Table of contents

What are Traditional IT Services Like?

Imagine you are producing an Idea with new technology, and you need three servers.This sparts you to order for:

- Hardware, Power, Cooling , Space
- Operational and Security installation
- Access granted to the requester
- Software installation by the requester.

This can take day or weeks or months . Any you will have to take care of:

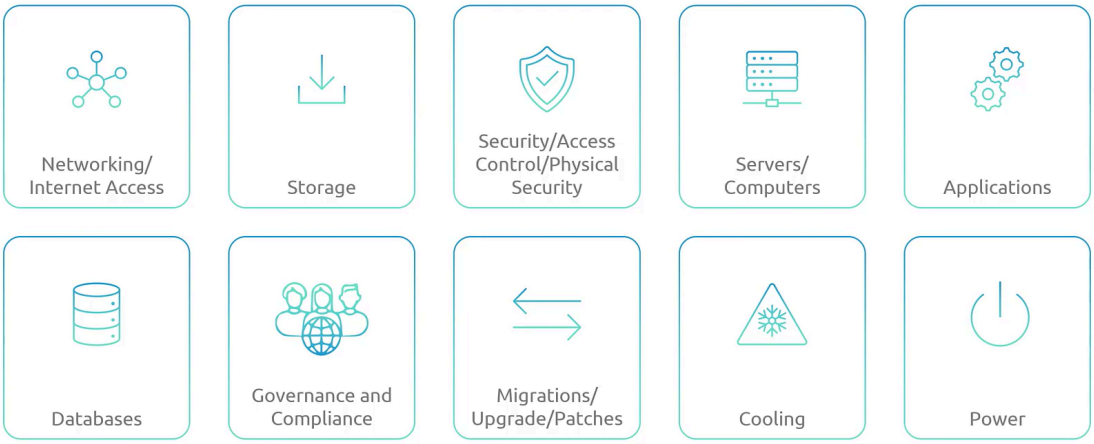


image.png

Pros to Traditional IT Services

- Improved Security
- Greater Customization
- More Control

Cons Of Traditional IT Services

- Increased cost / Poor ROI
- Limited scalability
- Responsible for everything
- Limited location
- More personnel
- Long Provisioning times

Clear the Cons outweigh the pros. What is the Solution?

Cloud Computing

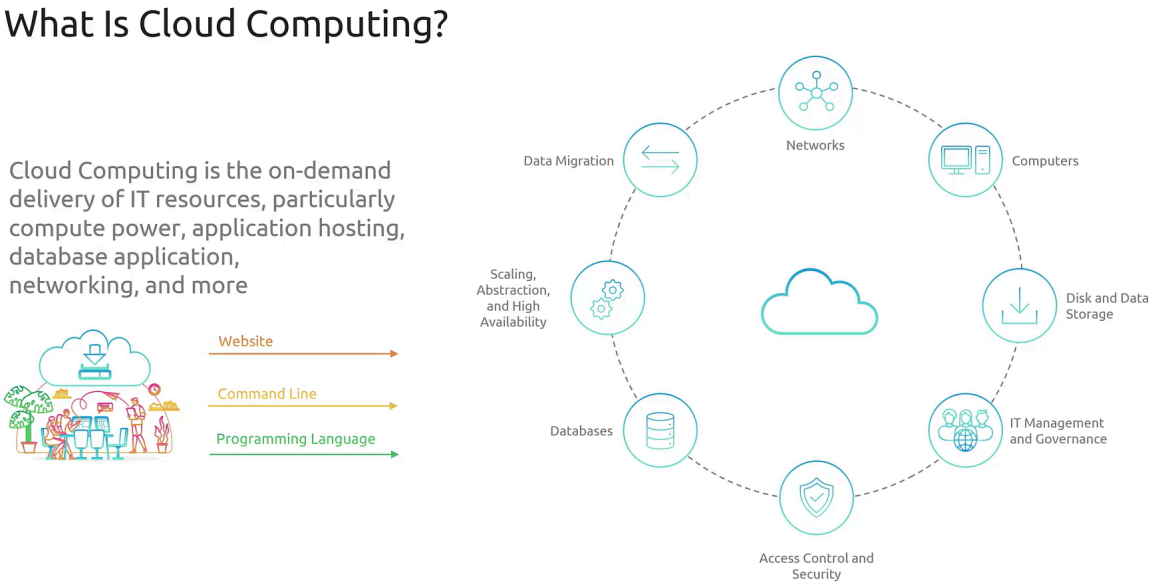


image_1.png

For example if you want a Server to run a Database or Application , all you have to do is make a request using the AWS console or AWS CLI or API calls and a response will be given back.

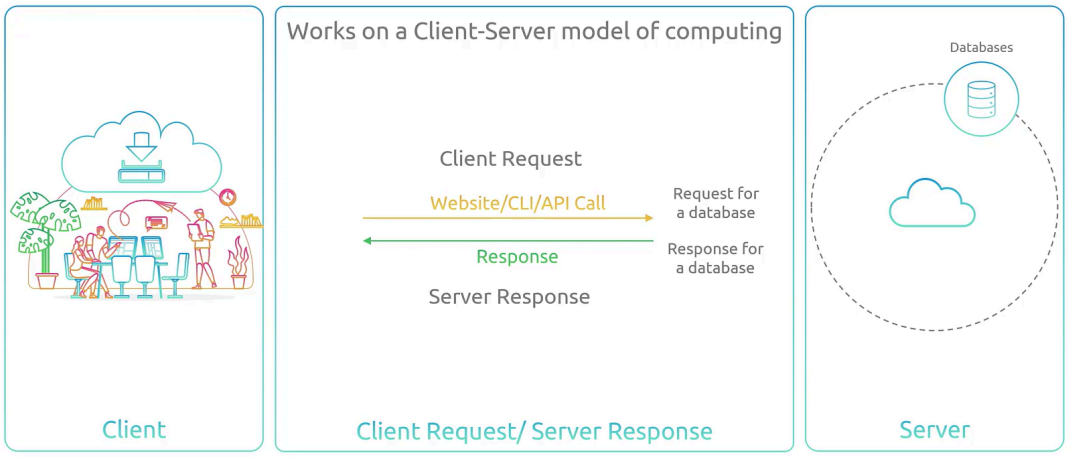


image_2.png

Here again you will pay for what you request or use.

You will access your requested resources in seconds or minutes or even instantly.

How does Cloud Computing Help? All the items in orange are taken care of by the cloud

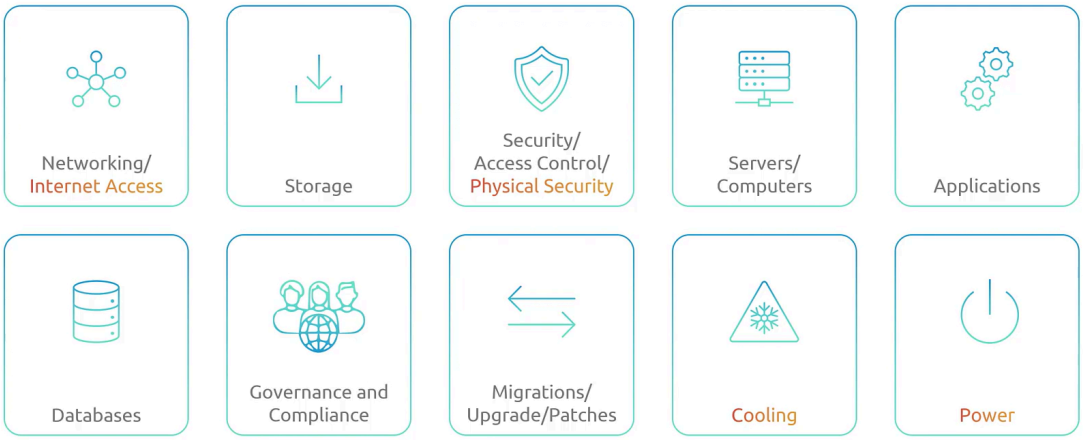


image_3.png

Cloud Computing Models

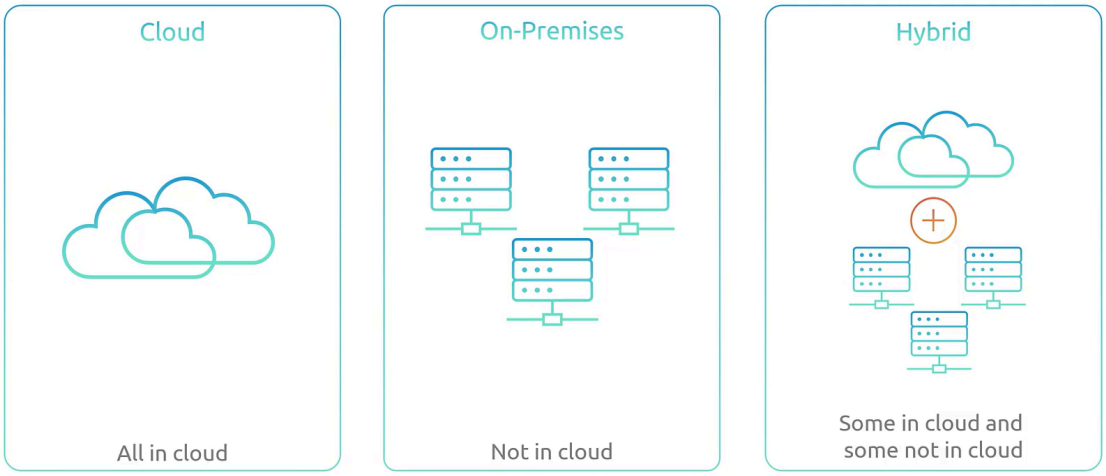


image_4.png

Cloud Model

- Usually includes startup of all sizes.
- Runs everything " in the cloud"
- Shared Responsibility for security and operations

On Premises Model

- Minimal to no cloud computing usage
- All projects run in a owned data center or a rented data center
- Responsible for all security and operations
- Can be legacy companies that haven't had a reason to move to the cloud
- Can be companies that need strict control and security over the entire infrastructure e.g. Companies dealing with Government Data .

Hybrid Model

- Runs some parts of the application in cloud; other parts are on-premise

- Migrates existing applications to the cloud; may leave legacy applications on-premise
- Most new applications are designed and built for the cloud
- Usually fast connection between on-premise and cloud resources

Amazon Web Services Versus Traditional IT

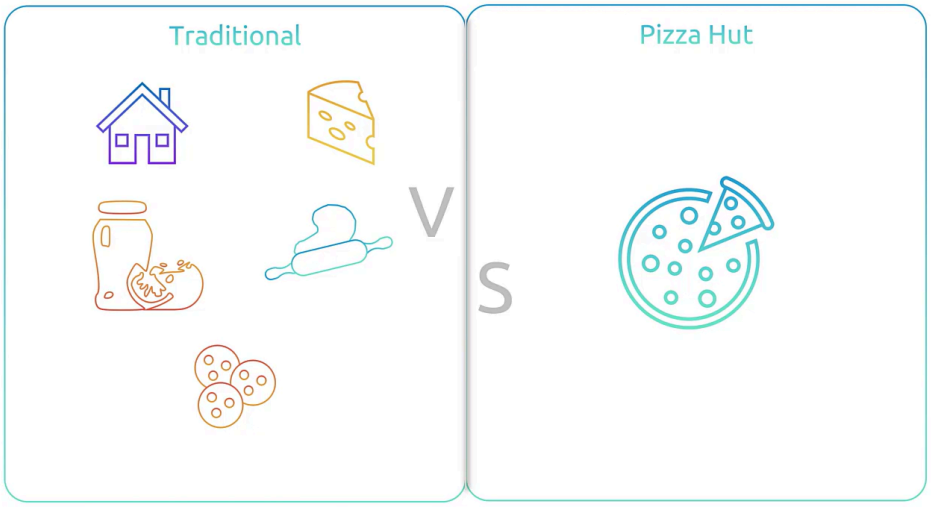


image_5.png

Its more of making Pizza from Scratch vs Ordering one.

In Traditional Services if you have an idea with new technologies and you are using the traditional way, you have to request for:

- Hardware
- Network Cables
- Power Cable
- Server placement with cooling
- Operational and Security Installation
- Access granted to requestor in 3-6 weeks
- Software installations by requestor

> ⚠ But imagine instead of provisioning hardware, you provisioned a hardware service that included networking and power

> ⚠ Instead of managing physical objects, you get access to services that you can configure with ease.

Now imagine all this:

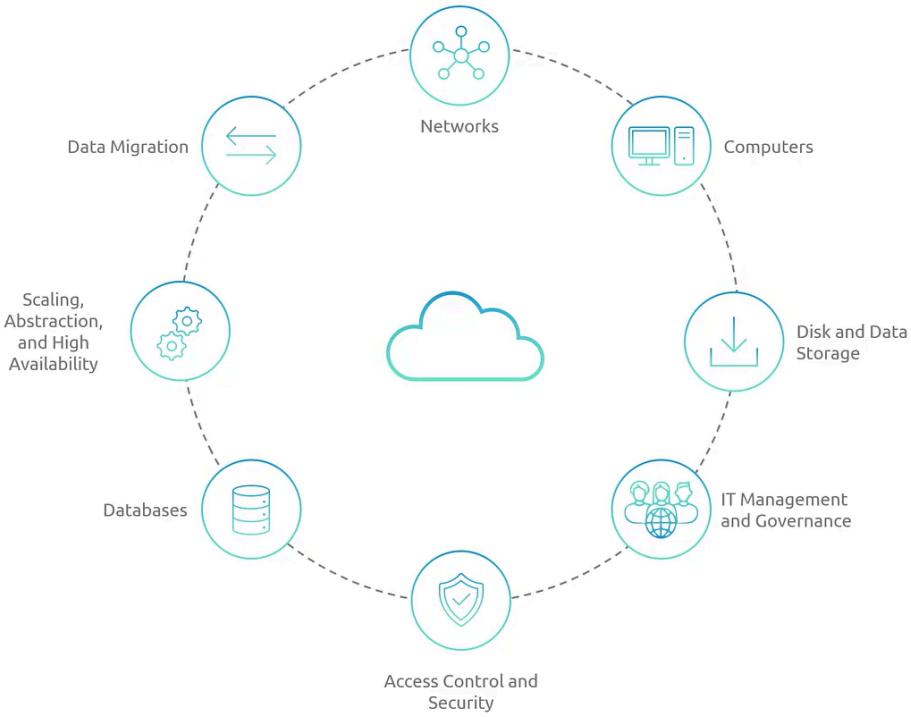


image_6.png

Was Replaced with:

image_7.png

Now the above is Amazon Web Services.

Amazon Web Services

AWS has some Code service categories .

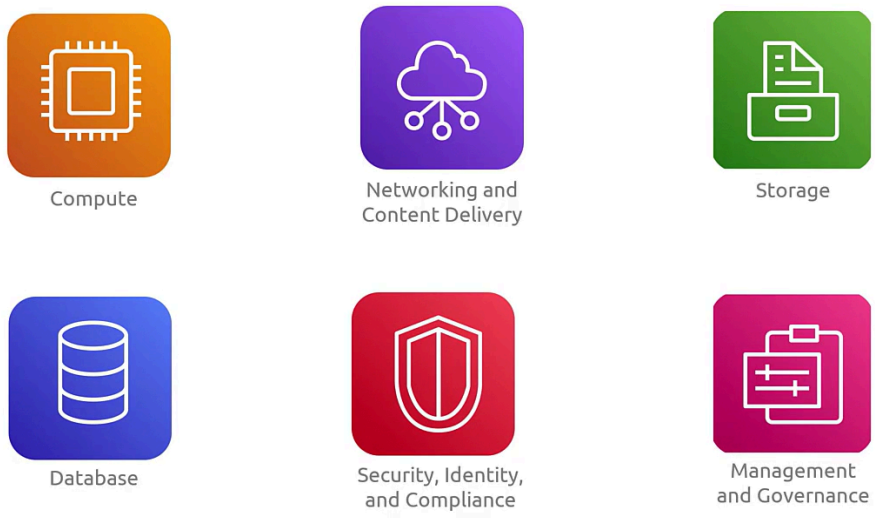


image_8.png

Three Ways to Interact with AWS

Console

It's Great to Learn and confirm.It more of accessing AWS through a website.

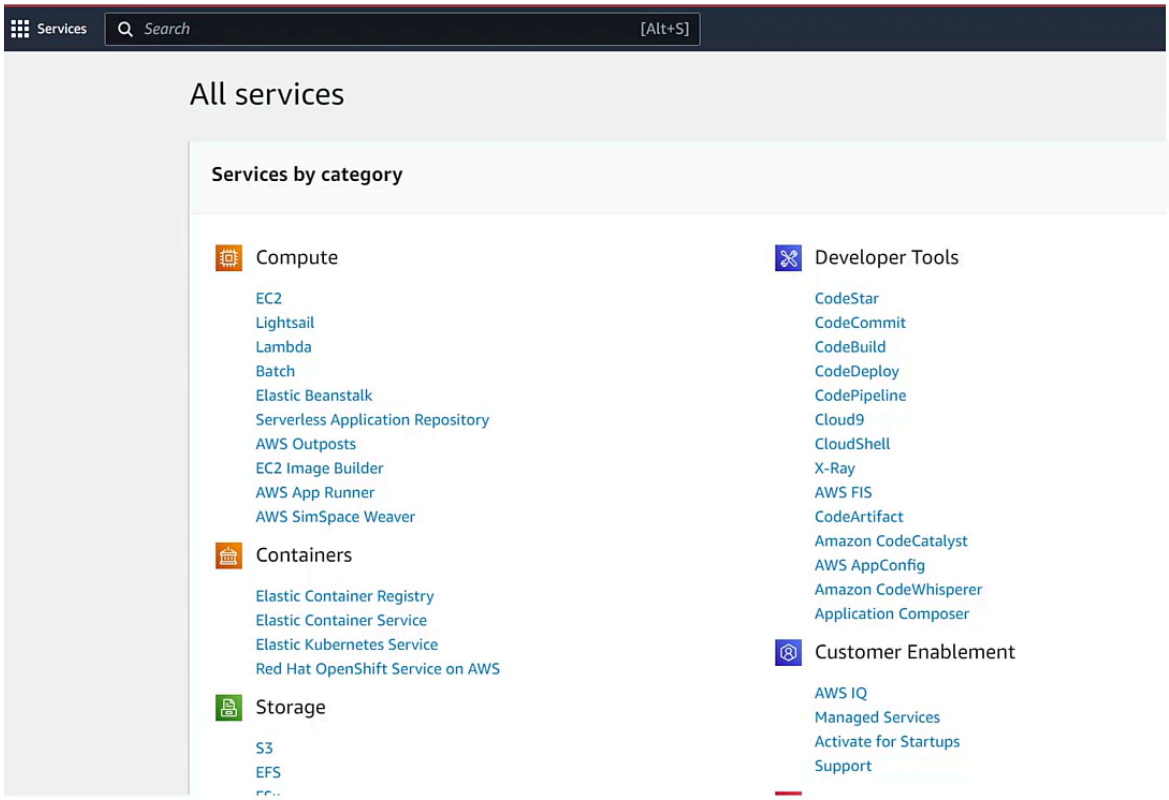


image_9.png

AWS command Line Interface (CLI)

It's Great for engineers . Allows you to type command that get and manipulate information in AWS

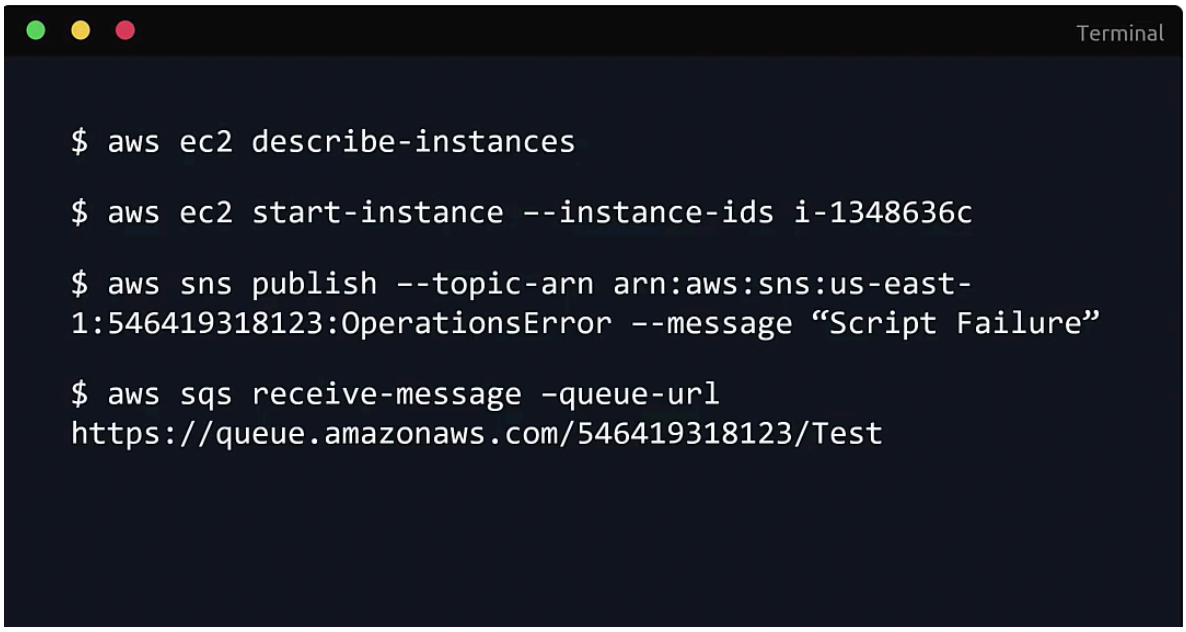


image_10.png

AWS SDK

It's great for developers and different types of coders. SDKs take the complexity out of coding by providing language specific APIs for AWS services. Think of these as Libraries just like any other library e.g Nodemailer in JavaScript

image_11.png

AWS Summary

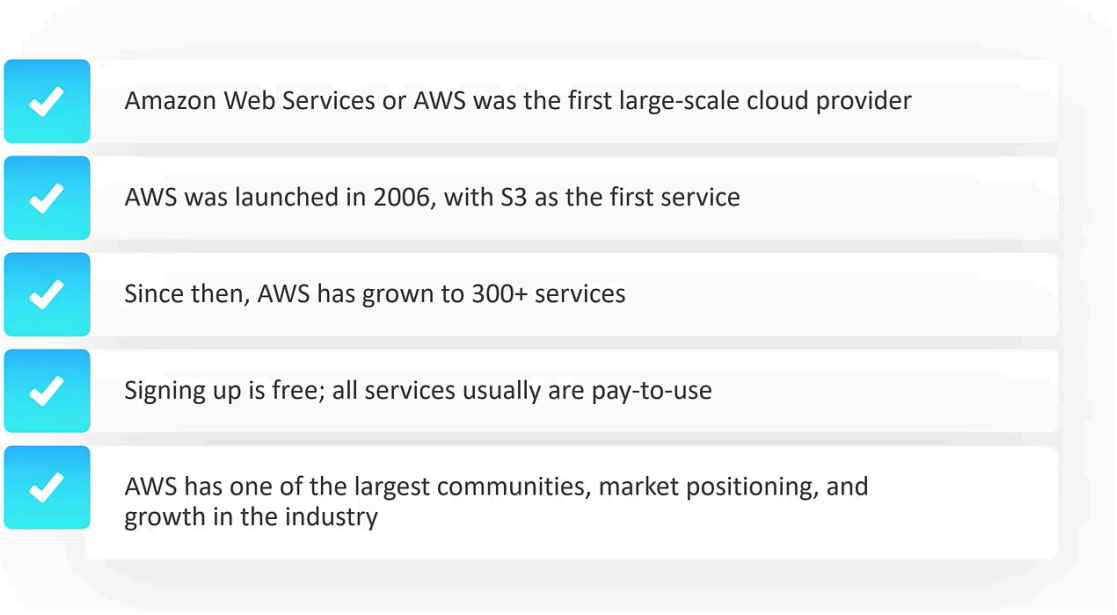


image_12.png

Benefits of Clouds

Trade UpFront Expense for Variable Expense

image_13.png

Variable Expenses (OPEX)

Pay for usage and requests

Give back what you don't need

image_14.png

Stop focusing on Data Centers

Focus on Data Centers

Invest time and money; focus on data centers

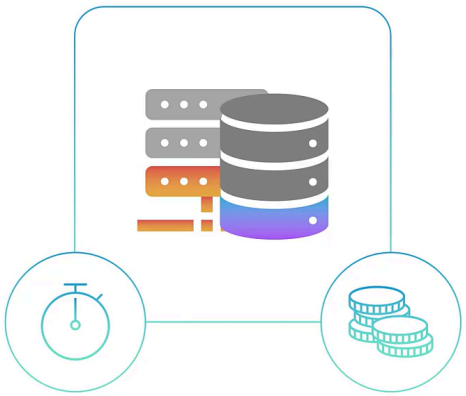


image_15.png

Focus on Customers

Invest time and money; focus on customers and applications

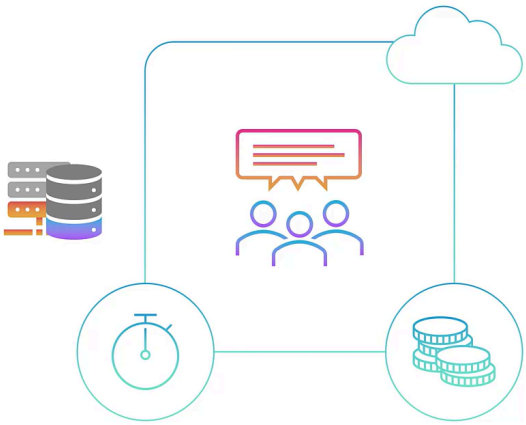


image_16.png

Stop Guessing Capacity (or easily fix bad guesses)

