JYOTI NIVAS COLLEGE

POST GRADUATE CENTRE

DEPARTMENT OF MCA

TECH-ON-TAP(eJOURNAL)

AMAZON WEB SERVICES

By,
(6th sem MCA)
Syeda Saniya Kayeenath
Deepika N
Lavanya C
Amina SN

Issue 8
April 2022

CONTENTS

AWS Global Infrastructure

The AWS Global Infrastructure is designed and built to deliver a flexible, reliable, scalable, and secure cloud computing environment with high-quality global network performance. suppose you need to deploy your application workloads across the globe in a single click, or you want to build and deploy specific applications closer to your end-users with single-digit millisecond latency, AWS provides you the cloud infrastructure where and when you need it. With millions of active customers and tens of thousands of partners globally, AWS has the largest and most dynamic ecosystem. Customers across virtually every industry and of every size, including start-ups, enterprises, and public sector organizations, are running every imaginable use case on AWS.

AWS Region

The AWS Cloud infrastructure is built around Regions. AWS has 22 Regions worldwide. An AWS Region is a physical geographical location with one or more Availability Zones. **Availability Zones** in turn consist of one or more data canters. To achieve fault tolerance and stability, Regions are isolated from one another. Resources in one Region are not automatically replicated to other Regions. When you store data in a specific Region, it is not replicated outside that Region. It is your responsibility to replicate data across Regions if your business needs require it. AWS Regions that were introduced before March 20, 2019 are enabled by default. Regions that were introduced after March 20, 2019—such as Asia Pacific (Hong Kong) and Middle East (Bahrain)—are disabled by default. You must enable these Regions before you can use them. You can use the AWS Management Console to enable or disable a Region. Some Regions have restricted access. An Amazon AWS (China) account provides access to the Beijing and Ningxia Regions only. The isolated AWS GovCloud (US) Region is designed to allow US government agencies and customers to move sensitive workloads into the cloud by addressing their specific regulatory and compliance requirements. For accessibility: Snapshot from the infrastructure. Aws website that shows a picture of downtown London including the Tower Bridge and the Shard. It notes that there are three Availability Zones in the London region.

Selecting a region

There are a few factors that you should consider when you select the optimal Region or Regions where you store data and use AWS services. One essential consideration is data governance and legal requirements. Local laws might require that certain information be kept within geographical boundaries. Such laws might restrict the Regions where you can offer content or services. For example, consider the European Union (EU) Data Protection Directive. All else being equal, it is generally desirable to run your applications and store your data in a region that is as close as possible to the user and systems that will access them. This will help you reduce latency.

Availability zones

Each AWS Region has multiple, isolated locations that are known as availability Zones. Each Availability Zone provides the ability to operate applications and databases that are more highly available, fault-tolerant, and scalable than would be possible with a single data center. Each Availability Zone can include multiple data centers (typically three), and at full scale, they can include hundreds of thousands of servers. They are fully isolated partitions of the AWS Global Infrastructure. Availability Zones have their own power infrastructure, and they are physically separated by many kilometers from other Availability Zones—though all Availability Zones are within 100 km of each other. All Availability Zones are interconnected with high-bandwidth, low-latency networking over fully redundant, dedicated fiber that provides high throughput between Availability Zones. The network accomplishes synchronous replication between Availability Zones. Availability Zones help build highly available applications. When an application is partitioned across Availability Zones, companies are better isolated and protected from issues such as lightning, tornadoes, earthquakes, and more. You are responsible for selecting the Availability Zones where your systems will reside. Systems can span multiple Availability Zones. AWS recommends replicating across Availability Zones for resiliency. You should design your systems to survive the temporary prolonged failure of an Availability Zone if a disaster occurs.

Data centre

The foundation for the AWS infrastructure is the data centers. Customers do not specify a data centre for the deployment of resources. Instead, an Availability Zone is the most granular level of specification that a customer can make. However, a data centre is a location where the actual data resides. Amazon operates state-of-the-art, highly available data centers. Although rare, failures can occur that affect the availability of instances in the same location. If you host all your instances in a single location that is affected by such a failure, none of your instances will be available. Data centers are securely designed with several factors in mind: Each location is carefully evaluated to mitigate environmental risk. Data centers have a redundant design that anticipates and tolerates failure while maintaining service levels. To ensure availability, critical system components are backed up across multiple Availability Zones. To ensure capacity, AWS continuously monitors service usage to deploy infrastructure to support availability commitments and requirements. Data centre locations are not disclosed and all access to them is restricted. In case of failure, automated processes move data traffic away from the affected area. AWS uses custom network equipment sourced from multiple original device manufacturers (ODMs). ODMs design and manufacture products based on specifications from a second company. The second company then rebrands the product for sale.

Point of Presence

Amazon CloudFront is a content delivery network (CDN) used to distribute content to end users to reduce latency. Amazon Route 53 is a Domain Name System (DNS) service. Requests going to either one of these services will be routed to the nearest edge location automatically in order to lower latency.

AWS Points of Presence are located in most of the major cities around the world. By continuously measuring internet connectivity, performance, and computing to find the best way to route requests, the Points of Presence deliver a better near real-time user experience.
They are used by many AWS services, including Amazon CloudFront, Amazon Route 53, AWS Shield, and AWS Web Application Firewall (AWS WAF) services regional edge caches are used by default with Amazon CloudFront. Regional edge caches are used when you have content that is not accessed frequently enough to remain in an edge location. Regional edge caches absorb this content and provide an alternative to that content having to be fetched from the origin server.

Identity and Access Management

AWS Identity and Access Management (IAM) is a service that enables you to manage AWS users and their permission levels. In an organization with multiple AWS users, it becomes crucial to manage their access levels. IAM allows you to manage users without creating a separate AWS account for each user. Using IAM, users can do the following:

• Manage user access levels based on their organization roles (for example, admin, dev)

• Manage bills in a single account

• Restrict users to access resources in a single region

• Restrict access to AWS services.

