What is cloud computing?

●Hosting all the services virtually is called cloud computing.
Cloud computing, often referred to as simply the cloud, is of on-demand delivery of IT resources and applications via the Internet with pay-as-you-go pricing.
● With cloud computing, you don't need to make large up-front investments in hardware and spend a lot of time managing that hardware. Rather we can buy all those services virtually from cloud service providers.

Cloud computing models:-

Saas (Software as a service)
Softwares used for free or via subscription.Software as a service is the easiest way to cloud compute. The software's are accessed over the internet.
Eg. Googledrive,dropbox, etc(over internet for free).Via subscription: (MS office,Netflix,Youtube)
Limitations:License expiration or needs to be upgraded accordingly when required.

Paas (Platform as a service)
Provides environment to design and develop tools for software applications.
Eg**.** Google app engine, salesforce,Go daddy.
Limitation**:**
● Application build on one platform cannot be moved to another.Need to purchase it from other vendors.

Iaas (Infrastructure as a Service)
Hosting entire infrastructure on cloud**.**
We can buy the services from cloud service providers in order to host the entire infrastructure accordingly as per organization requirement.
Types of IAAS Cloud:-
Public cloud
● Cloud service providers use the internet to make revenue,showcasing resources being let out by them in the market and which best suits their organizations.
● Ex:AWS, IBM Cloud, Sun Cloud, Google cloud platform and Microsoft AZURE,Alibaba,Digital Ocean.

Private cloud
● Private clouds mainly is restricted at organization level only, eventually all the companies moving to cloud will have its own private cloud set up.
● Private clouds can be expensive, so most typically use by large enterprises. Private clouds are driven by concerns around security and compliance, and keeping assets within the firewall.

Hybrid cloud

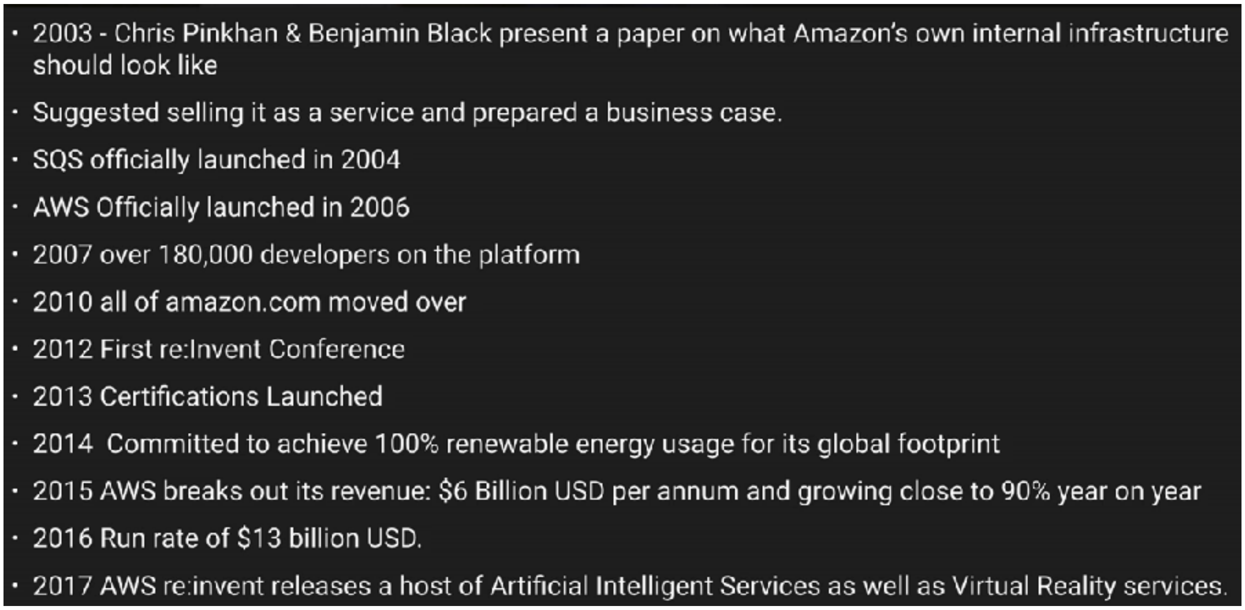
● In cloud computing, hybrid cloud refers to the use of both on-premises resources in addition to public cloud resources. A hybrid cloud enables an organization to migrate applications and data to the cloud, extend their datacenter capacity, utilize new cloud-native capabilities, move applications closer to customers, and create a backup and disaster recovery solution with cost-effective high availability.

● Hybrid cloud architecture is the integration of on-premises resources with cloud resources. For most organizations with on-premises technology investments, operating in a hybrid architecture is a necessary part of cloud adoption. Migrating legacy IT systems takes time. Therefore, selecting a cloud provider who can help you implement a thoughtful hybrid strategy, without requiring costly new investments in on-premises hardware and software, is important to simplify operations and more easily achieve your business goals.

Community cloud

● A community cloud is a multi-tenant infrastructure where a group of organizations coming together in order to share the resources for cost optimization purpose.
Ex: DELL (EMC2/VmWare) Group of organizations coming together in order to share the same services from same cloud service provider in order to reduce the cost is called as community cloud.

AWS HISTORY



- 2003 - Chris Pinkhan & Benjamin Black present a paper on what Amazon's own internal infrastructure should look like
- Suggested selling it as a service and prepared a business case.
- SQS officially launched in 2004
- AWS Officially launched in 2006
- 2007 over 180,000 developers on the platform
- 2010 all of amazon.com moved over
- 2012 First re:Invent Conference
- 2013 Certifications Launched
- 2014 Committed to achieve 100% renewable energy usage for its global footprint
- 2015 AWS breaks out its revenue: $6 Billion USD per annum and growing close to 90% year on year
- 2016 Run rate of $13 billion USD.
- 2017 AWS re:invent releases a host of Artificial Intelligent Services as well as Virtual Reality services.

Terminologies:

REGION:
AVAILABILITY ZONE:
EDGE LOCATION:

- **Region** is a distinct geographic location where amazon has its infrastructure.

AWS had its infrastructures in 16 regions and now 6 more regions are been added as well.

- All the regions are designed to be independent of each other with separate power sources, internet connectivity and geographic location
- An **availability zone** is a datacenter within a region. AZ’s are independent of each other if one goes down it does not have effect on other because they are supplied by seperate power and other resources accordingly.We had about 44 AZ’s and now 17 more are included.
- For e.g.India region has availability zone in Mumbai ap-south-1a, ap-south-1b.
- **Edge location** are CND(Content Delivery Network) end points.

These edge locations are used to cache files so that its quickly available to the nearest location users with same quality and performance as that of our original resources or contents accessed. We had about 50 edge locations and now 96 more are being included.

WHY AWS and HOW IS IT DIFFERENT FROM ITS COMPETITORS?

AWS vs Azure - The Showdown

AWS	**Azure**
On-demand cloud computing platform for Amazon	Public cloud platform for Microsoft
Friendly with the open source model from the beginning.	Not so good relationship with the open source community.
Has an edge over Azure in terms of government cloud offerings.	Limited reach when it comes to government cloud offerings.
Flexible Pricing Model	Comparatively less flexible pricing model when compared to AWS.
AWS is yet strengthening its offerings to support Hybrid clouds.	Excels in Hybrid Cloud Space-Organizations can integrate onsite servers with Cloud instances.
AWS has a software marketplace with extensive partner ecosystem -Windows and Linux	With limited Linux options, Azure is still building its partner ecosystem.
EBS storage is superfast for big data.	Standard storage has difficulties for big data and hence premium storage is required.
More mature cloud environment for big data.	Less mature for big data but Azure’s services are improving.
Machines can be accessed individually.	Machines are grouped into cloud service and respond to the same domain name but different ports.
Elastic Compute Cloud (EC2); pay by the hour.	Azure Infrastructure Services , pay by the minute.
S3 – Short-term archiving and retrieval. Long term data archiving and retrieval through Amazon Glacier.	Blobs, Queues and Tables- Similar to S3. No long term data archiving and retrieval option yet.

Security is provided through user defined roles with exceptional permission controls.	Provides security by enabling permissions on the whole account.

AWS vs Azure - Overview

AWS remains the global market share leader in public cloud services at 33% followed by Azure at 13% and Google Cloud at 6%. – Synergy Research Group Report

AWS and Azure offer largely the same basic capabilities around flexible compute, storage, networking and pricing. Both share the common elements of a public cloud – autoscaling, self-service, pay-as-u-go pricing, security, compliance, identity access management features and instant provisioning.

"With AWS a new server can be up and running in three minutes (it used to take Eli Lilly seven and a half weeks to deploy a server internally) and a 64-node Linux cluster can be online in five minutes (compared with three months internally)…The deployment time is really what impressed us."~ Dave Powers, Associate Information Consultant at Eli Lilly and Company.

With over a million customers, 2 million servers, 100,000 Weather-Forecasting Computer Cores and $10 billion in annual revenue, AWS is the largest cloud computing platform. AWS commands 40% of the cloud computing market share, more than the market share of its three biggest competitors put together. The most experienced and oldest cloud player with 11 years in operation provides extensive list of computing services and functions of mobile networking, deployments, machine learning and more. Meanwhile, growing at a rate of 120K new customers per month, 5 million organizations using Azure Active directory, 4 million developers registered with visual studio team services,1.4 million SQL databases, 2 trillion message per week processed by Azure IoT, and 40% of revenue generated from start-ups and ISVs- Azure is on the verge of dominating AWS cloud services.

AWS vs Azure – Compute

Calculate, process, and compute – that is the fundamental role of a computer. The right cloud service provider can help scale to 1000's of processing node in just couple of minutes. For organizations that need faster data analysis or graphics rendering, there are two choices available – buy additional hardware or shift to the cloud. This is what is the goal of public cloud services.

For compute, AWS' primary solution is its EC2 instances which provide scalable computing on-demand and can be customized for different options' also provides other related services like the EC2 container service, AWS Lambda, Autoscaling, and Elastic Beanstalk for app deployment. Azure's compute offerings are based on VMs with multiple other tools such as Cloud Services and Resource Manager which help deploy applications on the cloud.

AWS still offers the largest range of services, close to 100 across compute, storage, database, analytics, networking, mobile, developer tools, management tools, IoT, security and enterprise applications.

AWS vs Azure - Compute

Service	*AWS*	*Azure*
Deploy, Manage, and Maintain Virtual Servers	EC2 (Elastic Compute Cloud)	Virtual Machines and Virtual Machine Scale Sets
Docker Container Registry	ECR (EC2 Container Registry)	Container Registry
Scale Instances Automatically	Auto Scaling	Virtual Machine Scale Sets

		Auto Scaling App Service Scale Capability (PAAS)
Platform-as-a-service	Elastic Beanstalk	Cloud Services
Integrating systems and running backend logic processes	AWS Lambda	Event Grid Web Jobs Functions

AWS vs Azure – Storage

A key functionality of cloud service providers is their storage capability. Running services in the cloud involve data processing that needs to be saved at some point of time. AWS’ storage services are longest running , however, Azure’s storage capabilities are also extremely reliable. Both Azure and AWS are strong in this category and include all the basic features such as REST API access and server-side data encryption. Azure’s storage mechanism is referred to as Blob storage , and AWS’s is called Simple Storage Service (S3).

AWS’s cloud object storage solution offers high availability and automatic replication across regions. Temporary storage in AWS starts functioning when an instance starts and stops when an instance terminates also provides block storage that is similar to hard disks and can be attached to any EC2 instance or kept separate. Azure uses temporary storage and page blobs for VM based volumes. Azure’s Block Storage option is similar to S3 in AWS. There are two classes of storage offered by Azure -Hot and Cool. Cool storage is comparatively less pricey than Hot but one has to incur additional read and write costs.

Azure vs AWS - Object Storage

Service	***AWS***	***Azure***
Service Name	S3	Azure Storage-Blobs
Hot	S3 Standard	Hot Blob Storage
Cool	S3 Standard -Infrequent Access	Cool Blob Storage
Cold	Amazon Glacier	Archive Blob Storage
Object Size Limits	5 TB	4.75 TB
# of Object Limits	Unlimited	Unlimited

Azure vs AWS– Block/Disk Storage

Services	***AWS***	***Azure***
Service Name	EBS	Managed Disks
Volume Types	Cold HDD General Purpose SSD PIOPs SSD Throughput Optimized HDD	Standard Premium SSD
Availability SLA	99.9%	99.9%
IOPs/GB for SSD	GP SSD -3 PIOPS SSD up to 50/GB.	1.8 to 4.9 – This is fixed based on the disk type.