Limited Scalability

Bound by hardware limitations or vendor supply limits

image_17.png

Extreme Scalability

Scales in and out as needed; experiments with load and performance as needed

image_18.png

Benefit from massive Economies of Scale

Smaller Scale

Pay higher prices based on only your own usage, with no price drops per unit

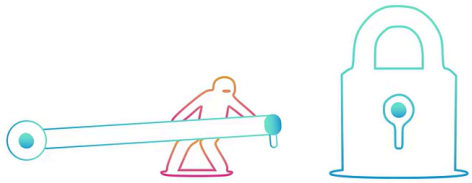


image_19.png

Economies of Scale

Benefit from customers' aggregated usage, with price drops per unit used

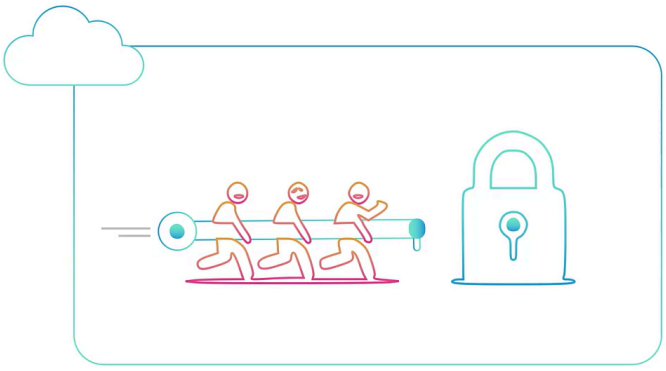


image_20.png

Increase provision speed and Business agility

Data Center Requests

Often provision in days or weeks or months

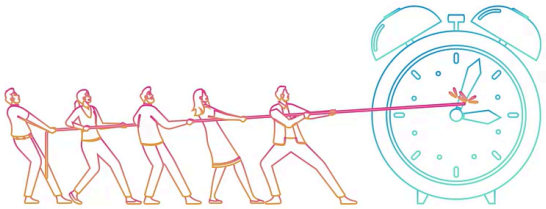


image_21.png

Cloud Computing Requests

Always provision in seconds or minutes

image_22.png

Go Global(with your app) in minutes

Limited Deployment Options

Limited to places where data centers are available

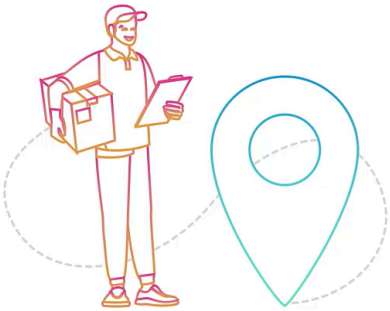


image_23.png

Global Deployment Options

AWS global infrastructure available at 31 locations globally

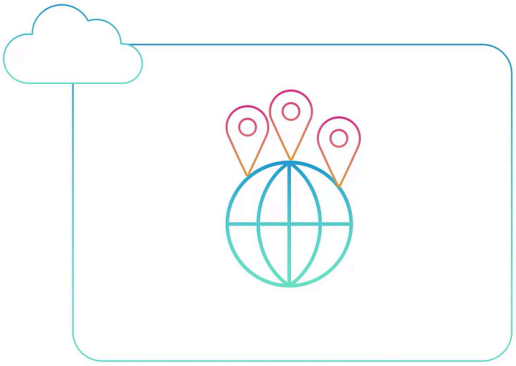


image_24.png

Summary

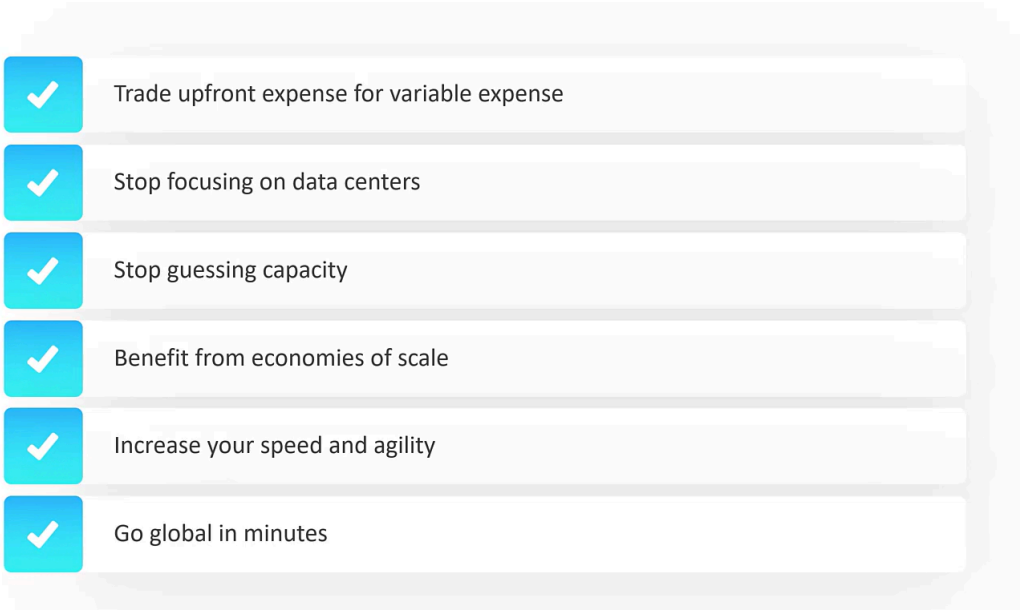


image_25.png

Cloud Computing Economics

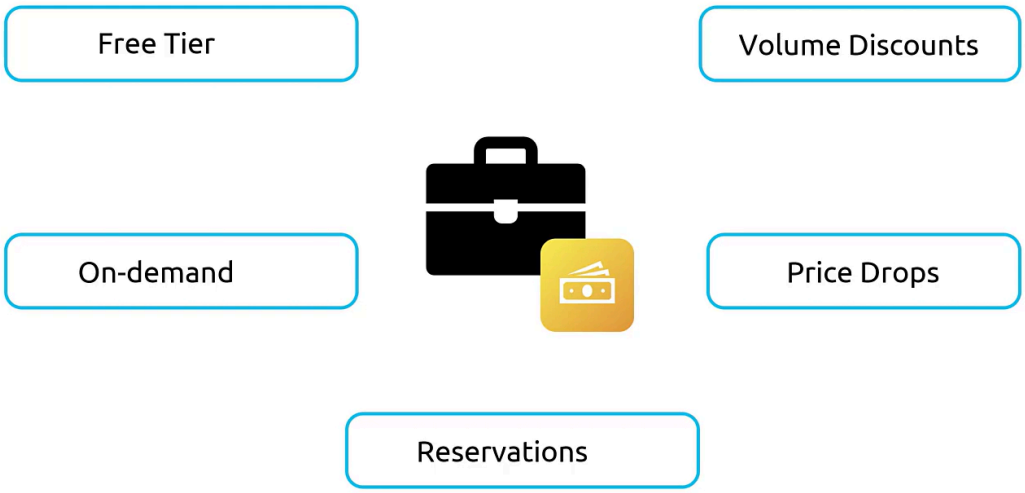


image_26.png

Free Tier

Some services are free forever , some for 12 months after new account creation.

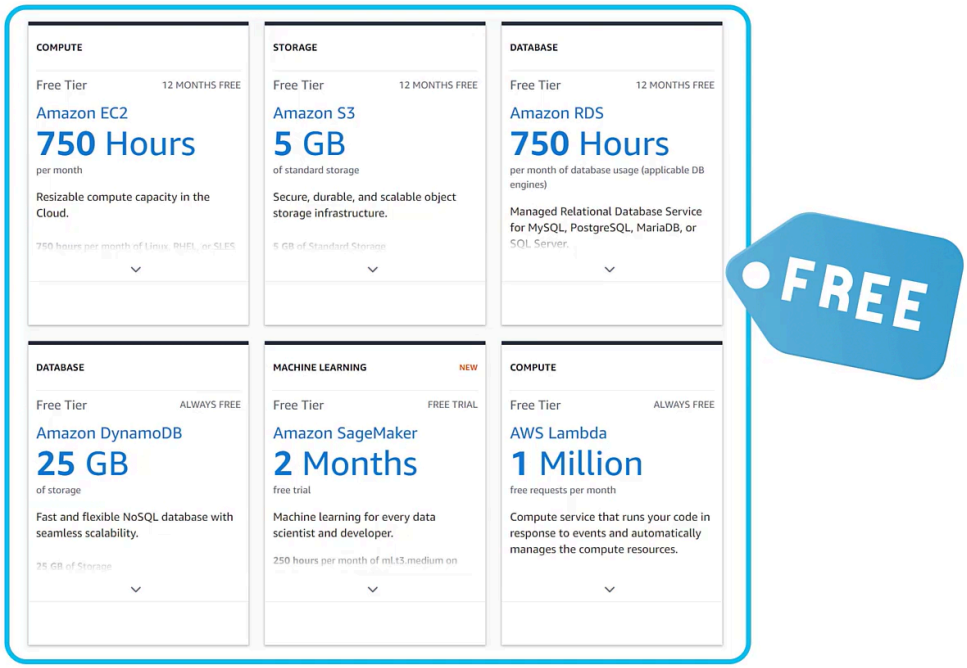


image_27.png

On-Demand

You pay for the Size of the machine you use. For Example here you will pay for using 2 processors , 16Gb of storage and it ran for 10 hours.

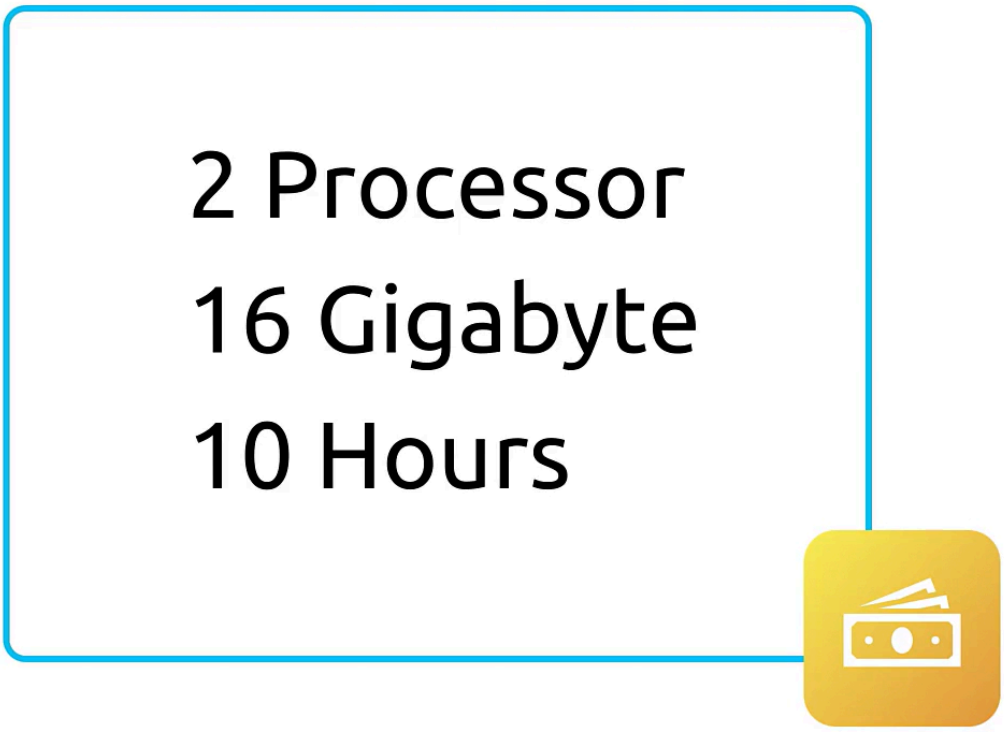


image_28.png

Reservation

You make a one year or 3 year commitment to rent a virtual machine.

Volume Discounts

As your per unit consumption goes up amazon will give you discount for spending more.The more you use the less it const.

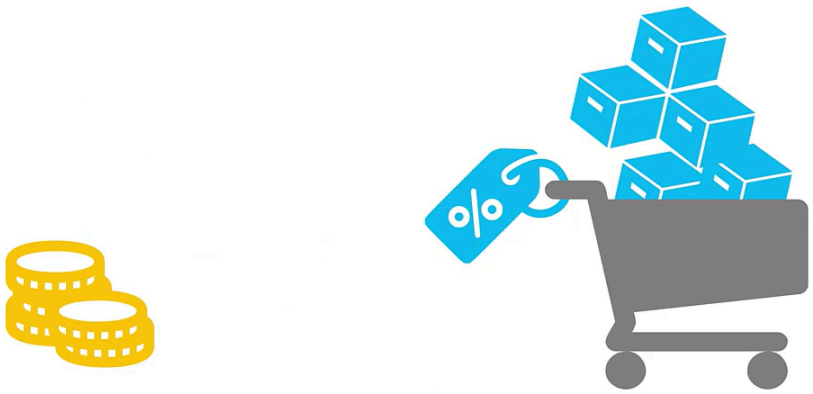


image_29.png

Price Drops

AWS will drop price as they keep upgrading their resources.

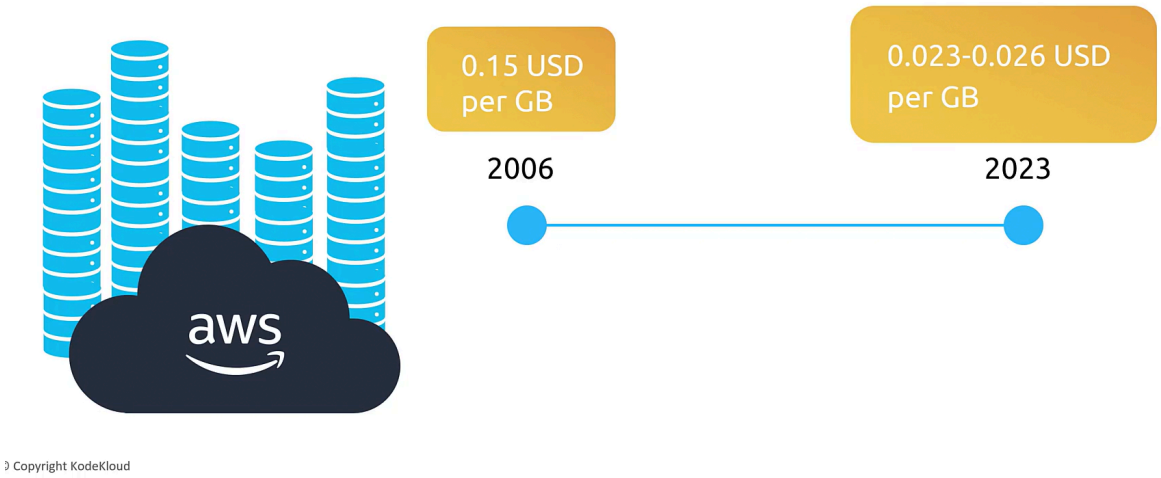


image_30.png

Summary

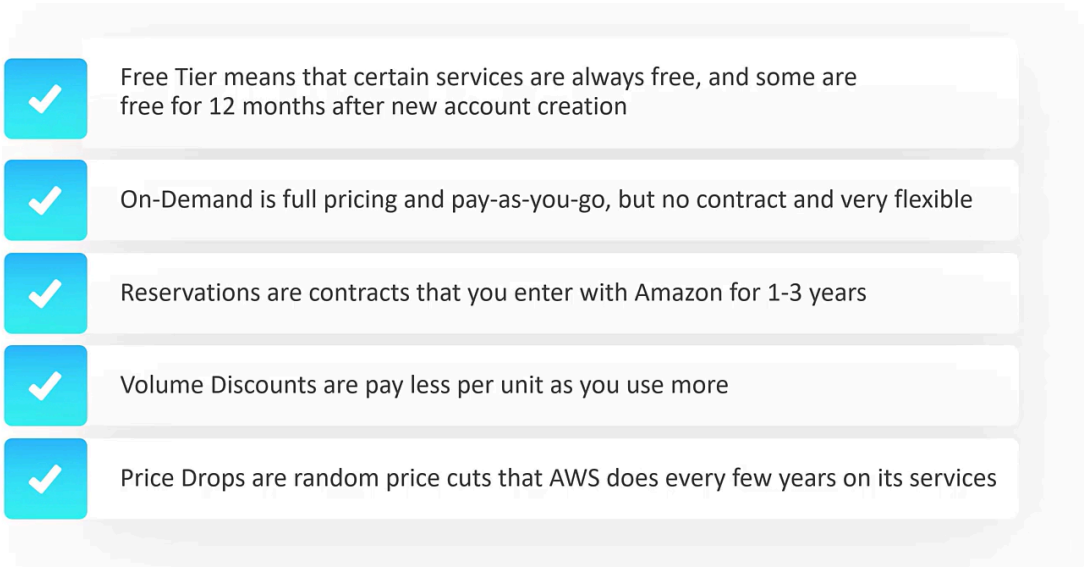

Free Tier means that certain services are always free, and some are free for 12 months after new account creation
On-Demand is full pricing and pay-as-you-go, but no contract and very flexible
Reservations are contracts that you enter with Amazon for 1-3 years
Volume Discounts are pay less per unit as you use more
Price Drops are random price cuts that AWS does every few years on its services

image_31.png

Principles of Cloud Native for AWS

There are four design principles in AWS :

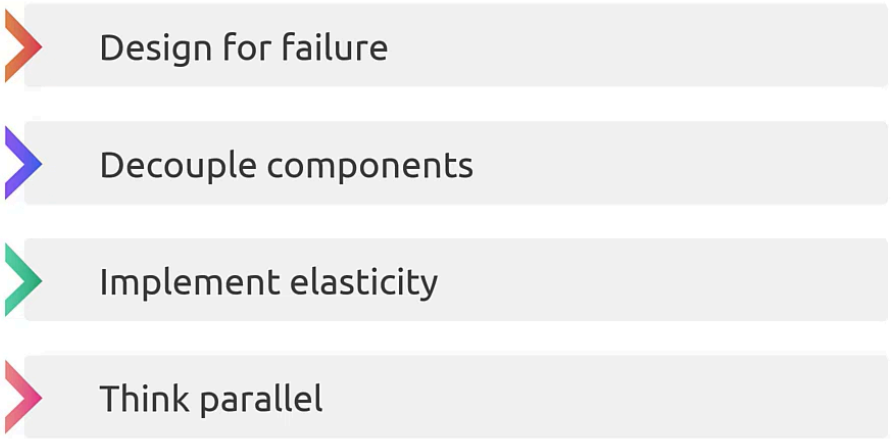


image_32.png

Design for Failure

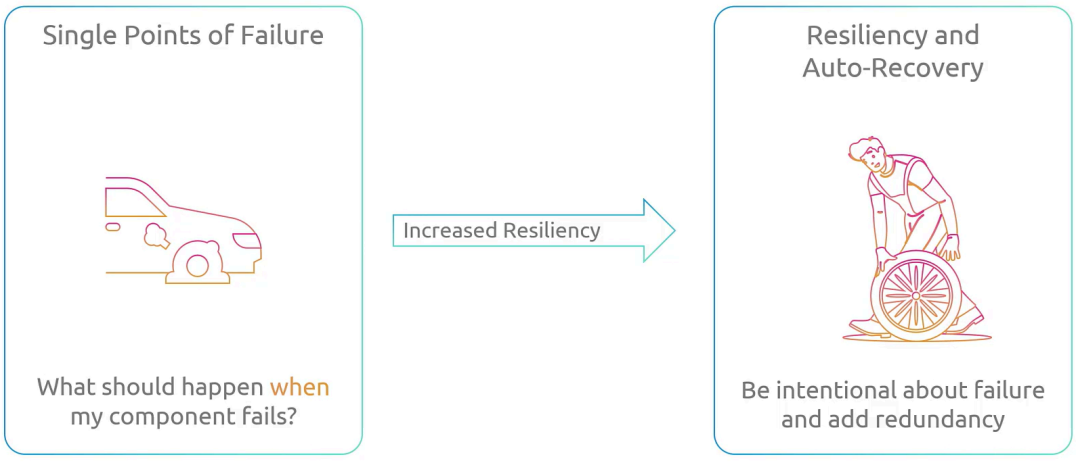


image_33.png

We want to increase resiliency of our applications.

Tight Coupling

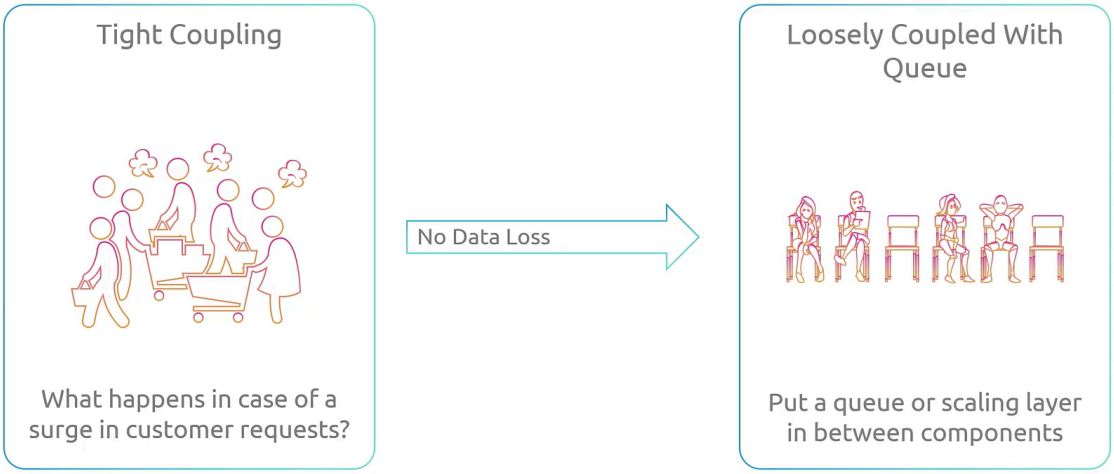


image_34.png

Implement Elasticity

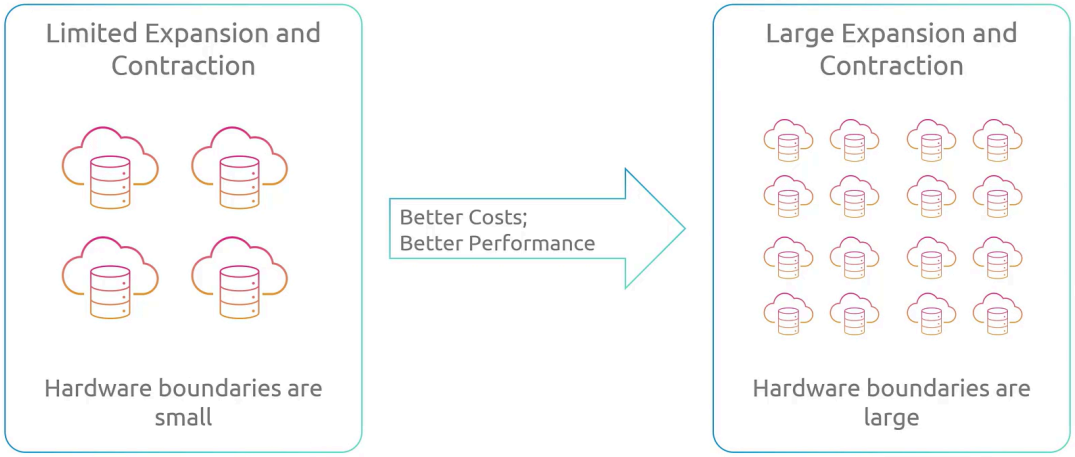


image_35.png

Acquire resource when you need them and release them when you dont.

Think Parallel

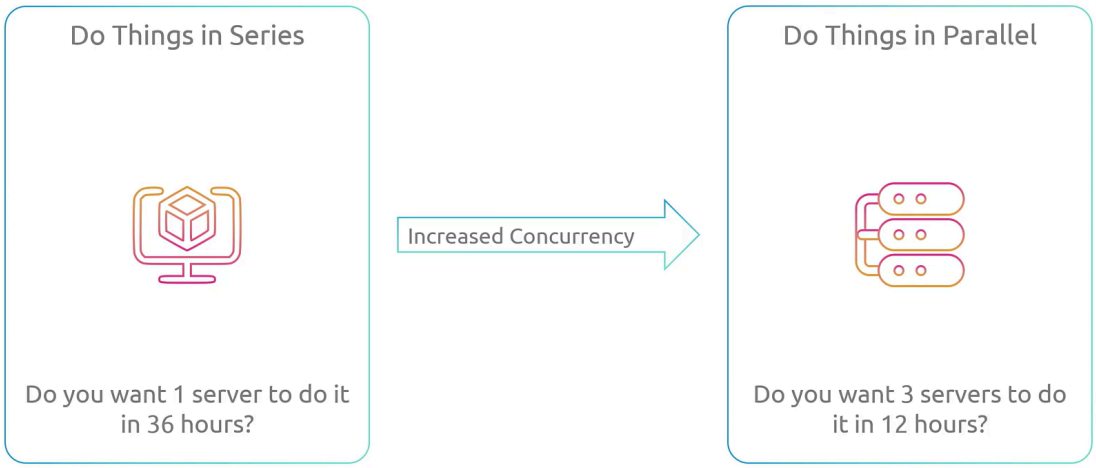

Do Things in Series
Do you want 1 server to do it in 36 hours?
Increased Concurrency
Do Things in Parallel
Do you want 3 servers to do it in 12 hours?

image_36.png

AWS Security

Security The Traditional Way

The traditional security way you are responsible for:

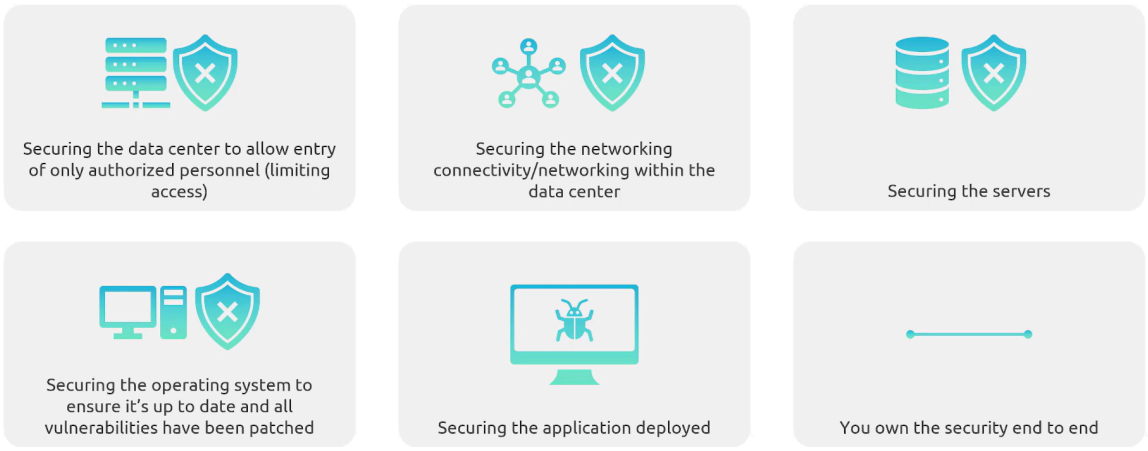


image_37.png

But Security in the Cloud is a shared responsibility model between:

image_38.png

Shared Responsibility Model

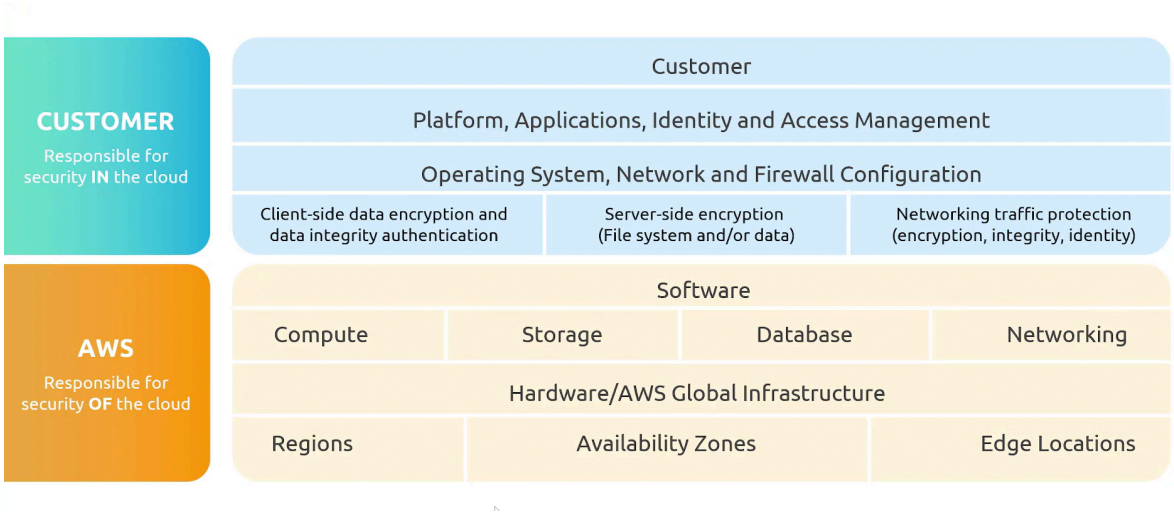


image_39.png

Infrastructure as a Service(IaaS)

AWS will give you a virtual machine that you can use to run application. An example is Amazon Elastic Compute (Amazon EC2)

image_40.png

Platform as a Service (PaaS)

This is where AWS will provide you a platform that you can use to run your code for example. They will set up the operating system and the networking for you and all you need is to just run your code.

image_41.png

Software as a Service (SaaS)

Here AWS provides you a software and they own everything except the application data.

image_42.png

AWS Security Compliance

What is Compliance

Organizations in specific industries must adhere to certain rules and guidelines specific to that industry (Finance, Health, Federal Government).

Compliance and regulatory frameworks are sets of guidelines and best practices. Organizations follow these guidelines to meet regulatory requirements, improve processes, strengthen security, and achieve other business goals

AWS Compliance

AWS undergoes certification reviews and audits to meet regulatory requirements too. Audit reports are available on **AWS Artifacts**. Customers and review and accept agreements in the Artifact.

Customer Compliance Center

image_43.png

AWS AUdit Manager

AWS Audit Manager continuously collects data to prepare for audits and ensures that you are achieving compliance with regulatory standards. It helps build audit-ready reports.

AWS Config

It tracks how the resource is configured and records the previous configuration states, so you can see how the configs for it have changed over time

With AWS config it keeps track of all the changes made to a resource . Example if I launch an EC2 instance , then add a security group, then add an EBS volume . AWS config will keep track of all these configurations that im making to the ECS. This will be a great tool for auditing and compliance.

Short Summary

image_44.png

Identity and Access Management (IAM)

To create an AWS account you must provide a unique email, account name and a credit card.You can use same credit card for different accounts.

When you create an account you get a Root user, The Root user has access to everything in the account and can perform any action on any resource within the account.