IAM User and groups

When a new employee joins an organization, he/she will require access to AWS services. The task will be to adding a new user to the AWS account. A user can be identified using an Amazon Resource Name (ARN) or a unique identifier (used for AWS API calls). Access keys for the users must be created. These access keys are not the same as the AWS master account key. A group is a collection of users having similar responsibilities. Groups are used to propagate permissions to users. For example, it will be challenging to assign permissions to every user joining the organization. Permissions can be assigned to a group and new users can be added to the group. One can create groups called “admin” and “developer” and specify the rights and privileges of these groups. After that, the user can be simply added to these groups based on their job profile.

IAM Role

Roles in AWS are used to categorize, authorize, and authenticate the resources based on their purpose. Instead of assigning permissions and policies for a user or group, users can provide permissions to an AWS resource through the role. For example, one can allow an EC2 instance full access to S3 buckets. The permissions assigned to a role are not tied to a user or group. Therefore, users do not need to distribute or share access keys. Users can also use roles when they need an application to access an AWS resource, but do not want to store the access key in the application itself. At runtime, the AWS service assumes the role, and AWS returns temporary security credentials to the user.

AWS Auto Scaling and monitoring

Elastic Load Balancer

Modern high-traffic websites must serve hundreds of thousands—if not millions—of concurrent requests from users or clients, and then return the correct text, images, video, or application data in a fast and reliable manner. Additional servers are generally required to meet these high volumes.Elastic Load Balancing is an AWS service that distributes incoming application or network traffic across multiple targets—such as Amazon Elastic Compute Cloud (Amazon EC2) instances, containers, internet protocol (IP) addresses, and Lambda functions—in a single Availability Zone or across multiple Availability Zones.

Elastic Load Balancing scales your load balancer as traffic to your application changes over time. It can automatically scale to most workloads.

Elastic Load Balancing is available in three types:

•An **Application Load Balancer** operates at the application level (Open Systems Interconnection, or OSI, model layer 7). It routes traffic to targets—Amazon Elastic Compute Cloud (Amazon EC2) instances, containers, Internet Protocol (IP) addresses, and Lambda functions—based on the content of the request. It is ideal for advanced load balancing of Hypertext Transfer Protocol (HTTP) and Secure HTTP (HTTPS) traffic. An Application Load Balancer provides advanced request routing that is targeted at delivery of modern application architectures, including microservices and container-based applications. An Application Load Balancer simplifies and improves the security of your application by ensuring that the latest Secure Sockets Layer/Transport Layer Security (SSL/TLS) ciphers and protocols are used at all times.

•A **Network Load Balancer** operates at the network transport level (OSI model layer 4), routing connections to targets—EC2instances, microservices, and containers—based on IP protocol data. It works well for load balancing both Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) traffic. A Network Load Balancer is capable of handling millions of requests per second while maintaining ultra-low latencies. A Network Load Balancer is optimized to handle sudden and volatile network traffic patterns.

•A **Classic Load Balancer** provides basic load balancing across multiple EC2instances, and it operates at both the application level and network transport level. A Classic Load Balancer supports the load balancing of applications that use HTTP, HTTPS, TCP, and SSL. The Classic Load Balancer is an older implementation. When possible, AWS recommends that you use a dedicated Application Load Balancer or Network Load Balancer.

How does Load Balancer work?

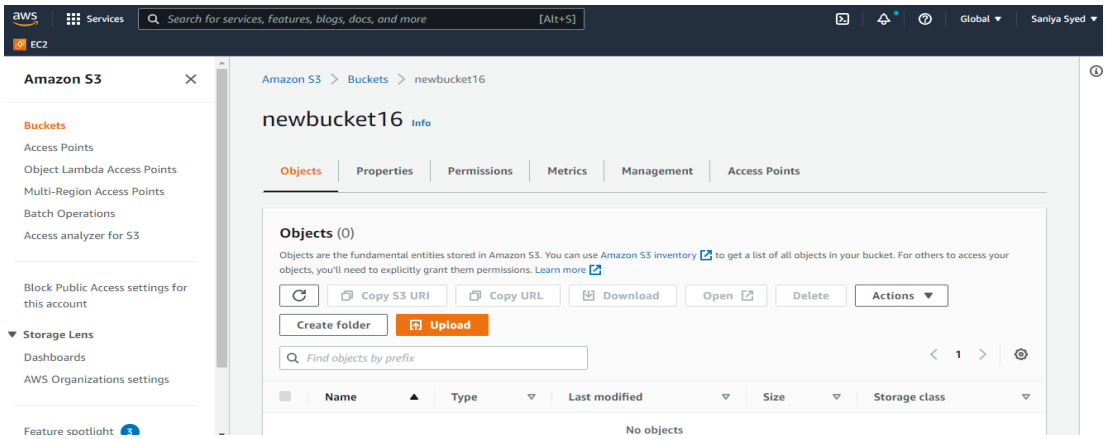
A load balancer accepts incoming traffic from clients and routes requests to its registered targets (such as EC2 instances) in one or more Availability Zones. You configure your load balancer to accept incoming traffic by specifying one or more listeners. A listener is a process that checks for connection requests. It is configured with a protocol and port number for connections from clients to the load balancer. Similarly, it is configured with a protocol and port number for connections from the load balancer to the targets. You can also configure your load balancer to perform health checks, which are used to monitor the health of the registered targets so that the load balancer only sends requests to the healthy instances. When the load balancer detects an unhealthy target, it stops routing traffic to that target. It then resumes routing traffic to that target when it detects that the target is healthy again. There is a key difference in how the load balancer types are configured. With Application Load Balancers and Network Load Balancers, you register targets in target groups and route traffic to the target groups. With Classic Load Balancers, you register instances with the load balancer.