AWS vs Azure Pricing

Cost is a major factor of attraction for organizations planning to move to the cloud. With increasing competition amongst cloud service providers, there has been a continued downward trend on prices since quite some time now. AWS and Azure offer free introductory tiers with restricted usage limits that let users try and use their services before they can buy. Also, both offer credits to grab the attention of start-ups onto their cloud platforms.

AWS provides pay-as-you-go model and charges per hour while Azure's pricing model is also pay-as-you-go , they charge per minute. AWS can help you save more with increased usage- the more you use, the less you pay. AWS instances can be purchased based on one of the following models –

- Reserved Instances – Paying an upfront cost based on the use, one can reserve an instance for 1 to 3 years.
- On-demand Instances -Just pay for what you use without paying any upfront cost.
- Spot Instances- Bid for extra capacity based on the availability.

Azure offers short term commitments to its users allowing them to choose between pre-paid or monthly charges. Azure is a little less flexible than AWS when it comes to pricing model.

AWS vs Azure – Databases

All software applications today require a database to save information. Azure and AWS both provide database services, regardless of whether you need a relational database or a NoSQL offering. Amazon's RDS (Relational Database Service) and Microsoft's equivalent SQL Server database both are highly available and durable and also provide automatic replication.

AWS works perfectly with NoSQL and relational databases providing a mature cloud environment for big data. AWS' core analytics offering EMR (a managed Hadoop, Spark and Presto solution) helps set up an EC2 cluster and provides integration with various AWS services.

Azure also supports both NoSQL and relational databases and as well Big Data through Azure HDInsight and Azure table. Azure provides analytical products through its exclusive Cortana Intelligence Suite that comes with Hadoop, Spark, Storm, and HBase.

Amazon's RDS supports six popular database engines – MariaDB, Amazon Aurora, MySQL, Microsoft SQL, PostgreSQL, and Oracle while Azure's SQL database service is solely based on MS SQL Server. Azure's interface and tooling makes it easy to perform various DB operations while AWS has more instance types which you can provision and get that additional control over DB instances.

AWS vs Azure – Content Delivery and Networking

Every cloud service provider offers multiple networks and partners that interconnect the data centres across the globe through diverse products. AWS provides Virtual Private Cloud (VPC) for users to create isolated networks within the cloud. A user can create route tables, private IP address ranges, subnets, and network gateways within a VPC. Similarly, Azure offers Virtual Network (VNET) for users to create isolated networks. Both AWS and Azure provide firewall option and solutions to extend on-premise data centre into the cloud.

AWS vs Azure - Content Delivery and Networking

Service Name	AWS	Azure
Isolated private cloud	Virtual Private Cloud (VPC)	Virtual Network (VNET)

Global Content Delivery Networks	CloudFront	Content Delivery Network (CDN)
Manage DNS Names and records	Route 53	Traffic Manager Azure DNS
Dedicated Private Network Connection	DirectConnect	ExpressRoute

Amazon Elastic Compute Cloud (Amazon EC2)

- Amazon Elastic Compute Cloud (Amazon EC2) provides resizable computing capacity in the cloud.
- It reduces the time take to boot new server instances to minutes allowing to quickly scale up and down as your computing requirement changes.

Basically it’s a virtual computing environment,where you can login and perform actions as required.

- They are called as servers but in AWS language they are termed as INSTANCES.
- **Free tier account 20 EC’s provided and charged on hourly basis (free for about 750 hrs).**

EC2 States/Status:

START
STOP
STOP-HIBERNATE
REBOOT
TERMINATE

EC2 Purchase Options

- **On-Demand instances**

Instantly created based on requirement. Pay for the instances that you use by the hour(by second charged nowadays),hence charged on hourly basis.

- **Reserved Instances**

Provides you with a capacity reservation and offers a significant discount as they are bought in bulk making upfront payment on a contract basis.(for a one- or three-year term.

- **Spot instances**

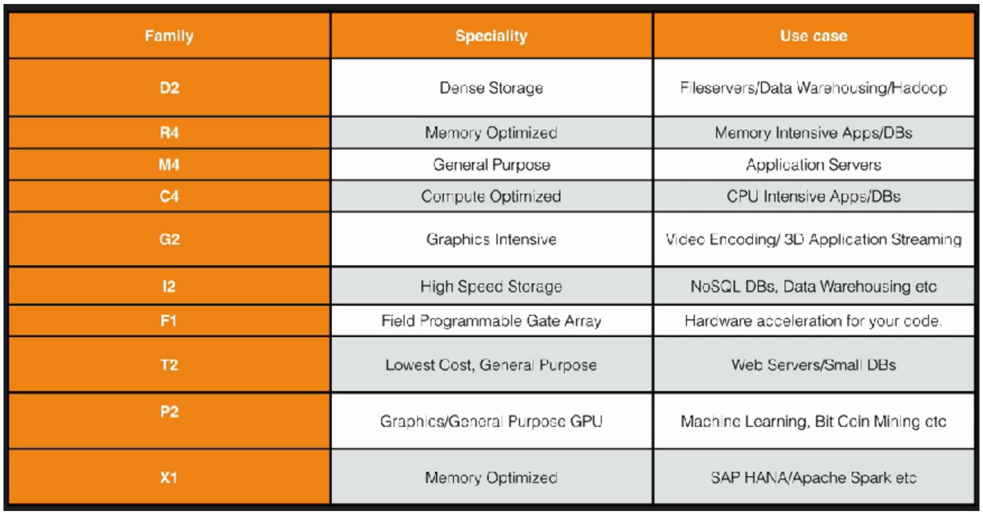
Enables you to Bid for the prices you want for instance capacity providing for greater savings if your applications have flexible start and end time on it.

- **Dedicated hosts**

Physical EC2 server dedicated for your use.They can help reduce costs by allowing you to use existing server bound licenses.

EC2 instance types:

D-Density
I-IOPS(input/output operations per second)
R-RAM
T-Cheap general purpose
M-Main choice for general purpose apps
C-Compute
G-Graphics



Family	Speciality	Use case
D2	Dense Storage	Fileservers/Data Warehousing/Hadoop
R4	Memory Optimized	Memory Intensive Apps/DBs
M4	General Purpose	Application Servers
C4	Compute Optimized	CPU Intensive Apps/DBs
G2	Graphics Intensive	Video Encoding/ 3D Application Streaming
I2	High Speed Storage	NoSQL DBs, Data Warehousing etc
F1	Field Programmable Gate Array	Hardware acceleration for your code.
T2	Lowest Cost, General Purpose	Web Servers/Small DBs
P2	Graphics/General Purpose GPU	Machine Learning, Bit Coin Mining etc
X1	Memory Optimized	SAP HANA/Apache Spark etc

EC2 Terminologies:

AMI(Amazon Machine Image): A template for the root volume for the instance (for example, an operating system, an application server, and applications)

SNAPSHOTS: You can back up the data on your Amazon EBS volumes to Amazon S3 by taking point-in-time snapshots. Snapshots are incremental backups, which means that only the blocks on the device that have changed after your most recent snapshot are saved.

VOLUMES: An Amazon EBS volume is a durable, block-level storage device that you can attach to a single EC2 instance.Default EBS volumes gets created while creating EC2.For Linux its:8GB and for Windows its :30GB.

SECURITY GROUPS: A security group acts as a virtual firewall that controls the traffic for one or more instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group.

KEYPAIRS: Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. Public–key cryptography uses a public key to encrypt a piece of data, and then the recipient uses the private key to decrypt the data. The public and private keys are known as a *key pair*. Public-key cryptography enables you to securely access your instances using a private key instead of a password.

EBS (Elastic Block Storage)

● An Amazon EBS volume is a durable, block-level storage device that you can attach to a single EC2 instance. You can use EBS volumes as primary storage for data that requires frequent updates, such as the system drive for an instance or storage for a database application. You can also use them for throughput-intensive applications that perform continuous disk scans. EBS volumes persist independently from the running life of an EC2 instance.

After a volume is attached to an instance, you can use it like any other physical hard drive. EBS volumes are flexible. For current-generation volumes attached to current-generation instance types, you can dynamically increase size, modify the provisioned IOPS capacity, and change volume type on live production volumes.

Amazon EBS provides the following volume types: General Purpose SSD (gp2), Provisioned IOPS SSD (io1), Throughput Optimized HDD (st1), Cold HDD (sc1), and Magnetic (standard, a previous-generation type).

● **General Purpose SSD (gp2) Volumes**

General Purpose SSD (gp2) volumes offer cost-effective storage that is ideal for a broad range of workloads. These volumes deliver single-digit millisecond latencies and the ability to burst to 3,000 IOPS for extended periods of time. Between a minimum of 100 IOPS (at 33.33 GiB and below) and a maximum of 16,000 IOPS (at 5,334 GiB and above), baseline performance scales linearly at 3 IOPS per GiB of volume size. AWS designs gp2 volumes to deliver 90% of the provisioned performance 99% of the time. A gp2 volume can range in size from 1 GiB to 16 TiB.

● **Provisioned IOPS SSD**

Provisioned IOPS SSD (`io1`) volumes are designed to meet the needs of I/O-intensive workloads, particularly database workloads, that are sensitive to storage performance and consistency. Unlike `gp2`, which uses a bucket and credit model to calculate performance, an `io1` volume allows you to specify a consistent IOPS rate when you create the volume, and Amazon EBS delivers within 10 percent of the provisioned IOPS performance 99.9 percent of the time over a given year.

● **Throughput Optimized HDD** (st1)

Throughput Optimized HDD (st1) volumes provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. This volume type is a good fit for large, sequential workloads such as Amazon EMR, ETL, data warehouses, and log processing. Bootable st1 volumes are not supported. Throughput Optimized HDD (st1) volumes, though similar to Cold HDD (sc1) volumes, are designed to support frequently accessed data. This volume type is optimized for workloads involving large, sequential I/O, and we recommend that customers with workloads performing small, random I/O use gp2.

● **Cold HDD (sc1**)

Cold HDD (sc1) volumes provide low-cost magnetic storage that defines performance in terms

of throughput rather than IOPS. With a lower throughput limit than st1, sc1 is a good fit ideal for large, sequential cold-data workloads. If you require infrequent access to your data and are looking to save costs, sc1 provides inexpensive block storage. Bootable sc1 volumes are not supported.
Cold HDD (sc1) volumes, though similar to Throughput Optimized HDD (st1) volumes, are designed to support infrequently accessed data.

LOAD BALANCER

● Elastic Load Balancing distributes incoming application or network traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses, in multiple Availability Zones. Elastic Load Balancing scales your load balancer as traffic to your application changes over time, and can scale to the vast majority of workloads automatically.

● Load Balancer Benefits

A load balancer distributes workloads across multiple compute resources, such as virtual servers. Using a load balancer increases the availability and fault tolerance of your applications. You can add and remove compute resources from your load balancer as your needs change, without disrupting the overall flow of requests to your applications.
You can configure health checks, which are used to monitor the health of the compute resources so that the load balancer can send requests only to the healthy ones

Features of Elastic Load Balancing

Elastic Load Balancing supports three types of load balancers:
1.Classic Load Balancer.
2.Application Load Balancer.
3.Network Load Balancer.

 Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level. Classic Load Balancer is intended for applications that were built within the EC2-Classic network. We recommend Application Load Balancer for Layer 7 and Network Load Balancer for Layer 4 when using Virtual Private Cloud (VPC).
Key Features

High Availability

You can distribute incoming traffic across your Amazon EC2 instances in a single Availability Zone or multiple Availability Zones. Classic Load Balancer automatically scales its request handling capacity in response to incoming application traffic.

Health Checks

Classic Load Balancer can detect the health of Amazon EC2 instances. When it detects unhealthy EC2 instances, it no longer routes traffic to those instances and spreads the load across the remaining healthy instances.

Security Features

When using Amazon Virtual Private Cloud (Amazon VPC), you can create and manage security groups associated with Classic Load Balancer to provide additional networking and security options. You can also create a Classic Load Balancer without public IP addresses to serve as an internal (non-internet-facing) load balancer.

SSL Offloading

Classic Load Balancer supports SSL termination, including offloading SSL decryption from application instances, centralized management of SSL certificates, and encryption to back-end instances with optional public key authentication. Flexible cipher support allows you to control the ciphers and protocols the load balancer presents to clients.

Sticky Sessions

Classic Load Balancer supports the ability to stick user sessions to specific Amazon EC2 instances using cookies. Traffic will be routed to the same instances as the user continues to access your application.

IPv6 Support

Classic Load Balancer supports the use of both the Internet Protocol version 4 and 6 (IPv4 and IPv6) for EC2-Classic networks.

