**Module 1 : Introduction**

* client-server model

In computing, a**client** can be a web browser or desktop application that a person interacts with to make requests to computer servers. A **server** can be services such as Amazon Elastic Compute Cloud (Amazon EC2), a type of virtual server.

* Cloud Computing

When selecting a cloud strategy, a company must consider factors such as required cloud application components, preferred resource management tools, and any legacy IT infrastructure requirements.

The three cloud computing deployment models are **cloud-based**, **on-premises**, and **hybrid**.

**cloud-based deployment**

* Run all parts of the application in the cloud.
* Migrate existing applications to the cloud.
* Design and build new applications in the cloud.

In a **cloud-based deployment** model, you can migrate existing applications to the cloud, or you can design and build new applications in the cloud. You can build those applications on low-level infrastructure that requires your IT staff to manage them. Alternatively, you can build them using higher-level services that reduce the management, architecting, and scaling requirements of the core infrastructure.

For example, a company might create an application consisting of virtual servers, databases, and networking components that are fully based in the cloud.

* Deploy resources by using virtualization and resource management tools.
* Increase resource utilization by using application management and virtualization technologies.

**On-premises deployment**is also known as a *private cloud* deployment. In this model, resources are deployed on premises by using virtualization and resource management tools.

For example, you might have applications that run on technology that is fully kept in your on-premises data center. Though this model is much like legacy IT infrastructure, its incorporation of application management and virtualization technologies helps to increase resource utilization.

**hybrid deployment**

* Connect cloud-based resources to on-premises infrastructure.
* Integrate cloud-based resources with legacy IT applications.

In a **hybrid deployment**, cloud-based resources are connected to on-premises infrastructure. You might want to use this approach in a number of situations. For example, you have legacy applications that are better maintained on premises, or government regulations require your business to keep certain records on premises.

For example, suppose that a company wants to use cloud services that can automate batch data processing and analytics. However, the company has several legacy applications that are more suitable on premises and will not be migrated to the cloud. With a hybrid deployment, the company would be able to keep the legacy applications on premises while benefiting from the data and analytics services that run in the cloud.

**Benefits of Cloud computing**

**Trade upfront expense for variable expense**

–

Upfront expense refers to data centers, physical servers, and other resources that you would need to invest in before using them. Variable expense means you only pay for computing resources you consume instead of investing heavily in data centers and servers before you know how you’re going to use them.

By taking a cloud computing approach that offers the benefit of variable expense, companies can implement innovative solutions while saving on costs.

**Stop spending money to run and maintain data centers**

–

Computing in data centers often requires you to spend more money and time managing infrastructure and servers.

A benefit of cloud computing is the ability to focus less on these tasks and more on your applications and customers.

**Stop guessing capacity**

–

With cloud computing, you don’t have to predict how much infrastructure capacity you will need before deploying an application.

For example, you can launch Amazon EC2 instances when needed, and pay only for the compute time you use. Instead of paying for unused resources or having to deal with limited capacity, you can access only the capacity that you need. You can also scale in or scale out in response to demand.

**Benefit from massive economies of scale**

–

By using cloud computing, you can achieve a lower variable cost than you can get on your own.

Because usage from hundreds of thousands of customers can aggregate in the cloud, providers, such as AWS, can achieve higher economies of scale. The economy of scale translates into lower pay-as-you-go prices.

**Increase speed and agility**

–

The flexibility of cloud computing makes it easier for you to develop and deploy applications.

This flexibility provides you with more time to experiment and innovate. When computing in data centers, it may take weeks to obtain new resources that you need. By comparison, cloud computing enables you to access new resources within minutes.

**Go global in minutes**

–

The global footprint of the AWS Cloud enables you to deploy applications to customers around the world quickly, while providing them with low latency. This means that even if you are located in a different part of the world than your customers, customers are able to access your applications with minimal delays.

Later in this course, you will explore the AWS global infrastructure in greater detail. You will examine some of the services that you can use to deliver content to customers around the world.

**Module 2 : Compute in Cloud**

* Amazon Elastic Compute Cloud (Amazon EC2)

provides secure, resizable compute capacity in the cloud as Amazon EC2 instances.

Imagine you are responsible for the architecture of your company's resources and need to support new websites. With traditional on-premises resources, you have to do the following:

* Spend money upfront to purchase hardware.
* Wait for the servers to be delivered to you.
* Install the servers in your physical data center.
* Make all the necessary configurations.

