computing

## Hybrid Cloud

* Combination of private and public cloud
* Highly sensitive data resides in private cloud, application that reads these data run in AWS and they communicate with each other using a service provided by AWS called AWS Direct Connect.



Note:

Hybrid deployments are supported by Direct Connect.

# AWS Global Infrastructure

1. Availability Zone
2. Region
3. Edge Location

Note:

240 countries

200 Edge Locations

80 availability zone

25 regions

Data Center-> Collection of servers

## Availability Zone

* Consists of one or more physically separated data centers, each with redundant power, networking, and connectivity, housed in separate facilities.
* Characteristics of Availability Zone
  + They are physically separated
  + They use separate power grids
  + They are connected among themselves within a region through low-latency links
  + Fault tolerance- If one Availability zone gets out of service, other should not get impact
  + Allows for high availability
* An availability zone is associated with a single region
* Availability Zone contains server you are renting, and it is where you deploy your applications
* To coordinate Availability Zones across accounts, you must use the *AZ ID*, which is a unique and consistent identifier for an Availability Zone.
* An Availability Zone is represented by a region code followed by a letter identifier; for example, ***us-east-1a.***
* You can also use Elastic IP addresses to mask the failure of an instance in one Availability Zone by rapidly remapping the address to an instance in another Availability Zone.
* To ensure that resources are distributed across the Availability Zones for a region, AWS independently map Availability Zones to names for each AWS account. For example, the Availability Zone ***us-east-1a*** for your AWS account might not be the same location as us-east-1a for another AWS account.
* AZs are physically separated within a typical metropolitan region and are located in lower risk flood plains.
* AZs are all redundantly connected to multiple tier-1 transit providers.

## Region

* Physical location
* Collection of multiple availability zones
* When deploying your application, select region that is closest to users. It improves performance and availability
* Characteristics of Region
  + Fully independent and isolated- if one region gets impact, other will not be affected
  + Resource and service specific- resources are not automatically replicated across regions.
* You can replicate data within a region and between regions using private or public Internet connections.
* You retain complete control and ownership over the region in which your data is physically located, making it easy to meet regional compliance and data residency requirements.
* Note that there is a charge for data transfer between regions.
* When you launch an EC2 instance, you must select an AMI that’s in the same region. If the AMI is in another region, you can copy the AMI to the region you’re using.

## Regions and Endpoints

* When you work with an instance using the command line interface or API actions, you must specify its regional endpoint.
* To reduce data latency in your applications, most Amazon Web Services offer a regional endpoint to make your requests.
* An endpoint is a URL that is the entry point for a web service.
* For example, https://dynamodb.us-west-2.amazonaws.com is an entry point for the Amazon DynamoDB service.

## AWS Local Zone

* With AWS Local Zones, you can easily run highly-demanding applications that require single-digit millisecond latencies to your end-users.
* Run latency sensitive applications
* Places AWS services closer to end users

## AWS Wavelength

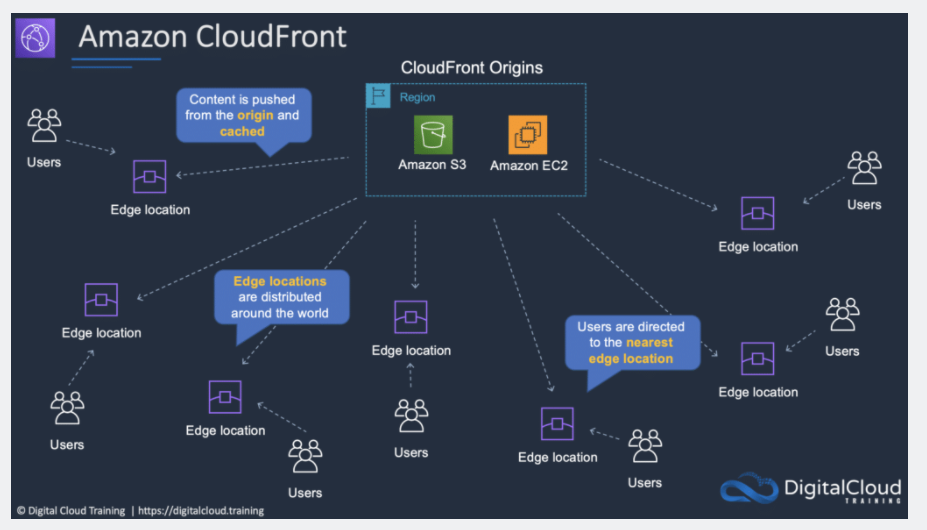
* AWS Wavelength enables developers to build applications that deliver single-digit millisecond latencies to mobile devices and end-users.
* AWS Wavelength brings AWS services to the edge of the 5G network, minimizing the latency to connect to an application from a mobile device.

## Edge Location

* Mini data center
* Cache content for faster delivery of content to users
* There are more edge locations than availability zones and regions
* It is not used to launch resources like EC2 instance. It is just used to cache content.
* Reduces latency and speeds up delivery of your application
* Related to Content Delivery Network (CDN) and Amazon CloudFront

## Regional Edge Cache

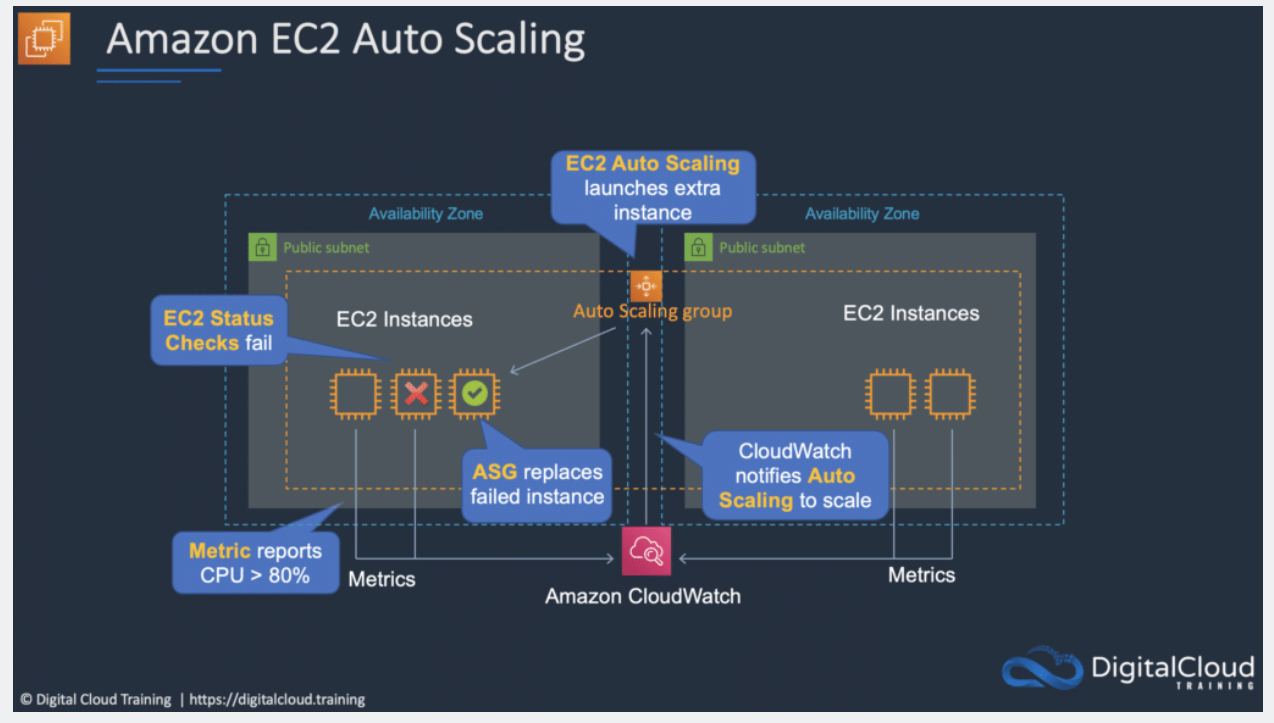
* Sits between CloudFront Origin server and Edge Location
* Regional Edge Cache has a larger cache-width than each of the individual Edge Locations



# Autoscaling and Load balancing

## Autoscaling

* Amazon EC2 Auto Scaling automates the process of launching (scaling out) and terminating (scaling in) Amazon EC2 instances based on the traffic demand for your application.
* Auto Scaling helps to ensure that you have the correct number of EC2 instances available to handle the application load.
* Amazon EC2 Auto Scaling provides elasticity and scalability.
* You create collections of EC2 instances, called an Auto Scaling group (ASG).
* You can specify the minimum number of instances in each ASG, and AWS Auto Scaling will ensure the group never goes beneath this size.
* You can also specify the maximum number of instances in each ASG, and the group will never go above this size.
* A desired capacity can be configured and AWS Auto Scaling will ensure the group has this number of instances.
* You can also specify scaling policies that control when Auto Scaling launches or terminates instances.



## Amazon Elastic Load Balancing (ELB)

* ELB automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses.
* There are four types of Elastic Load Balancer (ELB) on AWS:
  + Application Load Balancer (ALB) – layer 7 load balancer that routes connections based on the content of the request.
  + Network Load Balancer (NLB) – layer 4 load balancer that routes connections based on IP protocol data.
  + Classic Load Balancer (CLB) – this is the oldest of the three and provides basic load balancing at both layer 4 and layer 7
  + Gateway Load Balancer (GLB)

Application Load Balancer (ALB)

* ALB is best suited for load balancing of HTTP and HTTPS traffic and provides advanced request routing targeted at the delivery of modern application architectures, including microservices and containers.
* Operating at the individual request level (Layer 7)

Network Load Balancer (NLB)

* NLB is best suited for load balancing of TCP traffic where extreme performance is required.
* Operating at the connection level (Layer 4)

# Ways to access AWS

1. AWS Management Console
2. AWS Command Line Interface (CLI)
3. AWS Software Development Kit (SDK)

## AWS Management Console

* Allows you to access your AWS account and manage resources using web browser
* Root user is created when we initially sign up for your account. He has unrestricted access to your account, and it cannot be restricted.
* For best practice, use root user for first time access. After that create separate users for day-to-day activities.
* For best practice, protect root user with MFA (Multi-Factor Authentication) This is where a code is sent to your phone that you then have to enter in order to officially sign into your accounts. MFA is sometimes called 2-factor authentication because you have to enter two things your password and code sent to your phone.
* Following tasks can only be performed by root user:
  + Deleting account
  + Changing email address
  + Changing service plan

## AWS Command Line Interface (CLI)

* It is called programmatic way to access your AWS account
* Allows you to access AWS account through a terminal or command window on your local laptop
* Mostly used by developers
* Some new features are available via command line before console
* When you set up CLI locally, you will have to do some configurations like generating secret key and access key and then using that access keys locally.

## AWS Software Development Kit (SDK)

* Access AWS account through application code
* Allows you to access AWS Services from popular programming languages like Java, Python, C# and many more

# Compute Services

1. EC2 (Elastic Cloud Compute)
2. Lambda
3. AWS Fargate
4. Amazon Lightsail
5. Amazon Outposts
6. AWS Batch

## EC2 (Elastic Cloud Compute)

### Introduction

* Elastic Cloud Compute
* Virtual server in the cloud
* Allows you to rent and manage virtual servers in the cloud
* Has elastic compute power: It can grow and shrink based on load in your application
* Are not serverless
* You are able to provision an EC2 instance at the click of a button
* You can also use pre-configured template called AMI (Amazon Machine Image) to launch your resources.
* You will receive 750 compute hours per month on Free Tier Plan
* You can deploy your applications directly to EC2 instances.
* Amazon EC2 instances can run the Windows, Linux, or MacOS operating systems.
* **Integrated** – EC2 is integrated with most AWS services such as S3, RDS, and VPC to provide a complete, secure solution
* **Secure** – EC2 works in conjunction with VPC to provide a secure location

### Amazon Machine Image (AMI)

* An Amazon Machine Image (AMI) is a special type of virtual appliance that is used to create a virtual machine within the Amazon Elastic Compute Cloud.
* An AMI includes the following:
  + One or more [**EBS**](https://digitalcloud.training/certification-training/aws-certified-cloud-practitioner/aws-storage/) snapshots, or, for instance-store-backed AMIs, a template for the root volume of the instance
  + Launch permissions that control which AWS accounts can use the AMI to launch instances.
  + A block device mapping that specifies the volumes to attach to the instance when it’s launched.
* AMIs come in three main categories:
  + **Community AMIs** – free to use, generally you just select the operating system you want.
  + **AWS Marketplace AMIs** – pay to use, generally come packaged with additional, licensed software.
  + **My AMIs** – AMIs that you create yourself.

### Metadata and User Data

* User data is data that is supplied by the user at instance launch in the form of a script.
* Instance metadata is data about your instance that you can use to configure or manage the running instance.
* User data is limited to 16KB.
* User data and metadata are not encrypted.
* Instance metadata is available at http://169.254.169.254/latest/meta-data.

### Instance Metadata Query

* The Instance Metadata Query tool allows you to query the instance metadata without having to type out the full URI or category names.

### EC2 in real world

1. Deploy a database

Deploy a database to EC2 gives you full control over the database.

1. Deploy a web application

Deploy web application to multiple availability zones to make web application highly available.

### Methods to access EC2 instance

1. AWS Management Console
2. Secure Shell (SSH)
3. EC2 Instance Connect (EIC)
4. AWS Systems Manager
5. AWS Management Console

* You are able to configure and manage instances via a web browser.

1. Secure Shell (SSH)

* Allows you to connect to your instance from local laptop using SSH Client and keys
* Puttygen converts .pem(privacy enhanced mail) file to ..ppk (putty private key)

1. EC2 Instance Connect

* Allows you to connect to instance without using SSH client and keys and directly using terminal in web browser
* For this, you must grant IAM users permission to push public key to the instance.

1. AWS System Manager

* Allows you to manage EC2 instances via a web browser or CLI

Note

The must common way to connect to Linux EC2 instance is via Secure Shell (SSH)

For this key pair is generated.

A key pair, consists of a private key and a public key, which proves your identity when connecting to an EC2 instance.

When user connects to SSH Client laptop, he uses private key. When he connects to EC2 instance, he uses public key.