Layer 4 or Layer 7 Load Balancing

You can load balance HTTP/HTTPS applications and use Layer 7-specific features, such as X-Forwarded and sticky sessions. You can also use strict Layer 4 load balancing for applications that rely purely on the TCP protocol.

Operational Monitoring

Classic Load Balancer metrics such as request count and request latency are reported by Amazon CloudWatch.

Logging

Use the Access Logs feature to record all requests sent to your load balancer, and store the logs in Amazon S3 for later analysis. The logs are useful for diagnosing application failures and analyzing web traffic. You can use AWS CloudTrail to record Classic Load Balancer API calls for your account and deliver log files. The API call history enables you to perform security analysis, resource change tracking, and compliance auditing.

Application Load Balancer operates at the request level (layer 7), routing traffic to targets – EC2 instances, containers, IP addresses and Lambda functions based on the content of the request. Ideal for advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications. Application Load Balancer simplifies and improves the security of your application, by ensuring that the latest SSL/TLS ciphers and protocols are used at all times.

Key Features

Layer-7 Load Balancing

You can load balance HTTP/HTTPS applications and use layer 7-specific features, such as X-Forwarded-For headers.

HTTPS Support

An Application Load Balancer supports HTTPS termination between the clients and the load balancer. Application Load Balancers also offer management of SSL certificates through AWS Identity and Access Management (IAM) and AWS Certificate Manager for pre-defined security policies.

Server Name Indication (SNI)

Server Name Indication (SNI) is an extension to the TLS protocol by which a client indicates the hostname to connect to at the start of the TLS handshake. The load balancer can present multiple certificates through the same secure listener, which enables it to support multiple secure websites using a single secure listener. Application Load Balancers also support a smart certificate selection algorithm with SNI. If the hostname indicated by a client matches multiple certificates, the load balancer determines the best certificate to use based on multiple factors including the capabilities of the client.

IP addresses as Targets

You can load balance any application hosted in AWS or on-premises using IP addresses of the application backends as targets. This allows load balancing to an application backend hosted on any IP address and any interface on an instance. Each application hosted on the same instance can have an associated security group and use the same port. You can also use IP addresses as targets to load balance applications hosted in on-premises locations (over a Direct Connect or VPN connection), peered VPCs and EC2-Classic (using ClassicLink). The ability to load balance across AWS and on-prem resources helps you migrate-to-cloud, burst-to-cloud or failover-to-cloud.

Lambda functions as Targets

Application Load Balancers support invoking Lambda functions to serve HTTP(S) requests enabling users to access serverless applications from any HTTP client, including web browsers. You can register Lambda functions as targets for a load balancer and leverage the support for content-based routing rules to route requests to different Lambda functions. You can use an Application Load Balancer as a common HTTP endpoint for applications that use servers and serverless computing. You can build an entire website using Lambda functions or combine EC2 instances, containers, on-premises servers and Lambda functions to build applications.

High Availability

An Application Load Balancer requires you to specify more than one Availability Zone. You can distribute incoming traffic across your targets in multiple Availability Zones. An Application Load Balancer automatically scales its request handling capacity in response to incoming application traffic.

Security Features

When using Amazon Virtual Private Cloud (VPC), you can create and manage security groups associated with Elastic Load Balancing to provide additional networking and security options. You can configure an Application Load Balancer to be Internet facing or create a load balancer without public IP addresses to serve as an internal (non-internet-facing) load balancer.

Content-based Routing

If your application is composed of several individual services, an Application Load Balancer can route a request to a service based on the content of the request.

Host-based Routing

You can route a client request based on the Host field of the HTTP header allowing you to route to multiple domains from the same load balancer.

Path-based Routing

You can route a client request based on the URL path of the HTTP header.

HTTP header-based routing

You can route a client request based on the value of any standard or custom HTTP header.

HTTP method-based routing

You can route a client request based on any standard or custom HTTP method.

Query string parameter-based routing

You can route a client request based on query string or query parameters.

Source IP address CIDR-based routing

You can route a client request based on source IP address CIDR from where the request originates.

Containerized Application Support

Application Load Balancer provides enhanced container support by load balancing across multiple ports on a single Amazon EC2 instance. Deep integration with the Amazon EC2 Container Service (ECS), provides a fully-managed container offering. ECS allows you to specify a dynamic port in the ECS task definition, giving the container an unused port when it is scheduled on the EC2 instance. The ECS scheduler automatically adds the task to the load balancer using this port.

HTTP/2 Support

HTTP/2 is a new version of the HyperText Transfer Protocol (HTTP) that uses a single, multiplexed connection to allow multiple requests to be sent on the same connection. It also compresses header data before sending it out in binary format and supports SSL connections to clients.

WebSockets Support

WebSockets allows a server to exchange real-time messages with end-users without the end users having to request (or poll) the server for an update. The WebSockets protocol provides bi-directional communication channels between a client and a server over a long-running TCP connection.

Native IPv6 Support

Application Load Balancers support native Internet Protocol version 6 (IPv6) in aVPC. This will allow clients to connect to the Application Load Balancer via IPv4 or IPv6.

Sticky Sessions

Sticky sessions are a mechanism to route requests from the same client to the same target. Application Load Balancer supports sticky sessions using load balancer generated cookies. If you enable sticky sessions, the same target receives the request and can use the cookie to recover the session context. Stickiness is defined at a target group level.

Health Checks

An Application Load Balancer routes traffic only to healthy targets. With an Application Load Balancer, you get improved insight into the health of your applications in two ways: (1) health check improvements that allow you to configure detailed error codes from 200-499. The health checks allow you to monitor the health of each of your services behind the load balancer; and (2) new metrics that give insight into traffic for each of the services running on an EC2 instance.

Operational Monitoring

Amazon CloudWatch reports Application Load Balancer metrics such as request counts, error counts, error types, and request latency.

Logging

You can use the Access Logs feature to record all requests sent to your load balancer, and store the logs in Amazon S3 for later analysis. The logs are compressed and have a gzip file extension. The compressed logs save both storage space and transfer bandwidth and are useful for diagnosing application failures and analyzing web traffic.

You can also use AWS CloudTrail to record Application Load Balancer API calls for your account and deliver log files. The API call history enables you to perform security analysis, resource change tracking, and compliance auditing.

Delete Protection

You can enable deletion protection on an Application Load Balancer to prevent it from being accidentally deleted.

Network Load Balancer operates at the connection level (Layer 4), routing connections to targets - Amazon EC2 instances, microservices, and containers – within Amazon Virtual Private Cloud (Amazon VPC) based on IP protocol data. Ideal for load balancing of TCP traffic, Network Load Balancer is capable of handling millions of requests per second while maintaining ultra-low latencies. Network Load Balancer is optimized to handle sudden and volatile traffic patterns while using a single static IP address per Availability Zone. It is integrated with other popular AWS services such as Auto Scaling, Amazon EC2 Container Service (ECS), Amazon CloudFormation, and AWS Certificate Manager (ACM).

Key Features

Connection-based Load Balancing

You can load balance TCP traffic, routing connections to targets - Amazon EC2 instances, microservices, and containers.

High Availability

Network Load Balancer is highly available. It accepts incoming traffic from clients and distributes this traffic across the targets within the same Availability Zone. The load balancer also monitors the health of its registered targets and ensures that it routes traffic only to healthy targets. When the load balancer detects an unhealthy target, it stops routing traffic to that target and reroutes traffic to remaining healthy targets. If all of your targets in one Availability Zone are unhealthy, and you have set up targets in another Availability Zone, Network Load Balancer will automatically fail-over to route traffic to your healthy targets in the other Availability Zones.

High Throughput

Network Load Balancer is designed to handle traffic as it grows and can load balance millions of requests/sec. It can also handle sudden volatile traffic patterns.

Low Latency

Network Load Balancer offers extremely low latencies for latency-sensitive applications.

Preserve source IP address

Network Load Balancer preserves the client side source IP allowing the back-end to see the IP address of the client. This can then be used by applications for further processing.

Static IP support

Network Load Balancer automatically provides a static IP per Availability Zone (subnet) that can be used by applications as the front-end IP of the load balancer.

Elastic IP support

Network Load Balancer also allows you the option to assign an Elastic IP per Availability Zone (subnet) thereby providing your own fixed IP.

TLS Offloading

Network Load Balancer supports TLS termination between the clients and the load balancer. Network Load Balancer also offers management of SSL certificates through AWS Identity and Access Management (IAM) and AWS Certificate Manager in addition to pre-defined security policies that provides flexibility around the ciphers and protocols are preferred when completing a TLS handshake between the client and the load balancer. Source IP continues to be preserved to your back-end applications when TLS is terminated on the Network Load Balancer.

Health Checks

Network Load Balancer supports both network and application target health checks. Network-level health is based on the overall response of your target to normal traffic. If the target becomes unable, or too slow, to respond to new connections then the load balancer will mark the target as unavailable. Application-level health checks can also be used to go deeper. By periodically probing a specific URL on a given target, it can integrate the health of the actual application. For quick diagnosis and powerful debugging, full visibility into health checks and why they may be failing is also available through ‘reason codes’ in the Network Load Balancer API, and the Amazon CloudWatch metrics attached to target health checks.

DNS Fail-over

If there are no healthy targets registered with the Network Load Balancer or if the Network Load Balancer nodes in a given zone are unhealthy, then Amazon Route 53 will direct traffic to load balancer nodes in other Availability Zones.

Integration with Amazon Route 53

In the event that your Network Load Balancer is unresponsive, integration with Route 53 will remove the unavailable load balancer IP address from service and direct traffic to an alternate Network Load Balancer in another region.

Integration with AWS Services

Network Load Balancer is integrated with other AWS services such as Auto Scaling, Elastic Container Service (ECS), CloudFormation, Elastic BeanStalk, CloudWatch, Config, CloudTrail, CodeDeploy, and AWS Certificate Manager (ACM).

Long-lived TCP Connections

Network Load Balancer supports long-lived TCP connections that are ideal for WebSocket type of applications.

Central API Support

Network Load Balancer uses the same API as Application Load Balancer. This will enable you to work with target groups, health checks, and load balance across multiple ports on the same Amazon EC2 instance to support containerized applications.

LOAD BALANCER RESTRICTIONS:

- Max 20 load balancer per region.
- Max 5 SG for load balancer.

AUTO SCALING

Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application.

● You create collections of EC2 instances, called *Auto Scaling groups*
● You can specify the minimum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes below this size.
● Before creating an Auto Scaling group, you must create a launch configuration.
● We can have maximum 20 auto scaling groups per region.
●EX:Grocerries bought from super market being shared in classroom training.

CloudWatch

Amazon CloudWatch monitors your Amazon Web Services (AWS) resources and the applications you run on AWS in real time. You can use CloudWatch to collect and track metrics, which are variables you can measure for your resources and applications.
The CloudWatch home page automatically displays metrics about every AWS service you use. You can create alarms which watch metrics and send notifications or automatically make changes to the resources you are monitoring when a threshold is breached
With CloudWatch, you gain system-wide visibility into resource utilization, application performance, and operational health.
The following services are used along with Amazon CloudWatch:
Amazon Simple Notification Service (Amazon SNS) coordinates and manages the delivery or sending of messages to subscribing endpoints or clients. You use Amazon SNS with CloudWatch to send messages when an alarm threshold has been reached.
Amazon EC2 Auto Scaling enables you to automatically launch or terminate Amazon EC2 instances based on user-defined policies, health status checks, and schedules.
AWS CloudTrail enables you to monitor the calls made to the Amazon CloudWatch API for your account, including calls made by the AWS Management Console, AWS CLI, and other services. When CloudTrail logging is turned on, CloudWatch writes log files to the Amazon S3 bucket that you specified when you configured CloudTrail.
AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources for your users
● Two types of monitoring
1. Basic Monitoring – Monitors every 5 minutes.
2. Detailed Monitoring - Monitors every 1 minute.
Amazon CloudWatch Concepts
Namespaces- A namespace is a container for CloudWatch metrics. Metrics in different namespaces are isolated from each other, so that metrics from different applications are not mistakenly aggregated into the same statistics
Metrics- Metrics are the fundamental concept in CloudWatch. A metric represents a time-ordered set of data points that are published to CloudWatch.
Dimensions- A dimension is a name/value pair that is part of the identity of a metric. You can assign up to 10 dimensions to a metric.
Statistics- Statistics are metric data aggregations over specified periods of time.
Percentiles- A percentile indicates the relative standing of a value in a dataset.
Alarms- You can use an alarm to automatically initiate actions on your behalf. An alarm watches

a single metric over a specified time period, and performs one or more specified actions, based on the value of the metric relative to a threshold over time. The action is a notification sent to an Amazon SNS topic or an Auto Scaling policy
Time Stamps- Each metric data point must be associated with a time stamp. CloudWatch alarms check metrics based on the current time in UTC. Custom metrics sent to CloudWatch with time stamps other than the current UTC time can cause alarms to display the Insufficient Data state or result in delayed alarms.