> ⚠ But assuming You have a new employee in your organization and you want to provide access and also maybe restrict access to ceartain resources.

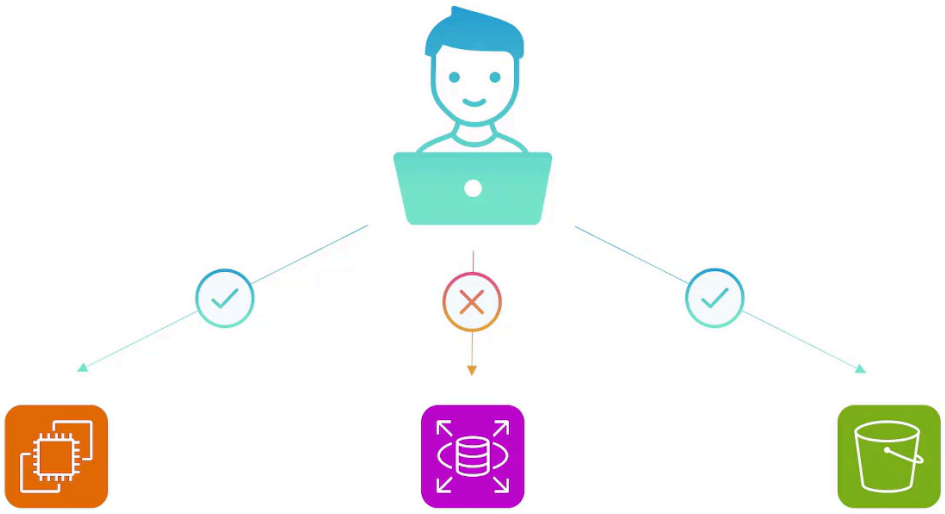


image_46.png

This is where IAM comes in.

Identity and Access Management (IAM)

IAM manages access to AWS resources. IAM handles who is authenticated and what are they authorized to do.

There are three types of Identities:

- Users
- Groups
- Roles

IAM **policies** define what resources a user/group/role has access to and what actions they can perform on them.

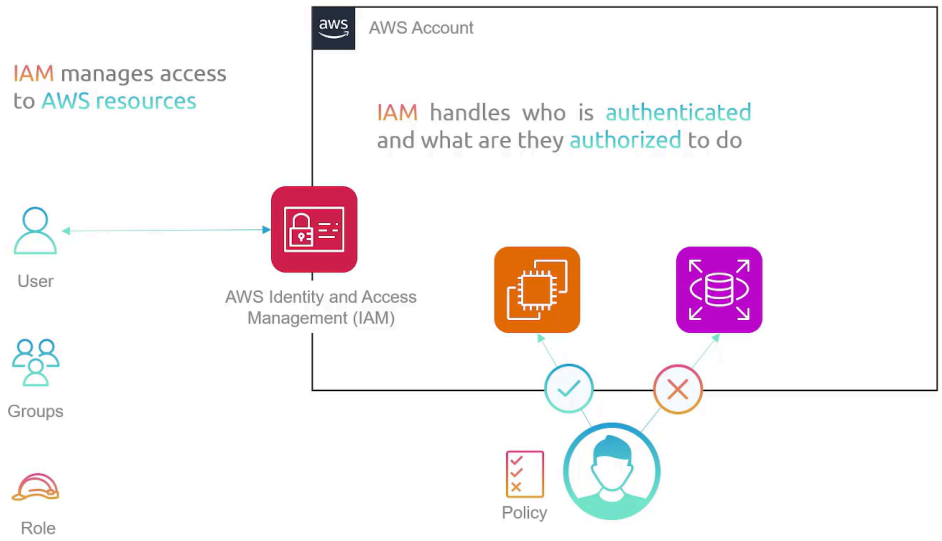


image_47.png

User

IAM user is just an organization or user who wants to access a set of services within AWS .

But what do users have access to?

> ⚠ Users have to be granted explicit permissions to access specific services/resources.
>
> Users are implicitly denied all permissions by default

Policies

To grant users access to resources, IAM policies need to be applied to the user, giving them permissions.

Policies are documents that either grant or deny access to specific AWS services/resources.

image_48.png

Policy Example

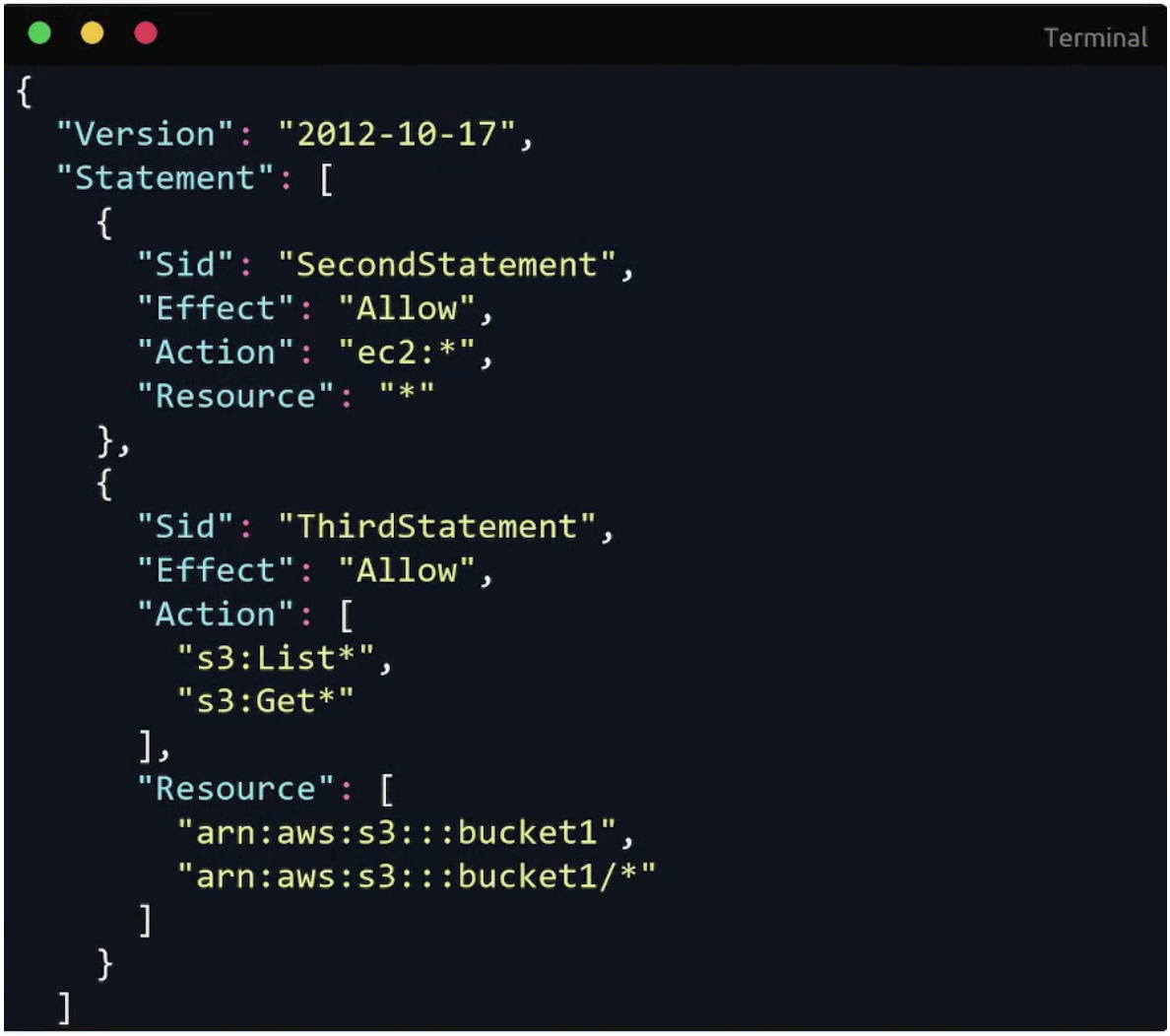


image_49.png

This policy has a version which can change based on the recent AWS policy version. Statement is what permissions you want to grant access to.

image_50.png

The above policy has a unique id and allows a user to access to all EC2 resources.

```
{
  "Sid": "ThirdStatement",
  "Effect": "Allow",
  "Action": [
    "s3:List*",
    "s3:Get*"
  ],
  "Resource": [
    "arn:aws:s3:::bucket1",
    "arn:aws:s3:::bucket1/*"
  ]
```

image_51.png

The second permission allows a user to list and get all objects in a S3 bucket

> ⚠ Now we have a user and policies

Users and Policies

We now attach a policy to a user since by default they don't have access to anything

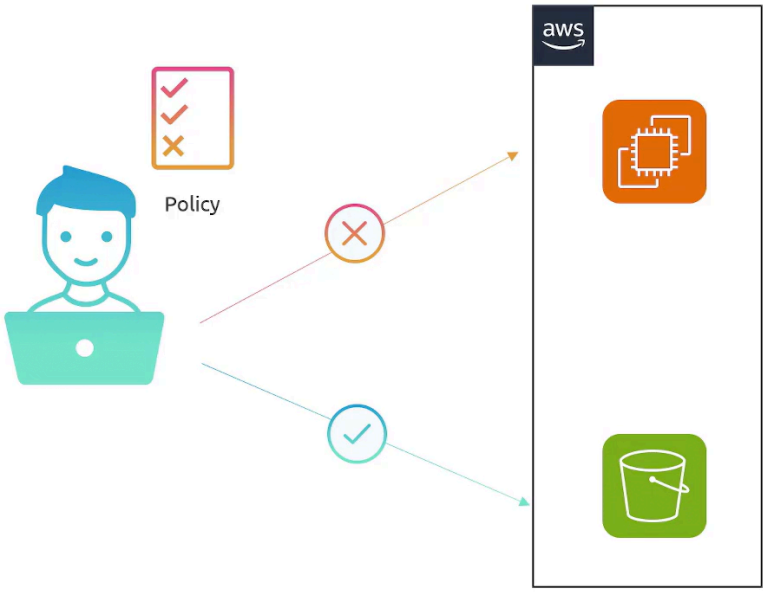


image_52.png

Least Privilege Permission

Identities/ users should only be granted the minimum permissions necessary for the to perform their job.

> ⚠ For every new user created you have to give them certain permissions and you have to remember to always give them permission.
>
> If I have a bunch of users that require same permission we can create groups.

Groups

Groups are a collection of IAM users.

- Policies can be added to groups
- IAM users within a group automatically inherit all the policies from the group
- Example – having a separate group for each department

Roles

Roles allow users, applications, or services to assume temporary permissions.

- Roles are assigned permissions/policies just like users and groups
- When someone assumes an IAM role, they inherit the permission of the role temporarily and return to their original permission when done

Multi-Factor Authentication (MFA)

To login to an AWS root account you must provide an Email and a Password. If they leak or hacked then other people will have access to your account and might do malicious acts.

To prevent this we can create an Extra security , a code that will be generated on an MFA device/app which changes periodically. This code will then be sent to AWS to verify the user. So they can have the email and password but if they dont have a code then they cant access your account.

Organizations

Organizations help manage multiple AWS accounts Organizational units (OUs) allow you to group accounts with similar business or security requirements

Service Control Policies (SCPs) restrict what an account can do

- SCPs can be applied to individual accounts or OUs
- When applied to OUs, all AWS accounts within the OU inherit the policies

AWS Security Resources

Prevention Security Resources

Web Application Firewall (WAF)

Monitors HTTP requests and prevents a variety of attacks like SQL injection and XSS attacks.

AWS Shield

AWS Shield detects and mitigates sophisticated distributed denial of service (DDoS) attacks. A DDoS attack is when a perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting the services of a host connected to a network.

- Denial of service is typically accomplished by flooding the targeted machine or resource with fake requests to overload systems and prevent some or all legitimate requests from being fulfilled.

AWS Network Firewall

It is a stateful managed firewall and intrusion detection and prevention service for VPCs. It monitors traffic going into and out of a VPC.

Detection Security Resources

AWS Inspector AWS inspector scans workloads running on AWS for software vulnerabilities and undesired network exposure.

- It automatically discovers and scans EC2 instances, container images in ECR, and Lambda function.
- It continues to assess your environment throughout the lifecycle of your resources by automatically rescanning resources in response to changes that could introduce new vulnerabilities.

1. Installing new package

2. Installing a patch

3. New common vulnerabilities and Exposures (CVE) are released ● If vulnerabilities are identified, AWS Inspector produces a finding that you can investigate.

Amazon GuardDuty

image_53.png

Amazon Detective

Amazon Detective helps you quickly analyze and investigate security events across one or more AWS accounts.

1. It ingests data from VPC flow logs, CloudTrail logs, and GuardDuty findings.

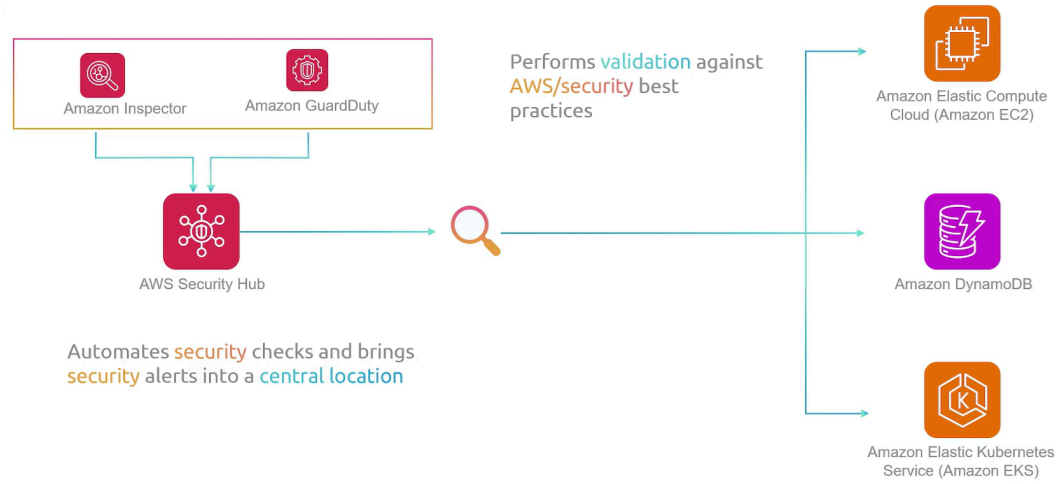
2. It uses machine learning and statistical analysis to create advanced visualizations that show resource behaviors and interactions over time. Amazon Detective simplifies the process of investigating security incidents by providing interactive dashboards and visualizations.

AWS Config

It tracks how the resource is configured and records the previous configuration states, so you can see how the configs for it have changed over time

AWS Security Hub It automates security checks and brings security alerts into a central location:

- Ingests data from other services like AWS Config, GuardDuty, Firewall Manager, Inspector, etc.
- Performs validation against AWS/Security best practices



- image_54.png

CloudTrail

CloudTrail tracks user activity with an AWS account.

- It tells us who is doing what.
- Anytime a user performs an action within AWS, an event is logged in CloudTrail, such as:

1. Logging in
2. Modifying security policies
3. Accessing/creating/deleting a service • It logs actions taken in the console, command line, and AWS SDK/APIs.

Security Lake

It collects security logs and events from multiple sources including on-premises, AWS services, and third-party services. It transforms the data into storage and query-efficient Parquet format

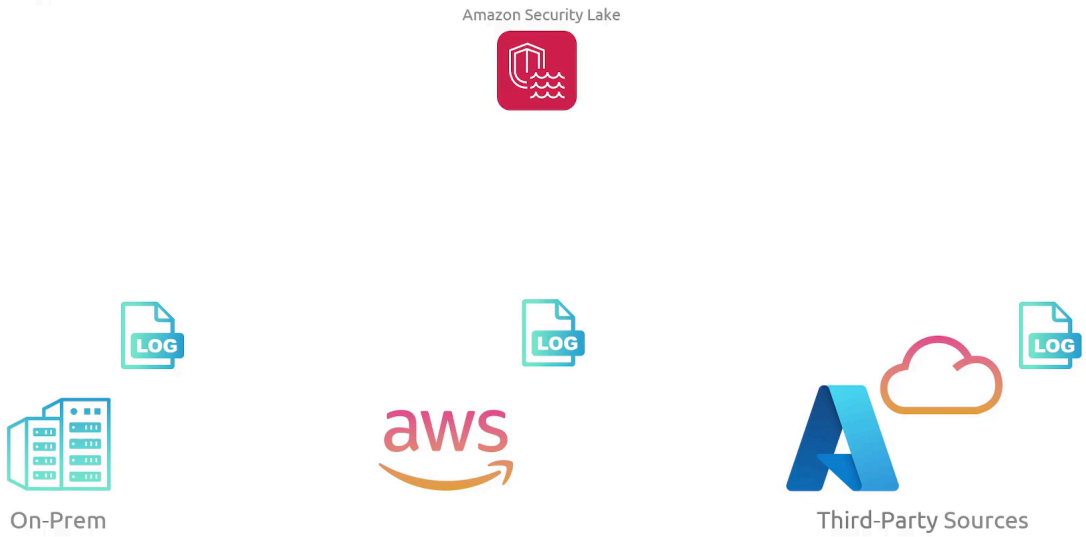


image_55.png

AWS Macie

AWS Macie uses pattern matching and machine learning to automatically discover sensitive data. Macie will generate an inventory report of S3 buckets and scan objects for sensitive data and notify you.

Detection Summary

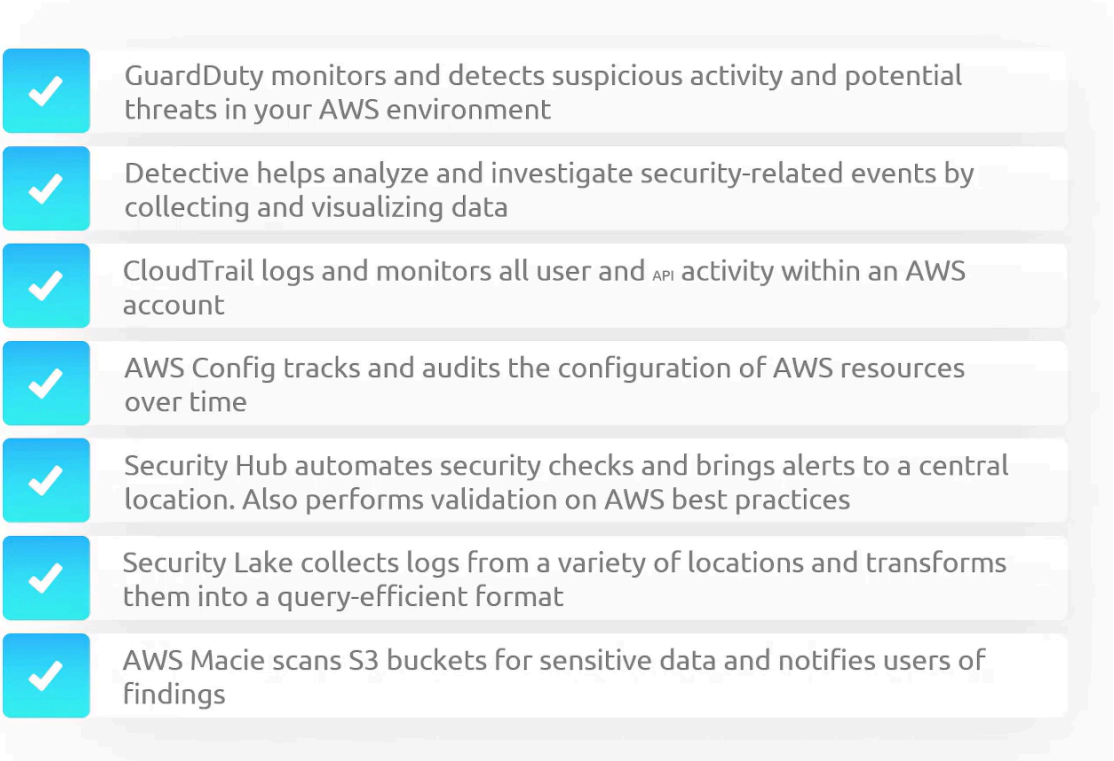


image_56.png

Management Security Resources

Firewall Manager

It helps manage security services (WAF, security groups, network firewall) across multiple AWS accounts. It configures rules just once and has them available across all accounts.

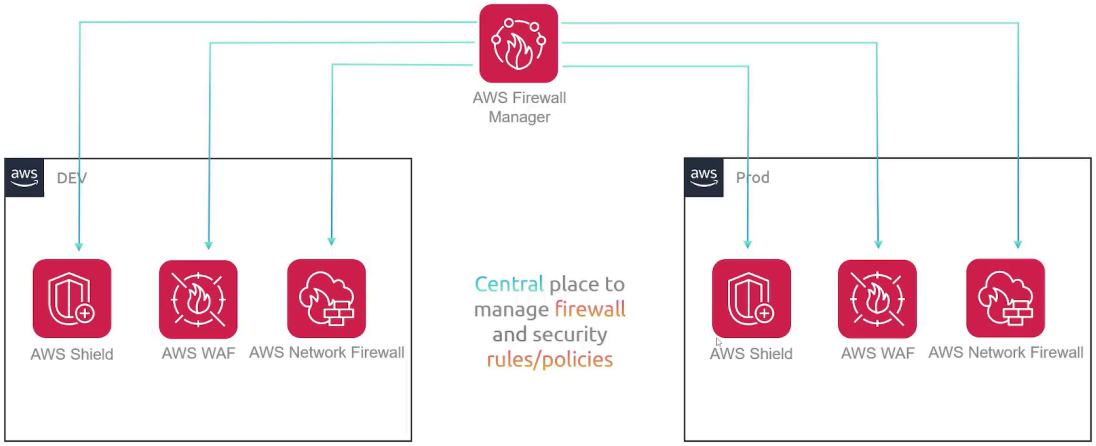


image_57.png

Resource Access Manager

It helps you securely share resources across accounts, organizations, and OUs. So, you can create a resource once and have the Resource Access Manager make that resource usable by other accounts.

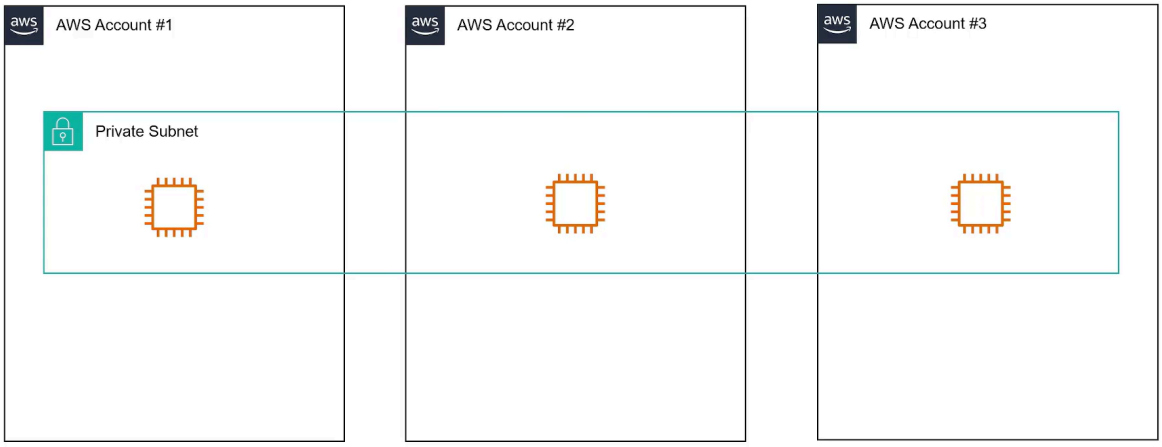


image_58.png

AWS Cognito

AWS Cognito helps implement customer identity and access management for mobile and web applications. ● It manages all user credentials. ● It makes adding signup/login functionality easy without social authentication on any app.

IAM Identity Center

It simplifies managing users across multiple AWS accounts. It also allows you to manage sign-in security for your users centrally and grant them access across all AWS accounts and resources.

Secrets Manager

The Secrets Manager helps retrieve and rotate secrets and other sensitive data like credentials and Auth tokens.

- No need to hard code credentials in application source code
- Makes it more difficult to leak or compromise sensitive credentials
- Can configure automatic rotation schedule for your secrets

- Having short-term secrets decreases the risk of compromise

AWS Certificate Manager (ACM)

It handles the complexity of creating, storing, and renewing public and private SSL/TLS certificates and keys that protect AWS websites and applications.

- If users need to communicate with an ELB or API gateway, you now no longer need to purchase a paid certificate from a third party. -ACM can generate them for you, and end-user traffic will be encrypted.
- You can issue certificates directly from ACM or import third-party certificates.

AWS Private Certificate Authority

It provides users with a private CA that is managed by AWS.

- Issues certificates for authenticating internal users, computers, and applications
- Certificates issued by a private CA are trusted only within your organization and not on the internet
- Saves the hassle of having to set up, configure, and maintain your own internal certificate authority

Key Management Service (KMS)

KMS helps create and manage cryptographic keys used for encrypting/decrypting data.

- It allows you to provide granular control over who has access to the keys
- Key rotation can be configured

CloudHSM

CloudHSM provides customers with a hardware security module in the cloud.

- All cryptographic keys are securely stored on the HSM, and they never exit the device
- Data is sent from the clients to the HSM for encryption/decryption

- Because all keys are securely stored on the HSM in a central location, it helps minimize the risk of keys getting leaked or stolen

Global Infrastructure

region

An AWS region is a geographic location where resources can be deployed.Each region id designed to be isolated from each other for:

- Fault tolerance
- Stability

> ⚠ But all the regions are not the same

Selecting a Region

- Not all services are available in all regions.
- GovCloud regions designed for U.S. Government and customers who need to meet regulatory and compliance requirements.
- You should select a region as close to your customers as possible to minimise latency.
- Cost od services varies from region to region
- Due to compliance and legal requirement , you may need to run your infrastructure in certain regions.

Availability Zones (AZ)

Within a region there are multiple Availability zones. An Availability zone is one or more discrete data centers with redundant power, networking, and connectivity in an AWS region. Availability zones provide redundancy within a region.

If you have an application deployed in only one AZ, then the entire region is affected in case of a failure in that AZ

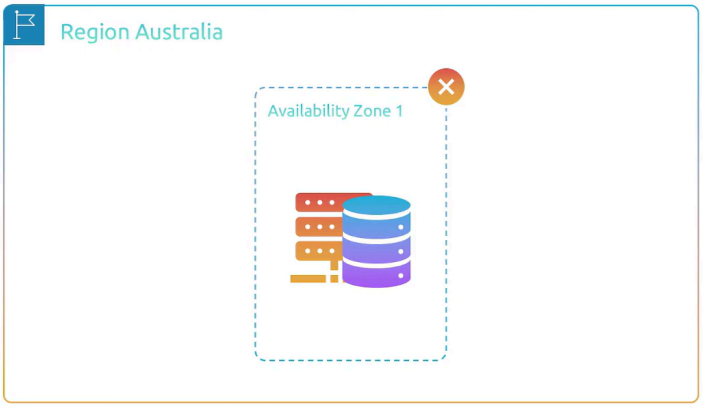


image_59.png

But instead we can have the application in multiple availability zones so that if one is down , another can cover it up.