### EC2 Pricing Model

1. On demand
2. Spot
3. Reserved Instance
4. Dedicated Hosts
5. Savings Plans
6. On demand

* Fixed price in which you are billed based on instance type used
* You pay for what you use
* No contract
* Use on-demand instances when
  + You cannot make upfront payment or long-term commitment
  + Your applications have unpredictable workloads that cannot be interrupted
  + Your applications are under-development
  + Your applications will not run for more than a year

1. Spot

* Spot instance lets you take advantage of unused EC2 capacity
* Your request is available only if capacity is available
* Use Spot instance when
  + Your application is not concerned about start and stop time of execution
  + Your workload can be interrupted
  + Applications that are only feasible at very low compute prices.
  + Users with an urgent need for a large amount of additional compute capacity.
* You can save upto 90% off on-demand prices
* You pay for spot that is in effect at the beginning of each hour
* If Amazon terminate your instances you do not pay, if you terminate you pay for the hour.

1. Reserved Instance

* Allows you to commit to a specific instance type for 1 or 3 years
* Use reserved instance when
  + Applications with steady state or predictable usage
  + Your application has steady state usage and you can commit for 1 or 3 years
  + You can make upfront payment
  + Your application requires capacity reservation
* You can save upto 75% off on-demand prices using Standard Reserved Instance
* You are required to sign a contract of 1 or 3 years
* You can pay all-upfront, partial upfront or no upfront. All upfront for maximum term earns the highest discount
* Convertible RIs provide up to 54% off on-demand price – provides the capability to change the attributes of the RI as long as the exchange results in the creation of RIs of equal or greater value.
* Scheduled RIs are available to launch within the time window you reserve. With Scheduled Reserved Instances, you can reserve capacity that is scheduled to recur daily, weekly, or monthly, with a specified start time and duration, for a one-year term.

1. Dedicated Hosts

* Allows you to pay for physical server that is dedicated to running your instances. No other AWS customers is going to have applications running on this server. The server basically belongs to you
* Use dedicated hosts when
  + You want to bring your own server-bound software license from vendors like Microsoft and Oracle
  + You have regulatory or corporate compliance requirements around tenancy model
* You can save upto 70% off on-demand prices
* You can bring your existing per-core, per-socket and per-VM software licenses.
* Physical servers dedicated just for your use.
* You then have control over which instances are deployed on that host.
* Available as On-Demand or with Dedicated Host Reservation.
* Each dedicated host can only run one EC2 instance size and type.
* Most expensive option.
* Billing is per host.

1. Dedicated Instance

* Virtualized instances on hardware just for you.
* Also uses physically dedicated EC2 servers.
* Does not provide the additional visibility and controls of dedicated hosts (e.g. how instance are placed on a server).
* Billing is per instance.
* May share hardware with other non-dedicated instances in the same account.
* Available as On-Demand, Reserved Instances, and Spot Instances.
* Cost additional $2 per hour per region.

1. Savings Plan

* You do not commit for specific instance type instead you commit to compute usage (measured per hour) for 1 to 3 years
* Use savings plan when
  + You want to lower your bill across multiple compute services
  + You want flexibility to change compute services, instance types, OS or regions
* You can save upto 72% off on-demand prices
* Savings can be shared across various compute services like EC2, Fargate and lambda

### EC2 Instance Type

1. General Purpose
2. Compute Optimized
3. Memory Optimized
4. Accelerated Computing
5. Storage Optimized

### Features

1. Elastic Load Balancing
2. EC2 Autoscaling
3. Elastic Load Balancing

* Automatically distributes your incoming traffic across multiple EC2 instances
* Types
  + Classic Load Balancer
  + Application Load Balancer
  + Gateway Load Balancer
  + Network Load Balancer

1. EC2 Autoscaling

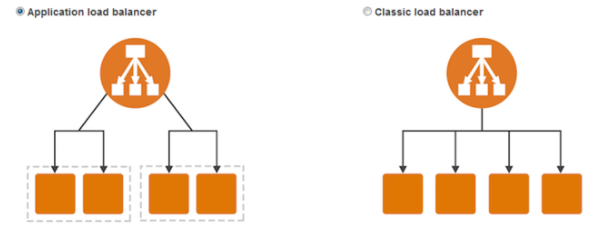
* Scale up or scale down based on workload in your application
* Reduces impact of system failure and improves availability of your applications
* You can also use autoscaling with Aurora and DynamoDB

Classic Load Balancer

* All incoming traffics will be equally distributed among number of servers we have

Application load Balancer

* Incoming traffics are distributed based on needs of customers
* It has some level of intelligence.
* It enables content-based routing and allows requests to be routed to different application/server behind single load balance.



Note

Horizontal Scaling vs Vertical Scaling

Horizontal Scaling (Scaling out)- adds or replaces instances

Vertical Scaling (Scaling up)- upgrades an existing instance

Tag

Allows you to track your instance by adding key/value pair

Eg: We can add a tag to specify “This instance is dedicated for Management purpose.”

## AWS lambda

### Introduction

* Serverless compute service that allows you to run your code without managing servers
* Allows developers to focus on core business logic for the application they are developing instead of worrying about managing servers
* Scales automatically
* Lambda functions can be written in popular programming languages like Java, Go, Node Js, C#, Python
* You can author your code using your favorite development environment locally and upload or you can write code via the console.
* Lambda can execute your code in response to some events
* Lambda functions have a 15-minute timeout. So, lambda is not for processes that run longer than 15 minutes.

**Pricing Model**

1. Compute time
2. Request
3. Always free
4. Compute time

* Time between start of execution and end of execution

1. Request

* A request is counted each time it starts execution
* Test invokes on the console is also considered as a request

1. Always free

* 1 million free requests per month (even after free usage tier expires)

Note

Duration: execution tine

Billed Duration: time you are actually billed for

## AWS Fargate

### Introduction

* Serverless compute service for containers
* Allows you to manage containers like Docker
* Works with Amazon Elastic Container Service (ECS)
* Scales automatically
* There is no additional charge for Amazon ECS. You pay for AWS resources (e.g. EC2 instances or EBS volumes) you create to store and run your application.

### The Elastic container registry (ECR)

* The Elastic container registry (ECR) is a managed AWS Docker registry service for storing, managing and deploying Docker images.
* Amazon ECR is integrated with Amazon EC2 Container Service (ECS).
* With Amazon ECR, there are no upfront fees or commitments. You pay only for the amount of data you store in your repositories and data transferred to the Internet.

## AWS Lightsail

### Introduction

* Quickly launch all the resources you need for small projects
* Deploy preconfigured applications like WebPress Website with the click of a button
* Includes Virtual Server, SSD based storage, static IP address, data transfer and DNS management
* Provides a low, predictable monthly fee, as low as $3.50
* Used to start small and scale as you grow
* Limited to 20 Amazon Lightsail instances, 5 static IPs, 3 DNS zones, 20 TB block storage, 40 databases, and 5 load balancers per account.
* Up to 20 certificates per calendar year.
* Application templates include WordPress, WordPress Multisite, Drupal, Joomla!, Magento, Redmine, LAMP, Nginx (LEMP), MEAN, Node.js, and more.

## Amazon LightSail Databases

### Introduction

* Amazon Lightsail databases are instances that are dedicated to running databases.
* An Amazon Lightsail database can contain multiple user-created databases, and you can access it by using the same tools and applications that you use with a stand-alone database.
* Amazon Lightsail managed databases provide an easy, low maintenance way to store your data in the cloud.
* Amazon Lightsail automatically backs up your database and allows point in time restore from the past 7 days using the database restore tool.
* Amazon Lightsail databases support the latest major versions of MySQL. Currently, these versions are 5.6, 5.7, and 8.0 for MySQL.
* Amazon Lightsail databases are available in Standard and High Availability plans.
* High Availability plans add redundancy and durability to your database, by automatically creating standby database in a separate Availability Zone.
* Amazon Lightsail plans are billed on an on-demand hourly rate, so you pay only for what you use.

## AWS Outspots

### Introduction

* AWS delivers and installs servers in your internal data center
* Allows you to run cloud services in your internal data center
* Supports hybrid deployment model

## AWS Batch

### Introduction

* Allows you to process large workloads in smaller chunks (or batches)

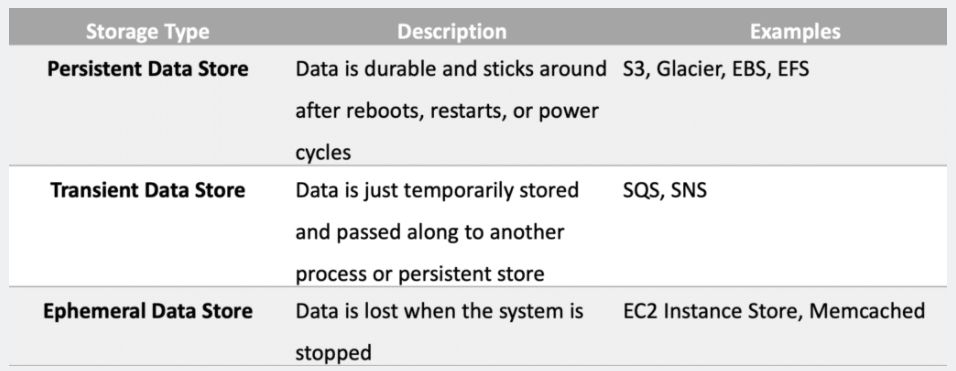
# Storage Services

1. Amazon S3
2. EBS
3. EC2 Instance Store
4. EFS
5. Storage Gateway

## Amazon S3

### Introduction

* Object Storage Service
* Objects are stored in buckets
* Unlimited storage- Millions of objects can be stored per bucket (Unlimited)
* Objects can be private or public
* Objects can be uploaded via Management Console, CLI or from code using SDK
* Is a regional service, but bucket names must be globally unique
* You can set security at bucket level or individual object level using Access Control Lists (ACL), Bucket policies or Access Point policies
* You can enable versioning to create multiple versions of your file in order to protect against accidental deletion or roll back to previous version or file
* You can use S3 access logs to track access to your buckets and objects
* Data stored is replicated across multiple servers and so it gives high availability and durability in regional level.
* We can also set up Cross-Region Replication where your data is replicated across regions. But, S3 does not do that out of box. You have to set that up.
* By default, buckets and their contents are private.
* You can store any type of file in S3.
* S3 is designed to deliver 99.999999999% durability
* Files can be anywhere from 0 bytes to 5 TB.
* There is unlimited storage available.
* Buckets are root level folders.
* Any subfolder within a bucket is known as a “folder”.
* When you successfully upload a file to S3 you receive a HTTP 200 code
* Persistent data stores are non-volatile storage systems that retain data when powered off.
* This is in contrast to transient data stores and ephemeral data stores which lose the data when powered off.



Bucket names must follow a set of rules:

1. Names must be unique across all of AWS.
2. Names must be 3 to 63 characters in length.
3. Names can only contain lowercase letters, numbers and hyphens.
4. Names cannot be formatted as an IP address.

Objects consist of:

1. Key (name of the object).
2. Value (data made up of a sequence of bytes).
3. Version ID (used for versioning).
4. Metadata (data about the data that is stored).

Sub resources: Access control lists, Torrent

Object sharing – the ability to make any object publicly available via a URL.

Lifecycle management – set rules to transfer objects between storage classes at defined time intervals.

Versioning – automatically keep multiple versions of an object (when enabled).

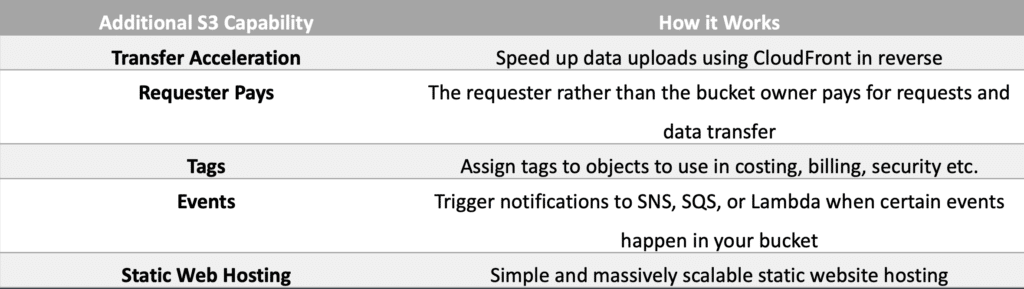
Encryption can be enabled for bucket.

Data is secured using ACLs and bucket policies.

Charges

1. Storage
2. Storage class
3. Data transfer
4. Request and Data Retrieval
5. Transfer acceleration
6. Lifecycle

Additional capabilities offered by Amazon S3 include:



Two aspects of data accessibility

1. Durability
2. Availability
3. Durability

* Objects are never lost or compromised
* Amazon S3 Standard gives 99.999999999% (11 9’s) durability

1. Availability

* Access data quickly
* Amazon S3 Standard gives 99.99% availability

### Use cases

Typical use cases include:

1. Backup and Storage
2. Application Hosting
3. Media Hosting
4. Software Delivery
5. Static Website

### S3 Storage Class

There are different ways of storing your data

1. S3 Standard
2. S3 Intelligent Tiering
3. S3 Standard- Infrequent Access
4. S3 One Zone- Infrequent Access
5. S3 Glacier
6. S3 Glacier Deep Archive
7. S3 Outposts
8. S3 Standard

* General purpose storage
* Data is stored across multiple availability zones
* Recommended for
  + Data that are frequently accessed
* Durability of 99.999999999%
* Availability of 99.99%

1. S3 Intelligent Tiering

* Automatically moves your data to most cost-effective storage class
* Automatic cost savings
* Data is stored across multiple availability zones
* Recommended for
  + Data with unknown or changing access pattern like new app or data lake
* Durability of 99.999999999%
* Availability of 99.9%

1. S3 Standard- Infrequent Access

* Data is accessed less frequently but requires rapid access (millisecond access)
* Cheaper than S3 Standard
* Data is stored across multiple availability zones
* Recommended for
  + Data which is not frequently accessed but requires rapid access (millisecond access)
* Durability of 99.999999999%
* Availability of 99.9%

1. S3 One Zone- Infrequent Access

* Data is accessed less frequently but requires rapid access (millisecond access)
* Costs 20% less than S3 Standard Infrequent Access
* Data is stored in single availability zone
* Data can be lost
* Recommended for
  + Data which is not frequently accessed but requires rapid access (millisecond access)
  + Availability and Durability is not so essential
* Durability of 99.999999999%
* Availability of 99.5%