IAM (Identity Access Management)

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources.
IAM provides access control in an authenticated (signed in) and authorized (has permissions) way to use the AWS resources.
When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account.
We can access IAM via AWS management console,AWS CLI,AWS SDK's,IAM HTTPS API's.
Every account has an account ID which is unique and ARN(Amazon resource name that can be connected to on the link displayed with ID).

→IAM Features:
Shared access to your AWS account
Granular permissions
Secure access to AWS resources for applications that run on Amazon EC2
Multi-factor authentication (MFA)
Identity federation
Identity information for assurance
PCI DSS Compliance
Integrated with many AWS services
Eventually Consistent
Free to use

IAM Users

An AWS Identity and Access Management (IAM) *user* is an entity that you create in AWS to represent the person or application that uses it to interact with AWS. A user in AWS consists of a name and credentials.
An IAM user with administrator permissions is not the same thing as the AWS account root user
● Max 5000 users in an aws account.

IAM Groups

An IAM group is a collection of IAM users. Groups let you specify permissions for multiple users, which can make it easier to manage the permissions for those users. For example, you could have a group called *Admins* and give that group the types of permissions that administrators typically need. Any user in that group automatically has the permissions that are assigned to the group.

- Max 100 groups in an aws account.

IAM 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.

- Each role can have up to 10 policies attached.
- Max 250 roles in an aws account.

IAM Policies

A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when a principal entity (user or role) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents.

AWS supports six types of policies: identity-based policies, resource-based policies, permissions boundaries, Organizations SCPs, ACLs, and session policies.

Identity-based policies – Attach managed and inline policies to IAM identities (users, groups to which users belong, or roles).
Resource-based policies – Attach inline policies to resources.
Permissions boundaries – Use a managed policy as the permissions boundary for an IAM entity (user or role).
Organizations SCPs – Use an AWS Organizations service control policy (SCP) to define the maximum permissions for account members of an organization or organizational unit (OU). SCPs limit permissions that identity-based policies or resource-based policies grant to entities (users or roles) within the account, but do not grant permissions.
Access control lists (ACLs) – Use ACLs to control which principals in other accounts can access the resource to which the ACL is attached. ACLs are similar to resource-based policies, although they are the only policy type that does not use the JSON policy document structure. ACLs are cross-account permissions policies that grant permissions to the specified principal entity. ACLs cannot grant permissions to entities within the same account.
Session policies – Pass an advanced session policy when you use the AWS CLI or AWS API to assume a role or a federated user. Session policies limit the permissions that the role or user's identity-based policies grant to the session. Session policies limit permissions for a created session, but do not grant permissions.

IAM Identity Providers

When you want to configure federation with an external identity provider (IdP) service, you create an IAM *identity provider* to inform AWS about the IdP and its configuration. This establishes "trust" between your AWS account and the IdP. You can try it with **Open ID connect(OIDC)/SAML(Security Assertion Markup Language)** to provide the identity permission.

Multi-Factor Authentication (MFA)
MFA adds extra security because it requires users to provide unique authentication from an AWS supported MFA mechanism in addition to their regular sign-in credentials when they access AWS websites or services:
You can get them enabled on:
Virtual MFA devices.
U2F security key.(Universal 2nd Factor Key).
Hardware MFA device.
SMS text message-based MFA.

Setting a password policy options:

List of options available to make/configure password rotation policy.
Minimum password length
Require at least one uppercase letter
Require at least one lowercase letter
Require at least one number
Require at least one nonalphanumeric character
Allow users to change their own password
Enable password expiration/Password rotation policy.
Prevent password reuse
Password expiration requires administrator reset

Credential Report
● Status reports can be retrived all your account's users and the use of their various credentials including passwords, access keys, and MFA devices via report.

Encryption keys
● AWS Key Management Service (AWS KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data. AWS KMS is integrated with other AWS services including Amazon Elastic Block Store (Amazon EBS), Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Elastic Transcoder, Amazon WorkMail, Amazon Relational Database Service (Amazon RDS),EC2 snapshots and others to make it simple to encrypt your data with encryption keys that you manage.
● AWS KMS lets you create CUSTOMER MASTER KEYS that can never be exported from the service and which can be used to encrypt and decrypt data based on policies you define.

Best practices for IAM:

Lock Away Your AWS Account Root User Access Keys
Create Individual IAM Users
Use Groups to Assign Permissions to IAM Users
Grant Least Privilege
Get Started Using Permissions With AWS Managed Policies
Use Customer Managed Policies Instead of Inline Policies
Use Access Levels to Review IAM Permissions
Configure a Strong Password Policy for Your Users
Enable MFA for Privileged Users
Use Roles for Applications That Run on Amazon EC2 Instances
Use Roles to Delegate Permissions
Do Not Share Access Keys
Rotate Credentials Regularly
Remove Unnecessary Credentials
Use Policy Conditions for Extra Security
Monitor Activity in Your AWS Account

S3 -Simple Storage Service

● Amazon Simple Storage Service (Amazon S3) is object storage with a simple web service interface to store and retrieve any amount of data from anywhere on the web.
● S3 is object based storage type.
● A bucket is a logical unit of storage used to store data in S3. Buckets have a unique namespace for each region.
●Storage is global service but buckets created are region specific..
● Durability delivered here is 99.999999999%.
● Files size to be stored can be from 1 byte to 5 TB.
● By default, you can create up to 100 buckets in each of your AWS accounts.
● A bucket has no size limit. It can store numbers of objects of any size,as storage.
they are giving unlimited.
●Unique name space of the file stored in bucket is:https://s3.region name.amazon.aws.com/bucketname/filename.
● You can ground the permission level to bucket, file and folders as well.

Storage types

1. Standard s3 storage
2. Standard s3 - Infrequent Access(I.A)
3. Reduced redundancy Storage(RRS)
4. Amazon glacier ---Now its renamed to S3 Glacier.

● **Standard s3 storage:** This storage class is ideal for performance-sensitive use cases and frequently accessed data. It is the default storage class; if you don't specify storage class at the time that you upload an object, Amazon S3 assumes the standard storage class.
● **Standard s3 - Infrequent Access** (**Standard - IA**): This storage class (IA, for infrequent access) is optimized for long-lived and less frequently accessed data, for example backups and older data where of access has diminished, but the use case still demands high performance**.**
● **Reduced redundancy:** The Reduced Redundancy Storage (RRS) storage class is designed for noncritical, reproducible data stored at lower levels of redundancy than the STANDARD storage class, which reduces storage costs. The durability level corresponds to an average annual expected loss of 0.01% of objects. For example, if you store 10,000 objects you may loss 100 files.
● **Amazon glacier:** The GLACIER storage class is suitable for archiving data where data access is infrequent. Archived objects are not available for real-time access. You must first restore the objects before you can access them. The GLACIER storage class uses the very low-cost Amazon Glacier storage service.

(note: initially you might upload objects using the STANDARD storage class, and then use a bucket lifecycle configuration rule to transition objects STANDARD_IA or GLACIER storage)
(note: consistency model s3 uses read-after-write consistency for PUTS of new objects and eventual consistency for overwrite PUTS and DELETES)

Permission

● Bucket permissions specify who is allowed access to the objects in a bucket and what permissions you have granted them.
● You can grant the permission for:

1. **Everyone—**Use this group to grant anonymous access
2. **Authenticated Users—**This group consists of any user that has an Amazon AWS Account. When you grant the Authenticated User group permission, any valid signed request can perform the appropriate action. The request can be signed by either an AWS Account or IAM User.
3. **Log Delivery—**This group grants write access to your bucket when the bucket is used to store server access logs.
4. **Me—**This group refers to your AWS root account, and not an IAM user.

S3 VERSIONING

● Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures.
● Once we enable version in a bucket, it can never return to an unversioned state. You can, however, suspend versioning on that bucket.

Lifecycle Management

- You can manage the lifecycle of objects by using Lifecycle rules.
- Lifecycle rules enable you to automatically transition objects to the Standard - Infrequent Access Storage Class, and/or archive objects to the Glacier Storage Class, and/or remove objects after a specified time period.

Static website hosting

- You can host a static website on Amazon S3. On a static website, individual web pages include static content.
- To host your static website, you configure an Amazon S3 bucket for website hosting and then upload your website content to the bucket.

Logging

- In order to track requests for access to your bucket, you can enable access logging.
- Each access log record provides details about bucket name, request time, request action, response status, and error code, if any.
- Access log information can be useful in security and access audits.
- Logging is region specific.

Cross-region replication

- Cross-region replication is a bucket-level feature that enables automatic, asynchronous copying of objects across buckets in different AWS regions.
- The object replicas in the destination bucket are exact replicas of the objects in the source bucket. They have the same key names and the same metadata.
- Existing objects of source bucket will not be copied to destination bucket.
- The source and destination buckets must be versioning-enabled.
- The source and destination buckets must be in different AWS regions.
- You can replicate objects from a source bucket to only one destination bucket.

S3 Multipart Upload

- S3 multipart allows you to upload a single object in multiple part. The object is assembled after all uploads.
- Parts can be uploaded in parallel for high throughput.
- Uploads can be paused and resumed.
- Objects can be uploaded and while we are creating it.

S3 Data Encryption

- S3 data encryption provides added security for your data.
- Server-side encryption encrypts your data before storing it in its data center and decrypts it when you access it.
- S3 uses 256-bit Advanced Encryption Standard (AES) to encrypt your data.

Events

● The Amazon S3 notification feature enables you to receive notifications when certain events happen in your bucket.
● Events are
1. A new object created event
2. An object removal event
3. A Reduced Redundancy Storage (RRS) object lost event

Tags

● Tags are used to identify and categories your aws resources.
● We can use tags to organize your AWS bill to reflect your own cost structure.
● Tags consists of key and value.

(note: mainly used to identify from which bucket bill is high)

Requester Pays bucket

● In general, bucket owners pay for all Amazon S3 storage and data transfer costs associated with their bucket.
● With Requester Pays buckets, the requester instead of the bucket owner pays the cost of the request and the data download from the bucket. The bucket owner always pays the cost of storing data.
● We can configure buckets to be Requester Pays when you want to share data but not incur charges associated with others accessing the data.

Amazon S3 Transfer Acceleration

● Amazon S3 Transfer Acceleration enables fast, easy, and secure transfers of files over long distances between your client and an S3 bucket
● Transfer Acceleration takes advantage of Amazon CloudFront's globally distributed edge locations. As the data arrives at an edge location, data is routed to Amazon S3 over an optimized network path.
● When using Transfer Acceleration, additional data transfer charges may apply.

Use

● customers that upload to a centralized bucket from all over the world.
● transfer gigabytes to terabytes of data on a regular basis across continents.
● underutilize the available bandwidth over the Internet when uploading to Amazon S3

Storage Management

● Amazon S3 Storage Management capabilities helps you better analyze and manage your storage by
1. S3 Object Tagging
2. S3 Analytics, Storage Class Analysis
3. S3 Inventory

4. S3 CloudWatch Metrics

● **S3 Object Tagging** – With S3 Object Tagging you can manage and control access for Amazon S3 objects. S3 Object Tags are key-value pairs applied to S3 objects which can be created, updated or deleted at any time during the lifetime of the object. With these, you'll have the ability to create Identity and Access Management (IAM) policies, setup S3 Lifecycle policies, and customize storage metrics. These object-level tags can then manage transitions between storage classes and expire objects in the background.
● **S3 Analytics, Storage Class** Analysis – With storage class analysis, you can analyze storage access patterns and transition the right data to the right storage class. This new S3 Analytics feature automatically identifies the optimal lifecycle policy to transition less frequently accessed storage to SIA. You can configure a storage class analysis policy to monitor an entire bucket, a prefix, or object tag. Once an infrequent access pattern is observed, you can easily create a new lifecycle age policy based on the results. Storage class analysis also provides daily visualizations of your storage usage in the AWS Management Console. You can export these to an S3 bucket to analyze using the business intelligence tools of your choice, such as Amazon QuickSight.
● **S3 Inventory** – You can simplify and speed up business workflows and big data jobs using S3 Inventory, which provides a scheduled alternative to Amazon S3's synchronous List API. S3 Inventory provides a CSV (Comma Separated Values) flat-file output of your objects and their corresponding metadata on a daily or weekly basis for an S3 bucket or a shared prefix.
● **S3 CloudWatch Metrics** – Understand and improve the performance of your applications that use Amazon S3 by monitoring and alarming on 13 new S3 CloudWatch Metrics. You can receive 1-minute CloudWatch Metrics, set CloudWatch alarms, and access CloudWatch dashboards to view real-time operations and performance such as bytes downloaded and the 4xx HTTP response count of your Amazon S3 storage. For web and mobile applications that depend on cloud storage, these let you quickly identify and act on operational issues.