Load Balance Monitoring

You can use the following features to monitor your load balancers, analyse traffic patterns, and troubleshoot issues with your load balancers and targets:•Amazon CloudWatch metrics–Elastic Load Balancing publishes data points to Amazon CloudWatch for your load balancers and your targets. CloudWatch enables you to retrieve statistics about those data points as an ordered set of time-series data, known as metrics. You can use metrics to verify that your system is performing as expected. For example, you can create a CloudWatch alarm to monitor a specified metric and initiate an action (such as sending a notification to an email address) if the metric goes outside what you consider an acceptable range. •Access logs–You can use access logs to capture detailed information about the requests that were made to your load balancer and store them as log files in Amazon Simple Storage Service (Amazon S3). You can use these access logs to analyse traffic patterns and to troubleshoot issues with your targets or backend applications.•AWS CloudTrail logs–You can use AWS CloudTrail to capture detailed information about the calls that were made to the Elastic Load Balancing application programming interface (API) and store them as log files in Amazon S3. You can use these CloudTrail logs to determine who made the call, what calls were made when the call was made, the source IP address of where the call came from, and so on.

Important Services of AWS

AWS S3:



S3 stands for simple storage service. It is used to store your data in the form of objects. It is addressed as the infinitely scaling storage. So the basic idea behind S3 is that it allows the people to store objects (files) in buckets (directories). The bucket that is the container for the objects should have a globally unique name that is no other bucket in the entire aws should have a same name as the bucket you created. The buckets are defined on a regional level. The buckets have different naming conventions such as no uppercase characters must be used, underscores are not allowed, they must be 3-36 characters long, the name of the bucket should not be the name of the IP address, must start with a lowercase character or a number.

The maximum object size a bucket can have is 5 TB(5000 GB). If we upload more than 5 GB we must make use of a multipart upload.

Features of S3:

Versioning:

A bucket in AWS can be versioned, the benefit of versioning your bucket is that it will protect your data to be deleted unintentionally also we can roll back easily to the previous version.

Encryption:

There are four methods by which you can encrypt the objects in S3:
Server Side Encryption- S3: Encrypts s3 objects using keys handled managed by AWS.
Server Side Encryption- KMS: AWS Key Management Service to manage encryption keys.
Server Side Encryption- Client: When the client wants to manage his/her own keys.
Client Side Encryption: Keys as well as encryption is managed by the client.

S3 Bucket Policies:

A bucket policy is a resource-based AWS Identity and Access Management (IAM) policy. You add a bucket policy to a bucket to grant other AWS accounts or IAM users access permissions

for the bucket and the objects in it. Object permissions apply only to the objects that the bucket owner creates.

Access Control:

By default all public access in blocked in S3. This feature was introduced by the AWS to prevent data leaks. Once you disable the public access option your bucket becomes publically accessible over the internet by anyone. Also you can customise this setting by specifying to allow only desired users to access the objects by using bucket policies.

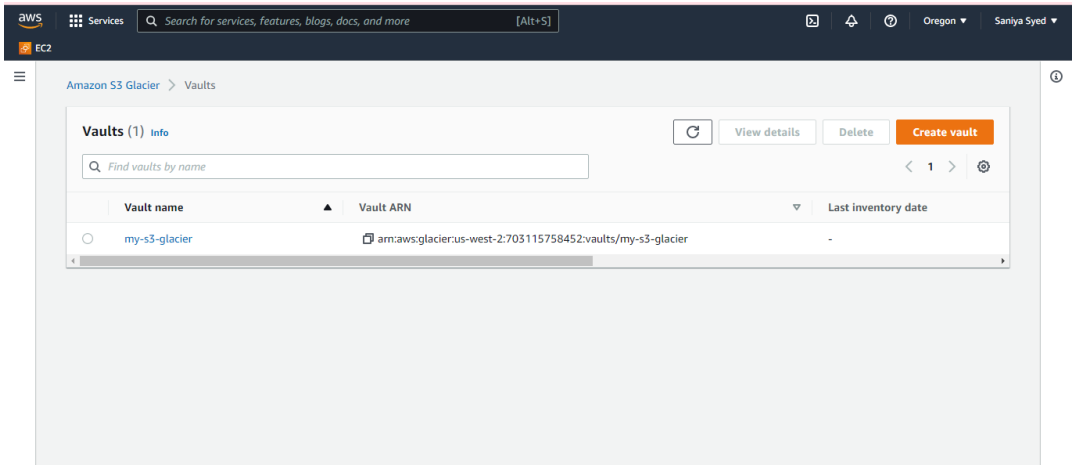
Replication:

Replication enables automatic, asynchronous copying of objects across Amazon S3 buckets. Buckets that are configured for object replication can be owned by the same AWS account or by different accounts. You can replicate objects to a single destination bucket or to multiple destination buckets. The destination buckets can be in different AWS Regions or within the same Region as the source bucket.

Pricing:

First 50 TB / Month. $0.023 per GB. Next 450 TB / Month. $0.022 per GB.

AWS S3 Glacier:



With S3 Glacier, you can store your data cost effectively for months, years, or even decades. S3 Glacier helps you offload the administrative burdens of operating and scaling storage to AWS, so you don't have to worry about capacity planning, hardware provisioning, data replication, hardware failure detection and recovery, or time-consuming hardware migrations.

Amazon Simple Storage Service (Amazon S3) also provides three Amazon S3 Glacier archive storage classes. These storage classes are designed for different access patterns and storage duration. These storage classes differ as follows:

S3 Glacier Instant Retrieval – Used for archiving data that is rarely accessed and requires milliseconds retrieval.