1. S3 Glacier

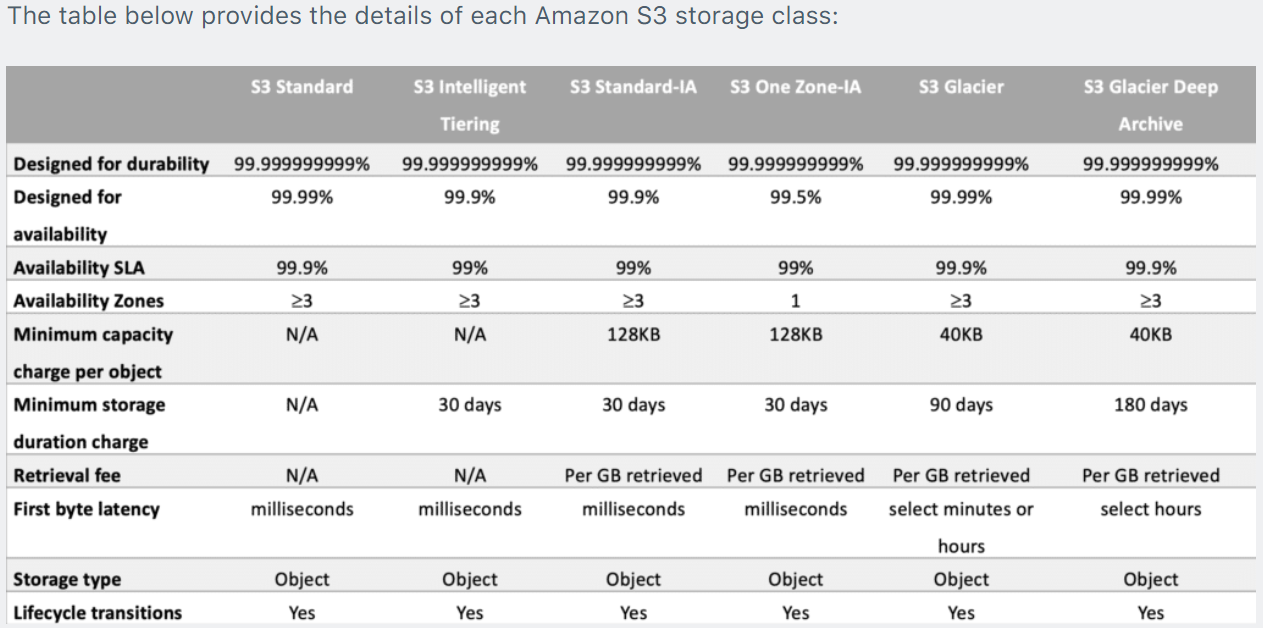
* Cold Storage for archival data
* Data Retrieval takes longer time
* 3 retrieval options: 1-5 min, 3-5 hours, 5-12 hours
* Data is stored in multiple availability zones
* Recommended for
  + Storage of archival data
* Durability of 99.999999999%

1. S3 Glacier Deep Archive

* Like S3 Glacier but longer access time
* 2 retrieval options: 12 hours, 48 hours
* Cheapest of all S3 options
* Data is stored in multiple availability zones
* Recommended for
  + Long-term archival data which is accessed once or twice a year
  + Retaining data for regulatory compliance requirement
* Durability of 99.999999999%

1. S3 Outposts

* Provides object storage on-premises
* Recommended for
  + Data that needs to be kept local (on-premises)



### S3 in real world

1. Static Website

Deploy static websites to S3 and use CloudFront for global distribution

1. Data Archive

Archive data using Amazon Glacier as a storage option for Amazon S3

1. Analytics Services

Stores data in Amazon S3 for use with analytics services like Redshift and Athena

Note

AWS Snowball Client is software that is installed on a local computer and is used to identify, compress, encrypt, and transfer data. Uses 256-bit encryption (managed with the AWS KMS)

Snowball can import to S3 or export from S3.

Snowball must be ordered from and returned to the same region.

To speed up data transfer it is recommended to run simultaneous instances of the AWS Snowball Client in multiple terminals and transfer small files as batches

## EBS (Elastic Block Storage)

### Introduction

* Block storage service
* Like physical hard drive
* Accessible to a single EC2 instance at a time
* EBS instances can be either General Purpose SSD or Provisioned IOPs SSD
* Data persists even when instance is not running
* Recommended for
  + Quickly accessible data
  + Long-term data storage
  + Running a database on an instance
* Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability.
* EBS volume data persists independently of the life of the instance.
* EBS volumes do not need to be attached to an instance.
* You can attach multiple EBS volumes to an instance.
* You cannot attach an EBS volume to multiple instances (use Elastic File Store instead).
* EBS volumes must be in the same AZ as the instances they are attached to.
* Termination protection is turned off by default and must be manually enabled (keeps the volume/data when the instance is terminated).
* Root EBS volumes are deleted on termination by default.
* Extra non-boot volumes are not deleted on termination by default.
* The behavior can be changed by altering the “DeleteOnTermination” attribute.

## EBS Snapshot

* Snapshots capture a point-in-time state of an instance.
* Snapshots are stored on S3.
* Snapshot can be used to create AMI
* Does not provide granular backup (not a replacement for backup software).
* If you make periodic snapshots of a volume, the snapshots are incremental, which means that only the blocks on the device that have changed after your last snapshot are saved in the new snapshot.
* Even though snapshots are saved incrementally, the snapshot deletion process is designed so that you need to retain only the most recent snapshot in order to restore the volume.
* Snapshots can only be accessed through the EC2 APIs.
* EBS volumes are AZ specific but snapshots are region specific.

## EC2 Instance Store

Introduction

* Local storage that is physically attached to host computer and cannot be removed
* Storage is temporary as data loss occurs when EC2 instance is stopped
* Faster with higher I/O speed
* Are ephemeral or temporary
* Recommended for
  + Temporary storage need
  + Data replicated across multiple instances
* Instance store volume root devices are created from AMI templates stored on S3.
* Instance store volumes cannot be detached/reattached.

## Amazon Elastic File System (EFS)

Introduction

* Hierarchical structure
* Can be mounted by multiple EC2 instances can be mounted in EC2 or on-premises servers
* Only supports the Linux file system
* More expensive than EBS
* Recommended for
  + Main directories for business-critical applications
  + Lift and shift existing enterprise apps
* Can connect to instance in another VPC
* Uses NFS protocol
* Can concurrently connect 1 to 1000s of EC2 instances, from multiple AZs.
* By default you can create up to 10 file systems per account.
* There are two performance modes:
  + “General Purpose” performance mode is appropriate for most file systems.
  + “Max I/O” performance mode is optimized for applications where tens, hundreds, or thousands of EC2 instances are accessing the file system.

Note

In S3, API calls are made to access data

S3 can be accessed by other cloud services (unlike EBS and EFS)

## Storage Gateway

Introduction

* Hybrid Storage Service
* Supports hybrid model
* moving backups to the cloud
* To support these use cases, Storage Gateway offers three different types of gateways:
  + File Gateway – provides file system interfaces to on-premises servers.
  + Volume Gateway – provides block-based access for on-premises servers.
  + Tape Gateway – provides a virtual tape library that is compatible with common backup software (block and file interfaces).

Content Delivery Services

Introduction

* Mechanism to deliver content quickly and efficiently based on your geographical location

1. Amazon CloudFront
2. Amazon Global Accelerator
3. Amazon S3 Transfer Acceleration
4. Amazon API Gateway

Amazon CloudFront

Introduction

* Allows global distribution of content
* Makes content available globally
* Delivers data and applications globally with low latency
* Speeds up delivery of static and dynamic web content
* Uses Edge Location to cache content
* If content is already in Edge Location, CloudFront delivers it immediately if not CloudFront retrieves the files from origin
* Integrates with many AWS services (S3, EC2, ELB, Route 53, Lambda)

CloudFront in real world

1. S3 static websites

CloudFront is often used with S3 to deploy content globally.

1. Prevent attacks

Prevent web attacks like DDoS.

1. IP address blocking

Geo restriction prevents users in certain countries from accessing content.

Amazon Global Accelerator

Introduction

* Networking Service that improves performance and availability of your application upto 60% using AWS Global Network Infrastructure
* When the Internet is congested, AWS Global Accelerator optimizes path to your application to keep packet loss, jitter and latency consistently low
* You are provided with two global static Ips that acts as fixed entry point to your application
* On the back end, add or remove your AWS application end points like Application Load Balancer, Network Load Balancer, EC2 instances and Elastic Ips without making user facing changes
* Automatically reroutes traffic to nearest healthy available endpoint to mitigate endpoint failure
* Performs health check of end point
* AWS Global Accelerator continually monitors the health of application endpoints and will detect an unhealthy endpoint and redirect traffic to healthy endpoints in less than 1 minute.

Similarities between Amazon CloudFront and Amazon Global Accelerator

1. Both use Edge Location and Amazon Global Network Infrastructure
2. Both are integrated with Amazon Shield to protect DDoS attack

Amazon S3 Transfer Acceleration

Introduction

* Fast, easy and secure transfer of files over long distance between your client and S3 bucket
* Uses Edge Location
* Customers can upload files to central bucket

Networking Services

Networking connects computers together and allows for sharing of data and applications around the globe, in a secure manner using virtual routers, firewalls and network management services.

1. Amazon Virtual Private Cloud (VPC)
2. Route 53
3. Direct Connect
4. AWS VPN
5. Amazon API Gateway

Amazon Virtual Private Cloud

Introduction

* It is regional service
* Allows you to create virtual private network in the AWS cloud where you launch your resources and deploy applications
* It is a way to isolate and protect resources
* You are able to setup a private virtual network and you specify various things like IP address ranges, security groups, subnet and you configure route tables
* You are able to launch your EC2 instances within your VPC
* To specify range of IP address, use CIDR
* A VPC spans all the Availability Zones in the region.
* When you first create your AWS account a default VPC is created for you in each AWS region.
* A default VPC is created in each region with a subnet in each AZ.
* By default you can create up to 5 VPCs per region.
* The default VPC has all-public subnets. Public subnets are subnets that have:
  + “Auto-assign public IPv4 address” set to “Yes”.
  + The subnet route table has an attached Internet Gateway.
* Instances in the default VPC always have both a public and private IP address.

Components

1. Subnet
2. NACL
3. Router and Route Table
4. Internet Gateway
5. NAT Gateway
6. Virtual Private Gateway
7. Customer Gateway
8. VPC Peering
9. Subnet

Allows you to split network inside the VPC.

1. NACL

Network Access Control List

Ensures proper traffic is allowed into subnet

Can block traffic to particular instance

1. Router and Route table

Defines where network traffic is routed

1. Internet Gateway

Allows communication between Internet and VPC

Allows traffic to internet from VPC

1. NAT Gateway

Communication between instances in private subnet and Internet

1. Virtual Private Gateway

Amazon size of VPN connection

1. Customer Gateway

Your side of VPN connection

1. VPC Peering

Allows you to connect two VPC’s together

Note

To specify a range of IP addresses for VPC, we use CIDR

Options for securely connecting to a VPC are:

1. AWS managed VPN – fast to setup.
2. Software VPN – use 3rd party software.
3. Direct Connect – high bandwidth, low-latency but takes weeks to months to setup.
4. VPN CloudHub – used for connecting multiple sites to AWS.

Flow log data

* Flow Logs capture information about the IP traffic going to and from network interfaces in a VPC.
* Flow log data is stored using Amazon CloudWatch Logs.
* Flow logs can be created at the following levels:
  + VPC.
  + Subnet.
  + Network interface.

Subnet

* After creating a VPC, you can add one or more subnets in each Availability Zone.
* Each subnet must reside entirely within one Availability Zone and cannot span zones.
* Types of subnet:
  + If a subnet’s traffic is routed to an internet gateway, the subnet is known as a public subnet.
  + If a subnet doesn’t have a route to the internet gateway, the subnet is known as a private subnet.
  + If a subnet doesn’t have a route to the internet gateway, but has its traffic routed to a virtual private gateway for a VPN connection, the subnet is known as a VPN-only subnet.

Firewall

1. Network Access Control Lists (ACLs) provide a firewall/security layer at the subnet level. Supports allow and deny rule. Stateless
2. Security Groups provide a firewall/security layer at the instance level. Supports allow rule only. Stateful

VPC Wizard

The VPC Wizard can be used to create the following four configurations:

1. VPC with a Single Public Subnet
2. VPC with Public and Private Subnets
3. VPC with Public and Private Subnets and Hardware VPN Access
4. VPC with a Private Subnet Only and Hardware VPN Access

NAT Instance- managed by you

NAT Gateway- managed by AWS

Route 53

Introduction

* Domain Name System (DNS) service that routes user to application by translating domain name to IP addresses
* Allows domain name registration
* Performs health checks on AWS resources
* Supports hybrid architecture
* Is used for disaster recovery by re-routing traffic from one instance to instance in another region
* Integrates with ELB, S3, and CloudFront as endpoints.

AWS Direct Connect

Introduction

* Supports hybrid model
* Sensitive Data remains in private cloud. Application that reads these data remains in public cloud and they communicate with each other through Direct Connect
* Data travels over private cloud
* Direct Connect is charged by port hours and data transfer.
* Available in 1Gbps and 10Gbps.
* Speeds of 50Mbps, 100Mbps, 200Mbps, 300Mbps, 400Mbps, and 500Mbps can be purchased through AWS Direct Connect Partners.

Direct Connect in real world

1. Transfer internal data directly to AWS, bypassing your Internet Service Provider

AWS VPN

Introduction

* Site-to-site VPN
* By default, instances that are launched into Amazon VPC cannot communicate with internal/ on-premises network
* We can enable access to internal network from your VPC by creating an AWS site-to-site VPN
* Supports hybrid environment
* Cheaper than Direct Connect
* Similar to Direct Connect but data travels over public internet

## **Amazon API Gateway**

### **Introduction**

* Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale.
* With a few clicks in the AWS Management Console, you can create an API that acts as a “front door” for applications to access data, business logic, or functionality from your back-end services.
* Back-end services may include [Amazon Elastic Compute Cloud (Amazon EC2)](https://aws.amazon.com/ec2/), code running on [AWS Lambda](https://aws.amazon.com/lambda/), or any web application.

AWS Outspots

* For hybrid experience
* Outposts is available as a 42U rack that can scale from 1 rack to 96 racks to create pools of compute and storage capacity.
* Services you can run on AWS Outposts include:
  + Amazon EC2.
  + Amazon EBS.
  + Amazon S3.
  + Amazon VPC.
  + Amazon ECS/EKS.
  + Amazon RDS.
  + Amazon EMR.

Databases

Different types of Databases

1. RDS
2. Aurora
3. DynamoDB
4. DocumentDB
5. ElastiCache
6. Neptune

RDS

* Supports popular database engines
  + MySQL
  + Microsoft SQL Server
  + Oracle
  + Amazon Aurora
  + PostgreSQL
  + MariaDB
* AWS manages database with automatic software patching, automatic backups, operating system management and more
* Launch read replicas across regions in order to provide enhanced performance and durability
* RDS is an Online Transaction Processing (OLTP) type of database.
* Aurora is Amazon’s proprietary database.
* RDS is a fully managed service and you do not have access to the underlying EC2 instance (no root access).