By default, 1-minute metrics are available at the S3 bucket level. You also have the flexibility to define a filter for the metrics collected using a shared prefix or object tag, allowing you to align metrics to specific business applications, workflows, or internal organizations.

EFS - Elastic File System(OBJECT BASED STORAGE)

● Amazon Elastic File System (Amazon EFS) provides simple, scalable file storage for use with Amazon EC2.

● Amazon EFS has a simple web services interface that allows you to create and configure file systems quickly and easily.

● The service manages all the file storage infrastructure for you, avoiding the complexity of deploying, patching, and maintaining complex file system deployments.

● Multiple Amazon EC2 instances can access an Amazon EFS file system at the same time, providing a common data source for workloads and applications running on more than one instance or server.

● Amazon EFS supports the Network File System version 4 (NFSv4.1 and NFSv4.0) protocol, so the applications and tools that you use today work seamlessly with Amazon EFS.

● Preprovisioning of storage is not required in EFS,storage capacity is elastic, automatically it shrinks and expands as you add and remove files, so your applications have the storage when they need it.

● Amazon EFS offers two storage classes, Standard and Infrequent Access. The Standard storage class is used to store frequently accessed files. The Infrequent Access (IA) storage class is a lower-cost storage class that's designed for storing long-lived, infrequently accessed files cost-effectively.

● The service is designed to be highly scalable, highly available, and highly durable. Amazon EFS file systems store data and metadata across multiple Availability Zones in an AWS Region.

● EFS file systems can grow to petabyte scale, drive high levels of throughput, and allow massively parallel access from Amazon EC2 instances to your data.

● Amazon EFS provides file system access semantics, such as strong data consistency and file locking. Amazon EFS also enables you to control access to your file systems through Portable Operating System Interface (POSIX) permissions.

● Amazon EFS supports two forms of encryption for file systems, encryption in transit and encryption at rest. You can enable encryption at rest when creating an Amazon EFS file system. If you do, all your data and metadata is encrypted. You can enable encryption in transit when you mount the file system.

● Amazon EFS is designed to provide the throughput, IOPS, and low latency needed for a broad range of workloads. With Amazon EFS, you can choose from two performance modes and two throughput modes:
The default general purpose performance mode is ideal for latency-sensitive use cases, like web serving environments, content management systems, home directories, and general file serving. File systems in the Max I/O mode can scale to higher levels of aggregate throughput and operations per second with a tradeoff of slightly higher latencies for file operations.
Using the default Bursting Throughput mode, throughput scales as your file system grows.
Using Provisioned Throughput mode, you can specify the throughput of your file system independent of the amount of data stored.

AMAZON CLOUDFRONT

● Amazon CloudFront is a web service that speeds up distribution of your static and dynamic web content, such as .html, .css, .js, and image files, to your users.
● CloudFront delivers your content through a worldwide network of data centers called edge locations.
● When a user requests content that you're serving with CloudFront, the user is routed to the edge location that provides the lowest latency (time delay), so that content is delivered with the best possible performance.

If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately.
If the content is not in that edge location, CloudFront retrieves it from an origin that you've defined—such as an Amazon S3 bucket, a MediaPackage channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

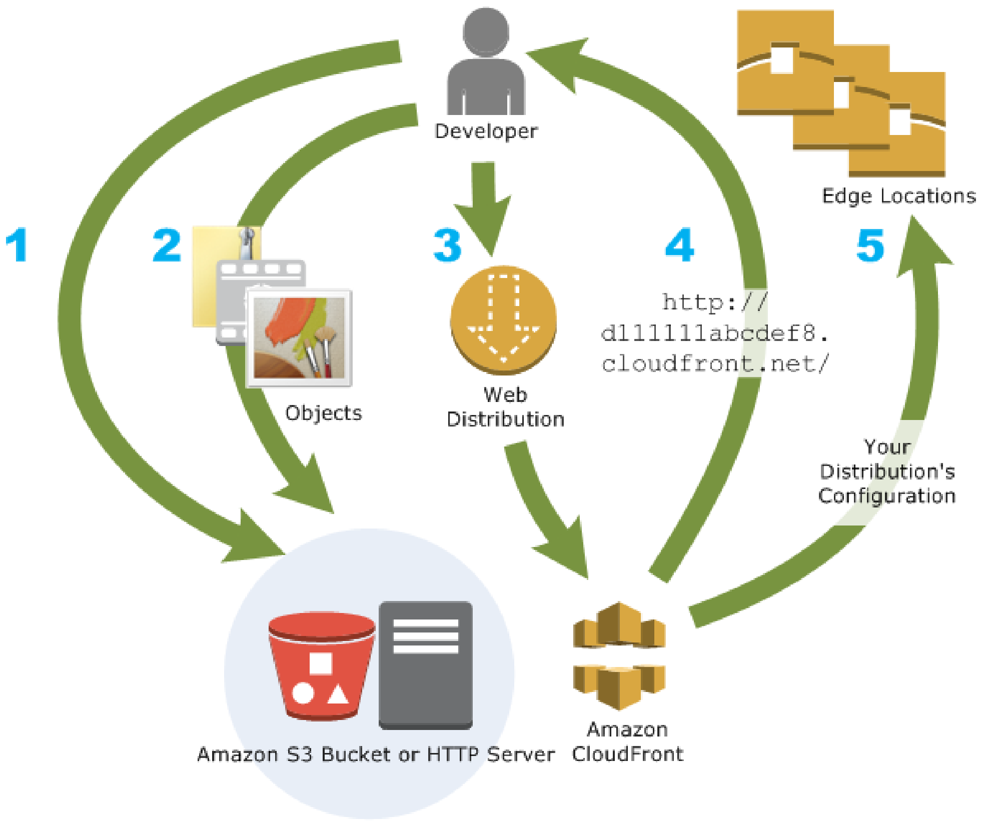
Types of CloudFront distributions

Web distribution –
Speed up distribution of static and dynamic content, for example, .html, .css, .php, and graphics files.
Distribute media files using HTTP or HTTPS.
Add, update, or delete objects, and submit data from web forms.
Use live streaming to stream an event in real time.
You store your files in an origin - either an Amazon S3 bucket or a web server. After you create the distribution, you can add more origins to the distribution.

RTMP distribution – An RTMP distribution allows an end user to begin playing a media file before the file has finished downloading from a CloudFront edge location.

Note:
To create an RTMP distribution, you must store the media files in an Amazon S3 bucket.
To use CloudFront live streaming, create a web distribution.

Configuring Cloud front to deliver your content:


Developer
Edge Locations
1
2
3
4
5
Objects
Web Distribution
http://d111111abcdef8.cloudfront.net/
Your Distribution's Configuration
Amazon S3 Bucket or HTTP Server
Amazon CloudFront

Terminologies:

ORIGIN SETTINGS:

1. Origin Domain Name - The domain name for your origin can be the Amazon S3 bucket, AWS MediaPackage channel endpoint, AWS MediaStoreContainer endpoint or web server from which you want CloudFront to get your web content.
2. Origin Path – Folder path in S3 if your content needs to be retrived from the orig
3. Origin ID – Origin location from where you want your content to be retrived/accessed.(This value lets you distinguish multiple origins in the same distribution from one another. The description for each origin must be unique within the distribution).

DISTRIBUTION SETTINGS:

1. Price Class
Select the price class that corresponds with the maximum price that you want to pay for CloudFront service. By default, CloudFront serves your objects from edge locations in all CloudFront regions.
2.AWS WAF Web ACL
If you want to use AWS WAF to allow or block HTTP and HTTPS requests based on criteria

that you specify, choose the web ACL to associate with this distribution.
3.Alternate Domain Names (CNAMEs):
CNAME is nothing but canonical name-Specify one or more domain names that you want to use for URLs for your objects instead of the domain name that CloudFront assigns when you create your distribution.
4.SSL Certificate
Accept the default value, Default CloudFront Certificate.

5.Cookie Logging
Amazon S3 as the origin for your objects, and Amazon S3 doesn't process cookies, so it's recommended that you select Off for the value of Cookie Logging.

CACHE BEHAVIOUR SETTINGS:
Forward all requests that use the CloudFront URL for your distribution,to the Amazon S3 bucket that you have specified.
Allow end users to use either HTTP or HTTPS protocols to access your objects.
Respond to requests for your objects.
Cache your objects at CloudFront edge locations for 24 hours.
TTL:Time To Live(how long do youb want your contents to be present in the cached location-Min=24hrs/86400 seconds).
Forward only the default request headers to your origin and not cache your objects based on the values in the headers.
Exclude cookies and query string parameters, if any, when forwarding requests for objects to your origin. (Amazon S3 doesn't process cookies and processes only a limited set of query string parameters.)
Not be configured to distribute media files in the Microsoft Smooth Streaming format.
Allow everyone to view your content.
Not automatically compress your content.

Maximum Length of a Request and Maximum Length of a URL
The maximum length of a request, including the path, the query string (if any), and headers, is 20,480 bytes.
CloudFront constructs a URL from the request. The maximum length of this URL is 8192 bytes.
If a request or a URL exceeds these limits, CloudFront returns HTTP status code 413, Request Header Fields Too Large, to the viewer, and then terminates the TCP connection to the viewer.
Maximum File Size
The maximum size of a response body that CloudFront will return to the viewer is 20 GB. This includes chunked transfer responses that don't specify the Content-Length header value.
Restricting the Geographic Distribution of Your Content
You can use geo restriction, also known as geoblocking, to prevent users in specific geographic locations from accessing content that you're distributing through a CloudFront web distribution. To use geo restriction, you have two options:

Use the CloudFront geo restriction feature. Use this option to restrict access to all of the files that are associated with a distribution and to restrict access at the country level.

Use a third-party geolocation service. Use this option to restrict access to a subset of the files that are associated with a distribution or to restrict access at a finer granularity than the country level.

When a user requests your content, CloudFront typically serves the requested content regardless of where the user is located. If you need to prevent users in specific countries from accessing your content, you can use the CloudFront geo restriction feature to do one of the following:

Allow your users to access your content only if they're in one of the countries on a whitelist of approved countries.

Prevent your users from accessing your content if they're in one of the countries on a blacklist of banned countries.

Removing Content so CloudFront Won't Distribute It

You can remove files from your origin that you no longer want to be included in your CloudFront distribution. However, CloudFront will continue to show viewers content from the edge cache until the files expire.

If you want to remove a file right away, you must do one of the following:

Invalidate the file. For more information, see Invalidating Files.

Use file versioning. When you use versioning, different versions of a file have different names that you can use in your CloudFront distribution, to change which file is returned to viewers.

VPC - Virtual Private Cloud

Overview:

● Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of Amazon Web Services (AWS) cloud where you can launch AWS resources into a virtual network that you define.

You have complete control over your virtual networking environment,including selection of your own IP address ranges,creation of subnets,configuration of route tables and network gateways.

Types of VPC

1. Default VPC
2. Custom VPC

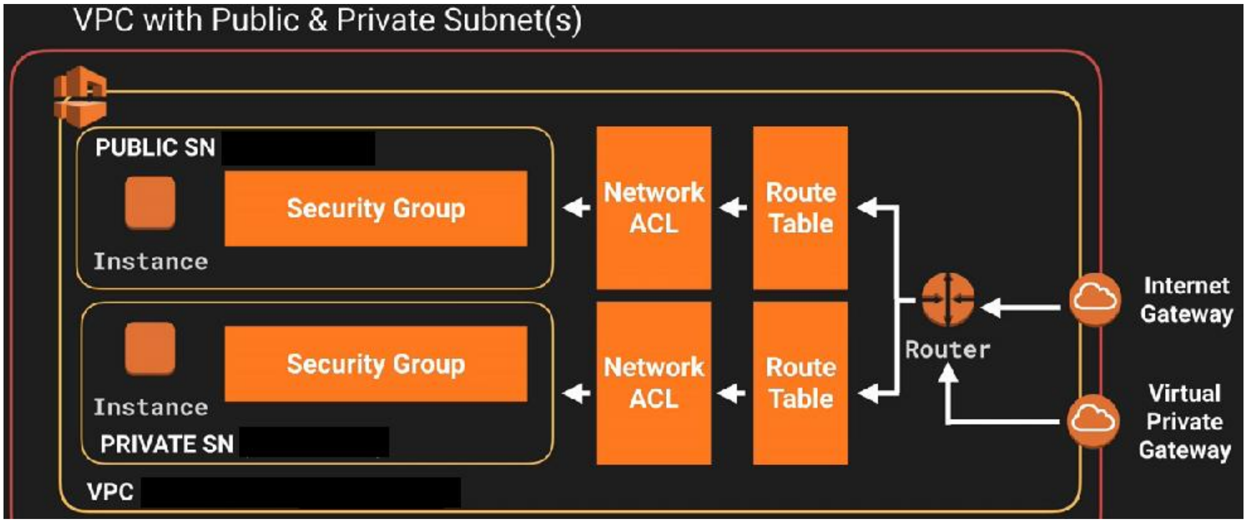
Default VPC

User friendly and instantly available for associating with other services.