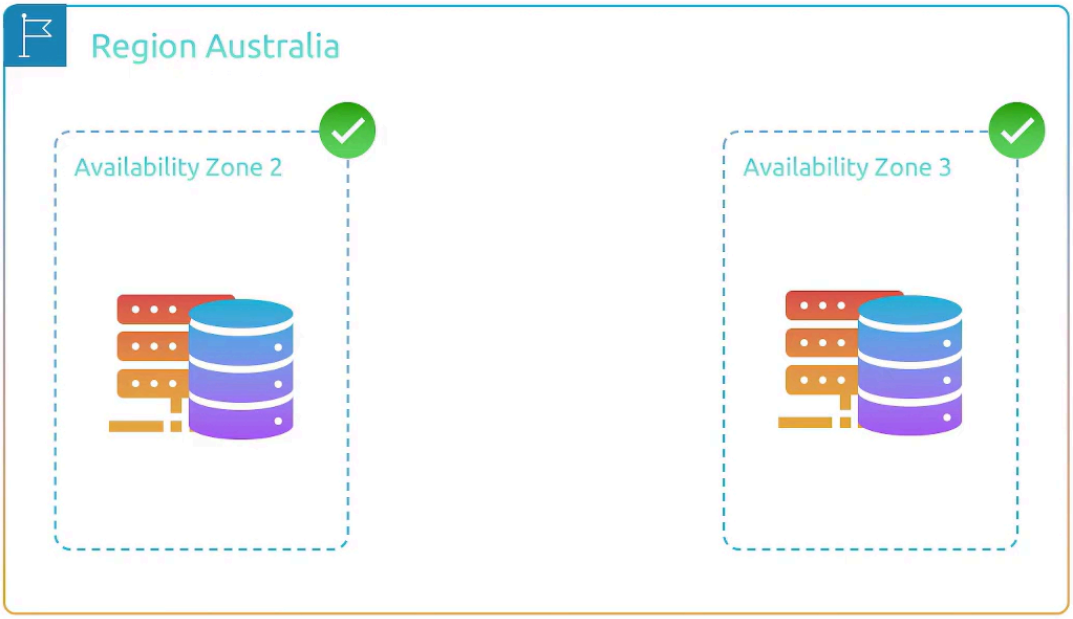


image_60.png

Global Content Delivery and Edge Locations

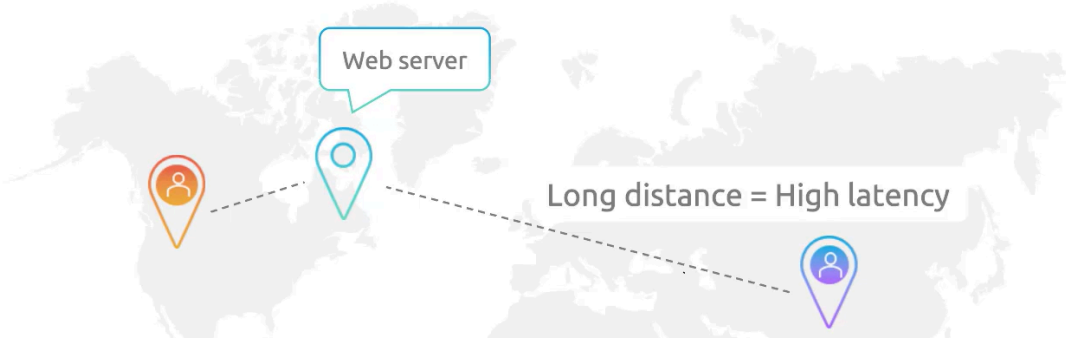


image_61.png

If we have a web server in N.America and a customer from the same country is accessing a resource then this will be faster but if the customer is accessing from India then that's a longer distance and we might experience latency.

AWS noticed this and Solved it using Edge Locations:

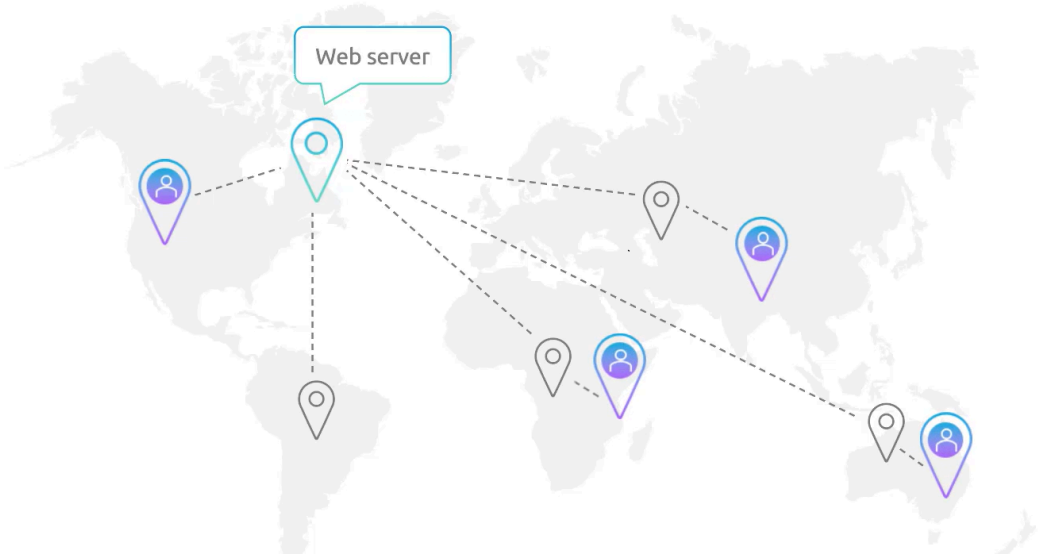


image_62.png

Edge Locations

AWS Edge Locations are strategically positioned points in the AWS network optimized for low-latency content delivery, ensuring that data reaches users swiftly. In contrast, AWS regions are larger geographic areas with multiple Availability Zones.

Now instead of the request being sent to N.America, Its going to be sent to an edge location near India hence low-latency.

Local Zones

Local zones are extension of an AWS Region. local zones allow you to use select AWS services like compute, and storage closer to end-user. Local Zones provide high-bandwidth , secure connection to parent AWS region. Great for low latency applications like real-time gaming, live-streaming and virtual workstations for the local city.

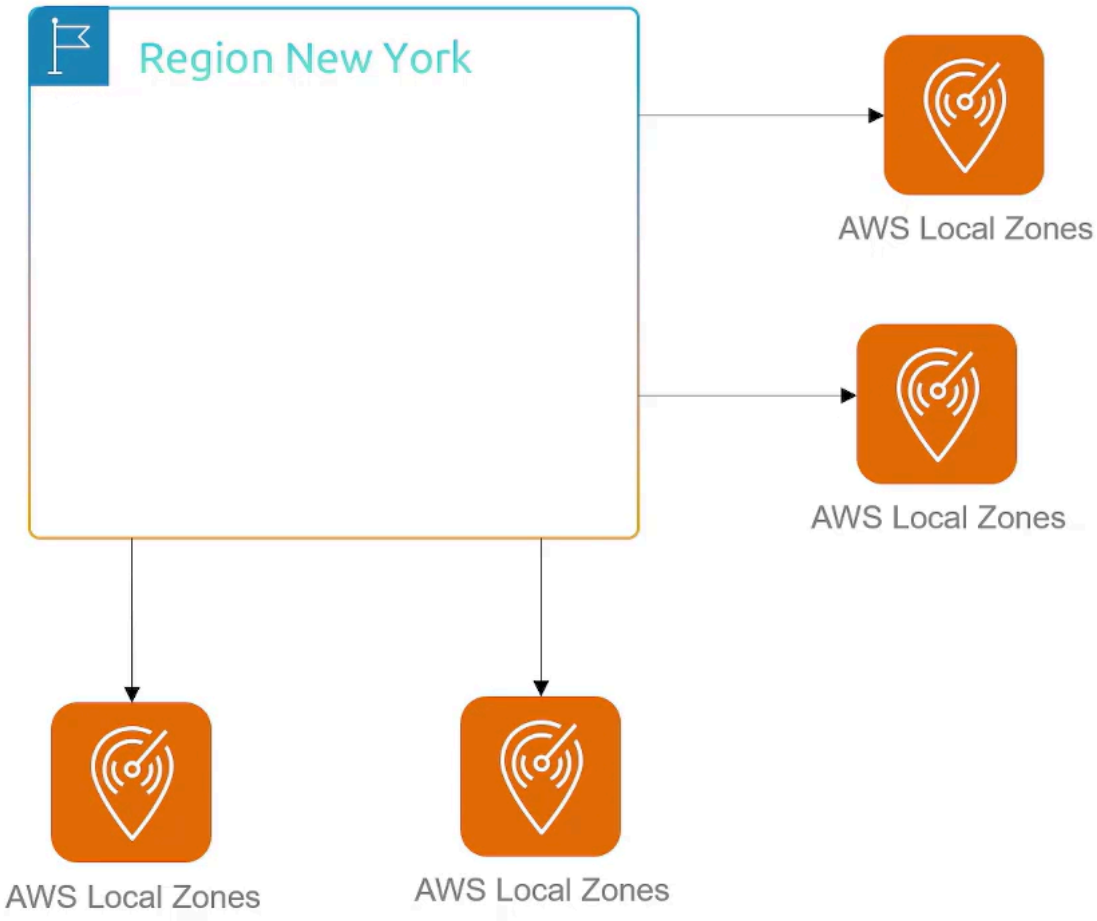


image_63.png

Edge Locations vs Local Zones

- Both help get services closer to end-users to improve user experience and minimize latency
- Local Zones are extensions of a regions physically located in major cities across the world
- Local Zones have their own isolated infrastructure but are connected to the parent AWS region through a high bandwidth network link
- Local Zones Provides access to subset of services like EC2 & EBS
- Edge Locations are small geographically dispersed compute sites that primarily support services like CloudFront(content delivery network), Route 53, AWS WAF
- Edge Locations are limited to mostly CDN services and do not provide full compute services
- There are hundreds of Edge Locations as they are not full sized datacenters; there are over 20+ local zones

image_64.png

Short Summary

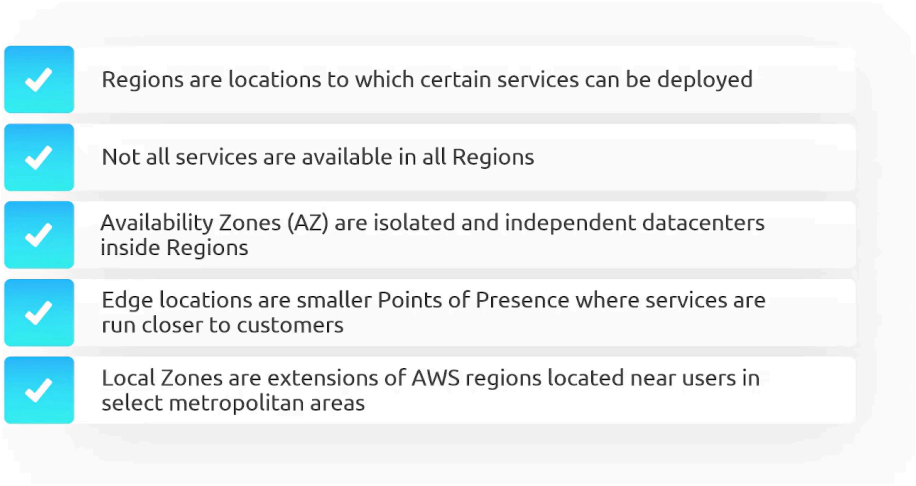


image_65.png

AWS Networking

In AWS you rent a slice of a machine because renting the whole machine, That means that one machine can have several customer applications. But does that mean that they will be able to communicate with one another?

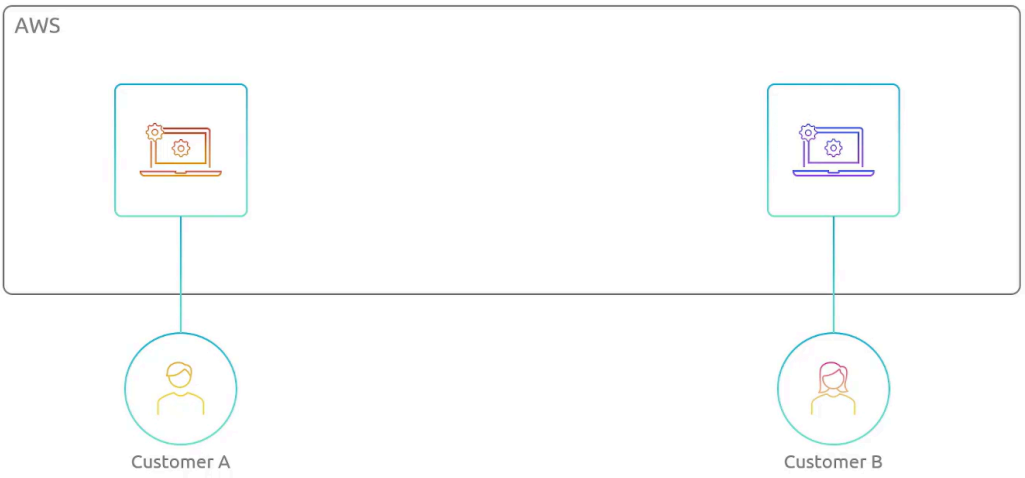


image_66.png

Amazon thought about this and came up with VPC

VPC

Every customer will have their application running on a private cloud which can't communicate with another private cloud.

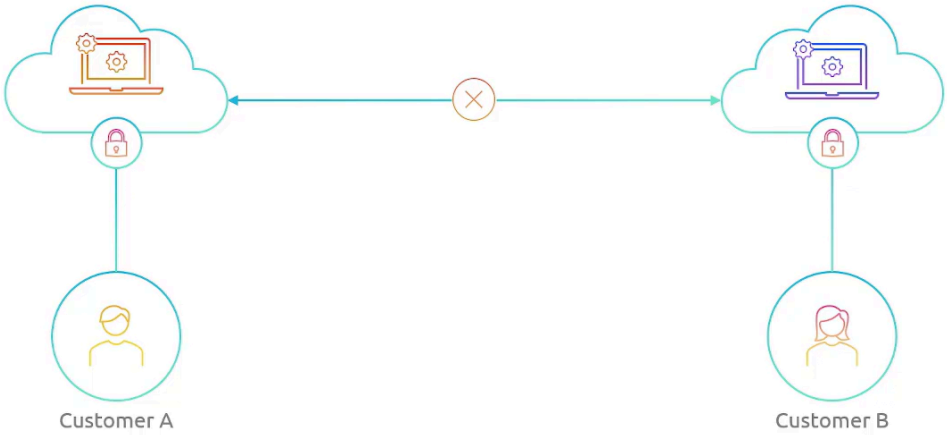


image_67.png

Virtual private cloud(VPC) is a secure, isolated network segment hosted within AWS. VPC isolates computing resources from other computing resources It gives the customer full control of the networking in the cloud and provides:

- Sub-netting
- Routing(Routes Tables)
- Firewalls (NACLs and Security Groups)
- Gateways

But does it mean every customer gets one VPC?

No you can create multiple VPC and you can create a VPC for different environments or applications.

VPC and Regions

A VPC is specific to a single region

Subnets

Subnets are a group of IP addresses in your VPC. For example if you deploy a website on an EC2 it means that the application will be running on one of the IP address of a subnet.

A subnet also resides within a single availability zone

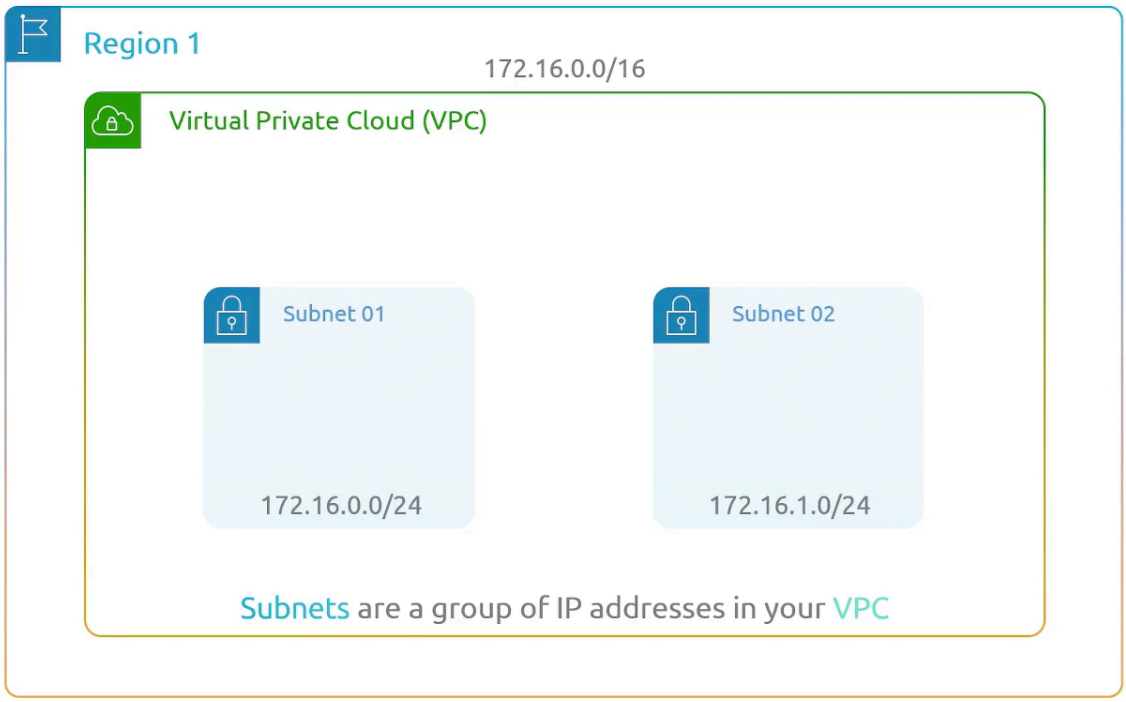


image_68.png

A Subnet can either be public or private to allow external access to resources within them.

A private subnet means that the subnet can't talk to the internet and also the internet cant talk to the subnet. It's good for something like a database

A Public subnet on the other side means that the subnet can talk to the internet and you can also access it from the internet.

Every VPC has a range of IP addresses assigned to it called the CIDR block.

A CIDR block defines the IP addresses that resources in the VPC can use

A CIDR block size can be anywhere from /16 to a /28

e.g. !92.168.0.0/16 : Will give you a range from 192.168.0.0 - 192.168.255.255 Subnet within a VPC must be within the CIDR range:

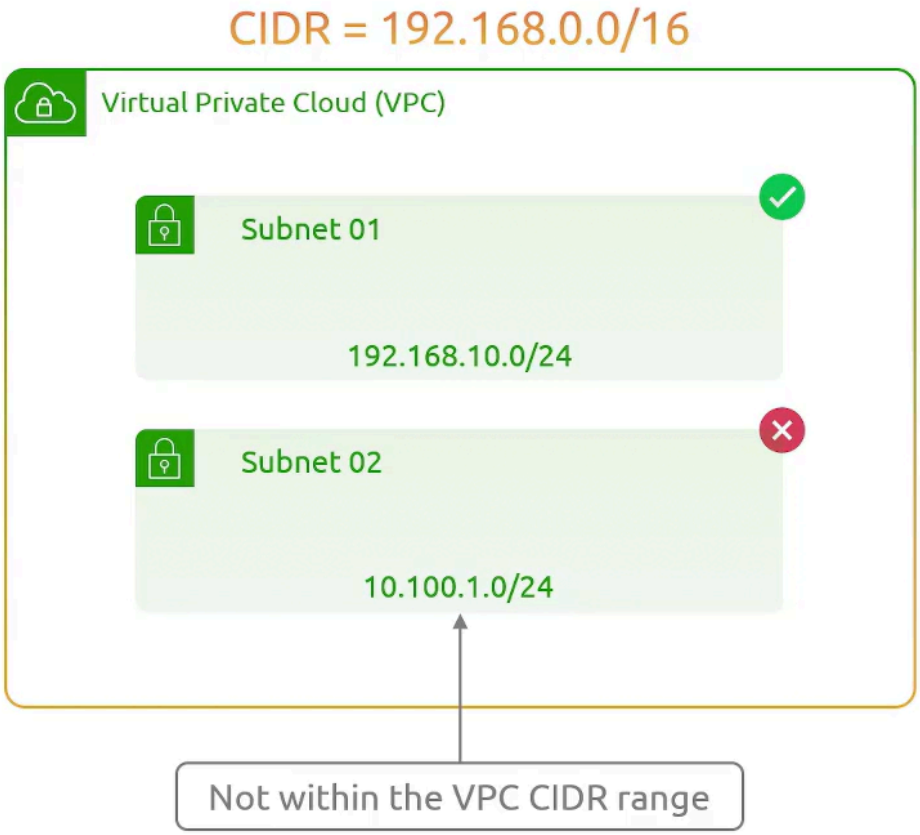


image_70.png

The first 4 IP addresses of a subnet are reserved and cannot be used.

- 192.168.10.0 : Network address
- 192.168.10.1 - 192.168.10.3 : Reserved for AWS
- The Last IP address of a subnet is reserved as the broadcast address : 192.168.10.255
- The first address ready for use should be 192.168.10.4

VPC External Connectivity

Internet Gateway

Subnet when first created are private and nor exposed to the internet.

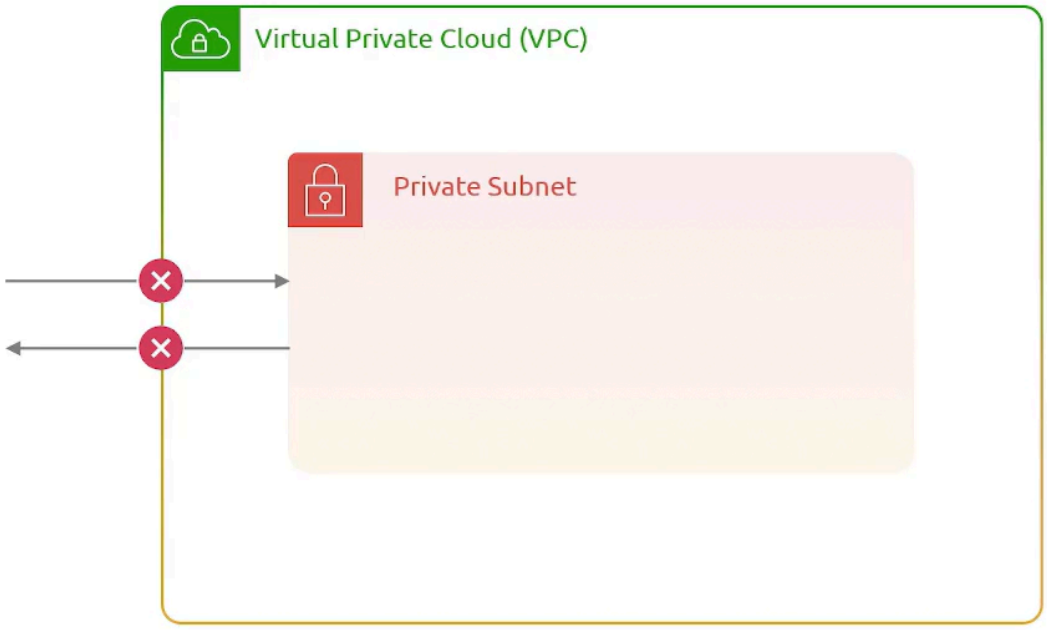


image_71.png

Then if private it means that the resources running inside it cannot be accessed from the internet.

To make a private subnet public an Internet gateway is attached to a VPC to allow subnet in a VPC to communicate with the internet and vice versa.

The ability to access an internet gateway is what determines whether a subnet is public or private.

NAT Gateway

With NAT Gateway a connection must be initiated from within the VPC, if the connection is from the internet its going to be blocked unlike internet gateway.

This means that a private subnet can use the NAT Gateway for Communication with the internet.

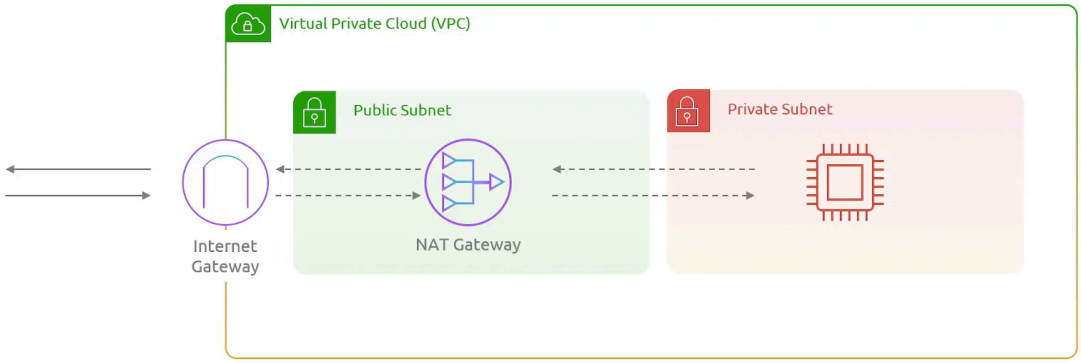


image_72.png

To connect to a private subnet you use a Virtual Network Gateway or Direct connect.

Default VPCs

There a two rypes of VPC

- Default
- Custom

Default VPC

Every account has a default VPC in each region.

- Default VPCs have configurations defined by AWS.
- Default VPC uses CIDR block 172.31.0.0/16 (65,536 addresses)
- In each availability zone we will have a subnet with a /20 (4096 addresses)
- An Internet gateway is attached to the VPC
- A route that points all traffic (0.0.0.0/0) to the **internet gateway**
- All the subnet are going to be public
- A Default security group is also created

- A Default network access control list too.

Custom VPC

Custom VPC allow you to create.modify all configurations associated with the VPC.

Firewalls

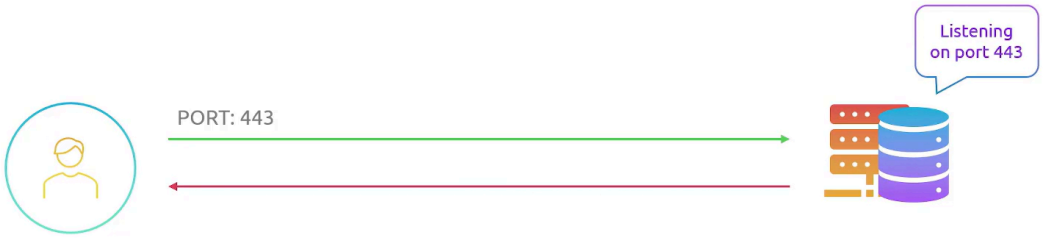


image_73.png

If you application is running on port 443 then we need to make sure that we only allow traffic from that port and not from other ports. That's where firewalls comes in

Firewalls monitor traffic and only allow traffic permitted by a set of predefined rules.

Firewalls rules are broken down into inbound and outbound rules.

Inbound rules - are a set of rules that defines what traffic is allowed from the internet into the resources protected by the firewall

Outbound rules - are a set of rules that define the traffic allowed to the internet

Stateless Firewalls

a Stateless firewall is configured to allow both inbound & outbound traffic.

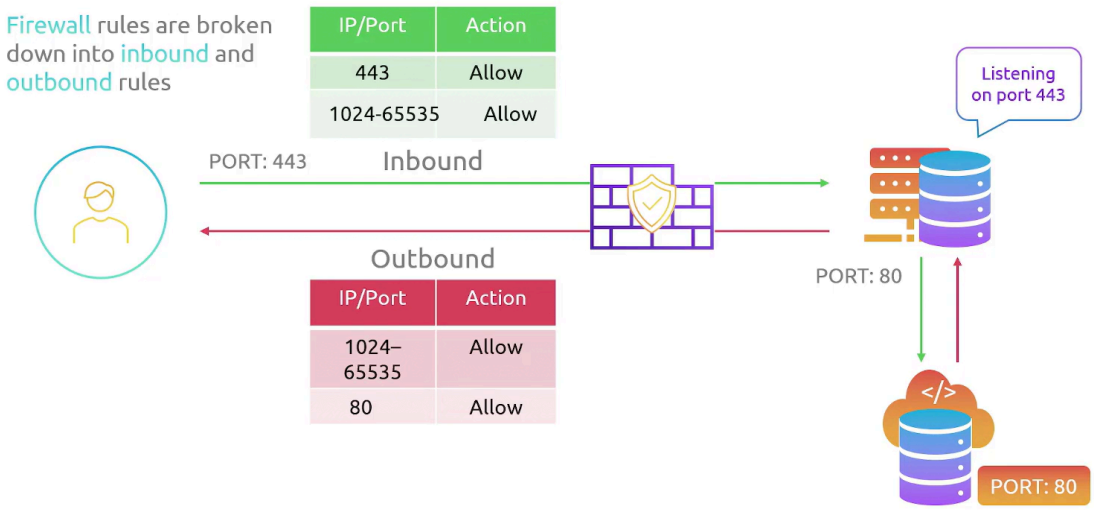


image_74.png

Stateful Firewalls

Stateful firewalls are intelligent enough to understand which request and response are part of the same direction.

If a request is permitted the response is automatically permitted as well in a stateful firewall.