Encryption

* You can encrypt your Amazon RDS instances and snapshots at rest by enabling the encryption option for your Amazon RDS DB instance.
* Encryption at rest is supported for all DB types and uses AWS KMS.
* You cannot encrypt an existing DB, you need to create a snapshot, copy it, encrypt the copy, then build an encrypted DB from the snapshot.

DB Subnet Groups

* A DB subnet group is a collection of subnets (typically private) that you create in a VPC and that you then designate for your DB instances.
* Each DB subnet group should have subnets in at least two Availability Zones in a given region.
* It is recommended to configure a subnet group with subnets in each AZ (even for standalone instances).

AWS Charge for:

1. DB instance hours (partial hours are charged as full hours).
2. Storage GB/month.
3. I/O requests/month – for magnetic storage.
4. Provisioned IOPS/month – for RDS provisioned IOPS SSD.
5. Egress data transfer.
6. Backup storage (DB backups and manual snapshots).
7. Multi-AZ

Scalability

* You can only scale RDS up (compute and storage).
* You cannot decrease the allocated storage for an RDS instance.
* You can scale storage and change the storage type for all DB engines except MS SQL.
* RDS provides multi-AZ for disaster recovery which provides fault tolerance across availability zones:
* There is an option to choose multi-AZ during the launch wizard.

Read Replicas

* provide improved performance for reads
* Read replicas are created from a snapshot of the master instance.

Aurora

* Relational database compatible with MySQL and PostgreSQL
* Supports MySQL and PostgreSQL database engines
* 5x times faster than MySQL and 3x times faster than PostgreSQL
* Managed by RDS

DynamoDB

Introduction

* Fully managed NoSQL database
* Serverless
* Push button scaling means that you can scale the DB at any time without incurring downtime.
* DynamoDB is a Web service that uses HTTP over SSL (HTTPS) as a transport and JSON as a message serialization format.
* Amazon DynamoDB stores three geographically distributed replicas of each table to enable high availability and data durability.
* Data is synchronously replicated across 3 facilities (AZs) in a region.
* Cross-region replication allows you to replicate across regions:
* DynamoDB can be used for storing session state.
* Provides two read models.
  + Eventually consistent reads (Default):
  + Strongly consistent reads:

Amazon DynamoDB Accelerator

Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to a 10x performance improvement – from milliseconds to microseconds – even at millions of requests per second.

DocumentDB

* Fully managed document database that supports MongoDB
* Serverless

ElastiCache

* Fully managed in memory datastore
* Compatible with Redis and Memcached engine
* Data can be lost
* ElastiCache EC2 nodes cannot be accessed from the Internet, nor can they be accessed by EC2 instances in other VPCs.
* Can be on-demand or reserved instances too (but not Spot instances).
* ElastiCache can be used for storing session state.
* There are two types of ElastiCache engine:
  + Memcached – simplest model
  + Redis – complex model
* ElastiCache is key/value store
* Can be put in front of databases such as RDS and DynamoDB
* Pay only for the resources you consume based on node hours used.

Neptune

* Fully managed graph database that supports highly connected datasets
* Supports highly connected datasets like social media networks
* Serverless

Use cases

1. Migrate an on-premises Oracle to Cloud- RDS
2. Migrate an on-premises PostgreSQL database to the cloud- RDS/ Aurora
3. Alleviate database load for data that is accessed often- ElastiCache
4. Process large sets of user profiles and social interactions- Neptune
5. NoSQL database fast enough to handle millions of requests per second- DynamoDB
6. Operate MongoDB workloads at scale- DocumentDB

Migration and Transfer Services

1. Database Migration Service (DMS)
2. Server Migration Service (SMS)
3. Snow Family
4. Datasync
5. AWS Migration hub

Database Migration Service (DMS)

Introduction

* Migrate database to or within AWS
* Supports homogenous (oracle-oracle) and heterogenous (oracle-sql) migrations
* Virtually no downtime i.e. your source database remains operational

DMS in real world

1. Oracle to Aurora MySQL

Migrate on-premises oracle database to Aurora MySQL

1. Oracle to Oracle

Migrate an on-premises Oracle database to Oracle in EC2

1. RDS Oracle to Aurora MySQL

Migrate an RDS Oracle database to Aurora MySQL

Server Migration Service (SMS)

Introduction

* Allows you to migrate on-premises servers to AWS
* Server is saved as a new AMI (Amazon Machine Image) and you use that AMI to launch that server as EC2 instance

AWS Snow Family

Snow Family includes

1. Snowcone
2. Snowball and Snowball Edge
3. Snowmobile
4. Snowcone

* Allows you to transfer large amount of on-premises data to cloud using physical device
* Smallest member of data transport devices
* 8 terabytes of storage

1. Snowball and Snowball Edge

* Petabyte scale of data transport
* Cheaper than internet transfer
* It is not just data migration device but is also an edge computing device i.e., it allows local processing
* Snowball edge supports EC2 and lambda

1. Snowmobile

* Multi-petabyte or exabyte scale of data transport
* Data is transported using 45-foot long shipping container
* People use this when they are completely shipping their on-premises database and moving to AWS
* With these trucks, there is GPS tracking, alarm monitoring, 24X7 video surveillance, and as escort security vehicle during transport

Datasync

Introduction

* Allows for online data transfer from on-premises to AWS Storage Services like S3 or EFS
* Data transfer speeds are upto 10 times faster than open source tools
* Can be used to replicate data across cross region or cross account

## AWS Migration Hub

### Introduction

1. View migration progress
2. View status of migration

# Analytics Services

1. Redshift
2. Athena
3. Glue
4. Kinesis
5. EMR (Elastic Map Reduce)
6. Data Pipeline
7. Amazon CloudSearch
8. Amazon ElasticSearch
9. Redshift

* Scalable data warehouse solution
* Redshift is data warehouse solution that supports querying, reporting, analytics and business intelligence
* RedShift is ideal for **processing** large amounts of data for business intelligence.
* RedShift is 10x faster than a traditional SQL DB.
* RedShift uses columnar data storage
* RedShift provides advanced compression
* RedShift always keeps three copies of your data
  + The original.
  + A replica on compute nodes (within the cluster).
  + A backup copy on S3.
* RedShift provides continuous/incremental backups
* RedShift provides fault tolerance for the following failures
  + Disk failures
  + Nodes failures
  + Network failures
  + AZ/region level disasters

1. Athena

* Analyze S3 data using SQL 🡪 run queries directly in S3
* Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run.
* Amazon Athena uses Presto with full standard SQL support and works with a variety of standard data formats, including CSV, JSON, ORC, Apache Parquet and Avro.
* Amazon Athena uses a managed Data Catalog to store information and schemas about the databases and tables that you create for your data stored in Amazon S3.

1. Glue

* Prepares data for analytics
* ETL (Extract, Transform, Load)
* AWS Glue automatically discovers and profiles data via the Glue Data Catalog
* Glue can automatically discover both structured and semi-structured data stored in data lakes on [**Amazon S3**](https://aws.amazon.com/s3/), data warehouses in [**Amazon Redshift**](https://aws.amazon.com/redshift/), and various databases running on AWS.
* Recommends and generates ETL code to transform your source data into target schemas.
* Glue automatically generates Scala or Python code for ETL jobs that you can further customize using tools you are already familiar with.
* AWS Glue runs the ETL jobs on a fully managed, scale-out Apache Spark environment to load your data into its destination.

1. Kinesis

* Analyze real time streaming data
* Analyze log data
* Data is processed in “shards”.
* There are four types of Kinesis service.
  + Kinesis Video Streams
  + Kinesis Data Streams
  + Kinesis Data Firehouse-loads streaming data
  + Kinesis Data Analytics

1. EMR (Elastic Map Reduce)

* Process large amount of data
* Process big data
* Analyze data using Hadoop
* Works with big data frameworks like Apache Spark

1. Data Pipeline

* Helps you move data between compute or storage services running either on AWS or on-premises
* Moves data at specific interval
* Moves data based on conditions
* Sends notifications on success or failure
* Eg: Move data from S3 to Redshift

1. Amazon CloudSearch

* Managed service
* Amazon CloudSearch is a managed service in the AWS Cloud that makes it simple and cost-effective to set up, manage, and scale a search solution for your website or application.
* Amazon CloudSearch supports 34 languages and popular search features such as highlighting, autocomplete, and geospatial search.
* With a few clicks in the [AWS Management Console](https://aws.amazon.com/console/), you can create a search domain and upload the data that you want to make searchable, and Amazon CloudSearch will automatically provision the required resources and deploy a highly tuned search index.

1. Amazon Elasticsearch

* Managed service
* Elasticsearch is a popular open-source search and analytics engine for use cases such as log analytics, real-time application monitoring, and click stream analytics.
* You can set up and configure your Amazon Elasticsearch domain in minutes from the AWS Management Console.
* You can send data in the form of JSON documents to Elasticsearch using the API or ingestion tools such as Logstash and Amazon Kinesis Firehose.

Analytics in Real World

1. Search data in S3- Athena
2. Log analytics- Kinesis

# Media Services

1. **Amazon Elastic Transcoder**

## Amazon Elastic Transcoder

### Indroduction

* media transcoding in [the cloud.](https://aws.amazon.com/what-is-cloud-computing/)
* It is designed to be a highly scalable, easy to use and a cost effective way for developers and businesses to convert (or “transcode”) media files from their source format into versions that will playback on devices like smartphones, tablets and PCs.

Machine Learning Services

1. Recognition
2. Comprehend
3. Polly
4. Sagemaker
5. Translate
6. Lex
7. Transcribe

Recognition

Introduction

* Allows you to automate image and video analysis

Real World Use cases

* Identify toppings on pizza to make sure that they are being made consistently
* Object detection
* Face recognition
* Identify text in image

Comprehend

Introduction

* Natural Language Processing service
* Analyze text

Real World Use case

* Review social media posts

Analyze comments in social media to find how many people are happy

Polly

Introduction

* Turns text into speech
* Can create a custom voice

SageMaker

Introduction

* Helps you build, train and deploy machine learning models quickly
* Prepare data for models
* Train and deploy models
* Provides Deep Learning AMIs

Real World Use case

* Recommendation Engine

Translate

Introduction

* Language translation

Lex

Introduction

* Build conversational interfaces like Chatbots

Real World Use case

* Integrate voice into a device

## Transcribe

### Introduction

* Amazon Transcribe is an automatic speech recognition (ASR) service that makes it easy for developers to add speech-to-text capability to their applications.
* Using the Amazon Transcribe API, you can analyze audio files stored in Amazon S3 and have the service return a text file of the transcribed speech.
* Amazon Transcribe can be used for lots of common applications, including the transcription of customer service calls and generating subtitles on audio and video content.
* The service can transcribe audio files stored in common formats, like WAV and MP3, with time stamps for every word so that you can easily locate the audio in the original source by searching for the text.

Developer Tools

1. Cloud9
2. Code Commit
3. Code Build
4. Code Deploy
5. Code Pipeline
6. X-Ray
7. AWS Code Star
8. Cloud9

Introduction

* IDE (Integrated Development Environment)
* Allows you to write and debug code from web browser

1. Code Commit

Introduction

* Similar to GitHub
* Create repositories to store code

Use case

* Manage versions of source code files for your applications

1. Code Build

Introduction

* Allows you to build and test your application source code
* Produces build artifacts that are ready to be deployed

Use case

* Run tests before deploying a new version of an application to production

1. Code Deploy

Introduction

* Manages deployment of code
* Maintains application uptime

Use case

* Maintains application uptime when rolling out a new version

1. Code Pipeline

Introduction

* Automate software release process
* Quickly deliver new features and updates
* Integrates with Code Commit to retrieve source code
* Integrates with Code Build to build and run application source code
* Integrates with Code Deploy to deploy your changes

Use case

* Automation of building, testing and deployment of your application

1. X-Ray

Introduction

* Analyze and debug production workloads
* Map application components
* View request flow end-to-end

Use case

* Trace calls to RDS Database

1. AWS CodeStar

Introduction

* AWS CodeStar enables you to quickly develop, build, and deploy applications on AWS
* Easily manage your software development activities in one place.
* With AWS CodeStar, you can set up your entire [continuous delivery](https://aws.amazon.com/devops/continuous-delivery/) toolchain in minutes, allowing you to start releasing code faster
* With AWS CodeStar, you can use a variety of project templates to start developing applications on [Amazon EC2](https://aws.amazon.com/ec2/details/), [AWS Lambda](https://aws.amazon.com/lambda/details/), and [AWS Elastic Beanstalk.](https://aws.amazon.com/elasticbeanstalk/details/)
* AWS CodeStar projects support many popular programming languages including Java, JavaScript, PHP, Ruby, and Python.

Deployment and Infrastructure Management Service

1. CloudFormation
2. Elastic Beanstalk
3. OpsWorks

Infrastructure as a code

It allows you to write code in JSON or YAML format to provision AWS resources

1. CloudFormation

Introduction

* Allows you to provision AWS resources using Infrastructure as a Code
* Works with most AWS services
* Create template for resources you want to provision

Use case

* Automate Infrastructure-provisioning process for EC2 servers

1. Elastic Beanstalk

Introduction

* Serverless compute service that allows you to deploy your web application or web services on Cloud
* It is not used to deploy your application on-premises
* Monitors application health via a health dashboard
* Elastic Beanstalk is ideal if you have a PHP, Java, Python, Ruby, Node.js, .NET, Go, or Docker web application.

Use case

* Deploy a scalable Java-based web application to AWS

1. OpsWorks

Introduction

* Allows you to use Chef or Puppet to automate configuration of your servers and deploy code on-premises or the cloud
* Can deploy applications on-premises
* OpsWorks has three offerings, [AWS OpsWorks for Chef Automate](https://aws.amazon.com/opsworks/chefautomate/), [AWS OpsWorks for Puppet Enterprise](https://aws.amazon.com/opsworks/puppetenterprise/), and [AWS OpsWorks Stacks](https://aws.amazon.com/opsworks/stacks/)

Use case

* Automate software configurations and infrastructure management for your application

Messaging and Integration Services

1. SQS (Simple Queue Service)
2. SNS (Simple Notification Service)
3. SES (Simple Email Service)
4. Simple Queue Service (SQS)

Introduction

* Message queuing service that allows you to build loosely coupled systems
* Allows component-to-component communication using messages
* Multiple components can add messages to the queue
* Messages are processed in an asynchronous manner

Use case

* Build money transfer app that performs well under heavy workload

1. Simple Notification Service (SNS)

Introduction

* Allows you to send plain text email or text messages from your application

SNS Concepts

1. Topics: Topic names are limited to 256 characters
2. Publishers
3. Subscribers

Data type is JSON.

