Amazon Web Services (AWS)

AWS Global Infrastructure

Region

A region is a geographical area with at least two availability zones in it, and each region is isolated from other AWS Regions.

Availability Zones

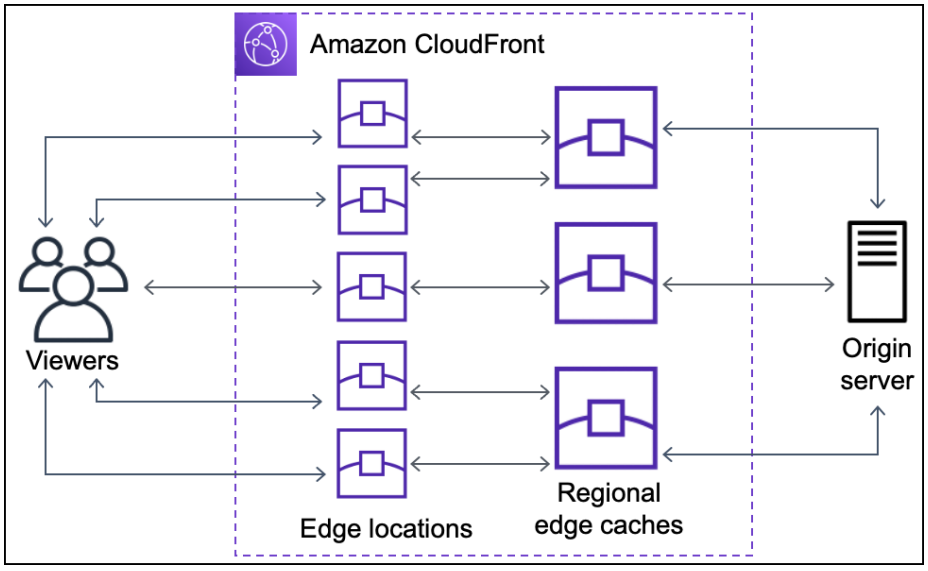
Availability zones are isolated data centers within a region, connected by low-latency, highly available, and high-bandwidth links. Deploy resources to separate availability zones for high availability and redundancy.

Local Zones

A Local Zone deploys compute, storage, databases, and other AWS services closer to large populations. Each AWS Local Zone location is an extension of a Region where you can run your latency-sensitive applications using AWS services.

Edge location and Regional edge caches

These locations are closer to your users, so the idea is to reduce latency when accessing cached content. Edge location is a cache of content that can be delivered at low latency to users. Regional edge caches are larger caches that sit between AWS services and edge locations. These are used by CloudFront. There are more edge locations than regions.



Security, Identity and Compliance

Factors of authentication — “factors” include something you know (e.g., password), something you have (e.g., token device), and something you are (e.g., retina or fingerprint scan).

Identity Access and Management (IAM)

It enables creation of multiple users and management of their permission within the AWS account. IAM user is an entity that represents a person or service. Every IAM user starts with no permissions. In other words, by default, users can do nothing, not even view their own access keys. To give a user permission to do something, you can add the permission to the user (that is, attach a policy to the user). Or, you can add the user to a group that has the intended permission. IAM is global and not specific to any one region.

IAM supports Multi-factor authentication (MFA). Through the AWS Management Console — the user is prompted for a username, password, and authentication code. Using the AWS API — restrictions are added to IAM policies, and developers can request temporary security credentials and pass MFA parameters in their AWS STS API requests. Using the AWS CLI — temporary security credentials can be obtained from STS (aws sts get-session-token).

IAM supports identity federation, which allows users with passwords elsewhere to get temporary access to an AWS account. Use IAM roles service for this purpose. It is a good practice to attach policies to a group and add users to the group. Every resource in AWS is accessible via the API, which is subject to IAM. However, if we have an EC2 instance with an elastic IP address hosting a web server, calls to that server do not touch any AWS-provided APIs and this is not subject to IAM policies. When the same web server requests DynamoDB, it goes through an AWS API.

IAM Role

Roles are assumed by trusted entities and can be used for delegation. Roles are essentially used by services and AWS. You can delegate permissions to users and services using roles without using permanent credentials (e-g username and password). Security credentials for roles are created dynamically. User who assumes a role, gives up his own permissions and uses the permissions of a role. IAM users or AWS services can temporarily assume a role.

Instance Profiles

IAM roles can be used for granting applications running on EC2 instances permission to AWS API requests using instance profiles. Only one role can be assigned to an EC2 instance at a time. You cannot add multiple IAM roles to a single EC2 instance.

Role delegation

- Create an IAM role with two policies:

    - Permissions policy — grants the user of the role the required permissions on a resource.
    - Trust policy — specifies the trusted accounts that are allowed to assume the role. Useful when resources can be accessed across accounts.
- Wildcards (*) cannot be specified as a principal.
- A permissions policy must also be attached to the user in the trusted account.

IAM Groups

A group is a collection of users and has policies attached to them. A group is not an identity and can not be used as a Principal in an IAM policy. This means they cannot directly own resources or perform actions in AWS. You can not nest groups. It can be used to assign permission to users.

Principals

Principal can be a user, role, aws services, federated users that performs actions on resources as long as they are authorized to do so.

Policy

An IAM policy is a JSON document with strictly defined structure. It can be attached to an IAM user, IAM group, and IAM role. It contains statements, which are building blocks of access control. Each statement can contain an Effect, an Actions, a Principal, a Resource, and a Condition parameter, depending on which entity a policy is attached to. Each of these parameters have a corresponding Not version e-g NotResource, NotPolicy. It is advisable to use Not parameters for “Deny” actions. You cannot store credentials in IAM policies.

IAM Permissions Boundary

IAM permissions boundary can be used on the developer IAM role that explicitly denies attaching the specific policy e-g a role may be allowed to access all S3 buckets but permission boundary can deny access to specific bucket for that role. It can override explicit allow but cannot override explicit deny. It only established a boundary within which you can give permissions, it does not give permissions itself.

AWSAuthenticationPlugin

Using AWSAuthenticationPlugin, and associate an IAM user account in the MySQL database.” is a great way to securely authenticate to RDS using IAM users or roles. It can also be used to provide access with short-lived credentials (MySQL can not be used directly with AWS STS).

IAM Query API

AWS recommends that you use the AWS SDKs to make programmatic API calls to IAM. However, you can also use the IAM Query API to make direct calls to the IAM web service. An access key ID and secret access key must be used for authentication when using the Query API.

Consider IAM Query API for:

- Custom integrations not directly supported by SDKs.
- Specific performance optimizations or advanced control requirements.

Attribute-based access control (ABAC)

Users and resources in AWS can be tagged with attributes. ABAC is an alternative strategy to Role-Based Access Control (RBAC). In RBAC, users are grouped based on job functions instead of individual users, while in ABAC, we assign tags to users and resources and allow an operation when matched.

Use ABAC in Highly dynamic environments, cross-organizational access, fine-grained control, compliance requirements based on attributes, and RBAC in Clear role definitions, structured access hierarchies, and less frequent access changes.

Identity Based Policy (IBP)

- Used with: users, groups, and roles.
- Purpose: Define what actions an identity (user, group, or role) is allowed or denied to perform on any resource in the same AWS account.
- Benefits:
    - Simple to manage: Attach the policy directly to the identity.
    - Centralized control: Manage permissions for all resources for an identity in one place.

Resource Based Policy (RBP)

- Used for: any AWS resource (not just for cross-account access).
- Purpose: Define what actions any identity is allowed or denied to perform on that specific resource, regardless of their other policies.
- Benefits:
    - Granular control: Restrict access for specific resources on a case-by-case basis.
    - Isolation: Secure sensitive resources even if the associated identity has broader permissions.

Request Evaluation Flow

- **First, a request context is built**. This context contains information about the request, such as the principal (who is making the request), the action (what the request is trying to do), and the resource (the object that the request is targeting).
- **Next, all the policies that match the request are collected**. This includes both identity-based policies and resource-based policies. When no policy matches the request, it will be allowed.

- **If there is an explicit deny for the request in any policy, the request is denied.** This means that even if other policies allow the request, it will still be denied if there is an explicit deny statement.
- **A request is allowed if any one of resource-based or identity-based policy allows it.** This means that if a resource-based policy allows the request, it will not be denied even if an identity-based policy denies it. Resource-based policy takes precedence over identity-based policies.
- **There are other types of optional policies against which a request can be evaluated.** These policies include IAM permission boundaries and session policies. These policies are only evaluated to explicitly deny a request. If they do not deny the request, it will move forward to be evaluated against resource and identity based policies.

Request is denied if there is at least one explicit deny or none of the policies allows it (implicit deny).

Avoid using AWS account credentials for accessing AWS services instead, use IAM user credentials.

User Accounts

Use it when:

- Multiple people need permanent access.
- One or more users require CLI access.
- Enhanced accountability and auditing.

Root Account

Root access tasks include modifying the root user, changing the AWS support plan, closing an AWS account.

IAM Access Analyzer

IAM Access Analyzer helps identify the required permissions for the IAM execution role (used to perform actions on other services). IAM Access Analyzer reviews your AWS CloudTrail logs over the date range that you specify and generates a policy template with only the permissions that the function used during that time.

Pricing

Using IAM is free.

Limitations

- We can have up to 5000 users per AWS account.
- Each user account has a friendly name and ARN.

- Access Key ID and Secret Access Key are not the same as password and cannot be used to login to the AWS console. These can only be generated once and regenerated if lost.
- User names can have path prefixes, but there are limitations on their structure and depth.

Compute / VMs

Amazon EC2

Introduction

It allows you to host virtual machines. Operating System (OS) and application software, configuration settings are bundled into Amazon Machine Image (AMI). EC2 instances can be deployed in one or more geographic regions.

By default, EC2 instances are placed in default VPC. Any resource inside the default VPC is publicly available.

Shared tenancy is the default instance hosting method. More than one customer's instance may be on the same physical machine at the same time.

Instances offer at least 99.99% (four nines) of availability.

Amazon EC2 gives you complete control over the instance, down to the root level. You can manage the instance as you would manage a physical server.

Debugging

- Key pairs are used to SSH into the instance. The public keys (stored by AWS) and private keys (stored by the recipient) are known as a key pair.
- We can also use EC2 instance connect using console, for running commands in EC2 instance.

User Data

Normally, we provide a script as **user data** for the instance. Script pulls application codebase from S3 bucket, installs packages, creates environment variables and runs the application.

You can pass two types of user data to Amazon EC2:

1. shell scripts.
2. cloud-init directives.

User data is limited to 16 KB. Instance user data is available at http://169.254.169.254/latest/user-data.The IP address 169.254.169.254 is a link-local address and is valid only from the instance.

Metadata

Instance metadata (e-g instance ID, public keys, network interfaces, and etc.) is data about your instance that you can use to configure or manage the running instance. It is available at http://169.254.169.254/latest/meta-data/. You can also use Instance Metadata Query Tool for downloading metadata.

AMIs

AMI includes the following:

- A template for the root volume for the instance (for example, an operating system, an application server, and applications).
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it is launched.

AMIs are regional. You can only launch an AMI from the region in which it is stored. However, you can copy AMIs to other regions using either the console, the command line, or the API.

AMIs are either instance-store backed or EBS backed.

Launch Template

Launch templates enable you to store launch parameters so that you do not have to specify them every time you launch an instance. When you launch an instance using the Amazon EC2 console, an AWS SDK, or a command-line tool, you can specify the launch template to use. You must ensure that your launch template includes all parameters required to launch an EC2 instance such as AMI ID and instance type otherwise, you'll get an error **'use fully formed launch template'**.

An IAM user or role that creates an Auto Scaling Group using a launch template must have permission to use the **EC2:RunInstances** action and permission to use the resources for the instance.

Storage

Each instance that you launch has an associated root device volume, either an Amazon EBS volume or an instance store volume. You can use block device mapping to specify additional EBS volumes or instance store volumes to attach to an instance when it's launched. You can also attach additional EBS volumes to a running instance.

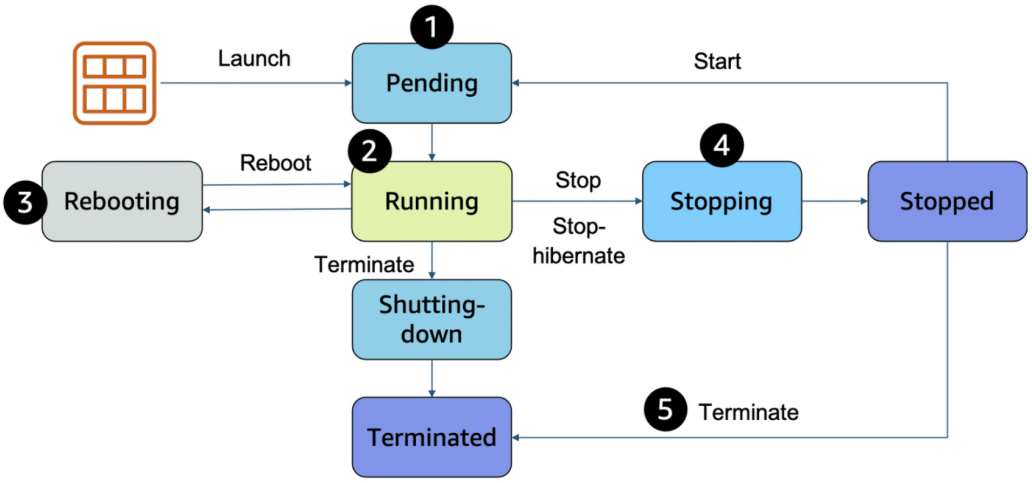
Instance Types

EC2 instance types name contains two pieces of information i) EC2 Instance Type; ii) Instance Size e-g in t3.medium t3 is instance type (where t is instance family and 3 is the generation of the instance) while medium is the instance size.

1. Each instance family is defined for different use cases for example, the c instance family is for compute optimization.
2. Hardware capabilities and resources are identified by instance type.
3. Instance size determines instance capacity (CPU, memory, and storage).

Instance Family	**Description**	**Use Cases**
General Purpose (m,t)	Provides balance of compute, memory, networking resources.	For applications that use these resources in equal proportions e-g web servers and code repositories.
Compute Optimized (c)	Contains high performance processors.	Well suited for batch processing workloads, media transcoding, high performance computing, scientific workloads.
Memory Optimized (r,x)	For processing large datasets in memory.	High performance databases, in-memory caches and databases, real-time big data analytics.
Accelerated Computing (g)	Use hardware accelerators and coprocessors to add more capabilities than CPUs. It has GPUs.	Machine Learning, HPC, computational fluid dynamics, speech recognition. It's a good option for processing videos.
Storage Optimized (d,i)	Optimized to deliver low-latency random I/O (read and write access) operations per second (IOPS).	NoSQL Databases (Cassandra, MongoDB and Redis).
HPC Optimized (h)	Offer the best price performance for running HPC workloads at scale on AWS.	Deep learning workloads or workloads that require high CPU performance, large memory, fast networking, and low latency

EC2 Instance Lifecycle



- In pending state, billing is not started and aws boots up ec2 instance using the AMI image.
- Instance is ready to use when in running state and billing also begins here.
- Rebooting an instance is equivalent to rebooting an operating system. Instance retains its public DNS names and private and public IP address in addition to any data stored on instance store volumes.
- Instance can stop and start if it has an Amazon EBS volume at its root device. On stopping and starting, an instance can be placed on a new underlying physical server. Public IP address is not retained, AWS releases the public IP address back into the pool of available IP addresses. However, the private IP address is retained and assigned to the network interface. At stop state, data transfer fees are not charged but Amazon EBS volume is still charged. At this state, memory (RAM) is lost and you can modify instance attributes (e-g instance type).
- On stop-hibernate, the instance is stopped but saves the latest information from memory (RAM) to EBS root volume so that the start process is faster.
- On termination, instance stores are erased as well as IP addresses (Public and Private) and you can no longer access the machine. Terminated instances are not billed.

The following are a few reasons why an instance might immediately terminate from pending state:

- You've reached your EBS volume limit.
- An EBS snapshot is corrupt.
- The root EBS volume is encrypted, and you do not have permission to access the KMS key for decryption.
- The instance store-backed AMI that you used to launch the instance is missing a required part (an image.part.xx file).

Security

EC2 security groups can be changed on the fly. There is a limit of 5 SGs per instance. EC2 instances have a default SG that can be detached. Default SG allows SSH connection for Linux instances and RDP (Remote Desktop Protocol) for windows instances.

Pricing

Instances are billed when they're in a running state — they need to stop or terminate to avoid inducing costs. Data transfer from EC2 to the internet is also billed.

Instance Pricing Types

On-Demand
No upfront payments and long-term commitments, you pay for compute capacity (per hour or per second), capacity can be increased or decreased as required. Best for users who prefer low cost and flexibility, applications that can't be interrupted.

Spot
For applications with flexible start and end times, very low compute prices, fault-tolerant and stateless workloads.

You set up a price you want to pay per hour, an instance is available to you if your spot price is more than aws determined (based on supply and demand). Saves up to 90% off the on-demand price. There are no upfront fees.

Spot instances receive a two-minute interruption notice when these instances are about to be reclaimed by EC2 because EC2 needs the capacity back.

- Instances are not interrupted because of higher competing bids.
- To reduce the impact of interruptions and optimize spot instances, diversify and run your application across multiple capacity pools.
- To further reduce the impact of interruptions, you can also set up spot instances and spot fleets to respond to an interruption notice by stopping or hibernating rather than terminating instances when capacity is no longer available.

Reserved
For applications with steady-state usage and requires reserved capacity, saves up to 75% compared to on-Demand pricing. You can select either 1-year or 3-year term commitment. Upfront payments are optional.

- **Standard Reserved:** Up to 72 % off on-demand pricing. A commitment of 1 or 3 years, charged whether it's on or off. Instance family, OS, tenancy, payment options can not be updated.
- **Convertible Reserved:** Up to 54 % off on-demand pricing and you get the capability to change the attributes of the reserved instance (RI) given that the resulting RI is equal or

greater than the value of the original one. Instance family, OS, tenancy, payment options can be updated.
- **Scheduled Reserved:** Available to launch within the reserved time windows. Reserved for specific periods of time; accrue charges hourly; billed in monthly increments over the term.

Dedicated Hosts
A physical Amazon EC2 server that is dedicated for your use. Good option for meeting compliance requirements, you have the ability to manage software licenses. Can reduce cost as you can use your own existing software licenses. Can be purchased on demand (hourly) or can be purchased as a reservation for up to 70% off the on-demand price.

Dedicated Instances
Dedicated instances are virtualized instances on hardware just for you. They also use physically dedicated EC2 servers; however, they do not provide the additional visibility and controls of dedicated hosts (e.g., how instances are placed on a server).
- Available as On-Demand, Reserved Instances, and Spot Instances
- May share hardware with other non-dedicated instances in the same account
- Dedicated instances are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer. Your dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts. Dedicated instances allow automatic instance placement and billing is per instance.
- They may run on a different physical machine on restart but no other customers can use that machine.

Combination of On-Demand and Spot instances can be used to minimize the costs, and On-Demand can be used when Spot Instances aren't available or when price is excessive.

Savings Plans:
Alternative to Reserved Instances offering flexible commitment options for EC2 and other services.

Spot Fleets:
Manage multiple Spot instances as a fleet for diversification and cost optimization.

Errors

InstanceLimitExceeded
If you've reached the limit on the number of instances you can launch in a region, you get an error when you try to launch a new instance or restart a stopped instance.

UnsupportedOperation
It is possible that an AMI does not support an instance type and cause a client error.

Run Command

Run Command is designed to support a wide range of enterprise scenarios, including installing software, running ad hoc scripts or Microsoft PowerShell commands, configuring Windows Update settings, and more.

It can be used to execute the script of all target EC2 instances (for running a PowerShell script on a fleet of EC2 instances running MS Windows).

It is a specific feature within the broader SSM service.

RAID (Redundant Array of Independent Disks)

It is a data storage technology that combines multiple physical disk drives into a single logical unit for data redundancy, performance improvement, or a combination of both. RAID can be implemented in hardware or software. In the context of EC2 instances, RAID is typically implemented in software.

RAID 0 = 0 striping - data is written across multiple disks and increases performance but no redundancy. If any one disk in a RAID 0 array fails, all of the data on the array is lost.

RAID 1 = 1 mirroring - creates 2 copies of the data but does not increase performance, only redundancy. If any one disk in a RAID 1 array fails, the data on the remaining disks remains intact.

RAID 10 - 10 - combination of RAID 1 and 2, resulting in an increase in performance and redundancy (at the cost of additional disks).

EBS Optimized EC2 Instances

It is another way of increasing performance, however, you need to ensure that the EC2 instance can handle bandwidth required for the increased performance.

IP Addresses

Public
A public address is assigned automatically to instances in public subnets and is reassigned (new) if an instance is stopped/started. AWS does not charge it.

Private
A private address is assigned automatically to all instances. They are retained even when the instance is stopped. It can be associated with public or elastic IP to redirect public traffic to private instances.

Elastic

This is a public address that is static. They are retained even when the instance is stopped. It can be remapped and moved b/w instances. They are region specific, and AWS charges for them even if it's not used.

It is charged when its not attached to any instance, when multiple Elastic IP address are attached to an instance, when it's attached to stopped instance

Elastic Network Interfaces with EC2

Elastic Network Interface (ENI)

It is a logical networking component that represents a virtual network card. It includes the following information (public, private, elastic IP address, MAC address, security group). These can be attached to instances in VPC. By default, eth0 is the only Elastic Network Interface created with an EC2 instance when launched and it cannot be moved or detached. An ENI is AZ specific, you can specify which subnet/AZ you want ENI to be added in. When you add a second interface, AWS will not assign a public IP address to eth0 (you will need to add an Elastic IP). It can be used with all instance types.

ENI can have one public IP address at max and multiple private IP addresses.

Multiple ENIs attached to a single instance allows dual homing (instances exist in multiple subnets). This can happen by using one public IP address and multiple private IP addresses. For example, a router is a dual home device connected to an internet and local network as well.

It is used by an interface endpoint, not a gateway endpoint.

Enhanced Networking - Elastic Network Adapter (ENA)

Enhanced Networking provides higher bandwidth, higher packet-per-second (pps) performance and consistently lower inter-instance latencies. Usually used when pps rate hits the ceiling. ENA is available for all instance types and is only supported in VPC.

- For general-purpose workloads requiring improved network performance without incurring extra cost.
- For web servers, databases, and other applications leveraging standard TCP/IP networking.

Elastic Fabric Adapter (EFA)

EFA is an ENA with added capabilities, it allows applications to access network interfaces without getting OS involved with each message.

EFA is a network interface for EC2 instances that enables customers to run applications requiring high levels of inter-node communications (tightly-coupled applications) at scale. It does so by bypassing the hardware interface.

It is used for HPC applications and Machine Learning applications. It can be used with certain instance types only.

- For demanding HPC and ML workloads where ultra-low latency and high throughput are critical.
- For applications specifically designed to utilize EFA's hardware offloading and OS-bypass capabilities.

Placement Groups

Are a logical grouping of instances.

Cluster
Groups instances into a low-latency and high throughput group in a single AZ. It has finite capacity. A cluster placement group provides low latency and high throughput for instances deployed in a single AZ.

It is for applications that benefit from low network latency, high network throughput, or both and if the majority of the network traffic is between the instances in the group.

Spread
Spreads instances across underlying hardware (can span AZs). A maximum of 7 instances per group per AZ are allowed.

Spread placement groups are recommended for applications that have a small number of critical instances that should be kept separate from each other.

In the spread placement group each instance is placed on a different hardware / rack.

Partition
Divides each group into logical segments called partitions. Each partition is on a separate rack. There can be multiple instances in one partition and can share underlying hardware resources (same rack). It is used for deploying large distributed and replicated workloads across different racks. It is not supported for dedicated hosts. Partitions can be deployed across AZs.

It can have up to 7 partitions per AZ.

Placement groups can’t be merged, an instance can not span multiple placement groups.

Monitoring

EC2 status checks are performed every minute, and each returns a pass or a fail status.

If all checks pass, the overall status of the instance is **OK**. If one or more checks fail, the overall status is **impaired**. These checks are built into Amazon EC2 and can not be disabled however, trigger alarms can be deleted or created.

Notes

1. ELB Source IP: When EC2 instances are behind an ELB, the ELB's IP is seen as the source.
2. Snapshot/AMI Access: Use the EC2 API, not S3 API, for snapshots and EBS-backed AMIs.
3. Cross-Region Snapshots: Enabled using Amazon Data Lifecycle Manager.
4. Instance Encryption: Supported by all EC2 families, but not all instance types.
5. COPY Command: Loads data in parallel from remote hosts via SSH.
6. Remote Access: RDS for Windows desktop access, SSH for CLI access.

Elastic Beanstalk (EB)

Elastic Beanstalk is used to build out entire solutions. They can include multiple instances, databases, VPCs, subnets and more.

It is different from EC2 instances in a way that customers specify the type of application they want to host, EB will automatically provision an environment for your application to work. In EC2, customers configure everything themselves.

AWS Elastic Beanstalk can be used to quickly deploy and manage applications in the AWS Cloud. Developers upload applications, and Elastic Beanstalk handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring. It is considered to be a Platform as a Service (PaaS) solution and allows full control of the underlying resources.

Pricing

There is no additional charge for Elastic Beanstalk – you pay only for the AWS resources needed to store and run your applications.

AWS Lightsail

Launches virtual private servers, which are VMs with individual operating systems but restricted access to physical server resources.

Lightsail vs EC2

EC2 is a more comprehensive and scalable service than Lightsail. It offers a wider range of instance types, networking options, and storage options. EC2 is also more customizable and flexible than Lightsail.

Lightsail is a simpler and more affordable service than EC2. It is a good choice for small businesses and individuals who are new to cloud computing. Lightsail offers a limited number of instance types, networking options, and storage options, but it is still a powerful and flexible service.

Container Services

Container orchestration is a process of managing (starting, stopping, restarting, monitoring, etc.) containers in multiple EC2 instances (EC2 cluster). ECS and EKS are two such services in the AWS ecosystem.

Containers run without the startup latency of Lambda or Amazon EC2.
There are no time-out limits when running. This is useful for applications that run longer than 15 minutes or that need to initiate instantly when called.

Amazon Elastic Container Service (ECS)

Used for managing containers on a cluster of Amazon EC2 instances. An Amazon ECS container agent needs to be installed on your EC2 instances. This container agent is responsible for running containers and communicating with ECS service about cluster management details.

Auto Scaling

Auto Scaling can be implemented for the ECS cluster instances using a capacity provider that is associated with an Auto Scaling Group (ASG).

- ASG manages EC2 instances.
- ECS manages container tasks and can trigger ASG actions for EC2 scaling.
- ECS can also directly scale Fargate tasks without ASG involvement.

Containers vs Tasks

- Containers are building blocks; tasks are blueprints for running them.
- Containers offer isolation and portability; tasks manage container execution on instances.
- Think of containers as ingredients; tasks are recipes for combining them to run an application.
- An ECS cluster can run multiple tasks, each launching its defined container(s) on available instances.
- Scaling in ECS primarily adjusts the number of running tasks (and indirectly the container instances) based on workload requirements.

Permissions

To specify permissions for a specific task on Amazon ECS, you should use IAM Roles for Tasks. The permissions policy can be applied to tasks when creating the task definition or by using an IAM task role override using the AWS CLI or SDKs. The taskRoleArn parameter is used to specify the policy. You should not apply the permissions to the container instance as they will then apply to all tasks running on the instance as well as to the instance itself.

The **AmazonECSTaskExecutionRolePolicy** policy is the Task Execution IAM Role. This is used by the container agent so it can pull container images, write log files, etc.

To grant additional permission to an individual ECS application container on an ECS cluster you need to create a separate task definition for the application container that uses a different Task role without granting additional permissions to other containers running on the cluster. You can only apply one IAM role to a Task Definition, so you must create a separate Task Definition. A Task Definition is required to run Docker containers in Amazon ECS, and you can specify the IAM role (Task Role) that the task should use for permissions.

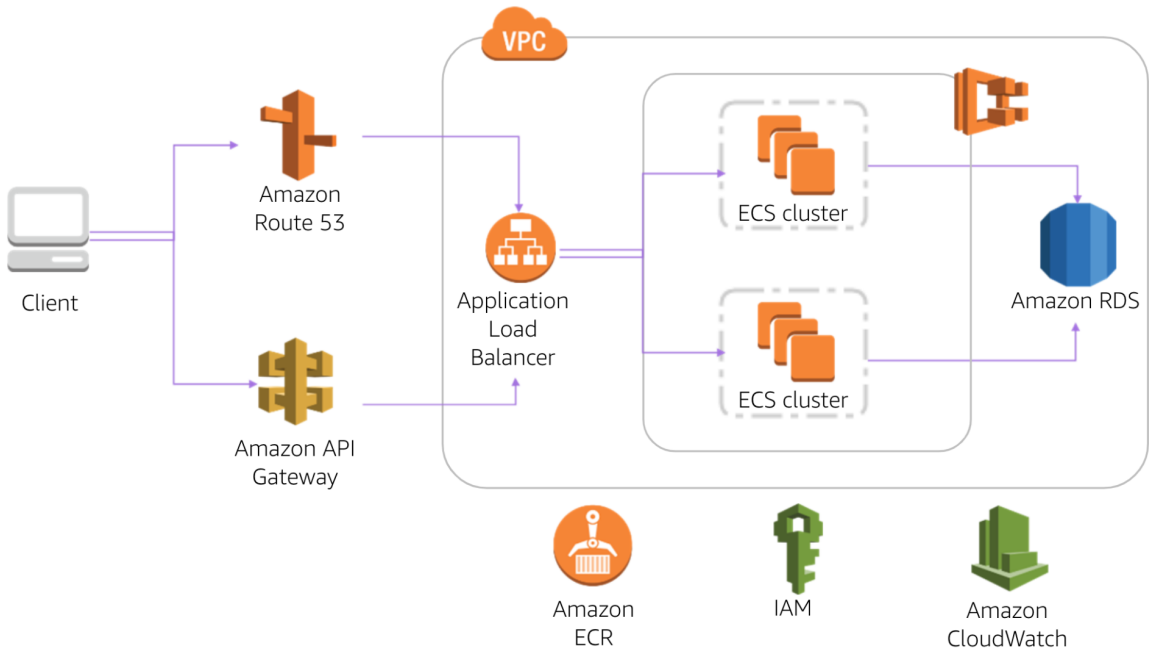
With the EC2 launch type, you can apply IAM roles at the container and task level, whereas with Fargate, you can only apply at the task level.

An instance profile is a container for an IAM role that you can use to pass role information to an EC2 instance when the instance starts.

Dynamic Port Mapping

Dynamic port mapping allows you to run multiple tasks over the same host using multiple random host ports (in spite of a defined host port).

Dynamic port mapping with an Application Load Balancer makes it easier to run multiple tasks on the same Amazon ECS service on an Amazon ECS cluster. With the Classic Load Balancer, you must statically map port numbers on a container instance. The Classic Load Balancer does not allow you to run multiple copies of a task on the same instance because the ports conflict.



Amazon Elastic Kubernetes Service (EKS)

Conceptually similar to ECS.

- In EKS, the machine that runs the containers is called a worker or kubernetes node while in ECS it is called a container instance.
    - In EKS only EC2 instances can be used as worker nodes.
- An ECS container is called a task. An EKS container is called a pod.
- Amazon ECS runs on AWS native technology. Amazon EKS runs on Kubernetes.

Pricing

There is no additional charge for Amazon ECS. You pay for EC2 instances or EBS volumes.

Launch Types Comparison (EC2 vs Fargate)

EC2: More granular control over infrastructure. EFS and EBS integration. Charged per running EC2 instance. You handle cluster optimization. It supports images hosted in private repositories.

Fargate: Limited control, infrastructure is automated. No EFS and EBS integration. Charged for running tasks (vCPU and the memory allocated to the containers you run). Fargate handles cluster optimization. It supports container images hosted on ECR or Docker Hub.

Additional Considerations

- A cluster may contain a mix of tasks hosted on AWS Fargate, Amazon EC2 instances, or external instances.
- Amazon ECS provides service discovery for a microservice architecture.
- ECS is currently designed for AWS environments and doesn't natively support multi-cloud integrations.

- While Amazon ECS doesn't have a built-in container registry, it's designed to integrate seamlessly with secure container registries, ensuring the confidentiality and integrity of your container images.

Use Cases

- Compute-intensive workloads, lambda is not the best fit for a heavily compute-intensive piece of code.
- Large monolithic applications that have many parts.
- When you need to scale quickly as containers can be built and taken down quickly.
- When you need to move your large application to the cloud without altering the code.

When not to use containers

- When applications need persistent data storage; Containers can absolutely support persistent storage; however, if containers are moved, the storage needs to be reconfigured and secured.
- When applications have complex networking, routing or security requirements.

Serverless Compute

Serverless means that users of the service can not see the underlying infrastructure hosting user’s solution.

AWS Fargate

A serverless compute platform for ECS. This is used when the user does not need access to the underlying OS and does not need to manage EC2 instances.

With Fargate, you only pay for compute and memory resources that you specify for your tasks, and you don’t have to pay for idle resources. With EC2, you pay for entire virtual machines, even if you are not using all of its resources. Fargate has a higher per-hour cost for instances than EC2, but it also has a lower minimum charge. Fargate charges you for a minimum of 1 minute charge, while EC2 charges you for a minimum of 1 hour of usage. Similar to EC2, fargate offers spot pricing options. Fargate also supports savings plan (reserved).

AWS Lambda Function

Configuration

You specify the amount of memory you need to be allocated to your Lambda functions. AWS Lambda allocates CPU power proportional to the memory you specify using the same ratio as a general-purpose EC2 instance type.

Deployment

Lambda function code can be deployed in two ways:

1. You deploy code in a .zip file archive.
2. You package your code into a container image.

Execution

Lambda function runs in response to triggers. AWS will scale your lambda functions in response to multiple triggers, each will run in their own containers. Customers get billed for code that is running.

- Continuous scaling — scales out, not up. Lambda scales concurrently, executing functions up to your default limit (1000).
- Lambda functions are serverless and independent; 1 event = 1 function. Functions can trigger other functions, so 1 event can trigger multiple functions.
- Versioning can be used to run different versions of your code.
- Concurrency limits: Control the maximum number of concurrent function executions.

Invocation Models

Synchronous - Lambda runs the function and waits for response. In this model there is no built-in retries, client must manage the retry logic.

The following AWS services invoke Lambda synchronously:

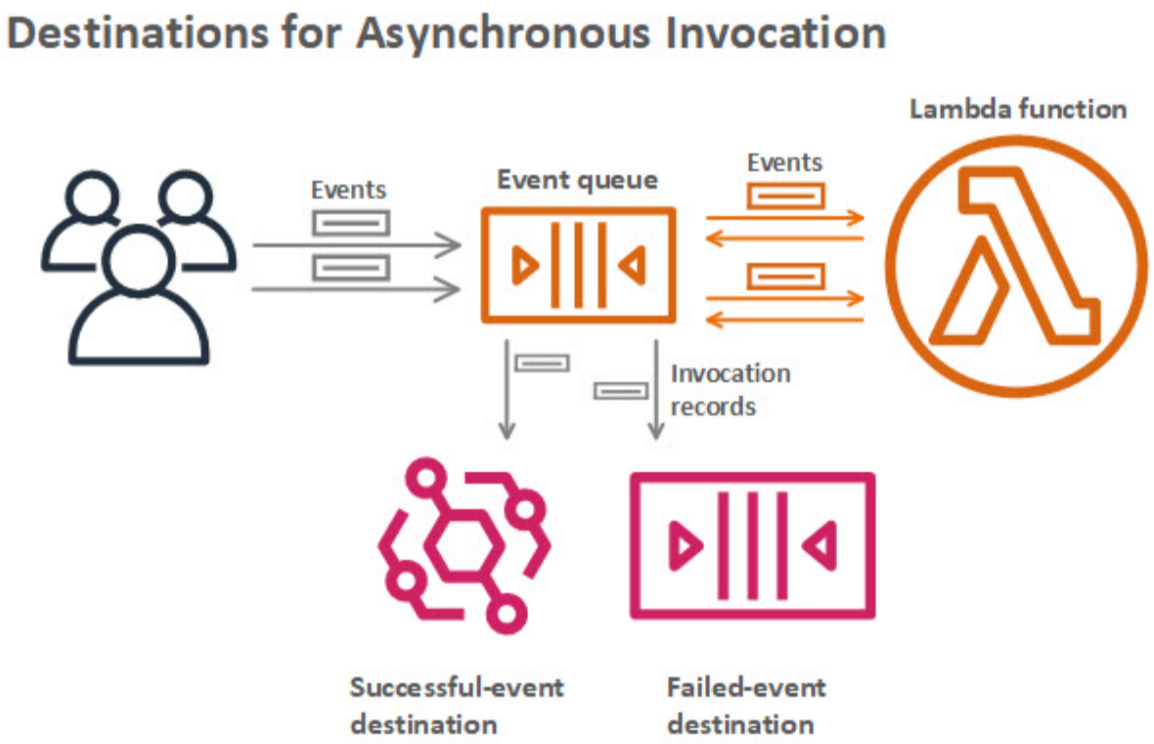
- Amazon API Gateway
- Amazon Cognito
- AWS CloudFormation
- Amazon Alexa
- Amazon Lex
- Amazon CloudFront

Asynchronous - When you invoke a function asynchronously, events are queued and the requestor doesn't wait for the function to complete. This model is appropriate when the client doesn't need an immediate response.

With the asynchronous model, you can make use of destinations. Use destinations to send records of asynchronous invocations to other services.

The following AWS services invoke Lambda asynchronously:

- Amazon SNS
- Amazon S3
- Amazon EventBridge



Polling - This invocation model is designed to integrate with AWS streaming and queuing based services with no code or server management. Lambda will poll (or watch) these services, retrieve any matching events, and invoke your functions. Lambda reads events from the following services:

- Amazon DynamoDB
- Amazon Kinesis
- Amazon MQ
- Amazon Managed Streaming for Apache Kafka (MSK)
- self-managed Apache Kafka
- Amazon SQS

With this type of integration, AWS will manage the poller on your behalf and perform synchronous invocations of your function.
With this model, the retry behavior varies depending on the event source and its configuration.

Cold Start

A cold start occurs when a new execution environment is required to run a Lambda function. When the Lambda service receives a request to run a function, the service first prepares an execution environment. During this step, the service downloads the code for the function, then creates the execution environment with the specified memory, runtime, and configuration. Once complete, Lambda runs any initialization code outside of the event handler before finally running the handler code.

If a function has not been used for some time, if more concurrent invocations are required, or if you update a function, new environments are created. Creation of these environments can introduce latency for the invocations that are routed to a new environment. After optimizing your function, another way to minimize cold starts is to use provisioned concurrency. **Provisioned concurrency** is a Lambda feature that prepares concurrent execution environments before invocations. If you need predictable function start times for your workload, provisioned concurrency ensures the lowest possible latency. This feature keeps your functions initialized and warm, and ready to respond in double-digit milliseconds at the scale you provision. Unlike with on-demand Lambda, this means that all setup activities happen before invocation, including running the initialization code.

You pay for the amount of provisioned concurrency that you configure and for the period of time that you have it configured.

Best practice: Write functions to take advantage of warm starts

- Store and reference dependencies locally.
- Limit re-initialization of variables.
- Add code to check for and reuse existing connections.
- Use tmp space as transient cache.
- Check that background processes have completed.

Concurrency

Reasons for setting a concurrency reserve for a function can include the following:

- Managing cost.
- Matching speed with a downstream resource.
- Regulating how long it takes to process events.

Accessing Resources in VPC

Lambda works globally. Enabling your Lambda function to access resources inside your virtual private cloud (VPC) requires additional VPC-specific configuration information, such as VPC subnet IDs and security group IDs. This functionality allows Lambda to access resources in the VPC. It does not change how the function is secured. You also need an execution role with

permissions to create, describe, and delete elastic network interfaces. Lambda provides a permissions policy for this purpose named "**AWSLambdaVPCAccessExecutionRole**".

When a Lambda function is connected to a VPC, all outbound requests go through your VPC. To connect to the internet, configure your VPC to send outbound traffic from the function's subnet to a NAT gateway in a public subnet.

Lambda and AWS PrivateLink

To establish a private connection between your VPC and Lambda, create an interface VPC endpoint. Interface endpoints are powered by AWS PrivateLink, which enables you to privately access Lambda APIs without an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection.

Instances in your VPC don't need public IP addresses to communicate with Lambda APIs. Traffic between your VPC and Lambda does not leave the AWS network.

Monitoring

AWS Lambda automatically monitors Lambda functions and reports metrics through Amazon CloudWatch. Lambda tracks the number of requests, the latency per request, and the number of requests resulting in an error. You can view the request rates and error rates using the AWS Lambda Console, the CloudWatch console, and other AWS resources.

You can use the Lambda Insights dashboard multi-function overview in the CloudWatch console to identify over and under utilized Lambda functions. You can use the Lambda Insights dashboard single-function view in the CloudWatch console to troubleshoot individual requests.

AWS X-Ray

X-Ray is an AWS service that can be used to detect, analyze, and optimize performance issues with Lambda applications. X-Ray collects metadata from the Lambda service and any upstream and downstream services that make up your application.

You can use AWS X-Ray to visualize the components of your application, identify performance bottlenecks, and troubleshoot requests that resulted in an error. Your Lambda functions send trace data to X-Ray, and X-Ray processes the data to generate a service map and searchable trace summaries. AWS X-Ray records how the Lambda functions are running. Use it to identify the call flow of your Lambda function and the performance of every API call within your application.

Dead Letter Queue

Dead-letter queues help you capture application errors that must receive a response, such as an ecommerce application that processes orders. If an order fails, you cannot ignore that order error. You move that error into the dead-letter queue and manually look at the queue and fix the problems.

- Use dead-letter queues to analyze failures for follow-up or code corrections.

- Dead-letter queues are available for asynchronous and non-stream polling events.
- A dead-letter queue can be an Amazon Simple Notification Service (Amazon SNS) topic or an Amazon Simple Queue Service (Amazon SQS) queue.

Lambda@Edge

Lambda@Edge allows you to run code across AWS locations globally without provisioning or managing servers and responding to end-users at the lowest network latency. It scales with the volume of CloudFront requests globally.

Lambda@Edge lets you run Node.js and Python Lambda functions to customize content that CloudFront delivers, executing the functions in AWS locations closer to the viewer. The functions run in response to CloudFront events without provisioning or managing servers. You can use Lambda functions to change CloudFront requests and responses at the following points:

- After CloudFront receives a request from a viewer (viewer request)
- Before CloudFront forwards the request to the origin (origin request)
- After CloudFront receives the response from the origin (origin response)
- Before CloudFront forwards the response to the viewer (viewer response)

Pricing

- Number of requests: The first 1 million are free, then $0.20 per 1 million is charged.
- Duration: (Calculated from the time your code begins execution until it returns or terminates). Pricing depends on the amount of memory allocated to a function.

Security

To store sensitive information in environment variables of Lambda, you can use encryption helpers that leverage AWS Key Management Service to store that sensitive information as Ciphertext.

Resource policies grant permissions to invoke the function, whereas the execution role strictly controls what the function can do within the other AWS service.

Use Cases

- Fast development
- Pay for value
- Short-lived applications
- Event-driven applications
- Automatic Scaling
- Redundancy and Resilience

Limitations

- Default regional account-level concurrency limit is 1000 (can be increased upon request).

AWS Batch

Batch computing is used for batch processing. Dynamically provisions optimal quantity and type of compute resources (e-g CPU vs memory optimized instances). It also manages the batch process, i.e. re-starting jobs that fail, scheduling jobs, etc.

Multi-node parallel Jobs

AWS Batch multi-node parallel jobs enable you to run single jobs that span multiple Amazon EC2 instances. With AWS Batch multi-node parallel jobs, you can run large-scale, tightly coupled, high-performance computing applications and distributed GPU model training without the need to launch, configure, and manage Amazon EC2 resources directly.

An AWS Batch multi-node parallel job is compatible with any framework that supports IP-based internode communication, such as Apache MXNet, TensorFlow, Caffe2, or Message Passing Interface (MPI).

Storage

File Storage

- Files are organized in a tree-like hierarchy that consist of folders and sub-folders.
- Each file has a metadata (file size, file path etc).
- File is retrieved using the file system hierarchy.
- This type of storage is ideal, when we need to share files with multiple host computers and integrate them with existing file system communication protocols.
- Common use case is when we need to adhere to the standards of communicating with data in files using file system protocols (NFS or SMB).
- Using Amazon EC2 and Amazon EBS, you can quickly create your own high-performance block storage for building your own network file system, including the following protocols and systems:
    - SMB
    - NFS
    - Extents File System (XFS)
    - General Parallel File System (GPFS)
    - Zettabyte File System (ZFS)
    - Other customer files systems
    - You can choose the file system that you need to optimize your applications or workflows. You can bring your media workflows and use their native file system running on EC2 instances and store your data on EBS volumes.

Amazon Elastic File System (Amazon EFS)

- Automatically grows and shrinks as you add and remove files.
- There is no need to provision and manage this system.
- Multiple compute modules can access an Amazon EFS file system at the same time. These modules include Amazon EC2, AWS Lambda, Amazon Elastic Container Service (Amazon ECS), and Amazon Elastic Kubernetes Service (Amazon EKS). Accessing compute services provides a common data source for workloads and applications running on more than one compute instance or container.

Storage Classes

Standard Storage Classes
EFS Standard and EFS Standard–Infrequent Access (Standard–IA), which offer multiple Availability Zones (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 additional savings by choosing to save data in a single-Availability Zone (Single-AZ).

Performance Modes

General Purpose
The default General Purpose performance mode is ideal for latency-sensitive use cases, such as web serving environments, content management systems, home directories, and general file serving.

Max I/O mode
File systems in the Max I/O mode can scale to higher levels of aggregate throughput and IOPS. The tradeoff is slightly higher latencies for file metadata operations.

Suitable for big data analytics, media processing, HPC, and large file transfers.

Bursting Throughput Mode
Using the default Bursting Throughput mode, throughput scales as your file system grows.

Provisioned Throughput Mode
Using Provisioned Throughput mode, you can specify the throughput of your file system independent of the amount of data stored.

High Availability and durability

Amazon EFS is designed to be highly available and is designed for 99.999999999 percent (11 9s) durability. Amazon EFS is designed to sustain concurrent device failures by quickly detecting and repairing any lost redundancy.

Lifecycle Management

You can start saving on your storage costs by enabling EFS lifecycle management for your file system and choosing an age-off policy of 7,14, 30, 60, or 90 days. With EFS lifecycle management policies enabled, files automatically move from Amazon EFS Standard storage to EFS Standard-IA storage, or from Amazon EFS One Zone storage to EFS One Zone-IA storage. Lifecycle management reduces storage costs by up to 92 percent.

Data Transfer and Backup

AWS Data Sync, AWS Backup and AWS Transfer family services can be used for this purpose.

EFS File Sync

EFS File Sync provides a fast and simple way to sync existing file systems into Amazon EFS securely. It copies file data and file system metadata such as ownership, timestamps, and access permissions.

- You can choose to run EFS File Sync either on-premises as a virtual machine (VM) or in AWS as an EC2 instance.

Security

- EFS offers the ability to encrypt data at rest and in transit. Data encryption in transit uses industry-standard Transport Layer Security (TLS) 1.2 to encrypt data sent between your clients and EFS file systems.
- You can control access to files and directories with POSIX-compliant user and group-level permissions.
    - POSIX permissions allow you to restrict access from hosts by user and group.
    - You do not use IAM to control access to files and directories by user and group. However, you can use IAM to control who can administer the file system configuration.
- You can control network access to your file systems by using Amazon Virtual Private Cloud (Amazon VPC) security group rules. You can also control application access to your file systems by using AWS Identity and Access Management (IAM) policies and Amazon EFS access points. Amazon EFS satisfies many eligibility and compliance requirements to help you meet your regulatory needs.
- **How to enable users to save files to the EFS?** After creating a file system, by default, only the root user (UID 0) has read-write-execute permissions. For other users, to modify the file system, the root user must explicitly grant them access. Create a subdirectory for each user, and grant read-write-execute permissions to the users. Then, mount the subdirectory to the users' home directory.

Limitations

- Amazon EFS only supports Linux systems and macOS.
- There is no lifecycle policy available for deleting files on EFS.
- Amazon EFS is an NFS file system so does not allow access to data using SMB protocol. Workarounds involve third-party NFS clients or tools for Windows.

Use Cases

- Containers and Serverless persistent file storage
- Move to a managed file systems
- Analytics and Machine Learning - To create personalized environments, these environments can include home directories storing notebook files, training data, and model artifacts.
- Web serving and content management
- Application testing and development - A common storage repository in which you can share code and other files in a secure and organized way.
- Media and entertainment - Media workflows often depend on shared storage to manipulate large files. Example workflows include video editing, studio production, broadcast processing, sound design, and rendering. Amazon EFS provides a strong data consistency model with high throughput and shared file access. This consistency model cuts the time it takes to perform these jobs and consolidate multiple local file repositories into a single location for all users.
- Database backups - Amazon EFS presents a standard file system that you can mount with NFSv4 from database servers. This provides an ideal platform to create portable database backups using native application tools or enterprise backup applications

Pricing

- Pay-as-you-go: No upfront fees or commitments.
- Storage: Charged for storage used, based on region and storage class:
    - EFS Standard (most expensive)
    - EFS Infrequent Access (IA)
    - EFS One Zone IA
    - EFS Archive (least expensive, for long-term storage)
- Throughput:
    - Bursting Throughput mode: Dynamic, no upfront provisioning.
    - Provisioned Throughput mode: Fixed throughput, additional charges.
- File Sync: Data transfer into EFS charged per GB.
- EFS is generally more expensive than S3, especially for large-scale, inactive data.

Amazon FSx

Simple and fully managed

FSx for Lustre is a fully managed AWS storage service.

Availability

- Amazon FSx offers a multiple Availability Zone deployment option (designed to provide continuous availability to data) even if an AZ is unavailable.
- Supports Distributed File System Replication in both Single-AZ and Multi-AZ deployments.

High Scalable Performance

- It is much more scalable, available, performant, and expensive than EFS.
- FSx for Lustre is deployed using solid state drive (SSD) drives to provide high performance for management operations and serving metadata to your workloads. You can choose between SSD storage or hard disk drive (HDD) storage used to serve data. You choose the baseline throughput performance you need to meet your workload requirements and optimize costs.
- With the built-in Lustre file locking, you can connect tens of thousands of CPU cores to the same storage resources and simultaneously access data as needed by your applications.

Seamless Access to Data repositories

- Native S3 Integration:
    a. Directly access S3 objects as files within FSx for Lustre.
    b. Streamline data processing workloads without complex data transfers.
    c. Write results back to S3 for long-term storage or further analysis.
    d. Ideal for tiered storage strategies: S3 for cold data, FSx for active processing.
- Native Drive for EC2 and EKS:
    a. FSx for Lustre appears as a standard drive to EC2 instances and EKS containers.
    b. Simplifies application integration and data access.
    c. No need for custom drivers or software.
- On-Premises Data Migration:
    a. Mount FSx for Lustre from on-premises clients over Direct Connect or VPN.
    b. Use parallel copy tools for efficient data transfer.
    c. Facilitates cloud migration and hybrid cloud scenarios.

Native-file-system compliant

POSIX Compliance:

- Adheres to POSIX standards: Ensures compatibility with Linux applications and tools.
- Integrated into Linux OS: Works seamlessly with Linux systems without requiring application modifications.
- Simplifies migration: Existing Linux workloads can move to FSx for Lustre without code changes.

Read-After-Write Close Consistency:

- Guarantees data consistency: Readers always see the latest version of data after a writer closes a file.
- Critical for HPC workloads: Ensures accurate results in compute-intensive and parallel processing scenarios.
- Enables reliable collaboration: Multiple users or processes can work on shared files with confidence.

Supported File Systems

File System Flexibility:

- Amazon FSx offers a wider range of file systems compared to EFS:
    - FSx for Lustre: High-performance parallel file system for compute-intensive workloads.
    - FSx for Windows File Server: Fully managed Windows file server for Windows applications and workloads.
    - FSx for OpenZFS: Highly scalable and adaptable file system with advanced data management features.
- EFS supports only NFSv4.1 for Linux-based clients.