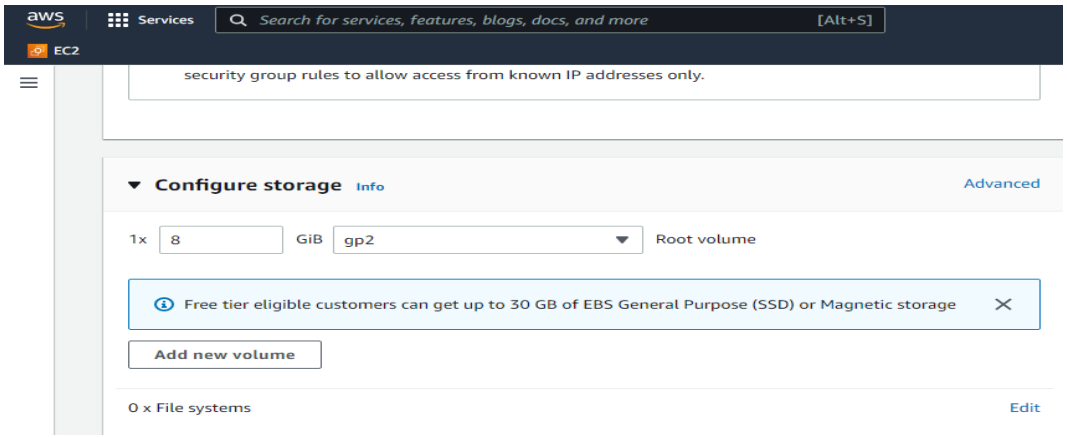
S3 Glacier Flexible Retrieval (formerly the S3 Glacier storage class) – Used for archives where portions of the data might need to be retrieved in minutes. Data stored in the S3 Glacier Flexible Retrieval storage class can be accessed in as little as 1-5 minutes by using Expedited retrieval. You can also request free Bulk retrievals in up to 5-12 hours.

S3 Glacier Deep Archive – Used for archiving data that rarely needs to be accessed. Data stored in the S3 Glacier Deep Archive storage class has a default retrieval time of 12 hours.

Pricing:

Amazon S3 Glacier provides low-cost, long-term storage. Starting at $0.004 per GB per month, Amazon S3 Glacier allows you to archive large amounts of data at a very low cost. You pay only for what you need, with no minimum commitments or upfront fees.

AWS Elastic Block Storage:



Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with EC2 instances. EBS volumes behave like raw, unformatted block devices. You can mount these volumes as devices on your instances. EBS volumes that are attached to an instance are exposed as storage volumes that persist independently from the life of the instance. You can create a file system on top of these volumes, or use them in any way you would use a block device (such as a hard drive). You can dynamically change the configuration of a volume attached to an instance. With Amazon EBS, you pay only for what you use.

Types of EBS Volumes:

General Purpose SSD volumes (gp2 and gp3): They balance price and performance for a wide variety of transactional workloads. These volumes are ideal for use cases such as boot volumes, medium-size single instance databases, and development and test environments.

Provisioned IOPS SSD volumes (io1 and io2): They are designed to meet the needs of I/O-intensive workloads that are sensitive to storage performance and consistency. They provide a consistent IOPS rate that you specify when you create the volume. This enables you to predictably scale to tens of thousands of IOPS per instance. Additionally, io2 volumes provide the highest levels of volume durability.

Throughput Optimized HDD volumes (st1) : They provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. These volumes are ideal for large, sequential workloads such as Amazon EMR, ETL, data warehouses, and log processing.

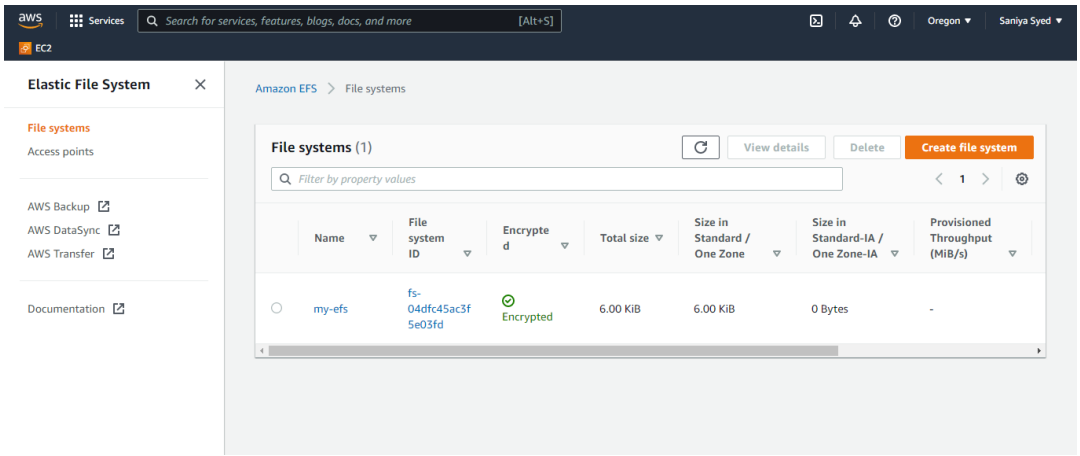
Cold HDD volumes (sc1) : They provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. These volumes are ideal for large, sequential, cold-data workloads. If you require infrequent access to your data and are looking to save costs, these volumes provides inexpensive block storage.

You can create point-in-time snapshots of EBS volumes, which are persisted to Amazon S3. Snapshots protect data for long-term durability, and they can be used as the starting point for new EBS volumes. The same snapshot can be used to create as many volumes as needed. These snapshots can be copied across AWS Regions.

Pricing:

Standard. $0.05/GB-month. Archive. $0.0125/GB-month.

AWS Elastic File Storage:



Amazon Elastic File System (Amazon EFS) provides a simple, serverless, set-and-forget elastic file system for use with AWS Cloud services and on-premises resources. It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth. 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, meaning that you can avoid the complexity of deploying, patching, and maintaining complex file system configurations. Amazon EFS supports the Network File System version 4 (NFSv4.1 and NFSv4.0) protocol.