By comparison, with an Amazon EC2 instance you can use a virtual server to run applications in the AWS Cloud.

* You can provision and launch an Amazon EC2 instance within minutes.
* You can stop using it when you have finished running a workload.
* You pay only for the compute time you use when an instance is running, not when it is stopped or terminated.
* You can save costs by paying only for server capacity that you need or want.
* How Amazon EC2 Works
* Launch

First launch an instance, begin by selecting a template with basic configurations. These include operating system, application server, or applications. Also select the instance type which is the specific hardware configuration of the instance.

* Connect

Next, connect to the instance. You can connect to instance in several ways. Your programs and applications have multiple different methods to connect to directly to the instance and exchange data. Users can also connect by logging in and accessing the computer desktop

* Use

After you have connected, you can begin using it. You can run commands to install software, add storage, copy and organise files and more.

* Amazon EC2 Instance Types

Are optimised for different tasks. When selecting an instance type, consider the specific needs of your workloads and applications. This might include requirements for compute, memory, or storage capabilities.

**General purpose instances** provide a balance of compute, memory, and networking resources. You can use them for a variety of workloads, such as:

* application servers
* gaming servers
* backend servers for enterprise applications
* small and medium databases

Suppose that you have an application in which the resource needs for compute, memory, and networking are roughly equivalent. You might consider running it on a general purpose instance because the application does not require optimization in any single resource area.

**Compute optimized instances** are ideal for compute-bound applications that benefit from high-performance processors. Like general purpose instances, you can use compute optimized instances for workloads such as web, application, and gaming servers.

However, the difference is compute optimized applications are ideal for high-performance web servers, compute-intensive applications servers, and dedicated gaming servers. You can also use compute optimized instances for batch processing workloads that require processing many transactions in a single group.

**Memory optimized instances** are designed to deliver fast performance for workloads that process large datasets in memory. In computing, memory is a temporary storage area. It holds all the data and instructions that a central processing unit (CPU) needs to be able to complete actions. Before a computer program or application is able to run, it is loaded from storage into memory. This preloading process gives the CPU direct access to the computer program.

Suppose that you have a workload that requires large amounts of data to be preloaded before running an application. This scenario might be a high-performance database or a workload that involves performing real-time processing of a large amount of unstructured data. In these types of use cases, consider using a memory optimized instance. Memory optimized instances enable you to run workloads with high memory needs and receive great performance.

**Accelerated computing instances** use hardware accelerators, or coprocessors, to perform some functions more efficiently than is possible in software running on CPUs. Examples of these functions include floating-point number calculations, graphics processing, and data pattern matching.

In computing, a hardware accelerator is a component that can expedite data processing. Accelerated computing instances are ideal for workloads such as graphics applications, game streaming, and application streaming.

**Storage optimized instances** are designed for workloads that require high, sequential read and write access to large datasets on local storage. Examples of workloads suitable for storage optimized instances include distributed file systems, data warehousing applications, and high-frequency online transaction processing (OLTP) systems.

In computing, the term input/output operations per second (IOPS) is a metric that measures the performance of a storage device. It indicates how many different input or output operations a device can perform in one second. Storage optimized instances are designed to deliver tens of thousands of low-latency, random IOPS to applications.

You can think of input operations as data put into a system, such as records entered into a database. An output operation is data generated by a server. An example of output might be the analytics performed on the records in a database. If you have an application that has a high IOPS requirement, a storage optimized instance can provide better performance over other instance types not optimized for this kind of use case.

* Amazon EC2 Pricing

With Amazon EC2, you pay only for the compute time that you use. Amazon EC2 offers a variety of pricing options for different use cases. For example, if your use case can withstand interruptions, you can save with Spot Instances. You can also save by committing early and locking in a minimum level of use with Reserved Instances.

**On-Demand Instances** are ideal for short-term, irregular workloads that cannot be interrupted. No upfront costs or minimum contracts apply. The instances run continuously until you stop them, and you pay for only the compute time you use.  
  
Sample use cases for On-Demand Instances include developing and testing applications and running applications that have unpredictable usage patterns. On-Demand Instances are not recommended for workloads that last a year or longer because these workloads can experience greater cost savings using Reserved Instances.

**Amazon EC2 Savings Plans** offers Savings Plans for several compute services, including Amazon EC2. **Amazon EC2 Savings Plans** enable you to reduce your compute costs by committing to a consistent amount of compute usage for a 1-year or 3-year term. This term commitment results in savings of up to 72% over On-Demand costs.