FSx for Windows File Server:

- NTFS File Systems: Provides familiar Windows file storage for Windows-based applications.
- SMB Protocol: Supports native SMB access for Windows clients, as well as Linux and macOS.
- Active Directory Integration: Seamlessly integrates with on-premises Active Directory or AWS Managed AD for user authentication and authorization.
- Administrative and Security Features: Offers a rich set of tools for managing file shares, permissions, quotas, snapshots, and more.

Cost Optimized

Storage Options:

- SSD vs. HDD: Choose based on workload requirements:
    - SSD for latency-sensitive or high IOPS/throughput workloads.
    - HDD for throughput-focused workloads without strict latency needs.
    - SSD cache for HDD-based file systems to improve performance.
- Long-term Data Storage:
    - S3 for cost-efficient archival and long-term retention.
    - On-premises storage for existing infrastructure.
    - HDD-based FSx for Lustre for less frequently accessed data.

S3 Integration:

- Import data as needed: Minimize S3 PUT/GET requests to reduce costs.
- Retain actively processed data in FSx: Optimize performance and reduce S3 access charges.

Security

- Amazon FSx automatically encrypts your data-at-rest and in-transit. It is assessed to comply with ISO, PCI-DSS, and SOC certifications, and is HIPAA eligible.
- Data is encrypted automatically before being written to the file system and decrypted automatically as it is read.
- Data is encrypted at rest using AWS KMS.

- Data is encrypted in-transit using Kerberos, however, in Amazon Fsx for OpenZFS encryption in-transit only when accessed from EC2 instances.
- In FSx, windows authentication through Active Directory can be self-managed or managed by AWS.
- You control access to your application using Amazon Virtual Private Cloud (Amazon VPC) security groups. You configure your security groups to allow the access required for only your workload. You configure your VPC to include the resources associated with your workload and allow access from resources in other VPCs, as required.
- You control access using AWS Identity and Access Management (IAM) to set up users, groups, and roles and assign access permissions. IAM access permissions are applied for management and application programming interface (API) access to the FSx for Lustre file system.

Data Protection

- **Automated daily backups** – To help ensure that your data is protected, FSx for Windows File Server automatically takes highly durable, consistent daily backups of your files systems. Backups are stored in a managed area of Amazon S3. FSx for Windows File Server uses the Volume Shadow Copy Service (VSS) to make your backups file system-consistent. You can take additional backups of your file system at any point.
- **Easy file-level restores (Microsoft Windows shadow copies)** – Users can easily undo changes and compare file versions. FSx for Windows File Server supports restoring individual files and folders to previous versions using Windows shadow copies.
- Centralized backup and compliance with AWS Backup.
- Cross-Region and cross-account backup compliance.

Data deduplication

You can enable data deduplication to reduce costs associated with redundant data automatically by storing duplicated portions of your dataset only once.

Monitoring

You can monitor and audit API calls using AWS CloudTrail. CloudTrail monitors and logs all API calls to the FSx for Lustre service. You can use these logs for auditing configuration changes and access to the service.

Amazon FSx also offers user storage quotas to monitor and control user-level storage consumption.

FSx for Windows File Server supports auditing user access to your files, folders, and file shares by using Windows Event Logs. Logs can be published to Amazon CloudWatch Logs or streamed to Amazon Kinesis Data Firehose. These features are not enabled by default.

Efficient Migration With AWS DataSync

With AWS DataSync, you can move your self-managed file systems to fully managed Windows storage on FSx for Windows File Server in minutes. Integration with AWS DataSync automates

and accelerates copying data over the internet or AWS Direct Connect. DataSync copies your files together with file attributes and metadata.

- File system compatibility: Verify that your source file system is compatible with FSx for Windows File Server.
- Network bandwidth: Ensure sufficient network bandwidth for optimal transfer speed.
- DataSync agent: Install the DataSync agent on servers hosting source file systems.
- Permissions: Configure appropriate permissions for DataSync to access source and destination file systems.
- Pricing: DataSync has a pay-as-you-go pricing model based on the amount of data transferred.

Amazon FSx for NetApp ONTAP

FSx for ONTAP is a storage service that allows you to launch and run fully managed NetApp ONTAP file systems in the AWS Cloud. It provides the familiar features, performance, capabilities, and APIs of NetApp file systems with the agility, scalability, and simplicity of a fully managed AWS service.

Amazon FSx for NetApp ONTAP provides feature-rich, fast, and flexible shared file storage that's broadly accessible from Linux, Windows, and macOS compute instances running in AWS or on premises. FSx for ONTAP supports block level storage over iSCSI and file storage using the NFS and SMB protocols.

FSx for ONTAP automatically tiers data from SSD storage to capacity pool storage based on your access patterns. Automated tiering allows you to achieve SSD levels of performance for your workloads and only pay for SSD storage for a small fraction of your data.

FSx for ONTAP offers multiple throughput capacity levels that you can choose from. This allows you to cost-optimize for the performance your workloads require. You can also optionally provision higher levels of IOPS as needed. IOPS provisioning is independent from the storage and throughput capacity of your file system. This allows you to pay only for the IOPS that you need.

Support for NetApp's on-premises caching solutions: NetApp Global File Cache and FlexCache

FSx for ONTAP use cases are similar to those for Amazon EFS, Amazon FSx for Windows File Server, and Amazon FSx for OpenZFS. The key difference is that it applies multi-protocol access for applications and workflows using NFS and SMB protocols with an option for block storage using the iSCSI protocol. If you plan to use applications with multi-protocol access or migrate your existing NetApp ONTAP storage, FSx for ONTAP is your logical choice.

Use Cases

Horizontal - Machine Learning
Machine learning workloads use massive amounts of training data. These workloads often use shared file storage because multiple compute instances need to process the training datasets concurrently.

Horizontal - High Performance Computing
High performance computing (HPC) workloads span across machine learning, oil and gas discovery, pharmaceuticals, financial services, and genomics. HPC workloads need to process massive amounts of data. Multiple compute instances with high levels of throughput must be able to access this data.

Vertical - Genomics and Life Sciences
Genomics and life sciences workloads use massive amounts of data points. These workloads often use shared file storage accessed by many compute instances that need to process the datasets concurrently. FSx for Lustre reduces the processing time and scales to meet the application high-throughput demands.

Vertical - Media Processing and Transcoding
Media data processing workloads have different file access requirements. Some workloads require access to the massive datasets, some require high throughput, and some require high IOPS.

Vertical - Autonomous Vehicles
An AV development workflow involves multiple workloads, including ingest, processing, analytics, labeling, training, and simulation and validation. Often the applications for training and simulation and validation require POSIX-compliant, high-throughput, low-latency file systems.

Vertical - SAS Grid Computing
SAS Grid computing distributes SAS computing tasks among multiple computers on a network, all under the control of SAS Grid Manager. In this environment, workloads are distributed across a grid cluster of computers. This workload distribution enables workload balancing, multiple users to distribute workloads to a shared pool of resources, and accelerated processing.

Pricing

You pay only for the resources you use. No minimum fees are charged or no setup charges are incurred. You pay for the storage and throughput capacity that you provision for your file system and for any backups of your file system. Pricing elements include:

- Backups
- HDD and SSD storage capacity
- Throughput capacity

- Data transfer in and out of FSx for Windows File Server across Availability Zones and VPC peering connections.
- Data transfer out of FSx for Windows File Server to other AWS Regions

Block Storage

1. Files are split into fixed-sized chunks of data called blocks.
2. Each block is assigned an address. Other than address there is no additional metadata for a block.
3. Less bandwidth is used, as only data for updated or required blocks is transferred.
4. Common use case is when you need to perform low-latency operations and I/O intensive tasks, and transactional and high performance workloads. It is also used for storing containerized applications. Virtual machines take advantage of it to readily increase or decrease virtual drive size.
5. This storage system is strongly consistent (not eventually consistent). This prevents it from becoming unlimitedly scalable.
6. Block storage does not have additional network protocol overhead. As a direct connection, the overhead is only what the operating system adds.
    a. File storage processes include overhead for processing the storage protocol. Requests are processed from the clients, and this processing time adds latency.
    b. Object storage processes include overhead to process the Hypertext Transfer Protocol (HTTP) requests using REST-APIs. The storage's operating system then processes the HTTP requests. This process adds latency to the request time.
7. HDDs are best suited for applications that require sustained read throughput. Applications that read and write large sequential files, such as video files, are well suited for HDDs. Random read operations are slower due to the seek times to locate the blocks and read them. Write operations to HDDs are also slower because of the seek time to locate and write to the blocks.
8. SSDs provide the benefit of low latency and higher read and write performance for random I/O. SSDs do not have the seek times required for HDDs and are able to perform operations much faster.

Amazon Elastic Block Storage (EBS Volume)

Amazon EBS is an easy-to-use, high performance, block storage service. It is designed for use with Amazon EC2 compute instances for both throughput and transaction-intensive workloads at any scale. AWS recommends Amazon EBS for data that must be quickly accessible and requires long-term persistence. EBS volumes are well suited for use as the primary storage for file systems, databases, or any applications that require fine granular updates and access to raw, unformatted, block-level storage.

Network attached storage to EC2 instance.

Multiple EBS volumes can be attached to a single EC2 instance. However, the OS of an EC2 instance needs to be configured on how to use those volumes.

Connection between instance and EBS volume is direct, no one else can intercept that connection.

EBS volume can be detached from an EC2 instance and attached to another EC2 instance in the same AZ. EBS volumes can be attached/detached while EC2 is running.

EBS volumes are automatically backed up by default. EBS volumes are replicated within an AWS Availability Zone and can scale to store petabytes of data.

Elasticity

Flexible changes like volume type, volume size can be made on the fly. Elastic Volumes allow you to increase volume size, adjust performance, or change the volume type while the volume is in use. However, you cannot decrease an EBS volume size.

Snapshots

Snapshots of EBS can be created and stored in S3 bucket. Storing snapshots in S3 is managed by AWS. Snapshots can be distributed in multiple AZs. EBS volumes are AZ specific while snapshots are region specific.

You pay for only the storage capacity consumed for your snapshot data.

You backup EBS volumes by taking snapshots. This can be automated via the AWS CLI command “create-snapshot”. You can use EBS snapshots with automated lifecycle policies to back up your volumes in Amazon Simple Storage Service (Amazon S3).

When you create a new EBS volume based on a snapshot, the new volume begins as an exact replica of the original volume that was used to create the snapshot. Your data is loaded into the new replicated volume in the background. You can begin to use your new volume immediately while the EBS volume data loads. If you access data that hasn't been loaded yet, the volume immediately downloads the requested data from Amazon S3. EBS Snapshots then continues loading the remainder of the volume's data in the background.

You can delete any snapshot whether it is a full or incremental snapshot. When you delete a snapshot, only the data that is referenced exclusively by that snapshot is removed. Unique data is only deleted if all of the snapshots that reference it are deleted.

You can copy a completed snapshot within the same Region or from one AWS Region to another. The snapshot copy receives an ID that is different from the ID of the original snapshot.

Even though snapshots are saved incrementally, the snapshot deletion process is designed so that you need to retain only the most recent snapshot to restore the volume.

Pricing Example:
Scenario: You initially have 70 GB of data in your provisioned 2,000 GB gp3 volume on day 1 in a 30-day month. You add 30 GB of data on day 15 of the month. The price is $0.05 per GB-month of data stored for the AWS Region you select.

- Your costs are calculated for the volume size using the formula for days.
    - ((Rate per GB-month) * (stored data size) * (time period)) / (time period units per month)
    - ($0.05 per GB-month * 70 GB * 30 days) + ($0.05 per GB-month * 30 GB * 15 days) / (30 days per month) = $4.25 for the data storage

Multi-Attach

Depending on instance type and EBS volume, one EBS volume can be attached to multiple instances (all instances should be in the same AZ), which is called Amazon EBS Multi-Attach.

Multi-Attach is only supported with Provisioned IOPS SSD (io1 and io2) EBS volume types.

Amazon EBS does not manage data consistency for multiple writers. Your application or operating system environment must manage data consistency operations.

Multi-attach makes it easier to achieve higher application availability for applications that manage storage consistency from multiple writers. Each attached instance has full read and write permission to the shared volume. There is no additional fee to enable Multi-attach.

Elastic Volume

Elastic Volumes is a feature that allows you to easily adapt your volumes as the needs of your applications change. The Elastic Volumes feature allows you to dynamically increase capacity, tune performance, and change the type of any new or existing current generation volume with no downtime or performance impact. You can easily right-size your deployment and adapt to performance changes.

By using Amazon CloudWatch with AWS Lambda, you can automate volume changes to meet the changing needs of your applications.

However, it is not a fully managed service that auto-scales.

With Elastic Volumes, volume sizes can only be increased within the same volumes. To decrease a volume size, you must copy the EBS volume data to a new smaller EBS volume.

Types

EBS volumes can be of two categories i) solid-state drives (SSDs) and hard-disk drives (HDDs).

SSD - General

General Purpose SSD (gp2):

- Burstable performance: Can burst to 3,000 IOPS using burst credits, but credits are finite and replenish slowly.
- Performance tied to volume size: To achieve higher baseline performance, you must increase the volume size. gp2 volumes do not support separate provisioned IOPS or provisioned throughput options.
- May be less cost-effective for sustained performance: While gp2 volumes can be suitable for workloads with intermittent bursts, they may not be as cost-effective as gp3 for those requiring consistent high performance.

General Purpose SSD (gp3):

- All gp3 volumes include a free baseline performance of **3,000 provisioned IOPS** and **125 provisioned MB/s throughput**.
- Independent scaling of performance and capacity: **Customize IOPS (up to 16,000)** and **throughput (upto 1,000 MB/s)** separately from **volume size (1 GB to 16 TB)**.
- Cost-effective for consistent performance: Well-suited for workloads that require balanced price-performance, such as boot volumes and low-latency interactive applications.
- For example, gp3 volumes include 3,000 provisioned IOPS. If you provision 8,000 IOPS, your cost is based on the 5,000 IOPS that are in excess of the 3,000 IOPS base amount.
    - (((Provisioned IOPS amount) - (base IOPS amount)) * (price per IOPS-month)) / (1 month)
- For example, gp3 volumes include 125 MB/s of provisioned throughput. If you provision 500 MB/s throughput, your costs are based on the 375 MB/s in excess of the 125 MB/s base throughput amount.
    - (((Provisioned MB/s throughput) - (base MB/s throughput)) * (price per MB/s-month)) / (1 month)

SSD - IOPS

Highest performance SSD volume designed for latency-sensitive transactional workloads. (I/O-intensive NoSQL and relational databases). **Up to 50 IOPS per GiB** and up to **64,000 IOPS per volume. Volume size is 4 GB to 16 TB.**

I/O optimized instances are also geared more towards storage performance than network performance.

Whenever you have a requirement of IOPS > 10,000 then provisioned IOPS SSD is the only option as General Purpose SSD caps out 10,000 IOPS.

io1 and io2 Provisioned IOPS SSD volume differences:

- **io1 volumes** are designed to provide **99.8–99.9 percent volume durability with an AFR no higher than 0.2 percent**, which translates to a maximum of two volume failures per 1,000 running volumes over a 1-year period.
- **io2 volumes** are designed to provide **99.999 percent volume durability with an AFR no higher than 0.001 percent**, which translates to a single volume failure per 100,000 running volumes over a 1-year period.
- **Provisioned IOPS SSD io2 and io1 volumes both can be configured to deliver up to 64,000 IOPS and 1,000 MB/s throughput**, however only io2 volumes have a tiered pricing structure for your provisioned IOPS. Therefore, as you provision higher IOPS on a single volume, the effective provisioned IOPS charges decrease, making it more economical to scale IOPS on a single volume. io2 and io2 Block Express provisioned IOPS tiers:
    - **Tier 1 – Up to 32,000 IOPS**
    - **Tier 2 – 32,001 IOPS to 64,000 IOPS**
    - **Tier 3 – Over 64,000 IOPS**
        - **Tier 3 applies only to io2 Block Express volumes attached to EC2 instances that are supported by io2 Block Express.**

IOPS-to-Volume Size Ratio:

- gp3 (General Purpose SSD): 500:1 ratio, meaning you get 500 IOPS for every 1 GB of provisioned volume size.
- io1 (Provisioned IOPS SSD): Unlike gp3 volumes, io1 volumes don't have a fixed IOPS-to-volume-size ratio. Instead, you explicitly provision the desired IOPS independently of the volume size.
- io2 (Block Express SSD): Similar to io1, io2 volumes don't have a fixed IOPS ratio. You provision IOPS and throughput independently.

HDD - Throughput optimized (st1)

Low cost HDD volume designed for frequently accessed throughput intensive-workloads. (Big data, data warehouses). Throughput Optimized HDD is the most cost-effective storage option, and for a small DB with low traffic volumes, it may be sufficient. Note that the volume must be **at least 500 GB in size**. Provides **up to 500 IOPS per volume** but does not provide an SLA for IOPS. It cannot be used as a boot volume. Throughput Optimized HDD st1 volumes are configurable for throughput only.

Baseline throughput: Determined by volume size, ranging from **40 MB/s for a 500 GB volume up to 500 MB/s for larger volumes**.

Volume storage for Throughput Optimized HDD (st1) volumes is charged by the amount you provision in GB per month until you release the storage. I/O is included in the price of the volumes, so you pay only for each GB of storage you provision.

Performance for st1 volumes is scaled by increasing or decreasing the provisioned volume size. st1 volumes do not support separate provisioned IOPS or provisioned throughput options.

Burst credits: Accumulate over time and allow for temporary bursts of higher throughput (**up to 250 MB/s per TB of volume size**).

HDD - Cold (sc1)

Lowest cost HDD volume designed for less frequently accessed workloads/data and for sequential data access. (Colder data requiring fewer scans per day). Lowest cost storage — cannot be a boot volume.

Volume storage for Cold HDD (sc1) volumes is charged by the amount you provision in GB per month until you release the storage. I/O is included in the price of the volumes, so you pay only for each GB of storage you provision.
Performance for sc1 volumes is scaled by increasing or decreasing the provisioned volume size. sc1 volumes do not support separate provisioned IOPS or provisioned throughput options.

- **Baseline throughput:** Ranges from **12 MB/s for a 500 GB volume up to 250 MB/s for larger volumes.**
- **Burst throughput**: **Up to 80 MB/s per TB of volume size**, using burst credits.
- IOPS: **Up to 250 IOPS per volume**, without an SLA.

HDD - Magnetic Standard

Cheap, infrequently accessed storage — lowest-cost storage. It cannot be a boot volume.

Additional Considerations

Make sure, EC2 instance is capable of supporting IOPS provided by EBS, for example, Provisioned IOPS SSD can support up to 32000 IOPS and free tier EC2 instances can not handle this number of IOPS however, compute optimized instances can.

Encryption

1. Encryption is supported on all Amazon EBS volume types.
2. Encryption is supported for data at rest inside the volume.
3. Data in transit between an instance and an encrypted volume is also encrypted.
4. You can have encrypted unencrypted EBS volumes attached to an instance at the same time.
5. Snapshots of encrypted volumes are encrypted automatically.
6. EBS volumes restored/created from encrypted snapshots are encrypted automatically.
7. There is no direct way to change the encryption state of a volume.
8. You can expect the same IOPS performance on encrypted volumes as on unencrypted volumes.

Data Key

Data key is used for encrypting volume and is stored on-disk with encrypted data. The data key is always stored encrypted (not in plain text). The same data key is shared by snapshots of the volume and any subsequent volumes created from those snapshots.

Customer Master Keys

- CMKs: Encryption keys used to protect EBS volumes and snapshots.
- AWS Key Management Service (KMS): Manages CMKs, including creation, rotation, and access control.
- Encryption at rest: Data on EBS volumes and snapshots is encrypted using CMKs.

Encryption Process:

1. Creating a CMK:
    - A default CMK is generated for the first encrypted volume.
    - Subsequent encrypted volumes use unique keys (AES 256 bit).
2. Encrypting Volumes:
    - A CMK is assigned to a volume during creation.
3. Encrypting Snapshots:
    - Snapshots inherit the CMK from the source volume.

Sharing Encrypted Volumes and Snapshots:

- Sharing Encrypted Snapshots:
    - Requires a non-default CMK.
    - Set cross-account permissions for CMK access.
    - Mark snapshot as private and share with specific accounts.
    - Receiving account must copy and re-encrypt the snapshot (recommended).
- Cannot Share Default CMK Snapshots:
    - Must use custom CMKs for sharing.

Key Points:

- Cannot Change CMK for Existing Volume:
    - Create a copy of the snapshot and change keys during copy.
- Cannot Make Encrypted Snapshots Public:
    - Share only with specific accounts.
- Recommended Re-Encryption:
    - Receiving account should re-encrypt shared snapshots with their own CMK.

Status Check

The possible values for An EC2 status check on an EBS volume are **ok, impaired, warning, or insufficient-data**. If all checks pass, the overall status of the volume is ok. If the check fails, the overall status is impaired. If the status is insufficient-data, the checks may still be taking place on your volume at the time.

Volume Monitoring

Performance metrics, such as bandwidth, throughput, latency, and average queue length, are available through the AWS Management Console. Amazon CloudWatch provides these metrics so that you can monitor the performance of your volumes. You can make sure that you are providing enough performance for your applications and paying only for resources you need.

Amazon EBS volume events

- createVolume
- deleteVolume
- attachVolume
- reattachVolume
- modifyVolume

Amazon EBS Snapshots events

- createSnapshot
- createSnapshots
- copySnapshot
- shareSnapshot

AWS Compute Optimizer for EBS volumes

Once your EBS volumes are in operation, you can monitor them and verify that your volumes are providing optimal performance and cost effectiveness using AWS Compute Optimizer. Compute Optimizer is a service that analyzes the configuration and utilization metrics of your AWS resources. It reports if your resources are optimized and generates optimization recommendations to reduce the cost and improve the performance of your workloads.

Amazon Data Lifecycle Manager

You can use Amazon Data Lifecycle Manager (Amazon DLM) to automate the creation, retention, and deletion of snapshots that you use to back up your EBS volumes and Amazon EBS-backed Amazon Machine Images (AMIs).

Lifecycle Policies

- **Policy type** - Defines the type of resources that the policy can manage. Amazon DLM supports two types of lifecycle policies:
    - **Snapshot lifecycle policy** - Used to automate the lifecycle of EBS snapshots. These policies can target EBS volumes and instances.
        - Cross-account copy event policy - Used to automate the copying of snapshots across accounts. This policy type should be used in conjunction with an EBS snapshot policy that shares snapshots across accounts.

    - **EBS-backed AMI lifecycle policy** - Used to automate the lifecycle of EBS-backed AMIs. These policies can target instances only.
- **Resource type** - Defines the type of resources that are targeted by the policy. Snapshot lifecycle policies can target instances or volumes. Use VOLUME to create snapshots of individual volumes, or use INSTANCE to create multi-volume snapshots of all of the volumes that are attached to an instance.
- **Target tags** - Specifies the tags that must be assigned to an EBS volume or an Amazon EC2 instance for it to be targeted by the policy.
- **Schedules** - The start times and intervals for creating snapshots or AMIs. The first snapshot or AMI creation operation starts within one hour after the specified start time. Subsequent snapshot or AMI creation operations start within one hour of their scheduled time.
- **Retention** - Specifies how snapshots or AMIs are to be retained. You can retain snapshots or AMIs based on their total count (count-based) or their age (age-based).
    - For snapshot policies, when the retention threshold is reached, the oldest snapshot is deleted.
    - For AMI policies, when the retention threshold is reached, the oldest AMI is deregistered and its backing snapshots are deleted.

Amazon Data Lifecycle Manager quotas:

- You can create up to 100 lifecycle policies per AWS Region.
- You can add up to 45 tags per resource.

Policy Schedules

- Define when snapshots or AMIs are created.
- Up to four schedules per policy:
    - One mandatory schedule
    - Up to three optional schedules
- Purpose:
    - Automate snapshot / AMI creation at different frequencies
    - Simplify management by avoiding multiple policies
- Schedule Configuration:
    - Frequency (e.g., daily, weekly, monthly)
    - Fast snapshot restore (snapshot policies only)
    - Cross-Region copy rules
    - Tags
- Activation:
    - Each schedule activates individually based on its frequency.
- Overlapping Schedules:
    - If multiple schedules activate simultaneously:
        - DLM creates only one snapshot/AMI.

    - Applies retention settings of the schedule with the highest retention period.
    - Applies tags from all activated schedules.

Use Cases

- Used as a boot volume for OS and as a data volume.
- Can also be used as a database (SQL / NoSQL), however, it can get expensive and complex to manage especially for large databases.
- Big data analytics engine.
- File systems and media workflows - you can easily scale with additional volumes to support growing file systems.

Limitations

- Maximum volume size is 64 tebibytes (TiB).
- It can not be accessed from the public internet.
- The Elastic Block Store (EBS) is not a good solution for concurrent access from many EC2 instances and is not the most cost-effective option either.
- You can backup EBS using mirroring. Mirroring data would provide resilience; however, both volumes would need to be mounted to the EC2 instance within the same AZ, so you are not getting the redundancy required.
- Lift and shift, self-managed database migrations are best suited for Amazon EBS.

Pricing

Pricing for EBS volumes is based on the volume type, provisioned volume size, and the provisioned IOPS and throughput performance. EBS volume pricing varies based on the Availability Zone where it resides. The pricing for Amazon EBS snapshots is based on the actual amount of storage space that you use.

You are charged for volumes attached to EC2 instances that are active, stopped, or even terminated as long as the volumes exist.

EBS Snapshots Archive offers you a lower price per GB compared to standard EBS Snapshots.

Instance Store

EC2 instance has an internal store, which is called instance store. If an EC2 instance is stopped or terminated all the data in the instance store is gone. Instance stores are good for temporary storage like caches, buffers , etc. EC2 instance stores are less expensive than EBS.

Instance stores offer very high performance and low latency. As long as you can afford to lose an instance, i.e., you replicate your data, these can be a good solution for high performance/low latency requirements.

You can specify the instance store volumes for your instance only when you launch an instance. You can't attach instance store volumes to an instance after you've launched it.

If an instance reboots (intentionally or unintentionally), data in the instance store persists.

VS EBS

EBS backed instances can be stopped. You will not lose the data on this instance if it is stopped. Instance store backed instances cannot be stopped. If the underlying host fails, the data will be lost.

EBS volume root devices are launched from AMIs that are backed by EBS snapshots. Instance store volume root devices are created from AMI templates stored on S3.

Pricing

Instance stores' cost is included in the instance charges, so it can also be more cost-effective than EBS Provisioned IOPS.

If instance stores are appropriate, you can reduce your costs because they are not provisioning EBS volumes. You only pay for the EC2 instance when it is in a running state.

EBS vs EFS

Amazon EFS	**Amazon EBS Provisioned IOPS**
Data is stored redundantly across multiple AZs.	Data is stored redundantly in a single AZ.
Up to thousands of Amazon EC2 instances, from multiple AZs, can connect concurrently to a file system.	A single Amazon EC2 instance in a single AZ can connect to a file system.
Used for, big data and analytics, media processing and workflows, content management, web serving and home directories.	Boot Volumes, transactional and NoSQL databases, data warehousing and ETL
Low and consistent latency	Lowest and consistent latency
Throughput: 10+ GB per second	Up to 2GB per second

Object Storage

Stored like files and treated as a single unit. Object storage is similar to file storage with few differences. Each object is identified by a unique identifier and there are no folders or complex hierarchies; it follows a flat structure. Additional metadata is bundled with the data.

Data is stored as a single object, for example, if you want to update a single point of data in the object we need to update the entire object.

Common use cases are long-term data retention (archiving), backup and recovery, cost-effective media storage and geographical data replication.

Amazon S3

Standalone storage solutions not tied to a compute instance. S3 is durable and distributed storage for large amounts of data (object). Allows you to store as many objects as desired with an individual object file size limit of 5 TB. Objects are stored in Buckets. Buckets are region specific but a bucket name should be unique globally (across all aws accounts in all aws regions within a partition (Standard Region, China Regions, AWS GovCloud(US)).

In S3 bucket we have a logical hierarchy, meaning object keys may appear to be following a folder structure but physically they are not folders. Breaking S3 Storage into departmental buckets can improve performance over placing all organizational data in a single bucket with folders (prefixes).

Objects are not deleted permanently, S3 puts a marker on the object that shows that you tried to delete it. Updates to an object are atomic — when reading an updated object, you will either get the new object or the old one; you will never get partial or corrupt data.

Bucket Overview

Buckets are permanent containers that hold objects. You can create between 1 and 100 buckets in each AWS account. You can increase the bucket limit to a maximum of 1,000 buckets by submitting a service limit increase. Bucket sizes are virtually unlimited so you don't have to allocate a predetermined bucket size the way you would when creating a storage volume or partition.

An Amazon S3 bucket is a versatile storage option with the ability to: host a static web site, retain version information on objects, and employ life-cycle management policies to balance version retention with bucket size and cost.

Buckets are permanent storage entities and only removable when they are empty. After deleting a bucket, the name becomes available for reuse by any account after 24 hours if not taken by another account.

There’s no limit to the number of objects you can store in a bucket. You can store all of your objects in a single bucket, or organize them across several buckets. However, you can't create a bucket from within another bucket, also known as nesting buckets.

Bucket Name

Once created you cannot change a bucket name. Bucket names are globally viewable and need to be DNS-compliant. Use a dot (.) in the name only if the bucket's intended purpose is to host an Amazon S3 static website; otherwise do not use a dot (.) in the bucket name

Path Style URLs

With path-style URLs, the bucket name comes after the global or region-specific endpoint.
https://s3-us-west-2.amazon.com/[bucket-name]/[object-name].jpg.
There is a deprecation plan for this URL.

Virtual Hosted Style URLs

Virtual hosting is the practice of serving multiple websites from a single web server. One way to differentiate sites is by using the host name (bucket name) of the request. In a virtual-hosted–style URL, the bucket name is part of the domain name in the URL, which makes the URL easier to read, and more end-user friendly.

https://[bucket-name].s3-us-west-2.amazonaws.com/[object-name].jpg

Virtual hosting also has other benefits. You can completely customize the URL of your Amazon S3 resources by naming your bucket after your registered domain name and making that name a DNS alias for Amazon S3.

Object Overview

An object consists of the following: Key, version ID, value, metadata, and access control information. The object key (or key name) uniquely identifies the object in a bucket. Object metadata is a set of name-value pairs. You can set object metadata at the time you upload it. After you upload the object, you cannot modify object metadata. The only way to modify object metadata is to make a copy of the object and set the metadata. When you upload a file as an object, you can set permissions on the object and any metadata.

Due to the distributed nature of Amazon S3, requests could temporarily route to the wrong Region. This is most likely to occur immediately after the creation or deletion of buckets. A temporary redirect is a type of error response that signals to the requester that they should resend the request to a different endpoint.

For example, if you create a new bucket and immediately make a request to the bucket, you might receive a temporary redirect, depending on the location constraint of the bucket. If you created the bucket in the US East (N. Virginia) AWS Region, you will not see the redirect because this is also the default Amazon S3 endpoint.

S3 Strong Data Consistency Model

Amazon S3 now delivers strong read-after-write consistency for any storage request, without changes to performance or availability, without sacrificing regional isolation for applications, and at no additional cost. Any request for S3 storage is now strongly consistent.

After a successful write of a new object or overwrite of an existing object, any subsequent read request immediately receives the latest version of the object. Amazon S3 also provides strong consistency for list operations, so after a write, you can immediately perform a listing of the objects in a bucket with any changes reflected.

For bucket operations such as reading a bucket policy or metadata, the consistency model remains eventually consistent.

PUT Operations

Use the PUT request operation to add an object to a bucket. You must have WRITE permissions on a bucket in order to add an object. Amazon S3 never adds partial objects; if you receive a success response, you can be confident that the entire object was stored durably.

If the object already exists in the bucket, the new object overwrites the existing object. Amazon S3 orders all of the requests that it receives but it is possible that if you send two requests nearly simultaneously, the received requests will be in a different order than sent.
The last request received is the one which is stored. This means that if multiple parties are simultaneously writing to the same object, they may all get a success response even though only the last write wins. This is because Amazon S3 is a distributed system and it may take a few moments for one part of the system to communicate that another part has received an object update.

You can upload or copy objects of up to 5 GB in a single PUT operation.

Multipart upload

Multipart upload can be used to speed up uploads to S3. It uploads objects in parts independently, in parallel, and in any order. It also improves throughput. It is performed using the S3 Multipart upload API.

- It is recommended for objects of size 100MB or larger.
- It can be used for objects from 5MB up to 5TB.
- It must be used for objects larger than 5GB.

If transmission of any part fails, it can be retransmitted. It can pause and resume object uploads. It can also begin an upload before you know the final object size.

When using multipart uploads, Amazon S3 retains all the parts on the server until you complete or discontinue the upload. To avoid unnecessary storage costs related to incomplete uploads,

make sure to complete or discontinue an upload. Use lifecycle rules to clean up incomplete multipart uploads automatically.

In a lifecycle policy, you can use the **AbortIncompleteMultipartUpload** element to set a maximum number of days for a multipart upload to remain in progress. If the upload doesn't complete within that specified number of days it becomes eligible for a cancel operation and Amazon S3 stops the multipart upload and deletes the parts associated with the multipart upload.

Use cases

Common use cases, backup and storage, media hosting, software delivery, data lakes, static websites, static content.

Replication

Cross-Region Replication

- Objects stored in a bucket will never leave the region where they are stored unless you move them to another region or enable Cross-Region Replication.
    a. Cross-Region replication relies on versioning.
    b. Cross-Region Replication creates a replica copy in another region but should not be used for spreading read requests across regions. There will be 2 S3 endpoints, and CRR is not designed for 2 way sync, so this would not work well.
- Buckets that are configured for object replication can be owned by the same AWS account or by different accounts.

Same-Region Replication

SRR makes another copy of S3 objects within the same AWS Region, with the same redundancy as the destination storage class. This allows you to automatically aggregate logs from different S3 buckets for in-region processing, or configure live replication between test and development environments.

Versioning

Buckets can be versioned to allow versioning of objects (if enabled). Multiple versions of the same objects (having the same name) are kept in the same bucket. Each version is billed as old versions are still stored and accessible, until they are permanently deleted. If buckets are versioned then they can not be restored to unversioned state however, we can suspend versioning. Versioning stores all versions of an object (including all writes even if an object is deleted). It protects against accidental object/data deletion or overwrites and enables “roll-back” and “un-delete” capabilities.

Only the S3 bucket owner can permanently delete objects once versioning is enabled. When you try to delete an object with versioning enabled, a DELETE marker is placed on the object. You can delete the DELETE marker, and the object will be available again. Deletion with versioning replicates the delete marker. But deleting the delete marker is not replicated.

Objects that existed before enabling versioning will have a version ID of NULL. If you suspend versioning, the existing objects remain as they are; however, new versions will not be created. While versioning is suspended, new objects will have a version ID of NULL, and uploaded objects of the same name will overwrite the existing object.

By default, an HTTP GET retrieves the most recent version. Reverting to previous versions isn't replicated.

Tags

Amazon S3 tags are key-value pairs and apply to a whole bucket or to individual objects to help with identification, searches, and data classification. Using tags for your objects allows you to effectively manage your storage and provide valuable insight on how your data is used. Newly created tags assigned to a bucket, are not retroactively applied to its existing child objects.

Security

- Everything in S3 is private by default. Access to the bucket can be controlled using Access control lists (ACL).
- S3 bucket policies are similar to IAM policies. Bucket policies are attached to buckets while IAM policies are attached to resources and users. In S3 bucket policies, we specify how objects in buckets can be accessed. S3 policy size limit is > IAM policy size. Ideally, we want to use both of these roles for providing more granular control to our data.
- Data in S3 is encrypted at rest and at transit by default (using base level encryption at no cost).
- MFA (Multi-Factor Authentication) delete can be set in amazon S3 bucket.
- S3 Access Points to simplify managing data access to shared datasets by creating access points with names and permissions specific to each application or sets of applications. By using S3 Access Points that are restricted to a VPC, you can secure your S3 data within your private network. Additionally, you can use AWS Service Control Policies to require that any new S3 Access Point in your organization is restricted to VPC-only access.
- Access Analyzer for S3 is a feature that monitors your bucket access policies, ensuring that the policies provide only the intended access to your S3 resources. Access Analyzer for S3 evaluates your bucket access policies so that you can discover and swiftly remediate buckets with potentially unintended access.

Server-side encryption options

- SSE-S3 — Server Side Encryption with S3 managed keys
    - Each object is encrypted with a unique key.
    - The encryption key is encrypted with a master key.
    - AWS regularly rotates the master key and uses AES 256.

    - Does not allow customers to control who can access the keys.
- SSE-KMS — Server-Side Encryption with AWS KMS keys
    - KMS uses Customer Master Keys (CMKs) to encrypt.
    - Can use the automatically created CMK key or you can select your own key (gives you control for the management of keys).
    - Allows the customer to control who can access the keys.
    - An envelope key protects your keys. Chargeable.
    - It is similar to SSE-S3, but with some additional benefits and charges for using this service. There are separate permissions for the use of a CMK that provides added protection against unauthorized access of your objects in Amazon S3. SSE-KMS also provides you with an audit trail showing when and who used the CMK.
- SSE-C — Server-Side Encryption with client provided keys
    - Client manages the keys, and S3 manages encryption.
    - AWS does not store the encryption keys.
    - If keys are lost, data cannot be decrypted.

You can't apply different types of server-side encryption to the same object simultaneously.

Client-side encryption

Client-side encryption is the act of encrypting sensitive data before sending it to Amazon S3. When using client-side encryption, the encryption performs locally and your data never leaves the run environment unencrypted. You maintain possession of your master encryption keys, and they are never sent to AWS therefore, it is important that you safely store them (i.e., as a file or using a separate key management system) and load them when uploading or downloading objects. This ensures that no one outside of your environment has access to your master keys and without access to the master keys; your data cannot be decrypted. If your master encryption keys are lost, you will not be able to decrypt your own data, therefore it is essential that if you use client-side encryption, that you store your keys safely.

To enable client-side encryption, you have the following options:

- Use a customer master key (CMK) stored in AWS Key Management Service (AWS KMS). With this option, you use an AWS KMS CMK for client-side encryption when uploading or downloading data in Amazon S3.
- Use a master key that you store within your application. With this option, you provide a client-side master key to the Amazon S3 encryption client. The client uses the master key only to encrypt the data encryption key that it generates randomly.

Permissions for IAM users

With IAM policies, you can grant IAM users fine-grained control to your S3 buckets. It is preferred over using bucket ACLs.

For an IAM user to access resources in another account, the following must be provided:

- Permission from the parent account through a user policy.
- Permission from the resource owner to the IAM user through a bucket policy or the parent account through a bucket policy, bucket ACL, or object ACL.

If an AWS account owns a resource, it can grant permissions to another account. That account can then delegate those permissions — or a subset of them — to users in the account (permissions delegation). An account that receives permissions from another account cannot delegate permissions cross-account to a third AWS account.

Bucket ACL

- Basic permissions: Grant READ, WRITE, and READ_ACP (view permissions) for buckets or objects.
- Limited use cases: Primarily for granting WRITE access to the S3 Log Delivery group.
- Consider using IAM policies instead for more control and flexibility.

Bucket Policies

- Granular access control: Grant permissions for specific S3 actions, resources, and conditions.
- Cross-account access: Allow access from users or accounts outside your own.
- Conditional permissions: Restrict access based on factors like IP address, referrer, or time.
- Use cases:
    - Cross-account permissions without IAM roles
    - Reaching IAM policy size limits
    - Preferring policies within the S3 environment

IAM User Policies

- Centralized control: Manage permissions for S3 and other AWS services.
- Account-wide permissions: Grant access to users within your account.
- Object access control: Manage object permissions when the object owner is the same as the bucket owner.
- Use cases:
    - Controlling access to multiple AWS services
    - Managing complex permission requirements across many buckets
    - Preferring centralized policy management in IAM

Prioritize IAM policies: Generally more flexible and manageable for most S3 access control scenarios.

Use bucket policies: Specifically for cross-account access without IAM roles or when reaching IAM policy size limits.

Avoid bucket ACLs: Except for specific use cases like S3 Log Delivery group access.

Storage types

S3 Standard

General purpose storage. Standard storage stores data in minimum three AZs.

S3 Intelligent Tiering

If data access pattern is unknown, S3 monitors access patterns and stores data in frequent, in-frequent and archive instance access tiers for cost-effectiveness.

S3 Standard-Infrequent Access

For data that is accessed less frequently but requires rapid access when needed. This combination of low cost and high performance makes S3 Standard-IA ideal for long-term storage and backups, and as a data store for disaster recovery files.

S3 One Zone-Infrequent Access

In this type of storage, data is stored in a single AZ. Lower cost option for infrequent accessed data that is secondary and easy to recreate.

S3 Glacier

- You cannot specify Glacier as the storage class at the time you create an object.
- Glacier objects are visible through S3 only and not through Glacier directly.
- Glacier does not archive object metadata; you need to maintain a client-side database to maintain this information.

S3 Glacier Standard Retrieval

Glacier Standard retrieval is 3 - 5 hours. Glacier is designed for durability of 99.999999999% of objects across multiple Availability Zones. Data is resilient in the event of one entire Availability Zone destruction. Glacier is “designed for” availability of 99.99%.

S3 Glacier Instant Retrieval

For data that is rarely accessed and required millisecond retrieval. Saves 68% compared with standard with same latency and throughput performance.

S3 Glacier Expedited Retrieval

Glacier Expedited retrievals allow you to quickly access your data when occasional urgent requests for a subset of archives are required. For all but the largest archives (250 MB+), data accessed using Expedited retrievals are typically made available within 1–5 minutes. Provisioned Capacity ensures that retrieval capacity for Expedited retrievals is available when you need it.

S3 Glacier Flexible Retrieval

For data that is accessed one or twice per year. If requested data can be accessed in 1 - 5 minutes (Expedited) and free bulk retrievals in up to 5-12 hours. Minimum storage duration of 90 days.

Retrieval fee per GB for non-bulk retrievals.

Specify an Amazon Simple Notification Service (Amazon SNS) topic to which S3 Glacier Flexible Retrieval can post a notification after the job is completed. S3 Glacier Flexible Retrieval sends a notification only after it completes the job.

Request job information explicitly — You can also use the S3 Glacier Flexible Retrieval describe job operation to periodically poll for job information; however, using Amazon SNS notifications is recommended.

S3 Glacier Bulk Retrieval

Bulk retrievals are S3 Glacier's lowest-cost retrieval option, which you can use to retrieve large amounts, even petabytes, of data inexpensively in a day. Bulk retrievals typically complete within 5–12 hours.

S3 Glacier Deep Archive

For data that is accessed one or twice per year with default retrieval time of 12 hours. Good for long term storage of data (7 - 10 years) to meet regulatory compliance requirements. It has a minimum storage duration of 180 days.

S3 Glacier Vault Locks

Vault Lock allows you to easily deploy and enforce compliance controls on individual Glacier vaults via a lockable policy (Vault Lock policy). S3 Glacier Vault lock is designed for data that requires the highest levels of protection and immutability. Here are some common types of data that are well-suited for this level of security:

In S3 Glacier vaults are the container for archived objects not buckets. Maximum 1000 vaults can be created per region, per account.

Storage charges:

- Per GB per month: Same as standard S3 Glacier, based on the amount of data stored in the vault. Costs decrease as storage duration increases (up to 50% discount for longer storage periods).

Vault lock charges:

- Vault lock activation fee: A one-time fee per vault to enable the vault lock feature.
- Vault lock monthly fee: A recurring monthly fee per vault for maintaining the vault lock active.

Retrieval charges:

- Standard retrievals: Similar to standard S3 Glacier, with fees based on data size and retrieval speeds (Expedited, Standard, or Bulk).
- Urgent retrievals: Additional fees applied to expedite data retrieval within specific timeframes (minutes or hours).

S3 on Outposts

For delivering data to on-premises AWS outposts environment. Used for satisfying local data residency requirements (data that need to be closed to on-premise locations for performance reasons).

Storage Lifecycle

- Instead of manually transitioning data from one storage type to another we can automate this process using two types of actions
    - Transition actions - When objects should transition to another storage class.
    - Expiration actions - When objects should expire and permanently deleted. Amazon S3 deletes expired objects on your behalf.
- S3 lifecycle actions apply to any storage class, including Glacier. You can transition objects from the S3 Standard storage class to any other storage class but you cannot create a lifecycle rule that moves objects from any storage class back into to the S3 Standard storage class. **Amazon S3 does not transition objects between storage classes if they are smaller than 128 KB because it's not cost effective to do so. Objects must remain for a minimum of 30 days in S3 Standard before they can transition to S3 Standard-IA, and S3 One Zone-IA. Lifecycle configuration on multi-factor authentication (MFA)-enabled buckets is not supported. Lifecycle actions are not captured by AWS CloudTrail object level logging. If logging is requried, you can use Amazon S3 Server access logs to capture S3 lifecycle-related actions.**
- You cannot transition to REDUCED_REDUNDANCY from any storage class.

S3 Object Lock

It supports WORM (Write Once Read Many) as a file storage method. Object locking can only be enabled when a bucket is created. Locked objects can’t be deleted.

Batch Operations

It creates jobs to enable automatic actions.

S3 Select

Amazon S3 Select is designed to help analyze and process data within an object (in place) in Amazon S3 buckets in a faster and cheaper way. It works by providing the ability to retrieve a subset of data from an object in Amazon S3 using simple SQL expressions.

Data Format and Compression:

- Supported formats: Only CSV and JSON files are currently supported, with UTF-8 encoding. Multi-line CSVs and complex JSON objects are not supported.
- Compression: Only uncompressed and GZIP compressed files can be used.

Query Capabilities:

- Limited SQL operations: Only a subset of SQL features are supported, including SELECT, FROM, WHERE, LIMIT, and basic expressions. Joins, subqueries, and aggregation functions are not supported.
- No schema inference: S3 Select doesn't automatically infer the schema of your data. You need to explicitly define the data types for each column in your SQL query.

Transfer Acceleration

Amazon S3 Transfer Acceleration enables fast, easy, and secure transfers of files over long distances between your client and your Amazon S3 bucket. It leverages Amazon CloudFront's globally distributed AWS Edge Locations and is used to accelerate object uploads to S3 over long distances.

- Transfer Acceleration is as secure as a direct upload to S3.
- You are charged only if there has been a benefit in transfer times.
- You need to enable transfer acceleration on the S3 bucket.
- The URL is <bucketname>.s3-accelerate.amazonaws.com.
- You can use multipart uploads with transfer acceleration.
- Once enabled, Transfer Acceleration cannot be disabled. It can only be suspended.
- It may take up to 30 minutes to implement.
- It is not used for downloading data.

Pre-signed URLs

Pre-signed URLs can be used to provide temporary access to a specific object to those who do not have AWS credentials. When you create a presigned URL, you must provide your security credentials and then specify a bucket name, an object key, an HTTP method (PUT for uploading objects), and an expiration date and time.

- By default, all objects are private and can only be accessed by the owner. To share an object, you can either make it public or generate a pre-signed URL (using your own security credentials).
- The expiration date and time must be configured.

- It can be used for downloading and uploading S3 objects.

The credentials that you can use to create a presigned URL include:

- IAM instance profile: Valid up to 6 hours.
- AWS Security Token Service: Valid up to 36 hours when signed with permanent credentials, such as the credentials of the AWS account root user or an IAM user.
- IAM user: Valid up to 7 days when using AWS Signature Version 4.

If you created a pre-signed URL using a temporary token, then the URL expires when the token expires, even if you created the URL with a later expiration time.

Byte-range fetches

If an application running on Amazon EC2 needs to regularly download large objects from Amazon S3, then performance can be optimized for high throughput use cases by issuing parallel requests and using byte-range fetches. Using the Range HTTP header in a GET Object request, you can fetch a byte-range from an object and transfer only the specified portion. You can use concurrent connections to Amazon S3 to fetch different byte ranges from within the same object. This helps achieve higher aggregate throughput versus a single whole-object request. Fetching smaller ranges of a large object also allows your application to improve retry times when requests are interrupted.

Event notifications

Amazon S3 event notifications can be sent in response to actions in Amazon S3 like PUTs, POSTs, COPIEs, or DELETEs. They enable you to run workflows, send alerts, or perform other actions in response to changes in your objects stored in S3.

You can configure notifications to be filtered by the prefix and suffix of the key name of objects. You will need to grant Amazon S3 permissions to post messages to an Amazon SNS topic or an Amazon SQS queue. We also need to grant Amazon S3 permission to invoke an AWS Lambda function on your behalf.

Static Website

You can use Amazon S3 to host a static website. On a static website, individual web pages include static content. They might also contain client-side scripts. To host a static website on Amazon S3, you configure an Amazon S3 bucket for website hosting, and then upload your website content to the bucket. When you configure a bucket as a static website, you must enable website hosting, set permissions, and create and add an index document. Depending on your website requirements, you can also configure redirects, web traffic logging, and a custom error document. An S3 static website is ideal for this use case, and it will also be the most cost-effective option.

You cannot connect to an Amazon S3 static website using HTTPS (only HTTP). However, you can connect it using HTTPs if content is served through CloudFront, API Gateway, or ALB.

Amazon S3 Object Ownership

Amazon S3 Object Ownership has two modes: (see image below)

1. Object writer – The account that is writing the object owns the object.
2. Bucket owner preferred – The bucket owner will own the object if uploaded with the bucket-owner-full-control canned ACL. Without this setting and canned ACL, the object is uploaded to the bucket but remains owned by the uploading account.

After setting S3 Object Ownership to bucket owner preferred, you can add a bucket policy to require all Amazon S3 PUT operations to include the bucket-owner-full-control canned ACL. This ACL grants the bucket owner full control of new objects, and with the S3 Object Ownership setting it transfers object ownership to the bucket owner. If the uploader fails to meet the ACL requirement in their upload, the request will fail. This enables bucket owners to enforce uniform object ownership across all newly uploaded objects in their buckets.

S3 Object Lambda

S3 Object Lambda uses AWS Lambda functions to process the output of a standard S3 GET request automatically. AWS Lambda is a serverless compute service that runs customer-defined code without requiring management of underlying compute resources.

Monitoring

S3 Storage Lens

S3 Storage Lens delivers organization-wide visibility into object storage usage and activity trends. Usage metrics describe the size, quantity, and characteristics of your storage. S3 Storage Lens provides automated recommendations to help you optimize your storage.

Amazon S3 Storage Lens aggregates your usage and activity metrics and displays the information in an interactive dashboard on the Amazon S3 console or through a metrics data export that is downloaded in CSV or Parquet format.

Amazon S3 Storage Lens collects metrics and usage data for all AWS accounts that are part of your AWS Organization's hierarchy. To allow this, you must activate S3 Storage Lens trusted access using your AWS Organizations management. By enabling trusted access, you allow Amazon S3 Storage Lens to have access to your AWS Organizations hierarchy, membership, and structure through the AWS Organizations APIs. S3 Storage Lens will be a trusted service for your entire organization’s structure. You can then add delegated administrator access to accounts in your organization. These accounts can then create organization-wide dashboards and configurations for S3 Storage Lens.

S3 Storage Class Analysis

Amazon S3 Storage Class Analysis analyzes storage access patterns to help you determine when to transition less frequently accessed storage to a lower-cost storage class.

Pricing

- The bucket owner will only pay for object storage fees. On the other hand, the requester will pay for requests (uploads/downloads) and data transfers. Charges will be calculated at the bucket level.
- There is no charge for data transferred between EC2 and S3 in the same region.
- Data transferred to other regions is charged.

Charges are applicable on:

- Per GB/month storage.
- Data transfer out of S3.
- Upload requests (PUT and GET).
- Retrieval requests (S3-IA, S3 One Zone IA, or Glacier).
- Transfer acceleration pricing
- Data management and analytics pricing
- Price to process your data with S3 Object Lambda
- Amazon S3 pricing varies based on the AWS Region where it resides.

Limitations

- An object can have up to 10 tags.
- Amazon S3 is not a storage layer that can be mounted. Not suitable for hosting SMB file systems.
- **When uploading data via the AWS Management Console, the maximum file that you can upload is 160 GB. To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API.**

Edge and Hybrid Cloud Storage Devices

AWS Storage Gateway

Used for backing up data in your data center in AWS S3 storage. A VM is installed in the data center that does this job. Its purpose is to extend storage capabilities of your data center.