Types of EFS:

Standard storage classes – EFS Standard and EFS Standard–Infrequent Access (Standard–IA), which offer multi-AZ resilience and the highest levels of durability and availability.

One Zone storage classes – EFS One Zone and EFS One Zone–Infrequent Access (EFS One Zone–IA), which offer customers the choice of additional savings by choosing to save their data in a single Availability Zone.

With Amazon EFS, you can choose from two performance modes and two throughput modes:

General Purpose performance mode : It is ideal for latency-sensitive use cases, like web serving environments, content management systems, home directories, and general file serving.

Max I/O mode: It can scale to higher levels of aggregate throughput and operations per second with a tradeoff of higher latencies for file system operations. For more information, see Performance modes.

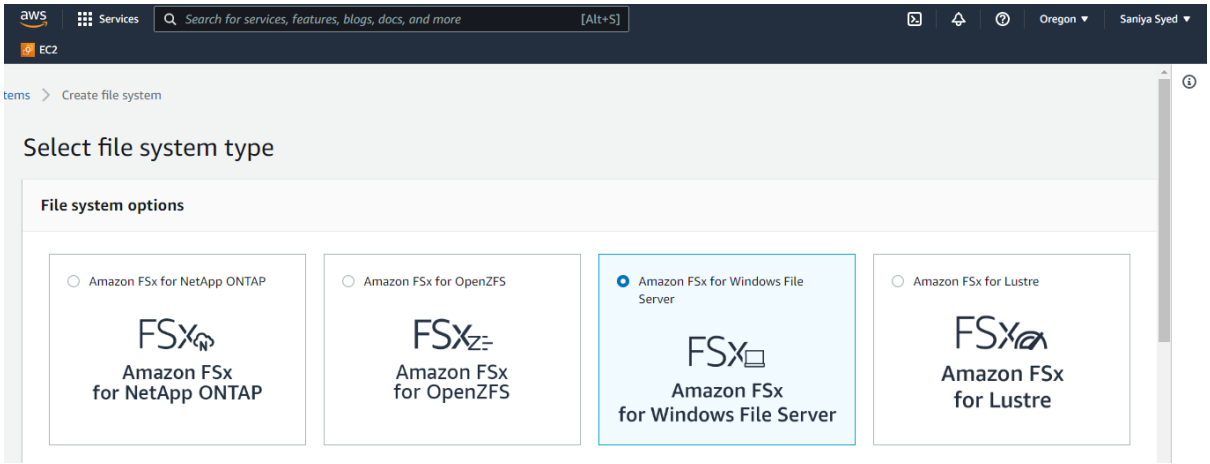
Bursting Throughput mode**:** In this the throughput scales as your file system grows.

Provisioned Throughput mode**:** In this you can specify the throughput of your file system independent of the amount of data stored.

The service is designed to be highly scalable, highly available, and highly durable. Amazon EFS file systems using Standard storage classes 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 compute instances to your data.

Pricing: Within your first 12 months on AWS, you can use up to 5 GB/month on the EFS Standard storage class for free.

AWS FSX for Windows:



Amazon FSx for Windows File Server provides fully managed Microsoft Windows file servers, backed by a fully native Windows file system. FSx for Windows File Server has the features,

performance, and compatibility to easily lift and shift enterprise applications to the AWS Cloud.

Amazon FSx supports a broad set of enterprise Windows workloads with fully managed file storage built on Microsoft Windows Server. Amazon FSx has native support for Windows file system features and for the industry-standard Server Message Block (SMB) protocol to access file storage over a network. Amazon FSx is optimized for enterprise applications in the AWS Cloud, with native Windows compatibility, enterprise performance and features, and consistent sub-millisecond latencies.

As a fully managed service, FSx for Windows File Server eliminates the administrative overhead of setting up and provisioning file servers and storage volumes. Additionally, Amazon FSx keeps Windows software up to date, detects and addresses hardware failures, and performs backups.

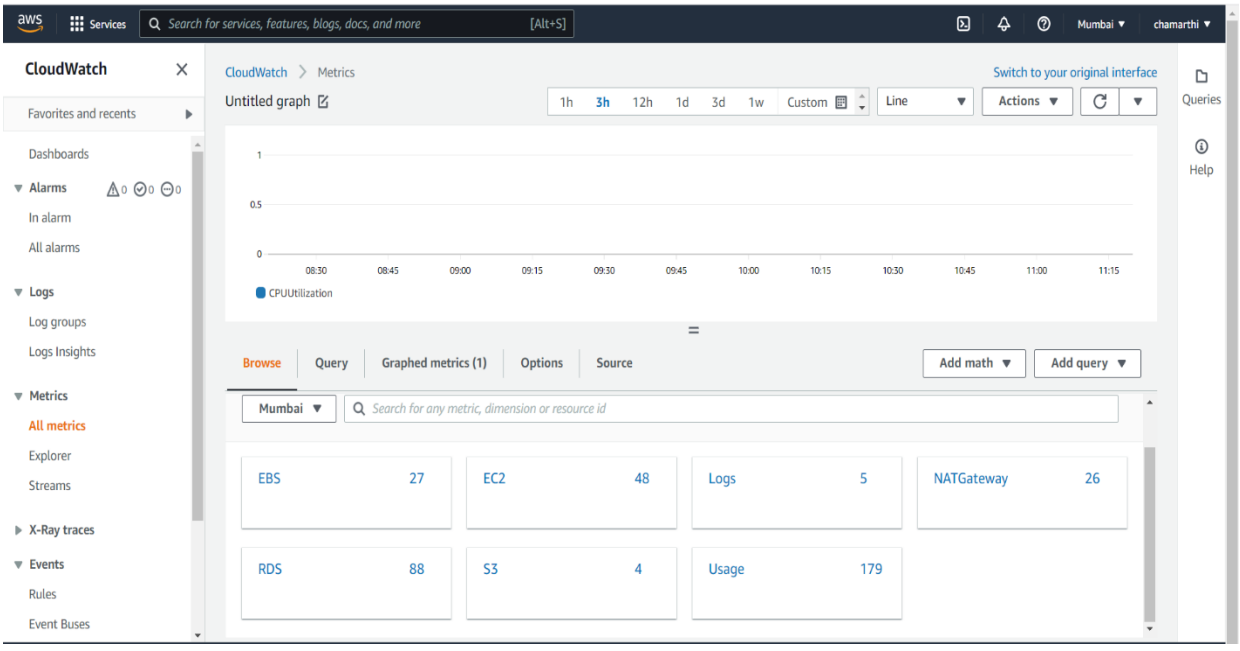
Pricing: With Amazon FSx, there are no upfront hardware or software costs. You pay for only the resources used, with no minimum commitments, setup costs, or additional fees.