SNS Subscribers:

1. HTTP
2. HTTPS
3. Email
4. Email-JSON
5. SQS
6. Application
7. Lambda

SNS supports notifications over multiple transport protocols:

1. HTTP/HTTPS
2. Email/Email-JSON
3. SQS
4. SMS

Use case

* Send as email when CPU utilization of EC2 instance goes above 80%

1. Simple Email Service (SMS)

Introduction

* Allows you to send richly formatted email from your application
* Unlike SNS, it is used to send HTML emails
* Ideal choice for marketing campaigns or professional emails

Use case

* Send marketing emails

# Mobile Services

1. **AWS AppSync**
2. **AWS Device Farm**

## **AWS AppSync**

### Introduction

* AWS AppSync makes it easy to build data-driven mobile and browser-based apps that deliver responsive, collaborative experiences by keeping the data updated when devices are connected, enabling the app to use local data when offline, and synchronizing the data when the devices reconnect.
* AWS AppSync uses the open standard GraphQL query language so you can request, change, and subscribe to the exact data you need with just a few lines of code.

## **AWS Device Farm**

### Introduction

* AWS Device Farm is an app testing service that lets you test and interact with your Android, iOS, and web apps on many devices at once, or reproduce issues on a device in real time.
* View video, screenshots, logs, and performance data to pinpoint and fix issues and increase quality before shipping your app.

# Application Integration

1. AWS Step Functions
2. **Amazon MQ**
3. Amazon SQS
4. **Amazon SWF**

## **AWS Step Functions**

* AWS Step Functions lets you coordinate multiple AWS services into serverless workflows so you can build and update apps quickly.
* Using Step Functions, you can design and run workflows that stitch together services such as AWS Lambda and Amazon ECS into feature-rich applications.
* Workflows are made up of a series of steps, with the output of one step acting as input into the next.

## **Amazon MQ**

* Amazon MQ is a managed message broker service for [Apache ActiveMQ](http://activemq.apache.org/) that makes it easy to set up and operate message brokers in the cloud.
* Message brokers allow different software systems–often using different programming languages, and on different platforms–to communicate and exchange information.

## **Amazon SQS**

* Message queuing service that allows you to build loosely coupled systems
* Allows component-to-component communication using messages
* Multiple components can add messages to the queue
* Messages are processed in an asynchronous manner
* Build money transfer app that performs well under heavy workload

## **Amazon SWF**

* fully managed state tracker and task coordinator in [the Cloud.](https://aws.amazon.com/what-is-cloud-computing/)
* Amazon SWF helps developers build, run, and scale background jobs that have parallel or sequential steps.

# Internet of Things

1. AWS IoT Core

## **AWS IoT Core**

* AWS IoT Core is a managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices.
* AWS IoT Core can support billions of devices and trillions of messages, and can process and route those messages to AWS endpoints and to other devices reliably and securely.
* With AWS IoT Core, your applications can keep track of and communicate with all your devices, all the time, even when they aren’t connected.

# Desktop & App Streaming

1. Amazon Workspace

## **Amazon Workspaces**

* Amazon WorkSpaces is a managed, secure cloud desktop service.
* You can use Amazon WorkSpaces to provision either Windows or Linux desktops in just a few minutes and quickly scale to provide thousands of desktops to workers across the globe.
* Unlike traditional on-premises Virtual Desktop Infrastructure (VDI) solutions, you don’t have to worry about procuring, deploying, and managing a complex environment – Amazon WorkSpaces takes care of the heavy lifting and provides a fully managed service.

Exploring Auditing, Monitoring and Logging Services

1. CloudWatch
2. CloudTrail
3. CloudWatch

Introduction

* Collection of services like CloudWatch Alarms, CloudWatch Logs, CloudWatch Metrics, CloudWatch Events
* CloudWatch Alarm is used to set alarm
* CloudWatch Logs is used to monitor application logs
* CloudWatch Metrics is used to visualize metrics over time
* CloudWatch Events is used to trigger an event based on condition
* With CloudTrail, you can create a trail that applies to either one region or all regions.
* CloudTrail delivers log files within 15 minutes of user activity.
* Log files are encrypted.
* Cloud Watch is for performance monitoring. Cloud Trail is for auditing.
* CloudWatch is accessed via API, command-line interface, AWS SDKs, and the AWS Management Console.
* CloudWatch integrates with IAM.
* Amazon CloudWatch Logs lets you monitor and troubleshoot your systems and applications using your existing system, application and custom log files.
* CloudWatch Logs can be used for real time application and system monitoring as well as long term log retention.
* CloudWatch Logs keeps logs indefinitely by default.
* CloudTrail logs can be sent to CloudWatch Logs for real-time monitoring.
* CloudWatch retains metric data as follows:
  + Data points with a period of less than 60 seconds are available for 3 hours.
  + Data points with a period of 60 seconds (1 minute) are available for 15 days.
  + Data points with a period of 300 seconds (5 minute) are available for 63 days.
  + Data points with a period of 3600 seconds (1 hour) are available for 455 days (15 months).
* Basic monitoring = 5 mins (free for EC2 Instances, EBS volumes, ELBs and RDS DBs).
* Detailed monitoring = 1 min (chargeable)
* Do not store logs on non-persistent disks:
* Best practice is to store logs in CloudWatch Logs or S3.

Use case

* Provide real time monitoring on EC2 instance
* Receive a notification when root user activity is detected in your account

1. CloudTrail

Introduction

* Track user activities within your account that has been performed using console, CLI or SDK
* Track which user made changes
* Detect unusual activity in your account
* If you want to track activities of above 90 days, you have to create custom trail.
* Cloud Watch is for performance monitoring. Cloud Trail is for auditing.
* delivers log files to an Amazon S3 bucket
* CloudTrail is enabled by default.
* CloudTrail is per AWS account.
* You can consolidate logs from multiple accounts using an S3 bucket
* You can integrate CloudTrail with CloudWatch Logs to deliver data events captured by CloudTrail to a CloudWatch Logs log stream.
* CloudTrail log file integrity validation feature allows you to determine whether a CloudTrail log file was unchanged, deleted, or modified since CloudTrail delivered it to the specified Amazon S3 bucket.

Use case

* Track time when a particular event occurred in your account

Security and Compliance

Shared Responsibility Model

In the public cloud, there is shared responsibility between you and AWS

AWS responsibility- Security of the cloud

Your responsibility- Security in the cloud

AWS is responsible for ensuring malicious traffic does not impair network hardware.

You are responsible for ensuring malicious traffic does not reach your EC2 instance.

Security of the cloud

1. AWS Global Infrastructure

AWS is responsible for global infrastructure elements like Region, Availability Zone and Edge Locations

1. Building Security

AWS controls access to its data centers where your data resides

1. Networking Components

AWS maintains networking components: Generators, Uninterruptable Power Supply, Computer room air conditioning units, fire suppression systems, etc.

1. Software

AWS is responsible for any managed services like RDS, S3, lambda.

Security in the cloud

1. Application Data

You are responsible for managing your application data which includes encryption options.

1. Security Configuration

You are responsible for securing your account like rotating credentials, restricting Internet access to VPC.

1. Patching

You are responsible for guest operating system which includes updates and security patches.

1. Identity and Access Management
2. Network Traffic

Firewall configurations

1. Installed Software

Who is responsible for what?

1. Firewall configuration- You
2. Data center security for physical building- AWS
3. Encryption of EBS volumes- You
4. Language versions of lambda- AWS
5. Taking DB backups in RDS- You
6. Updating firmware on underlying EC2 hosts- AWS
7. Ensuring data is encrypted at rest- You
8. Ensuring network infrastructure- AWS
9. Patching guest OS for EC2- You
10. Physically destroying storage media at the end of file- AWS

Which security responsibilities are shared?

1. Patch Management
2. Configuration Management
3. Awareness and Training
4. Patch Management

AWS- Patching infrastructure

You- Patching guest OS and applications

1. Configuration Management

AWS- Configuring infrastructure devices

You- Configuring applications and databases

1. Awareness and Training

AWS- AWS employees

You- Your employees

Note

Customer

For EC2 this includes network level security (NACLs, security groups), operating system patches and updates, IAM user access management, and client and server side data encryption.

Inherited Controls

* Controls which a customer fully inherits from AWS.
* Physical and Environmental controls.

Shared Controls

* Examples of shared controls include:
  + **Patch Management** – AWS is responsible for patching and fixing flaws within the infrastructure, but customers are responsible for patching their guest OS and applications.
  + **Configuration Management** – AWS maintains the configuration of its infrastructure devices, but a customer is responsible for configuring their own guest operating systems, databases, and applications.
  + **Awareness & Training** – AWS trains AWS employees, but a customer must train their own employees.

Customer Specific Controls

* Controls which are solely the responsibility of the customer based on the application they are deploying within AWS services.
* Examples of customer specific controls include:
  + Service and Communications Protection or Zone Security

How do I report abuse of AWS resources?

1. Contact [abuse@amazonaws.com](mailto:abuse@amazonaws.com)
2. Contact AWS Trust and Safety Team using Report Amazon AWS Abuse Form

Temporary and Disposable resources

Resources that can be instantiated within seconds

**Design Principles**

**Scalability**

* Applications that are expected to grow over time need to be build on top of scalable architecture
* Horizontal Scaling
  + Add more instances as demand increases
  + No downtime
* Vertical Scaling
  + Add more CPU and/or RAM to existing instance as demand increases
  + Requires a restart

**Stateless application**

* A stateless application is an application that needs no knowledge of previous interactions and stores no session information.
* A stateless application can scale horizontally since any request can be serviced by any of the available compute resources (e.g., EC2 instances, AWS)
* DynamoDB is often used for storing session state to maintain a stateless architecture.

Stateful components:

* Databases are stateful.
* Many legacy applications are stateful.

**Disposable Resources Instead of Fixed Servers**

* Think of servers and other components as temporary resources.
* Launch as many as you need, and use them only for as long as you need them

**Instantiating compute resources**

* You don’t want to manually set up new resources
* Use automated, repeatable processes

Bootstrapping

* Execute automated bootstrapping actions to modify default configurations.

Golden Images

* Some resource types can be launched from a golden image.
* Examples are EC2 instances, RDS instances and EBS volumes.
* A golden image is a snapshot of a particular state for that resource.

Infrastructure as a Code:

* Allows you to provision resources by using code

**Automation**

Don’t do manually. Use automation.

Examples of automations using AWS services include:

1. AWS Elastic Beanstalk – the fastest and simplest way to get an application up and running on AWS.
2. Amazon EC2 Auto Recovery – You can create an Amazon CloudWatch alarm that monitors an Amazon EC2 instance and automatically recovers it if it becomes impaired.
3. Auto Scaling – With Auto Scaling, you can maintain application availability and scale your Amazon EC2 capacity up or down automatically according to conditions you define.
4. Amazon CloudWatch Alarms – You can create a CloudWatch alarm that sends an Amazon Simple Notification Service (Amazon SNS) message when a particular metric goes beyond a specified threshold for a specified number of periods.
5. Amazon CloudWatch Events – The CloudWatch service delivers a near real-time stream of system events that describe changes in AWS resources.
6. AWS OpsWorks Lifecycle events – AWS OpsWorks supports continuous configuration through lifecycle events that automatically update your instances’ configuration to adapt to environment changes.
7. AWS Lambda Scheduled events – These events allow you to create a Lambda function and direct AWS Lambda to execute it on a regular schedule.

**Loose Coupling**

* loosely coupled components.
* This means that IT systems should be designed in a way that reduces interdependencies—a change or a failure in one component should not cascade to other components.

**Services, Not Servers**

* Managed services
* Serverless architecture

**Databases**

**Relational Databases**

* Amazon RDS is a relational database service.

Scalability:

* Relational databases can scale vertically (e.g. upgrading to a larger RDS DB instance).
* For read-heavy use cases, you can scale horizontally using read replicas.

High Availability:

* For production DBs, Amazon recommend the use of RDS Multi-AZ which creates a synchronously replicated standby in another AZ.

Anti-Patterns:

* If your application primarily indexes and queries data with no need for joins or complex transactions consider a NoSQL database instead.
* If you have large binary files (audio, video, and image), it will be more efficient to store the actual files in S3 and only hold the metadata for the files in your database.

**NoSQL Databases**

* NoSQL is a term used to describe databases that trade some of the query and transaction capabilities of relational databases for a more flexible data model that seamlessly scales horizontally.
* DynamoDB is Amazon’s NoSQL database service.

Scalability:

* NoSQL database engines will typically perform data partitioning and replication to scale both the reads and the writes in a horizontal fashion.

High Availability:

* DynamoDB synchronously replicates data across three facilities in an AWS region for fault tolerance.

Anti-Patterns:

* If your schema cannot be denormalized and your application requires joins or complex transactions, consider a relational database instead.
* If you have large binary files (audio, video, and image), consider storing the files in Amazon S3 and storing the metadata for the files in your database.

**Data Warehouse**

* It can be used to combine transactional data from disparate sources making them available for analysis and decision-making.
* Amazon Redshift is a managed data warehouse service

Scalability:

* Amazon Redshift achieves efficient storage and optimum query performance through a combination of massively parallel processing (MPP), columnar data storage, and targeted data compression encoding schemes.

High Availability:

* Multi-node clusters replicate data to other nodes within the cluster.
* Data is continuously backed up to S3.
* RedShift continuously monitors the health of the cluster and re-replicates data from failed drives and replaces nodes as necessary.

Anti-Patterns:

* Because Amazon Redshift is a SQL-based relational database management system (RDBMS), it is compatible with other RDBMS applications and business intelligence tools.
* Although Amazon Redshift provides the functionality of a typical RDBMS, including online transaction processing (OLTP) functions, it is not designed for these workloads.

Search

Amazon CloudSearch and Amazon ES use data partitioning and replication to scale horizontally.

Both services provide features that store data redundantly across Availability Zones.

**Removing Single Point of Failure**

1. Introducing Redundancy
2. Detect failure
   * Automate failure detection and reacting to that failure
   * Services like ELB and Route 53 mask failure by routing traffic to available healthy point
3. Durable data storage
   * Replication can take place in a few different modes:
     + Synchronous replication – transactions are acknowledged only after data has been durably stored in both the primary and replica instance. Can be used to protect data integrity (low RPO) and scaling read capacity (with strong consistency).
     + Asynchronous replication – changes on the primary node are not immediately reflected on its replicas
     + Quorum-based replication – combines synchronous and asynchronous replication
4. Automated Multi Data Center Resilience
5. Fault Isolation and Horizontal Scaling
6. Optimize for cost
   * Or use services for which you can modify capacity as and when need. For example:
     + DynamoDB
     + RDS
     + Elasticsearch Service

Note

Eliminate the need for guest operating system access to production environments.