This service also enables hybrid storage b/w on-premises environments and AWS Cloud. It provides low-latency performance by caching frequently accessed data on-premises while storing data securely and reliably in Amazon cloud storage services. All data transferred is encrypted using SSL. By default, all data stored by AWS Storage Gateway in S3 is encrypted server-side with Amazon S3-Managed Encryption Keys.

Amazon S3 File Gateway

It connects on-premises NFS and SMB file shares to customer-managed Amazon S3 object storage.

Here are the two requirements that must be met before using Amazon S3 File Gateway:

1. Configure your private networking, VPN, or AWS Direct Connect between your Amazon Virtual Private Cloud (Amazon VPC) and the on-premises environment where you are deploying your gateway. This establishes a secure network connection, allowing the gateway to communicate with S3 and other AWS services while maintaining privacy and security.
2. Make sure your gateway can resolve the name of your AWS IAM Identity Center. This is crucial for the gateway to authenticate and interact with AWS services correctly. It typically involves configuring DNS settings or using a proxy server to ensure proper address resolution.

File Gateway

File Gateway optimizes on-premises access to fully managed, highly reliable file shares in Amazon FSx for Windows File Server. Customers with unstructured or file data, whether from SMB-based group shares or business applications, might require on-premises access to meet low-latency requirements. You can transfer your data using an AWS File Storage Gateway over the internet or over an AWS Direct Connect connection.

Volume Gateway

Represents a family of gateways that support block-based volumes. These volumes can be used to store data such as databases, file systems, and virtual machines. Data can be backed up in EBS, S3, and Glacier. Similar to Storage Gateway, it is used to connect on-premise storage to AWS.

It has two modes i) **Cached volume mode** - the entire dataset is stored on S3, and a cache of the most frequently accessed data is cached on-site. ii) **Stored volume mode** - the entire dataset is stored on-site and is asynchronously backed up to S3. Snapshots are incremental and compressed.

Tape Gateway

Gateway virtual tape library is used for backup with popular backup software. Each gateway is preconfigured with a media changer and tape drives. It is used for backing up data in S3 and archiving in Glacier using existing tape based processes. It can be public or private (VPC).

Snow Family

Snowball

Used for transporting a large amount of data between AWS and our own data center.

The AWS Snowball Client is software that is installed on a local computer and is used to identify, compress, encrypt, and transfer data. Tamper-resistant enclosures with TPM. Snowball uses a secure storage device for physical transportation.

Snowball is a petabyte-scale data transport solution for transferring data into or out of AWS via hardware device..

Snowball can be used for migration on-premise to on-premise.

Snowball Edge

Can store data of terabyte-scale. In this device, you can run your instances to process your data and when it reaches AWS all your instances and data can go to the cloud.

The Snowball Edge appliance can hold up to 80 TB of data, so 7 devices would be required to migrate 500 TB of data. For example, to transfer 500 TB of data in an on-premises file share to Amazon S3 Glacier, over low-bandwidth Internet connection within a few weeks, we can Order 7 AWS Snowball appliances, and select an Amazon S3 bucket as the destination. Create a lifecycle policy to transition the S3 objects to Amazon S3 Glacier.

- You can order some AWS Snowball Edge devices for a clustered implementation only when combined with a Local compute and storage job type.
- AWS Snowball Edge Storage Optimized for data transfer only are available for Import to Amazon S3 and Export from Amazon S3 job types. This device cannot be used for local compute and storage use cases. It must have compute devices but for data transfer purposes only e-g adding tags based on file types and user information.
- Amazon S3 API and NFS are transfer options.

You can order AWS Snowball Edge devices with compute capabilities as a cluster to increase durability and compute processing capabilities. You can order clusters for local compute and storage only jobs. AWS Snowcone devices are not available in a cluster configuration.

In summary, choose the Storage Optimized configuration if your primary need is to transfer large amounts of data efficiently. Opt for the Compute Optimized configuration if you require more processing power for compute-intensive workloads.

Snowmobile

A large trailer with semi-truck AWS staff comes with it and helps you to get data in exabyte-scale to truck then to AWS. You can transfer up to 100 PB per Snowmobile.

AWS Snowcone

AWS Snowcone is the smallest member of the AWS Snow Family of edge computing, edge storage, and data transfer devices, weighing in at 4.5 pounds (2.1 kg). Snowcone is ruggedized, secure, and purpose-built for use outside of a traditional data center. AWS Snowcone comes in an 8TB HDD version and an 14TB SSD version. Either device could meet the requirements

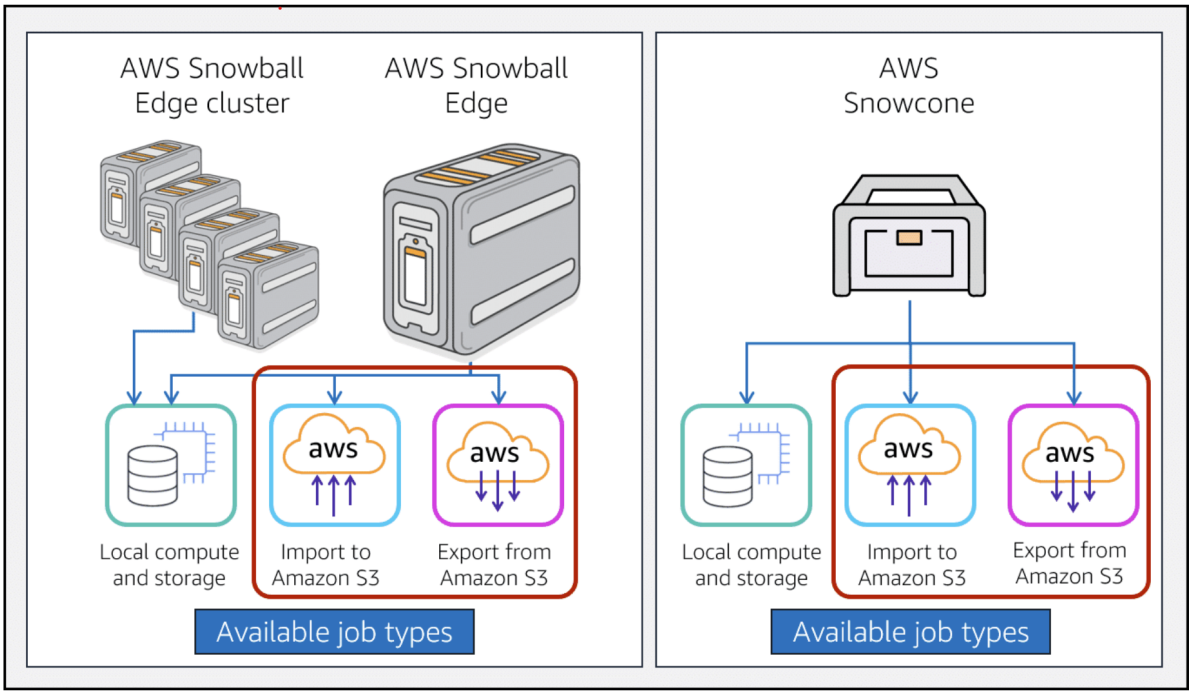
- AWS Snowcone includes AWS DataSync installed only with a Local compute and storage job type.

- AWS Snowcone is formatted for Network File System (NFS) storage with an Import to Amazon S3 job type. Snowcone is formatted for Amazon Elastic Block Store (Amazon EBS) storage with a Local compute and storage job type.
- NFS and AWS DataSync are transfer options.

AWS OpsHub

AWS OpsHub is an application for managing the AWS Snow Family devices, including AWS Snowcone. Use the OpsHub graphical user interface (GUI) to set up and manage AWS Snowcone devices. This way, you can rapidly deploy edge computing workloads and migrate data to the cloud, even when you don't have an internet connection.

Diagram



Select graphic to enlarge view

Security

All data on AWS Snowcone is automatically encrypted using 256-bit keys that you manage through AWS Key Management Service (AWS KMS). Encryption keys are never stored on the device. This helps ensure that your data stays secure during device transit.

Use Cases

- Local compute and storage only – Choose this option to perform compute and storage workloads on the device without any data transferred into AWS when returned. AWS erases all data on the device securely.

- Import into Amazon S3 – Choose this option to have AWS ship an empty Snowball Edge device to you. You can use this device for data collection and local processing. When you return the device to AWS, your data is uploaded to your Amazon S3 bucket selected during job creation. After your data is imported into Amazon S3, your data is securely erased from the device.
- Export from Amazon S3 – Choose this option to export data from your Amazon S3 bucket to your device. AWS loads your data on the device and ships it to you. After the device arrives, you copy data from your device to your local storage. When you are done, ship the device to AWS, and your data is securely erased from the device.
- With AWS Snowball Edge using self-managed compute operations, you can deploy and run machine learning models, such as document classification and image labeling, directly on the device. By doing this, you can tune processes, improve efficiency and productivity, and even anticipate model failures. Additionally, you can use Snowball Edge devices to transport data from remote or mobile locations to AWS for in-cloud machine learning.
- The ruggedized construction allows it to withstand vibrations, dust, and humidity in factory and other industrial environments.
- The battery power option allows for the device to be used in mobile workflows to capture sensor or machine data.

Pricing

You have on demand for shorter-term options or committed pricing for longer-term use cases. You pay a device rental fee for the time you have the device on site.

On-demand pricing includes a service fee per job. The fee includes a base number of days of device use and a per-day fee for every additional day you use the device before sending it back to AWS. The service fee and per-day fee vary by AWS Region and depend on which of the AWS Snowball Edge and AWS Snowcone device types that you choose.

If you know you will use an AWS Snowball Edge or AWS Snowcone device for one or three years, you can choose to pay the device fees upfront. Paying upfront provides a significant discount compared to on-demand pricing.

AWS Outposts

AWS Outposts is a fully managed service that offers the same AWS infrastructure, AWS services, APIs, and tools to virtually any data center, colocation space, or on-premises facility. These capabilities provide a consistent hybrid experience. AWS Outposts is ideal for the following:

- Workloads that require low latency access to on-premises systems, local data processing, and data residency
- Migration of applications with local system interdependencies

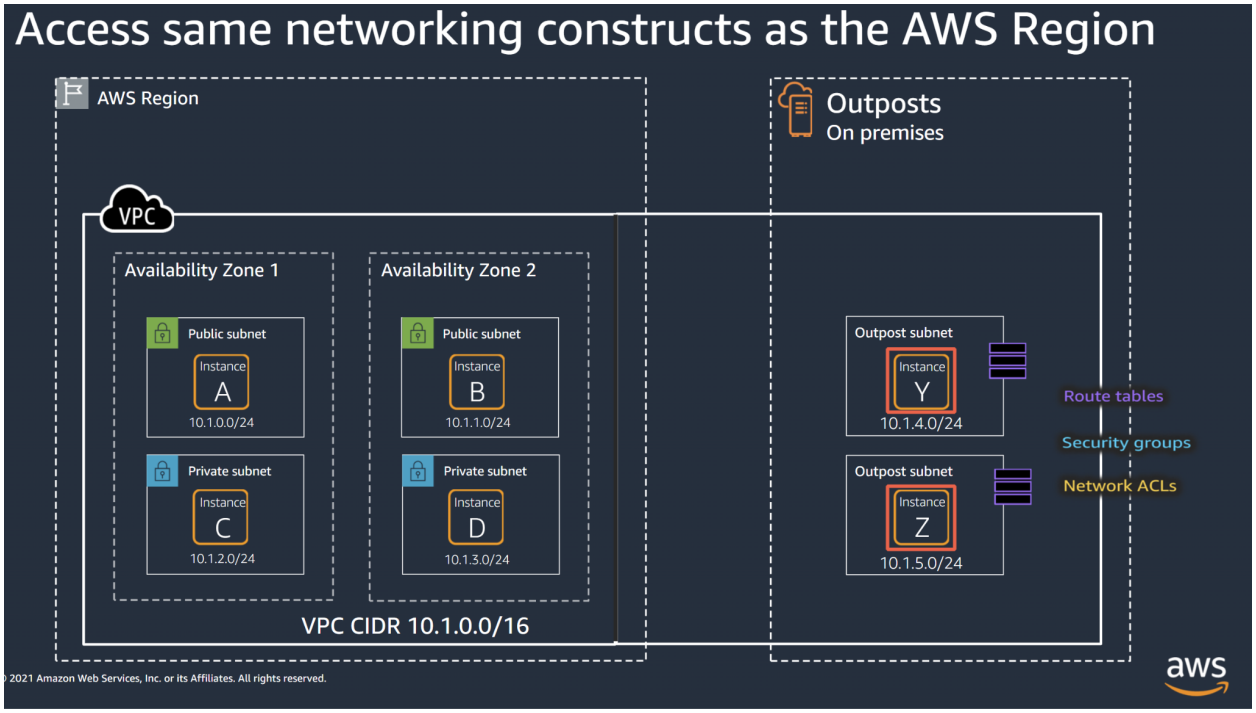
AWS compute, storage, database, and other services run locally on Outposts. You can access the full range of AWS services available in the Region to build, manage, and scale your on-premises applications using familiar AWS services and tools.

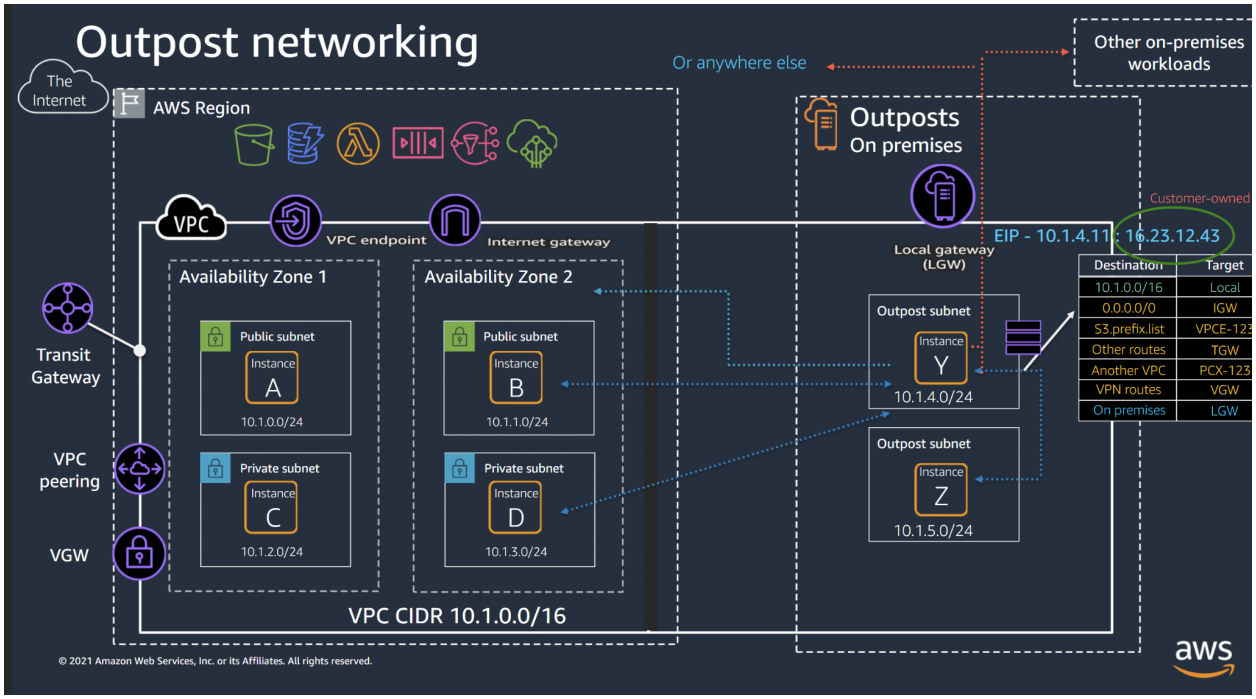
You can create subnets on your Outpost and specify them when you create AWS resources, such as the following:

- Amazon Elastic Compute Cloud (Amazon EC2) instances
- Amazon Elastic Block Store (Amazon EBS) volumes
- Amazon Elastic Container Service (Amazon ECS) clusters
- Amazon Relational Database Service (Amazon RDS) instances

Instances in Outpost subnets communicate with other instances in the AWS Region using private IP addresses, all within the same virtual private cloud (VPC).

- AWS installs your Outpost at your customer-managed physical buildings. A site must meet the facility, networking, and power requirements for your Outpost. Each configuration has unique power, cooling, and weight support requirements.
- An Outpost consists of physical hardware that provides access to the AWS Outposts service. It includes racks, servers, switches, and cabling that AWS owns and manages.
- An Outpost requires connectivity to an AWS Region. A service link is a network route that enables communication between your Outpost and its associated AWS Region. Each Outpost is an extension of an Availability Zone and its associated Region.





Security

Enhanced security with AWS Nitro – AWS Outposts builds on the AWS Nitro system technologies that enables AWS to provide enhanced security. AWS Nitro continuously monitors, protects, and verifies your Outpost instance hardware and firmware.

With AWS Nitro, virtualization resources are offloaded to dedicated hardware and software, minimizing the attack surface. Nitro system's security model is locked down and prohibits administrative access, eliminating the possibility of human error and tampering.

- Data at rest – Data is encrypted at rest by default on EBS volumes and S3 objects on Outposts.
- Data in transit – Data is encrypted in transit between Outposts and the AWS Region through the Service Link.
- Deleting data – All data is deleted when instances are terminated in the same way as in the AWS Region.

Pricing

You can choose from a variety of Outpost configurations, each providing a combination of EC2 instance types and EBS volumes. The pricing for these configurations includes the EC2 instances and EBS volumes, plus delivery, installation, and maintenance of the Outpost equipment. You can also increase your compute and storage capacity over time by upgrading your configuration. You can add Amazon S3 to your configuration for an additional fee.

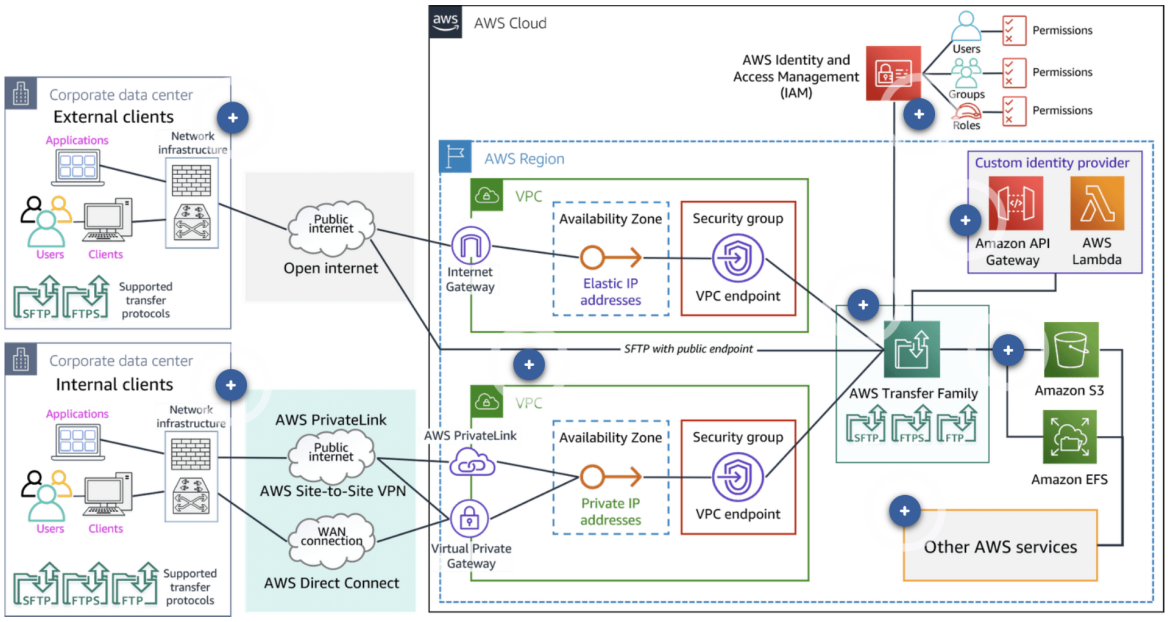
Data Transfer and Migration Services

AWS Transfer Family

The AWS Transfer Family provides fully managed support for file transfers directly into and out of Amazon S3 or Amazon EFS. It includes support for Secure File Transfer Protocol (SFTP), File Transfer Protocol over SSL (FTPS), and File Transfer Protocol (FTP).
The AWS Transfer Family helps you to migrate your file transfer workflows to AWS by doing the following so that nothing changes for you or your applications:

- Integrating with the specified authentication system
- Providing DNS routing with Amazon Route 53

Architecture Diagram



Pricing

With the AWS Transfer Family, you pay for only the protocols you have enabled for access to your endpoint and the amount of data transferred over each of these protocols. There are no upfront costs nor resources you have to manage yourself.

AWS DataSync - online transfer service

AWS DataSync makes it simple and fast to move large amounts of data online between on-premises storage and AWS storage services. DataSync comes with built-in monitoring and retry mechanisms and granular control over the portion of network bandwidth used to transfer your data.

All of your data is encrypted in transit with Transport Layer Security (TLS). For each transfer, the service performs integrity checks both in transit and at rest. These checks make sure that the data written to your destination matches the data read from your source, validating consistency.

DataSync preserves metadata between storage systems that have similar metadata structures, enabling a smooth transition of users and applications to your target AWS Storage service.

AWS DataSync supports asynchronous or one-direction at a time transfers between on-premises file systems to supported AWS Storage services in the AWS Cloud. DataSync also supports asynchronous data transfers between supported AWS Storage resources within the AWS Cloud.

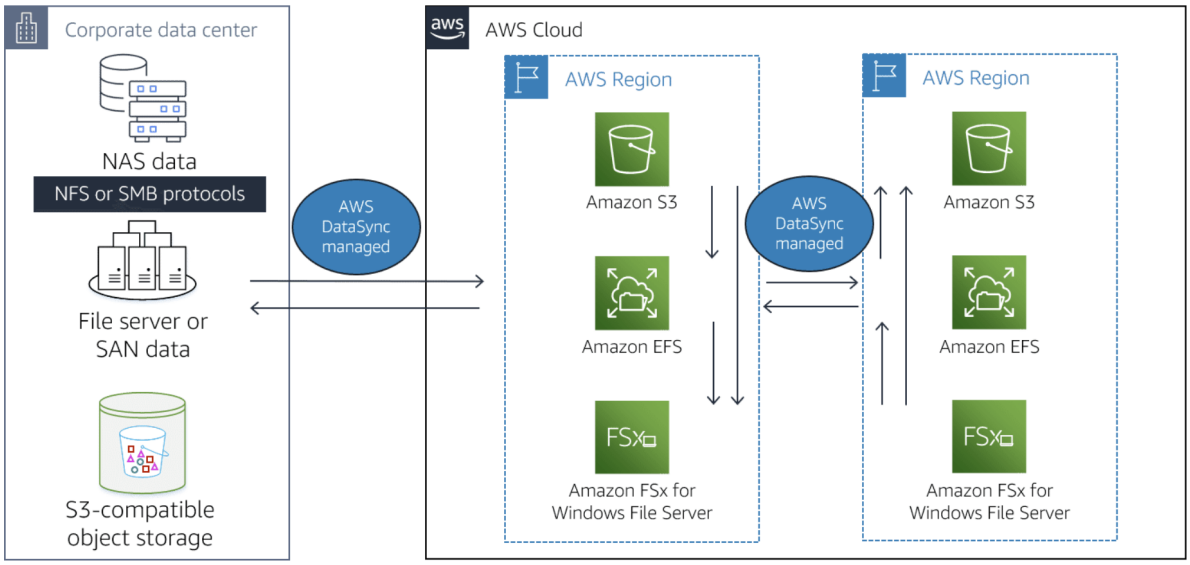
DataSync can copy data between the following resources or services:

- Self-managed object storage
- AWS Snowcone
- Amazon S3 buckets
- Amazon EFS file systems
- Amazon FSx for Windows File Server file systems

Manual tasks related to data transfers can slow down migrations and burden IT operations. DataSync eliminates or automatically handles many of these tasks, including scripting copy jobs, scheduling and monitoring transfers, validating data, and optimizing network utilization.

Task scheduling enables you to configure a task periodically to detect and copy changes from your source storage system to the destination. You can schedule your tasks using the AWS DataSync Console or AWS Command Line Interface (CLI) without writing and running scripts to manage repeated transfers.

Architecture Diagram



Transfer between on-premises systems and AWS Storage services or between AWS Storage services

Pricing

AWS DataSync has predictable, usage-based pricing. With DataSync, you pay for only the amount of data that you copy. Your costs are based on a flat per-gigabyte fee for the use of network acceleration technology, managed cloud infrastructure, data validation, and automation capabilities in DataSync. You are not required to manage resources or pay upfront costs or a minimum fee.

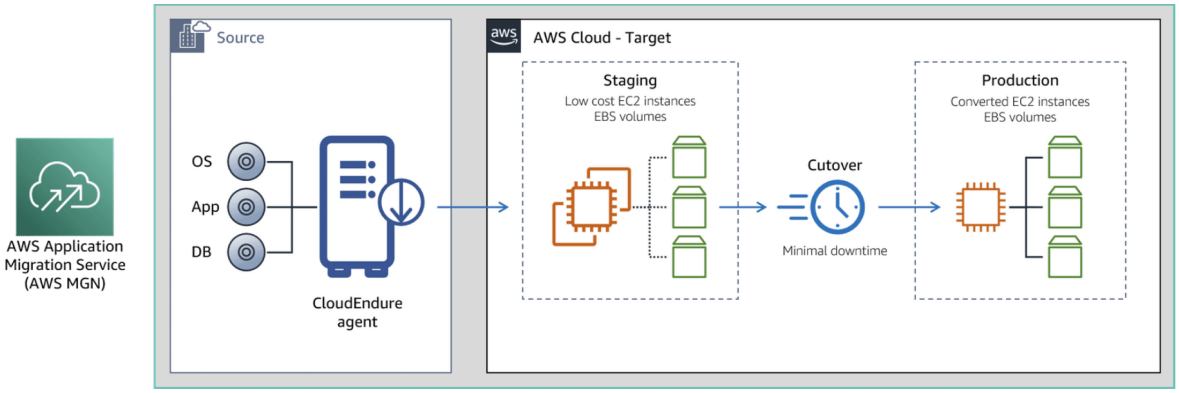
Offline data transfer

Offline data transfers are performed using AWS Snow Family devices. The AWS Snow Family helps customers who need to run operations in austere, non-datacenter environments, and in locations where there's lack of consistent network connectivity.

AWS Application Migration Services

AWS Application Migration Service (AWS MGN), which includes CloudEndure Migration, is a highly automated lift-and-shift (rehost) solution. AWS MGN simplifies, expedites, and reduces the cost of migrating applications to the AWS Cloud, AWS GovCloud (US), and AWS Outposts. You can use AWS MGN or CloudEndure Migration by itself to quickly lift-and-shift physical, virtual, or cloud servers without compatibility issues, performance impact, or long cutover windows. AWS MGN continuously replicates your source servers to your AWS account. When you're ready to migrate, it automatically converts and launches your servers on AWS.

CloudEndure Migration uses Amazon EC2 instances and Amazon EBS volumes. CloudEndure Migration copies and updates in real time your operating systems, applications, and data from your on-premises application servers to the AWS Cloud.



Select graphic to enlarge view

AWS Application Migration Service and CloudEndure Migration

AWS MGN provides similar capabilities as CloudEndure Migration, but it is available on the AWS Management Console. Because AWS MGN is available from the console, it can integrate easily with other AWS services, such as AWS CloudTrail, Amazon CloudWatch, and IAM. Currently, AWS MGN does not support all of the AWS Regions or operating systems that CloudEndure Migration supports.

Pricing

For each source server that you want to migrate, you can use AWS MGN for a free period of 2,160 hours, which is 90 days when used continuously. The free period starts as soon as you install the AWS Replication Agent on your source server and continues during active source server replication. If you do not complete your migration of a specific server within the free period, you will be charged per hour while you continue replicating that server. Most customers complete migrations of servers within the allotted free period.

AWS Migration Evaluator

- It's a service that helps you build a data-driven business case for migrating your on-premises workloads to AWS.
- It analyzes your on-premises infrastructure and applications to estimate the costs, risks, and benefits of migration.
- It can help you identify cost savings, performance improvements, and other benefits of moving to AWS.
- Features:
    - Cloud cost estimation: estimates the cost of running your workloads on AWS.
    - Migration path recommendation: recommends different migration patterns based on your workload and requirements.
    - Detailed reports: provides detailed reports on your migration potential.

Data Protection Services

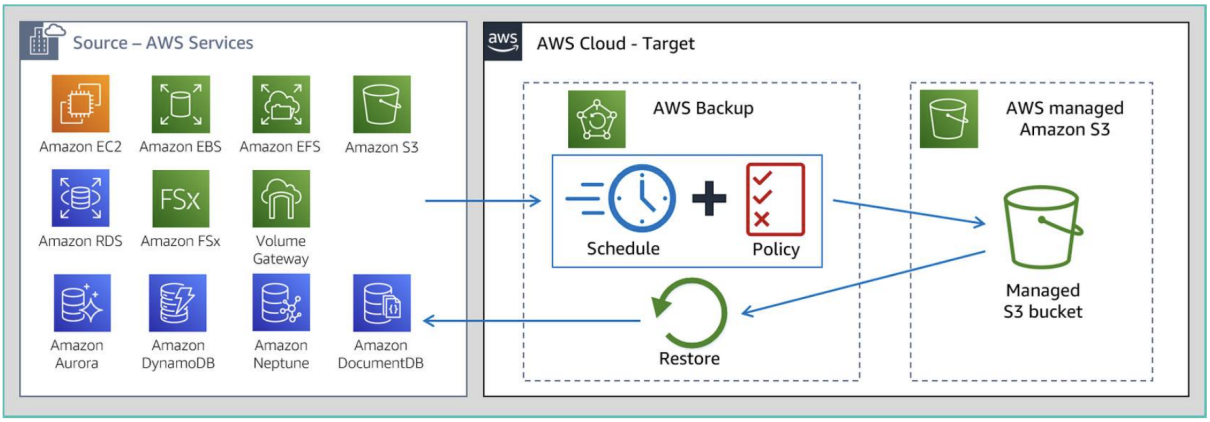
AWS Backup

Using AWS Backup, you can centralize and automate data protection across AWS services. AWS Backup offers a cost-effective, fully managed, policy-based service that further simplifies data protection at scale. AWS Backup also helps you support your regulatory compliance or business policies for data protection.

When you combine AWS Organizations with AWS Backup, you can deploy data protection policies centrally. Centrally deploy policies to configure, manage, and govern your backup activity across your company's AWS accounts and resources.

AWS Backup provides additional data durability by creating additional copies of your data. You can store copies of backups for as long as you are required to retain your data. Some compliance regulations require retention and immutable backup copies of your data.

Architecture Diagram



Pricing

AWS Backup storage pricing is based on the amount of storage space your backup data consumes. For the first backup of an AWS resource, a full copy of your data is saved. For each incremental backup, only the changed part of your AWS resource is saved.

Snapshots

Native snapshot services are built into most core services. Snapshots create backup copies of your data. Snapshots are stored in a protected part of Amazon S3 as part of the managed service. Storing snapshots on Amazon S3 protects your data with 99.999999999 percent (11 9s) of durability and provides you Regional access and availability.

Snapshots are incremental copies of the data, which means that only the data that has changed after your most recent snapshot is saved in the next incremental snapshot. Incremental snapshots reduce the time required to create the snapshot. These incremental snapshots save on storage costs by not duplicating previously saved data. Each snapshot contains all of the information for that point in time that is needed to restore your data.

Amazon EBS snapshot events are tracked through CloudWatch events. An event is generated each time you create a single snapshot or multiple snapshots, copy a snapshot, or share a snapshot.

With Amazon EBS snapshots, you can create backup copies of critical workloads, such as a large database or a file system that spans across multiple EBS volumes. Multi-volume snapshots let you take exact point-in-time, data-coordinated, and crash-consistent snapshots across multiple EBS volumes attached to an EC2 instance.

Pricing

With snapshots, you are charged only for the storage capacity taken up by the full and incremental snapshots. Pay only for the used backup capacity and the duration your data is retained. You are charged only for the backup capacity used and not your provisioned file system capacity. Pricing is based on a per GB-month.

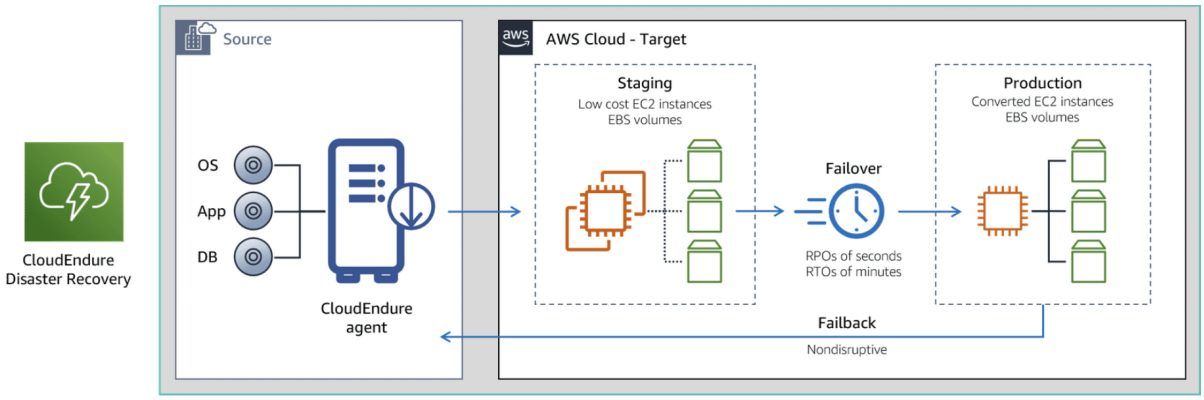
Replication

Storage replication is an available built-in feature for some of the core storage services. How replication is implemented varies for each service. Replication increases availability and protects your data by creating additional copies. Replication can be between Availability Zones within an AWS Region or between AWS Regions.

Disaster Recovery Services

CloudEndure Disaster Recovery continuously replicates your machines into a low-cost staging area in your target AWS account and preferred Region. Replication also includes operating system, system state configuration, databases, applications, and files. In the case of a disaster, you can instruct CloudEndure Disaster Recovery to automatically launch thousands of your machines in their fully provisioned state in minutes.

CloudEndure Disaster Recovery minimizes downtime and data loss by providing fast, reliable recovery of physical, virtual, and cloud-based servers into AWS Cloud, including AWS Regions, AWS GovCloud (US), and AWS Outposts.



Pricing

CloudEndure Disaster Recovery is billed hourly per source server registered, irrespective of provisioned storage capacity. This provides flexibility to easily use the disaster recovery solution on an hourly basis without long-term contracts or a minimum number of servers.
Pricing includes continuous data replication, virtually unlimited disaster recovery drills, point-in-time recovery, and automated failover and failback.

Terminologies

Durability: It refers to average annual expected data loss.

Amazon Macie

It is used to check your entire suite of applications for personally identifiable information. It is a security service that uses machine learning to automatically discover, classify, and protect sensitive data in AWS. It provides customers' with dashboards and alerts that give visibility on how data is being accessed or moved.

You can use Amazon Macie to discover and protect sensitive data stored in Amazon S3. Macie gathers a complete S3 inventory automatically and continuously evaluates every bucket to alert on any of the following:

- Any publicly accessible buckets
- Unencrypted buckets
- Buckets shared or replicated with AWS accounts outside of your organization

Databases

Amazon Relational Database Service (Amazon RDS)

It is a managed database service and used for managing relational databases in the cloud. Similar to EC2, we choose DB instance type for creating RDS instances to determine how much power and memory it will have. Amazon RDS stores data in EBS volumes. However, when using Amazon Aurora, data is stored in cluster volumes, which are single, virtual volumes that use SSDs. A cluster volume copies data across multiple AZs in a region. For temporary storage Aurora uses local storage.

Whenever you create a RDS instance. It is created inside a subnet of a VPC. This subnet should be private.

During failover, RDS automatically updates configuration (including DNS endpoint) to use the second node. The process of failover is not reliant on network connectivity as it is designed for fault tolerance.

You can create and modify Amazon RDS database instances by using the CLI, Amazon RDS API, or console.

Basic building block of RDS is the DB instance. DB instance uses EC2 instance underneath but you cannot SSH into it like EC2 instance.

Once your database instance is available, you can retrieve its endpoint via the database instance description in the AWS Management Console, DescribeDBInstances API or describe-db-instances command.

Availability

Multi-AZ

The Multi-AZ deployment instance is deployed in multiple AZs and RDS manages data replication.

- RDS with Multi-AZ does not span regions.
- You can create a Read Replica as a Multi-AZ DB instance. Amazon RDS creates a standby of your replica in another Availability Zone for failover support for the replica. Creating your Read Replica as a Multi-AZ DB instance is independent of whether the source database is a Multi-AZ DB instance or not.
- Multi-AZ is used for implementing fault tolerance. With Multi-AZ, you can failover to a DB in another AZ within the region in the event of a failure of the primary DB.
- You cannot scale write-capacity by enabling Multi-AZ as only one DB is active and can be written to.

RDS also manages failover, initially one instance is treated as primary while other as secondary. Both of these instances are in different AZs. When a DB instance is created a DNS name is provided and AWS uses it to failover to the standby database.

Replicated storage helps to increase availability and performance and it maintains consistency of data in multi-AZ deployments.

Read Replica

- Read Replicas must be explicitly deleted. If a source DB instance is deleted without deleting the replicas, each replica becomes a standalone single-AZ DB instance.
- For Read Replicas to work, you must have automated backups enabled on the primary (retention period > 0).
- Amazon RDS Read Replicas are used for read-heavy DBs. Read Replicas are for workload sharing and offloading.

Multi-AZ vs Read Replica

Multi-AZ	Read Replica
Multi-AZ deployment has synchronous replication - highly durable.	Read Replica has asynchronous replication - highly scalable.
In Multi-AZ only the primary instance is active at any point in time.	In Read Replica all replicas are active and can be used for read scaling.
Backups can be taken from secondary in Multi-AZ.	In Read Replica, no backups are configured by default.
Multi-AZ is always in two AZs within a region.	A Read Replica can be within an AZ, cross-AZ or cross-region.
Database engine version upgrades take place only on a primary instance in Multi-AZ.	In Read replica, db engine version upgrades are independent from source instance.
Automatic failover in Multi-AZ.	In Read Replica, failover can be manually promoted to a standalone database.

Backup

We select a time period for the backup. Automated backups are retained from 0 to 35 days. 0 means no backup. Data can be restored at point in time snapshots.

Manuals are used for keeping backup for more than 35 days. They persist until manually deleted.

The amount of data in the database impacts the time required to create a snapshot. In extreme instances, the DB instance may suffer in performance during snapshot creation. For this reason, it is best to perform snapshots during times of low usage.

Scalability

- You can only scale RDS up (compute and storage).
    - To handle a higher load in your database, you can vertically scale up your master database with a simple push of a button.
- RDS uses EC2 instances so you have to change your instance type/size in order to scale and compute vertically.
- You can scale storage and change the storage type for all DB engines except MS SQL. For MS SQL, the workaround is to create a new instance from a snapshot with the new configuration.
- Scaling storage can happen while the RDS instance is running without outage; however, there may be performance degradation.
- Scaling the compute will cause downtime. You can choose to have changes take effect immediately; however, the default is within the maintenance window.

Maintenance

Some maintenance items require that Amazon RDS take your DB instance offline for a short time. Maintenance items that require a resource to be offline include required operating system or database patching. Required patching is automatically scheduled only for patches that are related to security and instance reliability.

Enabling Multi-AZ, promoting a Read Replica, and updating DB parameter groups are not events that take place during a maintenance window.

Monitoring

RDS uses CloudWatch to store the Enhanced Monitoring statistics, by default, for 30 days.

Configuration

The Modify option allows you to configure many of the settings configured at creation time (with the important exception of encryption).

To ensure that you create only databases supported in the free tier. Only enable options eligible for RDS Free Usage Tier. Even with this selected, if you surpass storage and CPU usage limits, you could still be charged.

Performance

Amazon RDS uses EBS volumes (never uses instance store) for DB and log storage. There are three storage types available:i) General Purpose SSD, ii) Provisioned IOPS SSD, and Magnetic (standard).

Rapid ingestion of dynamic data is not an ideal use case for RDS.

Security

- Network ACLs and Security Groups are used to control access to the database. IAM policies are used to manage clients' access (who can create, modify, delete, etc DB instances) to the database.
- When a DB instance is first created all database access is prevented., except through rules specified by an associated Security Group.
- Use Secure Socket Layer (SSL) and Transport Layer Security (TLS) connections with DB instances running the DB engines (MySQL, MariaDB, PostgreSQL, Oracle, SQL Server database engines).
- Ensure security groups and NACLs allow your application servers to communicate with both the primary and standby instances.
- A DB subnet group is a collection of subnets (typically private) that you create in a VPC and that you then designate for your DB instance. The DB subnet group cannot be made publicly accessible; even if the subnets are public subnets.
- Amazon RDS does not have their own security groups. It uses EC2, VPC, and Database security groups to control access to an Amazon RDS database instance.
- Although STS can generate temporary tokens, it is not compatible with Amazon RDS. To validate that RDS can only be accessed using an instance profile for Amazon EC2 instance via authentication token and for enabling IAM token based authentication, you will need to enable AWS IAM DB authentication.

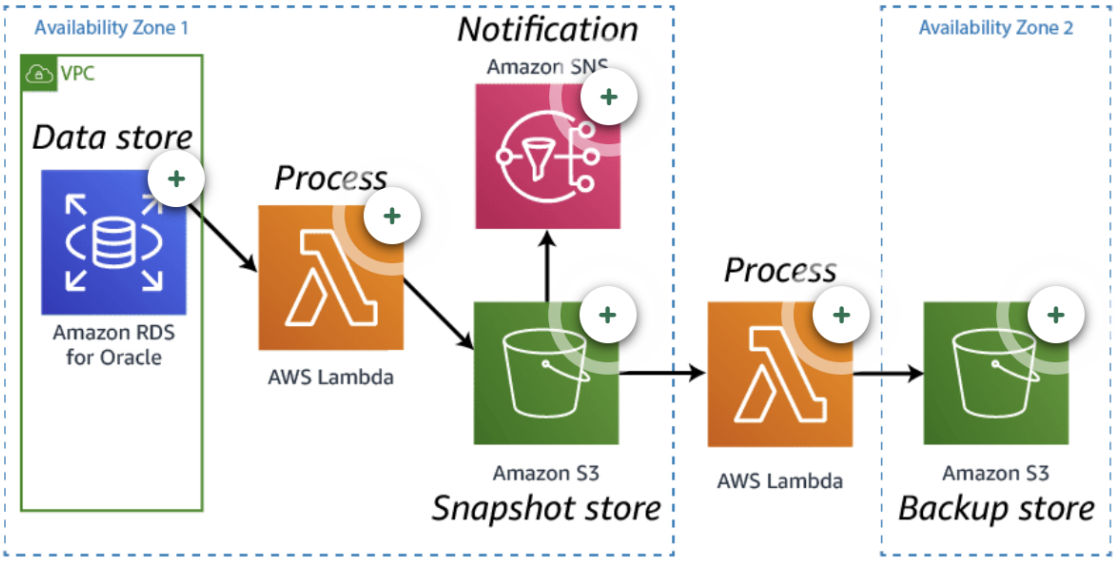
Encryption

- Amazon RDS encryption can be used to secure DB instances and snapshots at rest.
    - A read replica of an Amazon RDS encrypted instance is also encrypted using the same key as the master instance when both are in the same region. If the master and Read Replica are in different regions, you encrypt using the encryption key for that region.
- You cannot encrypt an existing DB; you need to create a snapshot, copy it, encrypt the copy and then build an encrypted DB from the snapshot. Finally, switch your connections to the new DB instance.

- RDS databases do support encryption; however, it must be enabled during database creation. The only other option is to backup the database and then recover it with encryption. However, database recovery is really the creation of a new database and that is why encryption can be enabled during recovery.
- You can't have an encrypted Read Replica of an unencrypted DB instance or an unencrypted Read Replica of an encrypted DB instance.

Disaster Recovery Architecture

This architecture is one option for creating a disaster recovery solution for the databases.



Real-time data analytics architecture

Pricing

RDS incurs charges based on the total number of hours the instance remains active, irrespective of whether the system is actively utilized during those hours or not.

- Storage GB/month.

- I/O requests/month — for magnetic storage.
- Provisioned IOPS/month — for RDS provisioned IOPS SSD.
- Egress data transfer.
- Backup storage (DB backups and manual snapshots).
- Data transfer is charged, if data goes outside of the region (including via the internet).

Backup storage for the automated RDS backup is free of charge up to the provisioned EBS volume size. For Multi-AZ, you are not charged for DB data transfer during replication from primary to standby. There is no charge for data transfer between primary and secondary RDS instances.

Migration

Snapshots and Log shipping are not a supported migration method from RDS to RedShift.

Restoration

When you restore a DB instance to a point in time, the default DB security group is applied to the new DB instance. If you need custom DB security groups applied to your DB instance, you must apply them explicitly using the AWS Management Console, the AWS CLI modify-db-instance command, or the Amazon RDS API ModifyDBInstance operation after the DB instance is available.

Restored DBs will always be a new RDS instance with a new DNS endpoint, and you can restore up to the last five minutes.

Limitations

- There is no such thing as a Multi-Master MySQL RDS DB (there is for Aurora).
- You cannot offload data from Amazon RDS to DynamoDB.
- Instance Type Changes: Modifying the instance type of an RDS DB instance typically requires downtime.
    - Scaling vertically (up or down) within the same DB instance class might be possible without downtime in some cases.
    - Aurora offers "Fast Scaling" for certain instance types, minimizing downtime to a few seconds.
- Auto Scaling is supported in RDS but not automatically configured, but you need to configure auto scaling policy for it to work.
- Read Replica Limitations:
    - Read replicas can't be used as standalone databases.
    - They can't be promoted to primary instances in most cases.

Amazon DynamoDB

- Fully managed NoSQL database.

- Stores key/value pair or document data.
- Stores data within a single table without a predefined schema.
- Good for high-scale and serverless applications.
- It can support OLTP (Online transaction processing) workloads.
- Data is organized into items and each item has attributes. A table is a collection of items and each item is a collection of attributes.
- All data is stored on SSDs and replicated across multiple AZs in a region.
- DynamoDB is a web service that uses HTTP over SSL (HTTPS) as transport and JSON as a message serialization format.
- Provisioned throughput is the maximum amount of capacity that an application can consume from a table or index. It doesn't improve the speed of the database or add in-memory capabilities.
- It is ACID compliant.
- Relational databases are comprised of tables, records, and fields. DynamoDB is a non-relational database comprised of tables, items, and attributes.
- Amazon DynamoDB Time to Live (TTL) allows you to define a per-item timestamp to determine when an item is no longer needed. Shortly after the date and time of the specified timestamp, DynamoDB deletes the item from your table without consuming any write throughput. TTL is provided at no extra cost as a means to reduce stored data volumes by retaining only the items that remain current for your workloads needs.(If automatic, fine-grained TTL is a crucial requirement, DynamoDB might be a better fit for your use case. If you need relational database features and can manage data expiration with external tasks or application logic, RDS can still be a viable option.)

How data is stored ?

Data is distributed on physical storage nodes. DynamoDB uses the partition key to determine which of those nodes the item is located on. DynamoDB items can have an optional sort key to store related attributes in a sorted order. This allows multiple items to be queried as a collection, which simplifies access patterns.

Each table also has a primary key, which represents the table's key or keys. If there is no sort key, the primary and partition keys are the same. If there is a sort key, the primary key is a combination of the partition and sort keys called a composite primary key.

Indexes

DynamoDB has two types of secondary indexes: local and global. These indexes improve the application's ability to access data quickly and efficiently.

A local secondary index uses the table's partition key with a unique sort key. You are allowed five per table. Local indexes must be created when you create the table.

A global secondary index uses a partition key and sort key that can be different from those on the table. This allows you to model very complex data access patterns that differ from the

original table. You are allowed up to 20 global indexes per table. Global indexes can be created and edited at any time.

Throttling

Throttling is the action of limiting the number of requests that a client can submit to a given operation in a given amount of time. Throttling prevents your application from consuming too many capacity units. When a request is throttled, it fails with an HTTP 400 Bad Request error and a **ProvisionedThroughputExceededException**.

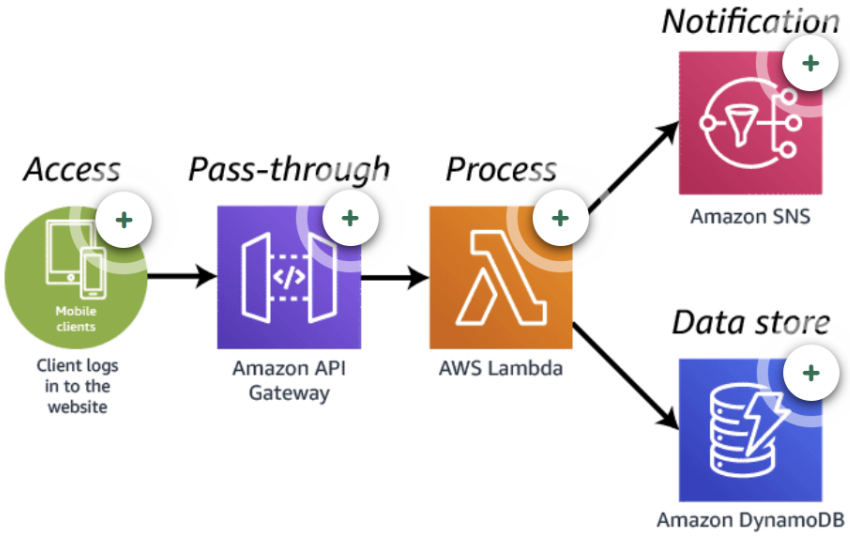
Security

- Can be used as primary data storage for mission critical applications.
- All data stored is encrypted at rest and in transit by default using keys stored in AWS KMS.
- It uses AWS Identity and Access Management, or IAM, to create and manage credentials. The same users and roles you have today in IAM can be used with DynamoDB. DynamoDB requires both authentication and permission to access tables and data. IAM allows you to control access at the table and item levels.

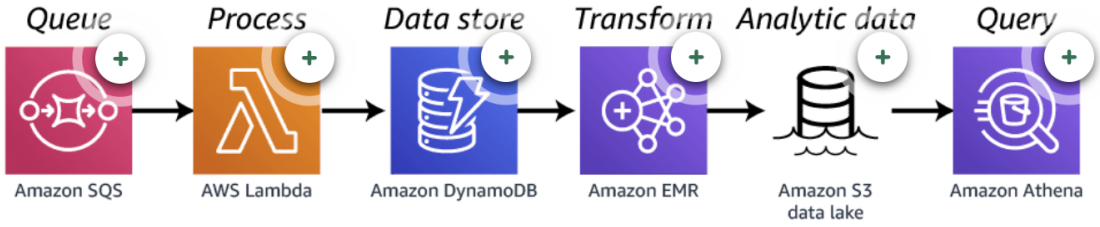
Use cases

- Highly concurrent and high traffic applications (e-g gaming platforms) that require concurrent access to data.
- User content metadata storage.
- Media metadata storage.
- It can be used for rapid ingestion of dynamic data.
- DynamoDB is designed for intentional interactions with many different AWS services. The service is an excellent database solution for transactional workloads.