**Reserved Instances** are a billing discount applied to the use of On-Demand Instances in your account. You can purchase Standard Reserved and Convertible Reserved Instances for a 1-year or 3-year term, and Scheduled Reserved Instances for a 1-year term. You realize greater cost savings with the 3-year option.

At the end of a Reserved Instance term, you can continue using the Amazon EC2 instance without interruption. However, you are charged On-Demand rates until you do one of the following:

* Terminate the instance.
* Purchase a new Reserved Instance that matches the instance attributes (instance type, Region, tenancy, and platform).

**Spot Instances** are ideal for workloads with flexible start and end times, or that can withstand interruptions. Spot Instances use unused Amazon EC2 computing capacity and offer you cost savings at up to 90% off of On-Demand prices.  
  
Suppose that you have a background processing job that can start and stop as needed (such as the data processing job for a customer survey). You want to start and stop the processing job without affecting the overall operations of your business. If you make a Spot request and Amazon EC2 capacity is available, your Spot Instance launches. However, if you make a Spot request and Amazon EC2 capacity is unavailable, the request is not successful until capacity becomes available. The unavailable capacity might delay the launch of your background processing job.  
  
After you have launched a Spot Instance, if capacity is no longer available or demand for Spot Instances increases, your instance may be interrupted. This might not pose any issues for your background processing job. However, in the earlier example of developing and testing applications, you would most likely want to avoid unexpected interruptions. Therefore, choose a different EC2 instance type that is ideal for those tasks.

Do not require any contracts or a commitment to a consistent amount of compute usage

**Dedicated Hosts**are physical servers with Amazon EC2 instance capacity that is fully dedicated to your use.

* Amazon EC2 Scaling

**Scalability** involves beginning with only the resources you need and designing your architecture to automatically respond to changing demand by scaling out or in. As a result, you pay for only the resources you use. You don’t have to worry about a lack of computing capacity to meet your customers’ needs.

If you wanted the scaling process to happen automatically, which AWS service would you use? The AWS service that provides this functionality for Amazon EC2 instances is **Amazon EC2 Auto Scaling**.

Within Amazon EC2 Auto Scaling, you can use two approaches: dynamic scaling and predictive scaling.

* Dynamic scaling responds to changing demand.
* Predictive scaling automatically schedules the right number of Amazon EC2 instances based on predicted demand.

**To scale faster, you can use dynamic scaling and predictive scaling together.**

The **minimum capacity** is the number of Amazon EC2 instances that launch immediately after you have created the Auto Scaling group

Next, you can set the **desired capacity** at two Amazon EC2 instances even though your application needs a minimum of a single Amazon EC2 instance to run.

**If you do not specify the desired number of Amazon EC2 instances in an Auto Scaling group, the desired capacity defaults to your minimum capacity.**

The third configuration that you can set in an Auto Scaling group is the **maximum capacity**. For example, you might configure the Auto Scaling group to scale out in response to increased demand, but only to a maximum of four Amazon EC2 instances.

Because Amazon EC2 Auto Scaling uses Amazon EC2 instances, you pay for only the instances you use, when you use them. You now have a cost-effective architecture that provides the best customer experience while reducing expenses.

* Elastic Load Balancing

**Elastic Load Balancing** is the AWS service that automatically distributes incoming application traffic across multiple resources, such as Amazon EC2 instances.

A load balancer acts as a single point of contact for all incoming web traffic to your Auto Scaling group.

Although Elastic Load Balancing and Amazon EC2 Auto Scaling are separate services, they work together to help ensure that applications running in Amazon EC2 can provide high performance and availability.

* Messaging and queuing

**Monolithic applications and microservices**

Suppose that you have an application with tightly coupled components. These components might include databases, servers, the user interface, business logic, and so on. This type of architecture can be considered a **monolithic application**.

In this approach to application architecture, if a single component fails, other components fail, and possibly the entire application fails.

In a **microservices approach**, application components are loosely coupled. In this case, if a single component fails, the other components continue to work because they are communicating with each other. The loose coupling prevents the entire application from failing.

**Amazon Simple notification service ( Amazon SNS)**

**Amazon Simple Notification Service (Amazon SNS)**is a publish/subscribe service. Using Amazon SNS topics, a publisher publishes messages to subscribers. This is similar to the coffee shop; the cashier provides coffee orders to the barista who makes the drinks.

In Amazon SNS, subscribers can be web servers, email addresses, AWS Lambda functions, or several other options.