If the Spot market price increases above your bid price, your instance will be terminated automatically, and you will not be charged for the partial hour that your instance has run.

Well-Architectured Framework

5 pillars

1. Operational Excellence
2. Reliability
3. Security
4. Performance Efficiency
5. Cost Optimization
6. Operational Excellence
   * Build application that effectively supports production workload
   * Design principles for Operational Excellence
     + Plan for and anticipate failure
     + Deploy smaller and reversible changes
     + Script operations as code
     + Learn from failure and refine
   * Key Services for Operational Excellence
     + Preparation- AWS CloudFormation, AWS Config
     + Operations- AWS CloudFormation, AWS Config, AWS CloudTrail, AWS CloudWatch
     + Responses- AWS CloudFormation, AWS Config, AWS CloudTrail, Amazon CloudWatch
   * Real World- CodeCommit, CloudFormation
7. Reliability
   * Build application that works consistently and recovers quickly
   * Design principles for Reliability
     + Recover from failure automatically
     + Reduce idle resources
     + Scale horizontally for resilience
     + Manage change through automation
     + Test recovery procedures
   * Key services for reliability
     + Foundations- AWS IAM, Amazon VPC
     + Change Management- AWS CloudTrail, AWS Config
     + Failure Management- AWS Cloud Formation
   * Real World- RDS, Multi-AZ deployment
8. Security
   * Build application that protects system and data
   * Design principles for security
     + Automate security tasks
     + Encrypt data in transit and at rest
     + Assign only least privilege required
     + Track who did what and when
     + Ensure security at all application layers
   * Key services for security
     + Identity and Access Management- AWS IAM, MFA Token
     + Detective Controls- AWS CloudTrail, AWS Config, Amazon CloudWatch
     + Infrastructure Protection- Amazon VPC
     + Data Protection- Elastic Load Balancing, Amazon EBS, Amazon S3, Amazon RDS, AWS Key Management Service
     + Incident Response- AWS IAM, AWS CloudFormation
   * Real World-CloudTrail
9. Performance Efficiency
   * Build application that meets business and system requirements while removing bottlenecks
   * Design principles for performance efficiency
     + Democratize advanced technologies
     + Go global in minutes
     + Use serverless architectures first
     + Use multi-region deployments
     + Delegate tasks to cloud vendor
     + Experiment with virtual resources
   * Key Services for Performance Efficiency
     + Selection- Amazon EBS, Autoscaling, Amazon S3, Amazon RDS, Amazon DynamoDB
     + Review- Amazon CloudFormation, AWS Blog
     + Monitoring- Amazon CloudWatch, AWS Lambda
     + Trade-off- Amazon CloudFront, Amazon ElastiCache, AWS Snowball
   * Real World- Amazon Lambda
10. Cost Optimization
    * Build application at the least cost to the user
    * Design principles for Cost Optimization
      + Adopt a consumption model
      + Benefit from economies of scale
      + Stop spending money on data center operations
      + Analyze and attribute expenditure
      + Use managed services to reduce cost of ownership
      + Utilize consumption-based pricing
      + Implement Cloud Financial Management
      + Measure overall efficiency
      + Pay only for resources your application requires
    * Key Services for Cost Optimization
      + Cost-effective resources- Reserved Instances, AWS Trusted Advisor
      + Matched supply and demand- Auto scaling
      + Expenditure awareness- Amazon CloudWatch, Amazon SNS
    * Real World- S3 Intelligent-Tiering

Five pillars in real world

1. Operational Excellence- You can use AWS Code Commit, CloudFormation
2. Security- CloudTrail
3. Reliability- RDS, Multi AZs deployment
4. Performance Efficiency- Use AWS lambda
5. Cost Optimization- S3 Intelligent-Tiering

IAM (Identity and Access Management)

It is global service.

It controls authentication and authorization within an AWS account.

IAM is not used for application-level authentication.

Identity- who can access your resources

Access- what are the resources they can access

Identities

1. Root Users
2. Individual Users
3. Groups
4. Roles

Access

1. Policies
2. Customer managed Policies
3. AWS managed Policies
4. Permissions boundaries

Users

* Person or application who need to access AWS resources
* Application can also be users
* Each IAM user has three main components:
  + A username.
  + A password.
  + Permissions to access various resources.
* IAM users can be created to represent applications and these are known as “service accounts”.
* You can have up to 5000 users per AWS account.
* Each user account has a friendly name and an ARN which uniquely identifies the user across AWS.
* A unique ID is also created which is returned only when you create the user using the API, Tools for Windows PowerShell or the AWS CLI.
* The Access Key ID and Secret Access Key are not the same as a password and cannot be used to login to the AWS console.
* The Access Key ID and Secret Access Key can only be used once and must be regenerated if lost.
* A password policy can be defined for enforcing password length, complexity etc. (applies to all users).
* You can allow or disallow the ability to change passwords using an IAM policy.
* Access keys and passwords should be changed regularly.

Groups

* Group of users who have common access controls
* Groups are collections of users and have policies attached to them.
* A group is not an identity and cannot be identified as a principal in an IAM policy.
* You cannot nest groups (groups within groups).

Roles

Assumed, no long-term credential, temporary credential, federated account, can delegate using roles, no credential associated with roles

* An IAM role is an IAM identity that you can create in your account that has specific permissions.
* An IAM role is similar to an IAM 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 assumable by anyone who needs it.
* A role does not have standard long-term credentials such as a password or access keys associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session.
* Roles are created and then “assumed” by trusted entities and define a set of permissions for making AWS service requests.
* With IAM Roles you can delegate permissions to resources for users and services without using permanent credentials (e.g. user name and password).
* IAM users or AWS services can assume a role to obtain temporary security credentials that can be used to make AWS API calls.
* You can delegate using roles.
* There are no credentials associated with a role (password or access keys).
* IAM users can temporarily assume a role to take on permissions for a specific task.
* A role can be assigned to a federated user who signs in using an external identity provider.
* IAM roles with EC2 instances
  + IAM roles can be used for granting applications running on EC2 instances permissions to AWS API requests using instance profiles.
  + Only one role can be assigned to an EC2 instance at a time.
  + A role can be assigned at the **EC2 instance creation time or at any time afterwards.**

Policies

* You can manage permissions for user, group and role by creating policies document in JSON format and attaching it
* Policies are documents that define permissions and can be applied to users, groups and roles.
* Policy documents are written in JSON (key value pair that consists of an attribute and a value).
* All permissions are implicitly denied by default.
* The most restrictive policy is applied.
* The IAM policy simulator is a tool to help you understand, test, and validate the effects of access control policies.
* The Condition element can be used to apply further conditional logic.

IAM Credential Report

* Lists all users in your account and status of various credentials like passwords, access keys and MFA devices
* Used for auditing and compliance

STS (Security Token Service)

* The AWS Security Token Service (STS) is a web service that enables you to request temporary, limited-privilege credentials for IAM users or for users that you authenticate (federated users).

Federation with Mobile Apps

Use Facebook/Amazon/Google or other OpenID providers to login

Cross Account Access

Lets users from one AWS account access resources in another.

IAM provides the following features:

* Shared access to your AWS account.
* Granular permissions.
* Secure access to AWS resources for application that run on Amazon EC2.
* Multi-Factor authentication.
* Identity federation.
* Identity information for assurance.
* PCI DSS compliance.
* Integrated with many AWS services.
* Eventually consistent.
* Free to use.

You can work with AWS Identity and Access Management in any of the following ways:

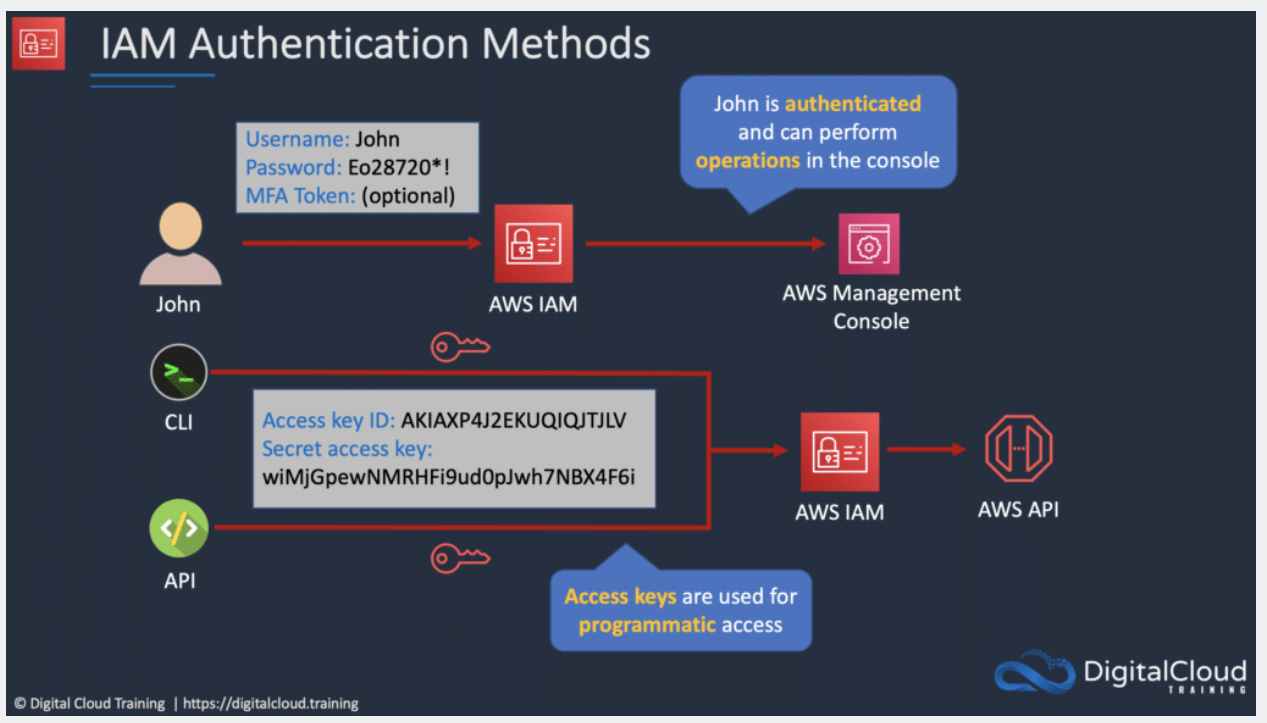
* AWS Management Console.
* AWS Command Line Tools.
* AWS SDKs.
* IAM HTTPS API.
* By default new users are created with NO access to any AWS services – they can only login to the AWS console. Permission must be explicitly granted to allow a user to access an AWS service.
* Identity Federation (including AD, Facebook etc.) allows secure access to resources in an AWS account without creating an IAM user account.
* Multi-factor authentication (MFA) can be enabled/enforced for the AWS account and for individual users under the account.
* MFA uses an authentication device that continually generates random, six-digit, single-use authentication codes.
* You can authenticate using an MFA device in the following two ways:
  + Through the **AWS Management Console** – the user is prompted for a user name, password and authentication code
  + Using the **AWS API** – restrictions are added to IAM policies and developers can request temporary security credentials and pass MFA parameters in their AWS STS API requests.
  + Using the **AWS CLI** by obtaining temporary security credentials from STS (aws sts get-session-token).
* IAM is universal (global) and does not apply to regions.
* IAM replicates data across multiple data centers around the world.
* It is a best practice to avoid using the root account for anything other than billing.
* Power user access allows all permissions except the management of groups and users in IAM.
* Temporary security credentials consist of the AWS access key ID, secret access key, and security token.

## **Authentication Methods**

* Console password
  + A password that the user can enter to sign in to interactive sessions such as the AWS Management Console.
  + You can allow users to change their own passwords.
  + You can allow selected IAM users to change their passwords by disabling the option for all users and using an IAM policy to grant permissions for the selected users.
* Access Keys
  + A combination of an **access key ID** and a **secret access key.**
  + You can assign two active access keys to a user at a time.
  + These can be used to make programmatic calls to AWS when using the**API** in program code or at a command prompt when using the **AWS CLI** or the **AWS PowerShell** tools.
  + You can create, modify, view or rotate access keys.
  + The secret access is returned only at creation time and if lost a new key must be created.
  + Ensure access keys and secret access keys are stored securely
* Server certificates:
  + SSL/TLS certificates that you can use to authenticate with some AWS services.
  + AWS recommends that you use the AWS Certificate Manager (ACM) to provision, manage and deploy your server certificates.

**IAM Best Practices**

* Lock away the AWS root user access keys.
* Create individual IAM users.
* Use AWS defined policies to assign permissions whenever possible.
* Use groups to assign permissions to IAM users.
* Grant least privilege.
* Use access levels to review IAM permissions.
* Configure a strong password policy for users.
* Enable MFA.
* Use roles for applications that run on AWS EC2 instances.
* Delegate by using roles instead of sharing credentials.
* Rotate credentials regularly.
* Remove unnecessary credentials.
* Use policy conditions for extra security.
* Monitor activity in your AWS account.



Application Security Services

1. WAF (Web Application Firewall)
2. Shield
3. Macie
4. WAF (Web Application Firewall)

Introduction

* Protects your application against common web attacks
* Protects your application against SQL Injection
* Protects your application against cross-site scripting
* WAF lets you create rules to filter web traffic based on conditions that include IP addresses, HTTP headers and body, or custom URIs.
* The rules are known as Web ACLs.

Cross-site scripting

It is which is when a hacker targets with malicious scripts that execute in your web browser that will trick you into sharing private information

DDoS (Distributed Denial of Service)

Traffic jam on a website or web application in an attempt to cause it crash

1. Shield

Introduction

* DDoS protection service
* Shield Standard is free. Shield Advanced is paid.
* DDoS Protection via Shield Advanced is supported on several services like CloudFront, AWS Global Accelerator, Route 53, Elastic Load Balancing

1. Macie

Introduction

* Helps you discover and protect sensitive data
* Uses machine learning behind the scene
* Identify Personally Identifiable Information (PII)

Use case

* Discover passport number stored on S3

Additional Security Services

1. Config
2. GuardDuty
3. Inspector
4. Artifact
5. Config

Introduction

* Allows you to access, audit and evaluate configuration of your resources
* Track configuration changes over time
* Send notifications for every configuration changes
* Delivers configuration history file to S3

Use case

* Identify system level configuration changes made to your EC2 instance

1. GuardDuty

Introduction

* Intelligent threat detection system that uncovers unauthorized behavior
* Uses machine learning behind the scene
* It has built-in detection for EC2, S3, IAM
* Review CloudTrail, VPN Flow Logs, DNS logs. i.e. GuardDuty continuously monitor CloudTrail logs, VPC Flow Logs, DNS Logs to uncover unauthorized behavior.
* Detects account compromise, instance compromise, malicious reconnaissance, and bucket compromise.