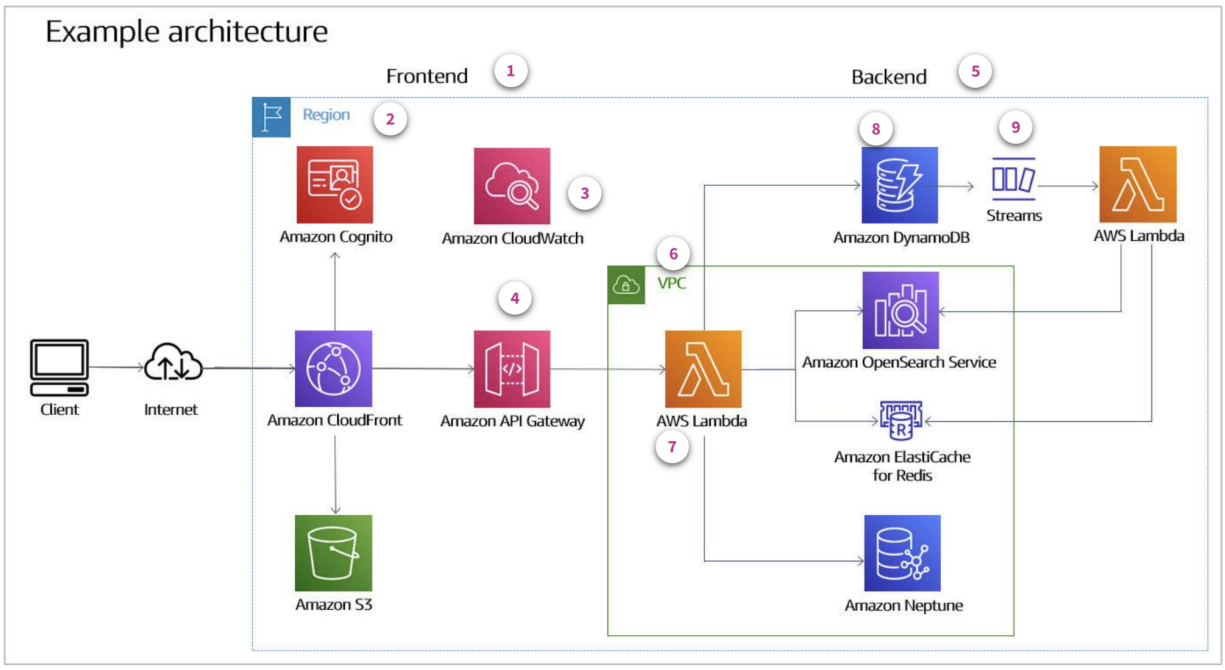
Mobile Application backend architecture



IOT sensor data capture architecture



Web Application Architecture



Pricing

Pricing is based on usage and amount of data reading from the table. Priced on throughput rather than compute. On-Demand Capacity provides flexible capacity at a small premium cost.

The cost for DynamoDB throughput is $0.25 per GB and is billed monthly.

On-demand capacity mode

DynamoDB charges you for the data reads and writes your application performs on your tables. You do not need to specify how much read and write throughput you expect your application to perform because DynamoDB instantly accommodates your workloads as they ramp up or down.

Provisioned capacity mode

You specify the number of reads and writes per second that you expect your application to require.

Read Capacity Units

One read capacity unit (RCU) represents one strongly consistent read per second, or two eventually consistent reads per second, for an item up to 4 KB in size. Transactional read requests require two RCUs to perform one read per second for items up to 4 KB. If you need to read an item that is larger than 4 KB, DynamoDB must consume additional RCUs. The total

number of RCUs required depends on the item size and whether you want an eventually consistent or strongly consistent read.

Write Capacity Units

One write capacity unit (WCU) represents one write per second for an item up to 1 KB in size. If you need to write an item that is larger than 1 KB, DynamoDB must consume additional WCUs. Transactional write requests require two WCUs to perform one write per second for items up to 1 KB. The total number of WCUs required depends on the item size.

DynamoDB Streams

DynamoDB Streams help you keep a list of item level changes or provide a list of item level changes that have taken place in the last 24 hours.

Amazon DynamoDB is integrated with AWS Lambda (you can associate the stream ARN with a Lambda function that you write) so that you can create triggers which are pieces of code that automatically respond to events in DynamoDB Streams. Immediately after an item in the table is modified, a new record appears in the table's stream. AWS Lambda polls the stream and invokes your Lambda function synchronously when it detects new stream records.

Amazon Kinesis can also be configured to respond to events in DynamoDB Streams and for example, it can be used to deliver streaming data to an S3 datalake from DynamoDB.

Using Amazon Kinesis has less operational overhead than using Lambda functions to process DynamoDB stream, this is important when both can be used.

Amazon DynamoDB Accelerator (DAX)

It is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to 10X performance improvements from milliseconds to microseconds.

- You can apply an IAM role to the DAX nodes.
- You can apply Security Groups to the DAX nodes.
- DynamoDB DAX sits within your VPC.
- Pricing is per node-hour consumed and is dependent on the instance type you select.

DynamoDB Auto Scaling Policy

Amazon DynamoDB Auto Scaling uses the AWS Application Auto Scaling service to dynamically adjust provisioned throughput capacity on your behalf in response to actual traffic patterns. This is the most efficient and cost-effective solution to optimizing for cost in case of variable load.

Best Practices

- Store more frequently and less frequently accessed data in separate tables.
- If possible, compress larger attribute values.
- Store objects larger than 400KB in S3, and use pointers (S3 Object ID) in DynamoDB.

- Keep item sizes small.
- If you are storing serial data in DynamoDB that will require actions based on data/time, use separate tables for days, weeks, and months.

Integration

ElastiCache can be used in front of DynamoDB for improving the performance of reads on infrequently changed data. Triggers integrate with AWS Lambda to respond to triggers.
Integration with RedShift

- RedShift complements DynamoDB with advanced business intelligence.
- When copying data from a DynamoDB table into RedShift, you can perform complex data analysis queries, including joins with other tables.

DynamoDB is integrated with Apache Hive on EMR. Hive can allow you to:

- Read and write data in DynamoDB tables allowing you to query DynamoDB data using a SQL-like language (HiveQL).
- Copy data from a DynamoDB table to an S3 bucket and vice versa.

DynamoDB global tables

Amazon DynamoDB global tables provide a fully managed solution for deploying a multiregion, multi-master database without having to build and maintain your own replication solution. With global tables, you can specify the AWS Regions where you want the table to be available. DynamoDB performs all of the necessary tasks to create identical tables in these Regions and propagate ongoing data changes to all of them. They enable you to deliver low-latency data access to your users no matter where they are located. DynamoDB global tables are ideal for massively scaled applications with globally dispersed users.

If your application requires strongly consistent reads, it must perform all of its strongly consistent reads and writes in the same region. It is important that each replica table and secondary index in your global table have identical write capacity settings to ensure proper data replication.

Limitations

- DynamoDB does not support strongly consistent reads across AWS Regions.
- You cannot create an AWS VPN connection to the Amazon DynamoDB endpoint.
    - VPC Endpoints: Use VPC Endpoints for DynamoDB to create a private connection within your VPC, enhancing security and reducing latency.
    - AWS PrivateLink: For private connectivity across accounts or VPCs, leverage AWS PrivateLink to securely access DynamoDB without a public endpoint.
- Limited Joins: DynamoDB has limited support for complex joins across tables.
- No Stored Procedures: It doesn't support stored procedures or user-defined functions.
- Item Size: Maximum item size is 400 KB.
- Throughput Capacity: Carefully plan and manage provisioned throughput to avoid throttling.

Amazon DocumentDB

Amazon DocumentDB (with MongoDB compatibility) is designed from the ground up to give you the performance, scalability, and availability you need when operating mission-critical MongoDB workloads at scale. In Amazon DocumentDB, the storage and compute are decoupled, allowing each to scale independently.

The cluster's data is stored in the cluster volume, which stores six copies of your data across three different Availability Zones. All writes are done through the primary instance. All instances (primary and replicas) support read operations.

The heart of Amazon DocumentDB is a purpose-built document database engine. Data is stored in the form of documents. These documents are stored into collections. Each document can have a unique combination and nesting of fields or key-value pairs.

It is good for content management, profile management applications, etc

Security

Amazon DocumentDB uses the 256-bit Advanced Encryption Standard (AES-256) to encrypt your data. Storage encryption is enabled cluster-wide and is applied to all instances, including the primary instance and any replicas, as well as your cluster’s storage volume. Amazon DocumentDB also allows you to encrypt your clusters using keys you manage through AWS Key Management Service (AWS KMS). Data encrypted at rest includes your cluster’s data, indexes, logs, automated backups (if enabled), replicas, and snapshots.

Backup

Amazon DocumentDB provides automatic, continuous, and incremental backups and point-in-time restore. There is no additional charge for backup storage of up to 100% of your total Amazon DocumentDB cluster storage for a Region. Additional backup storage is billed in per GB-months.

Pricing

With on-demand pricing and no up-front or long-term commitments, with Amazon DocumentDB you only pay for capacity you use.

On-Demand Instances let you pay by the second. Pricing is per instance-hour or per partial instance-hour consumed from the time you launch an instance until you delete it. Storage auto scales from 10GB up to 64TB with no interaction necessary. You only pay for what you consume, and your cluster is billed in per gigabyte per month increments.

Now you are also billed for IOPS (Input/Output Operations Per Second) are pay as you go, and IOPS consumed are billed in per million request increments. There is no charge for data

transferred into your Amazon DocumentDB database. Data transferred out of the database is charged per gigabyte per month.

Amazon ElastiCache

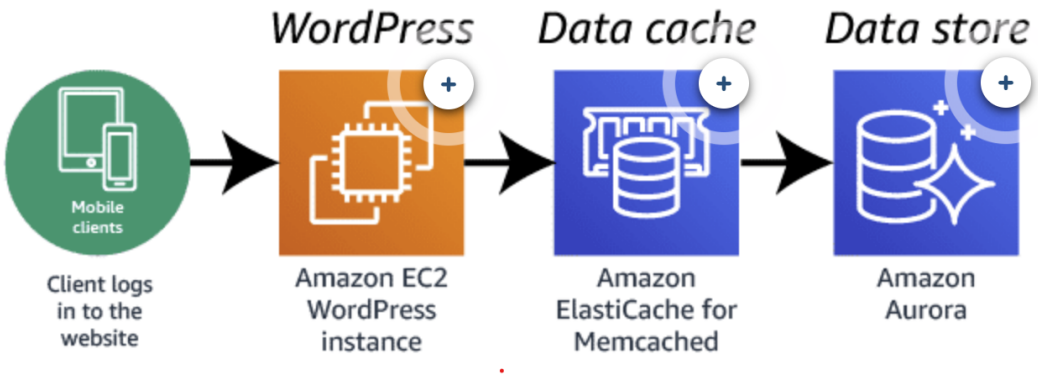
- A fully managed in-memory caching solution.
- Provides support for Redis and Memcached.
- Supports master/slave replication and multi-AZ deployments.
- With ElastiCache we create a cluster of caching servers, by default Redis is selected.

ElastiCache offers push-button scalability for memory, writes, and reads.

The in-memory caching provided by ElastiCache can be used to significantly improve latency and throughput for many read-heavy application workloads or compute-intensive workloads.

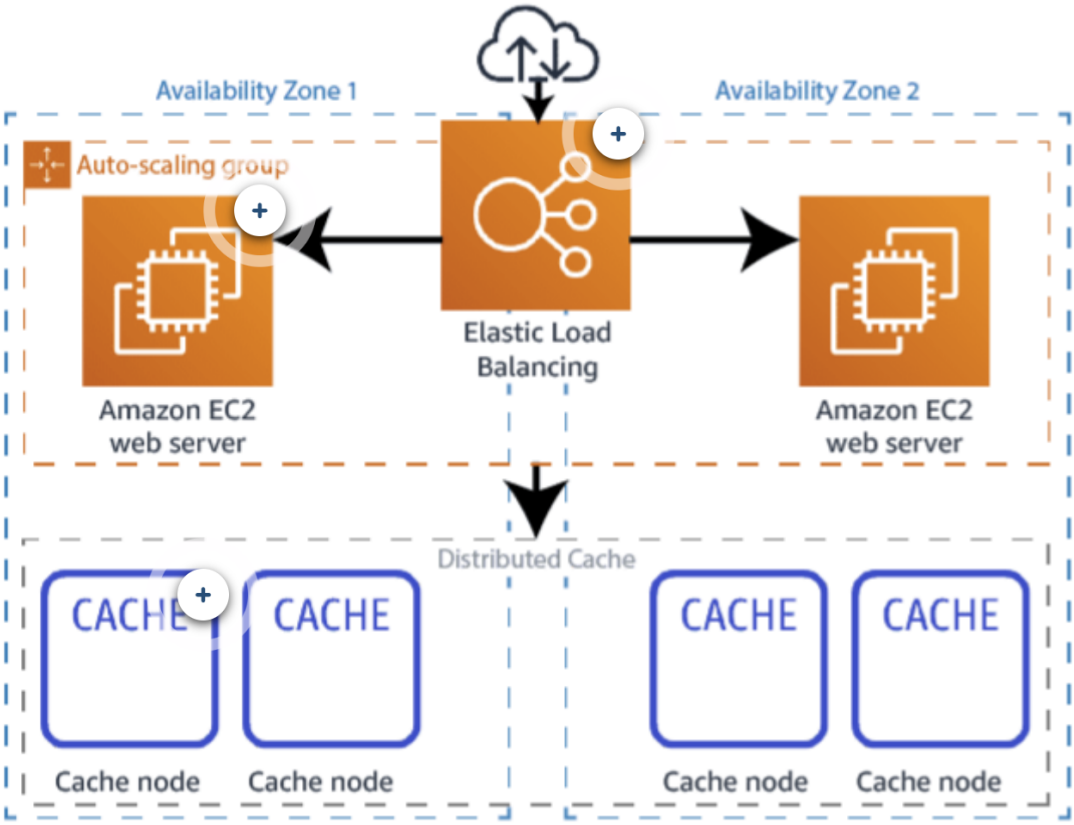
On-premises servers can use ElastiCache provided that there is connectivity between your VPC and data center through either a VPN or AWS Direct Connect.

Wordpress database cache architecture



Scalable distributed cache architecture

To address scalability and provide a shared data storage for sessions that can be accessible from any individual web server, you can abstract the HTTP sessions from the web servers themselves by storing them in a remote cache like ElastiCache. This architecture shows one way to accomplish this.



Two common approaches to caching are lazy loading and write-through. Lazy loading is reactive. Data is put into the cache the first time it is requested. Write-through is proactive. Data is put into the cache at the same time it is put into the database.

Limitations

- Elasticache EC2 nodes cannot be accessed from the internet, nor can they be accessed by EC2 instances in other VPCs. They can only be accessed from the same VPC where they are launched.
- You cannot configure an origin with ElastiCache.
- ElastiCache is a database caching service; it does not cache content from S3 buckets.

Pricing

ElastiCache supports two node types: On-Demand and Reserved. On-Demand Nodes let you pay for memory capacity by the hour with no long-term commitments. Pricing is per node-hour consumed, from the time you launch a node until you terminate it.

Pricing is per Node-hour consumed for each Node Type (node size). There is no charge for data transfer between Amazon EC2 and Amazon Elasticache within the same Availability Zone.

ElastiCache provides storage for one database snapshot at no charge. Each additional snapshot is charged per gigabyte per month. There is no ElastiCache data transfer charge for traffic in or out of the ElastiCache node itself.

Memcached

Simple, elastic, can run multiple CPU cores and threads, cache objects, not persistent, Cannot be used as a data store, primarily used for transient session data.

It does not support Multi-AZ failover or replication (You can launch multiple nodes across available AZs to minimize data loss on AZ failure). It also does not support snapshots.

With ElastiCache Memcached, each node represents a partition of data, and nodes in a cluster can span Availability Zones.

It is an in-memory database that can be used as a database caching layer. It does not support data replication or high availability.

Redis

Supports encryption, HIPAA compliant, allows clustering, allows replication, Pub/Sub scalability, geospatial indexing, backup and restore, multi-threading not allowed, allows auto-failover.

It does not support multiple CPU cores or threads.

Redis Auth Command: Redis authentication tokens enable Redis to require a token (password) before allowing clients to execute commands, thereby improving data security.

ElastiCache Redis has a good use case for autocompletion.

Memcached vs Redis

Feature	Memcached	Redis (cluster mode disabled)	Redis (cluster mode enabled)
Data persistence	No	Yes	Yes
Data types	Simple	Complex	Complex
Data partitioning	Yes	No	Yes
Encryption	No	Yes	Yes
High availability (replication)	No	Yes	Yes

Multi-AZ	Yes, place nodes in multiple AZs. No failover or replication	Yes, with auto-failover. Uses read replicas (0-5 per shard)	Yes, with auto-failover. Uses read replicas (0-5 per shard)
Scaling	Up (node type); out (add nodes)	Single shard (can add replicas)	Add shards
Multithreaded	Yes	No	No
Backup and restore	No (and no snapshots)	Yes, automatic and manual snapshots	Yes, automatic and manual snapshots

- Use memcache, if you can afford to lose cache and recover cache from database.
- Use Redis, for robust data persistence and for data that needs to be stored in complex data structure.
- Use Redis, for strict compliance requirements like PCI-DSS and HIPAA.

Amazon Neptune (Graph database)

Amazon Neptune is a fast, reliable, fully managed graph database service for applications that work with highly connected datasets. Neptune offers read replicas for high availability. You can create point-in-time copies, configure continuous backup to Amazon Simple Storage Service (Amazon S3) with replication across Availability Zones. Graph databases are purpose-built to store any type of data, whether it's structured, semistructured, or unstructured. Graph databases like Neptune fall under the category of NoSQL databases (non-relational).

Neptune supports two popular graph query languages: Apache TinkerPop and SPARQL.

Storing relations via RDBMS requires multiple tables with multiple foreign keys. SQL queries would be nested and joins will be complex. Neptune uses graph data structures such as nodes (data entities), edges (relationships), and properties to represent and store data. Relationships are stored as first-order citizens (an entity which supports all the operations generally available for other entities) of the data model.

Neptune also uses replicas. A Neptune replica connects to the same storage volume as the primary database instance and only supports read-only operations. There can be up to 15 of these replicas.

Lastly, Neptune uses a cluster volume. The cluster volume is where Neptune data is stored. It is designed for reliability and high availability. The cluster volume consists of copies of the data across multiple Availability Zones in a single Region.

How do you connect to a Neptune database?

So, by using an endpoint. An endpoint is a URL that contains a host address and a port. There are three different endpoints.

A cluster endpoint connects to the current primary database instance for the database cluster. There is only one cluster endpoint.

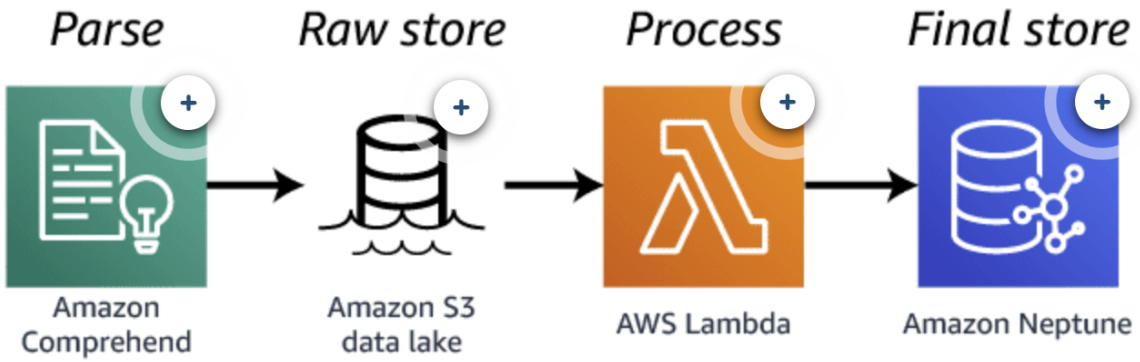
A reader endpoint connects to one of the available Neptune replicas. Each replica has its own endpoint.

An instance endpoint connects to a specific database instance. Each database instance in a database cluster has its own unique instance endpoint. The instance endpoint provides direct control over connections to the DB cluster, for scenarios where using the cluster endpoint or reader endpoint might not be appropriate.

Security

To control who can manage your Neptune database, you use AWS Identity and Access Management (IAM). To control the connections clients make when communicating with the database, you use HTTPS encrypted client connections. Your data at rest in the database is encrypted using the industry standard AES-256 bit encryption algorithm on the server that hosts your Neptune instance. Keys can also be used, which are managed through AWS Key Management Service (AWS KMS).

RSS Keyword capture architecture



Use Cases

- Recommendation Engines
- Fraud Detection
- Knowledge Graphs
- Network Security

Pricing

First, you pay for the instance hosting the databases, known as an On-Demand Instance. You pay for your database by the hour with no long-term commitments or upfront fees. Second, you pay for the storage consumed by your database. This is billed per gigabyte per month, and the first 50 gigabytes of backup storage is offered at no cost. Third, you pay for the number of requests to the database. Neptune was designed to eliminate unnecessary I/O operations in order to reduce costs and to ensure that resources are available for serving read/write traffic. Fourth, you pay for the amount of data transferred out of the database. You never pay for moving data into your database.

Amazon Redshift

Amazon Redshift is an enterprise-level, petabyte scale, fully managed data warehousing service. With Amazon Redshift, you can achieve efficient storage and optimum query performance through a combination of massively parallel processing, columnar data storage (which takes less storage space and allows for much better performance), and very efficient, targeted data compression encoding schemes.

RedShift can also improve performance for repeat queries by caching the result and returning the cached result when queries are re-run. Dashboard, visualization, and business intelligence (BI) tools that execute repeat queries see a significant boost in performance due to result caching.

Here are the three key features of Amazon Redshift :

- Analyze data from terabytes to petabytes and run complex analytical queries.
- Fast, fully managed, petabyte-scale data warehouse service.
- It offers both provisioned and serverless options.

Here's a brief explanation of each feature:

1. Massive Data Handling and Complex Analytics:
    - Redshift can efficiently store and process massive amounts of data, ranging from terabytes to petabytes.
    - It's specifically optimized for complex analytical queries, enabling you to uncover insights from large datasets quickly.
2. Fast, Fully Managed, Petabyte-Scale:
    - Redshift is designed for high performance and scalability, delivering fast query response times even for large datasets.
    - It's a fully managed service, meaning AWS handles infrastructure management and maintenance, reducing your operational burden.
3. Deployment Flexibility: Provisioned and Serverless:

    - Redshift offers two deployment options to suit different workload patterns and cost requirements:
        - Provisioned Redshift: Provides a dedicated cluster with fixed resources, ideal for predictable workloads.
        - Serverless Redshift: Automatically scales compute resources based on workload demands, suitable for unpredictable or bursty workloads.

While these are key features, it's worth noting two additional important characteristics of Redshift:

- Multi-Region Deployments: Redshift supports both multi-AZ and multi-region deployments for enhanced availability, disaster recovery, and data locality.
- On-Premises Option: Redshift on Outposts enables you to run Redshift clusters directly in your on-premises environment for control and security.

Amazon Redshift Spectrum

Using Amazon Redshift Spectrum, you can efficiently query and retrieve structured and semistructured data from files in Amazon S3 (in place) without having to load the data into Amazon Redshift tables. Redshift Spectrum queries employ massive parallelism to execute very fast against large datasets.

To use Redshift Spectrum, you need to create an external table in Redshift. The external table will point to the data in S3. You can then query the data in S3 using SQL statements.

Because Amazon Athena and Amazon Redshift share a common data catalog and data formats, you can use both Athena and Redshift Spectrum against the same data assets.

You would typically use Athena for ad hoc data discovery and SQL querying, and then use Redshift Spectrum for more complex queries and scenarios where a large number of data lake users want to run concurrent BI and reporting workloads.

Scalability

RedShift uses EC2 instances, so you need to choose your instance type/size for scaling compute vertically, but you can also scale horizontally by adding more nodes to the cluster.

Availability

- A single node RedShift cluster does not support data replication, and you’ll have to restore from a snapshot on S3 if a drive fails. RedShift can asynchronously replicate your snapshots to S3 in another region for DR.

- Amazon Redshift clusters are comprised of nodes. Compute nodes divide work among slices. Each slice is assigned a portion of the node's memory and drive space. When you connect to an Amazon Redshift cluster, you use the SQL endpoint.

Security

- You cannot directly access your AWS RedShift cluster nodes as a user, but you can do so through applications.
- RedShift supports SSL Encryption in-transit between client applications and Redshift data warehouse cluster.
- RedShift has VPC support for network isolation.
- RedShift allows encryption for data at rest (AES 256).
- By default, traffic between an Amazon Redshift cluster and Amazon S3 traverses the public AWS network. A VPC endpoint forces all COPY and UNLOAD traffic between your cluster and your data on Amazon S3 to stay in your VPC.
- Amazon Redshift needs permission to copy data to and from S3 buckets. This access is granted using AWS Identity and Access Management (IAM) roles.

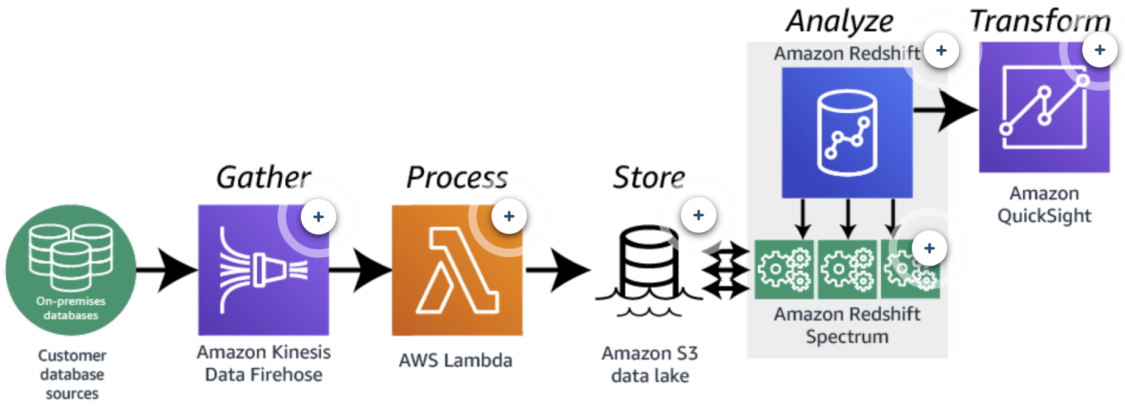
Monitoring

- RedShift allows audit logging and AWS CloudTrail integration.

Limitations

- RedShift can store huge amounts of data but cannot ingest huge amounts of data in real-time.

Event-driven data analysis architecture



Pricing

- Charged for compute nodes hours, 1 unit per hour (only compute node, not leader node).
- Data transfer — There is no charge for data transfer between RedShift and S3 within a region for backup, restore, load, and unload operations, but you may pay charges for other scenarios.
- On-Demand pricing has no upfront costs. You simply pay an hourly rate based on the type and number of nodes in your cluster.
- With Concurrency Scaling pricing, you simply pay a per-second on-demand rate for usage that exceeds the free daily credits. Each cluster earns up to one hour of free Concurrency Scaling credits per day, which is sufficient for most customers.
- Reserved Instance pricing enables you to save up to 75 percent over On-Demand rates by committing to using Amazon Redshift for a 1- or 3-year term.

Amazon MemoryDB for Redis

- Redis compatible fully managed in-memory database service that delivers ultra-fast performance.
- Microsecond read latency, single-digit millisecond write latency, high throughput, and Multi-AZ durability for modern applications.
- Both DynamoDB and In MemoryDB provides the same benefits but In MemoryDB is way faster than DynamoDB and provides durability, high availability (HA), scalability and persistence. It can also be used for storing session information.

Amazon Keyspaces (for Apache Cassandra)

- Managed Apache Cassandra compatible database service.
- A good option for high-volume applications with straightforward access patterns.

Amazon Timestream

- Time Series database service for Internet of Things and operational applications.
- It makes it easy for us to store and analyze millions of events per day (1000 times faster and for as little as one tenth of the cost of relational databases).
- Time series data is a sequence of data points recorded over a time interval (stock price, temperature measurements).

Amazon QLDB (Quantum Ledger database)

- Immutable database, no entry can be removed. it means that if the documents are updated or deleted, the record of the change is itself unchangeable and cannot be altered.
- It provides a cryptographically verifiable history of all changes made to your application data.

- Normally, applications use audit tables and audit trails to help track data lineage, which can be difficult to scale. QLDB solves this problem.
- Because Amazon QLDB is serverless, you don’t have to worry about provisioning capacity or configuring read and write limits. You create a ledger, define your tables, and Amazon QLDB automatically scales to support the demands of your application.
- Amazon QLDB supports PartiQL, a new open standard query language. With PartiQL, you can easily query, manage, and access the entire data and change history using familiar SQL operators.
- Amazon QLDB supports transactions with ACID semantics, a flexible document data model, and a simple query language.

Availability

Amazon QLDB's ledger is deployed across multiple Availability Zones with multiple copies per Availability Zone. AWS maintains redundancy within the Region and ensures full recoverability from Availability Zone failures. A write is acknowledged only after being written to multiple Availability Zones, and hence, Amazon QLDB is strongly durable.

Components

- Ledger - A ledger consists of a set of tables and a journal that maintains the complete, immutable history of changes to the tables. A journal is the chained set of all blocks committed in your ledger. A block is an object that is committed to the journal in a transaction. The journal represents a complete and immutable history of all the changes to your ledger data.
- Tables - Tables exist within a ledger and contain a collection of document revisions. This can include a record of the deletion of a document.
- Documents - Documents exist within tables and must be in Amazon Ion form. Ion is a superset of JSON that adds additional data types, type annotations, and comments. Amazon QLDB supports documents that contain nested JSON elements and gives you the ability to write queries that reference and include these elements.

Views

- **User view:** Shows the latest non-deleted revision of your application-defined data only. This is the default view in QLDB.
- **Committed view:** Shows the latest non-deleted revision of both your application-defined data and the system-generated metadata. This is the full system-defined table that corresponds directly to your user table.
- **History view:** Shows the revisions from the system-defined view of your table. This view includes both your data and the associated metadata in the same schema as the committed view.

Security

Use AWS Identity and Access Management (AWS IAM). An IAM administrator uses policies to specify who has access to AWS resources and what actions those users can perform on those resources. With IAM identity-based policies, you can specify allowed or denied actions and resources in addition to the conditions under which actions are allowed or denied.

Monitoring

AWS provides various tools that you can use to monitor Amazon QLDB. These include automated tools such as Amazon CloudWatch Alarms, Logs, and Events. You can use these monitoring tools to automatically watch Amazon QLDB and report when something is wrong. AWS always recommends that you automate monitoring tasks as much as possible.

Pricing

Amazon QLDB has a pay-as-you-go model. You are billed for read/write input and output (IO) requests, data transfer, journal storage, and indexed storage.
Storage consumed by your Amazon QLDB ledger is billed per GB per month, and IOs consumed are billed per million requests.

Use cases

- Banking transactions

Consider the following points when considering a DB on EC2

- You can run any database you like with full control and ultimate flexibility.
- You must manage everything including backups, redundancy, patching, and scaling.
- It is a good option if you require a database not yet supported by RDS.
- High availability approaches for databases.
- If possible, choose DynamoDB over RDS because of inherent fault tolerance.
- If DynamoDB can’t be used, choose Aurora because of redundancy and automatic recovery features.
- If Aurora can’t be used, choose Multi-AZ RDS.
- If the database runs on EC2, you have to design the HA yourself.

Amazon Aurora

An Amazon Aurora DB cluster consists of a DB instance compatible with either MySQL or PostgreSQL and a cluster volume that represents the data for the DB cluster copied across three Availability Zones as a single, virtual volume. The DB cluster contains a primary instance and optionally, up to 15 Aurora Replicas. A DB cluster does not necessarily scale read operations as it is an option to deploy Aurora Replicas.

Aurora supports burstable performance feature and memory-optimized instance class.

Aurora Replicas

Aurora Replicas are independent endpoints in an Aurora DB cluster, best used for scaling read operations and increasing availability. Up to 15 Aurora Replicas can be distributed across the Availability Zones that a DB cluster spans within an AWS Region.

An Aurora Replica is both a standby in a Multi-AZ configuration and a target for read traffic. The architect simply needs to direct traffic to the Aurora Replica, to reduce latency for write requests.

All Aurora Replicas return the same data for query results with minimal replica lag — usually, much less than 100 milliseconds after the primary instance has written an update. Replica lag varies depending on the rate of database change. That is, during periods where a large amount of write operations occur for the database, you might see an increase in replica lag.

Global Database

For globally distributed applications, you can use Global Database, where a single Aurora database can span multiple AWS Regions to enable fast local reads and quick disaster recovery. Aurora replicates data to the secondary AWS regions with a typical latency of under a second.

Aurora Global Database uses dedicated infrastructure that leaves your database fully available to serve application workloads. In the unlikely event of a regional degradation or outage, one of the secondary regions can be promoted to full read/write capabilities in less than 1 minute.

It is the best solution if you are looking for an automated way to enable inter-region disaster recovery capabilities with fast replication and fast failover

Multi-Master

Amazon Aurora Multi-Master is a new feature of the Aurora MySQL-compatible edition. It adds the ability to scale out write performance across multiple Availability Zones, allowing applications to direct read/write workloads to multiple instances in a database cluster and operate with higher availability.

With single-master Aurora, a failure of the single writer node requires the promotion of a Read Replica to be the new writer. In the case of Aurora Multi-Master, the failure of a writer node merely requires the application to use the writer to open connections to another writer.

Aurora Serverless

Amazon Aurora Serverless is an on-demand, auto-scaling configuration for the MySQL-compatible and PostgreSQL-compatible editions of Amazon Aurora. The database automatically starts up, shuts down, and scales capacity up or down based on application needs.

It enables you to run a database in the cloud without managing any database instances. It is a simple and cost-effective option for infrequent, intermittent, or unpredictable workloads. You simply create a database endpoint and optionally specify the desired database capacity range and connect applications.

You can migrate between standard and serverless configurations with a few clicks in the Amazon RDS Management Console.

With Aurora Serverless, you only pay for database storage and the database capacity and the I/O that your database consumes while it is active.

Scalability

Aurora cluster volumes automatically grow as the amount of data in your database increases. An Aurora cluster volume can grow to a maximum size of 64 tebibytes (TiB).

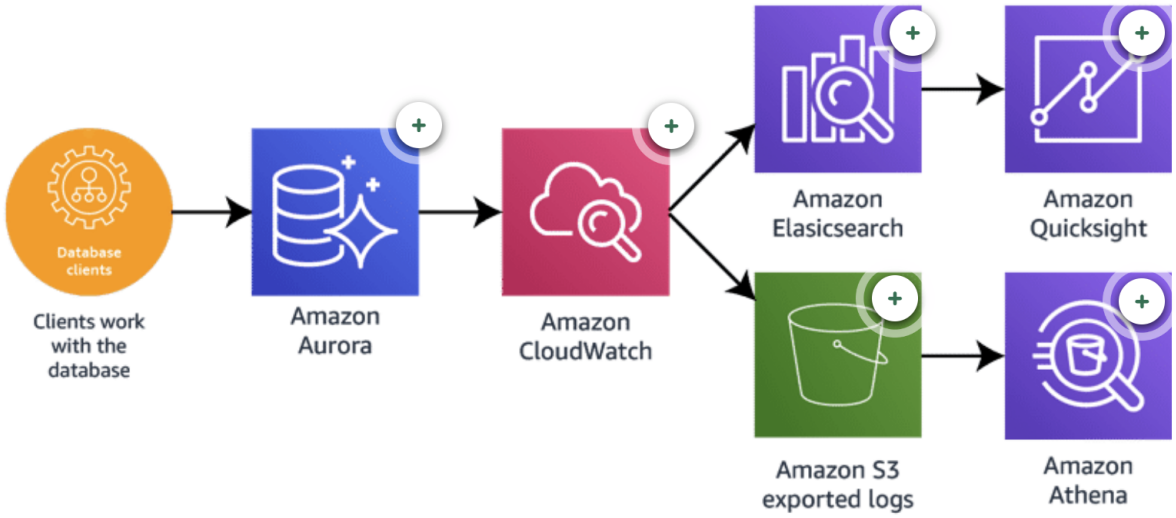
Backup

Amazon Aurora's backup capability enables point-in-time recovery for your instance. This allows you to restore your database to any second during your retention period, up to the last five minutes. Only automated backups can be used for point-in-time DB instance recovery. The granularity of point-in-time recovery is 5 minutes.

- The retention period is the period in which AWS keeps the automated backups before deleting them.
- You can disable automated backups by setting the retention period to zero (0).
- An outage occurs if you change the backup retention period from zero to a non-zero value or the other way around.
- There is no additional charge for backups, but you will pay for storage costs on S3.

Public source data ingestion architecture

Logs Analytics architecture



Pricing

On-Demand, Reserved, and Serverless are pricing options for Aurora.

Amazon DynamoDB global tables vs Amazon Aurora Global Database

Amazon DynamoDB global tables provide a fully managed solution for deploying a multi-region, multi-master database. This is the only solution from the options given above that provides an active-active configuration where reads and writes can take place in multiple regions with full bi-directional synchronization. While Amazon Aurora Global Database provides read access to a database in multiple regions, it does not provide an active-active configuration with bi-directional synchronization (although you can failover to your read-only DBs and promote them to writable).

AWS Database Migration Service

It helps you migrate databases to AWS quickly and securely. The source database remains fully operational during the migration, minimizing downtime to applications that rely on the database. You can use this along with the Schema Conversion Tool (SCT) to migrate databases to AWS RDS or EC2-based databases.

At its most basic level, AWS DMS is an instance in the AWS Cloud that runs replication software.

You can also use AWS DMS to keep your source and target synced by migrating ongoing transactions as they occur on the source.

AWS DMS is highly resilient and self-healing. It continually monitors source and target databases, network connectivity, and the replication instance.

Homogeneous migrations, where you migrate between same database engines, may require the use of native database tools to migrate database elements.
Heterogenous migrations, where you migrate between different database engines, require the use of the AWS Schema Conversion Tool (AWS SCT) to first translate your database schema to the new platform. You can then use AWS DMS to migrate the data. It is important to understand that AWS DMS and SCT are two different tools that serve different needs.

Schema Conversion Tool

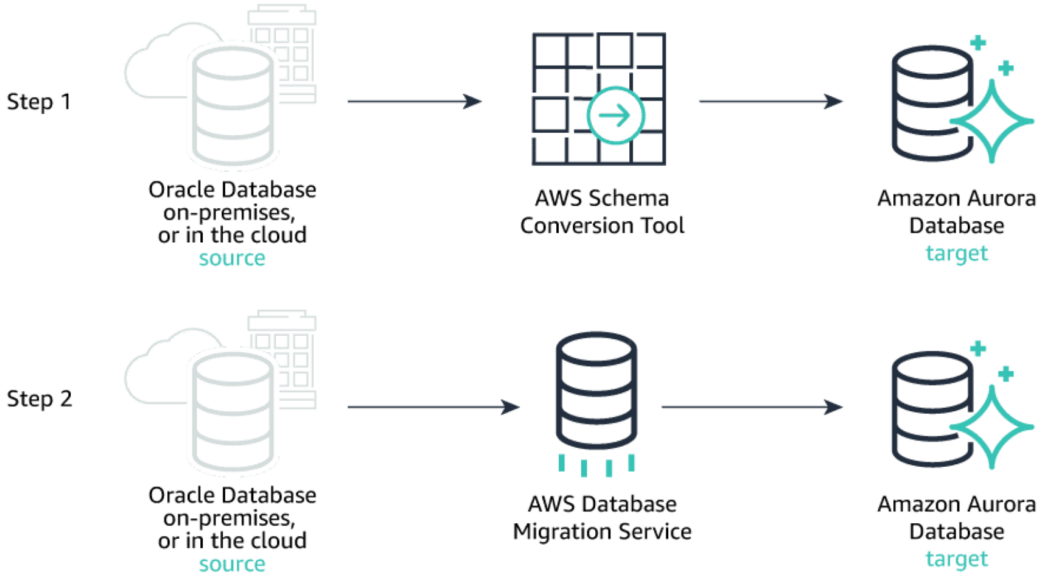
Schema Conversion Tool can copy database schemas for homogenous migrations (same database) and convert schemas for heterogeneous migrations (different databases). SCT is used for larger, more complex datasets like data warehouses.

If you want to switch database engines, AWS SCT can convert your existing database schema to the target platform. This includes tables, indexes, views, and stored procedures, plus your application code. If the schema from your source database can't be converted automatically, AWS SCT provides guidance on how you can create equivalent schema in your target database engine.

Once you use a database migration service to migrate a database to RDS, then you do not need to use SCT if migrating to the same destination database engine.

AWS SCT helps you modernize your SQL code to work with your new database. You can use AWS SCT to extract SQL statements that are embedded in your application code. AWS SCT will track all the places where SQL is present, convert the SQL to work with the target database, and rebuild your application program with the converted code.

The Schema Conversion Tool (SCT) agent extracts your data and uploads the data to either Amazon S3 or, for large-scale migrations, an AWS Snowball Edge device.



Pricing

AWS DMS is a low-cost service. You only pay for the compute resources used during the migration process, any additional log storage, and data transfer if not in the same AWS Region.

Limitations

You cannot use AWS DMS to migrate from an on-premises database to another on-premises database.

Reliability and Fault Tolerance

Auto-Scaling

A feature to automatically scale your capacity based on metrics received from Cloudwatch. Amazon EC2 auto scaling service is used to scale EC2 instances. It is a region-specific service, however, it can span multiple AZs within the same AWS region. It will try to distribute EC2 instances evenly across AZs. There is no additional cost for Auto Scaling; you just pay for the resources (EC2 instances) provisioned. One or more ELBs can be attached to existing ASG; however ELBs must be in the same region. Auto Scaling scales horizontally only.

We can set a limit, how much maximum cost we can pay. We can also set a minimum number of instances that should always run.

Scalable Resources

- EC2 Auto Scaling Groups
- Aurora DB Clusters
- DynamoDB Global Secondary Indexes (DynamoDB tables that replicate a subset of the data from the base table, indexed by a different partition key and sort key)
- DynamoDB tables
- Elastic Container Service Services
- Spot Fleet Requests

Components

Launch Configuration

Which resources should be automatically scaled. Configuration of resources are also defined here. Here which subnets Auto Scaling will launch new instances into is also specified.

- You cannot disable the launch configuration, and you cannot modify a launch configuration after you've created it.
- If you need to update AMI for launching any new instances: Create a new launch configuration that uses the AMI, and update the ASG to use the new launch configuration.

Auto Scaling Groups (*ASGs*)

A collection of instances with similar characteristics can be placed in Auto Scaling Group. To configure Auto Scaling Group, you must know the relevant metrics for your application (CPU usage, memory requirements, etc.). ASGs can span multiple AZs.

Launch Methods

- Launch Configuration
- Launch Templates
- Using an existing EC2 instance that is in production

When to use launch templates:

- You need more flexibility and control over your instance launch parameters.
- You need to use advanced features or integrate with other AWS services.
- You need to manage multiple versions of your instance configuration.

When to use launch configurations:

- You have a simple use case and do not need the extra features provided by launch templates.
- You prefer a more straightforward, less complex approach to managing your instances.
- You are using an older AWS service that does not support launch templates.

Termination Policy

We can tell ASG to terminate instances (scale in) based on:

- Oldest Instance

- Newest Instance
- Oldest Launch Configuration
- Closest to Next Instance Hour (that's about to be billed again)
- Default (checks for Oldest Instance, if not then Closest to Next Instance Hour, if not then terminates randomly)
    - It first terminates the instance in AZ with most instances.

Amazon EC2 Auto Scaling groups

Where should the resources be deployed (VPC, Subnets)?. Here you also specify the type of purchase (On-Demand instances, Spot Instances).

- Automatically replaces an instance if the instance health check fails.
- Attempt to respond to availability zone outages by temporarily adding additional instances to the remaining healthy zone.
- Automatically inform CodeDeploy about new instances so that we won't have to trigger a deployment manually anymore when a new EC2 instance comes online.
- A single auto scaling Group can not be used for launch instances in multiple regions.
- You can attach one or more Target Groups to your ASG to include instances behind an ALB. The ELBs must be in the same region. Once you do this, any EC2 instance existing or added by the ASG will be automatically registered with the ASG defined ELBs. If adding an instance to an ASG results in exceeding the maximum capacity of the ASG, the request will fail.
- Auto Scaling groups can be edited once created (However, launch configurations cannot be edited).

Multi-AZ

If EC2 instances are in single AZ and you want to deploy in multiple AZs then you need to create a new Auto Scaling group with multi-az configuration. Attaching to the existing auto scaling group (without multi-az config) will not serve the purpose.

You can merge multiple single AZ Auto Scaling Groups into a single Multi-AZ ASG. The process involves rezoning one of the groups (By choosing one of the ASGs to be the Multi-AZ ASG) to cover/span the other AZs for the other ASGs and then deleting the other ASGs. Merging can be performed on ASGs with or without ELBs attached to them. The resulting ASG must be one of the pre-existing ASGs.

Scaling Policies

When should the resources be added or removed. There are three types of scaling policies:

- **Simple** - We create a cloudwatch alarm and specify what to do when it is invoked, whenever this policy is invoked it enters a cooldown period before taking any action.
- **Step** - Responds to additional alarms even when a scaling activity or health check replacement is in progress.
- **Target tracking** - When you are not sure about when to add and remove instances then this policy is the right option, it only needs information what to track (CPU utilization,

average network utilization, requests count) and it automatically creates the required CloudWatch alarms.
- **Scheduled Scaling Policy** - Scaling based on a schedule allows you to set your own scaling schedule for predictable load changes. It is ideal for situations where you know when and for how long you are going to need the additional capacity.

Target tracking in place of step scaling is RDS recommended for most use cases.

Scaling Options

It defines the triggers and when instances should be provisioned/de-provisioned.
- Maintain: Keep a specific or minimum number of instances running.
- Manual: Use maximum, minimum, or a specific number of instances.
- Scheduled: Increase or decrease the number of instances based on a schedule.
- Dynamic: Scale based on real-time system metrics (e.g., CloudWatch metrics).

Rebalancing

Auto Scaling can perform rebalancing when it finds that the number of instances across AZs is not balanced. Auto Scaling rebalances by launching new EC2 instances in the AZs that have fewer instances first; only then does it start terminating instances in AZs that had more instances. Auto Scaling may go over the maximum number of instances by 10% temporarily for rebalancing.

Health Checks

- By default, ASGs use EC2 status checks.
- It can also use ELB health checks and custom health checks.
- ELB health checks are in addition to the EC2 status checks.
- With ELB, an instance is marked as unhealthy if ELB reports it as OutOfService.
- If any health check returns an unhealthy status, the instance will be terminated. Unhealthy instances are auto-replaced. Any status other than “running” is unhealthy.
- For the “impaired” status, the ASG will wait a few minutes to see if the instance recovers before taking action. If the “impaired” status persists, termination occurs. Unlike AZ rebalancing, termination of unhealthy instances happens first. Then, Auto Scaling attempts to launch new instances to replace terminated instances.

Terminated Instances

Once in a terminating state, an EC2 instance cannot be put back into service again. However, there is a short time period in which a CLI command can be run to change an instance to healthy.

Troubleshooting

You can suspend and then resume one or more of the scaling processes for your Auto Scaling group. Suspending scaling processes can be useful when investigating a configuration problem

or other issues with your web application and then making changes to your application without invoking the scaling processes.

Standby Instances

You can manually move an instance from an ASG and put it in the standby state. Auto Scaling still manages instances in the standby state. These instances are charged as normal, and do not count towards available EC2 instances for workload/application use.

Auto Scaling does not perform health checks on instances in the standby state. Standby state can be used for performing updates/changes/troubleshooting etc., without health checks being performed or replacement instances being launched.

Load Balancing

Incoming traffic is distributed using an Elastic Load Balancing service. ELB service is also used for checking the health of resources (e-g EC2 instances). This also avoids propagation time issues (the time frame it takes for DNS changes to be updated across the internet) as it is situated between client and the server.

ELB may look like a single point of failure, but by design it is highly available and automatically scalable (similar to s3 and DynamoDB).

If requests need to be sent to the same backend server (same client to the same target) for stateful applications then sticky sessions are used which uses an HTTP Cookie to remember which server to send the traffic to across connections.

ELBs are not placed in front of RDS instances; they are placed in front of EC2 instances.

ELB can be used with EC2, ECS, and Auto Scaling services.

Categories

Sender (Clients) Initiated - Sender locates the best target. Load Balancer will return a list of resources and the sender selects a target from them.

Receiver (Load Balancer) Initiated - Receiver selects the target.

Static Loading Balancing - Targets are selected based on action, actions are always processed on assigned target, no scalability. Multi-tier applications are often implemented with static load balancing. Certain components of the application are forced to specific nodes.

Dynamic Load Balancing - Actions dynamically assigned, scalability is provided. This is used by Elastic Load Balancing in AWS

Algorithms

Round Robin - first request to first resource, second to second and so forth.
Randomized - Uses random algorithm to decide
Centrally Managed - Sophisticated algorithms, depends on use case
Threshold Based - e-g send all requests to a particular server until a threshold is reached, after that threshold send request to some other server.

Health Checks

ELB supports two types of health checks.

- Establishing a connection to a backend EC2 instance using TCP and making the instance as available if connection is successful.
- Making an HTTP or HTTPs request to a webpage that you specify and validating that an HTTP response code is returned.
    - Ideally that webpage should request all other application resources (s3, rds etc), to make sure that ELB success health status means all components of applications are working.
- In case ELB receives unhealthy status from backend systems then it informs EC2 Auto Scaling to remove the instance from the group and replace it with a new instance.

Connection Draining

Whenever EC2 is in the process of being deregistered, ELB uses its Connection Draining feature to make sure existing connections close cleanly.

Connection draining is enabled by default and provides a period of time for existing connections to close cleanly. When connection draining is in action, a CLB will be in the status “InService: Instance deregistration currently in progress”.

If Amazon EC2 Auto Scaling has a scaling policy that calls for a scale down action, it informs ELB that the EC2 instance will be terminated. ELB can prevent Amazon EC2 Auto Scaling from terminating an EC2 instance until all connections to that instance end. It can also prevent any new connections.

If using an ELB, it is best to enable ELB health checks because otherwise, EC2 status checks may show an instance as healthy that the ELB has determined as unhealthy. In this case, the instance will be removed from service by the ELB but will not be terminated by Auto Scaling.

ELB nodes have public IPs and route traffic to the private IP addresses of the EC2 instances. You need one public subnet in each AZ where the ELB is defined and the private subnets are located.

Public load balancer

It is associated with a public IP address and has a regional scope. It requires two subnets, each in a separate availability domain.

Private load balancer

It has a private IP address, with both the primary and standby load balancers located in the same subnet. It is bound within an availability domain. To ensure availability across multiple domains, customers can configure multiple private load balancers and DNS servers to set up a round robin DNS configuration with the IP addresses of the private LBs.

Types

Application Load Balancer

Balances HTTP/HTTPS traffic (Layer 7). It routes connections based on the content of the request.

Authentication

It can also authenticate users before they can pass through the load balancer. It uses OpenID Connect (OIDC) protocol (Google, Facebook, and Amazon) and integrates with AWS services to support protocols, such as (SAML, LDAP, Microsoft Active Directory, etc). It is implemented through an authentication action on a listener rule that integrates with Amazon Cognito to create user pools.

Routing

It performs content-based routing that allows the routing of requests to a service based on the content of the request. Following are types of content-based routing:

- **Host-based routing:** Route client requests based on the HTTP header's Host field, allowing you to route to multiple domains from the same load balancer.
- **Path-based routing:** Route a client request based on the HTTP header's URL path (e.g., /images or /orders).
- Combined

Limitations

- An ALB does not support static IP addresses and is not suitable for a proxy function.
- HTTP/HTTPS protocol is supported by the ALB. SSL, TCP, ICMP is not supported by ALB.

Pricing

Application Load Balancers are billed at an hourly rate and an additional rate based on the load placed on your load balancer.

Network Load Balancer (NLB)

Balances TCP/UDP/TLS (Layer 4). It routes connections based on IP protocol data.

It supports static and elastic IP Address. It can handle millions of requests/second, sudden volatile traffic patterns, and provides extremely low latencies. It has high throughput - it is designed to handle traffic as it grows and can load millions of requests/second.

Network Load Balancers also support routing requests on multiple applications on a single Amazon EC2 instance and supports the use of containerized applications.