Custom VPC

●We customise our VPC with IP ranges,network gateways,subnets etc..as per requirement.

.



When a VPC is created there are 3 things which by default creates:

1.NACL
2.Security group
3.Route Table.

And the things which doesnt get created and we have to customize is:

1.Subnets
2.Internet Gateway

Terminologies:

Subnet: Part of network chosen is called subnet.
NACL: Network Access Control List-
A network access control list is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.
These are also called as stateless because as you add your inbound rules need to add your outbound rules as well.
Route table: A route table contains a set of rules, called routes, that are used to determine where network traffic is directed.
Internet Gateway: An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet

Amazon VPC Limits

The following tables list the limits for Amazon VPC resources per Region for your AWS account.

VPC and Subnets

Resource	Default limit	Comments
VPCs per Region	5	The limit for internet gateways per Region is directly correlated to this one. Increasing this limit increases the limit on internet gateways per Region by the same amount. Customers can have 100s of VPCs per Region for their needs even though the default limit is 5 VPCs per Region. You can request an increase for these limits using the Amazon VPC limits form.
Subnets per VPC	200	-
IPv4 CIDR blocks per VPC	5	This limit is made up of your primary CIDR block plus 4 secondary CIDR blocks.
IPv6 CIDR blocks per VPC	1	This limit cannot be increased.

Elastic IP Addresses (IPv4)

Resource	Default limit	Comments
Elastic IP addresses per Region	5	This is the limit for the number of Elastic IP addresses for use in EC2-VPC.

Flow Logs

Resource	Default limit	Comments
Flow logs per single network interface, single subnet, or single VPC in a Region	2	This limit cannot be increased. You can effectively have 6 flow logs per network interface if you create 2 flow logs for the subnet, and 2 flow logs for the VPC in which your network interface resides.

Gateways

Resource	Default limit	Comments
Customer gateways per Region	50	-
Egress-only internet gateways per Region	5	This limit is directly correlated with the limit on VPCs per Region. To increase this limit, increase the limit on VPCs per Region. You can attach only one egress-only internet gateway to a VPC at a time.
Internet gateways	5	This limit is directly correlated with the limit on VPCs per Region.

per Region		To increase this limit, increase the limit on VPCs per Region. Only one internet gateway can be attached to a VPC at a time.
NAT gateways per Availability Zone	5	A NAT gateway in the pending, active, or deleting state counts against your limit.
Virtual private gateways per Region	5	You can attach only one virtual private gateway to a VPC at a time.

Network ACLs

Resource	Default limit	Comments
Network ACLs per VPC	200	You can associate one network ACL to one or more subnets in a VPC. This limit is not the same as the number of rules per network ACL.
Rules per network ACL	20	This is the one-way limit for a single network ACL, where the limit for ingress rules is 20, and the limit for egress rules is 20. This limit includes both IPv4 and IPv6 rules, and includes the default deny rules (rule number 32767 for IPv4 and 32768 for IPv6, or an asterisk * in the Amazon VPC console). This limit can be increased up to a maximum of 40; however, network performance might be impacted due to the increased workload to process the additional rules.

Route Tables

Resource	Default limit	Comments
Route tables per VPC	200	This limit includes the main route table.

Security Groups

Resource	Default limit	Comments
VPC security groups per Region	2500	The maximum is 10000. If you have more than 5000 security groups in a Region, we recommend that you paginate calls to describe your security groups for better performance.
Inbound or outbound rules per security group	60	You can have 60 inbound and 60 outbound rules per security group (making a total of 120 rules). This limit is enforced separately for IPv4 rules and IPv6 rules; for example, a security group can have 60 inbound rules for IPv4 traffic and 60 inbound rules for IPv6 traffic. A rule that references a security group or prefix list ID counts as one rule for IPv4 and one rule for IPv6. A limit change applies to both inbound and outbound rules. This limit

		multiplied by the limit for security groups per network interface cannot exceed 1000. For example, if you increase this limit to 100, we decrease the limit for your number of security groups per network interface to 10.
Security groups per network interface	5	To increase or decrease this limit, contact AWS Support. The maximum is 16. The limit for security groups per network interface multiplied by the limit for rules per security group cannot exceed 1000. For example, if you increase this limit to 10, we decrease the limit for your number of rules per security group to 100.

VPC Peering Connections

Resource	Default limit	Comments
Active VPC peering connections per VPC	50	The maximum limit is 125 peering connections per VPC. The number of entries per route table should be increased accordingly; however, network performance might be impacted.
Outstanding VPC peering connection requests	25	This is the limit for the number of outstanding VPC peering connection requests that you've requested from your account.
Expiry time for an unaccepted VPC peering connection request	1 week (168 hours)	-

VPC Endpoints

Resource	Default limit	Comments
Gateway VPC endpoints per Region	20	You cannot have more than 255 gateway endpoints per VPC.
Interface VPC endpoints per VPC	20	The maximum limit for interface endpoints per Region is this limit multiplied by the number of VPCs in the Region.

VPC and Subnet Sizing for IPv4

When you create a VPC, you must specify an IPv4 CIDR block for the VPC. The allowed block size is between a /16 netmask (65,536 IP addresses) and /28 netmask (16 IP addresses). After you've created your VPC, you can associate secondary CIDR blocks with the VPC.

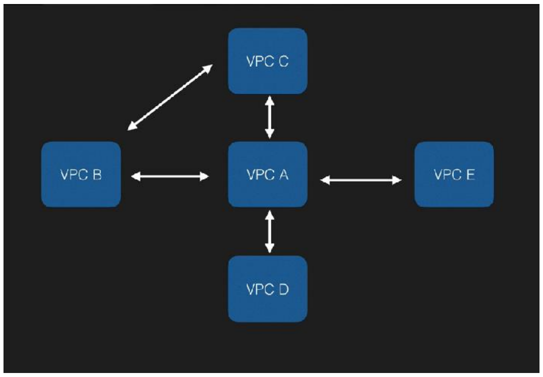
When you create a VPC, we recommend that you specify a CIDR block (of /16 or smaller) from the private IPv4 address ranges as specified in RFC 1918:

10.0.0.0 - 10.255.255.255 (10/8 prefix)
172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

The first four IP addresses and the last IP address in each subnet CIDR block are not available for you to use, and cannot be assigned to an instance. For example, in a subnet with CIDR block 10.0.0.0/24, the following five IP addresses are reserved:
10.0.0.0: Network address.
10.0.0.1: Reserved by AWS for the VPC router.
10.0.0.2: Reserved by AWS.
10.0.0.3: Reserved by AWS for future use.
10.0.0.255: Network broadcast address. We do not support broadcast in a VPC, therefore we reserve this address

VPC peering

● A **VPC peering** connection allows you to connect one VPC with other via direct network route using private IP addresses.
● We can peer VPC with other AWS account as well as with other VPC in the same account.
● Peering is in a star configuration i.e. 1 central VPC peers with 4 others."NO TRANSITIVE PEERING".



RDS –Relational Database Services:

● A **database** is a collection of information that can stored,organized,so that it can be easily accessed, managed and updated without altering databases.

● Types of databases
1. **Relational DB -**A **relational database** is a collection of data items organized as a set of formally-described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database itself.
● Amazon have a service called RDS (relational database service) including 6 different db MySQL, MariaDB, SQL, PostgreSQL, oracle, Amazon Aurora.

Ex:Excel/Spreadsheet---TABLES,ROWS,COLUMS

2. **Non-relational DB**
● A non-relational database is any database that does not follow the traditional approach.
● DynamoDB is an AWS non relational DB service.

Ex:Documents-Collections-Key valuepairs.

Amazon Aurora

● Amazon Aurora is a fully managed, MySQL-compatible, relational database engine that combines the speed and reliability of high-end commercial databases with the simplicity and cost-effectiveness of open-source databases

● Amazon Aurora provides 5 times better performance than MySQL, at a price point one tenth of a commercial DB while delivering similar performance and availability.

● Amazon Aurora default size is 10GB max it can scale upto 64TB.

● Compute resource can scale upto 32 vCPUs and 244 GB of memory.

●It is designed to transparently handle the loss of data,upto 2 copies of data without affecting the database write availability and upto 3 copies of data without affecting read availability.

●It also has self healing property,by continuosly scanning for data blocks and disk errors.

Types of Aurora replica are available:
Aurora Replications-Currently 15
MYSQL read replicas-Currently5

When a replica is created the endpoint you can see it as”Cluster endpoint”,whereas the read replica would be the “Instance end point”.Main one will be the writer and replicated would be the reader.

Amazon RDS DB Instance

● A *DB instance* is an isolated database environment running in the cloud.

● A DB instance can contain multiple user-created databases.

● We can have up to 40 Amazon RDS DB instances.

● Production environment mainly uses multi AZ deployment, it provides enhanced availability and data durability for instance.

● RDS automatically provision and maintain a synchronous “standby” replica in different AZ.

● RDS automatically fails over to the up-to-date standby database ensuring that database operations resume quickly without administrator
intervention, in the event of panned database maintenance or unplanned service disruption.

Read Replica

● It makes it easy for scaling it beyond the capacity constraints of a single DB instance for read-heavy database workloads.

● They can be used for serving read traffic when the primary database is unavailable

DB Snapshot and Automated Backup

● RDS provides 2 ways of backing and restoring your instance
1. Snapshots
2. Automated Backup

● Snapshots are user triggered (can be automated via script or application)
● Automated backup are automatic and give the ability to restore point-in-time.
● Both are billable in terms of storage.

DynamoDB

● Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need Single digit millisecond latency at any scale.
● It is a fully managed database and supports both the documents and key value data models.
● Amazon DynamoDB automatically spreads the data and traffic for the table over a sufficient number of servers to handle the request capacity specified by the customer and the amount of data stored, while maintaining consistent and fast performance.
● Its flexible data model and reliable performance make it a great fit for the mobile, web, gaming,IOT and ad-tech etc.
● It always stores on SSD storage there is no magnetic storage.
●Spread across 3 geographically distinct data center(multiple AZ's)
Eventual consistent reads(defaul) and strong consistent reads.

DynamoDB pricing:
1.a.Provisioned thoughput capacity:Write throughput $ 0.0065/hour for every 10units.
1.b.Read throughput capacity:$0.0065/hour for every 50 units.
2.Storage costs-$0.25GB/month.

Datawarehouse:

● A data warehouse exists as a layer on top of another database or databases.
● RedShift is a dataware house service in AWS.
●Used to calculate or for data analytics,in order to forecast/calculate current performance vs Targets by management.
Concepts used:
OLTP:Online transaction processing
OLAP:Online Analytics processing.

Redshift

● Amazon Redshift is a fast and powerful, fully managed, petabyte-scale data warehouse service in the cloud.
● Customers can start small for just $0.25 per hour with no commitments or upfront cost and scale to a petabyte or more for $1000 or more tera byte per year, less than a 10th of most other data base solution.
● Amazon Redshift is 10 times faster than traditional warehousing

storage). i.e. instead of storing data as a series of rows, amazon redshift organize data by column.

- Datawarehouse is all about suming up colums.

Redshift Configuration:
Single node(160gb)
Leader node(manages client connections and receives queries).
Compute node(Stor data and perform queries and computations(can have upto 128 compute nodes).

Redshift Pricing:
You will be charged in Redshift based on:Compute node & Data Backup & Data transfer.

Redshift Security/Encryption:
Encrypted in transit using SSL.
Encrypted at rest using AES-256 encryption.
By default Redshifttakes care of key management
1.Manage your own keys through HSM.
2.AWS key management service.

Redshift Features:

Columnar data storage:
Instead of storing data as a series of rows,Amazon redshift organizes the data by column.
Unlike row based systems which are ideal for transaction processing,column based systems are ideal for datawarehousing and analytics,where queries often involve aggregates performed over large data sets.