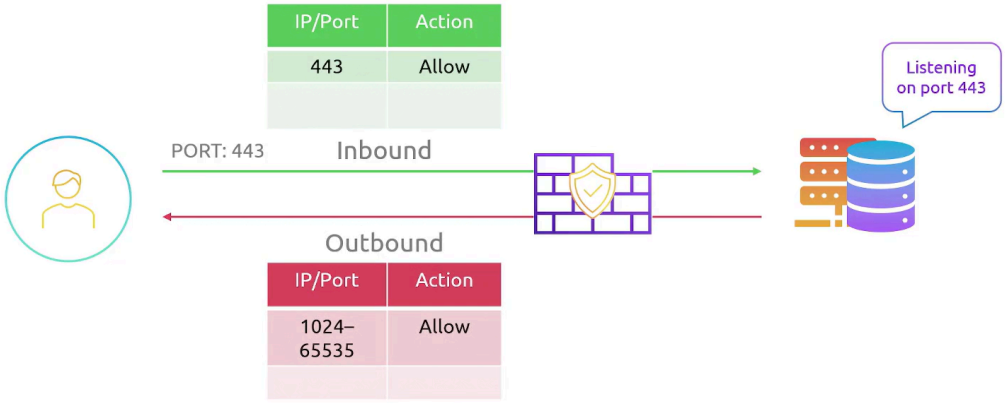


image_75.png

Network Access Control List (NACL)

NACls filter traffic entering and leaving a subnet.

NACLs do not filter traffic within a subnet (If you have two server on one subnet they can talk to each other)

NACs are stateless firewalls so rules must be set for both inbound and outbound traffic.

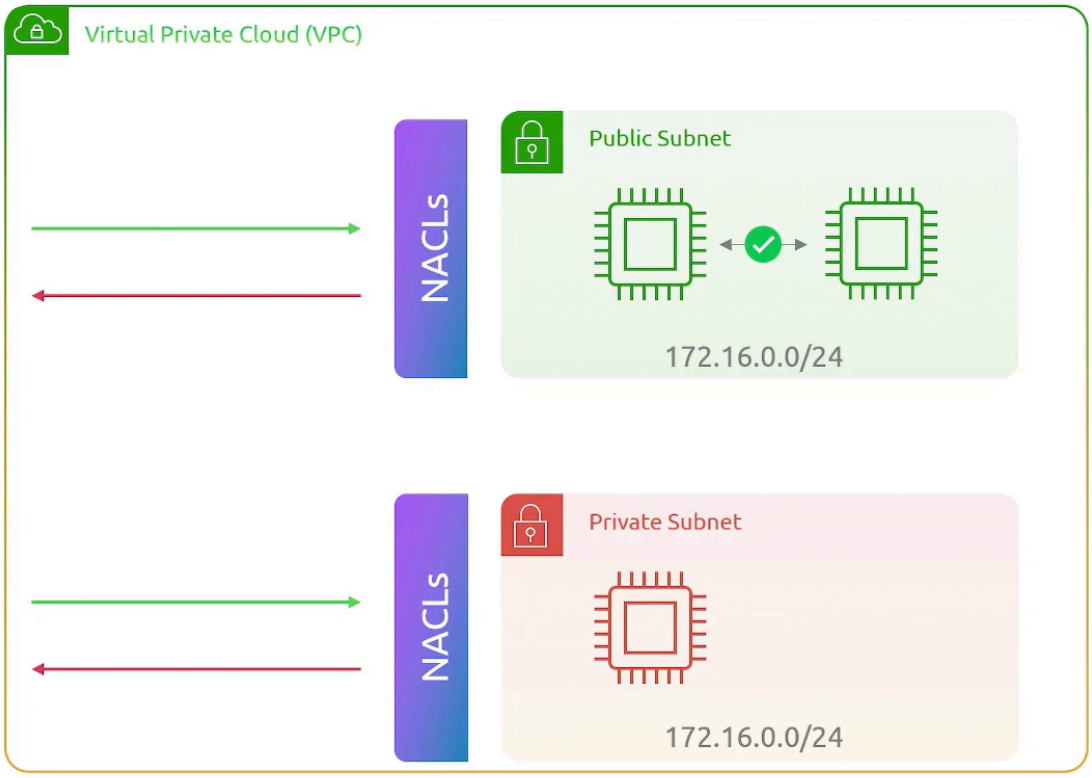


image_76.png

Security Groups

Security groups acts as a firewall for individual resources (EC2, LB, RDS)

Security groups are stateful so only the request needs to be allowed.

NACLS VS Security Groups

NACLs vs Security Groups

- NACLs monitor traffic entering and leaving a subnet
- NACLs are stateless firewalls
  - Traffic must be permitted in both ingress and egress directions
- Security groups act as personal firewalls for individual resources
- Security groups are stateful
  - Only the direction of the request needs to be permitted
  - The response will automatically be permitted as well

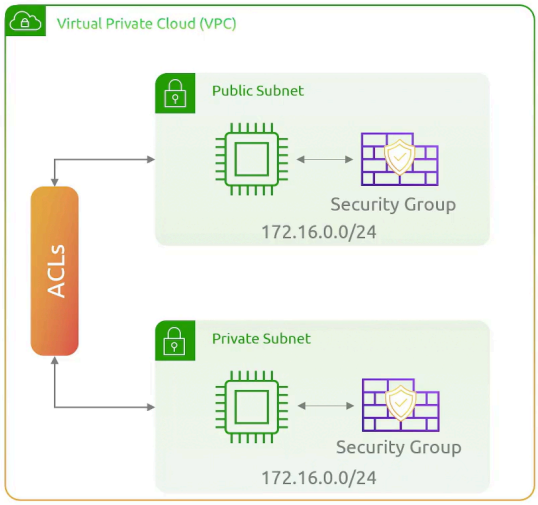


image_77.png

Her is like the whole picture of Networking

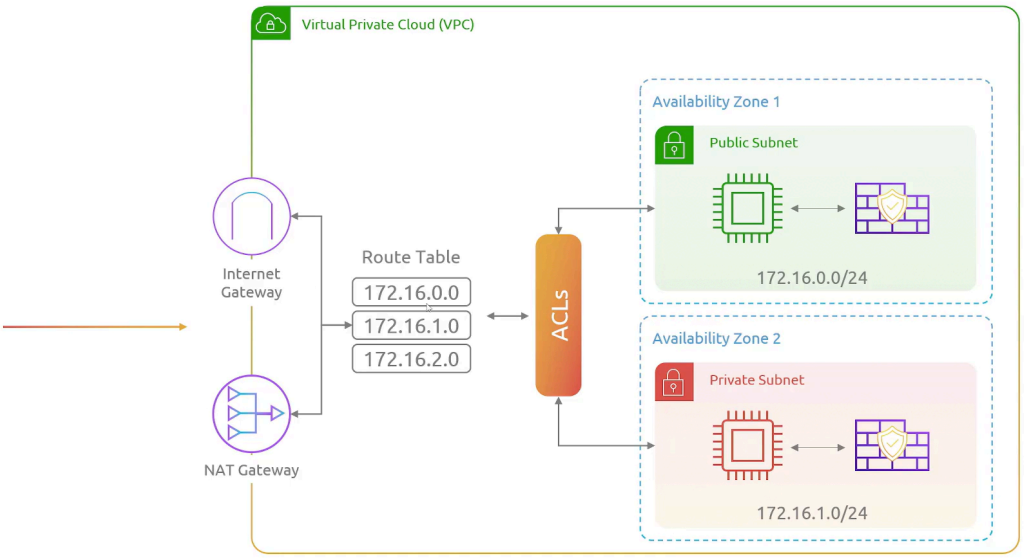


image_78.png

AWS Storage

Types of Storage

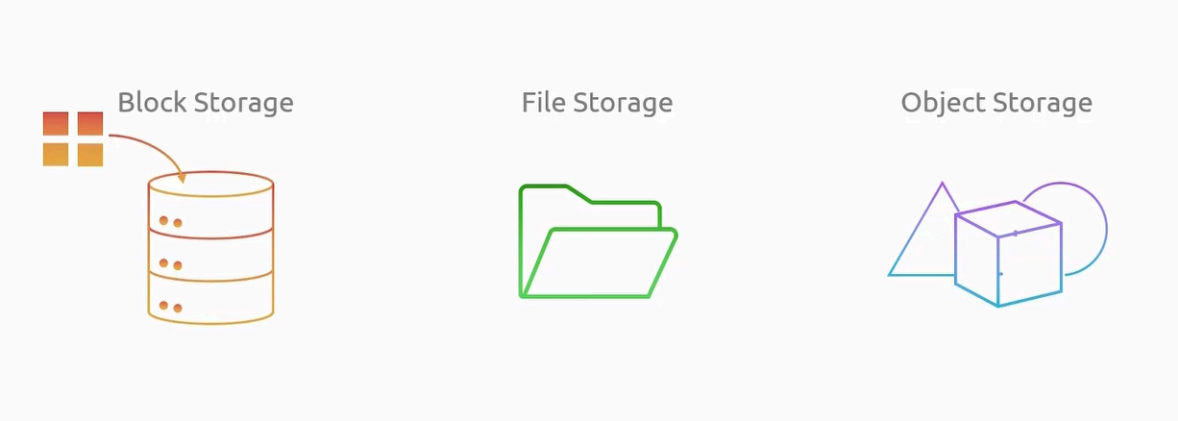


image_79.png

Block Storage(Elastic Block Store)

Amazon has a service for block storage called **Elastic Block Store**

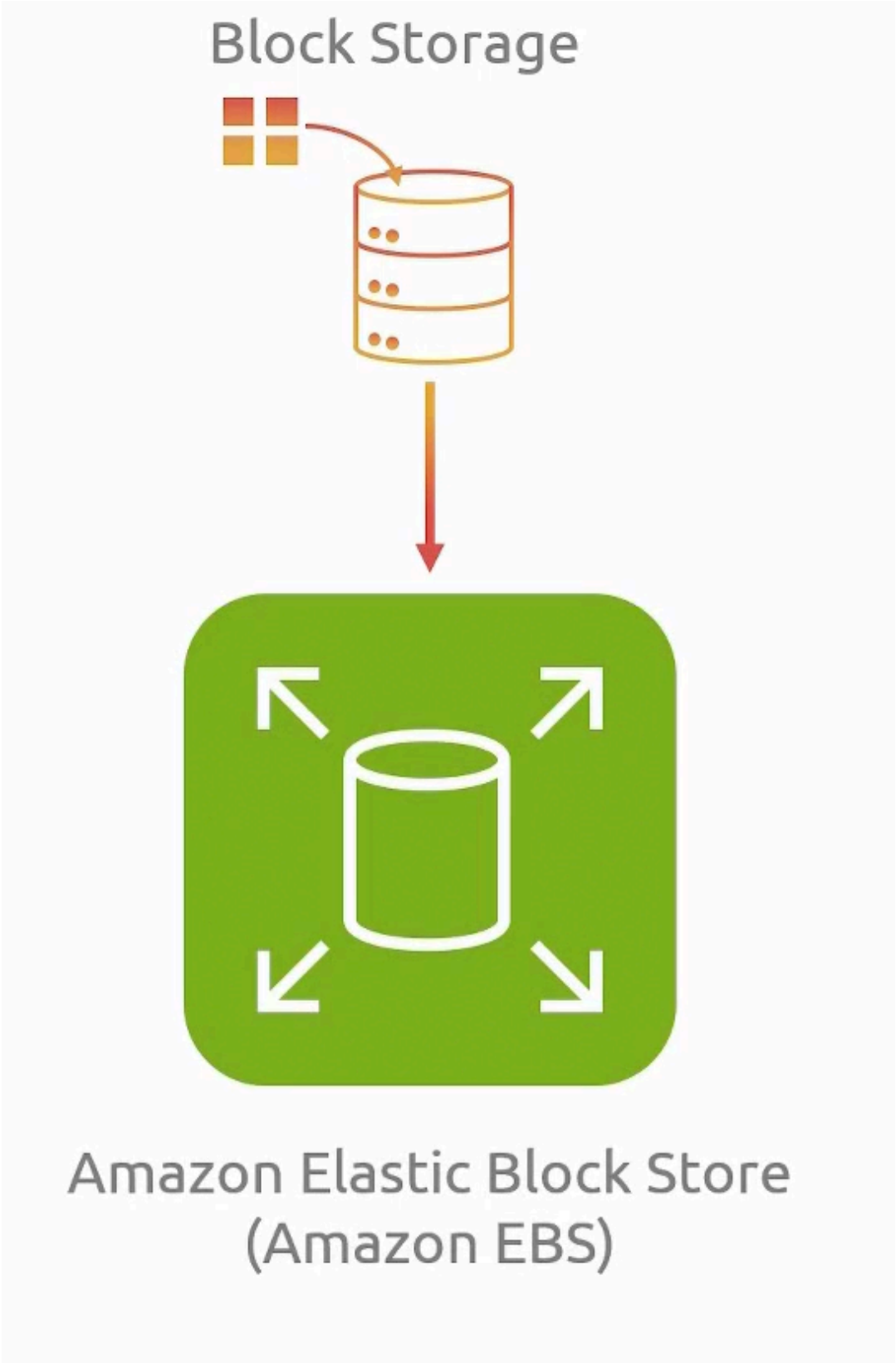


image_81.png

Block storage breaks up data into blocks and then stores those blocks as separate pieces, each with a unique identifier.

A collection of blocks can be presented to the os as a volume

- The operating system the creates a filesystem on top of it.

A collection of blocks can be presented as a hard drive

- Block storage is bootable(Only storage that is bootable), which means **operating systems** can be installed on it.

EBS volume and EC2 Must be in one availability zone to connect.

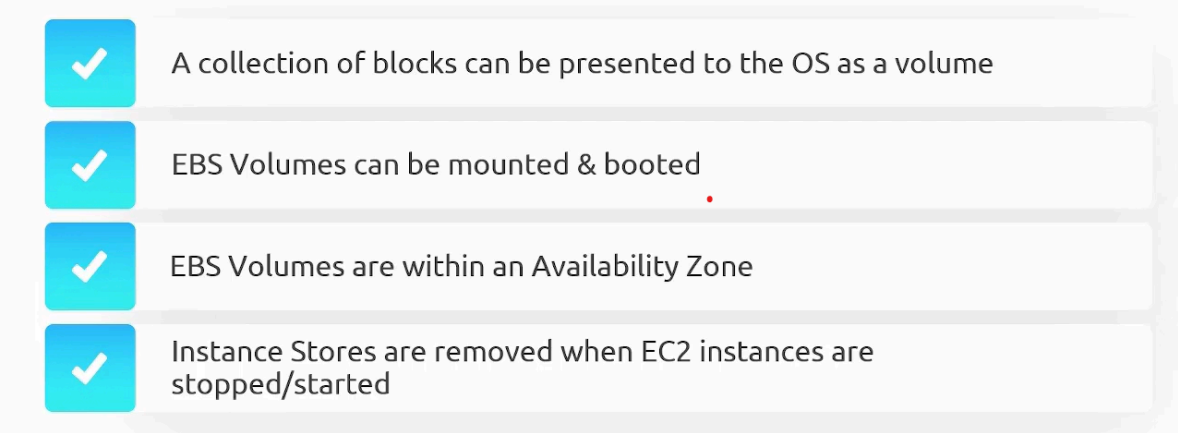


image_80.png

File Storage(Elastic File Storage)

File Storage - Stores data in a hierarchical structure of files and folders

- similar to how your computer stores files/folders.

Filesystem is accessible remotely, multiple clients can access the same data.

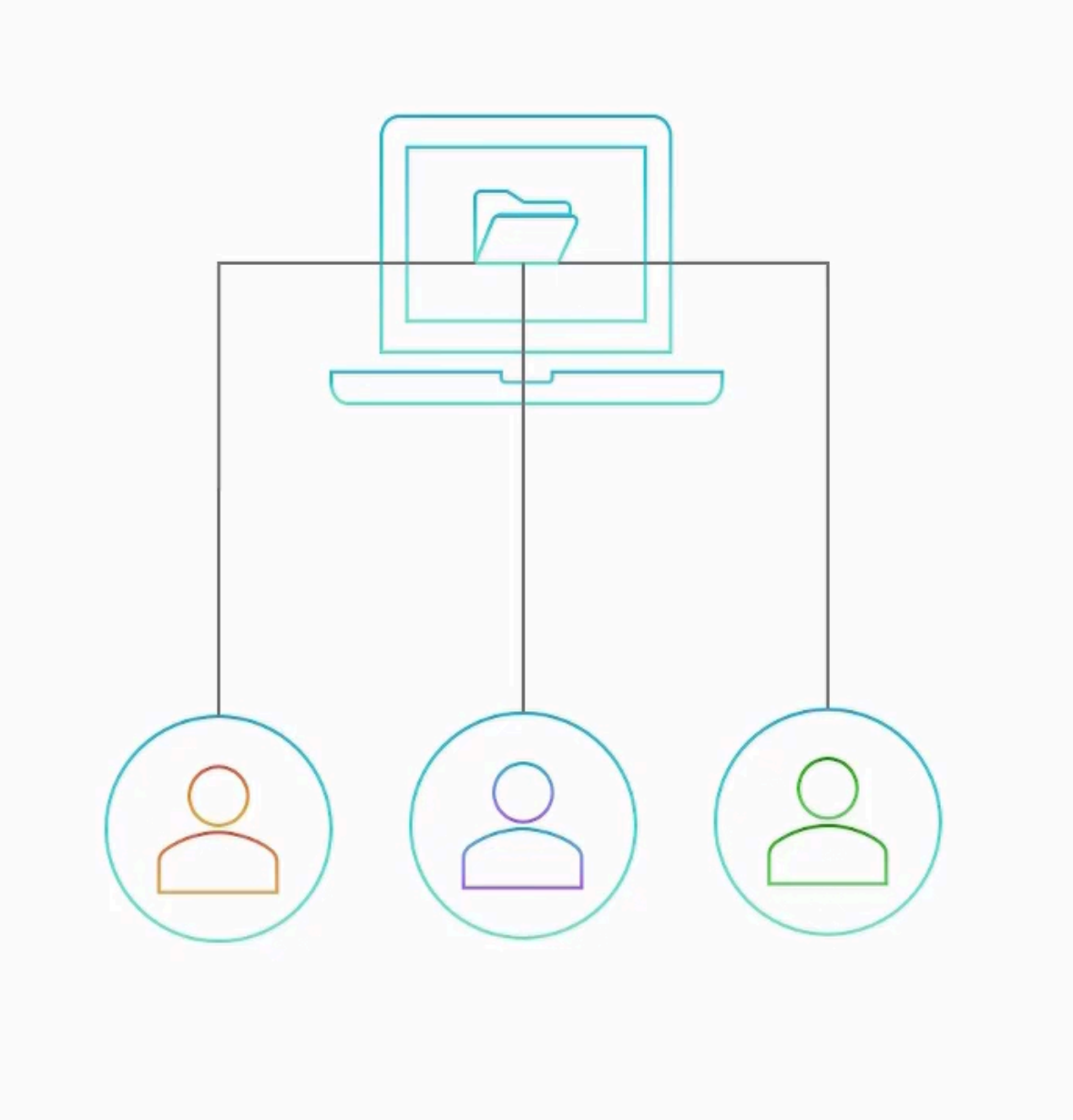


image_82.png

Files storage can be mounted but its not bootable , so you cannot install an operating system

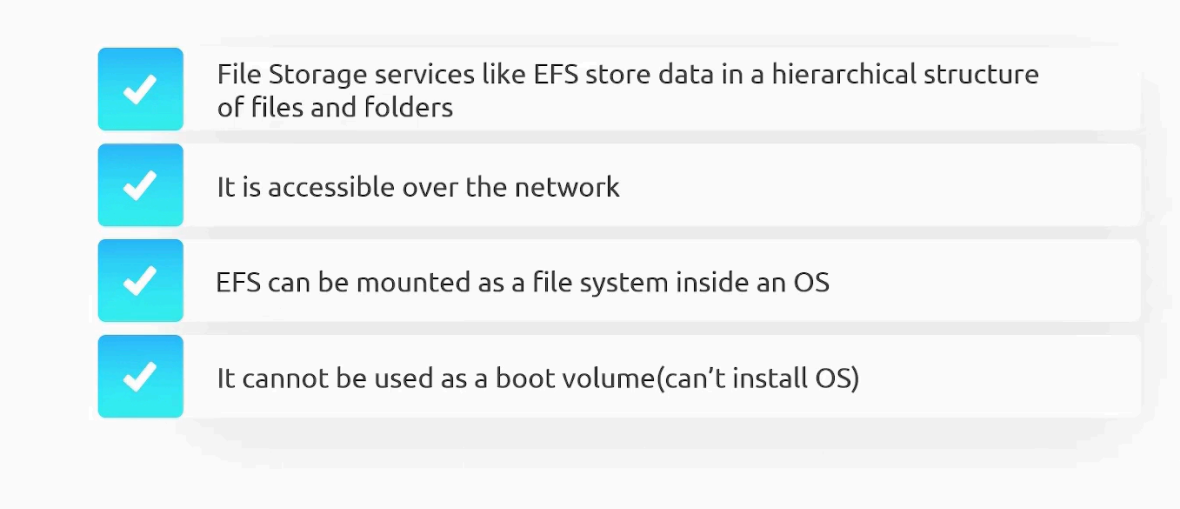


image_83.png

S3- Object storage

Object storage stores objects

- Objects are nothing more than files
- Can store any type of file Does not have a folder structure (Think of it as Google drive or DropBOx)
- Flat file system - where everything is in the same folder Since there is no folder structure, object storage cannot be mounted or booted Great fo storing logs and media files

Types of S3 Storage Classes

S3 has different classes because the following factors make S3 Classes different:

- Data Access
- Resiliency
- Cost

S3 Standard(Default)

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- This is the most expensive S3 storage class
- Data will be accessed instantly
- You will be charged based on the number of gigabytes you use in a month.

if you don't need to access data frequently you can use:

S3 Standard-IA

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- You will be charged based on the number of gigabytes you use in a month.
- Cheaper that standard
- You have a retrieval fee
- Has a minimum duration charge of 90 days -If you store an object in these classes and delete it before the minimum duration ,you will still be charged as if the object was stored for the full 90 days.
- Minimum size charge of 128 KB per object- If you store an object in these storage classes that is smaller than 128 KB, Amazon S3 will still charge you as if the object were 128 KB in size.
- If data is accessed frequently or files are too small then you will pay more

S3 One Zone-IA

- Similar to S3 Standard IA but the difference is data is stored in one Availability Zone but replicated in multiple data centers which makes it cheaper
- You have a retrieval fee
- Has a minimum duration charge of 90 days

- Minimum size charge of 128 KB per object

S3 Glacier-Instant

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- Low-cost option for rarely accessed data
- Performance same as that of S3 standard
- You have a retrieval fee but higher that the previous ones.
- Has a minimum duration charge of 90 days
- Minimum size charge of 128 KB per object
- Cheaper than the previous
-

S3 Glacier-Flexible

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- Data is not immediately available , it takes sometime to retrieve the data
- You have a retrieval fee
- How long to wait for the data to be retrieved depends on the option selected:
  - Bulk : 5-12 Hours
  - Expedited: 1-5 Minutes
  - Standard: 3-5 Hours (The less the time the higher the cost)
  - During retrieval data is stored in S3 standard-IA class temporarily.
- Has a minimum duration charge of 90 days
- Minimum size charge of 40 KB per object

S3 Glacier Deep Archive

Cheapest storage class in S#

- You have a retrieval fee
- How long to wait for the data to be retrieved depends on the option selected:
  - Bulk : 12Hours
  - Standard: 48Hours
- Has a minimum duration charge of 90 days
- Minimum size charge of 40 KB per object

What happens when you don't know the storage class to use?

S3 Intelligent-Tiering

- Here AWS will analyse the access pattern on a file and move it to a fitting category.
- Automatically reduces storage costs by intelligently moving data to the most cost-effective access tier
- Apart from the cost of the storage class and all objects also incur a monitoring/automation cost per 1,000 objects.

How to Select the Best Storage Class

How do you select the best storage class , use the diagram below to guide you.

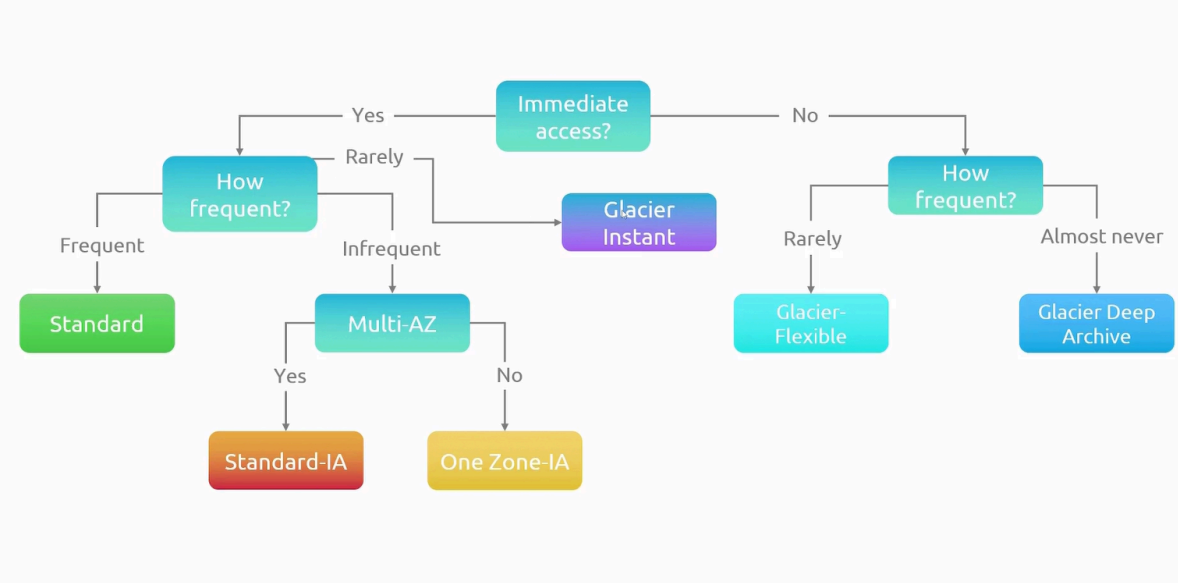

Immediate access?
Yes
No
How frequent?
Rarely
Glacier Instant
Frequent
Infrequent
Standard
Multi-AZ
Yes
No
Standard-IA
One Zone-IA
How frequent?
Rarely
Almost never
Glacier-Flexible
Glacier Deep Archive

image_84.png

Compute Services

In order to run an application we need servers in order to deploy our code.

Before AWS :

You needed to find a data center or build a data center then:

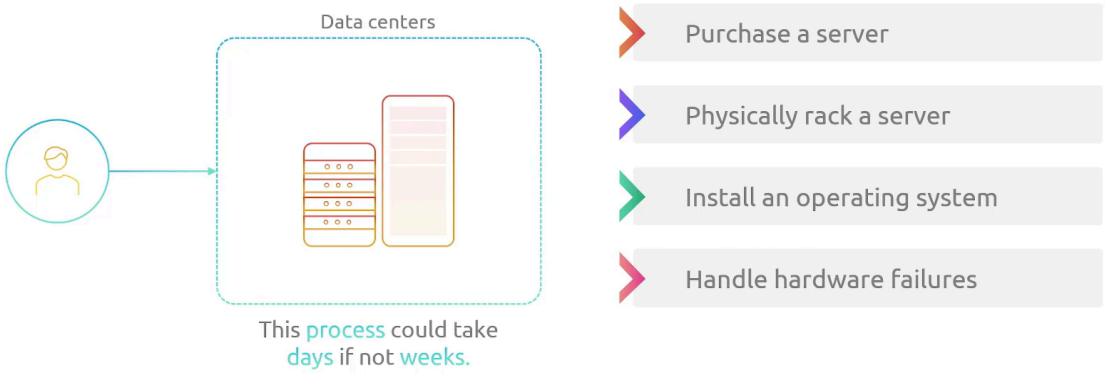


image_85.png

This is time-consuming , costly and involved multiple people.

With AWS a Server is Just a Click-away. The service responsible for providing servers in the cloud is an EC2.

Amazon EC2

EC2 allows you to provision a server on AWS within minutes, it provides secure, resizable compute capacity in the cloud

There is a potential that An AWS server has several EC2 instances deployed there. VPCs are used to logically isolate each customers' infrastructure.