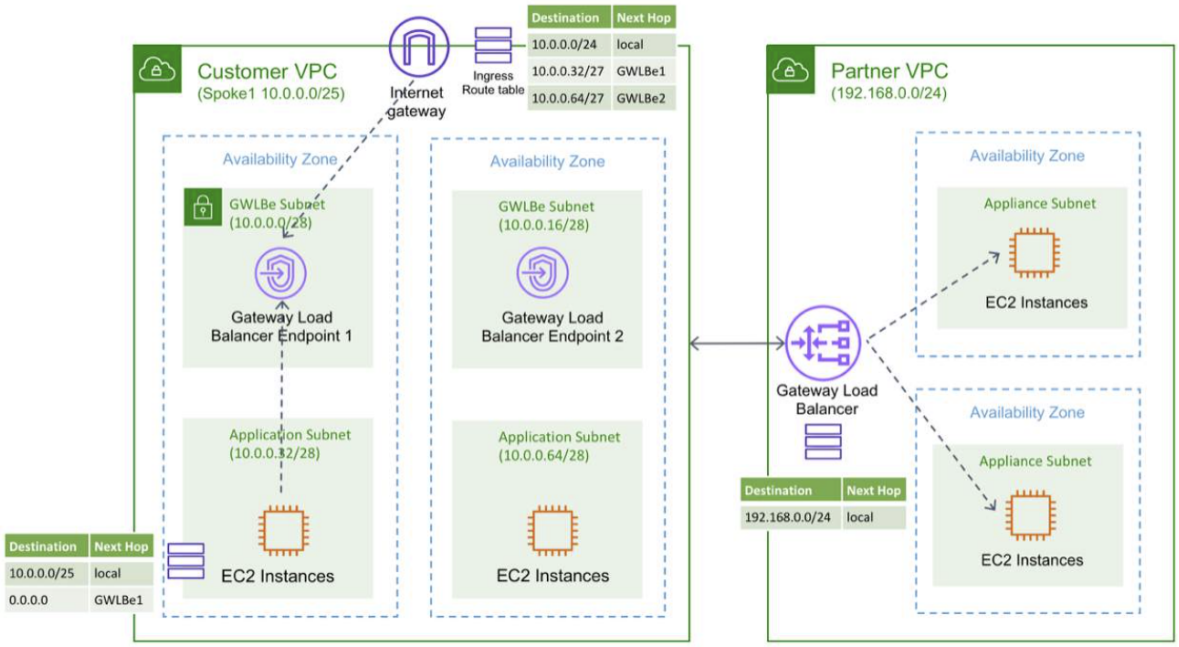
NLB can distribute traffic between regions if VPC are peered.

Gateway Load Balancer (GWLB)

Gateway Load Balancers let you deploy, scale, and manage virtual appliances, such as firewalls, intrusion detection and prevention systems, and deep packet inspection systems. It combines a transparent network gateway (that is, a single entry and exit point for all traffic) and distributes traffic while scaling your virtual appliances with the demand.

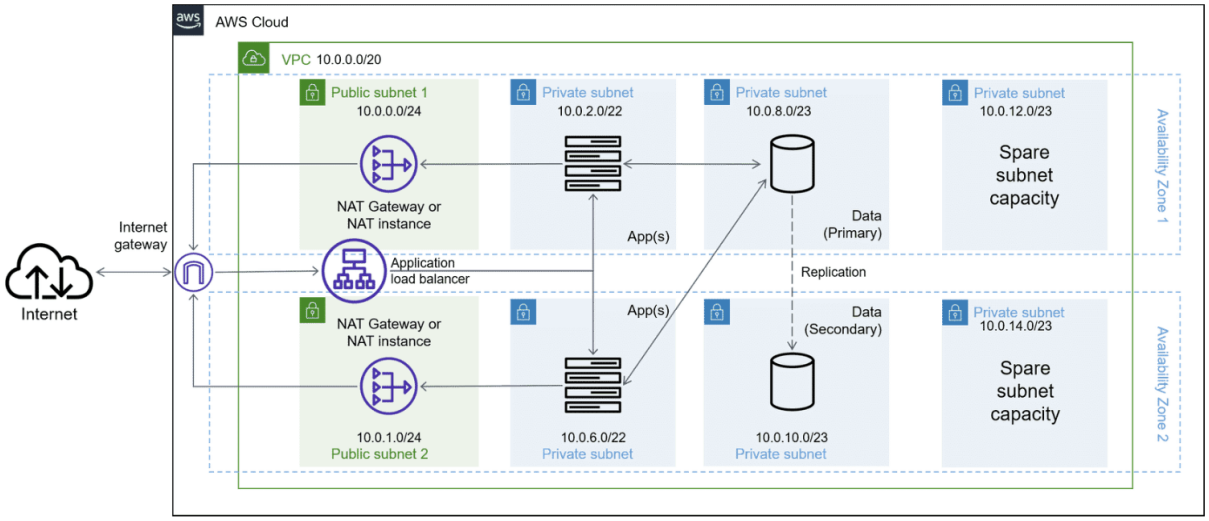
A Gateway Load Balancer operates at the third layer of the Open Systems Interconnection (OSI) model, the network layer.

Gateway Load Balancers use Gateway Load Balancer endpoints to securely exchange traffic across VPC boundaries. A Gateway Load Balancer endpoint is a VPC endpoint that provides private connectivity between virtual appliances in the service provider VPC and application servers in the service consumer VPC. You deploy the Gateway Load Balancer in the same VPC as the virtual appliances. You register the virtual appliances with a target group for the Gateway Load Balancer.



Feature	GWLB	NLB	ALB
Layer of operation	Network (Layer 3)	Transport (Layer 4)	Application (Layer 7)
Routing Capabilities	VPC endpoints, security appliances, NAT Gateways, other load balancers	TCP / UDP target	HTTP/HTTPS targets
Advanced routing features	Sticky Sessions	Sticky Sessions, path-based routing, header-based routing	Sticky sessions, path-based routing, header-based routing
Scalability	Highly Scalable	Scalable	Scalable
Use cases	Centralized inspection, east-west and north-south traffic management	High-performance, low-latency applications	Applications that require advanced routing features

Multi-tier Architecture



Components

Listener

Listens on a configured port (80/443) for traffic (depends on type of load balancer). There can be many listeners for a single load balancer.

Target Group

It is a type of backend resource to which requests will be forwarded (e-g EC2 instances, ECS containers, lambda functions, IP addresses). Health checks must be defined for each target group. Public IP addresses can not be used as targets. You cannot mix different types within a target group (EC2, ECS, IP). It can only be associated with one load balancer. Following attributes are defined for a target group:

- **Deregistration Delay:** It is the amount of time for ELB to wait before deregistering a target.
- **Slow Start Duration:** It is the time period, in seconds, during which the load balancer sends a newly registered target a linearly increasing share of the traffic to the target group.
- **Stickiness:** Indicates whether sticky sessions are enabled or not.

Rule

It is associated with listeners to perform action based on attributes of incoming traffic, for example, if the request url ends with /info then distribute the traffic to target group 2.

- After the load balancer receives a request, it evaluates the listener rules in priority order to determine which rule to apply and then selects a target from the target group for the rule action using the round-robin routing algorithm.
- Routing is performed independently for each target group, even when a target is registered with multiple target groups.

Security

Perfect Forward Secrecy: Provides additional safeguards against eavesdropping of encrypted data through the use of a unique random session key.

Server Order Preference: It lets you configure the load balancer to enforce cipher ordering, providing more control over the level of security used by clients to connect with your load balancer.

ELB does not support client certificate authentication (API Gateway does support this).

Distributed Denial of Service (DDoS) Protection

- ELB distributes traffic in a way that minimizes the risk of overloading a single resource.
- ELB only supports valid TCP requests,so DDoS attacks such as UDP and SYN floods are not able to reach EC2 instances.

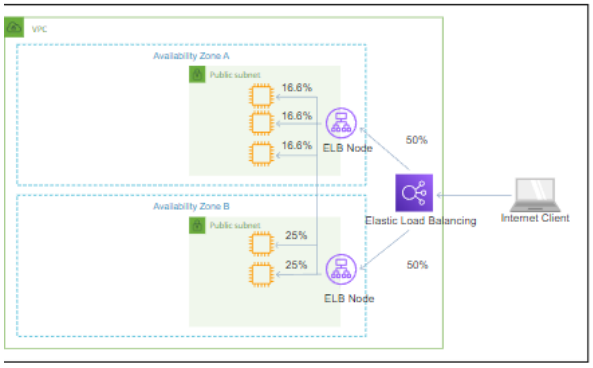
Monitoring

Access Logs on ELB are disabled by default. Information includes information about the clients (not included in CloudWatch metrics), such as the identity of the requester, IP, request type, etc. Logs are stored in S3 by default. CloudWatch can be configured to collect and parse the ELB logs from S3.

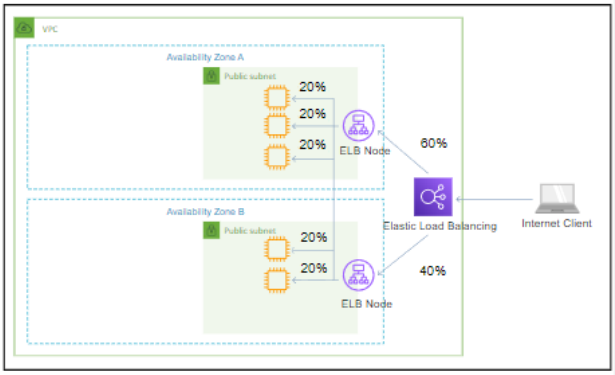
CloudWatch, CloudTrail, and Access Logs (requester IP address, request type etc) are available for ELB.

Cross-zone load balancing

It is enabled by default for ALB, and can be enabled/disabled for other load balancer types. When enabled each load balancer node distributes traffic across the registered targets in all enabled availability zones. If disabled, each load balancer node distributes traffic across the registered targets in its availability zone only.



Cross-Zone Load Balancing - Disabled



Cross-Zone Load Balancing - Enabled

Availability

Active-passive systems

Only one of the two instances is available at a time. This is useful for stateful applications.

Active-active systems

Active-passive systems disadvantage is scalability. In active-active systems with both servers available, the second server can take the same load for the application, and the entire system can take more load. This is good for stateless applications.

Tips for making architecture more resilient and automating infrastructure provisioning

- Automatic failover in case of failure. Failover is a process when the primary component fails then secondary component takes the responsibility. Architecture can failover gracefully using elastic ip.

- Utilize multiple availability zones (AZ) e-g automatic replication of RDS across multiple AZ's.
- Maintain a machine image that can be used to clone environments in a different AZ.
- Use cloudwatch for visibility and taking action when needed.
- Use Auto Scaling group to ensure that the required number of healthy instances are always available. Auto Scaling can be configured to send an SNS email when:
    - An instance is launched.
    - An instance is terminated.
    - An instance fails to launch.
    - An instance fails to terminate.
- Automated backups of RDS. Update backups after a retention period.
- Automation of uploading snapshots of EBS to S3.
- Decouple components such that there are minimum dependencies among them and can communicate with each other asynchronously. SQS can be used for this purpose of isolating components and buffers b/w them. Components should be stateless and state should be stored outside of components.
- Store and retrieve machine configuration dynamically.
- Design build process to dump the latest build to the bucket.
- Build resource management tools (automated script) and config management tools (Chef, Puppet).
- Bundle a just enough operation system and dependencies into AMI.
- Reduce launch times by booting from Amazon EBS volumes.
- Application components should not assume health and location of hardware on which they are hosted. A dynamic IP address can be attached to a new node on a cluster.
- Think parallel. Use parallel computations whenever possible and cloud resources allow you to achieve it.
    - Multi-thread requests to amazon services.
    - Use amazon elastic map reduce for parallel processing of batch jobs.
    - Use an elastic load balancer to spread load across multiple servers.
- Keep static content closer to the end-user and dynamic data closer to compute.
- Amazon cloudfront can be used to cache content in S3 across all edge locations around the world.
- Make critical components of architecture redundant. This can be done in two ways.
    - **Standby Mode:** A secondary or standby component runs side by side with the primary component. When primary fails, the standby component takes over. This mode is used for stateful applications.
    - **Active Mode:** No components are designed as primary or standby; all components are performing the same task simultaneously. Tasks of failed components are distributed to another component. This mode is used for stateless applications.

Disaster Recovery Strategies

Pilot Light

Pilot light is a disaster recovery strategy where a minimal copy of the production environment is always running in a separate region. This environment typically includes the core infrastructure and data needed to boot up the full production environment quickly in the event of a disaster.

Warm Standby

Warm standby is a disaster recovery strategy where a fully functional copy of the production environment is always running in a separate region. This environment is typically scaled down to reduce costs, but it can be scaled up quickly to handle full production traffic in the event of a disaster.

Backup and Recovery

This is the lowest cost DR approach that simply entails creating online backups of all data and applications.

Feature	**Warm Standby**	**Pilot Light**	**Backup and Recovery**
Level of Readiness	Always ready to handle production traffic	Needs to be scaled up and configured before it can handle production traffic	Requires time to restore the backup to a new environment
Cost	More Expensive	Less Expensive	Less expensive
RTO (Recovery Time objective)	Shorter	Longer	Longer
RPO (Recovery Point Objective) - maximum amount of data loss acceptable	Depends on the frequency of backups	Typically very low	Typically very low

Multi-Site

A multi-site solution runs on AWS as well as on your existing on-site infrastructure in an active-active configuration.

Networking

Network Foundations

Amazon VPC

It is used for logical isolation of resources within the cloud.

IPV4: It is a group of 4 bytes. Each byte is converted into a decimal format e-g 192.168.1.30
CIDR: Classless Inter-domain routing, it is a notation for representing a range of IP addresses. It is a way to tell how many IP addresses are available. For example, 192.168.1.30/24 means that the first 24 bits of the IP address are fixed and the last 8 bits are flexible, meaning 256 IP addresses are available.

Although you should not have identical entries in route tables that apply to the same resource, the CIDR blocks can overlap. When the CIDR blocks for route table entries overlap, the more specific (smaller range) CIDR block takes priority.

IPv6 addresses are all public and the range is allocated by AWS.

In AWS, we choose network size using CIDR notation, /28 is the smallest range of IP addresses (16) and /16 is the largest range (65,536).

- By default, you can have 5 VPCs within a region. This limit can be increased upon request.
- For each region, you need to have a separate VPC.
- When you create a VPC, you need to select a region and range of IP addresses (CIDR notation). Each VPC can have up to 5 CIDRs, one primary and one secondary.
    - Once range of IP addresses are defined for VPC it can't be modified
- Then, space inside VPC is divided into smaller subspaces called subnets.
    - Subnets need to live inside an Availability Zone (AZ).
    - Each subnet has a CIDR range.
- Resources are placed inside of these subnets.
- Public resources are placed inside a public subnet and private resources in a private subnet. Private subnets will not be connected to the internet.
- By default, whatever is inside the custom VPC is only accessible within the VPC.
- Generally, VPC is created with an IP address range of /16 and subnets with an IP range of /24.

Subnet

AWS reserves five IP addresses (first 4 and last one) in each subnet for routing, network management and Domain Name Systems. When you create a new subnet, it is automatically associated with the main route table.

Each subnet must reside entirely within one Availability Zone.

The subnet that's associated with a Route Table that's connected to an internet gateway is public. A subnet with a Route Table that's not connected to an internet gateway is private.

Private

Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate data center via an IPsec Virtual Private Network (VPN) tunnel.

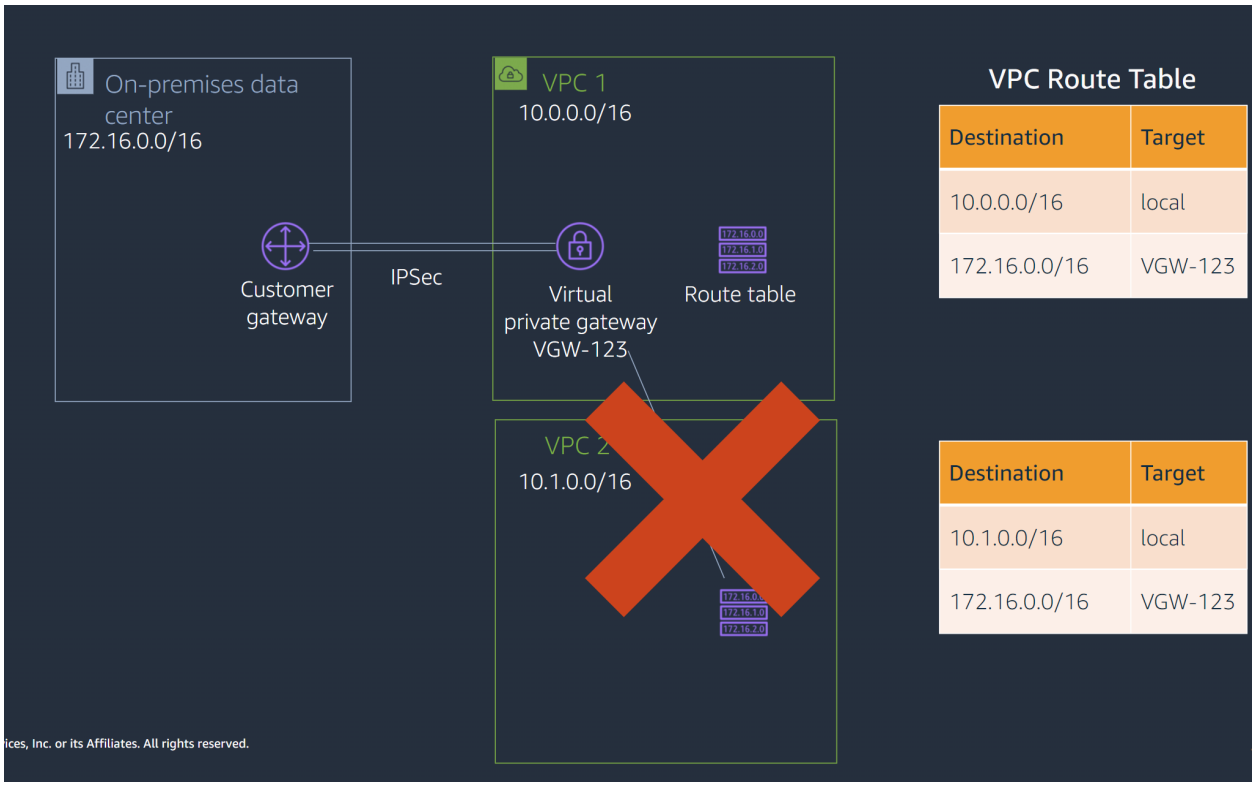
Virtual Private Gateway

If you want to make resources available to your on-premise data center only (or to other private networks) , then you need to create a Virtual Private Gateway and attach it to VPC. Then via VPN connection the data center can access VPC resources.

Virtual Private Gateway is attached to the AWS side while, customer gateway (a physical device) needs to be attached to the other side of the private network. Then an encrypted connection is established between these two gateways. This connection can further be strengthened using **AWS Direct Connect**. This acts as an interface between two sides of the connection (e-g data center and AWS VPC). It is a dedicated private network connection.

It resides inside a region.

One VPC can not use a VPG attached to another VPC directly.



Multiple VPN connections to the same virtual private gateway are possible.

AWS Direct Connect connection takes too long to provision. In cases where both can be used VPG will be preferred. Direct connect bypasses Internet service providers and removes network congestion.

An IPSec VPN can be used to connect to AWS; however, it does not bypass the ISPs or the Internet.

Default VPC

AWS creates a default VPC for every region. However, the resources inside the default VPC are publicly available on the internet.

In default VPC, instances will be assigned a public and private IP addresses and DNS hostnames (not very human friendly). In a non-default VPC, instances will be assigned a private but not a public DNS hostname.

Within the default VPC, there is a default security group, with following configurations:

- There is an inbound rule that allows all traffic from the security group itself.
- There is an outbound rule that allows all traffic to all addresses.

Note:

Custom security group without rules in VPC:

- There is an outbound rule that allows all traffic to all IP addresses.
- There are no inbound rules and traffic will be implicitly denied.

Route Table

Route Table is just a list of CIDR blocks (IP ranges) that our traffic can leave and come from. By default, newly created Route Tables will have the CIDR of our VPC defined. This means that traffic from anywhere within our VPC is allowed.

In addition to a list of IP ranges that our Route Table connects traffic between, it also has Subnet Associations. Simply put, these are which subnets use this route table. A Route Table can have many subnets, but a subnet can belong to only one Route Table. A subnet is associated with a Route Table and the Route Table dictates what traffic can enter and leave the subnet.

Amazon VPC Routing

Traffic outside the VPC can enter the Internet Gateway but we need route tables to direct the traffic to resources it needs to access. Route table contains a set of rules or routes. When VPC is created a main route table is created for routing traffic to VPC and subnets. Using the local route defined in the main route table traffic can go from one subnet to another.

Whether a subnet can be accessed from the public or not it is defined in the route table. To control routing we create a custom route table that includes a local route that allows traffic to flow freely within VPC. Every custom route table has a local route by default. To allow access to public subnets, you create a route with an internet gateway target and associate it to public subnets. To create a route for a private subnet, you will not set an internet gateway to a route (it will be set as local) and you will associate the route with private subnets.

Routers interconnect subnets and direct traffic between Internet gateways, virtual private gateways, NAT gateways, and subnets.

- You cannot delete the main route table. You can, however, manually set another route table to become the main route table.
- We can not set a gateway route table as the main route table.
- We can replace the main route table with a custom subnet route table.
- We can explicitly associate the main route table with a subnet, even if it is implicitly associated.
- Each subnet has a route table the router uses to forward traffic within the VPC. Route tables also have entries to external destinations.
- If no route table has been specified, a subnet will be assigned to the main route table at creation time.

- There is a default rule that allows all VPC subnets to communicate with one another. This rule cannot be deleted or modified.
- If a subnet doesn't have a route to the Internet gateway but has its traffic routed to a virtual private gateway for a VPN connection, the subnet is known as a VPN-only subnet.
- The best practice is to create at least two VPN tunnels into your Virtual Private Gateway.
- Traffic can be routed from private subnet to public internet by creating a route from private subnet's route table to public NAT gateway or instance that will redirect traffic to internet gateway.

NAT instance

- NAT instances are managed by you. They are used to enable private subnet instances to access the Internet.
    - NAT instances are EC2 instances that are used, in a similar way to NAT gateways, by instances in private subnets to access the Internet. However, they are not redundant and are limited in bandwidth.
- A NAT instance must live on a single public subnet with a route to an Internet gateway. Private instances in private subnets must have a route to the NAT instance, usually the default route destination of 0.0.0.0/0.
- Performance is dependent on instance size.
- NAT instances can scale up instance size or use enhanced networking.
- NAT instances can scale out by using multiple NATs in multiple subnets.
- NAT instance can be used as a bastion host.

NAT gateways VS NAT instances

- NAT gateways are managed for you by AWS.
- NAT gateways are highly available within each AZ.
- They are not associated with any security groups and can scale automatically up to 45 Gbps.
- NAT instances are managed by you. They must be scaled manually and do not provide HA. NAT instances can be used as bastion hosts and can be assigned to security groups.
- Both a NAT gateway or a NAT instance can be used for downloading data over the internet from a private subnet.
- NAT gateway imposes a limit of 45 Gbps.
- Enterprises prefer NAT gateways as they are more secure (e.g., you cannot access them with SSH and there are no security groups to maintain).
- You can't use a NAT gateway to access VPC peering, VPN, or Direct Connect, so be sure to include specific routes to those in your route table.
- Using the NAT gateway as a bastion host server is not supported.

Endpoints

They are simply a connection point that allows VPCs to access AWS services (outside of VPC). Policies (VPC endpoint policy) can be enforced on endpoints,e-g one endpoint for one service and another endpoint for another service.

VPC endpoint

VPC endpoints enable private connectivity to services hosted in AWS from within your VPC without using an Internet gateway, VPN, Network Address Translation (NAT) devices, or firewall proxies.

A private API endpoint can only be accessed from your Amazon Virtual Private Cloud (VPC) using an interface VPC endpoint, which is an endpoint network interface (ENI) that you create in your VPC. Private endpoints are VPC endpoints and are connected to by instances in subnets via route table entries or via ENIs (depending on which service).

Amazon S3 and DynamoDB do not have interface endpoints.

- A VPC endpoint is used to access public services from a VPC without traversing the Internet.
- VPC endpoint is an Elastic Network Interface with a private IP address as an entry point for traffic destined to a supported service.
    - It is commonly used when the applications run on EC2 instances in private subnets and these applications are providing services (e-g transmitting healthcare data) to service consumers in different AWS accounts. This configuration uses a NLB and can be fault-tolerant by configuring multiple subnets in which the EC2 instances are running.
- With VPC endpoints, resources inside a VPC do not require public IP addresses to communicate with resources outside the VPC.

An interface VPC endpoint (interface endpoint) enables you to connect to services powered by AWS PrivateLink.

- VPC endpoint cannot be used for Direct Connect gateways.

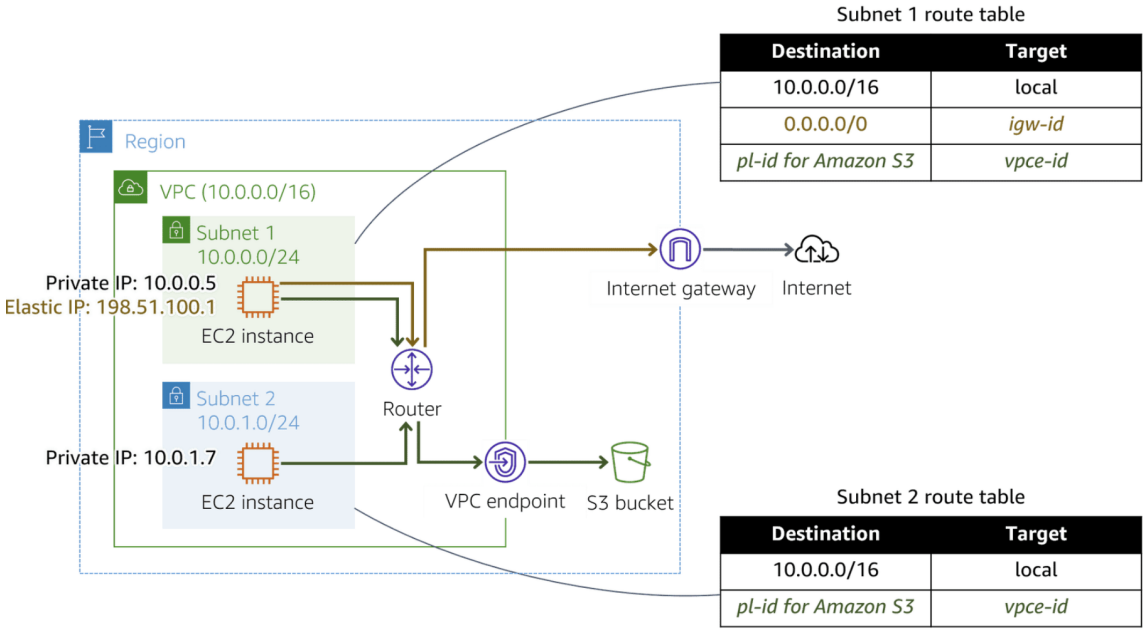
Gateway endpoint

A gateway endpoint is a gateway that is a target for a route that you have specified in your route table. It is used for traffic destined to a supported AWS service. Gateway endpoints are only available for Amazon DynamoDB and Amazon S3

- A policy is attached to a gateway to allow access to a service.
- With a gateway endpoint, you configure your route table to point to the endpoint. Amazon S3 and DynamoDB use gateway endpoints.

VPC endpoint vs Gateway endpoint

	VPC Endpoint	Gateway Endpoint
What	Elastic Network Interface with a Private IP	A gateway that is a target for a specific route
How	Uses DNA entries to redirect traffic	Uses prefix lists in the route table to redirect traffic
Which services	API Gateway, CloudFormation, CloudWatch, etc.	Amazon S3, DynamoDB
Security	Security Groups	Endpoint policies



In the above diagram, instances in subnet 1 can send and receive traffic to and from the internet and the S3 bucket. Instances in subnet 2 only have access to the S3 bucket.

Interface endpoints

Powered by AWS PrivateLink, an interface endpoint is an elastic network interface with a private IP address from the IP address range of your subnet. It serves as an entry point for traffic destined to a supported AWS service or a VPC endpoint service.

Gateway Load Balancer Endpoint

A Gateway Load Balancer endpoint is an elastic network interface with a private IP address from the IP address range of your subnet. This type of endpoint serves as an entry point to intercept traffic and route it to a service that you've configured using Gateway Load Balancers, for example, for security inspection. You specify a Gateway Load Balancer endpoint as a target for a route in a route table. Gateway Load Balancer endpoints are supported for endpoint services that are configured for Gateway Load Balancers only.

Like interface endpoints, Gateway Load Balancer endpoints are also powered by AWS PrivateLink.

DNS

When an interface endpoint is created, endpoint-specific DNS hostnames are generated that can be used to communicate with the service. After creating the endpoint, you can submit requests to the provider's.

Pricing

Interface endpoints and Gateway Load Balancer endpoints are charged for each hour the VPC endpoint remains provisioned in each Availability Zone and for each gigabyte processed through the VPC endpoint.

There is no additional charge for using gateway endpoints. Standard charges for data transfer and resource usage apply. You might be able to reduce costs by selecting gateway endpoints for traffic destined to DynamoDB or Amazon S3.

Bastion

A bastion is a host that is needed for remote connectivity to private instances over the public internet. It sits between private instances and the internet (if Network ACLs and Security Groups are configured correctly).

Service Gateway

- Managed service that enables private connectivity between VPCs and AWS services.
- Simplifies access to AWS services that aren't accessible through VPC endpoints.
- Supports traffic encryption for secure communication.
- Does not require customer-managed VPN hardware or software.

VPC Peering

A networking connection between two VPCs and VPCs communicate with each other as if they are in the same network (VPCs can be in different regions and different accounts). AWS uses the existing infrastructure of a VPC to create a VPC peering connection. It is neither a gateway nor a VPN connection and does not rely on a separate piece of physical hardware. There is no single point of failure for communication or a bandwidth bottleneck.

Only connected peers can communicate with each other using this connection, transitive peering relationships are not supported.

A VPC Peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses.

Each VPC needs to define a route to the other VPC. You might need to update Security Group rules as well.

All inter-Region traffic is encrypted with no single point of failure or bandwidth bottleneck. Traffic always stays on the global AWS backbone and never traverses the public internet, which reduces threats such as common exploits and distributed denial of service (DDoS) attacks. Inter-Region VPC peering provides an uncomplicated and cost-effective way to share resources between Regions or replicate data for geographic redundancy.

Pricing

There is no charge for setting up or running a VPC peering connection. Data transferred across peering connections is charged per gigabyte for send and receive, regardless of the Availability Zones involved.

Limitations

- Only the owner of the VPC can initiate and accept a peering connection request.
- IP ranges (CIDR blocks) should not overlap between VPCs.
- One VPC can only accept 125 connections at max.

Network Access Control

A firewall (who can enter or leave) for the subnet. By default every subnet has an ACL that allows traffic in and out of the subnet. The configurations can be updated as needed.

It is not ideal to use Network ACL to block the IP address ranges associated with the specific countries as it will be extremely difficult to manage.

A VPC automatically comes with a modifiable default network ACL. By default, it allows all inbound and outbound IPv4 traffic. Custom network ACLs deny everything inbound and outbound by default.

Network ACLs are stateless, which means that if you add a rule for inbound traffic, then you must also add the same rule for outbound traffic. Your network ACL needs to explicitly see that traffic that was allowed inbound is also allowed out.

Network ACLs (NACLs) function at the subnet level. The VPC router hosts the network ACL function. With NACLs, you can have permit and deny rules. Network ACLs contain a numbered list of rules that are evaluated in order from the lowest number until the explicit deny. NACLs only apply to traffic that is ingress or egress to the subnet, not to traffic within the subnet.

Network ACL can apply to many subnets, but a subnet can only belong to one Network ACL.

Security Groups (SGs)

These are applied at the resource level (e-g EC2 instance) level and are not optional. Default configuration of SGs denies all inbound traffic but allows all outbound traffic. SGs are stateful, for example, if inbound rule is set for a particular source then we don’t need to set outbound rule

to allow traffic from our resource back to source. A security group can control both ingress and egress traffic.

- By default, custom security groups do not have inbound allow rules (all inbound traffic is denied by default).
- By default, default security groups have inbound allow rules (allowing traffic from within the group).
- All outbound traffic is allowed by default in both custom and default security groups.

Normally, we configure SGs and keep the default configuration of ACLs.

Security groups cannot block traffic by country. SGs are used to allow/disallow outbound and inbound connections; they cannot be used to allow read and write operations on resources.

Security group members can be within any AZ or subnet within the VPC. Security group membership can be changed whilst instances are running.

We can attach multiple security groups to servers / services. The resulting security is the sum of the security group rules, i.e. if one allows HTTP traffic and the other allows SSH traffic, the result is allowed traffic from both.

Security groups are stateful, and have inbound and outbound rules.

Limitations

- You cannot create deny rules in security groups.
- SGs do not apply to S3 buckets.

Security Group vs Network ACL

Security Group	Network ACL
Operates at the instance (interface) level	Operates at the subnet level
Supports allow rules only	Supports allow and deny rules
Stateful	Stateless
Evaluates all rules	Processes rules in order
Applies to an instance only if associated with a group	Automatically applies to all the instances in the subnets that it is associated with
By default, default security groups have inbound allow rules (allowing traffic from within the group). Custom security groups do not have inbound allow rules (all inbound traffic is denied by default).	By default, it allows all inbound and outbound IPv4 traffic. Custom network ACLs deny everything inbound and outbound by default.

All outbound traffic is allowed by default in both custom and default security groups.	

AWS Managed VPN

AWS managed IPSec VPN connection over your existing internet. A quick and usually simple way to establish a secure tunneled connection to a VPC. For this connection, you need to create a virtual private gateway (VPG) on AWS, and a Customer Gateway on the on-premises side.

For route propagation, you need to point your VPN-only subnet's route tables at the VPG.

It has two dimensions for cost i) Number of hours ii) Amount of Data transferred

AWS VPN CloudHub

AWS VPN CloudHub is a managed VPN service that simplifies connectivity between multiple AWS Virtual Private Clouds (VPCs) within a single region. It eliminates the need for individual site-to-site VPN connections between each pair of VPCs, offering a more automated and centralized approach.

- Hub-and-spoke topology: Each VPC connects to a central "hub" gateway, simplifying configuration and reducing complexity.
- Automated routing: Routes traffic between spokes securely without manual configuration of routing tables.
- Scalability: Supports a large number of VPC connections (up to 500 spokes per hub) without performance degradation.
- Cost-effective: Pays only for the VPN connections actively used and the data transferred through the hub.
- Easy integration: Seamlessly integrates with other AWS services like VPC Peering and AWS Direct Connect.

VPN CloudHub is used for hardware-based VPNs and allows you to configure your branch offices to go into a VPC and then connect that to the corporate DC (hub and spoke topology with AWS as the hub).

AWS VPN CloudHub also uses the public Internet, so it is not a private or reliable connection. It is not the best solution, if the requirement is to implement high-bandwidth, low-latency connections.

Use Cases:

- Connecting multiple branches, offices, or applications within a single AWS region.
- Backing up data between VPCs for disaster recovery purposes.
- Sharing resources and data securely between different VPC environments.
- Building hybrid cloud architectures by connecting VPCs to on-premises networks through an existing VPN connection to the hub.

Software VPN

Amazon VPC offers you the flexibility to fully manage both sides of your Amazon VPC connectivity by creating a VPN connection between your remote network and a software VPN appliance running in your Amazon VPC network. This option is recommended if you must manage both ends of the VPN connection either for compliance purposes or for leveraging gateway devices that are not currently supported by Amazon VPC's VPN solution. Software VPN should follow AWS VPNs protocol.

Building on the Software VPN design mentioned above, you can create a global transit network on AWS. A transit VPC is a common strategy for connecting multiple, geographically dispersed VPCs and remote networks in order to create a global network transit center. It simplifies network management and minimizes the number of connections required to connect multiple VPCs and remote networks.

Shared Services VPC

You can allow other AWS accounts to create their application resources (such as EC2 instances, Relational Database Service (RDS) databases, Redshift clusters, and Lambda functions) into shared, centrally-managed Amazon Virtual Private Clouds (VPCs).
VPC sharing enables subnets to be shared with other AWS accounts within the same AWS Organization.

VPCs can be shared among multiple AWS accounts. Resources can then be shared amongst those accounts. However, to restrict access so that consumers cannot connect to other instances in the VPC, the best solution is to use PrivateLink to create an endpoint for the application. The endpoint type will be an interface endpoint and will use an NLB in the shared services VPC. VPC peering could be used along with security groups to restrict access to the application and other instances in the VPC. However, this would be administratively difficult as you would need to ensure the maintenance of the security groups as resources and also ensure that addresses change.

Hardware VPN Connection

A hardware VPN connection is a connection between your Amazon VPC and your datacenter, home network, or co-location facility.

- To propagate traffic coming to AWS public subnet to on-premise instance via hardware VPN connection, add a route for your remote network in the public subnet route table and specify VPG as the target.

Best Practices

If you are going to have two VPCs it's advisable not have matching or overlapping CIDR block for those VPCs reason being that you cannot create a VPN connection or a Direct Connect connection, or a VPC peering connection between two VPCs that have matching or overlapping CIDR range.

Amazon allows a customer to create a subnet with a slash 28 bit mask, however if you are going to have a load balancer in your network design, then you need to make sure that each Availability Zone subnet for your load balancer nodes, has a slash 27 bit mass. This just gives enough available IP addresses for the load balancer to scale properly in case of any large scale event.

Networking Gateways

A network gateway is a device or node that connects networks with different transmission protocols and performs protocol conversions to translate communications.

A gateway connects networks, and a router delivers data within a network. Gateways and routers are usually separate devices. However, it's becoming more common for their functions to be combined in a router. For example, in your home network, your router can also be your default gateway.

Internet Gateway

To make your VPC accessible via the internet, you need to create an internet gateway and attach it to VPC. There is a one to one relation between VPC and internet gateway. An Internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the Internet.

- An egress-only Internet gateway is a stateful gateway to provide egress-only access for IPv6 traffic from the VPC to the Internet.
- You cannot add a route to an Internet gateway to a private subnet route table

All IPv6's are internet accessible by default, so the way you make them "private" is by directing traffic out through an Egress Only Internet Gateway. This means that IPv6 hosts can initiate internet requests outwards (and receive responses), but that we can't access the IPv6 hosts outside of the VPC. Egress Only meaning, outgoing traffic only.

It resides inside a region.

Internet gateways are:

- Horizontally scaled
- Redundant

- Highly available

This means that even though each Amazon VPC has a single internet gateway, this internet gateway is not a bottleneck nor a single point of failure.

Customer Gateway

It is a physical or software appliance that you own or manage in your on premises network. Applications include manages routing to and from your environment.

Same customer gateways can be reused for multiple site-to-site VPN connections, but you must consider high availability.

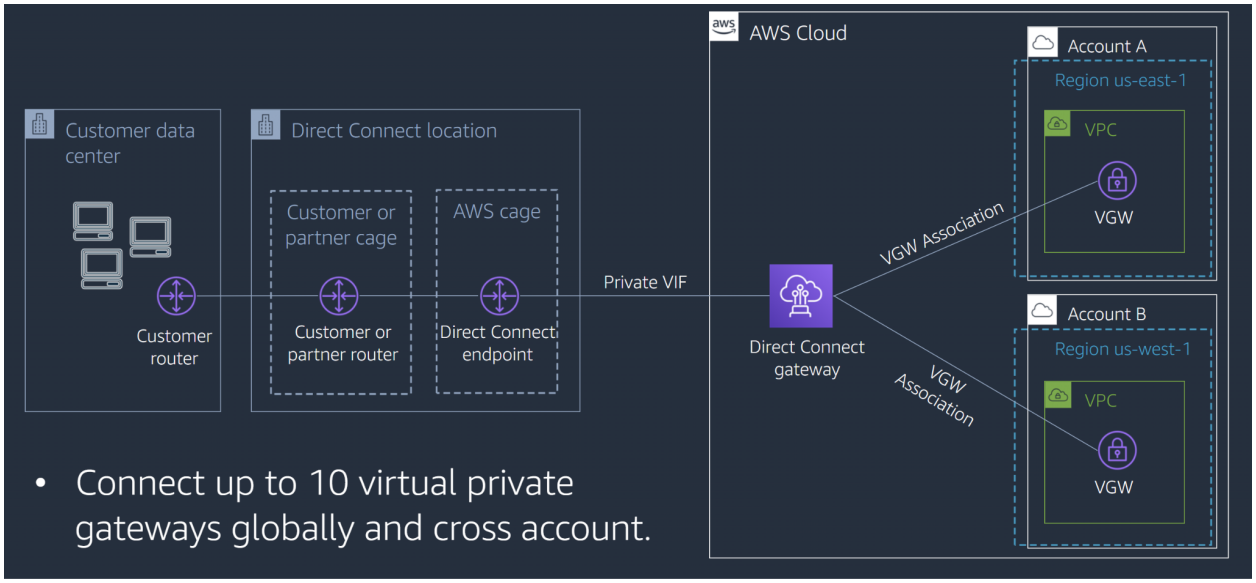
It resides locally.

VPN Gateway

It is the gateway on the AWS side of site-to-site VPN connection. Applications include Amazon EC2 instances, Amazon S3, Amazon RDS, Amazon Lambda, and so on.

Direct Connect Gateway

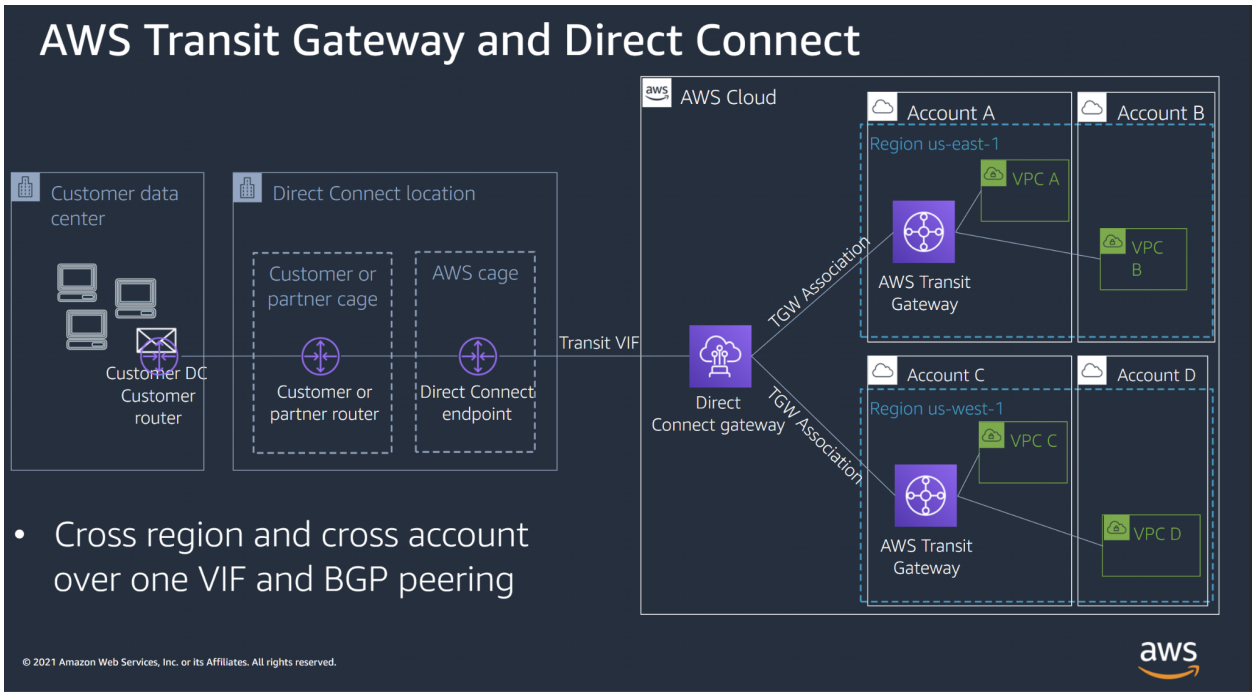
Direct Connect gateway provides a grouping of Virtual Private Gateways (VGWs) and Private Virtual Interfaces (VIFs) that belong to the same AWS account and enables you to interface with VPCs in any AWS Region (except the AWS China Region).



It cannot be used to send traffic between VPCs. It is primarily used to allow connection from one Direct Connect to VPCs in different regions.

Direct Connect Gateway is a global resource not a regional one, but it is created using a regional account.

It can be associated with a VPG or transit Gateway



NAT Gateway

NAT translates a single external address (external internet address) to multiple internal addresses. It resides inside a region. AWS provides NAT gateways to implement NAT services.

It is used so that instances in a private subnet can access external services outside of VPC but external services cannot initiate a connection with these instances. Traffic is routed from NAT gateway to internet gateway. To connect to other VPC or on-premises networks, you route traffic from NAT gateway through a transit gateway or a virtual private gateway.

NAT gateway replaces the source ip address with the NAT gateway IP address. It will be a private IP address for private NAT gateway and elastic IP address for public NAT gateway. When sending response traffic back to the original source (instances) the IP addresses are translated to the original source IP address. It only enables outbound internet access.

NAT gateway has to be created for each availability zone. Route table is used to enable communication cross-AZ

The route tables of the private subnets where the EC2 instances reside are configured to forward Internet-bound traffic to the NAT gateway. You pay for using a NAT gateway based on hourly usage and data processing.

A NAT gateway is used by instances in private subnets to access the Internet, which is less secure than a VPC endpoint.

You cannot route traffic to a NAT gateway through a VPC peering connection, site-to-site VPN connection, or Direct Connect.

NAT Gateway is used for Internet connectivity through private subnets for IPv4 addresses, not for IPv6 addresses. Egress Only Internet Gateway (IPv6). Specifically designed for IPv6 traffic, allowing outbound IPv6 connections without permitting inbound connections. It doesn't perform network address translation, as IPv6 has a vast address space eliminating the need for NAT in most cases.

AWS Transit Gateway

You can build a hub-and-spoke topology with AWS Transit Gateway that supports transitive routing. This simplifies the network topology and adds additional features over VPC peering. AWS Resource Access Manager can be used to share the connection with the other AWS accounts. This is the preferred solution for connecting multiple VPCs (with transitive routing).

You can manage a single connection for multiple VPCs or VPNs that are in the same region by associating a Direct Connect gateway to a transit gateway. The solution involves the following components:

- A transit gateway that has VPC attachments
- A Direct Connect gateway
- An association between the Direct Connect gateway and the transit gateway
- A transit virtual interface that is attached to the Direct Connect gateway.

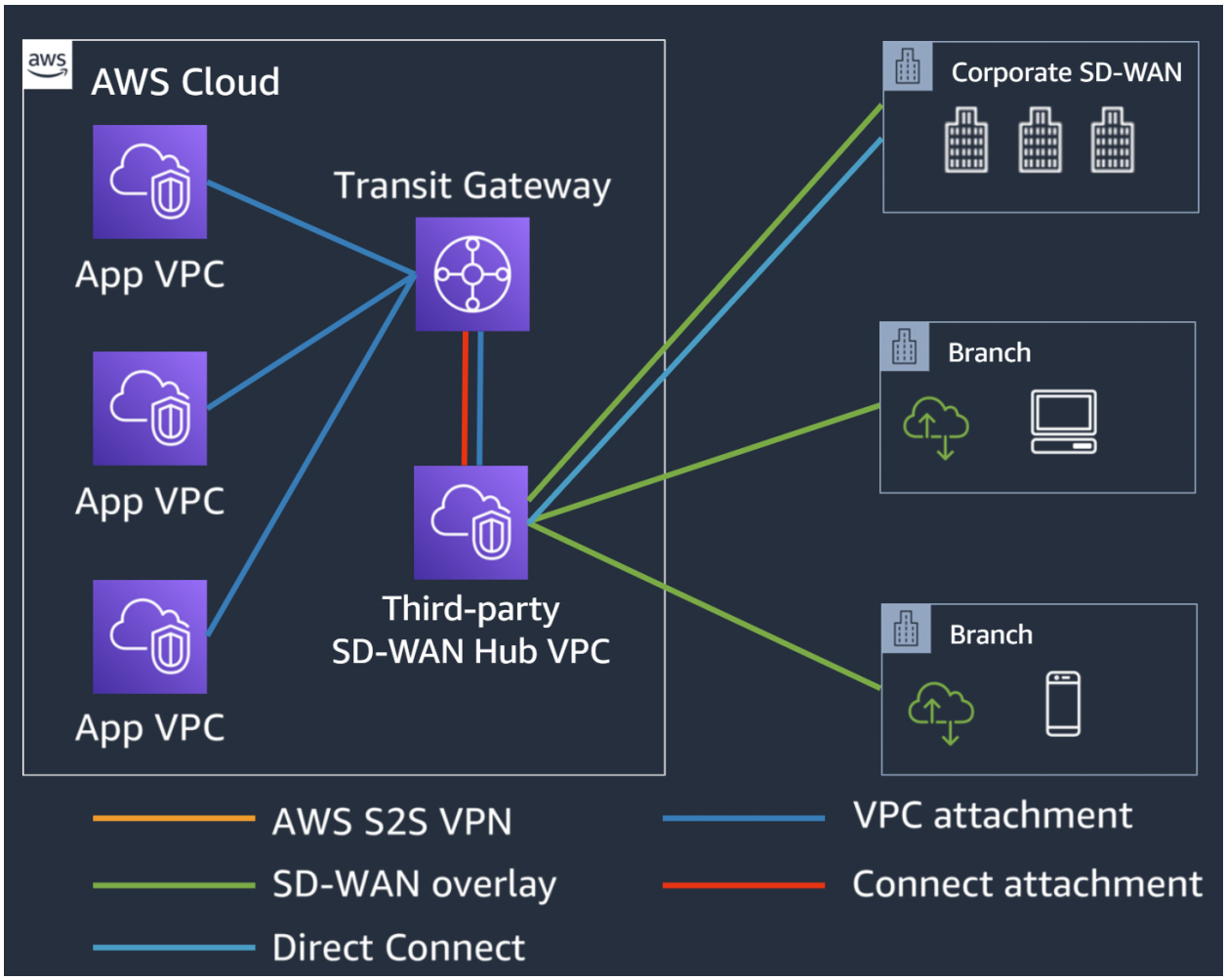
This is useful when you have direct connections to multiple on-premise centers and you want to connect to multiple VPCs.

Inter-Region peering connects AWS Transit Gateways together using the AWS global network. This adds automatic encryption for your data, and your data never travels over the public internet (only for inter-region peering between transit gateways). Transit gateways adds scalability, connectivity, better visibility and control, and improved security over peering connections. Transit gateways can also be used to connect your AWS environment to your on premises infrastructure creating a hybrid network of AWS and physical networks. Transit gateway offers AWS Transit Gateway Network Manager, which adds a unique view over your entire network, even connecting to Software-Defined Wide Area Network (SD-WAN) devices.

It resides inside a region.

Key Concepts

Attachment - The connection from Amazon VPC, a VPN, Direct Connect, or a connect attachment to a transit gateway.



Association - The route table used to route packets from an attachment (from Amazon VPC and VPN).

Propagation - The route table where the attachment's routes are installed.

Route table

A transit gateway has a default route table and can optionally have additional route tables. A route table includes dynamic and static routes that decide the next hop based on the destination IP address of the packet. The target of these routes can be any transit gateway attachment (One or more VPCs, SD-WAN appliance, Direct Connect Gateway, peering connection with another transit gateway, A VPN connection to a transit gateway).

Each attachment is associated with exactly one route table. Each route table can be associated with zero to many attachments.