Advanced data compression:
Columnar data stores can be compressed much more than a row based data stores,because similar data is stored sequentially on disk.

Massive parallel processing:
It automatically distributes data and query load across all nodes.
Redshift makes it easy to add nodes to your datawarehouse to maintain fast query performance as your datawarehouse grows.

Redshift Availability:
Currently available in only one AZ,as its only for management usage and not for clients to store their data.
Can store snapshots to new AZ's in the event of an outage.

Route 53

● Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking
● Route 53 is named after or associated with DNS port#53.
● It provides secure routing connection to aws service such as EC2, ELB, S3.
● Route 53 lets you register a name for your website or web application, known as a *domain name*. Route 53 is global service.
● You can **1. Register domain names**
● You can **2. Route internet traffic to the resources for your domain**
● You can **3. Check the health of your resources.**

Domain Registration Concepts:

Here's an overview of the concepts that are related to domain registration.

domain name

The name, such as example.com, that a user types in the address bar of a web browser to access a website or a web application. To make your website or web application available on the internet, you start by registering a domain name.

domain registrar

A company that is accredited by ICANN (Internet Corporation for Assigned Names and Numbers) to process domain registrations for specific top-level domains (TLDs). For example, Amazon Registrar, Inc. is a domain registrar for .com, .net, and .org domains. Our registrar associate, Gandi, is a domain registrar for hundreds of TLDs, such as .apartments, .boutique, and .camera.

domain registry

A company that owns the right to sell domains that have a specific top-level domain. For example, VeriSign is the registry that owns the right to sell domains that have a .com TLD. A domain registry defines the rules for registering a domain, such as residency requirements for a geographic TLD. A domain registry also maintains the authoritative database for all of the domain names that have the same TLD. The registry's database contains information such as contact information and the name servers for each domain.

domain reseller

A company that sells domain names for registrars such as Amazon Registrar. Amazon Route 53 is a domain reseller for Amazon Registrar and for our registrar associate, Gandi.

top-level domain (TLD)

The last part of a domain name, such as .com, .org, or .ninja. There are two types of top-level domains:

generic top-level domains

These TLDs typically give users an idea of what they'll find on the website. For example, domain names that have a TLD of *.bike* often are associated with websites for motorcycle or bicycle businesses or organizations. With a few exceptions, you can use any generic TLD you want, so a bicycle club could use the .hockey TLD for their domain name.

geographic top-level domains

These TLDs are associated with geographic areas such as countries or cities. Some registries for geographic TLDs have residency requirements, while others, such as .io, allow or even encourage use as a generic TLD.

Domain Name System (DNS) Concepts

Here's an overview of the concepts that are related to the Domain Name System (DNS).

alias record

A type of record that you can create with Amazon Route 53 to route traffic to AWS resources such as Amazon CloudFront distributions and Amazon S3 buckets. For more information, see Choosing Between Alias and Non-Alias Records.

authoritative name server

A name server that has definitive information about one part of the Domain Name System (DNS) and that responds to requests from a DNS resolver by returning the applicable information. For example, an authoritative name server for the .com top-level domain (TLD) knows the names of the name servers for every registered .com domain. When a .com authoritative name server receives a request from a DNS resolver for example.com, it responds with the names of the name servers for the DNS service for the example.com domain.

Route 53 name servers are the authoritative name servers for every domain that uses Route 53 as the DNS service. The name servers know how you want to route traffic for your domain and subdomains based on the records that you created in the hosted zone for the domain.

(Route 53 name servers store the hosted zones for the domains that use Route 53 as the DNS service.)

For example, if a Route 53 name server receives a request for www.example.com, it finds that record and returns the IP address, such as 192.0.2.33, that is specified in the record.

DNS query

Usually a request that is submitted by a device, such as a computer or a smart phone, to the Domain Name System (DNS) for a resource that is associated with a domain name. The most common example of a DNS query is when a user opens a browser and types the domain name in the address bar. The response to a DNS query typically is the IP address that is associated with a resource such as a web server. The device that initiated the request uses the IP address to communicate with the resource. For example, a browser can use the IP address to get a web page from a web server.

DNS resolver

A DNS server, often managed by an internet service provider (ISP), that acts as an intermediary between user requests and DNS name servers. When you open a browser and enter a domain name in the address bar, your query goes first to a DNS resolver. The resolver communicates with DNS name servers to get the IP address for the corresponding resource, such as a web server. A DNS resolver is also known as a recursive name server because it sends requests to a sequence of authoritative DNS name servers until it gets the response (typically an IP address) that it returns to a user's device, for example, a web browser on a laptop computer.

Domain Name System (DNS)

A worldwide network of servers that help computers, smart phones, tablets, and other IP-enabled devices to communicate with one another. The Domain Name System translates easily understood names such as example.com into the numbers, known as *IP addresses*, that allow computers to find each other on the internet.

hosted zone

A container for records, which include information about how you want to route traffic for a domain (such as example.com) and all of its subdomains (such as www.example.com, retail.example.com, and seattle.accounting.example.com). A hosted zone has the same name as the corresponding domain.

For example, the hosted zone for example.com might include a record that has information about routing traffic for www.example.com to a web server that has the IP address 192.0.2.243, and a record that has information about routing email for example.com to two email servers, mail1.example.com and mail2.example.com. Each email server also requires its own record.

IP address

A number that is assigned to a device on the internet—such as a laptop, a smart phone, or a web server—that allows the device to communicate with other devices on the internet. IP addresses are in one of the following formats:

Internet Protocol version 4 (IPv4) format, such as 192.0.2.44

Internet Protocol version 6 (IPv6) format, such as 2001:0db8:85a3:0000:0000:abcd:0001:2345

Route 53 supports both IPv4 and IPv6 addresses for the following purposes:

You can create records that have a type of A, for IPv4 addresses, or a type of AAAA, for IPv6 addresses.

You can create health checks that send requests either to IPv4 or to IPv6 addresses.

If a DNS resolver is on an IPv6 network, it can use either IPv4 or IPv6 to submit requests to Route 53.

name servers

Servers in the Domain Name System (DNS) that help to translate domain names into the IP addresses that computers use to communicate with one another. Name servers are either recursive name servers (also known as DNS resolver) or authoritative name servers.

private DNS

A local version of the Domain Name System (DNS) that lets you route traffic for a domain and its subdomains to Amazon EC2 instances within one or more Amazon virtual private clouds (VPCs).

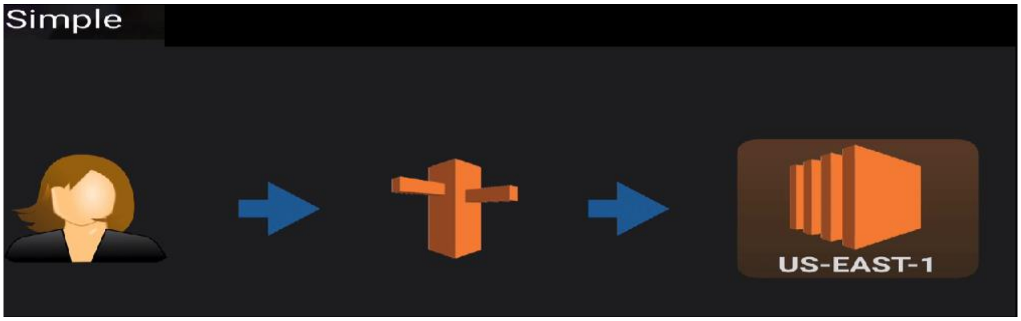
record (DNS record)

An object in a hosted zone that you use to define how you want to route traffic for the domain or a subdomain. For example, you might create records for example.com and www.example.com that route traffic to a web server that has an IP address of 192.0.2.234.

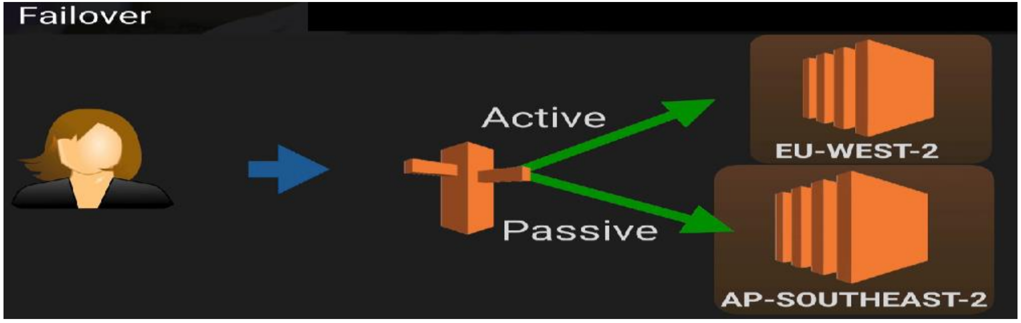
Types of Routing policies

A setting for records that determines how Route 53 responds to DNS queries. Route 53 supports the following routing policies:

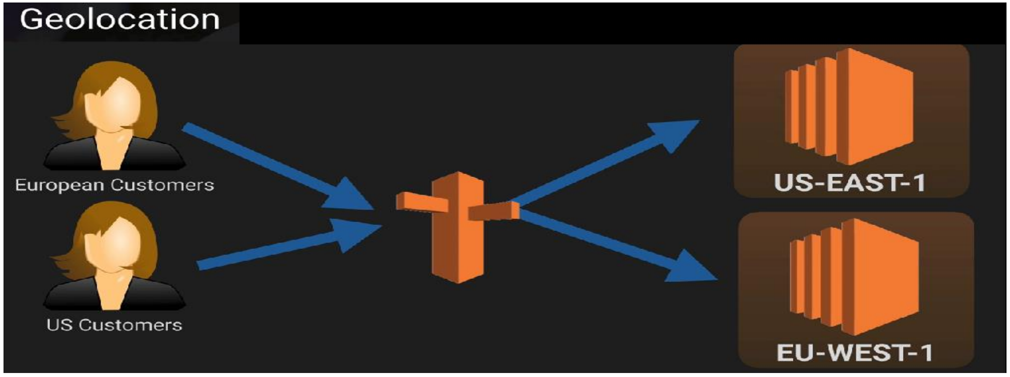
Simple routing policy – Use to route internet traffic to a single resource that performs a given function for your domain, for example, a web server that serves content for the example.com website.



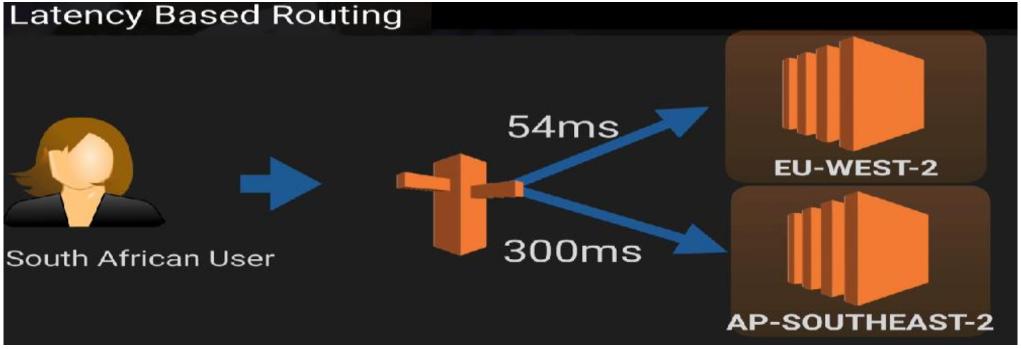
Failover routing policy – Use when you want to configure active-passive failover.



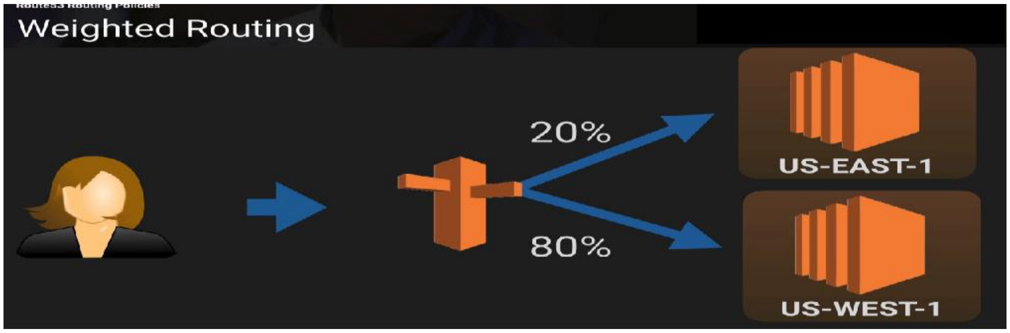
Geolocation routing policy – Use when you want to route internet traffic to your resources based on the location of your users.