CLOUDWATCH

CloudWatch is a monitoring service for AWS cloud resources and the applications. We can use CloudWatch to collect and track metrics, collect, and monitor log files, and set alarms. CloudWatch can monitor AWS resources such as Amazon EC2 instances, Amazon DynamoDB tables, and Amazon RDS DB instances, as well as custom metrics generated by our applications and services, and any log files your applications generate. we can use CloudWatch to the gain system-wide visibility into resource utilization, application performance, and operational health. we can use these kind insights to keep your application running smoothly.

How it Works

CloudWatch collects all the data from user in the form of logs, metrics, and events, and visualizes it using automated dashboards so we can get a unified view of our AWS resources, applications, that run on AWS and on premises. We can visualize the experience of your application end users and validate design choices through experimentation. Correlate your metrics and logs to better understand the health and performance of your resources. We can create alarms based on metric value thresholds you specify, for example, we can an set up automated actions to notify you if the alarm is triggered and automatically start auto scaling to help reduce mean time to resolution. we can also deeply analyse our metrics, logs, and traces to better understand how to improve application performance.

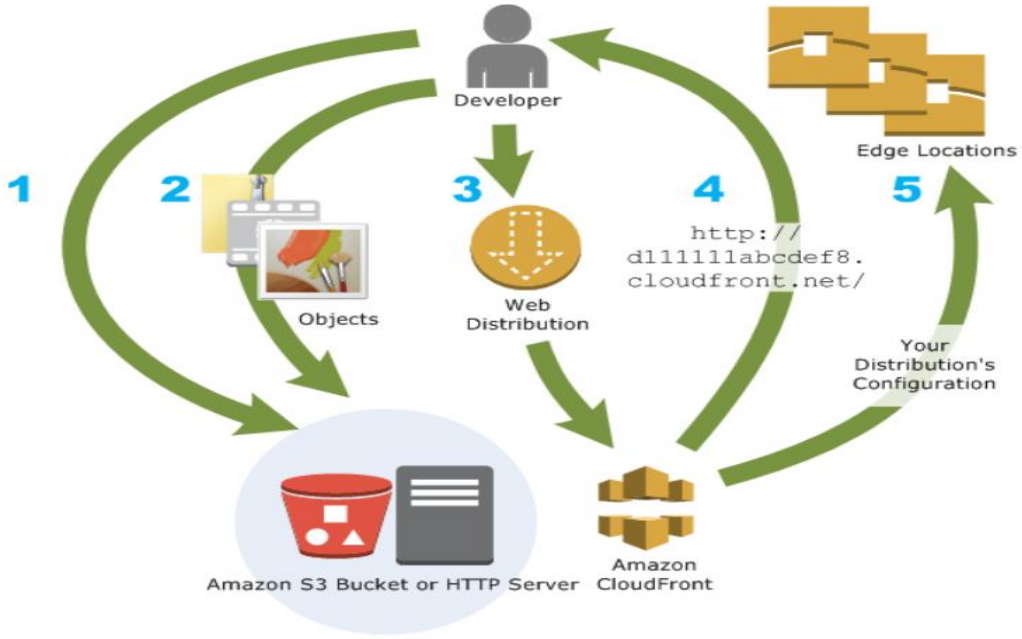


Customers Currently Using this Service

Tech Mahindra, Booking.com, Cognizant, EBSCO, PushPay, Canva, Panasonic, Just Eat and Pro Quest etc.

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 centres called edge locations. When a user requests content that you are serving with CloudFront, the request is routed to the edge location that provides the lowest latency so that content is delivered with the best possible performance.



How it Works

When a viewer makes a request on your website or through your application, DNS routes the request to the POP that can best serve the user's request. This location is typically the nearest CloudFront edge location in terms of latency. In the POP, CloudFront checks its cache for the requested object. If the object is in the cache, CloudFront returns it to the user. If the object is not in the cache, the POP typically goes to the nearest regional edge cache to fetch it.

Accessing CloudFront

AWS Management Console – The procedures throughout this guide explain how to use the AWS Management Console to perform tasks.

AWS SDKs – If you are using a programming language that AWS provides an SDK for, you can use an SDK to access CloudFront.

Customers Currently Using this Service

Amazon, Hulu, PBS, Sky News, Jio Saavan, NED Media Works, Future Sports Media, Discovery Communications etc.