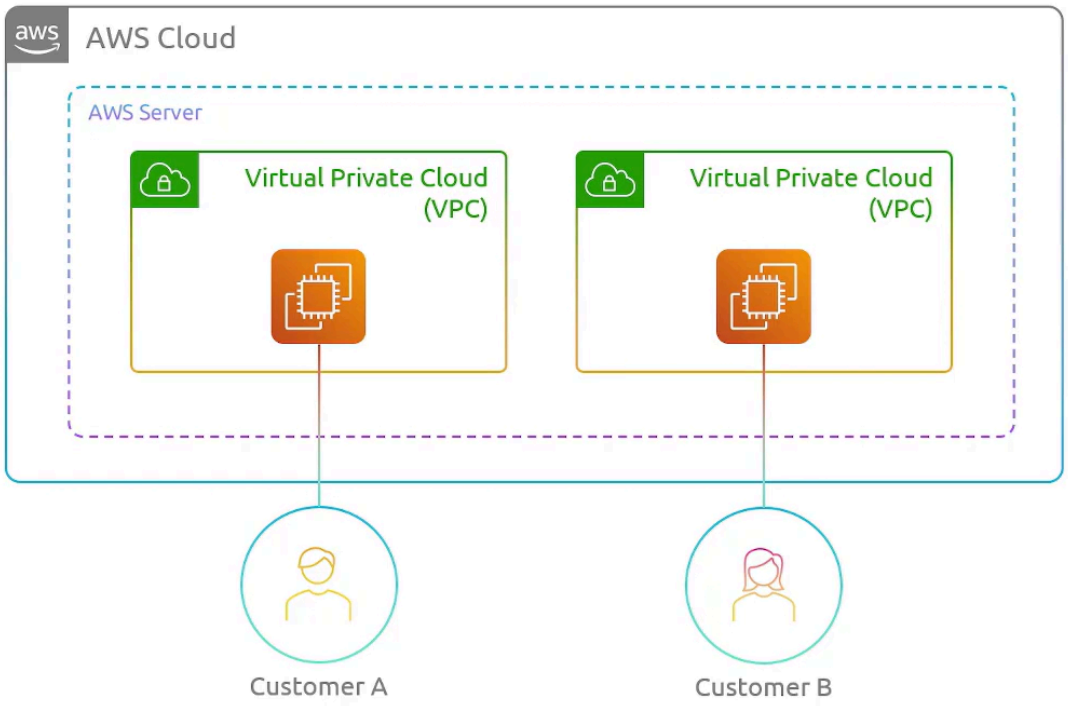


image_86.png

If you are using a physical server you will have to install an operating system but what about EC2 instances?

Amazon Machine Images(AMI)

AMI is an image that provides the information required to launch an EC2 instance.

AMI is a blueprint that contains information such as what operating system and what software/packages should be installed on an instance.

We can use one AMI for multiple EC2 instances. Think of AMI as an installer disk used for installing a new Server

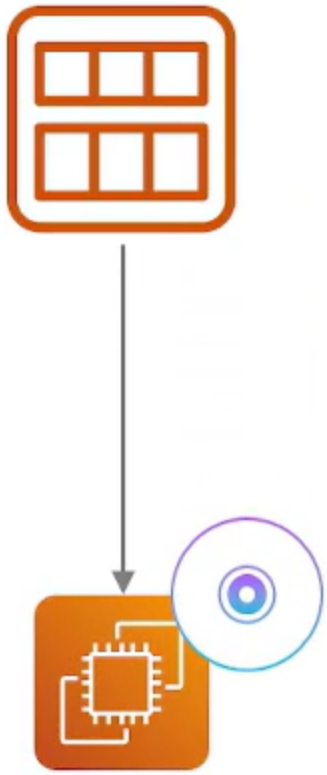


image_87.png

AMI Can be fully customized to:

- Add application source code
- Add dependencies
- Customize OS firewall

If you are shopping for a physical server there will be specifications that you will consider e.g. Memory, CPUs even storage Now how does that happen in the cloud.

Instance Types

EC2 provides a wide selection of instance types optimized to fit different use cases.

Instance types have varying combination of CPU, Memory, storage, and Networking capacity. They are just different server models to select from.

There are different types:

General Purpose

- Provide a good balance of compute, memory and networking resources
- Can be used for diverse workloads
- Ideal for applications that use resources in equal proportions

Compute Optimized

- Optimized for compute-heavy applications
- Contain high-performance CPUs
- Ideal for batch processing workloads, media transcoding, machine learning and gaming servers.

Memory Optimized

- Deliver fast performance for memory-intensive workloads
- Suited for database

Storage Optimized

- Optimized for workloads that require high, sequential read and write access to large data sets on local storage.
- Can deliver tens of thousands of low-latency, random I/O operations per second (IOPS)

Accelerated Computing

- Utilize hardware accelerators to perform expensive calculations
- Great for graphics processing and data pattern matching

EC2 Pricing

On-demand Pricing

Here you pay for compute capacity by the hour. Only billed when instance is running:

- Still charged for storage attached to instance NO upfront payment or long-term commitment
- Great for short-term, irregular or unpredictable workloads.

Spot Pricing

Amazon tries to maximize the usage of their servers. For example if they have a machine with 64GB and the EC2 instances running inside the machine only use 20 GB then 44GB is idle.

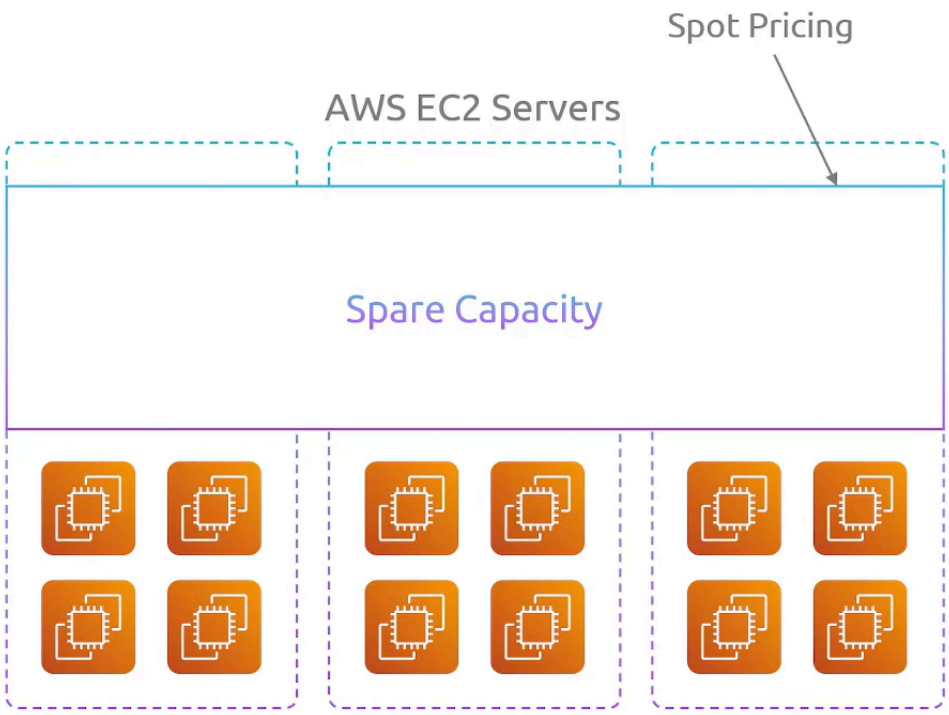


image_88.png

Instead of it being idle, Amazon offers spare compute capacity at discounted rates Spot instances are recommended for:

- Applications that have flexible start time and end time

- Applications that need low compute prices Not suitable for application that cannot tolerate interruptions.

Reserve Pricing

Reserve instance is a billing discount that allows you to save on your EC2 costs. Offers discounted rates when reserved for a certain period (1year or 3 year contract)

When purchasing a reserved instance, you are not actually buying an instance. You are merely committing to using an on-demand instance for a long-term period.

When you deploy an on-demand instance with matching attributes as the reservation(instance type, region, platform), it will be charged at the reserved price and not the default.

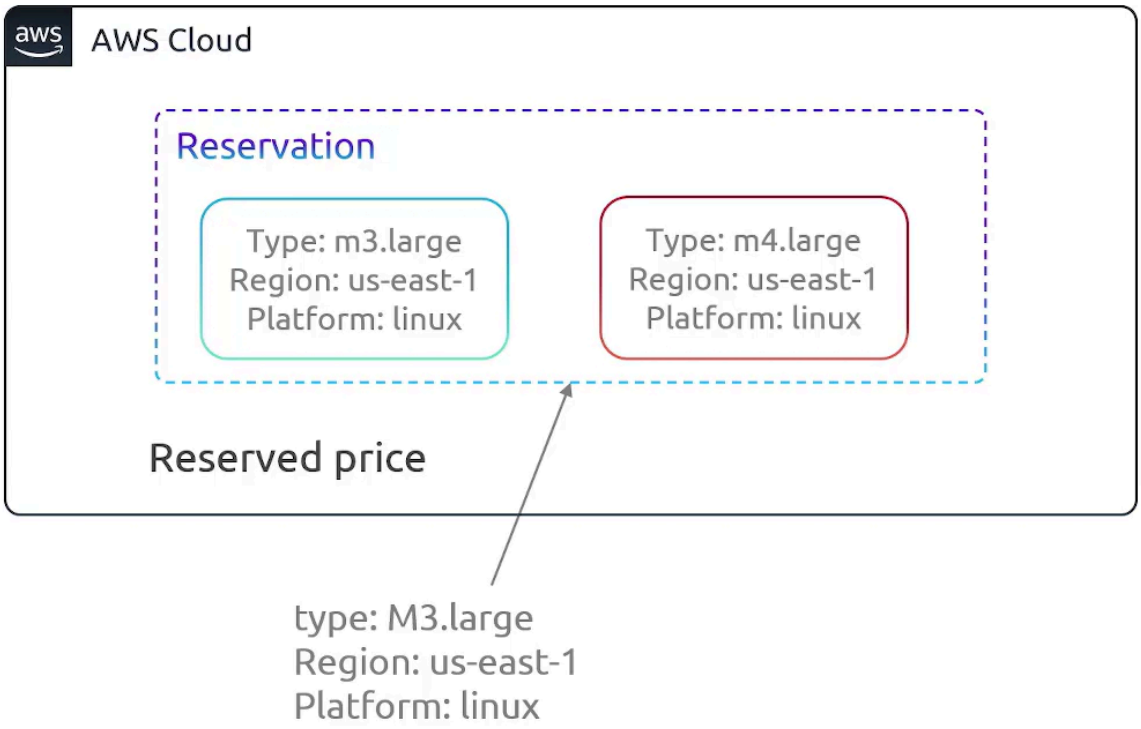


image_89.png

In the above diagram,what reserved instance means is any instance that is M3.Large you will be charged as a reserved instance but any other type of instance you will be charged as an On-demand instance.

AWS Lambda

Before we check on Lambda let check at what we have right now:

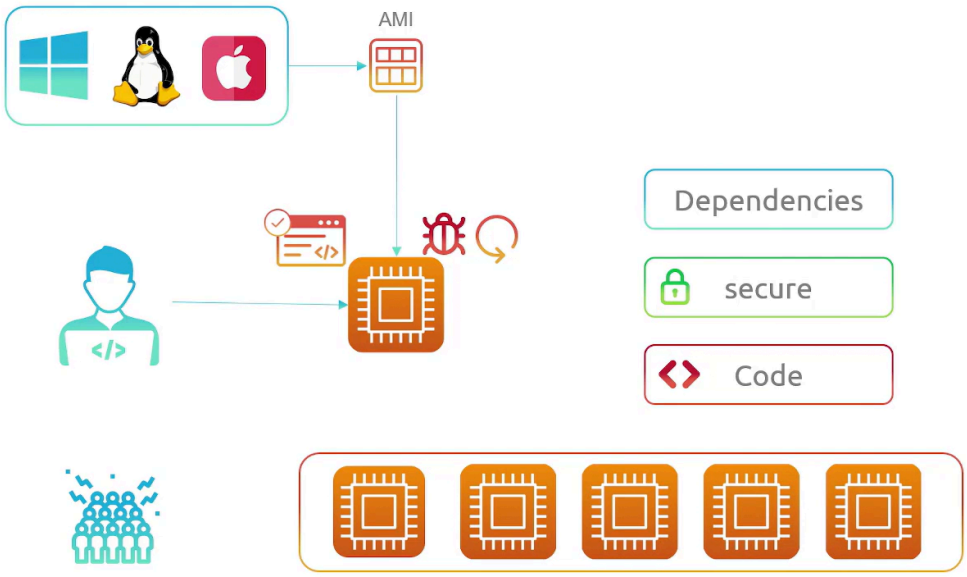


image_90.png

When we want a server running, we must choose the right AMI, we must add the right dependencies , we must copy the code and Even after the application is running we must watch out for updates and software patches.Not forgetting that in case of traffic we need to spin up or down more servers. That's a lot of work and a developer just wants to write code. This is where AWS Lambda Comes in

What is Lambda

AWS Lambda is a compute service that let's you run your code without having to provision or manage servers.You just copy the code to AWS and AWs will manage the underlying infrastructure like starting a server and running it. Lambda is AWS' serverless offering:

- AWS manages the server maintenance,scaling, capacity provisioning and logging.

But serverless , does that mean there is no servers?

Serverless

No, to run and application there will always need to be a server but with Lambda serverless means from the user perspective you just upload your code and then AWS will manage the infrastructure.

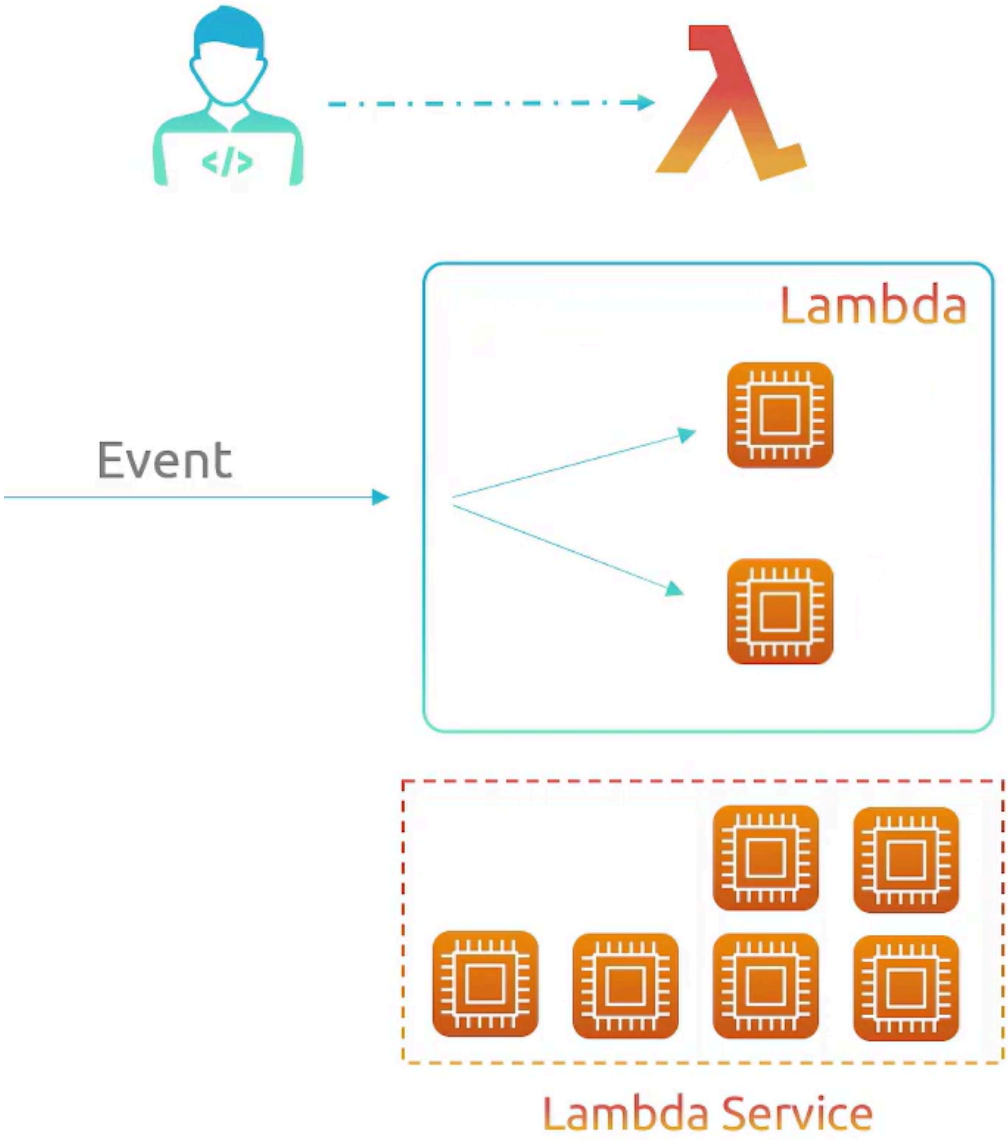


image_91.png

What happens is AWS will have a Lambda service reserve pool that contains several servers and there will be an event that will be triggered for you code to run, when the event is triggered then some servers are pulled from the list of servers in the service pool

, they will run the code and when done they will be freed and taken back to the service pool.

Lambda Use Case

- File processing - Eg resizing an image everytime an image is uploaded.
- Stream processing
- Web application
- Mobile/ Web backend

Lambda Components

Lambda Function

This is basically your code

```
exports.handler = async function
(event, context) {
  console.log("Lambda function ran");
  return;
};
```

image_92.png

Trigger

Trigger determines when your lambda function should run.

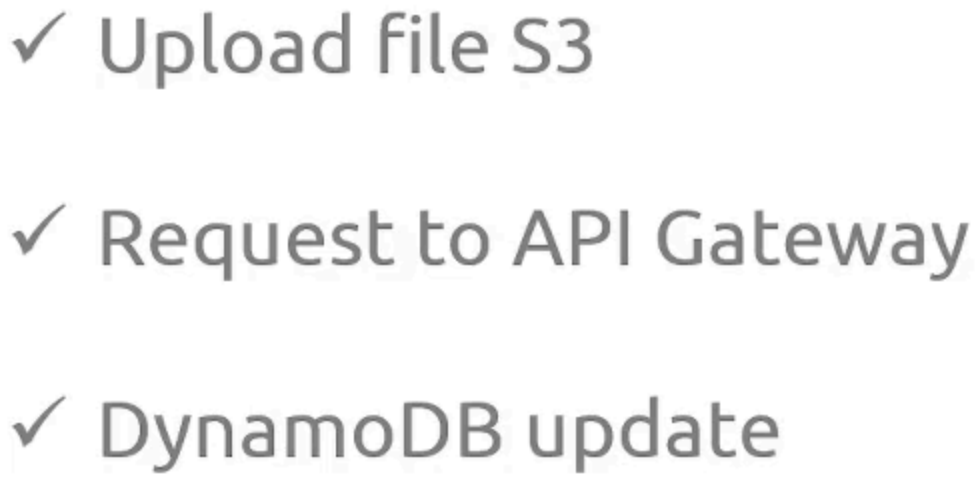


image_93.png

Event

Event is the information of what trigger took place, eg of an event is about file upload then the event could include the name of the file .

Benefits of Lambda

- No Server to manage
  - No need for an infra team
  - No patching/upgrading
  - Faster development
- Auto scale to handle traffic spikes
- Pay for what you use
  - pay per invocation

- No extra cost during low traffic

Downsides of Lambda

- No Local state
  - Will need a separate database to store data that needs to persist
- Limited execution duration
  - Function can run at most for 15 minutes
  - Not good for long-running tasks
- Cold starts
  - Cold start occur due to the time it takes to initialize and load a function.

Lambda Pricing

You will be charged for:

- Number of time the function ran
- How long did it run for
- How much memory/ CPU did it require.

Compute Container

What is a container first

Container are a tool that allows you to package an application and all the necessary files, libraries and dependencies the applications need to run.

Containers can be deployed on a machine and since they have everything the application needs to run there is nothing else that needs to be done. They are just small lightweight version of virtual machines.

Container Challenges

Ultimately we want our application to be in multiple hosts, we need to restart faulty container, we need to scale up and down,we also need to replace a host if its not working. . In that case we need something to manage the hosts. This is where container Orchestrators comes in.

Container Orchestrators

Container orchestrators are the brains of a containerized environment. Responsibilities include:

- Deploying containers across all available servers
- Load balancing request to containers
- Providing container to container connectivity
- Restarting failed containers
- Moving containers when hosts fail Here is example services for containers orchestrators

image_95.png

Elastic Container Service (ECS)

Fully managed container orchestration service that helps manage, and scale containerized applications.

- ECS is managed by AWS
- Containers run on EC2 instances or Fargate ECS is proprietary software (only available in AWS)
- Migrating to a different cloud provider will be more difficult.

EKS

Lets first understand Kubernetes

Kubernetes

Kubernetes is an open-source container orchestrator. Kubernetes cluster has two types of nodes

- Control-plane nodes - managers of the cluster
  - Watch over cluster and make sure cluster is kept in a working state
- Worker nodes- responsible for actually running the containerized workload

Managing Control planes and the worker nodes is difficult:

- Running & scaling control-plane
- Properly securing the control-plane

AWS Elastic Kubernetes Service is a managed Kubernetes service

- EKS will manage the control-plane for you
- Users are still responsible for managing worker nodes
- If you use Fargate AWS will manage worker nodes too

Benefit of EKS

- Runs and scales control-plane across multiple Availability Zones
- Scales control-plane instances based on load
- Can integrate with other AWS services
  - IAM for authentication
  - Elastic load balancer
  - ECR for container Images

ECS VS EKS

ECS is proprietary to AWS so migrating to another cloud provider can be difficult EKS is kubernetes which is open-source and can run on any platform.

- Keep in mind the more AWS service you configure your cluster to interact with the harder it will b to move to another provider because of service specific configuration.

ECs has a simple architecture

- Simpler API

- Easier to ramp up new team members Kubernetes is very complex and has a steep learning curve
- With EKS you'll have to learn kubernetes as well as EKS specific features.

Kubernetes has a larger community

- More support online
- More tooling like Helm/ Kustomize/ArgoCD

ECS pricing - No cost for control-plane , only pay for infrastructure running applications (EC2/ Fargate , EBS) EKS pricing - more expensive, you have to pay for control-plane and worker nodes/ Fargate.

AWS Databases

Database is basically an application that stores data.

Types of Databases (Datastores)

There are three types of databases:

- Self-Managed Data stores
- SQL/Relational Data stores
- NoSQL Data stores

Structured Data VS Unstructured Data

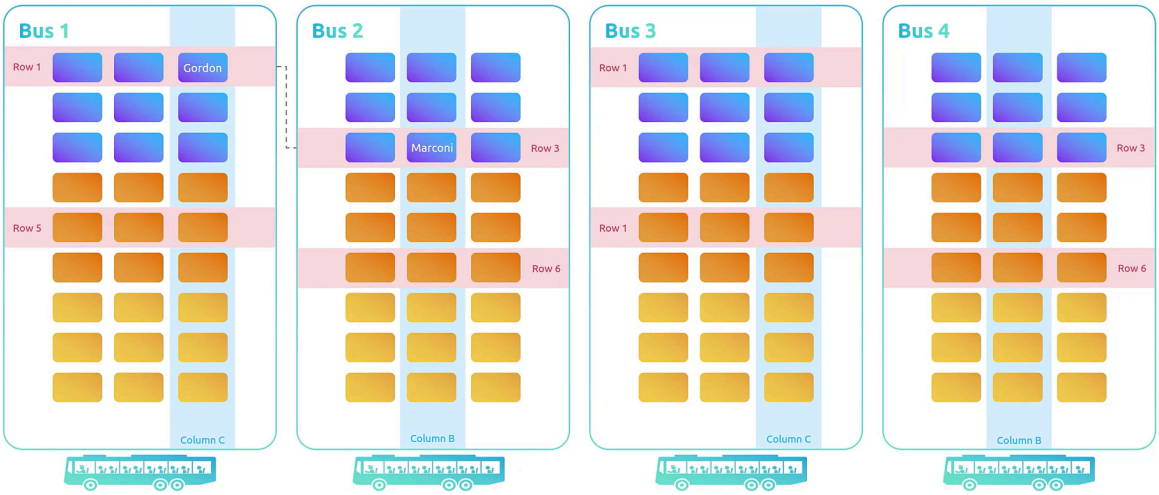


image_96.png

Lets use the above diagram to explain a relational database. Assuming we have buses an in one bus we have Gordon and in the other we have Marconi . If we need to find Gordon we need to go to Row 1 Column 3 Assuming there is a relationship between Marconi and Gordon(friends) Then if marconi moves rows we have to update Gordon where marconi is

This is structured data and what SQL datastore are used for.

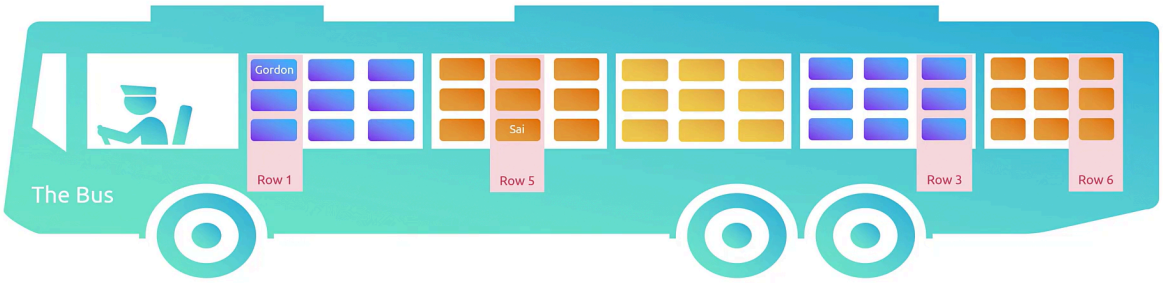


image_97.png

Assuming now we just come and ask if there is Gordon in the bus and Gordon raises his hand. Here we just use a key like 'name ' to search

Employee-Department Mapping Table

EmpID	DeptID	Joined On
01	01	20150101
02	02	20150102

EmpID	Name	Address
01	Emp01	A
02	Emp02	B

Employee Table

DeptID	Name	Location
01	DeptA	LocA
02	DeptB	LocB

Department Table

image_98.png

In Structured data stores , the buses represents tables and the tables have relationship with each other. Data her is a bit structured and organized. Used primarily when you have complex relationship with your data Think Transactional (like banking) or reporting use cases for these data stores

image_99.png

On the other side we just need to know a key like 'movieId' or 'bookName' and just search it. NoSQL is good for search. Used when you have simple but specific needs for data Think search, High performance, Documents, Relationship use case.

Self-Managed Data stores

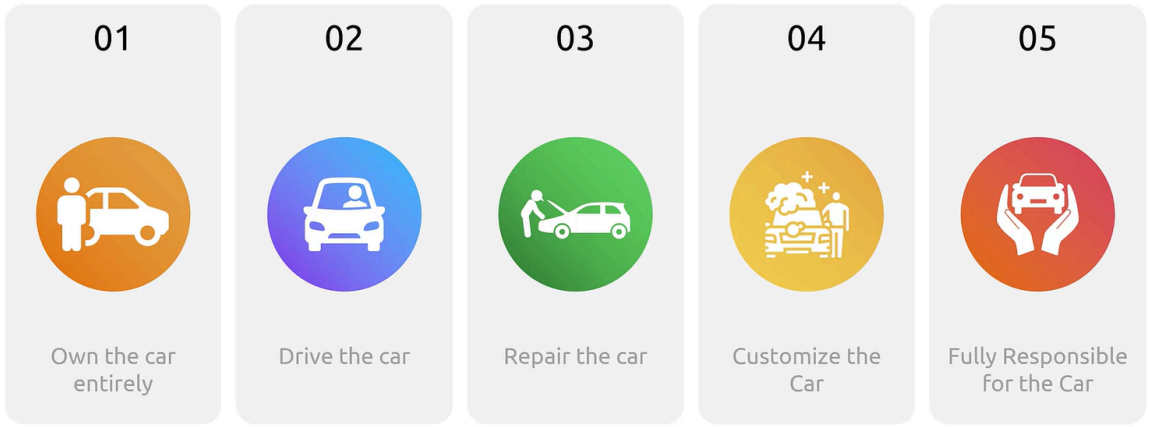


image_100.png

Assuming a database is a car and you want full control of everything Own the car entirely (Own the database) Drive it (In charge of the database) Repair the car (Repair database in case of any issue) Customize the Car (Customize it how you want it, make updates too) Fully Responsible for the car (You in charge of everything about the database)

If you decide to use EC2 Virtual Machines or ECSor EKs that is a Self-Managed Database.