Use case

* Detect unusual API calls in your account

1. Inspector

Introduction

* Works with EC2 instance to uncover unauthorized behavior
* Instances must be tagged.

Use case

* Identify unintended network access to an EC2 instance

1. Artifact

Introduction

* Offers an on-demand access to AWS Security and Compliance Report
* It is a central repository for compliance report from third party auditors like SOC (Service Organization Controls) and PCI reports (Payment Card Industry)
* Agreements available in AWS Artifact include the Business Associate Addendum (BAA) and the Nondisclosure Agreement (NDA).

Use case

* You need to access AWS Certification for ISO Compliance

Security, Identity and Compliance

* 1. **Amazon Cognito**
  2. **AWS Certificate Manager**
  3. **AWS CloudHSM**
  4. **AWS Directory Service**

**Amazon Cognito**

* Amazon Cognito lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily.
* Amazon Cognito scales to millions of users and supports sign-in with social identity providers, such as Facebook, Google, and Amazon, and enterprise identity providers via SAML 2.0.

**AWS Certificate Manager**

* AWS Certificate Manager is a service that lets you easily provision, manage, and deploy public and private (SSL/TLS) certificates for use with AWS services and your internal connected resources
* AWS Certificate Manager removes the time-consuming manual process of purchasing, uploading, and renewing SSL/TLS certificates

**AWS CloudHSM**

* Generate and store encryption keys
* AWS manage keys
* Dedicated hardware

**AWS Directory Service**

AWS Directory Service for Microsoft Active Directory, also known as AWS Managed Microsoft AD, enables your directory-aware workloads and AWS resources to use managed Active Directory in the AWS Cloud.

AWS Managed Microsoft AD is built on actual [Microsoft Active Directory](https://aws.amazon.com/directoryservice/active-directory/) and does not require you to synchronize or replicate data from your existing Active Directory to the cloud.

You can use standard Active Directory administration tools and take advantage of built-in Active Directory features, such as Group Policy and single sign-on (SSO).

With AWS Managed Microsoft AD, you can easily join [Amazon EC2](https://aws.amazon.com/ec2/) and [Amazon RDS for SQL Server](https://aws.amazon.com/rds/sqlserver/) instances to your domain, and use [AWS Enterprise IT applications](https://aws.amazon.com/enterprise-applications/) such as [Amazon WorkSpaces](https://aws.amazon.com/workspaces/) with Active Directory users and groups.

**AWS Artifact:**

AWS Artifact is your go-to, central resource for compliance-related information that matters to you.

It provides on-demand access to AWS’ security and compliance reports and select online agreements.

Reports available in AWS Artifact include our Service Organization Control (SOC) reports, Payment Card Industry (PCI) reports, and certifications from accreditation bodies across geographies and compliance verticals that validate the implementation and operating effectiveness of AWS security controls.

Agreements available in AWS Artifact include the Business Associate Addendum (BAA) and the Nondisclosure Agreement (NDA).

# Management Tools

* 1. AWS CloudFormation
  2. AWS OpsWorks
  3. AWS Config
  4. AWS Service Catalog
  5. AWS System Manager
  6. AWS Managed Services

## AWS Service Catalog

### Introduction

* Allows organization to create and manage catalog of IT services that are provided for us on AWS
* Enables users to quickly deploy only the approved IT services they need

Data Encryption and Secret Management Services

1. KMS (Key Management Service)
2. Cloud HSM (Hardware Security Model)
3. Secrets Manager
4. KMS (Key Management Service)

Introduction

* Allows you to generate and store encryption keys
* Automatically enabled for certain services like CloudTrail logs, S3 Glacier, Storage Gateway
* AWS KMS is integrated with AWS CloudTrail which provides you the ability to audit who used which keys, on which resources, and when.
* Both symmetric and asymmetric keys

Use case

* Create encrypted Amazon EBS Volumes

1. Cloud HSM

Introduction

* Allows you to generate encryption keys
* Provides you with encryption hardware so that you can generate and manage keys. AWS does not even have access to keys when you use Cloud HSM.
* With CloudHSM, you can manage your own encryption keys using FIPS 140-2 Level 3 validated HSMs.

Use case

* Meet compliance requirements for data security by using dedicated hardware

1. Secrets Manager

Introduction

* Rotate, manage, and retrieve secrets like database credentials, password, API Keys, etc.
* Integrates with services like RDS, DocumentDB and RedShift
* Similar to Parameter Store.
* Allows native and automatic rotation of keys.
* Central auditing for secret rotation.

Use case

* Retrieve database credential needed for your application code

Note

KMS- AWS manages keys

Cloud HSM- You manage keys generated with Cloud HSM

# Pricing, Billing and Governance

AWS Pricing

There is no termination fee.

3 fundamental drivers of cost

1. Compute
2. Storage
3. Outbound data transfer

Note

Normally, there is no cost for inbound data transfer.

Pricing policies

1. Pay as you go
2. Pay less when you reserve
3. Pay even less per unit when using more
4. Pay even less as AWS grows
5. Custom pricing (enterprise customers only)

Free services

1. Amazon VPC
2. Elastic Beanstalk (but not the resources created)
3. CloudFormation (but not the resources created)
4. Identity Access Management (IAM)
5. Auto Scaling (but not the resources created)
6. OpsWorks
7. Consolidated Billing

Free Offer Types

There are 3 different free offers available depending on the service you choose

1. 12 months free
2. Always free
3. Trails
4. 12 months free
   * 12 months free usage starting your first sign-up date to AWS.
   * When 12 months expire or your application usage exceeds the tier, pay as you go.
5. Always free
   * Never expires
   * Available to all AWS customers
6. Trails
   * Short-term free trails starting from date you activate a particular service

EC2 Pricing

EC2 pricing is based on:

1. Clock hours of server uptime
2. Instance configuration
3. Instance type
4. Number of instances
5. Operating systems and software packages
6. Auto Scaling (resources created)
7. Elastic IP addresses (charged if allocated but not used)
8. Load balancing
9. Detailed monitoring
10. Data Storage

EC2 Pricing Model

1. On demand
2. Spot
3. Reserved Instance
4. Dedicated Hosts
5. Savings Plans
6. On demand

* Fixed price in which you are billed based on instance type used
* You pay for what you use
* No contract
* Use on-demand instances when
  + You cannot make upfront payment or long-term commitment
  + Your applications have unpredictable workloads that cannot be interrupted
  + Your applications are under-development
  + Your applications will not run for more than a year
* You pay for compute capacity per hour or per second

1. Spot

* Spot instance lets you take advantage of unused EC2 capacity
* Your request is available only if capacity is available
* Use Spot instance when
  + Your application is not concerned about start and stop time of execution
  + Your workload can be interrupted
* You can save upto 90% off on-demand prices
* You pay for spot that is in effect at the beginning of each hour
* Instances still may be terminated (with a 2 minute warning) when EC2 needs the capacity back

1. Reserved Instance

* Allows you to commit to a specific instance type for 1 or 3 years
* Use reserved instance when
  + Your application has steady state usage and you can commit for 1 or 3 years
  + You can make upfront payment
  + Your application requires capacity reservation
* You can save upto 75% off on-demand prices
* You are required to sign a contract of 1 or 3 years
* You can pay all-upfront, partial upfront or no upfront. All upfront for maximum term earns the highest discount

1. Dedicated Hosts

* Allows you to pay for physical server that is dedicated to running your instances. No other AWS customers is going to have applications running on this server. The server basically belongs to you
* Use dedicated hosts when
  + You want to bring your own server-bound software license from vendors like Microsoft and Oracle
  + You have regulatory or corporate compliance requirements around tenancy model
* You can save upto 70% off on-demand prices
* You can bring your existing per-core, per-socket and per-VM software licenses.

1. Savings Plan

* You do not commit for specific instance type instead you commit to compute usage (measured per hour) for 1 to 3 years
* Use savings plan when
  + You want to lower your bill across multiple compute services
  + You want flexibility to change compute services, instance types, OS or regions
* You can save upto 72% off on-demand prices
* Savings can be shared across various compute services like EC2, Fargate and lambda
* Pay by $/hour

Reservations apply to various services, including:

1. Amazon EC2 Reserved Instances
2. Amazon RDS Reserved Instances
3. Amazon DynamoDB Reserved Capacity
4. Amazon ElastiCache Reserved Nodes
5. Amazon RedShift Reserved Instances

Lambda Pricing

1. Compute time
2. Request
3. Always free
4. Compute time

* Time between start of execution and end of execution

1. Request

* A request is counted each time it starts execution
* Test invokes on the console is also considered as a request

1. Always free

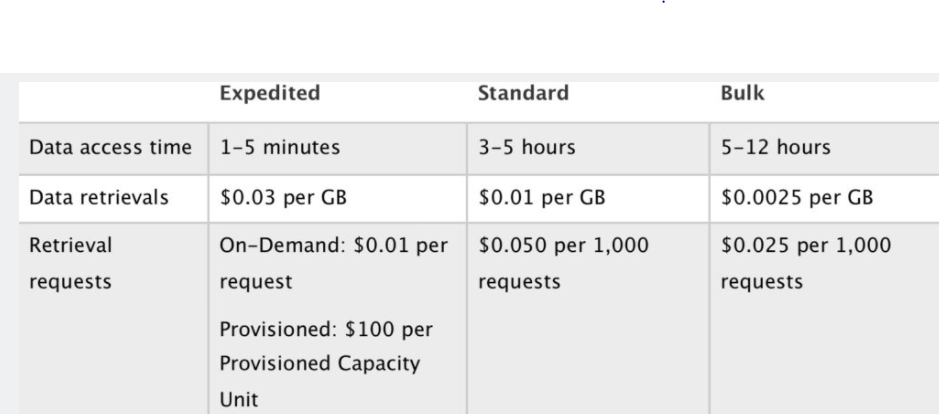
* 1 million free requests per month (even after free usage tier expires)

S3 Pricing

1. Storage class
2. Storage quantity – per GB basis
3. Data transfer
4. Request and data retrieval
5. Lifecycle transition request

Amazon Glacier pricing

* Extremely low cost and you pay only for what you need with no commitments of upfront fees.
* Charged for requests and data transferred out of Glacier.
* “Amazon Glacier Select” pricing allows queries to run directly on data stored on Glacier without having to retrieve the archive. Priced on amount of data scanned, returned, and number of requests initiated.
* Three options for access to archives, listed in the table below:



AWS Snowball Pricing

* Pay a service fee per data transfer job and the cost of shipping the appliance.
* Each job allows use of Snowball appliance for 10 days onsite for free.
* Data transfer into AWS is free and outbound is charged (per region pricing).

RDS Pricing

1. Running clock hour- from the time you launch a database instance to the time you terminate it
2. Database Count- Number of database instance
3. Purchase type- It is on demand database instance or reserved database instance
4. Storage
5. Type of database
6. API Calls- calls from dashboard
7. Requests
8. Deployment Type- is it deployed to single AZ or multiple AZs
9. Data transfer- Inbound data transfer is free while Outbound data transfer is paid.

Amazon CloudFront Pricing

CloudFront pricing is determined by:

* 1. **Traffic distribution** – data transfer and request pricing, varies across regions, and is based on the edge location from which the content is served.
  2. **Requests** – the number and type of requests (HTTP or HTTPS) and the geographic region in which they are made.
  3. **Data transfer out** – quantity of data transferred out of CloudFront edge locations.
  4. There are additional chargeable items such as invalidation requests, field-level encryption requests, and custom SSL certificates.

Amazon Elastic Block Store (EBS) Pricing

Pricing is based on three factors:

* 1. Volumes – volume storage for all EBS volumes type is charged by the amount of GB provisioned per month.
  2. Snapshots – based on the amount of space consumed by snapshots in S3
  3. Data transfer – inbound data transfer is free, outbound data transfer charges are tiered.
* Amount of data storage provisioned
* Provisioned iops

Amazon DynamoDB Pricing

Charged based on:

1. **Provisioned throughput (write)**
2. **Provisioned throughput (read)**
3. **Indexed data storage.**
4. **Data transfer** – no charge for data transfer between DynamoDB and other AWS services within the same region, across regions is charged on both sides of the transfer.
5. **Global tables** – charged based on the resources associated with each replica of the table (replicated write capacity units, or rWCUs).
6. **Reserved Capacity** – option available for a one-time upfront fee and commitment to paying a minimum usage level at specific hourly rates for the duration of the term. Additional throughput is charged at standard rates.

**On-demand capacity mode:**

1. Charged for reads and writes
2. No need to specify how much capacity is required
3. Good for unpredictable workloads

**Provisioned capacity mode:**

1. Specify number of reads and writes per second
2. Can use Auto Scaling
3. Good for predictable workloads

Resource Groups and Tagging

* Tags are key / value pairs that can be attached to AWS resources.
* Tags contain metadata (data about data).
* Tags can sometimes be inherited – e.g. resources created by Auto Scaling, CloudFormation or Elastic Beanstalk.
* Resource groups make it easy to group resources using the tags that are assigned to them. You can group resources that share one or more tags.

Consolidated billing

* AWS organizations allows you to consolidate multiple AWS accounts into an organization that you create and centrally manage.
* Includes root accounts and organizational units.
* Policies are applied to root accounts or OUs.
* Consolidated billing includes:
  + Paying Account – independent and cannot access resources of other accounts.
  + Linked Accounts – all linked accounts are independent.
* Consolidated billing is offered at no additional cost.
* Limit of 20 linked accounts (by default).
* One bill for multiple AWS accounts.
* Billing alerts enabled on the Paying account include data for all Linked accounts (or can be created per Linked account).
* Consolidated billing allows you to get volume discounts on all of your accounts.
* Unused reserved instances (RIs) for EC2 are applied across the group.
* CloudTrail is on a per account basis and per region basis but can be aggregated into a single bucket in the paying account.
* Best practices:
  + Always enable multi-factor authentication (MFA) on the root account.
  + Always use a strong and complex password on the root account.
  + The Paying account should be used for billing purposes only. Do not deploy resources into the Paying account.

AWS Quick Starts

* Quick Starts are built by [AWS architects](https://digitalcloud.training/certification-training/aws-solutions-architect-associate/) and partners to help you deploy popular solutions on AWS, based on AWS best practices for security and high availability.
* These reference deployments implement key technologies automatically on the AWS Cloud, often with a single click and in less than an hour.
* Leverages CloudFormation.