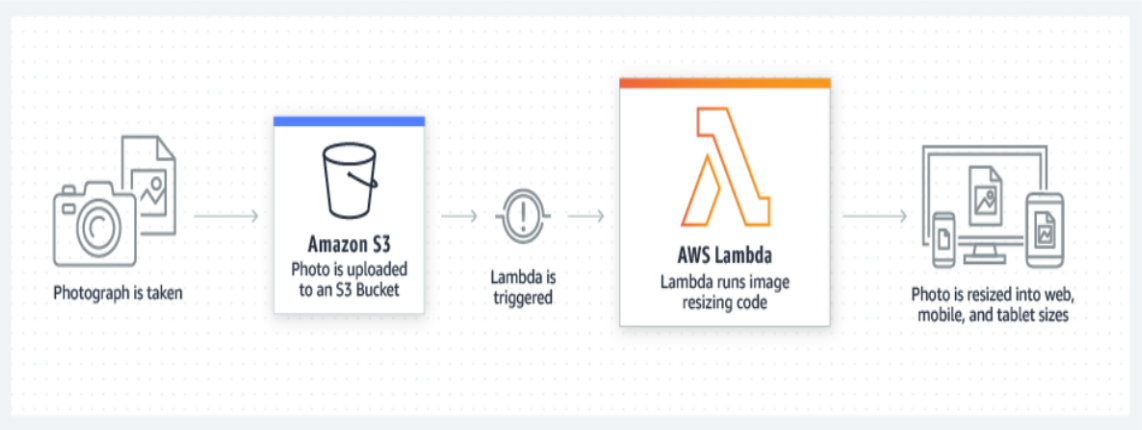
LAMBDA

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service all with zero administration. Just upload your code, and Lambda takes care of everything required to run and scale your code with high availability. we can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.

Serverless Computing allows you to build and run applications and services without thinking about servers. With serverless computing, your application still runs on servers, but all the server management is done by AWS. At the core of serverless computing is AWS Lambda, which lets you run your code without provisioning or managing servers.

How it Works

AWS Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers. Use Amazon Simple Storage Service (Amazon S3) to trigger AWS Lambda data processing in real time after an upload or connect to an existing Amazon EFS file system to enable massively parallel shared access for large-scale file processing.



Customers Currently Using this Service

Fender, Nielsen, Stedi, The Coco cola Company etc.

ELASTICACHE

Amazon ElastiCache is a fully managed, in-memory caching service supporting flexible, real-time use cases. You can use ElastiCache for caching, which accelerates application and database performance, or as a primary data store for use cases that do not require durability like session stores, gaming leaderboards, streaming, and analytics. ElastiCache is compatible with Redis and Memcached. ElastiCache is compatible with Redis and Memcached to support flexible scaling for demanding real-time applications. With ElastiCache, you pay only for what you use with no minimum fee. You are charged hourly based on the number of nodes, node type, and pricing model you select.

How it Works

Amazon ElastiCache is a fully managed, in-memory caching service supporting flexible, real-time use cases. You can use ElastiCache for caching which accelerates application and database performance, or as a primary data store for use cases that do not require durability like session stores, gaming leaderboards, streaming, and analytics. ElastiCache is compatible with Redis and Memcached.

Use Cases:

Increase Application Performance: Access data with microsecond latency and high throughput for lightning-fast application performance.

Ease backend database load: Cache your data to reduce pressure on your backend database, enabling higher application scalability and reducing operational burden.

Build low-latency data stores: Use ElastiCache to store non-durable datasets in memory and support real-time applications with microsecond latency.

Customers Currently Using this Service

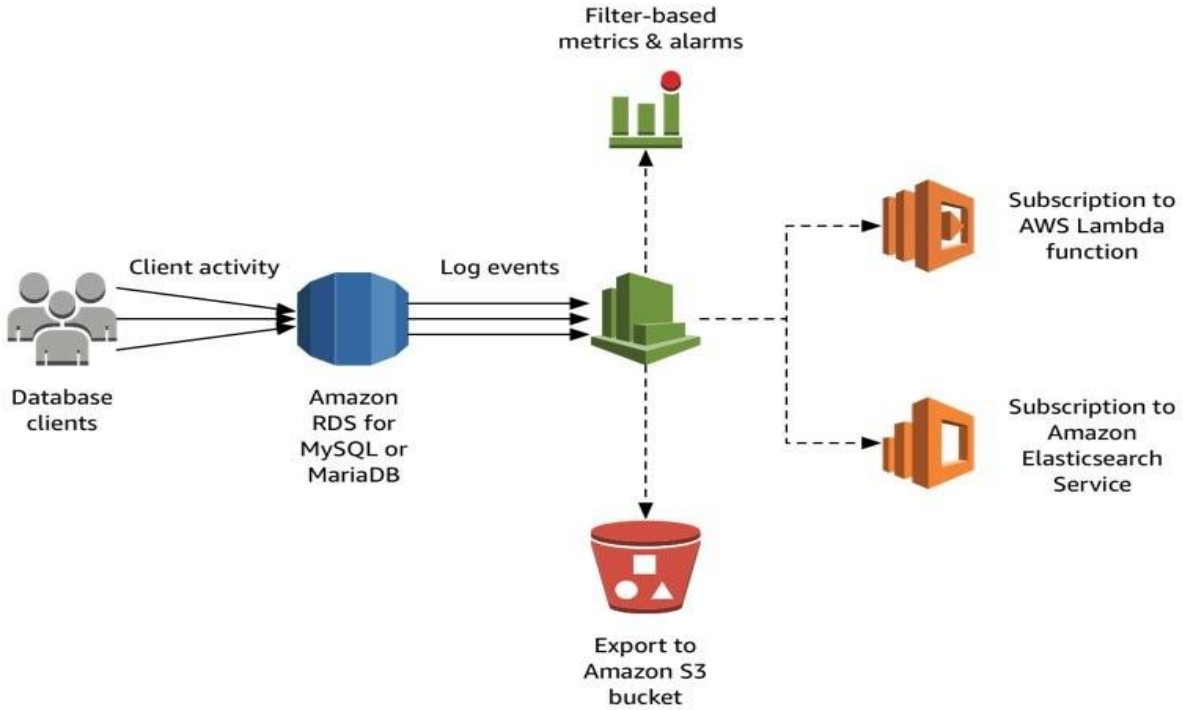
The Pokeman Company, A+E Networks, Tinder

RDS (Relational Database Service)

Amazon RDS gives you access to the capabilities of a familiar MySQL, MariaDB, Oracle, SQL Server, or PostgreSQL database. This means that the code, applications, and tools you already use today with your existing databases should work seamlessly with Amazon RDS. Amazon RDS can automatically back up your database and keep your database software up to date with the latest version. You benefit from the flexibility of being able to easily scale the compute resources or storage capacity associated with your relational database instance. In addition, Amazon RDS makes it easy to use replication to enhance database availability, improve data durability, or scale beyond the capacity constraints of a single database instance for read-heavy database workloads. As with all Amazon Web Services, there are no up-front investments required, and you pay only for the resources you use.