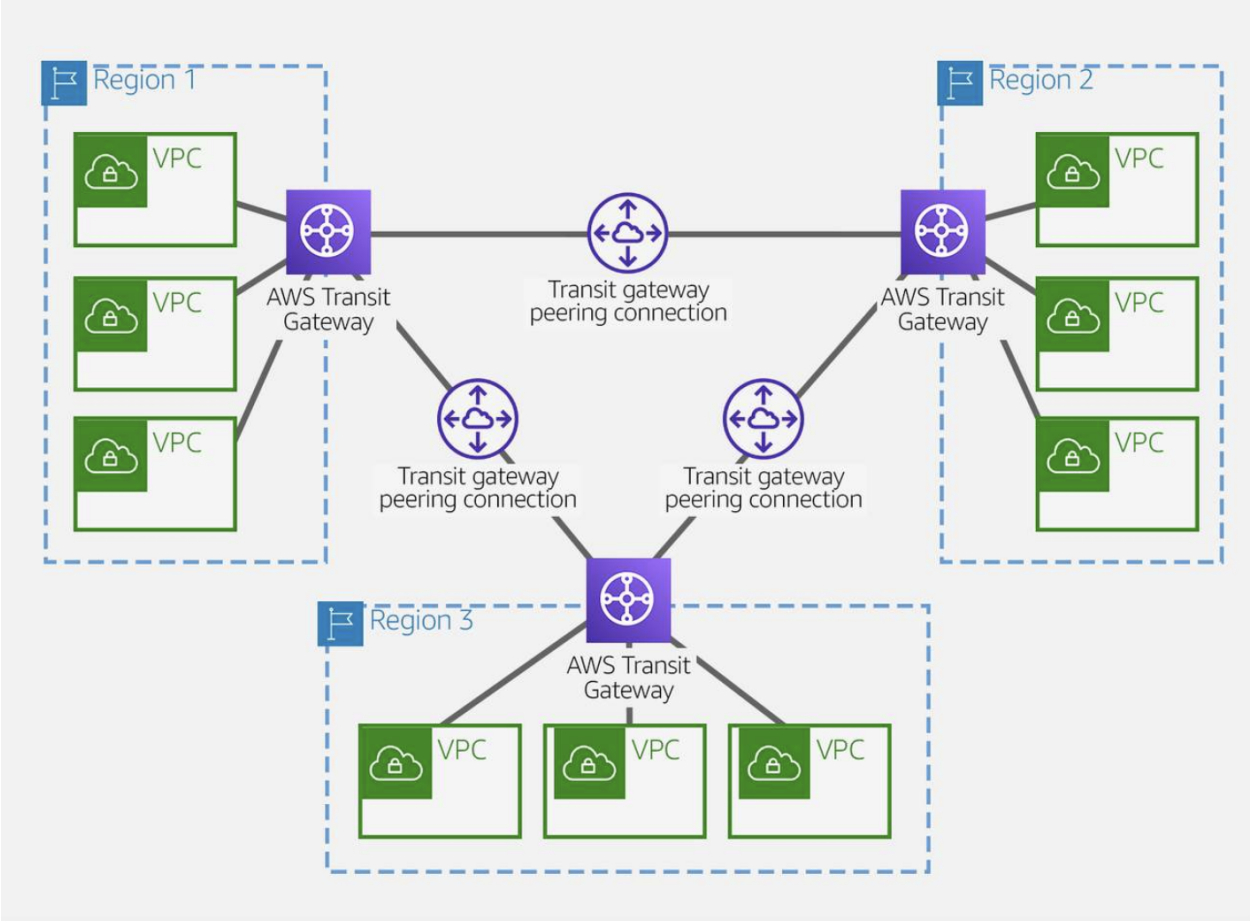
With a VPC, you must create static routes to send traffic to the transit gateway.

With a VPN connection or a Direct Connect gateway, routes are propagated from the transit gateway to your on-premises router using BGP.

With a peering attachment, you must create a static route in the transit gateway route table to point to the peering attachment.

AWS Transit Gateway inter-regional peering

AWS offers two types of peering connections for routing traffic between VPCs in different Regions: VPC peering and transit gateway peering. Both peering types are one-to-one, but transit gateway peering connections have a simpler network design and more consolidated management.



Pricing

AWS Transit Gateway charges for the number of connections per hour and per GB of data processed.

Virtual Private Gateway vs Transit Gateway

AWS Direct Connect Gateway is used for connecting multiple VPCs to multiple on-premise data centers. Transit Gateway and VPG can be used with direct connect gateway to allow other VPCs to use existing direct connect connections (that were created for one of the VPCs to connect with on-premise data centers). Transit Gateway is used when you have multiple VPCs in the same region and VPG is used for connecting VPCs across regions.

Transit Gateway VS VPN CloudHub

- Choose VPN CloudHub:
    - For connecting fewer VPCs within a single region.
    - If cost-effectiveness and ease of use are primary concerns.
    - You don't require advanced routing or multi-region connectivity.
- Choose Transit Gateway:
    - For complex hybrid cloud architectures and high-bandwidth workloads.
    - If you need multi-region connectivity or granular control over routing and security.
    - You have numerous VPCs and on-premises networks requiring centralized management.

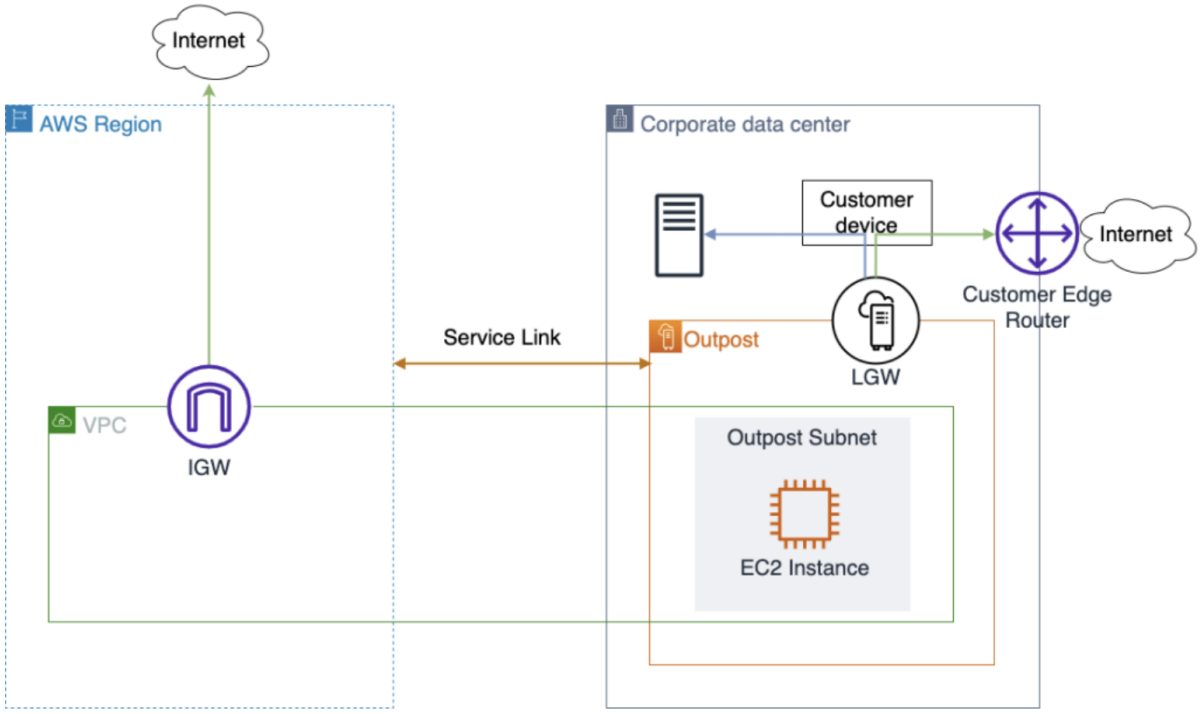
Service Mesh Virtual Gateway

It allows resources that are outside of your mesh network to communicate to resources that are inside. Applications include Amazon EC2, Amazon ECS, and Amazon EKS.

AWS Outposts Local Gateway

In the context of Amazon Web Services (AWS) Outposts, a local gateway is a service that enables secure connectivity between your on-premises network and your AWS Outpost deployment. It acts as a bridge between the two environments, allowing you to:

- Extend your on-premises network to your Outpost: This lets you treat your Outpost as an extension of your local network, simplifying network management and resource access.
- Connect to the internet through your on-premises network: You can utilize your existing internet connection on-premises to reach the internet from your Outpost resources, potentially saving costs compared to using internet access directly from the Outpost.
- Control network traffic flow: The local gateway provides granular control over how network traffic flows between your on-premises network and your Outpost, allowing you to implement security policies and optimize performance.



AWS PrivateLink

PrivateLink is a VPC endpoint service that solves the problems of needing to expose an application to other Amazon VPCs in other AWS accounts. You could:

- Make the application public, but then you are using the internet.
- Also set up VPC peering, but that may be more management overhead than necessary for what you are trying to accomplish. And VPC Peering connections are only a one-to-one connection.

Network traffic that uses PrivateLink doesn't traverse the public internet. This reduces exposure to brute force and distributed denial-of-service attacks, along with other threats. You can use private IP connectivity so that your services function as though they were hosted directly on your private network. You can also associate security groups and attach an endpoint policy to interface endpoints, which allow you to control precisely who has access to a specified service.

Know the difference between AWS PrivateLink and ClassicLink. ClassicLink allows you to link EC2-Classic instances to a VPC in your account within the same region.

You cannot use private link connections to publish a Direct Connect Gateway.

Setting up and maintaining a VPN connection has more operational overhead than PrivateLink.

Limitations

AWS PrivateLink does not support IPv6.

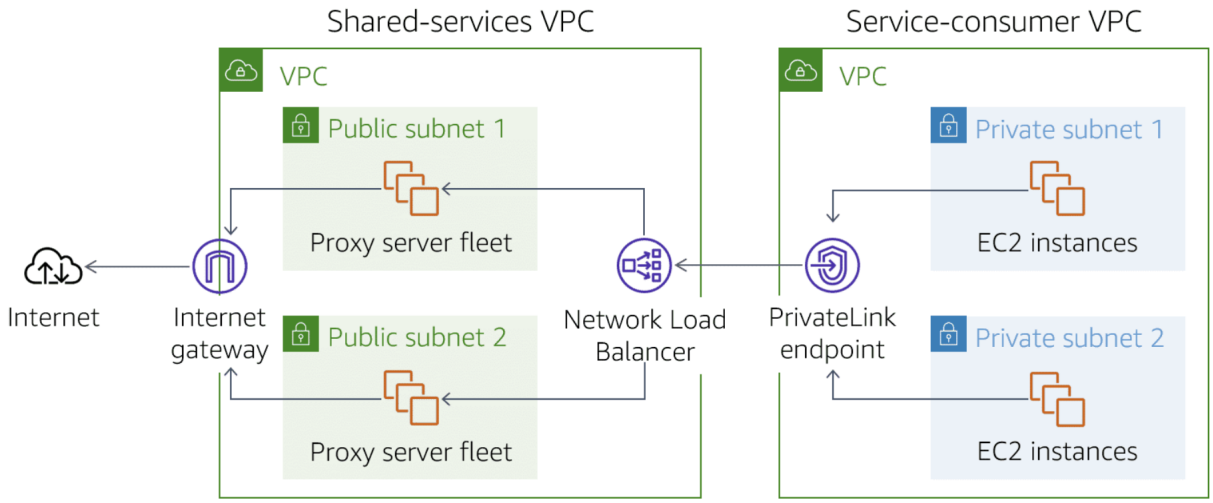
Pricing

You will be billed for each hour that your VPC endpoint remains provisioned in each Availability Zone, irrespective of the state of its association with the service.

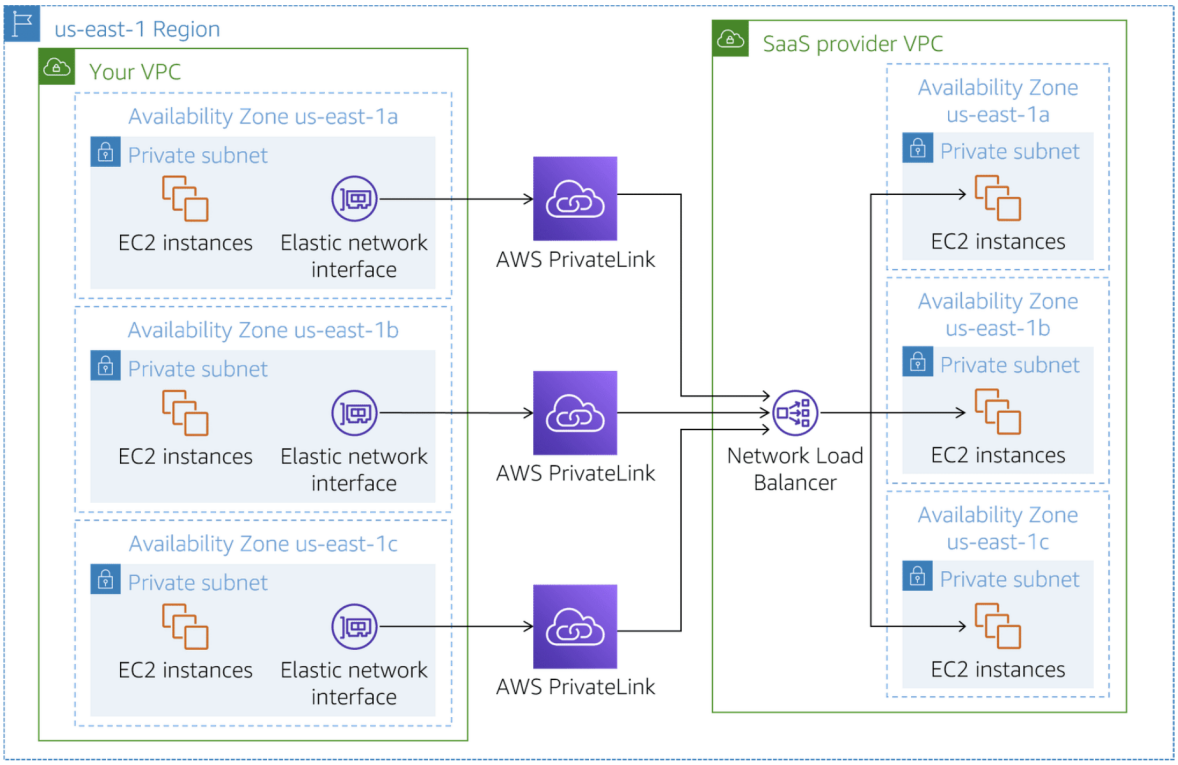
AWS Private Link vs AWS Direct Connect

Feature	**AWS Private Link**	**AWS Direct Connect**
Connectivity	VPCs to AWS Services	On-premise networks to AWS locations
Network Type	Private	Dedicated
Performance	Good	High
Security	High	High

In the following diagram, traffic from Amazon Elastic Compute Cloud (Amazon EC2) instances in private subnets is routed to a Network Load Balancer. The Network Load Balancer is connected to instances in public subnets that communicate with the internet. This architecture permits backend EC2 instances to communicate with the front-end instances through the AWS PrivateLink endpoint. And it avoids the security and cost implications of data traveling through the public internet.



In the following diagram AWS private link is used to send data to security company's logging and analytics software (SAAS) offering to ensure that data does not go through public circuit.



Hybrid Connectivity

AWS Direct Connect

Dedicated network connection over private lines straight into the AWS backbone. This private connection can reduce network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections. AWS Direct Connect does not encrypt your traffic that is in transit. You can use the encryption options for the services that traverse AWS Direct Connect. Route tables need to be updated to point to a Direct Connect connection.

When choosing to implement a Direct Connect connection, you should first consider bandwidth, connection type, protocol configurations (Border Gateway Protocol, Bidirectional Forwarding Detection), and other network configuration (IPV4, IPV6) specifications.

To create a standalone connection by logging in to the console using your account. Go to the Direct Connect dashboard and begin to configure the Direct Connect connection. When you have configured your connection, AWS will provide you with a Letter of Authorization and Connecting Facility Assignment, or LOA-CFA. You will share your LOA-CFA with your Direct

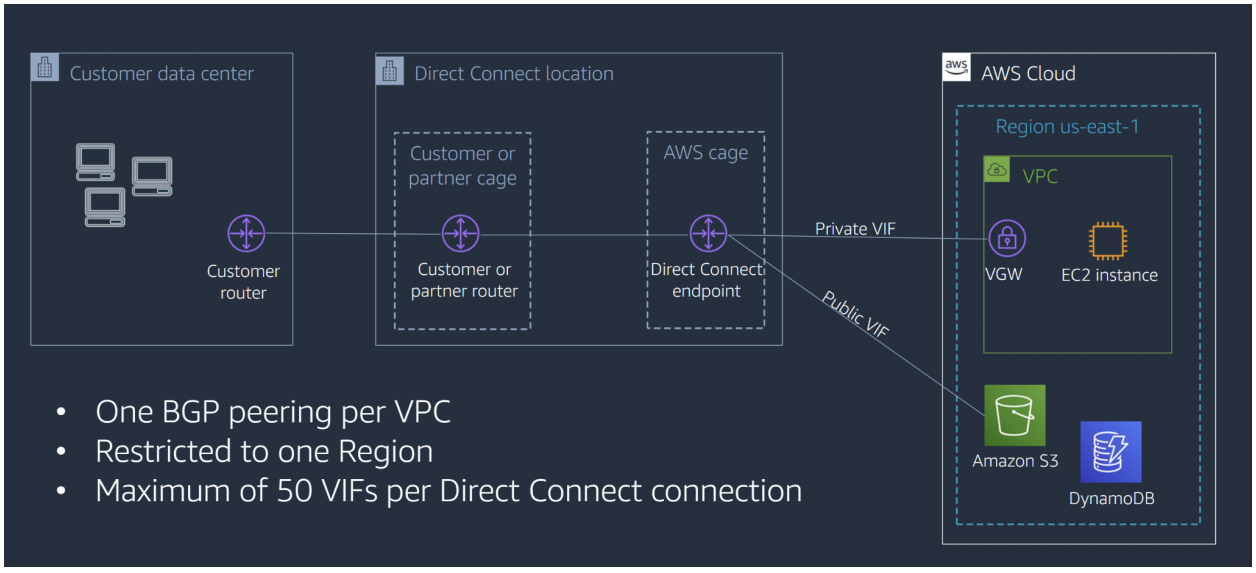
Connect Partner, showing them that AWS has authorized the completion of the last physical step for your Direct Connect connection.

After they receive the LOA-CFA, your Direct Connect Partner will physically complete the connection between your router and the AWS router with a cross connect. The next step in the process is to configure the virtual interface for your Direct Connect connection. AWS supports three types of virtual interfaces. Private, public, and transit.

The choices you made earlier for a public or private BGP and ASN will determine which of the three interfaces are available to you at this time. Choosing a private virtual interface lets you connect to all virtual private cloud, or VPC, resources within the private IP space in your AWS environment. Connect a single private virtual interface to multiple VPCs through private gateways within an AWS Region by associating it with your Direct Connect gateway.

Choosing a public virtual interface lets you route traffic to all VPC resources with a public IP address or that are connected to an AWS public endpoint. If you connect a public virtual interface to a Direct Connect location, you can connect to all public global AWS IP addresses and access AWS global IP route tables.

Choosing a transit virtual interface lets you connect your Direct Connect connection to AWS Transit Gateway. Then you can use the power of the AWS Transit Gateway and the AWS Transit Gateway Network Manager to manage the traffic moving between your AWS environment and your physical location. A transit virtual interface supports connecting three transit gateways to your Direct Connect gateway. Each connected transit gateway can connect to multiple VPCs within the same Region, even if they belong to different accounts. For this example, we will use a private virtual interface.

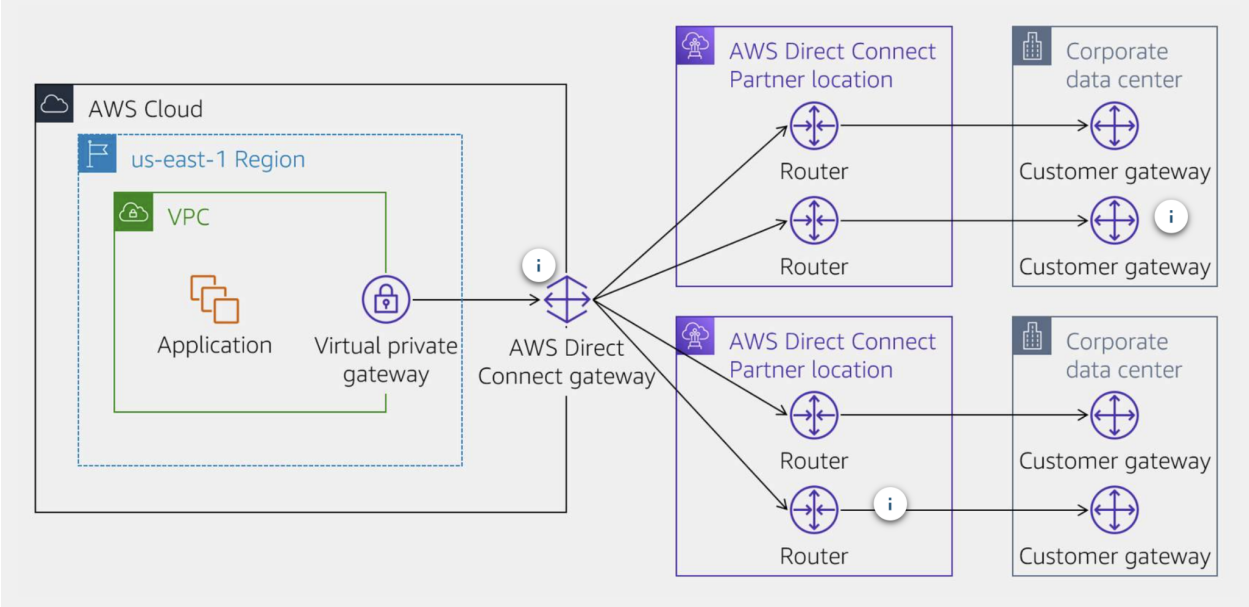


Virtual Interfaces

Hosted Virtual Interface: It is used to allow another account to access your Direct Connect link.
Transit Virtual Interface: A transit virtual interface is used to access Amazon VPC Transit Gateways.
Public Virtual Interface: This uses public internet.

Highly available hybrid network connections

If the company's router or circuit connecting to one of the company's data centers experience an issue at the Direct Connect Partner locations. The company will not be able to use the Direct Connect connection to reach AWS until the issue is resolved. To resolve this issue, following is the high available design:



In this design, the company has added a second router to the rack they rent. Each WAN circuit terminates to a different router, and each router has a separate cross-connect cable to the AWS router at the Partner location. Now, if one routers suffers a failure, the hardware gateway at the respective data center will detect the failure and redirect traffic to the active connection.

Two physical WAN circuit are added to each data center. This helps the company reduce the chances of an interruption to the company AWS environment.

Pricing

Direct Connect has two billing elements: port hours and outbound data transfer. Port hour pricing is determined by connection type (dedicated connection or hosted connection) and capacity. Data transfer out over Direct Connect is charged per GB.

AWS Direct Connect plus VPN

With AWS Direct Connect plus VPN, you can combine one or more AWS Direct Connect dedicated network connections with the Amazon VPC VPN. A VPG is used to set up an AWS VPN, which you can use in combination with Direct Connect to encrypt all data that traverses the Direct Connect link. This combination provides an IPsec-encrypted private connection that also reduces network costs, increases bandwidth throughput, and provides a more consistent network experience than Internet-based VPN connections. Internet-based VPNs are not reliable and can never guarantee latency over the internet.

AWS Direct Connect provides a secure, reliable, and private connection. However, lead times are often longer than 1 month, so it cannot be used to migrate data within one month.

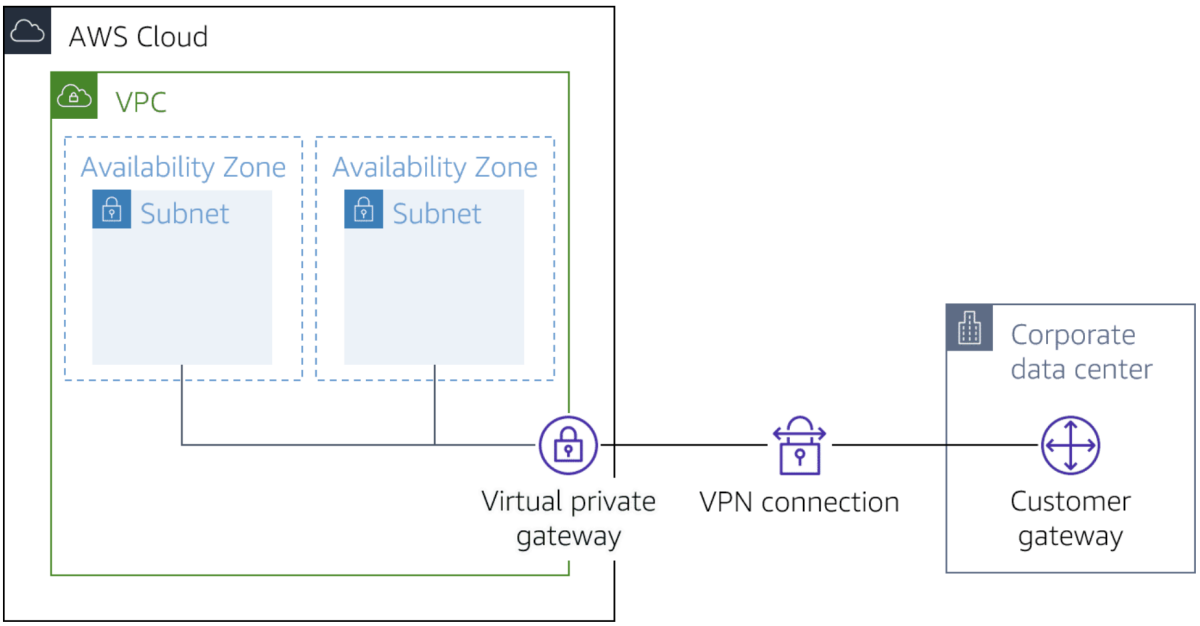
There is no option to enable IPSec encryption on the Direct Connect connection.

Transit Gateway vs VPC Peering

The VPC peering connection offers the best balance of cost and performance. Each transit gateway will incur a VPC connection and data transmission charge. VPC peering connections only incur a data transmission charge. VPC peering connections, unlike transit gateways, have no aggregate bandwidth restriction. Lastly, deploying a transit gateway adds a hop between the data source and the data destination. That could potentially increase the connection's latency. VPC peering does not add any additional hops.

AWS Site-to-Site VPN

If you need to connect remote offices to AWS, you can use AWS Site-to-Site VPN to create secure and encrypted connections quickly.



We can use Virtual Private Gateway, Transit Gateway on Amazon depending on the use case.

Monitoring

You can monitor VPN tunnels using Amazon CloudWatch, which collects and processes raw data from the VPN service into readable, near real-time metrics. These statistics are recorded for a period of 15 months. You can access historical information and gain a better perspective on how your web application or service is performing. VPN metric data is automatically sent to CloudWatch as it becomes available.

Limitations

- IPv6 traffic is partially supported. IPv6 for outer tunnel connection not supported.
- Throughput of AWS Site-to-Site VPN connections is limited.
- AWS Site-to-Site VPN endpoints use public IPv4 addresses and therefore require a public virtual interface to transport traffic over Direct Connect. Support for AWS Site-to-Site VPN over private Direct Connect is not yet available.
- For globally distributed applications, the accelerated Site-to-Site VPN option provides a connection to the global AWS backbone through AWS Global Accelerator. Because the Global Accelerator IP space is not announced over a Direct Connect public virtual interface, you cannot use accelerated Site-to-Site VPN with a Direct Connect public virtual interface.

Pricing

- AWS Site-to-Site VPN connection per hour (varies by Region).
- Data transfer out charges.
- Hourly charges for two AWS Global Accelerators per VPN connection.

AWS Client VPN

If you need to connect remote team access, you can use AWS Client VPN to securely connect your remote team to AWS and your on premises resources. With Client VPN, you can access your resources from any location using an OpenVPN-based VPN client.

Your Client VPN administrator creates and configures a Client VPN endpoint in AWS. Your administrator controls which networks and resources you can access when you establish a VPN connection.

VPN Client application is the software application that you use to connect to the Client VPN endpoint and establish a secure VPN connection.

This is a configuration file that is provided to you by your Client VPN administrator. The file includes information about the Client VPN endpoint and the certificates required to establish a VPN connection. You load this file into your chosen VPN client application.

Monitoring

- End-user usage reporting is possible through Amazon CloudWatch Logs.
- You can use a client connect handler to do basic posture assessment with Lambda because Client VPN does not support native posture assessment.
- Client VPN publishes metrics to CloudWatch for your Client VPN endpoints. Metrics are published to CloudWatch every five minutes.

Limitations

- Client VPN supports IPv4 traffic only. IPv6 is not supported.
- Security Assertion Markup Language (SAML) 2.0-based federated authentication only works with an AWS provided client v1.2.0 or later.
- Client CIDR ranges must have a block size of at least /22 and must not be greater than /12.
- A Client VPN endpoint does not support subnet associations in a dedicated tenancy VPC.
- Client VPN is not compliant with Federal Information Processing Standards (FIPS).
- Client CIDR ranges cannot overlap with the local CIDR of the VPC in which the associated subnet is located.
- The client CIDR range cannot be changed after you create the Client VPN endpoint.
- The subnets associated with a Client VPN endpoint must be in the same VPC.
- You cannot associate multiple subnets from the same Availability Zone with a Client VPN endpoint.

Pricing

In Client VPN, you are charged for the number of active client connections per hour and the number of subnets associated to Client VPN per hour.

You start by creating a Client VPN endpoint and associating subnets to that endpoint. You can associate multiple subnets from within a VPC to a Client VPN endpoint if they are all part of the same AWS account. Each subnet associated with the Client VPN endpoint must belong to a different Availability Zone and VPC account owner will be billed.

AWS Cloud WAN

If you need to connect cloud routing and software-defined wide area networks (SD-WAN), you can use AWS Cloud WAN to provide a central dashboard for making connections between your offices, data centers, and Amazon VPCs.

Note

AWS Site-to-Site VPN and Direct Connect are both services that let you connect an on-premises network to instances in a VPC.
Global Accelerator, AWS PrivateLink, and AWS Transit Gateway do not, by themselves, provide a method for directly connecting an on-premises environment to the AWS Cloud. They can be used in concert with AWS Site-to-Site VPN and Direct Connect in certain configurations. But as a standalone service, they do not serve the purpose that the two applicable services do.

Application Networking

AWS App Mesh

- App Mesh is a service mesh provided by AWS, essentially acting as a dedicated infrastructure layer for handling service-to-service communication in your microservices applications.
- It simplifies and standardizes how your services communicate, regardless of their location or deployment platform (e.g., EC2, EKS, Fargate).

Amazon API Gateway

An Amazon API Gateway is a collection of resources and methods that are integrated with back-end HTTP endpoints, Lambda functions, or other AWS services. It is a fully managed service that makes it easy for developers to publish, maintain, monitor, and secure APIs at any scale. API calls include traffic management, authorization, access control and monitoring, and API version management. Together with Lambda, API Gateway forms the app-facing part of the AWS serverless infrastructure.

- Back-end services include Amazon EC2, AWS Lambda, or any web application (public or private endpoints).
- API gateway can be used to enable requests from domains other than the APIs domain.

- It allows the sharing of resources between different domains.
- The method (GET, PUT, POST, etc.) for which you will enable CORS must be available in the API Gateway API before you enable CORS.
- A regional API endpoint is intended for clients from the same region. A regional API reduces connection overhead.
- If you deploy a regional API in multiple regions, it can have the same custom domain name in all regions.
- You can use custom domains together with Amazon Route 53 to perform tasks such as latency-based routing.
- Regional and Private API endpoints pass all header names through as-is.
- API Gateway can scale to any level of traffic received by an API
- An origin is the origin of the files that the CDN will distribute. Origins can be either an S3 bucket, an EC2 instance, and Elastic Load Balancer, or Route 53. They can also be external (non-AWS).
- For RTMP CloudFront distributions files must be stored in an S3 bucket. An RTMP distribution is a method of streaming media using Adobe Flash.

The two API types supported by Amazon API Gateway are:

- HTTP API
- WebSocket API
- While HTTPS is the standard protocol for secure communication over the internet, Amazon API Gateway doesn't offer a distinct "HTTPS API" type. Instead, it ensures HTTPS encryption for both of its supported API types:

CloudFront and Edge Optimized

- CloudFront is used as the public endpoint for API Gateway.
- CloudFront also enables the use of custom domains and SNI (Server Name Identification).
- CloudFront sorts HTTP cookies in natural order by cookie name before forwarding the request to your origin.
- The API endpoint type can be edge-optimized. It can be either regional or private, depending on where most of your API traffic originates.
    - An edge-optimized API endpoint is best for geographically distributed clients. API requests are routed to the nearest CloudFront Point of Presence (POP). This is the default endpoint type for API Gateway REST APIs.
    - An edge-optimized endpoint with CloudFront is the default endpoint with API Gateway, there's no need to configure this.

Throttling

Per-client throttling: Per-client throttling limits are applied to clients that use API keys associated with your usage policy as a client identifier. This can be applied to the single customer that is issuing excessive API requests. This is the best option that will ensure that only one customer is affected.

Server-side throttling:
Server-side throttling limits are applied across all clients.

Per-method throttling:
Per-method throttling limits apply to all customers using the same method.

Account-level throttling:
Account-level throttling limit applies to steady-state request rate and burst limits for the account. This does not apply to individual customers.

When request submissions exceed the steady-state request rate and burst limits, API Gateway fails the limit-exceeding requests and returns 429 Too Many Requests error responses to the client.

Caching

You can enable API caching in Amazon API Gateway to cache your endpoint's responses. With caching, you can reduce the number of calls made to your endpoint and also improve the latency of requests to your API.

When you enable caching for a stage, API Gateway caches responses from your endpoint for a specified time-to-live (TTL) period, in seconds. API Gateway then responds to the request by looking up the endpoint response from the cache instead of making a request to your endpoint. An API cache is not enabled for a method, it is enabled for a stage. Cache reduces the load on the back-end. The default TTL value for API caching is 300 seconds. The maximum TTL value is 3600 seconds. TTL=0 means caching is disabled.

Security

- API gateway supports API keys and usage plans for user identification, throttling, or quota management.
- An API can present a certificate to be authenticated by the back-end. By default, API Gateway assigns an internal domain that automatically uses the API Gateway certificates. When configuring your APIs to run under a custom domain name, you can provide your own certificate.
- If CORS is not enabled and an API resource receives requests from another domain, the request will be blocked.
- CORS can be enabled on the APIs resources using the selected methods under the API Gateway.

Monitoring

- The Amazon API Gateway logs (near real-time) back-end performance metrics such as API calls, latency, and error rates to CloudWatch. You can monitor it through the API Gateway dashboard (REST API). The dashboard allows you to monitor calls to the

services visually. API Gateway also meters utilization by third-party developers, and the data is available in the API Gateway console and through APIs.
- Amazon API Gateway is integrated with AWS CloudTrail to give a full, auditable history of the changes to your REST APIs. All API calls made to the Amazon API Gateway APIs to create, modify, delete, or deploy REST APIs are logged to CloudTrail.

Pricing

- With Amazon API Gateway, you only pay when your APIs are in use. There is no minimum fee or upfront commitments.
- You pay only for the API calls you receive and the amount of data transferred out. There are no data transfer out charges for private APIs (however, AWS PrivateLink charges apply when using private APIs in Amazon API Gateway).
- Amazon API Gateway also provides optional data caching charged at an hourly rate that varies based on the cache size that you select.

Limitations

- You cannot add an API gateway to the subnet that the EC2 instances are in; it is a public service with a public endpoint.
- You cannot use Amazon ElastiCache to cache API requests.

AWS Cloud Map

- Service Discovery: Facilitates the discovery and connection of services within a dynamic infrastructure.
- Key Features:
    - Registers service instances with their metadata (e.g., IP addresses, ports, health status).
    - Provides a lookup API for clients to discover available service instances.
    - Integrates with load balancers and service meshes for automatic service discovery.
- Use Cases:
    - Building microservices architectures
    - Deploying containerized applications
    - Managing dynamic infrastructure with frequent service updates
    - Enabling service-to-service communication without hardcoding endpoint addresses

Edge Networking

Amazon Cloudfront

Copies of objects can be cached and distributed at 14 edge locations using Amazon Cloudfront service. CloudFront is a good choice for distributing frequently accessed static content that benefits from edge delivery like popular website images, videos, media files, or software downloads. It can be used for dynamic, static, streaming, and interactive content. It distributes content with low latency and high data transfer rates by serving requests using a network of edge locations worldwide.

Uses an origin group in which you designate a primary origin for CloudFront plus a second origin that CloudFront automatically switches to when the primary origin returns specific HTTP status code failure responses.

It can also be used for uploading content from around the globe. CloudFront can be used to offload requests from the backend. Using Amazon CloudFront as the front-end provides the option to specify a custom message instead of the default message. To specify the file that you want to return and the errors for which the file should be returned, you need to update your CloudFront distribution to specify those values.

Amazon CloudFront can be used to stream video to users across the globe using a wide variety of protocols that are layered on top of HTTP. This can include both on-demand video as well as real-time streaming video.

Object invalidation

- You can remove an object from the cache by invalidating the object. Invalidation can be used to immediately revoke cached objects — chargeable. Deletions propagate.
- You cannot cancel an invalidation after submission.
- You cannot invalidate media files in the Microsoft Smooth Streaming format when you have enabled Smooth Streaming for the corresponding cache behavior.
- Objects are cached for the TTL (always recorded in seconds, default is 24 hours, default max is 1 year). Consider how often your files change when setting the TTL.
- It only caches for GET requests (not PUT, POST, PATCH, DELETE). Dynamic content is also cached.

Security

CloudFront is PCI DSS compliant, but it is not recommended to cache credit card information at edge locations. It is also HIPAA compliant as a HIPAA eligible service.

- CloudFront distributes traffic across multiple edge locations and filters requests to ensure that only valid HTTP(S) requests will be forwarded to back-end hosts. CloudFront also supports geo-blocking, which you can use to prevent requests from particular geographic locations from being served.

    - Blacklists and whitelists can be used for geography. You can only use one at a time. There are two options available for geo-restriction (geo-blocking):
        - Use the CloudFront geo-restriction feature (use for restricting access to all files in a distribution and at the country level).
        - Use a 3rd party geo-location service (use it for restricting access to a subset of the files in a distribution and for finer granularity at the country level).
- You can restrict access to content using the following methods:
    - Restrict access to content by using signed cookies or signed URLs.
    - Restrict access to objects in your S3 bucket.
- A special type of user called an Origin Access Identity (OAI) can be used to restrict access to content in an Amazon S3 bucket (not on EC2). By using an OAI, you can restrict users so they are unable to access the content directly using the S3 URL; they must instead connect via CloudFront.
    - You cannot use CloudFront and an OAI when S3 bucket is configured as a website endpoint.
    - Users can not login with OAI.
- If you want to restrict the ability of users to circumvent CloudFront and access the content directly through the ELB (in front of an EC2 instance) then, The only way to get this working is by using a VPC Security Group for the ELB that is configured to allow only the internal service IP ranges associated with CloudFront. As these are updated from time to time, you can use AWS Lambda to automatically update the addresses. This is done using a trigger, which is triggered when AWS issues an SNS topic update when the addresses are changed.
    - Signed cookies and URLs are used to limit access to files, but this does not stop people from circumventing CloudFront and accessing the ELB directly.
- CloudFront does have DDoS prevention features.

Encryption

Amazon Cloudfront can be used to enforce secure end-to-end connections, using HTTPS, to origin servers. An additional security layer gets added via Field-level encryption. This lets you protect specific data throughout system processing so that only certain applications can see it. Field-level encryption allows you to enable your users to upload sensitive data to your web servers securely. This sensitive data is encrypted at the edge and remains encrypted throughout your entire application stack, ensuring that only the applications that require this data can do so.

Edge Location

An edge location is a location where content is cached (separate to AWS Regions/AZs). Requests are automatically routed to the nearest edge location. They are not tied to Availability Zones or regions.

Regional Edge Caches are located between origin web servers and global edge locations. Regional Edge Caches have a larger cache-width than any individual edge location, so your objects remain in cache longer at these locations. They aim to get content closer to users.

- Proxy methods PUT/POST/PATCH/OPTIONS/DELETE go directly to the origin from the edge locations and do not proxy through Regional Edge caches.
- Dynamic content goes straight to the origin and does not flow through Regional Edge caches.
- Edge locations are not just read-only; you can write to them too.

Charges

- There is an option for reserved capacity over 12 months or longer (starts at 10 TB of data transfer in a single region).
- You pay for:
    - Data transfer out to the Internet.
    - Data transfer out to Origin.
    - Number of HTTP/HTTPS requests.
    - Invalidation requests.
    - Dedicated IP custom SSL.
    - Field-level encryption requests.
- You do not pay for:
    - Data transfer between AWS regions and CloudFront.
    - Regional edge cache.
    - AWS ACM SSL/TLS certificates.
    - Shared CloudFront certificates.

CloudFront vs ElastiCache with Redis

CloudFront is very good for getting media content closer to users from S3. However, when requirements are to improve performance in case of high frequency reads and consistent throughput then ElastiCache with Redis is preferred. It helps in bringing down latency to sub-millisecond.

Limitations

- You cannot use CloudFront to pull data directly from an EBS volume.
- CloudFront does not sit within a subnet so network ACL's do not apply to it.
- Amazon CloudFront can not be configured with "a pair of static IP addresses".

Route 53

Highly available and scalable DNS (Domain Name System) web service. It routes end users to internet applications by translating readable web addresses to the numeric IP addresses like 192.0.2.1.

Route 53 offers the following functions:

- Domain name registry

- DNS resolution
- Health checking of resources

Route 53 can perform any combination of these functions. It provides a worldwide distributed DNS service as it is located alongside all edge locations. It supports health checks that verify if internet-connected resources are reachable, available, and functional.

- It is possible to have the domain registered in one AWS account and the hosted zone in another AWS account.
- Route 53 can be used to route Internet traffic for domains registered with another domain registrar (any domain).
- You can transfer a domain to another account in AWS; however, it does not migrate the hosted zone by default (optional).
- You can transfer a domain from Route 53 to another registrar by contacting AWS support.
- Private DNS is a Route 53 feature that lets you have authoritative DNS within your VPCs without exposing your DNS records (including the resource's name and its IP address(es) to the internet).
- To use Route 53 for an existing domain, the architect needs to change the NS records to point to the Amazon Route 53 name servers.
- Route 53 supports wildcard entries for all record types, except NS records.
- It can be used to load balance; however, it does not have the ability to route based on information in the incoming request path.
- DNS is configured at the VPC level. Amazon provides a DNS server to be used by default; however, you can launch an instance and run your own DNS and then reconfigure the VPC to use your DNS server instead.

Hosted zones

A hosted zone is a collection of records for a specified domain. It is analogous to a traditional DNS zone file; it represents a collection of records that can be managed together.
There are two types of zones:

- Public host zone — determines how traffic is routed on the Internet.
- With public hosted zones, Route 53 can automatically register EC2 instances based on their tags. This eliminates manual configuration and ensures DNS records are always up-to-date.
- You cannot automatically register EC2 instances with privately hosted zones (it would need to be scripted - AWS CLI or SDK) as private hosted zones are not publicly accessible.
    - PHZs are isolated from the internet, preventing Route 53 from directly accessing instance information.
    - This measure protects the confidentiality of private resources and prevents unauthorized access.
- You can extend an on-premises DNS to VPC; however, you cannot extend Route 53 to on-premises instances.

- Zone file can be imported from existing DNS

Private Hosted Zone

Private hosted zone for VPC — determines how traffic is routed within VPC (resources are not accessible outside the VPC). Single private hosted zone can be associated with multiple VPCs. Multiple private hosted zones associated with the same VPC.

For private hosted zones you need to set following VPC settings to true:
- enableDnsHostnames
- enableDnsSupport

Only Amazon Route53 resolver can resolve records in private hosted zone.
Any resource within VPC can use resolver for public and private domains.
It only supports simple, failover, multivalue answer, weighted routing policies

Split View DNS

Maintain internal and external versions of the same website or application.
Configure public and private hosted zones to return different internal and external IP addresses for the same domain name.
Create public and private hosted zones with the same domain name and create subdomains.

Record Types

Alias Record

- Function: Redirects a domain name or subdomain to another domain name.
- Example: `blog.example.com` CNAME record points to example.com.
- Benefits:
    - Easier to manage if the target domain changes as you only need to update the alias record.
    - Can point to another domain, enabling flexible configurations for subdomains or separate services.
    - Useful for migrating websites or consolidating multiple domains under one umbrella.
- Drawbacks:
    - May not work with all devices or older browsers that don't support CNAME records.
    - Introduces an extra layer of redirection, potentially impacting website loading speed.
    - Less control over the final destination as it relies on the target domain's DNS configuration.

It is a general term for any DNS record that points to another domain name instead of an IP address. It's an umbrella term encompassing different record types like CNAME, ALIAS (specific to AWS Route 53), or DNAME.

CNAME record

A CNAME record in Route 53 are a specific type of Alias record that maps one domain name to another domain name. This is useful for creating aliases or redirects. For example, you could create a CNAME record that maps example.com to www.example.com. This would tell DNS servers to return the IP address of www.example.com when clients resolve example.com.

It cannot be created for zone apex (e-g in hosted zone example.com, CNAME records with example.com cannot be created, The zone apex of the hosted zone is the name of the hosted zone.)

Custom TTL value can be configured.

- Use CNAME if:
    - You want a simple and widely compatible redirect within the same domain (e.g., subdomain redirects).
    - You already have an A record for the target domain.
    - Compatibility with older devices is critical.
- Use alias (specific type depending on your service) if:
    - You need more flexibility and want to redirect to any domain, regardless of an A record.
    - You're using a service like AWS Route 53 that offers specific alias record types.
    - Performance impact of an extra layer is minimal or not a concern.

Standard Record

- Function: Directly maps a domain name or subdomain to an IP address.
- Example: `example.com` A record points to 192.168.1.1.
- Benefits:
    - Simple and straightforward configuration.
    - Works with all types of internet connections and devices.
    - Provides complete control over the destination IP address.
- Drawbacks:
    - Requires managing and updating IP addresses manually if they change.
    - Cannot point to another domain name (only IP addresses).
    - May not be ideal for highly dynamic environments with frequent hostname changes.

PTR record

PTR records are reverse lookup records where you use the IP to find the DNS name.

Routing policies

Routing policies determine how Route 53 responds to queries.

Simple

Simple DNS response, providing the IP address associated with a name.

Failover

If the primary is down (based on health checks), routes to secondary destinations. The failover routing policy is used for active/passive configurations.

Geolocation

Uses the geographic location you're in (e-g Europe) to route you to the closest region. It contains users within a particular geography and offers them a customized version of the workload based on their specific needs.

- Geolocation can be used for localizing content and presenting some or all of your website in your users' language.
- You can create a default record for IP addresses that do not map to a geographic location.

Geo-proximity

Routes you to the closest region within a geographic area.

Latency

Directs you based on the lowest latency route to resources.

Multivalue answer

Returns several IP addresses and functions as a basic load balancer.

Weighted

Use the relative weights assigned to resources to determine which one to route. Weights can be specified per IP address. You create records with the same name and type and assign each record a relative weight, which is a numerical value that favors one IP over another (values must total 100). To stop sending traffic to a resource, you can change the weight to 0. Weights can be updated anytime.

One possible use case is that you have two versions of your applications deployed in separate EC2 instances. You want to expose the latest version to a limited number of users to test it. You will set relatively less weight for the IP address of the EC2 instance with the latest application version, to make sure only a small percentage of traffic gets routed to this instance.

Route53 Traffic Flow

Route 53 Traffic Flow provides Global Traffic Management (GTM) services. Traffic Flow policies allow you to create routing configurations for resources using routing types such as failover and geolocation.
Amazon Route 53 Traffic Flow also includes a versioning feature that allows you to:

- maintain a history of changes to your routing policies;
- easily roll back to a previous policy version using the console or API.

Route 53 Resolver

Route 53 Resolver is a set of features that enable bi-directional querying between on-premises and AWS over private connections. It is used to enable DNS resolution for hybrid clouds.

- Inbound query capability (to AWS) is provided by Route 53 Resolver Endpoints, allowing DNS queries that originate on-premises to resolve AWS hosted domains.
- Connectivity needs to be established between your on-premises DNS infrastructure and AWS through a Direct Connect (DX) or a Virtual Private Network (VPN).
- Outbound DNS queries are enabled through the use of Conditional Forwarding Rules.
- Domains hosted within your on-premises DNS infrastructure can be configured as forwarding rules in Route 53 Resolver.
- Rules will trigger when a query is made to one of those domains and will attempt to forward DNS requests to the DNS servers that were configured along with the rules.
- Like inbound queries, this requires a private connection over DX or VPN.

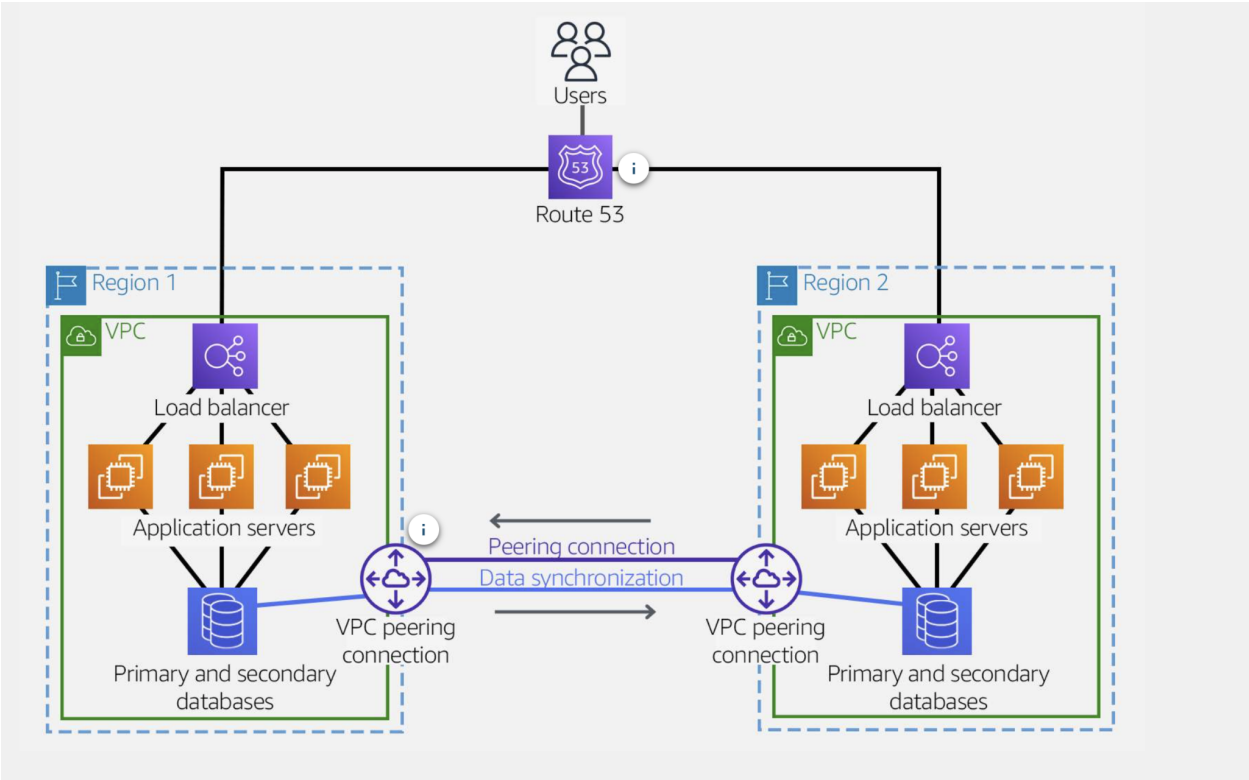
Pricing

- Hosted Zones:
    - First 25 hosted zones are free.
    - Additional hosted zones are charged at $0.50 per zone per month.
- DNS Queries:
    - First 1 million queries per month are free.
    - Additional queries are charged at $0.40 per million queries.
- Health Checks:
    - Simple health checks are $0.50 per health check per month.
    - Advanced health checks are $0.50 per endpoint per month.
- Traffic Flow:
    - First 50 rules are free.
    - Additional rules are $0.50 per rule per month.

Additional Considerations:

- Geolocation Routing:
    - No additional charges for geolocation routing.

- Latency-Based Routing:
    - No additional charges for latency-based routing.
- Failover Routing:
    - No additional charges for failover routing.
- Domain Registration:
    - Route 53 charges standard domain registration fees, which vary by domain extension.