**Amazon Simple Queue service ( Amazon SQS)**

Using Amazon SQS, you can send, store, and receive messages between software components, without losing messages or requiring other services to be available. In Amazon SQS, an application sends messages into a queue. A user or service retrieves a message from the queue, processes it, and then deletes it from the queue.

* Additional Compute Services

**Serverless computing**

Earlier in this module, you learned about Amazon EC2, a service that lets you run virtual servers in the cloud. If you have applications that you want to run in Amazon EC2, you must do the following:

1. Provision instances (virtual servers).
2. Upload your code.
3. Continue to manage the instances while your application is running.

The term “serverless” means that your code runs on servers, but you do not need to provision or manage these servers. With serverless computing, you can focus more on innovating new products and features instead of maintaining servers.

Another benefit of serverless computing is the flexibility to scale serverless applications automatically. Serverless computing can adjust the applications' capacity by modifying the units of consumptions, such as throughput and memory.

An AWS service for serverless computing is **AWS Lambda**.

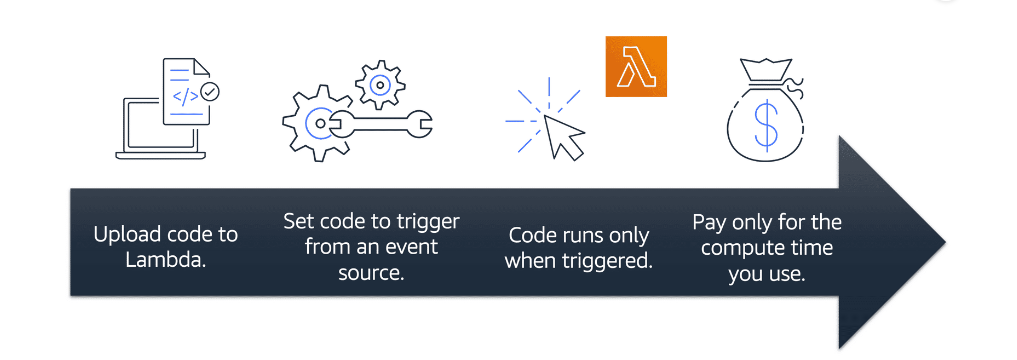
**AWS Lambda**

[**AWS Lambda**](https://aws.amazon.com/lambda) is a service that lets you run code without needing to provision or manage servers.

While using AWS Lambda, you pay only for the compute time that you consume. Charges apply only when your code is running. You can also run code for virtually any type of application or backend service, all with zero administration.

For example, a simple Lambda function might involve automatically resizing uploaded images to the AWS Cloud. In this case, the function triggers when uploading a new image.

**AWS Lambda – How it works**



**In AWS, you can also build and run containerized applications.**

**Containers** provide you with a standard way to package your application's code and dependencies into a single object. You can also use containers for processes and workflows in which there are essential requirements for security, reliability, and scalability.

[**Amazon Elastic Container Service (Amazon ECS)**](https://aws.amazon.com/ecs/) is a highly scalable, high-performance container management system that enables you to run and scale containerized applications on AWS.

Amazon ECS supports Docker containers. [Docker](https://www.docker.com/) is a software platform that enables you to build, test, and deploy applications quickly. AWS supports the use of open-source Docker Community Edition and subscription-based Docker Enterprise Edition. With Amazon ECS, you can use API calls to launch and stop Docker-enabled applications

[**Amazon Elastic Kubernetes Service (Amazon EKS)**](https://aws.amazon.com/eks/)is a **fully managed service** that you can use to run Kubernetes on AWS.

[Kubernetes](https://kubernetes.io/) is open-source software that enables you to deploy and manage containerized applications at scale. A large community of volunteers maintains Kubernetes, and AWS actively works together with the Kubernetes community. As new features and functionalities release for Kubernetes applications, you can easily apply these updates to your applications managed by Amazon EKS.

[**AWS Fargate**](https://aws.amazon.com/fargate/)is a serverless compute engine for containers. It works with both Amazon ECS and Amazon EKS.