Latency routing policy – Use when you have resources in multiple locations and you want to route traffic to the resource that provides the best latency.



Weighted routing policy – Use to route traffic to multiple resources in proportions that you specify.



Geoproximity routing policy – Use when you want to route traffic based on the location of your resources and, optionally, shift traffic from resources in one location to resources in another.

Multivalue answer routing policy – Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random.

Subdomain

A domain name that has one or more labels prepended to the registered domain name. For example, if you register the domain name example.com, then www.example.com is a subdomain. If you create the hosted zone accounting.example.com for the example.com domain, then seattle.accounting.example.com is a subdomain.

To route traffic for a subdomain, create a record that has the name that you want, such as www.example.com, and specify the applicable values, such as the IP address of a web server.

Time to Live (TTL)

The amount of time, in seconds, that you want a DNS resolver to cache (store) the values for a record before submitting another request to Route 53 to get the current values for that record. If the DNS resolver receives another request for the same domain before the TTL expires, the resolver returns the cached value.

A longer TTL reduces your Route 53 charges, which are based in part on the number of DNS queries that Route 53 responds to. A shorter TTL reduces the amount of time that DNS resolvers route traffic to older resources after you change the values in a record, for example, by changing the IP address for the web server for www.example.com.

Health Checking Concepts

Here's an overview of the concepts that are related to Amazon Route 53 health checking.

DNS failover

A method for routing traffic away from unhealthy resources and to healthy resources. When you have more than one resource performing the same function—for example, more than one web server or mail server—you can configure Route 53 health checks to check the health of your resources and configure records in your hosted zone to route traffic only to healthy resources.

endpoint

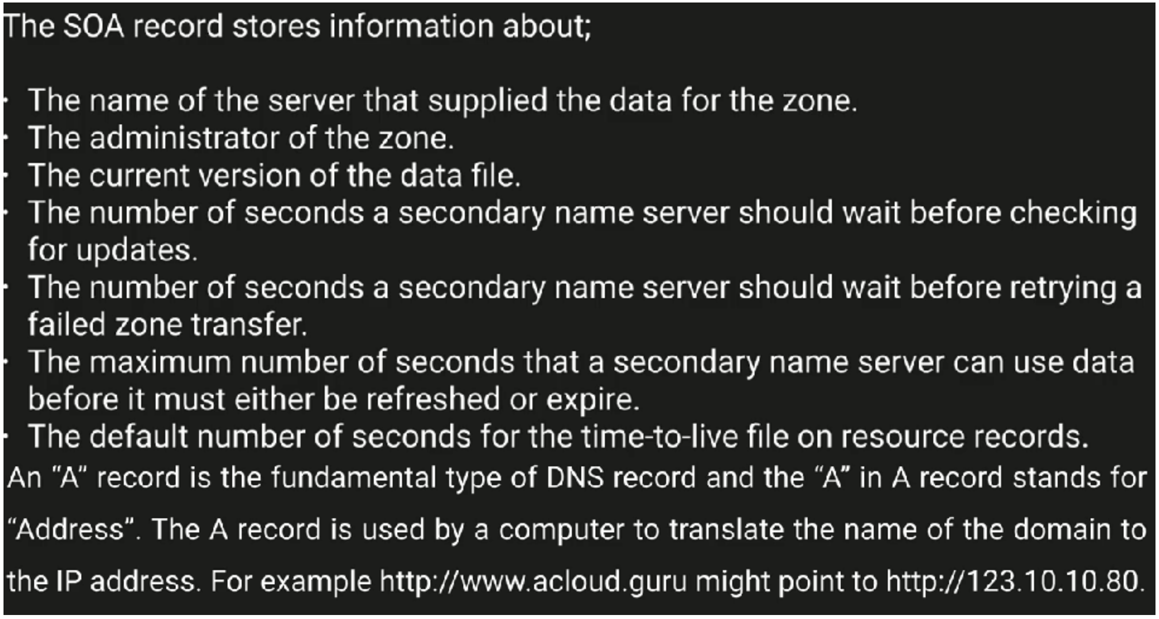
The resource, such as a web server or an email server, that you configure a health check to monitor the health of. You can specify an endpoint by IPv4 address (192.0.2.243), by IPv6 address (2001:0db8:85a3:0000:0000:abcd:0001:2345), or by domain name (example.com).

Note

You can also create health checks that monitor the status of other health checks or that monitor the alarm state of a CloudWatch alarm.

health check

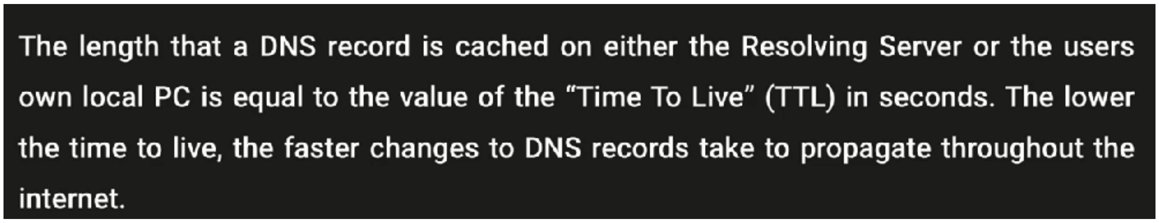
A Route 53 component that lets you do the following:
Monitor whether a specified endpoint, such as a web server, is healthy
Optionally, get notified when an endpoint becomes unhealthy
Optionally, configure DNS failover, which allows you to reroute internet traffic from an unhealthy resource to a healthy resource



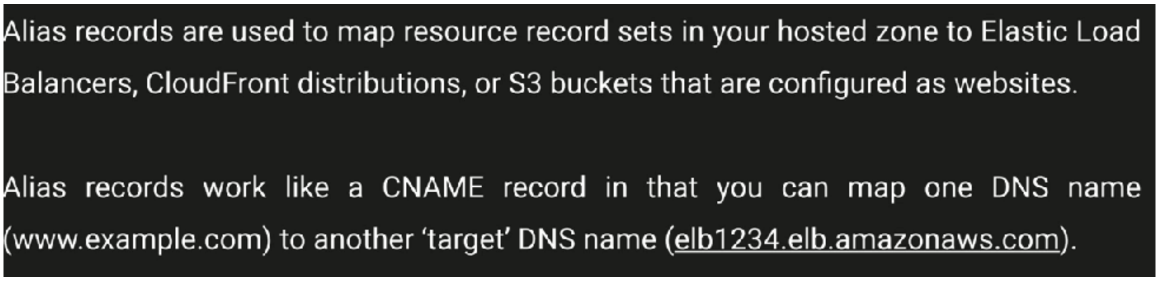
The SOA record stores information about;

- The name of the server that supplied the data for the zone.
- The administrator of the zone.
- The current version of the data file.
- The number of seconds a secondary name server should wait before checking for updates.
- The number of seconds a secondary name server should wait before retrying a failed zone transfer.
- The maximum number of seconds that a secondary name server can use data before it must either be refreshed or expire.
- The default number of seconds for the time-to-live file on resource records.

An "A" record is the fundamental type of DNS record and the "A" in A record stands for "Address". The A record is used by a computer to translate the name of the domain to the IP address. For example http://www.acloud.guru might point to http://123.10.10.80.



The length that a DNS record is cached on either the Resolving Server or the users own local PC is equal to the value of the "Time To Live" (TTL) in seconds. The lower the time to live, the faster changes to DNS records take to propagate throughout the internet.



Alias records are used to map resource record sets in your hosted zone to Elastic Load Balancers, CloudFront distributions, or S3 buckets that are configured as websites.

Alias records work like a CNAME record in that you can map one DNS name (www.example.com) to another 'target' DNS name (elb1234.elb.amazonaws.com).

SQS (Simple Queuing Service)

● Amazon Simple Queue Service (Amazon SQS) is a web service that gives you access to a message queue that can be used to store messages while waiting for a computer to process it. Very first service found in 2004 before aws was officially launched.

● SQS is a pull based and not push based service.

● Messages kept in queue can be from 1-14days,by default its 4days and the size of messages can be of 256KB in size.

●Visibility time out is the amount of time that the message is invisible in the SQS queue after the reader picks up that message.

●Visibility time out is maximum 12hours.

●SQS gurantees atleast your messages processed atleast once.

●SQS has standard queues and FIFO queues.

●SQS has short polling and long polling (while long polling is a way to retrieve messages and short polling is a way to return messages immediately).

SNS (Simple Notification services)

●It is a web service that makes it easy to set up operate and send notifications from cloud.It is a push based service.

● It provides highly scalable, flexible and cost effective capability to publish messages from an application and immediately deliver them to subscribers or other applications.

● It allows you to group multiple recipients using topics. A topic is an “access point” for allowing recipients to dynamically subscribe for identical copies of the same notification.

● SNS lets you send push notifications to mobile apps, text messages to mobile phone numbers, and plain-text emails to email addresses. You can fan out messages with a topic, or publish to mobile endpoints directly.

SES (Simple Email Service)

●Amazon Simple Email Service enables you to send and receive email using a reliable and scalable email platform.

●It is an email platform that provides an easy, cost-effective way for you to send and receive email using your own email addresses and domains.For example, you can send marketing emails such as special offers, transactional emails such as order confirmations, and other types of correspondence such as newsletters. When you use Amazon SES to receive mail, you can develop software solutions such as email autoresponders, email unsubscribesystems, and applications that generate customer support tickets from incoming emails.

Building a large-scale email solution is often a complex and costly challenge for a business.You must deal with infrastructure challenges such as email server management, network configuration,and IP address reputation. Additionally, many third-party email solutions require contract and price negotiations, as well as significant up-front costs. Amazon SES eliminates these challenges and enables you to benefit from the years of experience and sophisticated email infrastructure Amazon.com has built to serve its own large-scale customer base.

SWF(Simple Workflow Service)

● This service helps you to co-ordinate tasks across distributed applications components.

- It’s a workflow service for building scalable, resilient applications. With Amazon SWF, you can build many kinds of applications as workflows. Amazon SWF maintains the execution state of the workflow consistently and reliably so that you can focus on building and running your application.
- SWF enables the application for a range of use cases, including media processing web application backends, business process workflows, and analytics pipeline, to be designed as a coordinate of tasks.
- Amazon SWF helps developers build, run, and scale background jobs that have parallel or sequential steps.

Benefits

Logical Separation
Reliable
Simple
Scalable
Flexible

Lambda

- AWS Lambda is a compute service that lets you run code without provisioning or managing servers.
- AWS Lambda is an ideal compute platform for many application scenarios, provided that you can write your application code in languages supported by AWS Lambda, and run within the AWS Lambda standard runtime environment and resources provided by Lambda.
- You can also build server less applications composed of functions that are triggered by events and automatically deploy them using CodePipeline and AWS CodeBuild.
- Hence it is called as SERVERLESS approach service.
- AWS Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring and logging. All you need to do is supply your code in one of the languages that AWS Lambda supports.
Programming languages used in Lambda are:PHP,Python,JAVA,C#,RUBY,GO,Powershell etc.
- You pay only for the compute time you consume - there is no charge when your code is not running.
- In short lambda is a service to run your code, all you need is to supply the code.
- First 1st million request are free post which you will be charged $0.20-$0.25/month.
- Duration is calculated from the time your code begins till it returns an lambda function.

Elastic Beanstalk:

Amazon Web Services (AWS) comprises over one hundred services, each of which exposes an area of functionality. While the variety of services offers flexibility for how you want to manage your AWS infrastructure, it can be challenging to figure out which services to use and how to provision them.

With Elastic Beanstalk, you can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications.

Elastic Beanstalk reduces management complexity without restricting choice or control.
You simply upload your application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.
Elastic Beanstalk supports applications developed in Go, Java, .NET, Node.js, PHP, Python, and Ruby,Go and Docker on familiar servers such as Apache,Nginx,Passenger and IIS.
When you deploy your application, Elastic Beanstalk builds the selected supported platform version and provisions one or more AWS resources, such as Amazon EC2 instances, to run your application.
You don’t have to create Auto scale,ELB,security group etc.
No additional charge for Elastic Beanstalk-You only pay for AWS resources needed to store and run your applications.
Need not spend time in manging and configuring servers and databases,firewalls,load balancers,and Networks.

THE WELL FRAMED ARCHITECTURE:

FIVE PILLARS:

1.SECURITY

2.RELIABILITY

3.COST OPTIMIZATION

4.PERFORMANCE EFFICIENCY

5.OPERATIONAL EXCELLENCE