**AWS Price List API**

* Query the prices of AWS services.
  + Price List Service API (AKA the Query API) – query with JSON.
  + AWS Price List API (AKA the Bulk API) – query with HTML.

TCO (Total Cost of Ownership)

Introduction

* Financial estimate that helps you to understand direct and indirect cost of AWS
* There used to be TCO Calculator which is no longer available

Pricing Calculator

Introduction

* Calculate TCO
* Provided estimate of AWS fees and charges
* It helps you to explore services based on your use case and find instance type that best fits your needs
* Has quick estimate and advanced estimate

Application Discovery Service

Introduction

* Helps you to plan migration projects to AWS Cloud
* Used to estimate TCO
* Works with other services to migrate servers

Billing Services

1. Budgets
2. Cost and Usage Report
3. Cost Explorer
4. Budgets

Introduction

* Allows you to set custom budgets that alert you when your costs or usage exceed your threshold

Budget Types

1. Cost budgets- how much you want to spend on one or more services
2. Usage budgets- how much you want to use on one or more services
3. Reservation budgets

Use case

* Monitor free usage so you don’t incur unwanted costs

1. Cost and Usage Report

Introduction

* View most granular data about your AWS bill
* If you receive huge bill, go here and download detailed report
* You can aggregate usage data on a daily, hourly or monthly level

1. Cost Explorer

Introduction

* Allows you to visualize and forecast cost and usage over time
* View past 12 months
* Forecast upto 3 months

Use case

* Analyze your EC2 usage over the past 7,30 or 60 days

Governance Services

1. Organizations
2. Control Tower
3. System Manager
4. Trusted Advisor
5. Organizations

Introduction

* Allows you to centrally manage AWS accounts under one umbrella
* Single payment can be made for all accounts
* Group multiple account

Benefits

* Consolidated Billing- one bill for multiple accounts
* Cost Savings- reduce cost by sharing resources across accounts
* Account Governance- you have a quick and automated way to create accounts and invite existing accounts to join your organization

1. Control Tower

Introduction

* Sits on top of organization and ensures your accounts conform to company wide policies
* Provides dashboard to manage accounts
* Examples of guardrails AWS Control Tower can configure for you include:
  + Disallowing public write access to Amazon Simple Storage Service (Amazon S3) buckets.
  + Disallowing access as a root user without multi-factor authentication.
  + Enabling encryption for Amazon EBS volumes attached to Amazon EC2 instances.

Use case

* Disallow public write access to all S3 buckets across your accounts

1. System Manager

Introduction

* Get operational insights about the state of your infrastructure
* Helps you manage EC2 and on-premises systems at scale
* Free service
* Works for both Windows and Linux Operating System
* Patching automation
* Integrated with AWS Config
* Integrated with CloudWatch metrics/ dashboards
* With Systems Manager, you can group resources, like [Amazon EC2](https://aws.amazon.com/ec2/) instances, [Amazon S3](https://aws.amazon.com/s3/) buckets, or [Amazon RDS](https://aws.amazon.com/rds/) instances, by application, view operational data for monitoring and troubleshooting, and take action on your groups of resources.
* Systems Manager simplifies resource and application management, shortens the time to detect and resolve operational problems, and makes it easy to operate and manage your infrastructure securely at scale.

Features

* Parameter Store

Actions

* Automation (Shut down EC2, create AMI)
* Run Command
* Maintenance Windows
* System Manager
* Patch Manager
* Session Manager

Use case

* Deploy OS and software patches automatically across a group of instances

1. Trusted Advisor

Introduction

* Provides real-time guidance to help you provision your resources following AWS best practices
* Helps you to understand AWS best practices
* Checks your account and makes recommendations
* Helps you to see service limits
* It has service limit dashboard that helps you monitor service limits (50 service limit checks)
* 7 sets of trusted advisor checks (basic and developer plan)
  + S3 Bucket Permissions
  + Security Groups – Specific Ports Unrestricted
  + IAM Use
  + MFA on Root Account
  + EBS Public Snapshots
  + RDS Public Snapshot
* Trusted Advisor scans your AWS infrastructure and compares is to AWS best practices in five categories:
  + Cost Optimization
  + Performance
  + Security
  + Fault Tolerance.
  + Service Limits

Use case

* Check read and write capacity service limits for DynamoDB

Management Services

1. Managed Service
2. Professional Services
3. AWS Partner Network
4. Marketplace
5. Managed Service

Introduction

* Set of services and tools that automate infrastructure management task for AWS deployment
* EC2 is managed service
* Reduce operational overhead

1. Professional Services

Introduction

* It allows customers to migrate from on-premises to cloud
* Proposes solution
* Architects solution
* Implements solution

Use case

* Get help with evaluating an application for migration to the cloud

1. AWS Partner Network

Introduction

* Global community of partners who leverage AWS to build solutions and services for customers
* Technology partners provide software solutions.
* Consulting partner provides professional services. (want someone for deployment)

Use case

* You need help designing and building a new application. If your team lacks technical expertise to build and deploy cloud applications

1. Marketplace

Introduction

* Buy third party solutions/software
* Sell solutions to AWS customers

Use case

* Try out an application before making a long-term commitment

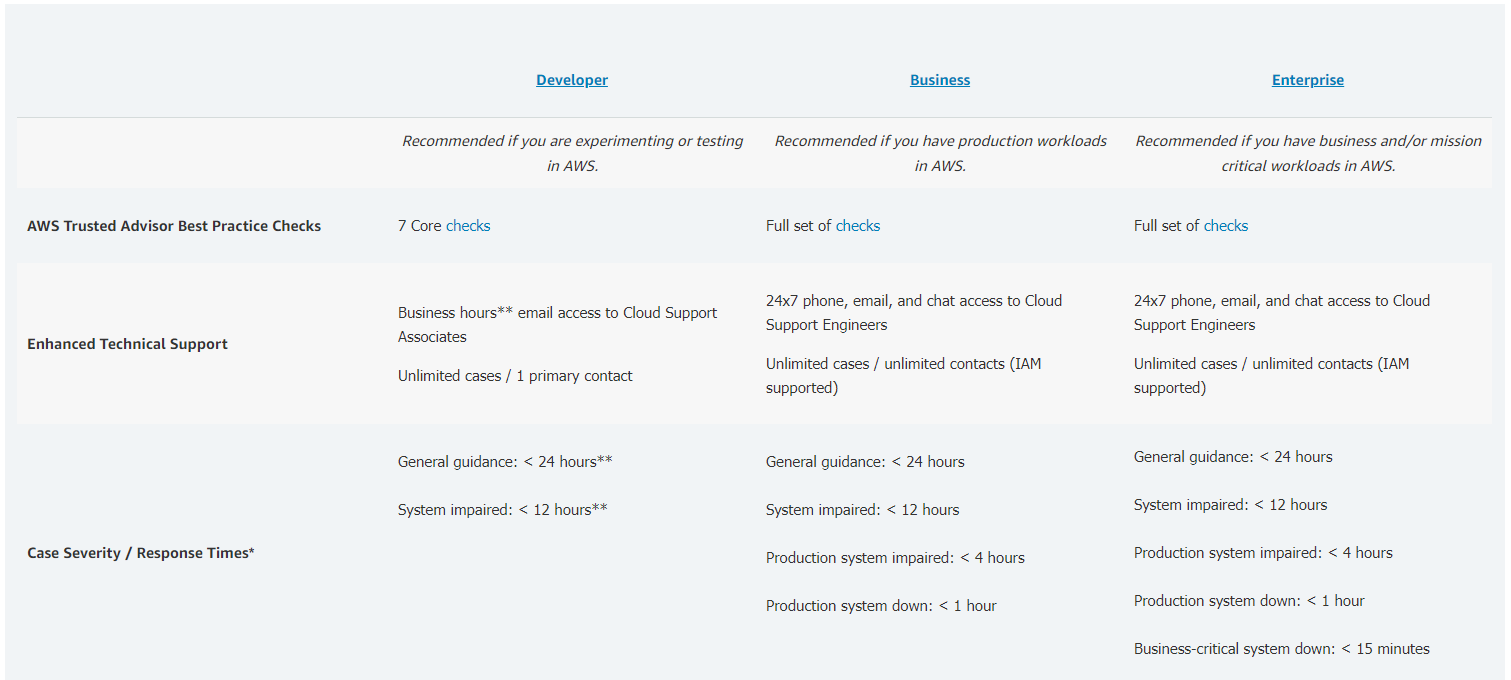
Note

AWS finds partners and providers from APN and Marketplace.

Support Plans

Four Support Plans

1. Basic
2. Developer
3. Business
4. Enterprise



1. Basic
   * Free for all AWS accounts
   * Gives you access to account and billing support case
   * Gives you access to service limit increase support case
   * Does not give you access to technical support case
   * You have access to customer support 24X7 via email only
   * You have access to Discussion Forum
   * You have access to 7 sets of Trusted Advisor Checks
   * Basic Support is included for all AWS customers and includes:
     1. Customer Service and Communities - 24x7 access to customer service, [documentation](https://docs.aws.amazon.com/), [whitepapers](https://aws.amazon.com/whitepapers/?whitepapers-main.sort-by=item.additionalFields.sortDate&whitepapers-main.sort-order=desc), and [support forums](https://forums.aws.amazon.com/index.jspa) (discussion forums).
     2. [AWS Trusted Advisor](https://aws.amazon.com/premiumsupport/technology/trusted-advisor/) - Access to the 7 core Trusted Advisor checks
     3. [AWS Personal Health Dashboard](https://aws.amazon.com/premiumsupport/technology/personal-health-dashboard/) - A personalized view of the health of AWS services, and alerts when your resources are impacted.
2. Developer
   * Starts at $29 per month
   * Recommended for development and testing purpose
   * Gives you access to account and billing support case
   * Gives you access to service limit increase support case
   * Gives you access to technical support case
   * You have access to one primary contact
   * You can open unlimited support cases
   * You have access to Cloud Support Associate during business hours via emails only and in this time they guarantee response time
     1. <24 hours for general guidance
     2. <12 hours if your system is impaired
   * You have access to 7 sets of Trusted Advisor Checks
3. Business
   * Starts at $100 per month
   * Recommended for production workloads
   * Gives you access to account and billing support case
   * Gives you access to service limit increase support case
   * Gives you access to technical support case
   * You have access to unlimited primary contact
   * You can open unlimited support cases
   * You have access to Cloud Support Engineer 24X7 via email, phone or chat and in this time they guarantee response time
     1. <24 hours for general guidance
     2. <12 hours if your system is impaired
     3. <4 hours if production system is impaired
     4. <1 hour if production system is down
   * You have access to full set of Trusted Advisor Checks
4. Enterprise
   * Starts at $15,000 per month
   * Recommended for business and mission-critical production workloads
   * Gives you access to account and billing support case
   * Gives you access to service limit increase support case
   * Gives you access to technical support case
   * You have access to unlimited primary contact
   * You can open unlimited support cases
   * You have access to Cloud Support Engineer 24X7 via email, phone or chat and in this time they guarantee response time
     1. <24 hours for general guidance
     2. <12 hours if your system is impaired
     3. <4 hours if production system is impaired
     4. <1 hour if production system is down
     5. <15 minutes if business critical system is down
   * You have access to full set of Trusted Advisor Checks
   * You have access to Technical Account Manager (TAM)
   * You have access to Concierge Support Team
   * You have access to Infrastructure Event Management

Support Case Type

There are three types of support cases you can open with AWS account

1. Account and Billing- Can be opened by all customers
2. Service Limit Increase- Can be opened by all customers
3. Technical Support- Can only be opened by customers on Developer, Business and Enterprise Plan

Note

AWS Support does not allow cases for code development, debugging software or performing system administrative tasks.

# AWS Cloud Management

## AWS Service Catalog

* Allows organization to create and manage catalog of IT services that are provided for us on AWS
* Enables users to quickly deploy only the approved IT services they need

## AWS Personal Health Dashboard

* AWS Personal Health Dashboard provides alerts and remediation guidance when AWS is experiencing events that may impact you.
* Personal Health Dashboard gives you a personalized view into the performance and availability of the AWS services underlying your AWS resources.
* Can integrate with Amazon CloudWatch Events, enabling you to build custom rules and select targets such as AWS Lambda functions to define automated remediation actions.

## Service Health Dashboard

* AWS publishes up-to-the-minute information on service availability.
* This information is not personalized to you (unlike Personal Health Dashboard).

Security

Compliance

Compliance Programs

* Certifications / attestations.
* Laws, regulations, and privacy.
* Alignments / frameworks.

Penetration Testing

* Penetration testing is the practice of testing one’s own application’s security for vulnerabilities by simulating an attack.
* There is a limited set of resources on which penetration testing can be performed.
* You do not need permission to perform penetration testing against the following services:
* Amazon EC2 instances, NAT Gateways, and Elastic Load Balancers
* Amazon RDS
* Amazon CloudFront
* Amazon Aurora
* Amazon API Gateways
* AWS Lambda and Lambda Edge functions.
* Amazon Lightsail resources.
* Amazon Elastic Beanstalk environments.

AWS Single Sign-On (AWS SSO)

* AWS Single Sign-On is a cloud-based single sign-on (SSO) service that makes it easy to centrally manage SSO access to all of your AWS accounts and cloud applications.
* AWS SSO includes a user portal where your end-users can find and access all their assigned AWS accounts, cloud applications, and custom applications in one place.

Amazon Cognito

* Amazon Cognito lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily.
* Amazon Cognito scales to millions of users and supports sign-in with social identity providers, such as Apple, Facebook, Google, and Amazon, and enterprise identity providers via SAML 2.0 and OpenID Connect.
* The two main components of AWS Cognito are user pools and identity pools:
  + User pools are user directories that provide sign-up and sign-in options for your app users.
  + Identity pools enable you to grant your users access to other AWS services.
* You can use identity pools and user pools separately or together.

WS Directory Services

* AWS provide a number of directory types.
* The following three types currently feature on the exam and will be covered on this page:
  + Active Directory Service for Microsoft Active Directory.
  + Simple AD.
  + AD Connector.
* As an alternative to the AWS Directory service you can build your own Microsoft AD DCs in the AWS cloud (on EC2).

AWS Systems Manager Parameter Store

* Provides secure, hierarchical storage for configuration data management and secrets management.
* You can store data such as passwords, database strings, and license codes as parameter values.
* You can store values as plaintext (unencrypted data) or ciphertext (encrypted data).
* You can then reference values by using the unique name that you specified when you created the parameter.