Amazon EC2 Relational Database AMIs

Amazon Web Services provides several database alternatives for developers. Amazon RDS enables you to run a fully featured relational database while offloading database administration. Using one of our many relational database AMIs on Amazon EC2 allows you to manage your own relational database in the cloud. There are important differences between these alternatives that may make one more appropriate for your use case. See Cloud Databases with AWS for guidance on which solution is best for you.



Use Cases

Build web and mobile applications: Support growing apps with high availability, throughput, and storage scalability. Take advantage of flexible pay-per-use pricing to suit various application usage pattern

Move to managed databases: Innovate and build new apps with Amazon RDS instead of worrying about self-managing your databases, which can be time consuming, complex, and expensive.

Customers Currently Using this Service

Mint, Samsung, Cathay Pacific, Oracle etc.

Amazon VPC

Amazon web service offers you a service – Amazon Virtual Private Cloud (abbreviated as Amazon VPC) helps you to launch your AWS recourses in an isolated virtual network that you simply a user has defined. The network is simply virtual representation of your traditional network that you would operate in your own data centre.

Components of VPC:

VPC: The VPC is simply virtual representation of your traditional network that you would operate in your own data centre.

Subnet: Subnet may be a subset of your VPC which must reside in a single Availability Zone. One VPC can have multiple subnets.

IP Addressing: There are two versions of IP Addressing -IPV4 and IPV6. Your VPC should be assigned with an IP address and your Subnets IP address must reside within your VPC range.

Routing: Your subnets contains Route Tables which helps you to route the traffic should be directed.

NAT Devices

A NAT gateway may be a Network Address Translation (NAT) service. You'll utilise a NAT Gateway so that cases during a private subnet can associate with administrations outside your VPC however outer administrations can’t start an association with those instances. At the purpose when you create a NAT gateway, you indicate one among the connectivity types; • Public - (Default) Instances privately subnets can associate with the web through a public NAT gateway, yet can't get spontaneous inbound associations from the online .You create a public NAT gateway during a public subnet and should relate a flexible IP address with the NAT passage at creation. You route traffic from the NAT gateway to the web gateway for the

VPC, but you can utilise a public NAT gateway to connect with other VPCs or your on-premises organisation.

For this example you route traffic from the NAT gateway through a transit gateway or a virtual private gateway.

• Private - Instances privately subnets can associate with other VPCs or your on-premises network through a private NAT gateway. You'll direct traffic from the NAT gateway through a transit gateway or a virtual private gateway. You cannot associate a elastic IP address with a private NAT gateway. You can attach a internet gateway to a VPC with a personal NAT gateway, however within the event that you route traffic from the private NAT gateway to the web gateway, the web gateway drops the traffic. The NAT gateway replaces the source IP address of the instances with the IP address of the NAT gateway. For a public NAT gateway, this is often the elastic IP address of the NAT gateway. For a personal NAT gateway, this is the private IP address of the NAT gateway.

VPC Peering

A VPC peering connection may be a networking connection between two VPCs which helps you to route traffic between different VPCs using private IPv4 addresses or IPv6 addresses. Instances in either VPC can communicate with one another as if they are within the same network. You'll create a VPC peering connection between your own VPCs, or with a VPC in several AWS account. The VPCs are often in other regions (also known as an inter-region VPC peering connection).

AWS uses the already present infrastructure of a VPC to make your VPC peering connection; it is neither a gateway nor a VPN connection and is not dependent on a separate piece of physical hardware. There's no single point of failure for communication or a bandwidth bottleneck.

A VPC peering connection helps you to with the transfer of Data.

Simple Queue Service (SQS)

Amazon Simple Queue Service (Abbreviated as Amazon SQS) provides you with a highly secure, durable, and available hosted queue which allows you to integrate and decouple distributed system. Amazon SQS provides you with several features like dead-letter queues and cost allocation tags. Dead Letter Queue: DLQ isn't another type of queue but a queue you configure with your existing main queue which helps you processes your unprocessed request in your existing queue. All the processes attempt to process until several configured time, once they are not processes until given time then it is moved to the Dead Letter Queue.

Types of Queue:

1. Standard Queue: when the processes are entered within the queue it will try and maintain the order but due to the processing there are high possibilities for the processes to not deliver in the same order as entered. Standard queue mainly concentrates on massive and highly distributed architecture.
2. FIFO (First In First Out) Queue: because the name defines it mainly concentrates on sequence of the requests by providing each incoming request with Message Group ID and Deduplication ID. It follows exactly once delivery.

SNS (Simple Notification Service)

Amazon Simple Notification Service (Amazon SNS) is a serverless service that enables message delivery through SMS from your publishers to subscribers. A publisher creates a topic that helps him to communicate asynchronously with subscribers by sending messages to a *topic*. Customers can subscribe to the SNS topic and receive published messages using a supported endpoint type, such as Amazon Kinesis Data Firehose, Amazon SQS, AWS Lambda, HTTP, email, mobile push notifications, and mobile text messages (SMS)