- Can run your own database software on EC2 or any container services (ECS or EKS)
- Increased control and Flexibility

- Increased Operational overhead and responsibility
- Mainly used when you have specific software or security requirements.

image_101.png

SQL Data stores (Structured)

Back to the Car database

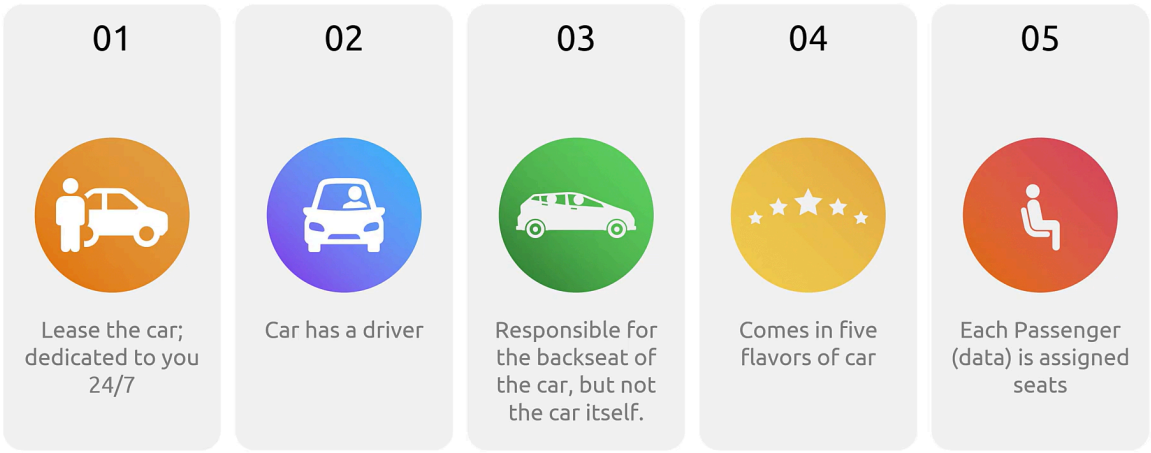


image_102.png

We have 5 databases in this case:

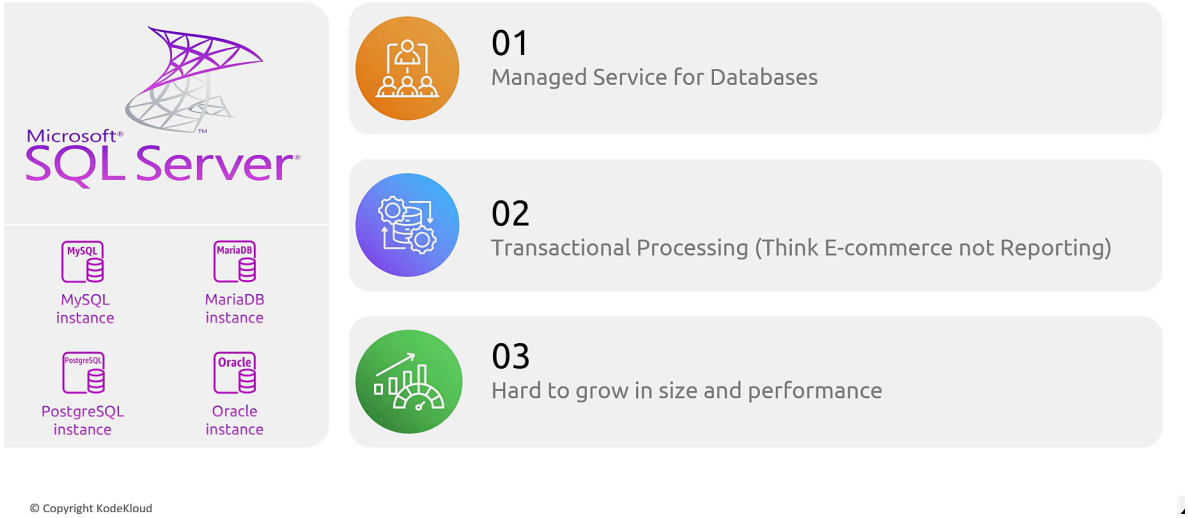


image_103.png

But it hard to grow in terms if write but easy to grow in terms of read.

But what if you want something that is a bit faster .

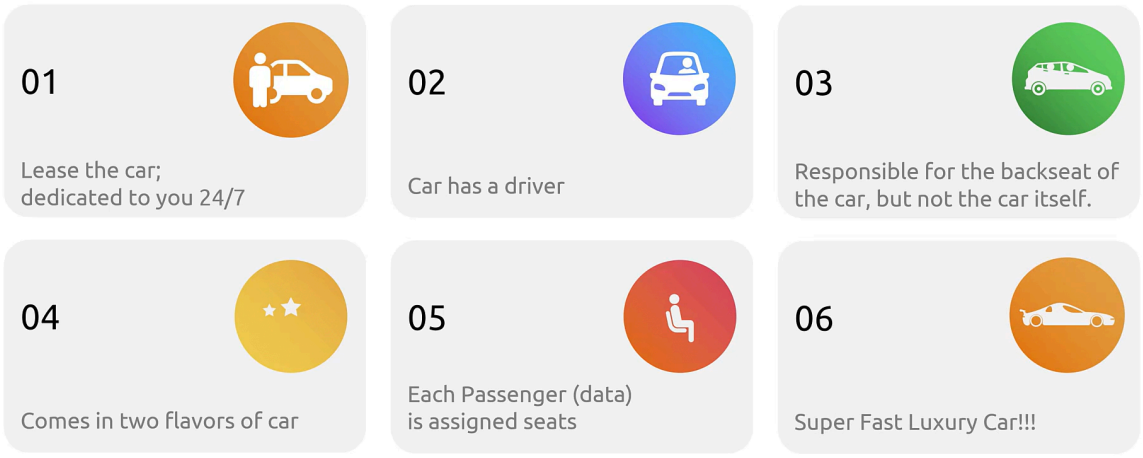


image_104.png

This ia where Aurora comes in

SQL Data stores - Aurora (really RDS Aurora)

Here they took Postgres and MySQL from the traditional RDS and made them :

- Higher capacity and higher performance
- Grows more easily that the main RDS service
- Managed services for Databases
- Cloud Native

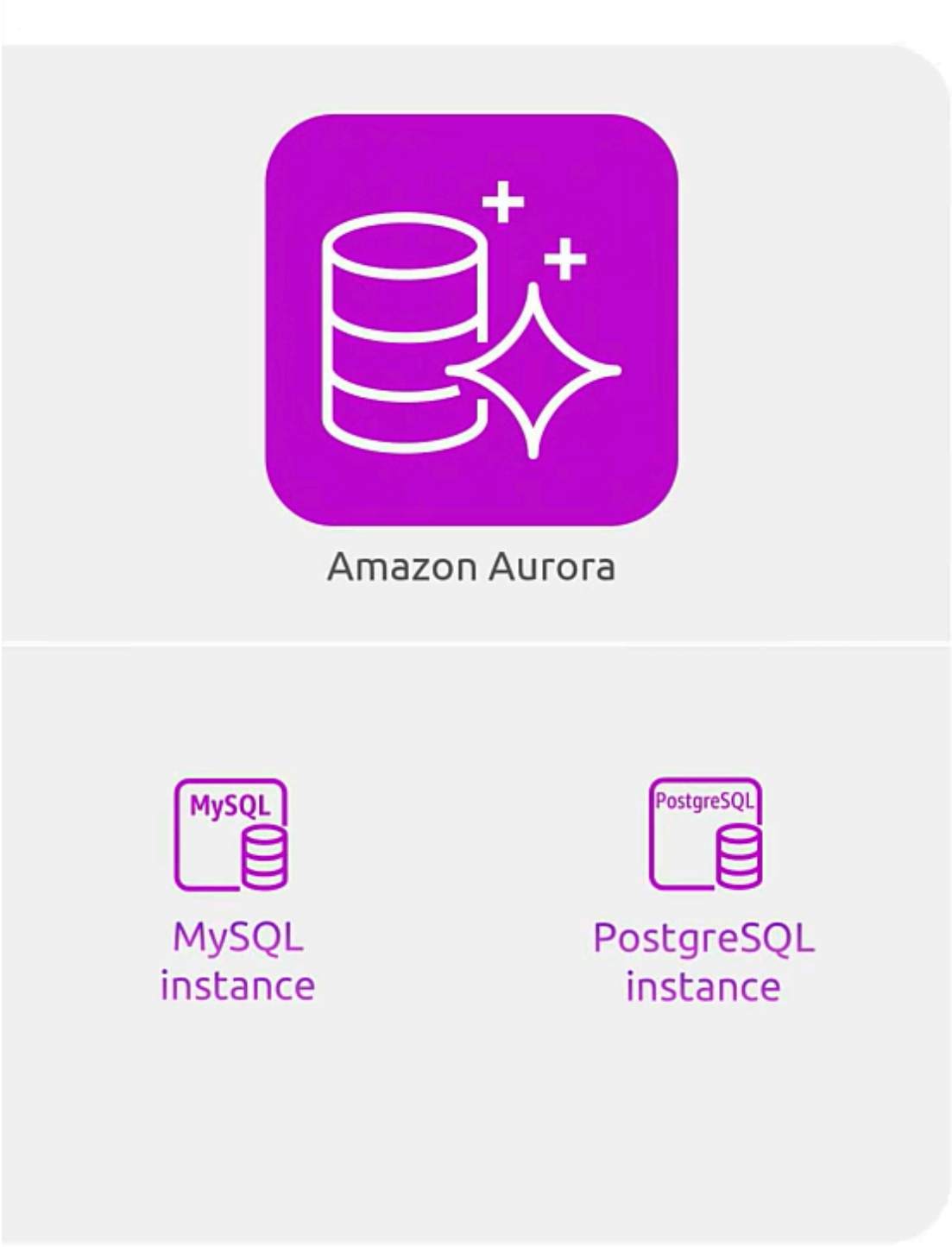


image_105.png

But assuming again you want something that you only pay for it when you use it (more of an uber here) Here we have:

Aurora Serverless V2

Again here we have a :

- Managed service for databases
- Cloud native
- Higher capacity and higher performance
- Capacity can go up and down easier than other RDs servicePay little storage, but not compute when you are not using it

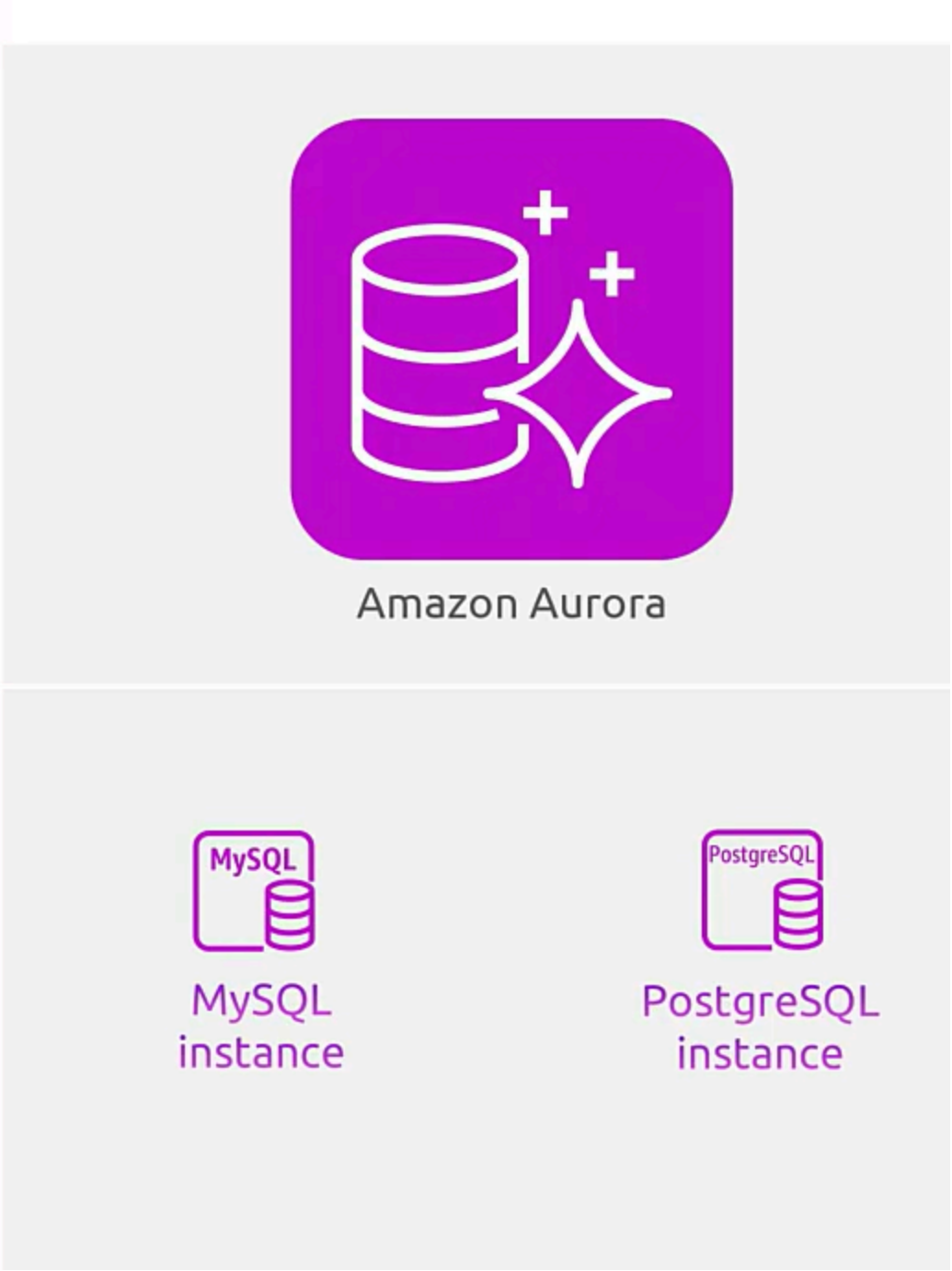


image_106.png

But sometimes you have a lot of data that needs analyzing and processing reports (you need a bus not a car)

RedShift

This is the SQL data warehouse service is SQL. Sometimes you don't want transactional database but a data warehouse. RedShift is used when:

- When you need a data warehouse not a transactional data store
- it's the SQL data warehouse in AWS
- Petabyte scale
- Serverless and 'Server' ed versions
- Think reporting and analyzing not transactional

Short Summary

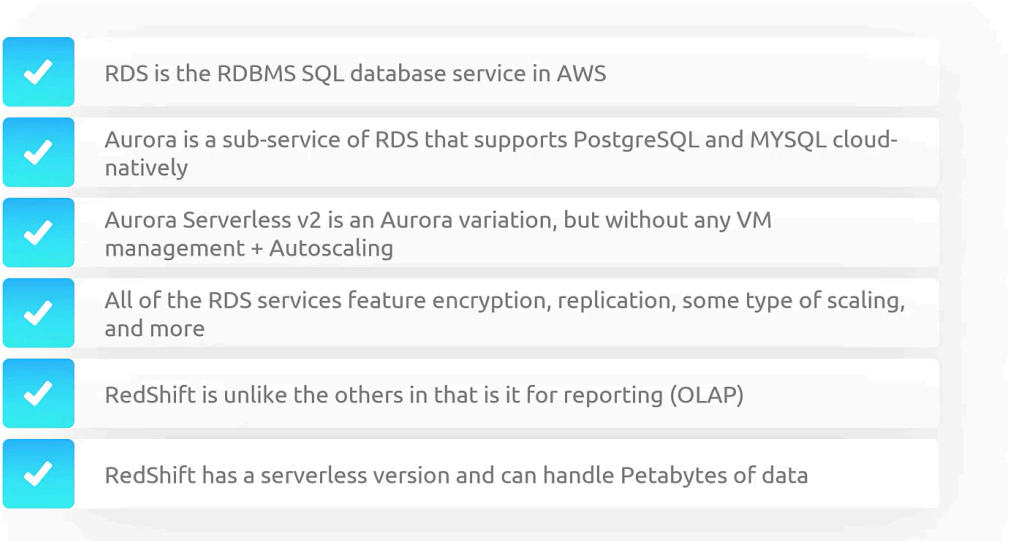


image_108.png

AWS NOSQL Data Sores

Amazon DynamoDB

This is 'The lightning-fast king of key-value at AWS'

If you want to create blobs of data that you can search for with a single keyword or phrase we use **Amazon DynamoDB**

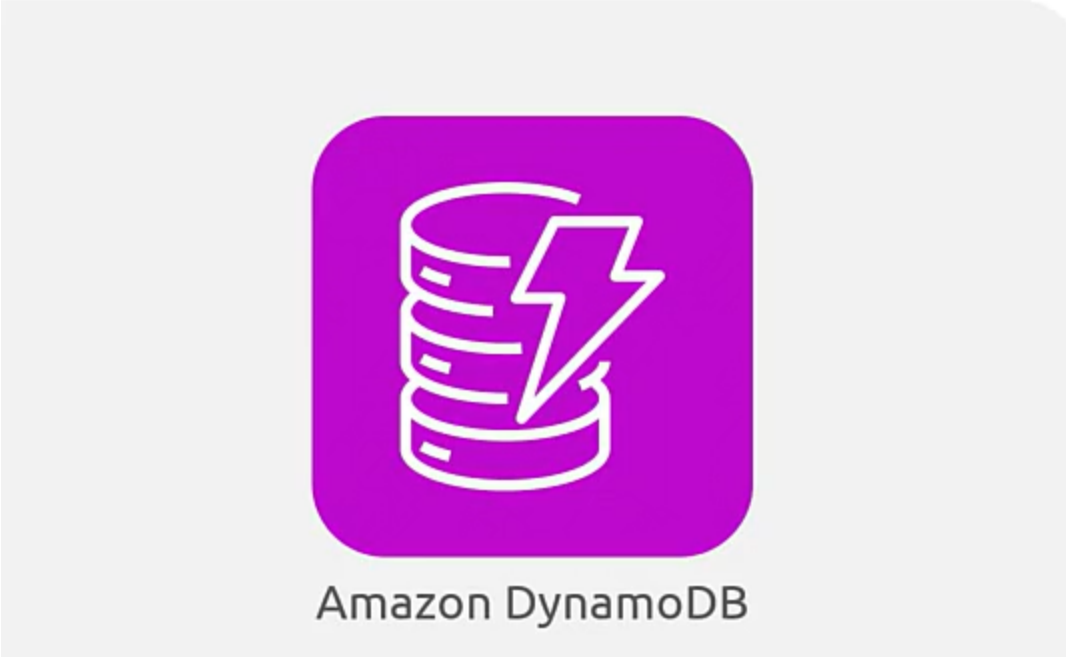


image_110.png

DocumentDB

We use this Service when we want to retrieve a document like essays, profiles and more like collections of data.

image_109.png

Keyspaces (Cassandra Compatibility)

you use these when you want a database that can run in many different locations across the planet and you need large-scale unstructured data.

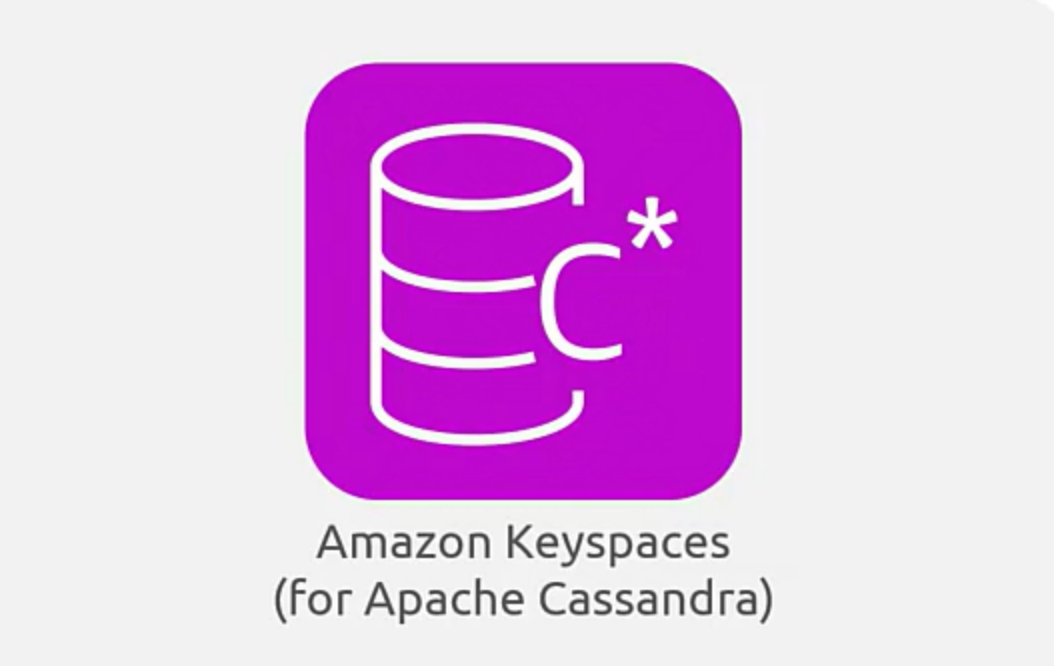


image_111.png

Neptune

When you need a database that will detect relationship between data like fraud detector or social network relationship this is when you use **Amazon Neptune**

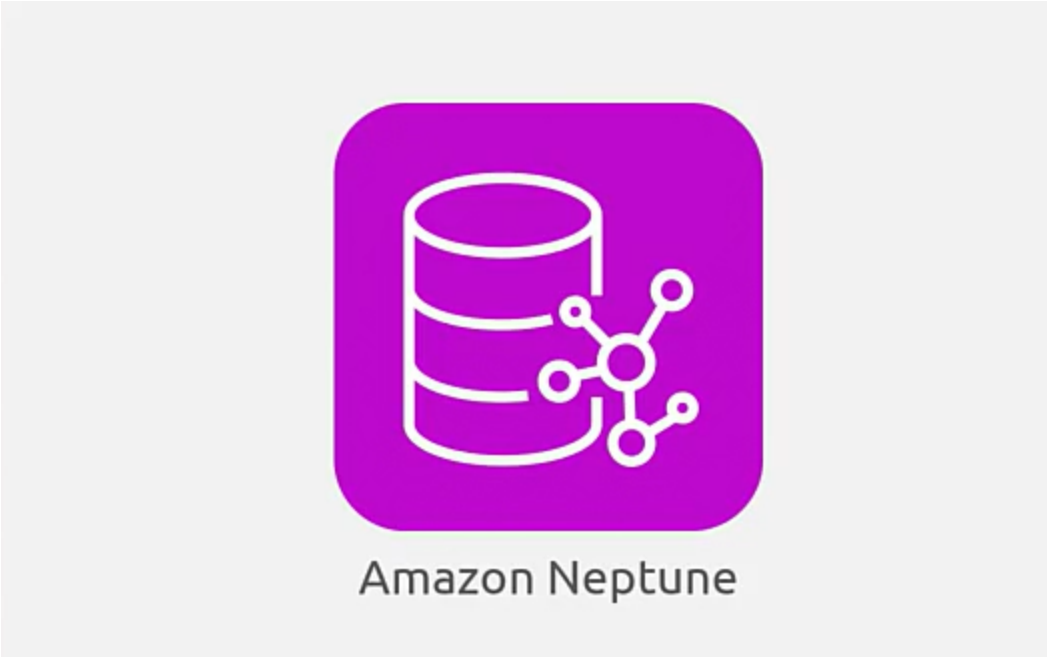


image_112.png

ELasticCache

Sometimes you need a database that can cache expensive database results and be able to get it faster. Here you need to store data in a location that is faster than my regular database. We have two databases:

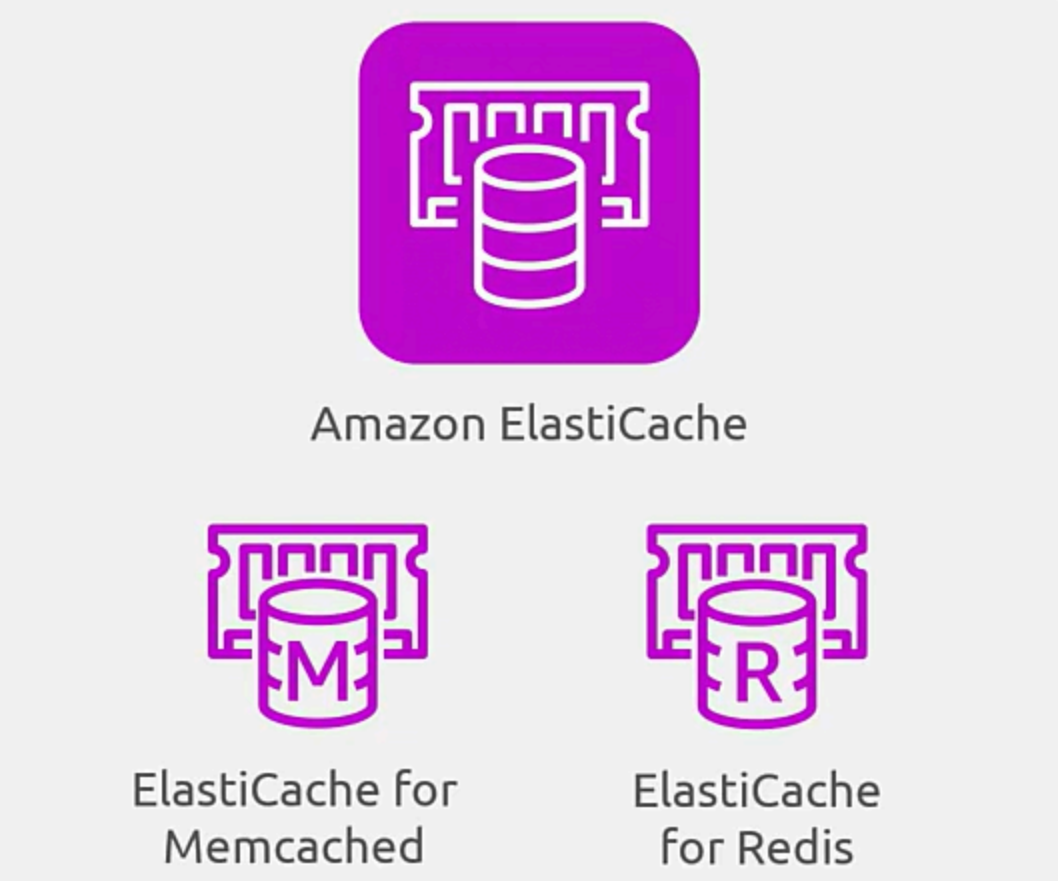


image_113.png

Here you will have your app listening to the cache but if something is not in the ache then we will have to go to the database but it will be a bit slow.