In the above diagram, Amazon Route 53 routes end users to the Region with the least latency. The databases in each Region are synchronized using a VPC peering connection between the two Regions. If a user reconnects after their session is interrupted, this design ensures they can resume their session regardless of which Region they reconnect to. For example, the user begins their session in Region 1 and is disconnected. When they reconnect, they are routed to Region 2. They can resume their session because the databases are synchronized through the VPC peering connection.

AWS Global Accelerator

It is a service that improves applications availability and performance for local and global users. It provides static IP address that act as a fixed entry point to application endpoints in a single or multiple AWS regions. Endpoints can be Network Load Balancers, Application Load Balancers,

EC2 instances, or Elastic IP addresses that are located in one or multiple AWS Regions. In such scenarios CloudFront cannot be used as it cannot expose static public IP addresses.

It uses the AWS global network to optimize the path from users to applications, improving TCP and UDP traffic performance. AWS Global Accelerator continually monitors application endpoints' health and will detect an unhealthy endpoint and redirect traffic to healthy endpoints in less than 1 minute.

- By using the static IP addresses, you don't need to make any client-facing changes or update DNS records as you modify or replace endpoints. The addresses are assigned to your accelerator for as long as it exists, even when you disable the accelerator and it no longer accepts or routes traffic.
- Can assign target weight within a region to control routing and also "dial" up or down traffic to a region.
- AWS Global Accelerator enables you to build applications that require maintaining state.
- For stateful applications where you need to route users to the same endpoint consistently, you can choose to direct all requests from a user to the same endpoint, regardless of the port and protocol.
- By default, AWS Global Accelerator is protected by AWS Shield Standard, which minimizes application downtime and latency from denial of service attacks by using always-on network flow monitoring and automated in-line mitigation.
- It uses 2 static anycast IP addresses.
    - Seamless failover is ensured as AWS Global Accelerator uses the anycast IP address, which means the IP does not change when failing over between regions, so there are no issues with client caches having incorrect entries that need to expire.
- By default, Global Accelerator provides you with two static IP addresses that you associate with your accelerator. (Instead of using the IP addresses that Global Accelerator provides, you can configure these entry points to be IPv4 addresses from your own IP address ranges that you bring to Global Accelerator.)

You can migrate up to two /24 IPv4 address ranges and choose which /32 IP addresses to use when you create your accelerator.

AWS Global Accelerator is an expensive option than CloudFront for getting content closer to users.

Network Security

AWS Shield

You get this as a default for your load balancers, cloud front, as well as Route 53. This is basically a DDoS mitigation service that prevents DDoS Attacks. AWS Shield Standard defends

against most common, frequently occurring network and transport layer DDoS attacks that target web sites or applications. It provides protection at Layers 3 (network/IP), 4 (transport TCP/UDP), and 7 (application/HTTPS/etc.).

Advance Shield

It is an AWS team that is in standby mode in the case of a DDOS attack. If you have advanced shield protection, then AWS will not charge you for any auto-scaling or added utilization of the AWS services during the attack. AWS Shield Advanced provides additional detection and mitigation against large and sophisticated DDoS attacks, near real-time visibility into attacks, and integration with AWS WAF.

AWS Shield Advanced includes DDoS cost protection, a safeguard from scaling charges as a result of a DDoS attack that causes usage spikes on protected Amazon EC2, Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator, or Amazon Route 53. If any of the AWS Shield Advanced protected resources scale-up in response to a DDoS attack, you can request credits via the regular AWS Support channel.

WAF – Web Application Firewall

WAF sits in front of your web server and it mitigates against injection, cross-scripting. WAF primarily protects your application layer from any malicious attacks.

AWS WAF is a web application firewall that lets you monitor HTTP and HTTPS requests that are forwarded to CloudFront. It also lets you control access to your content.
With AWS WAF, you can shield access to content based on conditions in a web access control list (web ACL) such as:

- Origin IP address.
- Values in query strings.

Web ACL may use reusable rule groups.

AWS WAF helps protect web applications from attacks by allowing you to configure rules that allow, block, or monitor (count) web requests based on conditions that you define. These conditions include IP addresses, HTTP headers, HTTP body, URI strings, SQL injection, and cross-site scripting.

New rules can be deployed within minutes, letting you respond quickly to changing traffic patterns.

When AWS services receive requests for websites, the requests are forwarded to AWS WAF for inspection against defined rules. Once a request meets a condition defined in the rules, AWS WAF instructs the underlying service to either block or allow the request based on the action you define.

WAF can handle errors e-g HTTP 403 error (forbidden). Default behavior can also be configured to handle requests that don't match any rules. Default can be allowed or denied.

WAF with CloudFront and ALB

AWS WAF is tightly integrated with Amazon CloudFront and the Application Load Balancer (ALB) services. When you use AWS WAF on Amazon CloudFront, rules run in all AWS Edge Locations located around the world close to end-users. This means security doesn’t come at the expense of performance. Blocked requests are stopped before they reach your web servers.

When you use AWS WAF on an Application Load Balancer, your rules run in the region and can be used to protect internet-facing and internal load balancers.

AWS WAF allows you to create a centralized set of rules that you can deploy across multiple websites. This means that you can create a single set of rules that you can reuse across applications rather than recreating that rule on every application you want to protect in an environment with many websites and web applications.

AWS WAF supports all IPv4 and IPv6 address ranges. An IP set can hold up to 10,000 IP addresses or IP address ranges to check.

Attackers sometimes insert scripts into web requests in an effort to exploit vulnerabilities in web applications. You can create one or more cross-site scripting (XSS attacks) match conditions to identify the parts of web requests (such as the URI or the query string) that you want AWS WAF to inspect for possible malicious scripts.

Monitoring

AWS WAF provides real-time metrics and captures raw requests that include details about IP addresses, geo-locations, URIs, User-Agent, and Referrers.

AWS WAF is fully integrated with Amazon CloudWatch, making it easy to set up custom alarms when thresholds are exceeded or particular attacks occur. This information provides valuable intelligence that can be used to create new rules to better protect applications.

Limitations

- You cannot use WAF with a Classic Load Balancer and Network Load Balancer.
- You cannot use AWS WAF to protect EFS data using users and groups.
- AWS WAF is a web application firewall and does not work at the instance level.

Pricing

With AWS WAF, you pay only for what you use. AWS WAF pricing is based on how many rules you deploy and how many web requests your web application receives. There are no upfront commitments.

AWS Network Firewall

It deploys network security access for your Amazon VPC.

AWS Firewall Manager

It helps to centrally configure and manage your firewall rules.
Firewall Manager monitors for new resources created to ensure they comply with a mandatory set of security policies.
You can enable AWS WAF rules, AWS Shield Advanced protections, Amazon VPC security groups and AWS Network Firewalls.

Monitoring and troubleshooting Services

Amazon CloudWatch

Amazon Cloudwatch service is used for monitoring resource utilization, disk read and writes, network traffic, memory utilization, etc. Each data point that a resource creates is a metric. Metrics that are collected and analyzed over time become statistics (average CPU utilization). Different resources create different metrics. To obtain more comprehensive insights into your application's performance, it is essential to create custom metrics (e-g page views) for detailed logs. While EC2 instance metrics provide general information about the health of the EC2 environment, they do not offer specific insights into your application. We can also set a dashboard for interactive logs visualization in the console. Dashboards can pull metrics of resources spread across multiple regions. IAM is used to manage access to the CloudWatch dashboard.

CloudWatch metrics are collected in a time ordered structure

Many AWS services send metrics to CloudWatch for free at a rate of 1 data point per metric per 5-minute interval. For many applications it is enough, but the rate of transfer can be increased for a fee.

- CloudWatch Logs can be used for real-time application and system monitoring as well as long-term log retention.
- CloudWatch Logs can track the number of errors that occur in your application logs and send you a notification whenever the rate of errors exceeds a threshold you specify.
- CloudTrail logs can be sent to CloudWatch Logs for real-time monitoring.
- CloudWatch Logs’ metric filters can evaluate CloudTrail logs for specific terms, phrases, or values.
- We can push on-premise logs into the cloud.

CloudWatch Alarm

Cloudwatch Alarm is created for triggering alarms. An alarm can be in three states i) OK, ii) Insufficient Data, iii) In alarm. Alarm transition from one state to another based on thresholds

and conditions you defined for the desired resource metric. Alarm is sent in an In alarm state. When creating an alarm, we also select an SNS topic to send alarms to subscribers of the topic.

You cannot associate an SQS queue with a CloudWatch alarm.

First, choose the metric to monitor.
Second, choose the metric alarm state (OK, ALARM, INSUFFICIENT_DATA).
Third and fourth, specify three settings to enable CloudWatch to evaluate when to change the alarm state:

- Period is the length of time to evaluate the metric or expression to create each individual data point for an alarm. It is expressed in seconds. If you choose one minute as the period, the alarm evaluates the metric once per minute.
- Evaluation Periods is the number of the most recent periods, or data points, to evaluate when determining alarm state.
- Datapoints to Alarm is the number of data points within the Evaluation Periods that must be breaching to cause the alarm to go to the ALARM state. The breaching data points don't have to be consecutive, they just must all be within the last number of data points equal to Evaluation Period.

Fifth, choose when to initiate the alarm and specify what actions an alarm takes when the state changes.

It watches metrics to alert and remediate with the proper configurations.

CloudWatch Events

Amazon CloudWatch Events delivers a near real-time stream of system events that describe changes in Amazon Web Services (AWS) resources. Although you can generate custom application-level events and publish them to CloudWatch Events, this is not the best tool for monitoring application logs.

Information that is not available in CloudWatch

- Deleted DynamoDB table items will not be recorded in CloudWatch Logs.
- There is no SwapUsage metric in CloudWatch. All memory metrics must be custom metrics.
- You do not create custom metrics in the EC2 instances console . You must configure the instances to send the metric information to CloudWatch.

If you use Amazon VPC to host your AWS resources, you can establish a private connection between your VPC and CloudWatch Logs. You can use this connection to send logs to CloudWatch Logs without sending them through the internet. To connect your Amazon VPC to CloudWatch Logs, you define an interface VPC endpoint for CloudWatch Logs. The endpoint provides reliable, scalable connectivity to CloudWatch Logs without requiring an internet gateway, network address translation (NAT) instance, or VPN connection.

Exporting log data to Amazon S3 buckets that are encrypted by AWS Key Management Service (AWS KMS) is not supported.

Data Retention

CloudWatch retains metric data as follows:

- Data points with a period of less than 60 seconds are available for 3 hours. These data points are high-resolution custom metrics.
- Data points with a period of 60 seconds (1 minute) are available for 15 days
- Data points with a period of 300 seconds (5 minutes) are available for 63 days
- Data points with a period of 3,600 seconds (1 hour) are available for 455 days (15 months)

Note: The longer CloudWatch data is stored, the less information is available, due to aggregation of those data points.

You can always retrieve metrics data for in CloudWatch, even if a resource is deleted. However, the CloudWatch console limits the search of metrics to two weeks after a metric is last ingested to ensure that the most up to date instances are shown in your namespace.

Pricing

- If you have a single CloudWatch alarm with multiple metrics, you are charged for each metric associated with a CloudWatch alarm.

Charges are also incurred:

- When you exceed three dashboards with up to 50 metrics.
- By ingesting and storing logs, as well as the amount of ingested logs scanned for each CloudWatch Insights query.
- Based on the number of custom events.
- Charges are also incurred when you monitor more than 10 custom metrics. Custom metrics can be metrics you create and also metrics from tools such as the CloudWatch agent. Metrics collected by the CloudWatch agent are billed as custom metrics.

VPC Flow logs

Flow logs capture information about the IP traffic going to and from network interfaces in a VPC. Flow log data is stored using Amazon CloudWatch Logs. Because CloudWatch is used to store the flow logs, using them incurs cost. Logs can also be exported to S3 bucket.

You can use VPC Flow Logs to capture detailed information about the traffic going to and from your Elastic Load Balancer. Create a flow log for each network interface for your load balancer. There is one network interface per load balancer subnet. You can also use VPC flow log for the subnets in which the ELB is but a more secure option is to use the EL network interfaces.

Not all traffic is monitored, e.g., the following traffic is excluded: Traffic that goes to Route53

- Traffic generated for Windows license activation
- Traffic to and from 169.254.169.254 (instance metadata)
- Traffic to and from 169.254.169.123 for the Amazon Time Sync Service
- DHCP (Dynamic Host Configuration Protocol) traffic
    - DHCP provides a standard for providing configuration information to host on a TCP/IP network. The options field of a DHCP message contains configuration parameters (e-g Domain Name, Domain Name Server, etc). They are used for providing dynamic addresses where required within VPC. You can have multiple DHCP option sets (e-g one per subnet).
- Traffic to the reserved IP address for the default VPC router.

Flow log data does not affect network throughput or latency because it's collected outside of the path of your network traffic.

They can monitor all activity at three different levels:

- VPC level monitors all the activity of your operations within your cloud environment.
- Subnet level monitors all activity for a specific subnet.
- Network interface level monitors specific interfaces on Amazon Elastic Compute Cloud (Amazon EC2) instances and capture flow logs from that interface.

Flow logs do not capture real-time log streams for your network interfaces.

Limitations

- Changes cannot be made to the configuration of a flow log or the format of a flow log record after they have been created. If the flow log you have created is not gathering the data you expected or if the nature of what you need to gather changes, you have to delete the existing flow log and create a new one. For example, you cannot change the IAM role associated with your flow log after creation. To associate a different IAM role, you would have to recreate the flow log.
- Flow logs can only be configured for VPC peering connections deployed by your account. VPC peering connections deployed by another account cannot be monitored using VPC flow logs even if they have been authorized to link to VPCs within your account.
- Network interfaces for EC2-Classic instances are not supported. This includes instances linked to your VPC through ClassicLink.

Traffic Mirroring

- Copies each IP packet, sent or received, by an elastic network interface (ENI) on an Amazon EC2 instance to a traffic mirror target. A traffic mirror target is an out-of-band security appliance, monitoring appliance, or Network Load balancer.

- Captures and inspects network traffic at scale and provides data for troubleshooting and intrusion detection, along with other types of threat monitoring and content inspection.
- Use VPC Traffic Mirroring in a multi-account AWS environment, capturing traffic from VPCs spread across many AWS accounts and then routing it to a central VPC for inspection.

The traffic mirror source is the network interface of an Amazon EC2 instance where AWS copies the network traffic from. VPC Traffic Mirroring supports the use of Elastic Network Interfaces (ENIs) as mirror sources.

A traffic mirror target is the destination for mirrored traffic. The traffic mirror target can be:

- A network interface or a network load balancer.
- Used in more than one traffic mirror session.

You can:

- Stream replicated traffic to any network packet collector or analytics tool, without having to install vendor-specific agents.
- Use open-source tools or choose a monitoring solution available on AWS Marketplace.

A traffic mirror filter is a set of rules that defines the traffic that is copied in a traffic mirror session. By default, no traffic is mirrored. To mirror traffic, add traffic mirror rules to the filter. Traffic mirror filter rules define what traffic gets mirrored.

- Rules are numbered and processed in order within the scope of a particular mirror session.
- The filter can specify a protocol, ranges for the source and destination ports, and CIDR blocks for the source and destination.

A traffic mirror session establishes a relationship between a traffic mirror source and a traffic mirror target that makes use of a traffic mirror filter.

A given packet is only mirrored one time. However, you can use multiple traffic mirror sessions on the same source. This is useful if you want to send a subset of the mirrored traffic from a traffic mirror source to different tools.

Mirrored network traffic is subject to connectivity considerations. The source and target can share an Amazon VPC, or exist in different ones with an intra-Region VPC peering connection or a transit gateway. The traffic target does not have to share an AWS account with the source. Thus, users must know the AWS rules that govern routing before they implement VPC Traffic Mirroring.

Traffic Mirroring and VPC Flow Logs

VPC Traffic Mirroring and VPC Flow Logs can be used together for a deeper understanding of your Amazon VPC traffic.

VPC Flow Logs are easier to implement and use, but they provide little context. VPC Flow Logs help to troubleshoot connectivity and security issues, and collects, stores, and analyzes network flow logs about the following:

- Allowed and denied traffic
- Source and destination IP addresses
- Ports
- Protocol number
- Packet and byte counts
- Action taken (accept or reject)

Traffic Mirroring provides deeper insight and gives more context into the network traffic to analyze actual traffic content, including payload. All traffic is combined into log files and shows pivot points, which are a technical analysis indicator used to determine the overall trend, such as certificate hashes, protocol version, and authentication status.
For example, if you connect by using SSH into the traffic mirroring target EC2 instance using VPC Flow Logs, you get no actual insight into what is happening. The log will show a TCP-based connection using port 22 and that the SSH session corresponds to flow entries. However, using both VPC Traffic Mirroring and VPC Flow Logs, you can track a full session of your network traffic, compare ports and TCP flags, which provides metadata about the actual traffic as well as combining them into useful logs to search.

The following traffic types cannot be mirrored:

- ARP
- DHCP
- Instance metadata service
- NTP
- Windows activation

VPC Reachability Analyzer

The VPC Reachability Analyzer is a network diagnostics tool that troubleshoots reachability between two endpoints in an Amazon VPC, or within multiple Amazon VPCs.

It can only be used for endpoints within the same Amazon VPC as a:

- Configuration analysis tool that helps to perform connectivity testing between a source resource and a destination resource.
- Network diagnostics tool that troubleshoots reachability between two endpoints.

There is a large array of available connection types and endpoints to test, it is not limited to Amazon EC2 instances. You can run a reachability analysis between:

- VPN gateways
- Network interfaces
- Internet gateways
- VPC endpoints

- VPC peering connections
- Transit gateways

Pricing

With VPC Reachability Analyzer, you are charged per analysis run between a source and a destination. It is best practice to run analysis during networking configuration changes and to troubleshoot connectivity issues that arise.

AWS Transit Gateway Manager

The AWS Transit Gateway Network Manager lets you to centrally manage your networks that are built around transit gateways. You can visualize and monitor your global network across Regions and on-premises locations.

It only monitors packet flows and routing, no troubleshooting functionality.

It does not dynamically discover changes in registered devices configurations, changes in configuration needs to be manually synced up with network manager.

There is no additional cost for using Network Manager. You pay for the network resources you use, like transit gateways, VPNs, and so on

Transit Gateway Manager vs VPC Reachability Analyzer

Purpose:

- VPC Reachability Analyzer: Focuses on identifying and troubleshooting potential connectivity issues within your VPCs and between VPCs connected through Transit Gateways.
- Transit Gateway Network Manager: Centrally manages and monitors your Transit Gateway network, including route tables, attachments, and peering relationships.

Key Features:

- VPC Reachability Analyzer:
    - Simulates traffic to test reachability between endpoints.
    - Generates path visualizations for traffic flow analysis.
    - Pinpoints potential connectivity blocks (e.g., misconfigured security groups, routing tables).
    - Provides insights into network configurations and bottlenecks.
- Transit Gateway Network Manager:
    - Centralized console for managing multiple Transit Gateways.
    - Automates route table management for attached VPCs and VPNs.
    - Enforces network policies for security and compliance.
    - Monitors network health and visualizes traffic flows.

    - Detects and troubleshoots connectivity issues across the Transit Gateway network

Use Cases:

- Reachability Analyzer:
    - Troubleshooting connectivity problems between resources within a VPC or across Transit Gateways.
    - Validating network configuration changes before deployment.
    - Performing network audits and security assessments.
- Transit Gateway Network Manager:
    - Simplifying management of large-scale Transit Gateway networks.
    - Automating route propagation for new VPCs and VPNs.
    - Enforcing security and compliance policies across the network.
    - Monitoring network health and identifying potential issues.

Relationship:

- Reachability Analyzer is often used within Network Manager: Network Manager can leverage Reachability Analyzer to troubleshoot connectivity issues within the Transit Gateway network.
- They complement each other: Network Manager provides centralized management and monitoring, while Reachability Analyzer offers in-depth troubleshooting capabilities.

AWS CloudTrail

It creates a trail or a log of everything any user does inside of AWS. It stores a trail record of a user for one week. Cloudtrail shows what is done, who does it and on what resource, not all of the metadata is available in cloud trail data.

A CloudTrail trail can be created, which delivers log files to an Amazon S3 bucket.

CloudTrail performs auditing of API actions, it does not do packet capturing.

AWS CloudTrail logs API calls made via:

- AWS Management Console.
- AWS SDKs.
- Command-line tools.
- Higher-level AWS services (such as CloudFormation).

CloudTrail is per AWS account. Trails can be enabled per region, or a trail can be applied to all regions.

Trails can be configured to log data events and management events.

- Data events: These events provide insight into the resource operations performed on or within a resource. These are also known as data plane operations. Data events are often high-volume activities. Example data events include:

    - Amazon S3 object-level API activity (for example, GetObject, DeleteObject, and PutObject API operations)
    - AWS Lambda function execution activity (the Invoke API)
- Management events: Management events provide insight into management operations that are performed on resources in your AWS account. These are also known as control plane operations. Management events can also include non-API events that occur in your account. Example management events include:
    - Configuring security (for example, IAM AttachRolePolicy API operations)
    - Registering devices (for example, Amazon EC2 CreateDefaultVpc API opera

CloudTrail log files are encrypted using S3 Server Side Encryption (SSE). You can also enable encryption using SSE KMS for additional security. A single KMS key can be used to encrypt log files for trails applied to all regions.
You can consolidate logs from multiple accounts using an S3 bucket.

- Turn on CloudTrail in the paying account.
- Create a bucket policy that allows cross-account access.
- Turn on CloudTrail in the other accounts and use the bucket in the paying account.

CloudTrail does not log in real time. There is a delay, but you can create a CloudTrail and store that data in Amazon S3 or CloudWatch logs.

CloudTrail for logging access to reports/files in Amazon S3 Bucket. All modifications to log files must be detected.

Use AWS CloudTrail to create a new trail. Configure the trail to log read and write data events on the S3 bucket that houses the reports. Log these events to a new bucket, and enable log file validation.

CloudWatch vs CloudTrail

CloudWatch: Performance Monitoring, Log events across AWS services - think operations, log from multiple accounts, Alarm history for 14 days, logs stored indefinitely. It is used for consolidating system and application-level logs and creating business key performance indicators (KPIs) as custom metrics for your specific needs.

CloudTrail: Log API activity across AWS services, Log from multiple accounts, Logs stored to S3 or CloudWatch indefinitely, no native alarming. It is used for auditing.

AWS Systems Manager (SSM Agent)

Used to roll out patches across your entire fleet of machines. Systems Manager provides a central place to view and manage your AWS resources so that you can have complete visibility and control over your operations. It is a centralized console and toolset for a wide variety of system management tasks. It has been designed for managing a large fleet of systems (tens or hundreds).

- AWS Systems Manager collects information about your instances and the software installed on them, helping you to understand your system configurations and installed applications.
- You can collect data about applications, files, network configurations, Windows services, registries, server roles, updates, and any other system property.
- AWS Systems Manager lets you scan your managed instances for patch compliance and configuration inconsistencies.
- You can collect and aggregate data from multiple AWS accounts and regions and then drill down into specific resources that aren't compliant.
- AWS Systems Manager allows you to safely automate common and repetitive IT operations and management tasks across AWS resources.
- With Systems Manager, you can create JSON documents that specify a specific list of tasks or use community published documents.
- These documents can be executed directly through the AWS Management Console, CLIs, and SDKs, scheduled in a maintenance window, or triggered based on changes to AWS resources through Amazon CloudWatch Events.
- AWS Systems Manager provides you safe and secure remote management of your instances at scale without logging into your servers. This replaces the need for bastion hosts, SSH, or remote PowerShell.
- It provides a simple way of automating common administrative tasks across groups of instances such as registry edits, user management, and software and patch installations.
- Through integration with AWS Identity and Access Management (IAM), you can apply granular permissions to control the actions users can perform on instances.
- All actions taken with Systems Manager are recorded by AWS CloudTrail, allowing you to audit changes throughout your environment.
- AWS Systems Manager lets you schedule windows of time to run administrative and maintenance tasks across your instances.

Amazon Systems Manager Distributor

- Simplifies packaging and distributing software packages (agents, tools, patches, etc.) to managed nodes across your AWS infrastructure.
- Streamlines deployment, updates, and maintenance of software across large-scale fleets of instances, edge devices, and on-premises servers.

SSM Session Manager

- Provides a secure, interactive shell session directly within the browser for managing EC2 instances, on-premises servers, and virtual machines (VMs) in hybrid environments.
- Eliminates the need for bastion hosts or SSH keys, enhancing security and simplifying access.

AWS Systems Manager Maintenance window

- Automate and manage the execution of tasks on managed instances within specified timeframes, ensuring controlled and predictable maintenance activities.
- Streamline patching, configuration updates, software installations, and other routine operations across multiple instances.

SSM Agent compared to CloudWatch Agent

- Use SSM Agent if:
    - You need to remotely manage and configure your instances.
    - You want to automate tasks and configuration changes.
    - You need a central point for managing your fleet of instances.
- Use CloudWatch Agent if:
    - You need to monitor the health and performance of your applications and infrastructure.
    - You want to analyze trends and anomalies in system metrics and logs.
    - You want to set up alarms and notifications for performance issues.

AWS CloudFormation

AWS CloudFormation is a service that gives developers and businesses an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. AWS CloudFormation provides a common language for you to describe and provision all the infrastructure resources in your cloud environment. Think of CloudFormation as deploying infrastructure as code.
Logical IDs are used to reference resources within the template. Physical IDs identify resources outside of AWS CloudFormation templates, but only after the resources have been created.

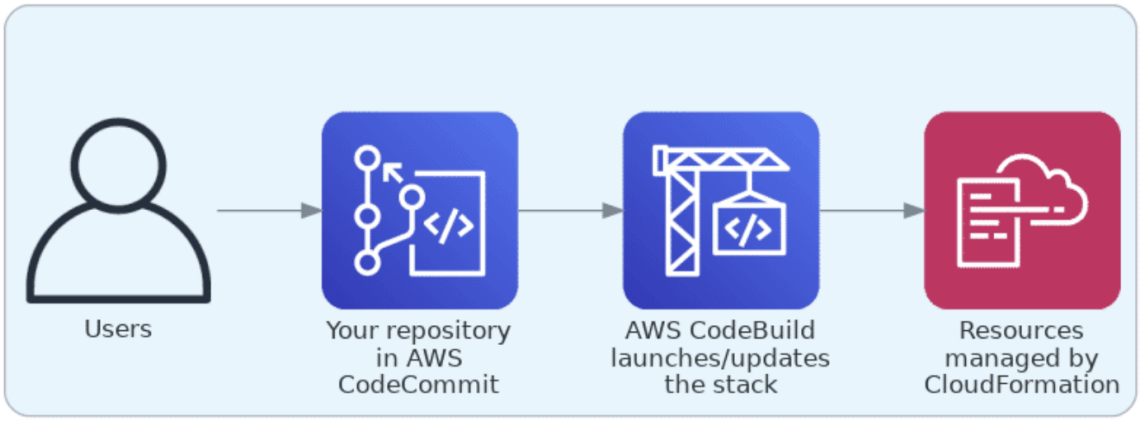
- AWS CloudFormation supports Puppet and Chef integration.
- It can use bootstrap scripts.
- It can define deletion policies.
- AWS CloudFormation provides WaitCondition function.
- It can create roles in IAM.
- VPCs can be created and customized.
- VPC peering in the same AWS account can be performed.
- Route 53 is supported.
- Automatic rollback on error is enabled by default.
- AWS CloudFormation provides two methods for updating stacks: direct update or creating and executing change sets.
- When you directly update a stack, you submit changes, and AWS CloudFormation immediately deploys them.
- Use direct updates when you want to deploy your updates quickly.
- With change sets, you can preview the changes AWS CloudFormation will make to your stack and then decide whether to apply those changes.

- There is no additional charge for AWS CloudFormation.
- You pay for AWS resources (such as Amazon EC2 instances, Elastic Load Balancing load balancers, etc.) created using AWS CloudFormation in the same manner as you would if you created them manually.
- By ensuring the CloudFormation templates are created and administered with the right security configurations for your resources, you can then repeatedly deploy resources with secure settings and reduce the risk of human error.

To be able to deploy resources across multiple AWS accounts and regions using a single toolset and template. Use a CloudFormation StackSet and specify the target accounts and regions in which the stacks will be created

Create an IAM policy with a Condition: TemplateURL parameter to ensure that users are restricted to a specified template.

With CloudFormation, you can apply DevOps and GitOps best practices using widely adopted processes such as starting with a Git repository and deploying through a continuous integration and continuous delivery (CI/CD) pipeline. You can also simplify auditing changes and trigger automated deployments with pipeline integrations such as GitHub Actions and AWS CodePipeline.



Using CloudFormation in Your Architecture

Application Integration Services

Amazon SNS (Simple Notification Service)

A fully managed pub/sub messaging service, that helps in decoupling subscribers from publishers using topics. This plays a role in scaling microservices, distributed systems, and serverless infrastructure. Usually, used for mass message delivery for mobile users.

SNS supports notifications over multiple transport protocols, which are explained below.

- HTTP/HTTPS: Subscribers specify a URL as part of the subscription registration.
- Email/Email-JSON: Messages are sent to registered addresses as email (text-based or JSON-object).
- SQS: Users can specify an SQS standard queue as the endpoint.
- SMS: Messages are sent to registered phone numbers as SMS text messages.
- Lambda: Amazon SNS supports Lambda functions as a target for messaging sent to a topic.

Limitations

- Up to 256 KB of data.
- Max size of single SMS is 140 bytes, larger messages sent as multiple transmissions. Aggregate SMS message size is 1600 bytes.

Amazon SQS (Simple Queue Service)

Eliminates the overhead of managing and operating message oriented middleware. It is a pull based queue.

SQS uses a pull-based (polling) and not a push-based approach.

- Messages can be kept in the queue from 1 minute to 14 days (the default is 4 days).
- The visibility timeout is the amount of time a message is invisible in the queue after a reader picks up the message.
    - If a job is processed within the visibility timeout, the message will be deleted.
    - If a job is not processed within the visibility timeout, the message will become visible again (it could be delivered twice).
    - The maximum visibility timeout for an Amazon SQS message is 12 hours.
- An Amazon SQS message can contain up to 10 metadata attributes.
- An Amazon Simple Queue Service (SQS) can be used to offload and decouple the long-running requests. They can then be processed asynchronously by separate EC2 instances. This is the best way to reduce the overall latency introduced by the long-running API call.

Polling

Short polling

- Short polling does not wait for messages to appear in the queue.
- It queries only a subset of the available servers for messages (based on weighted random execution).
- It is the default.
- ReceiveMessageWaitTime is set to 0.
- More requests are used, which implies a higher cost.

Long polling

- Long polling uses fewer requests and reduces cost.
- It eliminates false empty responses by querying all servers.
- SQS waits until a message is available in the queue before sending a response.
- SQS will return at least one message in the response to a ReceiveMessage request, up to the maximum number of messages that you specified in the request.
- It shouldn't be used if your application expects an immediate response to receiving message calls.
- ReceiveMessageWaitTime is set to a non-zero value (up to 20 seconds).
- It has the same charge per million requests as short polling.

Queues

Queue names must be unique within a region. In-flight messages are messages that have been picked up by a consumer but have not yet been deleted from the queue.

Standard Queue

Standard queues provide a loose-FIFO capability that attempts to preserve the order of messages. Because standard queues are designed to be massively scalable using a highly distributed architecture, receiving messages in the exact order that they are sent is not guaranteed. Standard queues provide at-least-once delivery, which means that each message is delivered at least once. Has unlimited throughput. Standard queues have a limit of 120,000 in-flight messages per queue.

FIFO queue

FIFO (first-in-first-out) queues preserve the exact order in which messages are sent and received. If you use a FIFO queue, you don't have to place sequencing information in your message. FIFO queues provide exactly-once processing, which means that each message is delivered once and remains available until a consumer processes it and deletes it. It has high throughput. FIFO queues have a limit of 20,000 in-flight messages per queue.

Scalability and durability

- You can have multiple queues with different priorities.
- Scaling is performed by creating more queues.
- SQS stores all message queues and messages within a single, highly-available AWS Region with multiple redundant AZs.

Security

You can use IAM policies to control who can read/write messages. Authentication can be used to secure messages within queues (who can send and receive).
SQS supports HTTPS and TLS versions 1.0, 1.1, and 1.2. SQS is PCI DSS level 1 compliant and HIPAA eligible.

Encryption

Server-side encryption (SSE) lets you transmit sensitive data in encrypted queues (AWS KMS).

- SSE encrypts messages as soon as SQS receives them.
- The messages are stored in encrypted form, and SQS decrypts messages only when sent to an authorized consumer.
- It uses AES 256 bit encryption.
- It is not available in all regions.
- SQS supports in-transit encryption using HTTPS. All communication between the SQS API and your application is encrypted using HTTPS.
- SQS also supports at-rest encryption using AWS Key Management Service (KMS). At-rest encryption encrypts your messages when they are stored on disk in SQS data centers.
- The body of the message is encrypted. The following is not encrypted:
    - Queue metadata
    - Message metadata
    - Per-queue metrics

Monitoring

- CloudWatch is integrated with SQS, and you can view and monitor queue metrics.
- No charge for CloudWatch (no detailed monitoring).
- CloudTrail captures API calls from SQS and logs to a specified S3 bucket.
- CloudTrail is used for recording API calls (auditing), whereas CloudWatch is used for recording metrics (performance monitoring). The solution can be deployed with a single trail that is applied to all regions. A single KMS key can be used to encrypt log files for trails applied to all regions. CloudTrail log files are encrypted using S3 Server Side Encryption (SSE), and you can also enable encryption SSE KMS for additional security. One CloudTrail can be used for all regions, you do not need to create a separate trail in each region.

Limitations

- Messages can be up to 256 KB.
- Messages can be retained for 14 days.

Amazon MQ

It is a managed message broker (Active MQ) service that makes it easy to set up and operate message brokers in the cloud. Message brokers allow different software systems - often using different programming languages, and on different platforms to communicate and exchange information. It has been designed as a drop-in replacement for on-premise message brokers.

It’s a managed implementation of Apache ActiveMQ. It provides cost-efficient and flexible messaging capacity – you pay for broker instance and storage usage as you go.

Use SQS if you're creating a new application from scratch. Use MQ if you want an easy, low-hassle path to migrate from existing message brokers to AWS.

Availability

- Amazon MQ is fully managed and highly available within a region.
- Amazon MQ stores your messages redundantly across multiple Availability Zones (AZs).
- Active/standby brokers are designed for high availability.
- In the event of a failure of the broker, or even a full AZ outage, Amazon MQ automatically fails over to the standby broker so you can continue sending and receiving messages.

Security

- Amazon MQ provides encryption of your messages at rest and in transit.
- It's easy to ensure that your messages are securely stored in an encrypted format.
- Connections to the broker use SSL, and access can be restricted to a private endpoint within your Amazon VPC, which allows you to isolate your broker in your own virtual network.
- You can configure security groups to control network access to your broker.

Monitor

Amazon MQ is integrated with Amazon CloudWatch and AWS CloudTrail. With CloudWatch, you can monitor metrics on your brokers, queues, and topics.

AWS SWF (Simple Workflow Service)

Used for triggering a workflow. For example, imagine that you have an expense report that needs to be approved by three people. When you submit an expense report the workflow is triggered and workflow services keep track of each individual who needs to approve your expenses report. SWF supports both sequential and parallel processing. It tracks the state of your workflow (which you interact with) and updates via API. It is best suited for human-enabled workflows like an order fulfillment system or for procedural requests.

AWS recommends Step Functions eventually replacing SWF.

AWS Step Functions

It is a managed workflow and orchestration platform that is scalable and highly available. You define your app as a state machine. You can create tasks, sequential steps, parallel steps, branching paths, or timers. Apps can interact and update the stream via the Step Function API.

Step Functions maintains application state, tracking exactly which workflow step your application is in, and stores an event log of data that is passed between application components.

Standard workflows are better for 2000 requests per second wile express workflows are better for 100,000 requests per second.

Standard Workflows are ideal for long-running, durable, and auditable workflows. Express Workflows are ideal for high-volume, event-processing workloads, such as IoT data ingestion, streaming data processing and transformation, and mobile application backends.

Express Workflows log to CloudWatch by default whereas Standard Workflows can be optionally chosen to log to CloudWatch.

There can be unlimited number of terminal states per state machine. Only one Next or End can be used in a state.

Here's a brief overview of the core states in AWS Step Functions:

- **Task State:** Performs work by invoking an AWS service or resource (e.g., Lambda function, SNS, SQS).
- **Choice State:** Conditionally branches execution based on input values or conditions.
- **Wait State:** Pauses execution for a specified duration or until a timestamp is reached.
- **Pass State:** Passes input to output without performing work, often used for transitions, delays, or input modification.
- **Parallel State:** Executes multiple branches of states concurrently.
- **Map State:** Iterates over an array of elements, executing a set of actions for each element concurrently.
- **Fail State:** Immediately terminates the execution with a failure status.
- **Succeed State:** Immediately terminates the execution with a success status.
- **Callback State:** Pauses execution and awaits a call to resume it.

Pricing

With AWS Step Functions (Standard Workflow), you pay for each transition from one state to the next. Billing is metered by state transition, and you do not pay for idle time, regardless of how long each state persists (up to one year). This keeps Step Functions cost-effective as you scale from a few executions to tens of millions.

Express workflows are priced by the number of times you run, their duration, and memory consumption.

Amazon Kinesis

Streaming Data

Data that is generated continuously by thousands or millions of data sources or data points that are captured and needs to be picked up and processed.

It is similar to Kafka but on AWS. It has three services:

Kinesis Streams

Consists of shards (clusters in Kafka).

Kinesis Video Streams

Kinesis Video Streams makes it easy to securely stream video from connected devices to AWS for analytics, machine learning (ML), and other processing.
It durably stores, encrypts, and indexes video data streams and allows access to data through easy-to-use APIs.

Kinesis Data Streams

Kinesis Data Streams enables you to build custom applications that process or analyze streaming data for specialized needs.
Kinesis Data Streams allows for real-time processing of streaming big data. It is useful for rapidly moving data off of data producers and continuously processing the data.
Kinesis Data Streams stores data for later processing by applications (This is the key difference between Kinesis and Firehose, which delivers data directly to AWS services).

Use Cases

- Accelerated log and data feed intake.
- Real-time metrics, reporting and data analytics.
- Complex stream processing.
- Producers continually push data to Kinesis Data Streams.
- Consumers process the data in real-time.
- Consumers can store their results using an AWS service such as Amazon DynamoDB, Amazon Redshift, or Amazon S3.
- Kinesis Streams applications are consumers that run on EC2 instances.
- Shards are uniquely identified groups or data records in a stream.
- Records are the data units stored in a Kinesis Stream.

Producer

A producer creates the data that makes up the stream. Producers can be used through the following:

- Kinesis Streams API
- Kinesis Producer Library (KPL)
- Kinesis Agent

Consumer

- Consumers are known as Amazon Kinesis Streams Applications.

Records

- A record is the unit of data stored in an Amazon Kinesis data stream. A record is composed of a sequence number, partition key, and data blob.
- By default, records of a stream are accessible for up to 24 hours from the time they are added to the stream (can be raised to seven days by enabling extended data retention).

Shard

A shard is the base throughput unit of an Amazon Kinesis data stream.

- A stream is composed of one or more shards.
- Partition keys are used to group data by shard within a stream.

When the data rate increases, add more shards to increase the size of the stream. Remove shards when the data rate decreases.

Replication

- Kinesis Data Streams replicates synchronously across three AZs.

Security

Encryption

- Kinesis Streams uses KMS master keys for encryption.
- To read from or write to an encrypted stream, the producer and consumer applications must have permission to access the master key.

Capacity and Throttling

One shard provides a capacity of 1 MB/sec data input and 2 MB/sec data output. One shard can support up to 1000 PUT records per second. The total capacity of the stream is the sum of the capacities of its shards.

In a case where multiple consumer applications have total reads exceeding the per-shard limits, you need to increase the number of shards in the Kinesis data stream. Read throttling is enabled by default for Kinesis data streams. If you're still experiencing performance issues, you must increase the number of shards. You cannot increase the number of read transactions per shard.

Kinesis Firehose

It is used to reliably load streaming data into data stores and analytics tools (Amazon S3, Amazon Redshift, and Splunk). You can configure Kinesis Data Firehose to transform your data before delivering it. Firehose can batch, compress, and encrypt data before loading it. With Kinesis Data Firehose, you don't need to write an application or manage resources. There are no shards, it is totally automated.

A delivery stream is the underlying entity of Amazon Kinesis Data Firehose. Each delivery stream stores data records for up to 24 hours.

- Firehose can encrypt data with an existing AWS Key Management Service (KMS) key.
- Firehose synchronously replicates data across three AZs as it is transported to destinations.
- For Amazon S3 destinations, streaming data is delivered to your S3 bucket. If data transformation is enabled, you can optionally back up source data to another Amazon S3 bucket.
- For Amazon Redshift destinations, streaming data is delivered to your S3 bucket first. Kinesis Data Firehose then issues an Amazon Redshift COPY command to load data from your S3 bucket to your Amazon Redshift cluster. If data transformation is enabled, you can optionally back up source data to another Amazon S3 bucket.
- For Amazon Elasticsearch destinations, streaming data is delivered to your Amazon ES cluster, and it can optionally be backed up to your S3 bucket concurrently.
- For Splunk destinations, streaming data is delivered to Splunk, and it can optionally be backed up to your S3 bucket concurrently.

Kinesis Analytics

Enables you to query streaming data using SQL to gain actionable insights. It can ingest data from Kinesis Streams and Kinesis Firehose. It can output to S3, RedShift, Elasticsearch, and Kinesis Data Streams. It sits over Kinesis Data Streams and Kinesis Data Firehose.

Input

It is the streaming source for your application. It supports two types of inputs:

Streaming Data Sources

It is continuously generated data that is read into your application for processing.

Reference Data Source

It is static data that your application uses to enrich data coming in from streaming sources.

Application code

A series of SQL statements that process input and produce output.

Output

One or more in-application streams to hold intermediate results.

Security

IAM can be used to provide Kinesis Analytics with permissions to read records from sources and write to destinations.

Kinesis Data Stream vs Amazon SQS

- Use Kinesis Data Streams if:
    - You have high-volume, real-time data streams to process.

    - You need low latency for near-instantaneous data analysis.
    - You prefer parallel processing with ordered or unordered data.
- Use SQS if:
    - You need reliable message delivery with decoupling between applications.
    - You require guaranteed message ordering for specific workflows.
    - You don't have real-time processing requirements and prefer asynchronous communication.

Networking and Content Delivery

MPLS network

An MPLS network is used to create a network topology that gets you closer to AWS in each region, but you would still need to use Direct Connect or VPN for connectivity into AWS. MPLS is not offered as a service by AWS.

Amazon Cognito

It is used for device authentication / OAuth service. This service provides temporary access to AWS resources. Imagine you have an app that lets users upload pictures onto your S3. You can do this by using Cognito.

Cognito controls access to AWS resources:

- Define roles
- Map users to roles

The two main components of AWS Cognito are user pools and identity pools.

User pools

User pools are user directories that provide sign-up and sign-in options for your app users.

Identity pools

Identity pools enable you to grant your users (federated identities) access to other AWS services. Amazon Cognito identity pools provide temporary AWS credentials for users who are guests (unauthenticated) and for users who have been authenticated and have received a token. An identity pool is a store of user identity data specific to your account. You can use identity pools and user pools separately or together. There is an import tool for migrating users into an Amazon Cognito user pool. Identity pools can provide access through user pools or through direct social identity provider login. Identity pool does not save user profiles., it saves only when integrated with user pools

AWS Cognito works with external identity providers that support SAML or OpenID Connect (is a provider for connecting external directories.) and social identity providers (such as Facebook,

Twitter, Amazon). You can also integrate your own identity provider. Users can sign-up and sign-in using email, phone number, or username. End-users of an application can also sign in with SMS-based MFA.

Cognito exposes server-side APIs. Cognito Identity provides temporary security credentials to access your app's backend resources in AWS or any service behind Amazon API Gateway.

After successfully authenticating a user, Amazon Cognito issues JSON web tokens (JWT) that you can use to secure and authorize access to your own APIs or exchange for AWS credentials.

Amazon Cognito is used for authenticating users to web and mobile apps, not for providing single sign-on between on-premises directories and the AWS Management Console.

Amazon Cognito Sync

It is an AWS service and client library that enables cross-device syncing of application-related user data. You can use it to synchronize user profile data across mobile devices and the web without requiring your own backend.

The client libraries cache data locally so your app can read and write data regardless of device connectivity status. When the device is online, you can synchronize data, and if you set up push sync, you can notify other devices immediately that an update is available.

Amazon Inspector

An agent installed on your virtual machine and you run tests for security vulnerabilities. Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. It is not used to secure the actual deployment of resources, only to assess the deployed state of the resources. Amazon Inspector removes the operational overhead that is necessary to configure a vulnerability management solution. Amazon Inspector works with both EC2 instances and container images in Amazon ECR to identify potential software vulnerabilities and to categorize the severity of the vulnerabilities.

Certificate Manager

Issues and manages certificates to domains registered via AWS/Routes 53.

AWS Secure Token Service (STS)

A web service that enables you to request temporary, limited-privilege credentials for IAM users or for users that you authenticate. All regions are enabled for STS by default but can be disabled. The region in which temporary credentials are requested must be enabled; however, credentials will always work globally.

With STS, you can request a session token using one of the following APIs:

AssumeRole	Can only be used by IAM users (can be used for MFA)
AssumeRoleWithSAML	Can be used by any user who passes a SAML authentication response that indicates authentication from a known (trusted) identity provider
AssumeRoleWithWebIdentity	Can be used by a user who passes a web identity token that indicates authentication from a known (trusted) identity provider
GetSessionToken	Can be used by an IAM user or AWS account root user (can be used for MFA)
GetFederationToken	Can be used by an IAM user or AWS account root user

SAML

It is used for authentication and grants temporary access based on the user's Active Directory credentials. Users do not need to be a user in IAM.

STS and SAML are used together to provide a single sign-on solution.

AWS Organizations

AWS Organizations helps you centrally govern your environment as you grow and scale your workloads on AWS. It helps you centrally manage billing, control access, compliance, security, and share resources across your AWS accounts. AWS Organizations is available to all AWS customers at no additional charge.

There is a default limit of 20 linked accounts for consolidated billing.

AWS account	An AWS account is a container for your AWS resources.
Master account	A master account is the AWS account that you use to create your organization.
Member Account	A member account is an AWS account, other than the master account, that is part of an organization.
Administrative root	An administrative root is the starting point for organizing your AWS accounts. The administrative root is the top-most container in your organization’s hierarchy.

Organizational Unit (OU)	An organizational unit (OU) is a group of AWS accounts within an organization. An OU can also contain other OUs enabling you to create a hierarchy. You can nest OUs up to five levels deep.
Policy	A policy is a “document” with one or more statements that define the controls that you want to apply to a group of AWS accounts.

Accounts Migration

Accounts can be migrated between organizations. This can be done using the AWS Organizations console for a few accounts. If there are many accounts then you can use Organizations API or AWS CLI and AWS CloudFormation to automate this process.

- You must have root or IAM access to both the member and master accounts.
- Before migration, download any billing or report history for any member accounts you want to keep.
- When a member account leaves an organization, all charges incurred by the account are charged directly to the standalone account.
- Even if the account move only takes a minute to process, it is likely that the member account will incur some charges.

Service Control Policy

AWS Organizations support a specific type of policy called a Service Control Policy (SCP). An SCP defines AWS service actions (such as Amazon EC2 RunInstances) that are available for use in different accounts within an organization. Policies can be assigned at different points in the hierarchy. To apply the restrictions across multiple member accounts, you must use a Service Control Policy (SCP). With IAM you need to apply the policy within each account rather than centrally, so this would require much more effort.

It requires that all Organization features are enabled.
SCPs alone are not sufficient for allowing access in the accounts in your organization. Attaching an SCP to an AWS Organizations entity (root, OU, or account) defines a guardrail for the specific actions that the principals can perform. You still need to attach identity-based or resource-based policies to principals or resources in your organization’s accounts to actually grant permissions to them.

SCP Inheritance

Following is the inheritance in SCP:

Root SCP -> OU SCP -> Account SCP.

Deny statements apply to all lower levels, if there is a policy that denies something in Root SCP then that applies to OU SCP and if a policy in OU SCP that denies something then it applies to Account SCP.

Allow statements do the same unless overridden. An implicit deny is the default without the FullAWSAccess SCP attached.

SCP Strategy

Deny List Strategy

- Leave the default SCP (FullAWSAccess) in place.
- Apply deny SCPs at appropriate lower levels.

Allow List Strategy

- Remove the default SCP (FullAWSAccess).
- Apply allow SCPs at appropriate lower levels.

Primary Rule: Implicit deny can be overridden, explicit deny can not be overridden.

Pricing

Free

AWS Key Management Store (KMS)

AWS Key Management Store (KMS) is a managed service that enables you to encrypt your data easily. It provides a highly available key storage, management, and auditing solution for you to encrypt data within your own applications and control the encryption of stored data across AWS services.

You can submit data directly to KMS to be encrypted or decrypted using these master keys. You set usage policies on these keys, which determine which users can use them to encrypt and decrypt data and under what conditions. KMS generates data keys that are used to encrypt data and the data keys themselves are encrypted using your master keys in KMS. Data keys are not retained or managed by KMS.

- Create keys with a unique alias and description.
- Import your own key material.
- Define which IAM users and roles can manage keys.
- Define which IAM users and roles can use keys to encrypt and decrypt data.
- Choose to have AWS KMS automatically rotate your keys on an annual basis.
- Temporarily disable keys so no one can use them.
- Re-enable disabled keys.
- Delete keys that you no longer use.

- Create custom key stores.
- Connect and disconnect custom key stores.
- Delete custom key stores.

Integration with CloudHSM Key Store

- The use of custom key stores requires CloudHSM resources to be available in your account.
- The AWS KMS custom key store feature combines the controls provided by AWS CloudHSM with the integration and ease of use of AWS KMS.
- You can configure your own CloudHSM cluster and authorize KMS to use it as a dedicated key store for your keys rather than the default KMS key store.
- With AWS CloudHSM, your keys are held in AWS in a hardware security module. Again, the keys are not on-premises; instead, they are in AWS

Monitoring

All requests to use your master keys are logged in AWS CloudTrail so you can understand who used which key under which context. You can audit the use of keys via CloudTrail.

Encryption

Envelope encryption

When you encrypt your data, your data is protected, but you still have to protect your encryption key. One strategy is to encrypt it. Envelope encryption is the practice of encrypting plaintext data with a data key and then encrypting the data key under another key. Protecting data keys: When you encrypt a data key, you don't have to worry about storing the encrypted data key because the data key is inherently protected by encryption. You can safely store the encrypted data key alongside the encrypted data.

- Encrypting the same data under multiple master keys: Encryption operations can be time-consuming, particularly when the data being encrypted are large objects. Instead of re-encrypting raw data multiple times with different keys, you can re-encrypt only the data keys that protect the raw data.
- Combining the strengths of multiple algorithms: In general, symmetric key algorithms (same key for encryption and decryption) are faster and produce smaller ciphertexts than public-key algorithms. But public-key algorithms (key pair) provide inherent separation of roles and easier key management. Envelope encryption lets you combine the strengths of each strategy.

AWS CloudHSM

What is it?

- CloudHSM provides secure hardware for storing and managing sensitive cryptographic keys like encryption keys, digital signing keys, and root of trust keys.
- It eliminates the need for managing your own physical HSMs, simplifying key management and enhancing security.

Key Features:

- Dedicated HSMs: FIPS 140-2 Level 3 validated HSMs offer the highest level of security for your keys.
- High Availability: Clusters of HSMs ensure redundancy and uptime for key access.
- Key Management Functions: Generate, import, export, rotate, and delete keys within the HSMs.
- Custom Cryptographic Operations: Perform various cryptographic operations using the HSMs.
- Integration with AWS Services: Limited native integration with services like CloudTrail and S3, but custom integrations possible via SDKs.
- Compliance: Helps meet strict compliance requirements like PCI DSS and HIPAA.

Benefits:

- Enhanced Security: HSMs provide tamper-proof hardware for key storage, mitigating risks of key theft or compromise.
- Simplified Management: Eliminates the need for managing and maintaining physical HSMs.
- Scalability: Easily scale your HSM capacity as your needs grow.
- High Availability: Redundant HSM clusters ensure constant access to your keys.
- Compliance: Helps meet key security and compliance requirements.