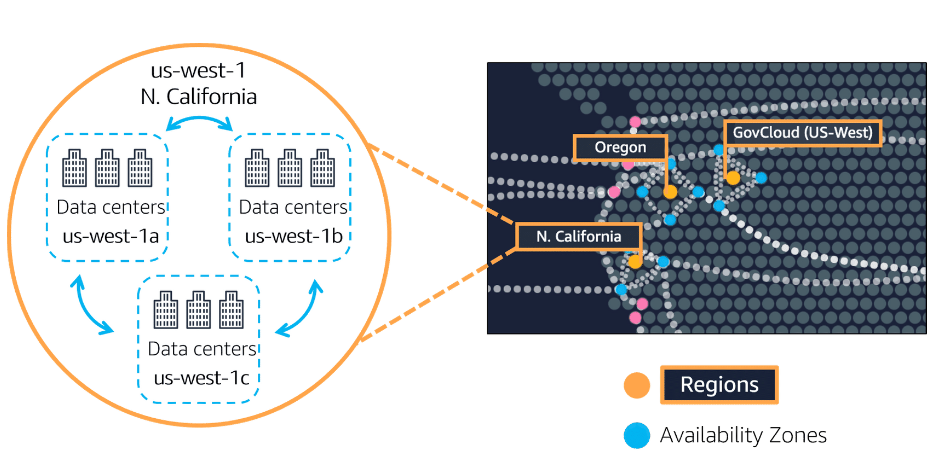
When using AWS Fargate, you do not need to provision or manage servers. AWS Fargate manages your server infrastructure for you. You can focus more on innovating and developing your applications, and you pay only for the resources that are required to run your containers.

**Module 3 : Global Infrastructure and Reliability**

* AWS Global Infrastructure

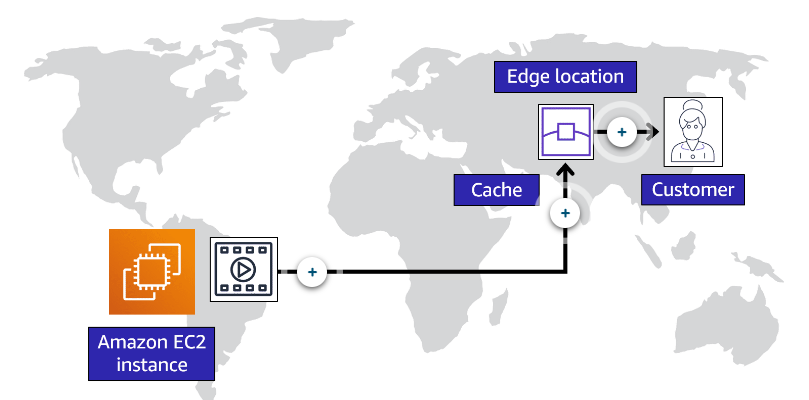
**Selecting a region for services, data, and applications consider the following 4 business factors**

* + **Compliance with data governance and Legal requirements**
  + **Proximity to customers**
  + **Available services within the region**
  + **Pricing – Eg ..** The way Brazil’s tax structure is set up, it might cost 50% more to run the same workload out of the São Paulo Region compared to the Oregon Region (US)



An **Availability Zone** is a single data center or a group of data centers within a Region. Availability Zones are located tens of miles apart from each other. This is close enough to have low latency (the time between when content requested and received) between Availability Zones. However, if a disaster occurs in one part of the Region, they are distant enough to reduce the chance that multiple Availability Zones are affected.

* EDGE Location



An **edge location** is a site that Amazon CloudFront uses to store cached copies of your content closer to your customers for faster delivery.

* How to provision AWS resources

Ways to interact with AWS

The **AWS Management Console** is a web-based interface for accessing and managing **AWS services.** You can quickly access recently used services and search for other services by name, keyword, or acronym. The console includes wizards and automated workflows that can simplify the process of completing tasks. You can also use the AWS Console mobile application to perform tasks such as monitoring resources, viewing alarms, and accessing billing information. Multiple identities can stay logged into the AWS Console mobile app at the same time.

To save time when **making API requests**, you can use the **AWS Command Line Interface** (AWS CLI). AWS CLI enables you to control multiple AWS services directly from the command line within one tool. AWS CLI is available for users on Windows, macOS, and Linux.

By using AWS CLI, you can automate the actions that your services and applications perform through scripts. For example, you can use commands to launch an Amazon EC2 instance, connect an Amazon EC2 instance to a specific Auto Scaling group, and more.

Another option for accessing and managing AWS services is the **software development kits (SDKs).** SDKs make it easier for you to use AWS services through an API designed for your programming language or platform. SDKs enable you to use AWS services with your existing applications or create entirely new applications that will run on AWS.

To help you get started with using SDKs, AWS provides documentation and sample code for each supported programming language. Supported programming languages include C++, Java, .NET, and more.