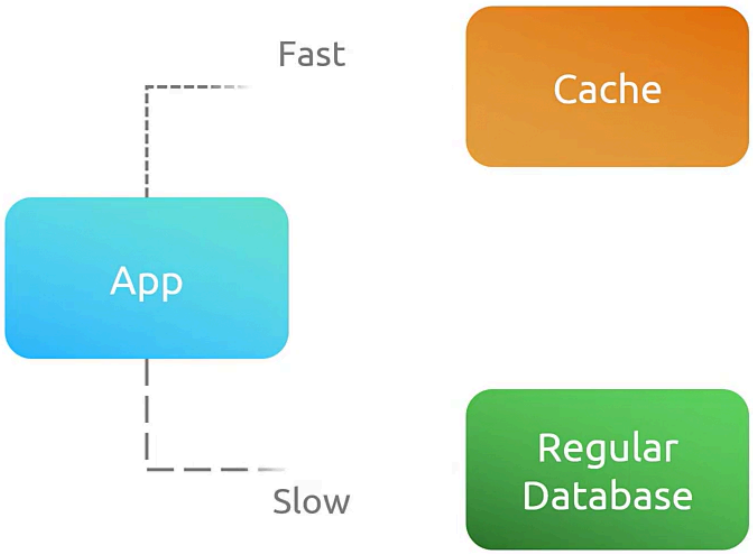


image_114.png

OpenSearch Service

When you want a database that you can use to search through a bunch of information like a google search that give you relevant or related results, here you use **Amazon OpenSearch Service**

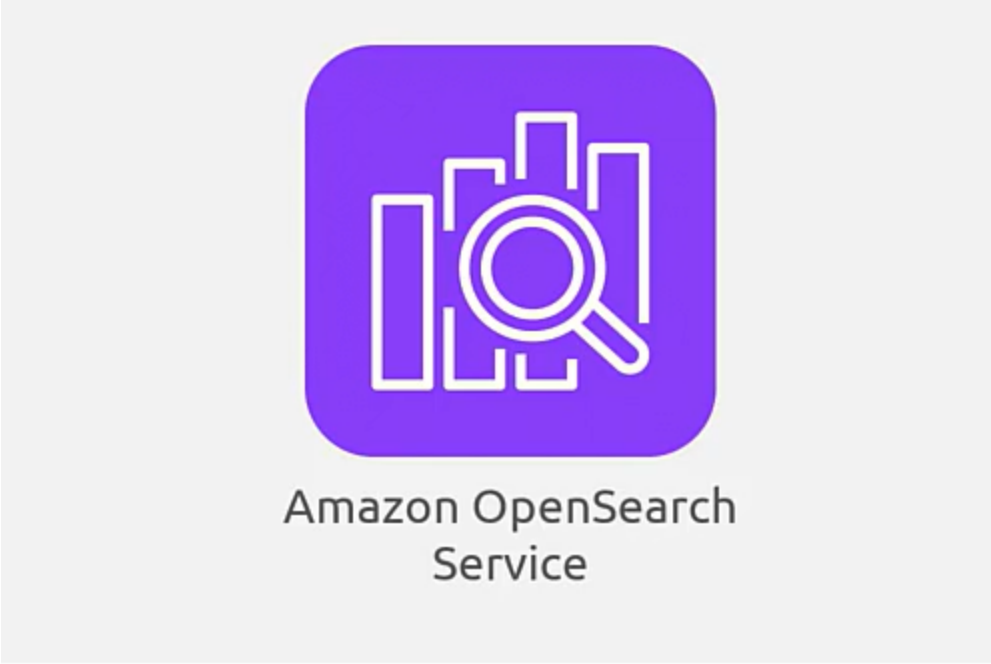


image_115.png

Amazon Quantum Ledger Database Services

A database where everytime you modify a number or a transaction it keeps a record of it (more like blockchain).It's great for security consistency. Example databases:

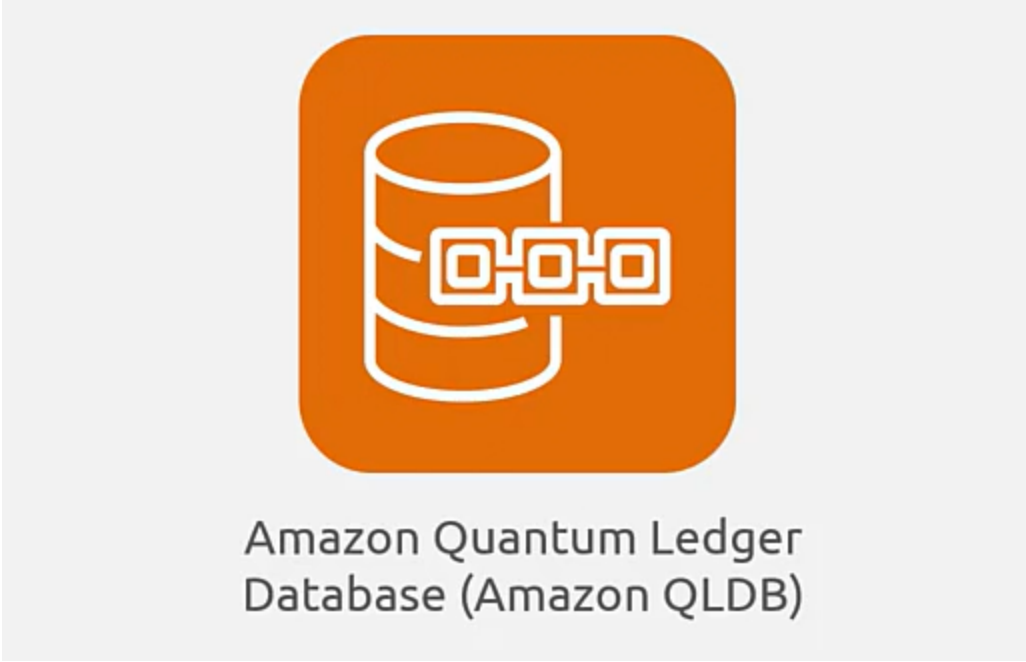


image_116.png

Timestream

This is the database you use when collecting data from IOT devices and you need a database that captures data from various sources at high scale and maintains the timestamp.

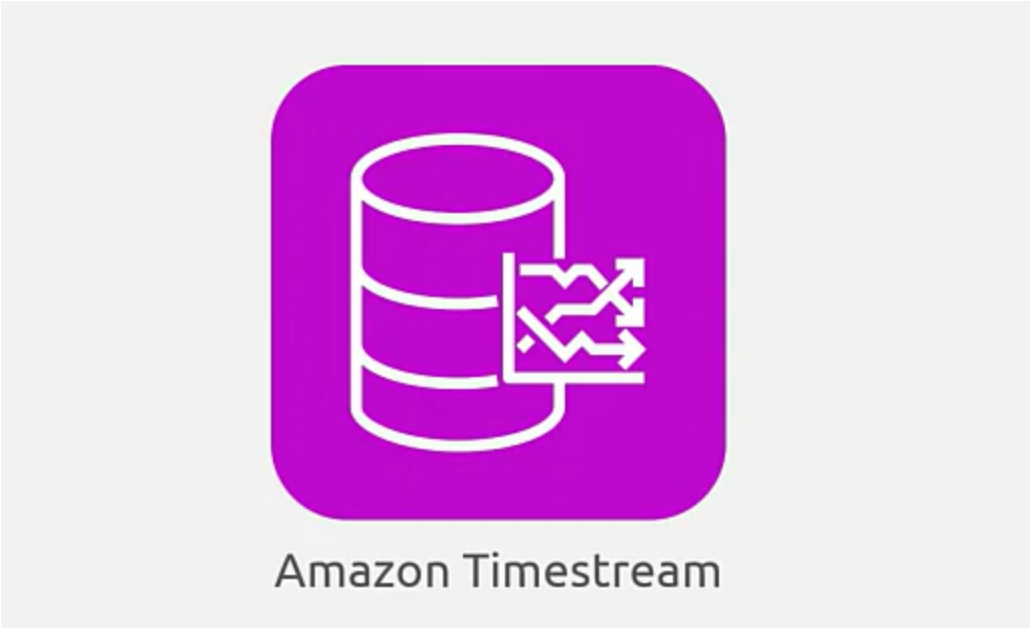


image_117.png

Good for collecting Logs from multiple devices

image_118.png

AWS Database SUMMARY

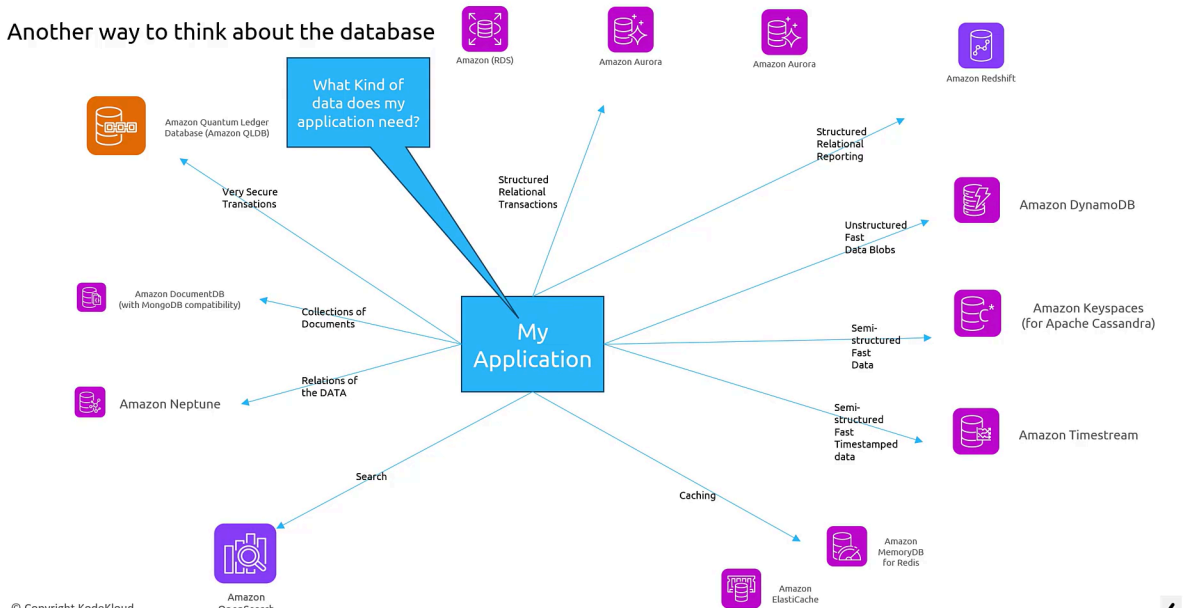


image_119.png

Another way Would be:

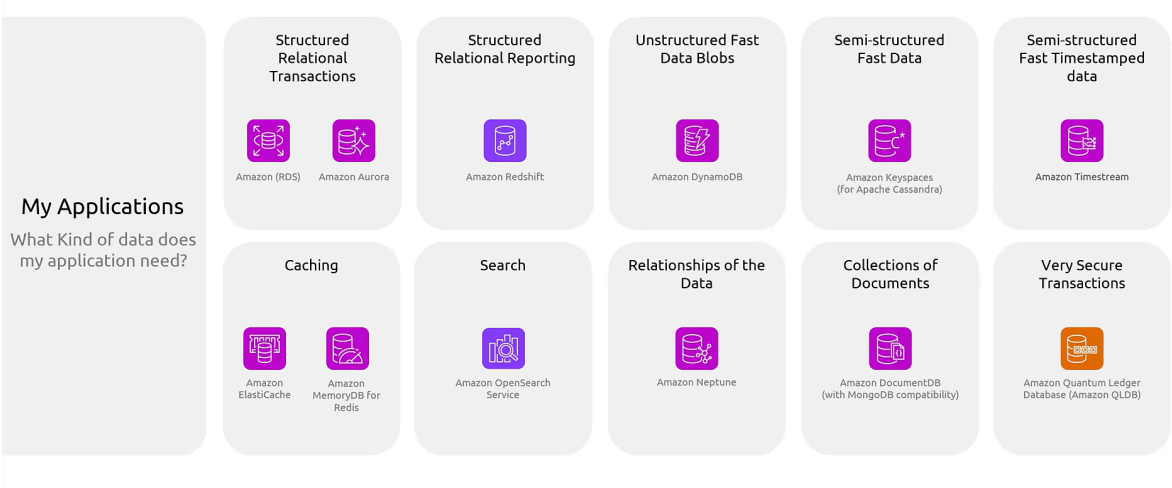


image_120.png

Summary

- Self-Managed is an option if you need management and control
- RDS and RedShift are the primary SQL database systems
- RDS has five engines is supports: Oracle, MySQL, MariaDB, MS SQL, and PostgreSQL
- DynamoDB, DocumentDB, and others are NoSQL services that are fully managed
- Make sure you look at the previous slide for use cases for each service

image_121.png

Application Integration

Application integration is something that manages or enhances the communication between application.

App to App

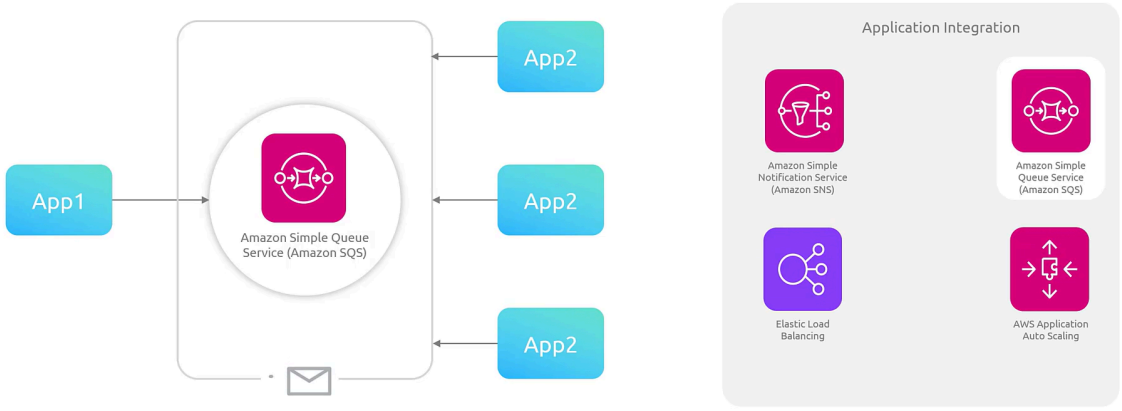


image_122.png

Internet to App

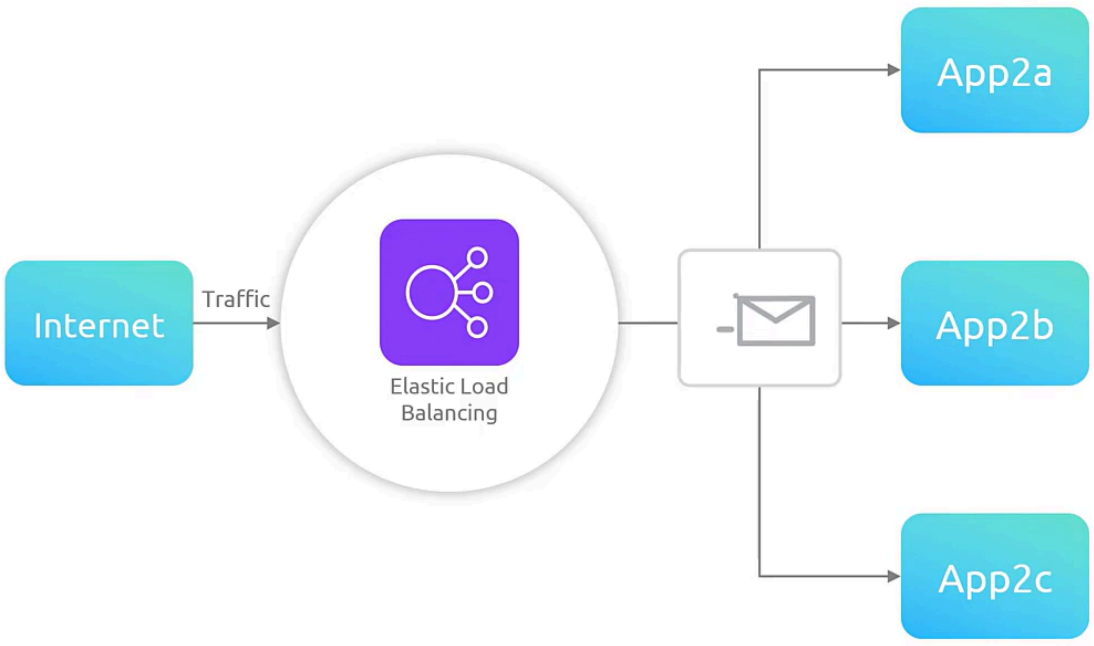


image_123.png

App to customer

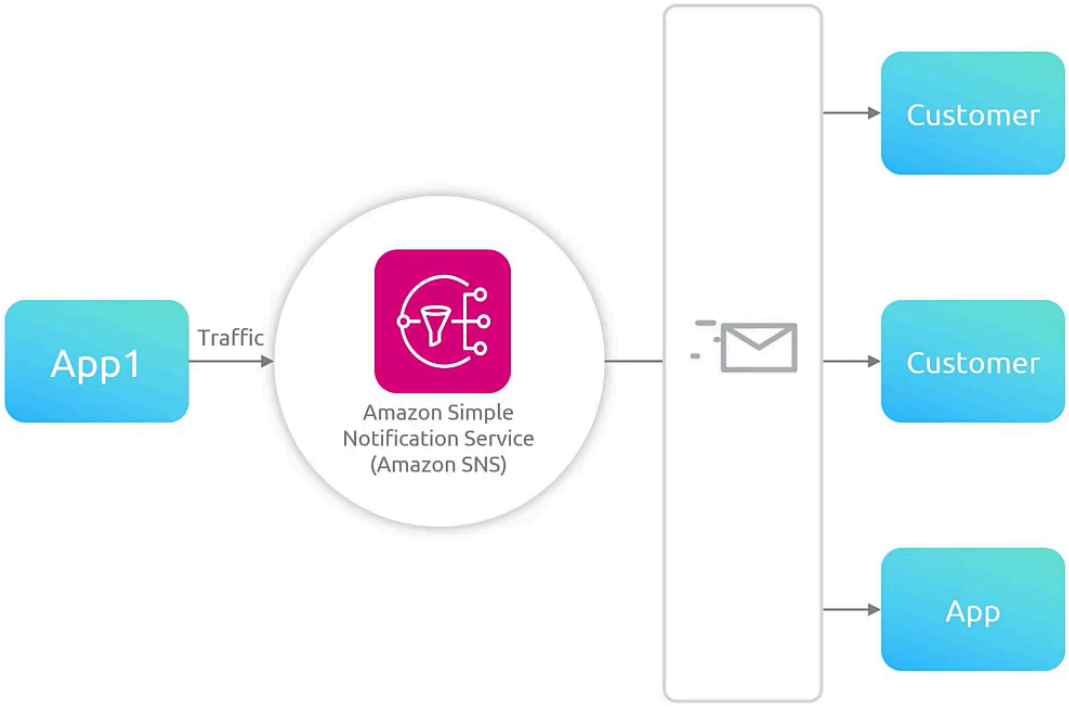


image_124.png

Application integration is managing the flow, quality, buffering, speed, rate of communication between two application

Amazon Simple Notification Service

Is a service used between two application in order to duplicate a message and send to multiple customers.

SNS is used when you want an application to send messages to customers via text, email, or mobile push or when you want to copy a single message to multiple applications.

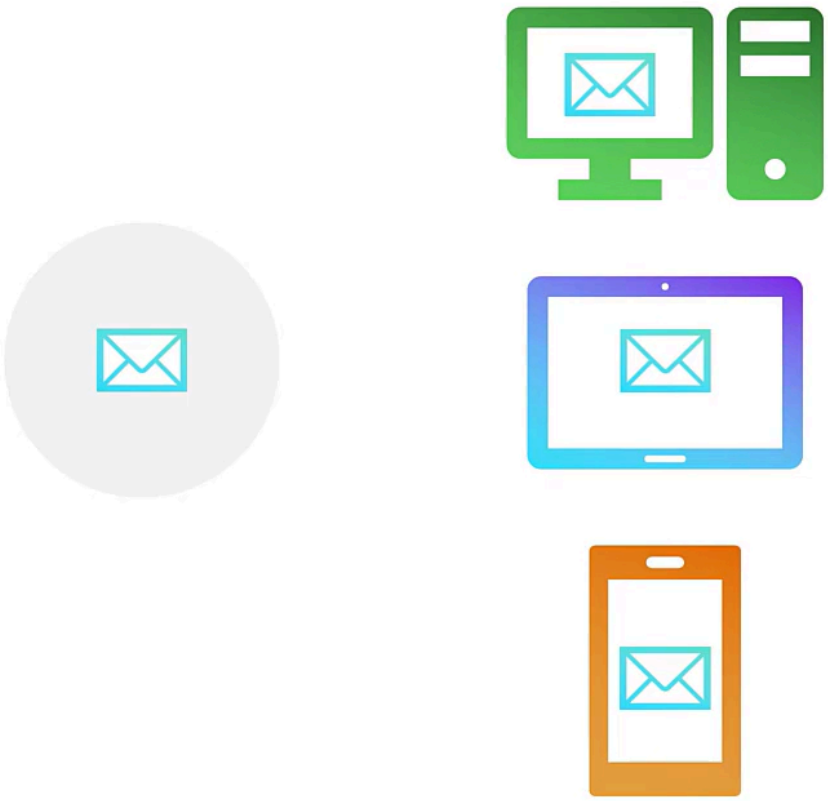


image_125.png

Amazon Simple Queue Service

SQS is used when you want to send a message to another application, but there is a chance that a sudden increase in user traffic could generate a large amount of message.

The orders will just queue until your backend can process them.

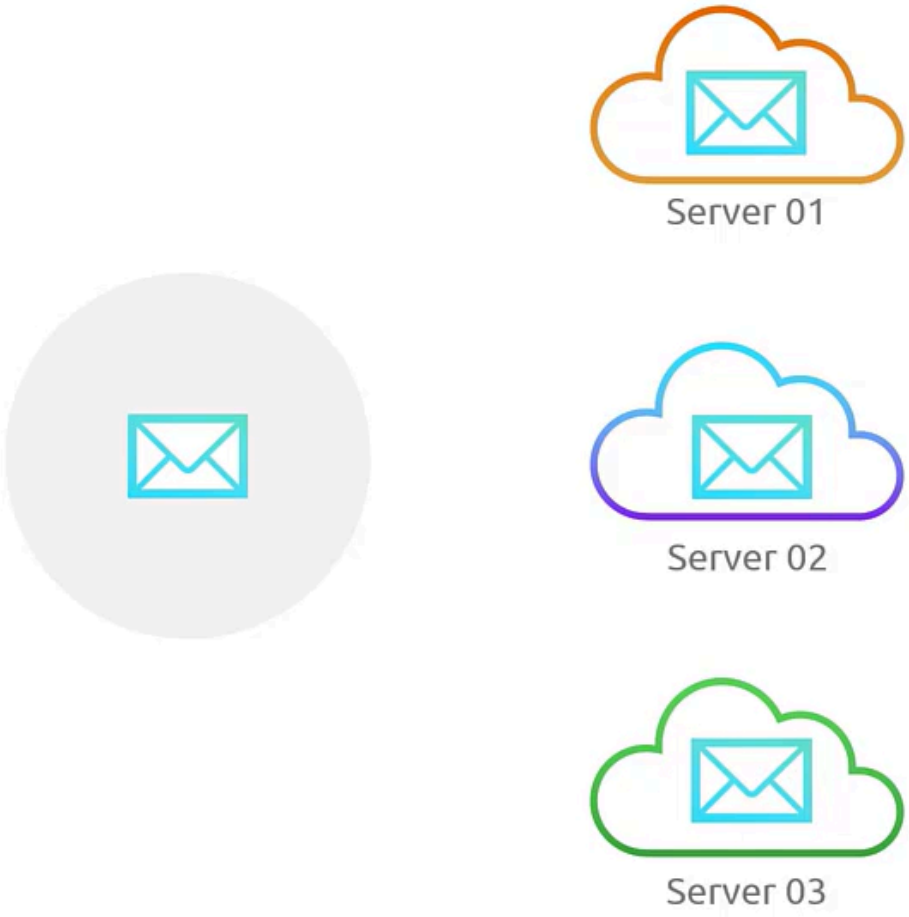


image_126.png

While SNS duplicate SQS will make sure that you won't overload your backend.

Amazon Elastic Load Balancing

Used to direct traffic to backend servers and distribute workloads evenly across servers. Unhealthy servers are not available if failing.

Can be used with EC2, ECS, EKS and lambda among others.

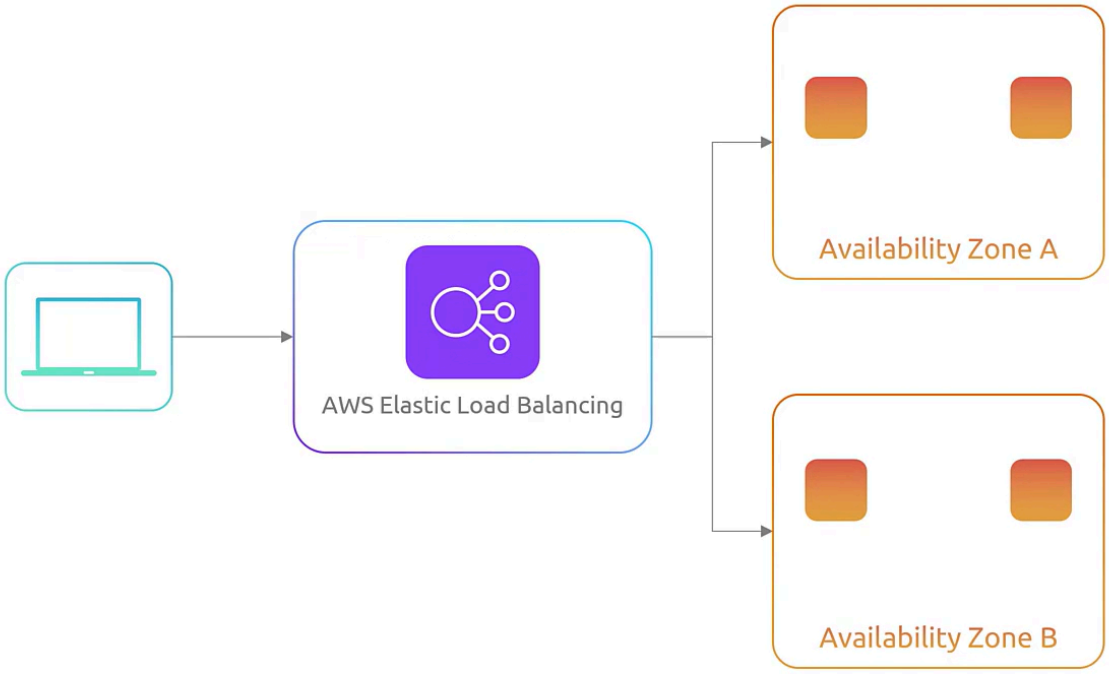


image_127.png

Amazon AutoScaling

A companion service that is used with elastic Load Balancer (ELB). It monitors traffic and checks the number of people waiting in line and if I get a certain number of people then it's going to add more servers to allow that traffic to be handled.

Many applications have autoscaling like DynamoDB and EC2.

Allows for scale up and down to numbers you specify.

Like 2 instances minimum and 4 instances maximum(shown below)

Scales as you need to within your limits.

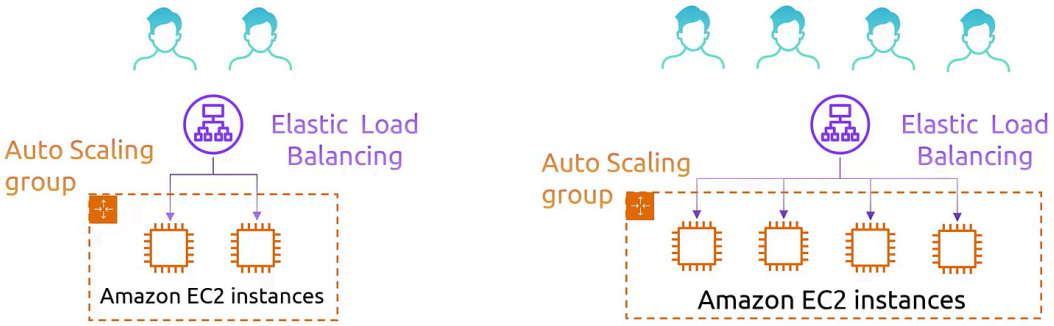


image_128.png

Other services

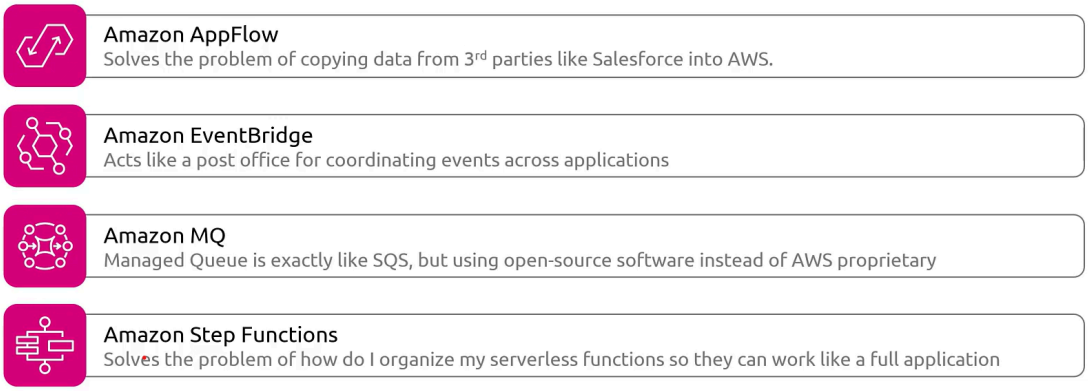


image_129.png