Use Cases:

- Protecting sensitive keys for encryption and decryption in various applications.
- Securing digital signatures for documents and applications.
- Managing root of trust keys for PKI environments.
- Meeting compliance requirements for data security and privacy.

Considerations:

- Cost: CloudHSM can be more expensive than software-based key management solutions.
- Custom Integration: Most AWS service integrations require custom application-level scripting.
- Complexity: Managing HSMs and cryptographic operations can be more complex than using software-based solutions.

Choosing the Right Option:

- KMS: If you prioritize ease of use, integration with AWS services, and don't have strict compliance requirements.
- CloudHSM: If you need the highest level of security for your keys, require fine-grained control over key management, and have strict compliance needs.

	CloudHSM	AWS KMS
Tenancy	Single-tenant HSM	Multi-tenant AWS service
Availability	Customer-managed durability and available	Highly available and durable key storage and management
Root of Trust	Customer managed root of trust	AWS managed root of trust
FIPS 140-2	Level 3	Level 2 / Level 3 in some areas
3rd Party Support	Broad 3rd Party Support	Broad AWS service support

Secrets Manager

Allows for applications to retrieve credentials through APIs.

AWS Resource Access Manager

- Purpose: Enables sharing of specific AWS resources across accounts or within your AWS Organization.
- Shareable Resources: Transit Gateways, Subnets, License Manager configurations, and Route 53 Resolver rules.
- Benefits: Eliminates duplicate resources, reduces operational overhead, simplifies cross-account sharing.
- Cost: No additional charge.
- Limitations: Not for sharing APIs or restricting access permissions.

Resource Groups:

- Purpose: Organize and manage AWS resources based on specific criteria.
- Organization: Group resources in the same region based on tags or CloudFormation stacks.
- Management: Facilitate bulk actions and automation on groups of resources.
- Query Types: Tag-based or CloudFormation stack-based.
- Nesting: Resource groups can contain other resource groups in the same region.

Resource Access Manager (RAM) vs Shared Services VPC

Use RAM if:

- You need to share individual resources with granular control.
- You have a multi-account environment with diverse resource requirements.
- You want centralized management of resource sharing policies.

Use Shared Services VPC if:

- You want to centralize common resources and services for multiple accounts.
- You have applications or environments with standardized configurations.
- You prioritize a single VPC structure for simplified management.

AWS Directory Service

Federated Identity

A system of trust between two parties. One party trusts the others to authenticate the users. AWS supports standardized protocols:

1. Security Assertion Markup Language (SAML) 2.0.
2. Open ID Connect (OIDC).
3. Open Authentication (OAuth) 2.0.

Federated Directory Service

An entire directory of users or groups may be linked. Extra authorization information can be provided.

You manage permissions and authentication in your directory.

AWS Single-Sign-On

Rebranded to IAM identity center. Connects existing directories to AWS. May use AWS Identity Center internal directory as well. Requires the use of AWS Organizations. May connect with Microsoft Active Directory (AWS Managed AD, Self-Managed AD (requires AD connector))

AWS provides several directory types.

- Active Directory Service for Microsoft Active Directory
- Simple AD
- AD Connector

As an alternative to the AWS Directory service, you can build your own Microsoft AD DCs in the AWS cloud (on EC2):

- When you build your own, you can join an existing on-premise Active Directory domain (replication mode).
- You must establish a VPN (on top of Direct Connect if you have it).
- Replication mode is less secure than establishing trust relationships.

Directory Service Option	Description	Use Case
AWS Cloud Directory	Cloud-native directory to share and control access to hierarchical data between applications	Cloud applications that need hierarchical data with complex relationships
Amazon Cognito	Sign-up and sign-in functionality that scales to millions of users and federated to public social media services	Develop consumer apps or SaaS
AWS Directory Service for Microsoft Active Directory	AWS-managed full Microsoft AD running on Windows Server 2012 R2	Enterprises that want hosted Microsoft AD or you need LDAP for Linux apps
AD Connector	Allows on-premises users to log into AWS services with their existing AD credentials; also allows EC2 instances to join AD domain	Single sign-on for on-premises employees and for adding EC2 instances to the domain
Simple AD	Low scale, low cost, AD implementation based on Samba	Simple user directory, or you need LDAP compatibility

The Active Directory Service is a fully managed AWS service on AWS infrastructure. It is the best choice if you have more than 5000 users and/or need a trust relationship set up. It includes software patching, replication, automated backups, replacing failed DCs, and monitoring.
It runs on a Windows Server and works with SharePoint, Microsoft SQL Server, and .Net apps. It can also perform schema extensions. You can set up trust relationships to extend authentication from on-premises Active Directories into the AWS cloud. On-premise users and groups can access resources in either domain using SSO. It requires a VPN or Direct Connect connection. The ability to grant your users access to external cloud applications and to allow your on-premises AD users to manage and have access to resources in the AWS Cloud.
AWS Microsoft AD directory includes security features such as:

- Fine-grained password policy management
- LDAP encryption through SSL/TLS
- HIPAA and PCI DSS approved
- Multi-factor authentication through integration with existing RADIUS-based MFA infrastructure

Monitoring and HA

- Monitoring is provided through CloudTrail, and notifications are provided through SNS. Daily automated snapshots are also available.
- It is a scalable service that scales by adding Domain Controllers.

- It is deployed in an HA configuration across two AZs in the same region. AWS Microsoft AD does not support replication mode where replication to an on-premise AD takes place.
- Standard Edition is optimized to be a primary directory for small and midsize businesses with up to 5,000 employees. It provides you with enough storage capacity to support up to 30,000 directory objects such as users, groups, and computers.
- Enterprise Edition is designed to support enterprise organizations with up to 500,000 directory objects.
- AWS Directory Service for Microsoft Active Directory allows you to use a directory in one account and share it with multiple accounts and VPCs.
- There is an hourly sharing charge for each additional account with which you share a directory.
- There is no sharing charge for additional VPCs with which you share a directory or for the account in which you install the directory.

Simple AD

Simple AD is an inexpensive Active Directory-compatible service with common directory features. It is a standalone, fully managed directory on the AWS cloud. It is powered by the SAMBA 4 Active Directory compatible server.

Simple AD is generally the least expensive option. It is the best choice for less than 5000 users and companies that do not need advanced AD features. You can create users and control access to applications on AWS. It provides a subset of the features provided by AWS MS AD.

AD Connector

AD Connector is a directory gateway for redirecting directory requests to your on-premise Active Directory. AD Connector eliminates the need for directory synchronization and the cost and complexity of hosting a federation infrastructure.

It connects your existing on-premise AD to AWS. It is the best choice when you want to use an existing Active Directory with AWS services.

The VPC must be connected to your on-premise network via VPN or Direct Connect. You can also join EC2 instances to your on-premise AD through AD Connector.

When users log in to AWS applications, the AD connector forwards sign-in requests to your on-premise AD DCs. You can also log in to the AWS Management Console using your on-premise AD DCs for authentication.

AWS Control Tower

It sets up and governs your multi-account AWS environment. It uses Organization Units. It also provides policy management and a dashboard for easy visibility.

There’s no extra charge to use AWS Control Tower, but it does make use of services that cost money.

Data Protection

- Encrypt data at rest.
- Encrypt data in motion.
- Regular key rotation.
    a. Advisable for user accounts, roles on the other hand are temporary and do not require rotation.
- Detailed logging of changes and access to files.
- Versioning to protect against accidental overwrites deletes.

Well-Architected Framework

A framework for evaluating cloud architectures against the qualities you expect from modern cloud based systems. It is based on five pillars.

1. Operational Excellence - Ability of a system to run and support existing and future processes that deliver business value.
2. Security - Ability to protect assets, information and have a risk assessment and mitigation strategies.
3. Reliability - Ability of a system to recover from disruptions and be able to adjust according to demand.
4. Performance Efficiency - Ability to use computing resources efficiently to meet system requirements and accommodate demand changes.
5. Cost optimization - Ability to avoid unneeded cost or suboptimal resources.

Security and Operation excellence are not compromised generally however, other pillars are compromised based on requirements e-g reliability can be compromised for saving cost.

AWS Cost Management

AWS Budgets

AWS Budgets tracks and takes action on your AWS cost and usage. AWS Budgets gives you the ability to set custom budgets that alert you when your costs or usage exceed (or are forecasted to exceed) your budgeted amount. You can also use AWS Budgets to set reservation utilization or coverage targets and receive alerts when your metrics drop below the threshold you define.

AWS requires approximately five weeks of usage data to generate budget forecasts. If you set a budget to alert based on a forecasted amount, this budget alert isn't initiated until you have enough historical usage information.

AWS Cost Explorer

- It is disabled by default.
- You need to wait at least 24 hours after enabling it to see the results.
- It gives you granular details about your spending before the end of month bill and it also gives recommendations to save money.

Cost allocation Tags

To help you manage your instances, images, and other AWS resources, you can assign your own metadata to each resource in the form of tags. Tags help you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. This is useful when you have many resources of the same type—you can quickly identify a specific resource based on the tags that you've assigned to it. Cost allocation tags are specific to billing and track your AWS costs on a detailed level. After you activate cost allocation tags, AWS uses the cost allocation tags to organize your resource costs on your cost allocation report, to make it easier for you to categorize and track your AWS costs. Cost allocation tags can be used on S3 buckets to assist in cost tracking. AWS provides two types of cost allocation tags—AWS generated tags and user-defined tags.

Management Services

AWS Amplify

AWS Amplify is a set of tools and services for building full-stack web and mobile applications on AWS. It simplifies the development process by providing pre-built components and features for common tasks like authentication, data storage, API creation, and hosting.

Benefits:

- Faster Development: Reduce development time with pre-built components and features.
- Simplified Infrastructure: Manage backend infrastructure through Amplify, eliminating the need for manual configuration.
- Scalability: Easily scale your application based on demand.
- Security: Built-in security features help protect your application from threats.
- Cost-Effectiveness: Pay only for resources you use.

AppSync

Provides a GraphQL-based solution for data integration. Data sources include DynamoDB, Lambda, and HTTP APIs. Effectively acts as a GraphQL proxy allowing for access to other data sources that may not be GraphQL native.

AWS Serverless Application Model (SAM)

AWS Serverless Application Model (AWS SAM) is an extension of AWS CloudFormation that is used for packaging, testing, and deploying serverless applications.

AWS SAM CLI for testing
With AWS SAM CLI for testing, you can do the following:

- Invoke functions and run automated tests locally.
- Generate sample event source payloads.
- Run API Gateway locally.
- Debug code.
- Review Lambda function logs.
- Validate AWS SAM templates.

Deploying Lambda functions through AWS CloudFormation requires an Amazon S3 bucket for the Lambda deployment package. The SAM CLI creates and manages this Amazon S3 bucket for you.

AWS Config

AWS Config is a fully managed service that provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and governance. With AWS Config, you can discover existing AWS resources, export a complete inventory of your AWS resources with all configuration details, and determine how a resource was configured at any point in time. These capabilities enable compliance auditing, security analysis, resource change tracking, and troubleshooting.

AWS Config Rules can check resources for certain desired conditions, and if violations are found, the resources are flagged as “non-compliant”.

Examples of Config Rules

- Is backup enabled on RDS?
- Is CloudTrail enabled on the AWS account?
- Are EBS volumes encrypted?

A Configuration Item (CI) is the configuration of a resource at a given point-in-time.

- With AWS Config, you are charged based on the number of configuration items (CIs) recorded for supported resources in your AWS account.
- AWS Config creates a configuration item whenever it detects a change to a resource type that it is recording.

Ops works

Uses Chef and Puppet in a way to automate your environments. It automates how applications are configured, deployed, and managed. It provides configuration management to deploy code, automate tasks, configure instances, perform upgrades, etc. OpsWorks lets you use Chef and Puppet to automate how servers are configured, deployed, and managed across your Amazon EC2 instances or on-premises compute environments. OpsWorks is an automation platform that transforms infrastructure into code.

OpsWorks consists of Stacks and Layers.

- Stacks are containers of resources (EC2, RDS, etc.) that you want to manage collectively.
- Every Stack contains one or more Layers, and Layers automate the deployment of packages.
- Stacks can be cloned but only within the same region.
- Layers represent different components of the application delivery hierarchy.
- EC2 instances, RDS instances, and ELBS are examples of Layers.

AWS OpsWorks Stacks lets you manage applications and servers on AWS and on-premises and uses Chef Solo.

OpsWorks is a global service, but when you create a stack, you must specify a region, and that stack can only control resources in that region.

AWS OpsWorks for Chef Automate is a fully-managed configuration management service that hosts Chef Automate, a suite of automation tools from Chef for configuration management, compliance and security, and continuous deployment. OpsWorks for Chef Automate is completely compatible with tooling and cookbooks from the Chef community and automatically registers new nodes with your Chef server. Cookbooks contain recipes and recipes are equivalent to layers. In recipes we define configuration settings (Admin defined, AWS defined, Third-party defined).

OpsWorks Puppet contains master servers, which contain pre-configured modules and modules are equivalent to layers.

Use Cases

In the cloud:

- Chef

- Puppet

On-premisses (local):

- Stacks

AWS OpsWorks vs AWS SSM

AWS OpsWorks: Primarily for automating deployments and managing configurations of applications on AWS, using Chef or Puppet as configuration management tools.

AWS SSM: Primarily for remotely managing and automating tasks on instances and other resources across your AWS environment.

- Use AWS OpsWorks if:
    - You need a managed solution for deploying and managing Chef or Puppet-based applications.
    - You want pre-built stacks and tools for easier application deployments.
    - You require scaling capabilities for your applications.
- Use AWS SSM if:
    - You prioritize remote management and automation across your entire environment.
    - You need a flexible solution for running commands and scripts on diverse resources.
    - You want centralized resource management and integration with other AWS services.

Additional Considerations:

- Cost: AWS OpsWorks charges for Chef and Puppet licenses, while SSM has a pay-per-use model for document executions and parameter store usage.
- Complexity: AWS OpsWorks comes with a steeper learning curve due to its focus on Chef and Puppet configuration management.
- Integration: AWS SSM offers wider integration with other AWS services compared to OpsWorks.

Service Catalog

It is used to manage a catalog of IT services used within the most enterprises.

Parameter Store

- AWS Systems Manager provides a centralized store to manage your configuration data, whether plain-text data such as database strings or secrets such as passwords.

- This allows you to separate your secrets and configuration data from your code. Parameters can be tagged and organized into hierarchies, helping you manage them more easily.
- Systems Manager is integrated with AWS Key Management Service (KMS), allowing you to automatically encrypt the data you store.
- You can also control user and resource access to parameters using AWS Identity and Access Management (IAM). Parameters can be referenced through other AWS services such as Amazon Elastic Container Service, AWS Lambda, and AWS CloudFormation.

Trusted Advisor

Trusted Advisor is an online resource to help you reduce cost, increase performance, and improve security by optimizing your AWS environment. Trusted Advisor provides real-time guidance to help you provision your resources following AWS best practices.
AWS Trusted Advisor offers a Service Limits check (in the Performance category) that displays your usage and limits for some aspects of some services.

Trusted Advisor vs Inpector

- Use AWS Trusted Advisor if:
    - You prioritize cost optimization and resource efficiency.
    - You want high-level recommendations for improving your overall AWS usage.
    - You prefer a free and easy-to-use tool.
- Use AWS Inspector if:
    - You prioritize security and want to proactively identify vulnerabilities.
    - You need detailed troubleshooting and actionable recommendations.
    - You require automation for vulnerability remediation.

AWS Compute Optimizer

It is used for optimizing (performance) use of compute resources (EC2 - not all instance types are supported, EC2 Auto Scaling Groups, Lambda Functions, EBS). It looks at current usages in CloudWatch logs and gives recommendations based on that. CloudWatch should be enabled, it needs 30 consecutive hours of CloudWatch metric data for giving recommendations (except lambda). For lambda, configuration memory should be less than 1792 MB and functions should be invoked 50 times in the last 14 days.

Security improvements are included in the free tier. With payment, you can receive recommendations for performance and cost improvements.

AWS Security Hub

It is a paid subscription. It runs automatic checks to scan for compliance with regulations and laws.

Security Hub vs Inspector

- Use Security Hub if:
    - You need a comprehensive security management solution across your AWS environment.
    - You want to aggregate findings from multiple sources and prioritize them.
    - You require compliance reporting and auditing capabilities.
- Use Inspector if:
    - You want to focus specifically on vulnerability scanning of EC2 instances and container images.
    - You need detailed reports on vulnerabilities and configuration issues.

Amazon GuardDuty

It is an intrusion detection system. Uses threat intelligence feeds and machine learning to identify threats. It is a threat detection service that continuously monitors for malicious activity and anomalous behavior. It does not scan for vulnerabilities.

GuardDuty Terms

Account

AWS account.

Detector

An entity that detects threats is region specific.

Data Source

- CloudTrail data
- S3 logs
- VPC flow logs
- DNS logs
- EBS volume data
- EKS audit logs

Finding

Possible security issue discovered during GuardDuty analysis.

Scan Options

Additional scanning options like malware scanning etc.

Suppression Rule

It lets you define specific attributes to tell GuardDuty that it is not a problem, GuardDuty will log information related to that attribute but will not send notifications.

Trusted IP List

You can define a list of trusted IPs to tell GuardDuty that requests originating from these IPs are not a threat.

Developer Tools / DevOps

CodeStar

Unified interface for managing the entire software development lifecycle, from project setup to deployment.

CodeCommit

Used for linking code repositories in the cloud.

CodeBuild

It builds and tests code in the cloud.

CodeDeploy

A deployment service that deploys your code onto an EC2 instance, lambda, S3 bucket, etc…

CodePipeline

It is a continuous delivery service, used to visualize and automate the steps required to release code.

Cloud9

It is a cloud based IDE that can be used with a browser. It comes with a linux shell terminal.

Miscellaneous Services

Media Services

Amazon Elastic Transcoder

Re-codes a particular media format to suit other media formats.
Amazon Elastic Transcoder is a highly scalable, easy to use, and cost-effective way for developers and businesses to convert (or “transcode”) video and audio files from their source format into versions that will playback on devices like smartphones, tablets, and PCs.
It supports a wide range of input and output formats, resolutions, bitrates, and frame rates. It also offers features for automatic video bit rate optimization, generation of thumbnails, overlay of visual watermarks, caption support, DRM packaging, progressive downloads, encryption, and more.

The Elastic Transcoder picks up files from an input S3 bucket and saves the output to an output S3 bucket. You are charged based on the duration of the content and the resolution or format of the media.

MediaTailor

MediaTailor provides the ability to insert individually targeted advertising into their video streams.

MediaLive

MediaLive is a live video processing service that enables the creation of high-quality video streams that can be broadcasted to TVs and Internet-connected devices.

Media Package

Packages and protects customers’ video and distributes content to a wide range of video playback devices.

MediaConvert

Broadcasts (multi screen display) video-on-demand content after transcoding it.

MediaStore

Video origination and storage service. It has

- Containers
- Folders
- Endpoint
- Object (Video)

- Policy

Use Cases

- Live Video Streams (It can be used as an origination endpoint)
- It is not well suited for storage based video (video on demand) as S3 buckets are good for this use case.

Machine Learning Services

SageMaker

A fully managed platform that enables developers and data scientists to train and deploy machine learning models.

Polly

Turns text into lifelike speech, allowing you to create applications that talk, and build entirely new categories of speech-enabled products.

Translate

Is a neural machine translation service for language translation.

Comprehend

Is a NLP service that is used to discover insights and relationships in a text.

Rekognition API

Allows you to automatically identify objects, people, text, scenes and activities as well as to detect any inappropriate content. Can be run against S3 buckets.

Transcribe

Is an automatic speech recognition service (ASR). Returns a text file containing the transcription of the audio.

Tensor Flow

A framework for deep learning research and development.

Lex

Provides advanced deep learning functionalities for ASR and NLU (Natural Language Understanding) to recognize intent of text. Useful for developing engaging applications. It can be used for building conversational interfaces into any application using voice and text.

Machine Learning

Provides visualization tools and wizards that guides you through the process of creating machine learning models.

Analytics Services

CloudSearch

A service that allows you to bring offline data to AWS Cloud and perform analysis on it.

Amazon ElasticSearch

Amazon Elasticsearch Service is a fully managed service that makes it easy for you to deploy, secure, operate, and scale Elasticsearch to search, analyze, and visualize data in real-time.

Search capability can be increased by scaling out instances. Number of instances in ElasticSearch cluster can be increased or decreased as needed.

Use Cases:

- Log Management and Analysis: Analyze application logs, web server logs, and system logs in real-time for troubleshooting, performance monitoring, and security insights.
- Search and Recommendations: Power search functionalities for e-commerce platforms, content websites, and internal knowledge bases.
- Real-time Analytics: Analyze real-time data streams from IoT devices, social media, and other sources for immediate insights and decision support.
- Application Monitoring: Monitor application performance and identify issues in real-time for proactive troubleshooting.
- Fraud Detection and Security: Analyze security logs and user behavior patterns to detect suspicious activity and prevent fraud.

Athena

Used for running SQL queries for big data analytics. Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run.

Athena is easy to use — simply point to your data in Amazon S3, define the schema, and start querying using standard SQL. It uses Presto with full standard SQL support. While Amazon Athena is ideal for quick, ad-hoc querying and integrates with Amazon QuickSight for easy visualization, it can also handle complex analysis including large joins, window functions, and arrays.

Athena allows you to easily query encrypted data stored in Amazon S3 and write encrypted results back to your S3 bucket. Both server-side encryption and client-side encryption are supported.

Amazon Athena uses a managed Data Catalog to store information and schemas about the databases and tables that you create for your data stored in Amazon S3. With Athena, there's no need for complex ETL jobs to prepare data for analysis. This makes it easy for anyone with SQL skills to analyze large-scale datasets quickly.

With Amazon Athena, you don't have to worry about managing or tuning clusters to get fast performance. It is optimized for fast performance with Amazon S3. Athena automatically executes queries in parallel so that you get query results in seconds, even on large datasets. This allows for most results to be delivered within seconds.

Athena is out-of-the-box integrated with AWS Glue Data Catalog. This allows you to create a unified metadata repository across various services; crawl data sources to discover schemas; populate your Catalog with new and modified table and partition definitions; and maintain schema versioning.

Pricing

- With Amazon Athena, you pay only for the queries that you run.
- You are charged based on the amount of data scanned by each query.
- You can get significant cost savings and performance gains by compressing, partitioning, or converting your data to a columnar format because each of those operations reduces the amount of data that Athena needs to scan to execute a query.

EMR - Elastic Map Reduce

What it is:

- A managed Hadoop framework service on AWS for processing vast amounts of data.
- Simplifies running big data frameworks like Apache Hadoop, Spark, Hive, and Presto.
- Manages infrastructure provisioning, configuration, and cluster scaling.
- Offers flexible options for cluster size, instance types, and software versions.

Key Features:

- Managed Hadoop Environment: Eliminates the need to set up and maintain Hadoop infrastructure manually.
- Choice of Frameworks: Supports various big data frameworks beyond Hadoop, including Spark, Hive, Presto, Flink, and HBase.
- Scalability: Easily scale your cluster up or down to match changing workloads.
- Integration with AWS Services: Seamlessly integrates with other AWS services like S3 for data storage, DynamoDB for NoSQL databases, and CloudWatch for monitoring.
- Cost-Effective: Pay only for the resources you use, with options for spot instances and reserved instances.

- Security: Includes encryption and access control features to protect sensitive data.

Common Use Cases:

- Data Processing and Analysis:
    - Log processing and analysis
    - Data warehousing and analytics
    - Machine learning and predictive modeling
    - Clickstream analysis
    - Genomics research
    - Financial modeling
- ETL (Extract, Transform, Load):
    - Moving and transforming large datasets between different systems
- Data Science and Machine Learning:
    - Training and deploying machine learning models on large datasets

Benefits:

- Simplified Big Data Handling: Makes it easier to process and analyze massive datasets without managing complex infrastructure.
- Scalability: Handles growing data volumes and computational needs seamlessly.
- Cost-Effective: Provides a cost-efficient way to run big data workloads in the cloud.
- Faster Time to Insights: Enables rapid data processing and analysis for quicker decision-making.
- Secure Data Handling: Protects sensitive data with encryption and access controls.

Additional Considerations:

- Learning Curve: Apache Hadoop and related frameworks have a learning curve.
- Optimization: Requires expertise to optimize for performance and cost.
- Alternatives: Consider serverless options like AWS Glue or Athena for less complex workloads.

EMR Cluster Nodes

Master node - Coordinates job distribution across one and task nodes.
Core node - Runs tasks assigned by the master node. Stores data in the cluster.
Task node - Runs only task that do not store data.

EMR vs ElasticSearch

Purpose:

- EMR: Batch processing and analytics of massive datasets using Hadoop and related frameworks.
- Elasticsearch: Real-time search, analysis, and visualization of large volumes of data.

Functionality:

- EMR: Distributes and parallelizes computations across multiple nodes for large-scale data processing.
- Elasticsearch: Indexes and stores data for fast retrieval and flexible querying, enabling real-time insights.

Querying: EMR uses SQL-like languages (Hive, Spark SQL), while Elasticsearch uses a RESTful API and its query language.

Visualization: EMR often integrates with BI tools for visualization, while Elasticsearch often uses Kibana for interactive dashboards.

Amazon Quick Sight

A BI service used to build visualization and get business insights from the data. It does not come with an AWS subscription, you would have to sign up separately.

Data Pipeline

Used to move data across various AWS services. Don't need to worry about availability of resources and sending notifications in case of failures or success. This also offers various ETL services. It can not be used as a replacement for AWS Glue as it can be used to automate the movement and transformation of data; it relies on other services to actually transform the data.

AWS Glue

A fully managed ETL service that makes it easy for customers to prepare and load data for analytics. AWS Glue is a fully-managed, pay-as-you-go, extract, transform, and load (ETL) service that automates the time-consuming data preparation steps for analytics. AWS Glue automatically discovers and profiles data via the Glue Data Catalog. It also recommends and generates ETL code to transform your source data into target schemas.

AWS Glue runs the ETL jobs on a fully managed, scale-out Apache Spark environment to load your data into its destination. AWS Glue also allows you to set up, orchestrate, and monitor complex data flows.

AWS Glue crawlers connect to a source or target data store, progress through a prioritized list of classifiers to determine the schema for the data, and then create metadata in the AWS Glue Data Catalog. The metadata is stored in tables in a data catalog and used in the authoring process of ETL jobs. You can run crawlers on a schedule, on-demand, or trigger them based on an event to ensure that your metadata is up-to-date.

AWS Glue automatically generates the code to extract, transform, and load data. The code is generated in Scala or Python and written for Apache Spark. AWS Glue also helps clean and prepare data for analysis by providing a Machine Learning Transform called FindMatches for deduplication and finding matching records.

- Use AWS Glue to discover properties of data, transform it, and prepare it for analytics.

- Glue can automatically discover both structured and semi-structured data stored in data lakes on Amazon S3, data warehouses in Amazon Redshift, and various databases running on AWS.
- It provides a unified view of data via the Glue Data Catalog that is available for ETL, querying, and reporting using services like Amazon Athena, Amazon EMR, and Amazon Redshift Spectrum.
- AWS Glue is serverless, so there are no compute resources to configure and manage.

Comparison

AWS Service	Primary Use Case	When to use
Amazon Athena	Query	Run interactive queries against data directly in Amazon S3 without worrying about formatting data or managing infrastructure. Can use with other services such as Amazon RedShift
Amazon RedShift	Data Warehouse	Pull data from many sources, format and organize it, store it, and support complex, high speed queries that produce business reports.
Amazon EMR	Data Processing	Highly distributed processing frameworks such as Hadoop, Spark, and Presto. Run a wide variety of scale-out data processing tasks for applications such as machine learning, graph analytics, data transformation, streaming data. Amazon EMR is a web service that enables businesses, researchers, data analysts, and developers to easily and cost-effectively process vast amounts of data. EMR utilizes a hosted Hadoop framework running on Amazon EC2 and Amazon S3.

AWS Glue	ETL Service	Transform and move data to various destinations. Used to prepare and load data for analytics. The data source can be S3, RedShift, or other databases. Glue Data Catalog can be queried by Athena, EMR, and RedShift Spectrum

Customer Engagement Services

Connect

It is a cloud-based contact center as a service that is managed by aws.

Simple Email Service

It is a SMTP-based email service. Think of this as a way to send all your transactional emails.

Business Productivity Services

Alex for Business

Executive assistant (start meetings, book your calendar, book meeting rooms, etc).

Chime

Similar to Zoom, it is an audio/video conference application.

WorkDocs

You can treat it as dropbox in the AWS ecosystem. Helps in storing, securing and sharing your documents.

WorkMail

Secure and managed business email and calendar services with support for existing desktop and mobile email client applications.

WorkDocs + WorkMail = Office365

Desktop and App Stream Services

AppStream 2.0

Fully managed secure application streaming service that allows a user to stream an application deployed in AWS to any device running a web browser. This helps in centrally managing your applications and not having to install applications on end-user machines.

AWS Workspaces

A fully managed, desktop as a service that runs on AWS. You can provision virtual cloud-based Microsoft Windows desktops for your use, providing them access to the documents, applications , and resources they need. Persistent storage can be made only in D: drive

IOT

IOT Core

AWS IoT Core is a managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices. AWS IoT Core can support billions of devices and trillions of messages. It can process and route those messages to AWS endpoints and to other devices reliably and securely.

IOT Device Management

Makes it easy to securely onboard, organize, monitor and remotely manage IOT devices at scale.

Amazon RTOS

This is an OS for microcontrollers, that make small, low power edge devices easy to program, deploy, secure, connect, and manage.

Green grass

Lets you run local compute, messaging, data caching sync for connected devices.

Game Lift

It is a service that provides a platform to build and develop games.

Mobile Services

Mobile Hub

A management console for all your mobile needs. It generates cloud configuration files, which stores information about configured following services .

PinPoint

It is used for sending targeted push notifications for mobile users.

Device Farm

Used for testing applications across several devices and operating systems.

Mobile Analytics

A mobile analytics service for the mobile.

AR/VR

Sumerian

Helps in creating VR, AR and 3D applications without requiring any specialized programming or 3D graphics expertise.

Questions

How to launch EC2 instances in another AWS Region in the event of a disaster ?

1. Create AMIs of the instances and copy them to another region.
2. Launch instances in the second region from the AMIs.

Which of the following Services can be used to decouple an architecture (Select Three) ?

- ELB (correct)
- Auto Scaling (incorrect - because it copies same instances)
- SQS (correct)
- SNS (correct)

You are asked to improve the performance of an image processing application that uses Amazon SQS and EC2 instances as consumers. High volumes of traffic causes message backlogs in SQS. What do you do ?

- Purchased Dedicated instances (incorrect)
- Convert to SQS FIFO queues (incorrect)
- Create an AWS Lambda Function to scale out # of consumer instances when backlog grows (incorrect - technically correct answer but option # 4 takes precedence because of auto scaling)
- Configure an Auto-Scaling group based on the ApproximateNumberOfMessages Amazon CloudWatch metric (correct)

You are required to design an online application running in a VPC on EC2 instances behind an ELB. The application tier must read and write data to customer-managed database cluster. There should be no access to the database from the internet, but the cluster must be able to obtain software patches from the internet. Which solution meets these requirements ?

- Public subnets for both the application tier and the database cluster (incorrect).
- Public subnets for the application tier, and private subnets for the database cluster (incorrect).
- Public subnets for the application tier and NAT Gateway, and private subnets for the database cluster (correct).
- Public subnets for the application tier, and private subnets for the database cluster and NAT Gateway (incorrect - because NAT Gateway in a private subnet will not have a public IP address and cannot route to the internet).

A company has updated its security policy to enforce encryption on all EBS volumes. There are multiple existing volumes that need to be encrypted under this policy ? What combination of steps should the solutions architect take in order to fulfill this requirement ? (Any Two)

- Convert existing volume to EBS encrypted type
- Create a snapshot and select the encrypted option while creating a copy of the snapshot. (correct)
- Create a new encrypted volume and copy the data to the new volume (could be correct but it's a tedious process as data needs to be copied) .
- Create a new volume from the encrypted snapshot. (Correct - Volume created from encrypted snapshot will be encrypted volume)
- Create an encrypted snapshot from the existing volume.

A mobile company has migrated their Over-the-air (OTA) software update application to AWS. They have currently deployed a fleet of EC2 instances to allow the delivery of the software updates. The average size of the software is around 2GB. The current architecture has a very high bandwidth cost due to the large number of downloads. As a solutions architect, what architectural change can you implement to save the bandwidth cost ?

- Create a DynamoDB index for the download file with DynamoDB Accelerator and store the download files in S3 (incorrect - possible solution for the given use case but not cost effective).
- Place the files in a S3 bucket and create a CloudFront Distribution to deliver the file (correct).
- Place the files in a public S3 bucket and use the S3 link to deliver the file.
- Replace the current instances with storage-optimized instances to reduce cost.

Which services use edge locations by default ?

- Amazon CloudFront
- AWS WAF
- Amazon Route 53
- All of the above (correct - they all can be part of Content Distribution Network).

A company has acquired a new company and needs to migrate all the applications to AWS. Each application has approx. 50 TB of data to be transferred. Both the companies require a consistent high speed connectivity between their data centers and AWS after the migration. A solutions architect must ensure one-time data migration and ongoing network connectivity. Which solutions will meet these requirements ?

a) AWS Snowball for the initial transfer and AWS Direct Connect for ongoing connectivity.
b) AWS Site-toSite VPN for both the initial transfer and ongoing connectivity.
c) AWS Snowball for the initial transfer and AWS Site-to-Site VPN for ongoing connectivity.
d) AWS Direct Connect for both the initial transfer and ongoing connectivity.

Correct - (a)

A media company has migrated its media application that serves content to its subscribers globally to AWS. The application serves the content through multiple Amazon EC2 instances in a private subnet behind and ALB. Due to the copyright restrictions, the content team wants to block access for certain countries. Which actions will meet these requirements ?

a) Modify the ALB security group to deny incoming traffic from blocked countries.
b) Modify the security group for EC2 instances to deny incoming traffic from blocked countries.
c) Use ALB listener rules to return access denied response to incoming traffic from blocked countries.
d) Use Amazon CloudFront to serve the application and deny access to blocked countries.

Correct - (d)

A finance company is facing poor write performance on its newly migrated RDS MySQL Database. On investigation, it was found that these performance issues were caused by users generating different real-time reports from the application during working hours. Which solution will you implement to improve the performance of the application when its moved to AWS ?

a) Create an Amazon Aurora MySQL Multi-AZ DB Cluster. Configure the application to use the backup instance of the cluster as an endpoint for the reports.
b) Import the data into Amazon DynamoDB table with provisioned capacity. Refactor the application to use DynamoDB table with provisioned capacity.

c) Create an Amazon Aurora MySQL Multi-AZ DB cluster with multiple read replicas. Configure the application to use the reader endpoint for reports.
d) Create the database on a compute optimized Amazon EC2 instance. Ensure compute resources exceed the on-premise databases.

Correct - c) . (a) is incorrect because backup instance should only be used for backup.

A company has over 2000 users and is planning to migrate data into the AWS Cloud. Some of the data is the users' home folders on an existing file share, and the plan is to move this data to Amazon S3. Each user will have a folder in a shared bucket under this folder structure: bucket/home/%username%. What steps should a solutions architect take to ensure that each user can access their own home folder and no one else's?

1. Create an IAM policy that applies folder-level permissions.
2. Create an IAM group and attach the IAM policy. Add IAM users to the group.

How to create an encrypted read replica in a separate region, if the master database is not encrypted ?

Encrypt a snapshot from the master DB instance, create a new encrypted master DB instance, and then create an encrypted Cross-Region read replica.

How to migrate large data from on-premise to AWS with AWS DMS ?

Larger data migrations with AWS DMS (Database Migration Service) can include many terabytes of information. This process can be cumbersome due to network bandwidth limits or just the sheer amount of data. AWS DMS can use Snowball Edge and Amazon S3 to migrate large databases more quickly than other methods.
When you are using an Edge device, the data migration process has the following stages:

- You use the AWS Schema Conversion Tool (AWS SCT) to extract the data locally and move it to an Edge device.
- You ship the Edge device or devices back to AWS.
- After AWS receives your shipment, the Edge device automatically loads its data into an Amazon S3 bucket.
- AWS DMS takes the files and migrates the data to the target data store. If you are using change data capture (CDC), those updates are written to the Amazon S3 bucket and then applied to the target data store.

How to migrate MongoDB on-premise to Amazon DynamoDB on Cloud ?

Use the Schema Conversion Tool (SCT) to extract and load the data to an AWS Snowball Edge device (Use of SCT is necessary for extracting and loading to an edge device). Use the AWS Database Migration Service (DMS) to migrate the data to Amazon DynamoDB.

How to reduce the amount of scaling events ?

- Modify the CloudWatch alarm period that triggers your Auto Scaling scale down policy.
- Modify the Auto Scaling group cool-down timers. The cooldown period is a configurable setting for your Auto Scaling group that helps ensure that it doesn't launch or terminate additional instances before the previous scaling activity takes effect, so using it would help.
- The CloudWatch Alarm Evaluation Period is the number of the most recent data points to evaluate when determining alarm state. This would help as you can increase the number of data points required to trigger an alarm.

A web application that runs on Amazon EC2 instances behind an Elastic Load Balancer. All data in transit must be encrypted ?

Use a Network Load Balancer (NLB) with a TCP listener, and then terminate SSL on EC2 instances. You can pass through encrypted traffic with an NLB and terminate the SSL on the EC2 instances. You cannot use a HTTPS listener with an NLB.

OR

Use an Application Load Balancer (ALB) with an HTTPS listener, and then install SSL certificates on the ALB and EC2 instances. You can use an HTTPS listener with an ALB and install certificates on both the ALB and EC2 instances. This does not use passthrough; instead, it will terminate the first SSL connection on the ALB and then re-encrypt the traffic and connect to the EC2 instances. You cannot use passthrough mode with an ALB and terminate SSL on the EC2 instances. You cannot use a TCP listener with an ALB.

How to configure load balancing to distribute traffic across both AWS and on-premise resources ?

- Provision a Direct Connect connection between your on-premises location and AWS. Create a target group on an ALB to use IP based targets for both your EC2 instances and on-premises servers. Using IP addresses as targets allows load balancing any application hosted in AWS or on-premises using IP addresses of the application back-ends as targets.
- You must have a VPN or Direct Connect connection to enable this configuration to work.
- You cannot use instance ID-based targets for on-premises servers, and you cannot mix instance ID and IP address target types in a single target group.

How to avoid the internet when you have AWS Direct Connect Connection from on-premise to Amazon VPC ?

Private virtual interface across the Direct Connect connection to connect to the VPC using private IP addresses. A VPC endpoint for Amazon API Gateway can be created, which will provide access to API Gateway using private IP addresses and will avoid the Internet completely.

How to increase resiliency of Direct Connect connection at low-cost ?

Implementing an IPSec VPN connection and using the same BGP (Border Gateway Protocol) prefix is the most cost-effective solution. With this option, both the Direct Connect connection and IPSec VPN are active and being advertised using the BGP. The Direct Connect link will always be preferred unless it is unavailable.

Implementing a second Direct Connect connection takes time to implement and is also costly.

How to restrict access to the private content stored in S3 Bucket and shared via CloudFront. Users from a specific IP address should be able to access the content and ensure direct access via the S3 bucket is not possible ?

- Configure CloudFront to require users to access the files using a signed URL, create an Origin Access Identity (OAI), and restrict access to the files in the Amazon S3 bucket to the OAI.
- A signed URL includes additional information (for example, expiration date and time) that gives you more control over access to your content. You can also specify the IP address (or range of IP addresses) of the users who can access your content.

What if public and private hosted zones that have overlapping namespaces?

Amazon EC2 confirms the private hosted zone matches the domain name request.
Match defined as:

- Identical match
- Private hosted zone is a parent of the domain name in the request

Example: accounting.example.com and example.com match as hosted zones because they are parents of seattle.accounting.example.com

When dealing with public and private hosted zones that have overlapping namespaces, it's important to understand how Route 53 prioritizes and resolves DNS queries. Here's what you need to know:

Prioritization:

- Route 53 resolves DNS queries based on the most specific match. This means that if both a public and private hosted zone have records for the same domain name, the record in the more specific zone will be used.
- Specificity is determined by the longest domain name. For example, "example.com" is more specific than "com".

Resolution scenarios:

1. Public record exists, no private record: The public record will be used for all DNS queries.
2. Private record exists, no public record: The private record will be used only for DNS queries originating from within the VPC associated with the private hosted zone. Queries originating from outside the VPC will resolve to an NXDOMAIN error.

3. Both public and private records exist: The most specific record will be used, based on the above rules.

Consider these additional points:

- Private hosted zones can only be associated with specific VPCs.
- Private hosted zones take precedence over public hosted zones when both have records for the same domain name and the query originates from within the associated VPC.
- You can use Route 53's Resolver service to configure how DNS queries are routed within your VPC, including specifying which hosted zones are used for resolution.

A company is deploying a new application that will consist of an application layer and an online transaction processing (OLTP) relational database. The application must be available at all times. However, the application will have periods of inactivity. The company wants to pay the minimum for compute costs during these idle periods.

Which solution meets these requirements MOST cost-effectively?

- Deploy the application on Amazon EC2 instances in an Auto Scaling group behind an Application Load Balancer. Use Amazon RDS for MySQL for the database.
    - Incorrect. With this solution, at least one instance and a database will run during the periods of inactivity. This solution does not minimize cost during the periods of inactivity. This solution is not the most cost-effective option.
- Run the application in containers with Amazon Elastic Container Service (Amazon ECS) on AWS Fargate. Use Amazon Aurora Serverless for the database.
    - Correct. When Amazon ECS uses Fargate for compute, it incurs no costs when the application is idle. Aurora Serverless also incurs no compute costs when it is idle.

A company is transitioning its Amazon EC2 based MariaDB database to Amazon RDS. The company has already identified a database instance type that will meet the company's CPU and memory requirements. The database must provide at least 40 GiB of storage capacity and 1,000 IOPS.

Which storage configuration for the Amazon RDS for MariaDB instance is MOST cost-effective?

- Provision 50 GiB of General Purpose SSD storage for the RDS instance.

    - Incorrect. Baseline I/O performance for General Purpose SSD storage is 3 IOPS for each GiB, with a minimum of 100 IOPS. For 50 GiB of storage, the baseline performance would be 150 IOPS.
- Provision 334 GiB of General Purpose SSD storage for the RDS instance.
    - Correct. Baseline I/O performance for General Purpose SSD storage is 3 IOPS for each GiB. For 334 GiB of storage, the baseline performance would be 1,002 IOPS. Additionally, General Purpose SSD storage is more cost-effective than Provisioned IOPS storage.

A company asks a solutions architect to implement a pilot light disaster recovery (DR) strategy for an existing on-premises application. The application is self contained and does not need to access any databases.

Which solution will implement a pilot light DR strategy?

- Back up the on-premises application, configuration, and data to an Amazon S3 bucket. When the on-premises application fails, build a new hosting environment on AWS and restore the application from the information that is stored in the S3 bucket.
    - Incorrect. This is a backup and restore DR strategy. Backup and restore DR strategies typically have the lowest cost but highest recovery time. A solution that manually rebuilds the hosting infrastructure on AWS could take hours.
- Recreate the application hosting environment on AWS by using Amazon EC2 instances and stop the EC2 instances. When the on-premises application fails, start the stopped EC2 instances and direct 100% of application traffic to the EC2 instances that are running in the AWS Cloud.
    - Correct. This is a pilot light DR strategy. This solution recreates an existing application hosting environment in an AWS Region. This solution turns off most (or all) resources and uses the resources only during tests or when DR failover is necessary. RPO and RTO are usually 10s of minutes. A pilot light strategy simplifies recovery at the time of a disaster because the core infrastructure requirements are all in place. A pilot light strategy also minimizes the ongoing cost of DR by minimizing the active resources.
- Recreate the application hosting environment on AWS by using Amazon EC2 instances. Direct 10% of application traffic to the EC2 instances that are running in the AWS Cloud. When the on-premises application fails, direct 100% of application traffic to the EC2 instances that are running in the AWS Cloud.
    - Incorrect. This is a warm standby DR strategy. This solution recreates an existing application hosting environment in an AWS Region. This solution serves a portion of live traffic. With this DR strategy, RPO and RTO are usually a few minutes. However, costs are higher because this solutions runs resources continuously.

A company is designing a disaster recovery (DR) architecture for an important application on AWS. The company has determined that the RTO is 5 minutes with a minimal instance capacity to support the application in the AWS DR site. The company needs to minimize costs for the DR architecture.

Which DR strategy will meet these requirements?

- Warm standby
    - Correct. This solution meets the requirement for an RTO of 5 minutes. The instances run at a low capacity and can scale within minutes.
- Pilot Light
    - Incorrect. This solution would not meet the requirement for an RTO of 5 minutes. The instances are idle and unable to run and scale to full capacity within 5 minutes.
- Multi-site active-active
    - Incorrect. Because this is an active-active environment, this solution would address the requirement for an RTO within moments. The services are already running at full capacity within that time. However, this solution costs more than is necessary to meet the company's requirements.
- Backup and restore
    - Incorrect. This solution would not achieve the RTO of 5 minutes that the company requires.

A Solutions Architect receives a large number of video files to upload to an Amazon S3 Bucket. The files sizes are 100 - 500 MB. The solutions architect wants to easily resume failed upload attempts. How should solutions architect perform the uploads in the LEAST amount of time ?

- Split each file into 5 MB parts. Upload the individual parts normally and use S3 multipart upload to merge the parts into a complete object.
    - Incorrect. Multipart is recommended for Objects over 100 MB, so splitting into 5 MB is unnecessary.
- Using the AWS CLI, copy individual objects into Amzon S3 bucket with the aws s3 cp command.
    - Correct. Copy command in AWS CLI automatically performs multipart uploading and downloading based on objects size.

Note: Multipart upload is not initiated using management console.Using SFTP and AWS Transfer family will not solve a problem of re-uploading failed upload attempts.

A company is developing an application that runs on EC2 instances in a private subnet. The EC2 instances use a NAT gateway to access the internet. A solutions architect must provide a secure option so that developers can log in to the instances ?

Which solution meets the requirements Most cost-effectively ?

- Configure AWS Systems Manger Session Manager for the EC2 instances to activate login.
    - Correct. Session Manager gives you the ability to set up secure and audible instat management without the need to open inbound port, maintain a bastion host or manage SSH Keys. There is no additional charge for accessing EC2 instances using Session Manager.
- Configure a bastion host in a public subnet to log in to the EC2 instances in a private subnet.
    - Incorrect. This is a possible way but since bastion host is also a EC2 instance so it would incur cost.
- Use the existing NAT gateway to log in to the EC2 instances in a private subnet.
    - Incorrect. Not possible as NAT gateway only allows egress traffic not ingress.
- Configure AWS Site-to-Site VPN to log in directly to the EC2 instances.
    - Incorrect. Possible but VPN connections do incur cost.

Your company uses Amazon Route 53 in their networking account to manage public hosted zone records for their root domain, example.com. A developer in a different account requires the ability to create, update, or delete DNS records for their public application, app1.

1. Create the app1.example.com Public Hosted Zone in the developer's account.
2. Create an NS record in the example.com Public Hosted Zone (in the networking account) for the subdomain with the name servers from the app1.example.com Public Hosted Zone.

Explanation:

- Delegation of Subdomain: By creating a separate public hosted zone for the subdomain (app1.example.com) in the developer's account, you grant them full control over DNS records within that subdomain without affecting the root domain.

- NS Record for Delegation: The NS record in the root domain's hosted zone acts as a pointer, directing queries for the subdomain to the developer's hosted zone, where they manage the records autonomously.

Your company uses an Identity Provider (IdP) for Single-sign on (SSO) and has tasked their solutions architect with connecting their AWS Account to the IdP so their users can leverage their corporate identity to access the environment.

- Create an AWS IAM Identity Provider by uploading the SAML metadata document from your IdP.
- Create an AWS IAM Role with a trust relationship with the IdP.

A company runs a batch application in the AWS Cloud hosted on 200+ Amazon EC2 instances. As a solutions architect, you are asked to push debug logs to an Amazon S3 bucket every 2:00 AM for all the EC2 instances.

What is the best possible solution from an operation point of view?

- Use Systems Manager Distributor to transfer the logs every 2:00 AM on all the AWS Systems Manager Managed instances.
- Use SSM Session Manager to run a shell script on all Amazon EC2 instances 2:00 AM in the morning.
- Inject a user script via Ops Work to all of the Amazon EC2 instances that will push the logs to this Amazon S3 bucket.
- Create a schedule in AWS Systems Manager Maintenance window to move the logs to S3 bucket every 2:00 AM in the morning.
    - Comments: This is the correct answer as SSM Maintenance window is used to schedule patching and some automation workflows.

Your company has a private Amazon VPC in their AWS account that cannot be connected to the internet due to data sensitivity concerns. A solutions architect needs a way to interactively troubleshoot an Amazon EC2 Instance running in this VPC.

What should the solutions architect do in order to gain access to the Amazon EC2 instance, without provisioning any type of internet connectivity?

1. Attach an AWS IAM Instance Profile with SSM Permissions:
    - Create an IAM instance profile that grants the necessary permissions for SSM Session Manager to interact with the EC2 instance.
    - Attach this instance profile to the EC2 instance.
2. Create Amazon VPC Interface Endpoints for SSM, SSMMessages, and EC2Messages:
    - Create VPC interface endpoints for the following services within the private VPC:
        - SSM: Enables access to SSM services for management tasks.
        - SSMMessages: Facilitates communication between SSM agents and the SSM service.
        - EC2Messages: Enables communication between EC2 instances and the SSM service.

As a cloud architect, you've been tasked with finding a way to restrict access to a specific content for all users in a specific country.
Which of the following services could be used? (Select TWO)

- AWS Network Firewall
- Amazon Route 53
- Amazon CloudFront (correct)
- AWS Shield
- AWS WAF (correct)

Comments: With CloudFront, you would create a distribution and use CloudFront's geo-blocking feature to restrict access at the country level. Also, AWS WAF can be used to restrict access to a CloudFront distribution at the country-level. Alternatively, Amazon Route 53, can be used. However that would re-route requests from a restricted geography, for example, to an error message page. References: Restricting the geographic distribution of your content. Using AWS WAF to control access to your content 3 Ways to Geo-Restrict your App.

Your company uses an Amazon EC2 instance in a public subnet to host a web application. The Amazon EC2 instance uses a default security group. The network ACL is configured to block all traffic to the instance. You must allow incoming traffic on port 443 to access the application from any source.

Which combination of steps will accomplish this requirement? (Select TWO)

- In the network ACL, add a rule to allow outbound TCP traffic on port 1024 - 65535 to destination 0.0.0.0/0
- In the security group, add a rule to allow TCP connections on port 443 from the source 0.0.0.0/0 (Correct)
    - Comments: Security groups are stateful, so if you allow traffic in, then the security group automatically allows that traffic back out. So you only need a rule for incoming traffic from port 443.
- In the security group, add a rule to allow TCP connections on port 443 to destination 0.0.0.0/0
- In the network ACL, add a rule to allow inbound TCP traffic on port 443 from source 0.0.0.0/0 and outbound traffic on port 1024-65535 to the destination 0.0.0.0/0 (Correct)
    - Comments: This is correct, because your network ACL need an outbound connection to the ephemeral ports.
- In the network ACL add rules to allow both inbound and outbound TCP connection on port 443 from source 0.0.0.0/0 and destination 0.0.0.0/0

Your CEO decided to migrate your data center to AWS. You are engaged as the AWS migration specialist to create a business case and decide to use AWS Migration Evaluator.

Which of the following are included in the business case report? (Select THREE)

- Recommend AWS services required for your target architecture. (Correct)
- Provide configuration data about your on-premises servers.
- Recommendation for the customer on next steps for a successful migration.
- A breakdown of what went into the on-premises costs. (Correct)
- Recommendation on how to automatically converting your source servers from physical, virtual, or cloud infrastructure to run natively on AWS
- An executive summary of the savings across a combination of scenarios applied to different workloads. (Correct)