With **AWS Elastic Beanstalk**, you provide code and configuration settings, and Elastic Beanstalk deploys the resources necessary to perform the following tasks:

* Adjust capacity
* Load balancing
* Automatic scaling
* Application health monitoring

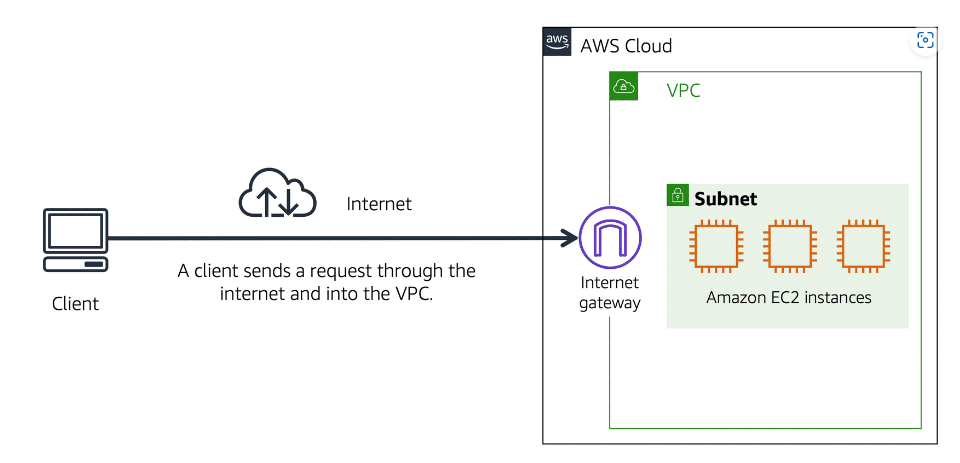
With **AWS CloudFormation,** you can treat your **infrastructure as code.** This means that you can build an environment by writing lines of code instead of using the AWS Management Console to individually provision resources.

AWS CloudFormation provisions your resources in a safe, repeatable manner, enabling you to frequently build your infrastructure and applications without having to perform manual actions. It determines the right operations to perform when managing your stack and rolls back changes automatically if it detects errors.

**AWS Outposts** Run AWS infrastructure and services on premises for a truly consistent hybrid experience

**Module 4 : Networking**

* Connectivity to AWS

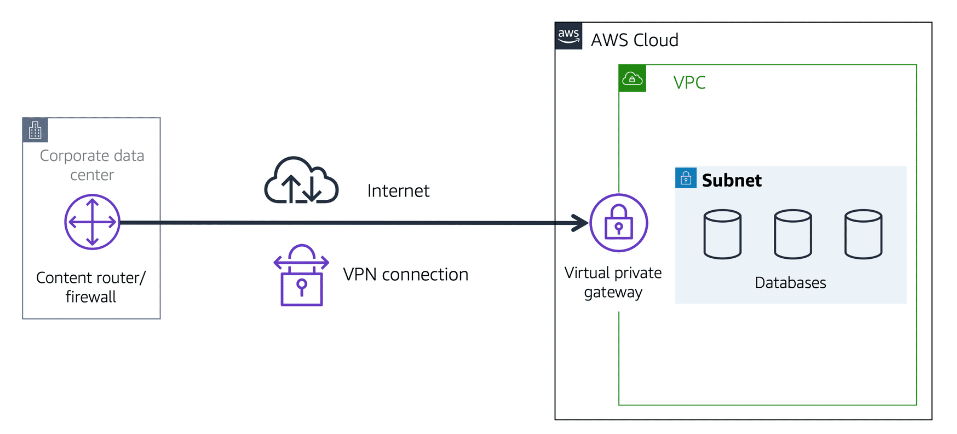


A networking service that you can use to establish boundaries around your AWS resources is [**Amazon Virtual Private Cloud (Amazon VPC)**](https://aws.amazon.com/vpc/).

Amazon VPC enables you to provision an isolated section of the AWS Cloud. In this isolated section, you can launch resources in a virtual network that you define. Within a **virtual private cloud (VPC)**, you can organize your resources into subnets. A **subnet** is a section of a VPC that can contain resources such as Amazon EC2 instances.

An **internet gateway** is a connection between a **VPC and the internet.** You can think of an internet gateway as being similar to a doorway that customers use to enter the coffee shop. Without an internet gateway, no one can access the resources within your VPC.

What if you have a VPC that includes only private resources?



To access private resources in a VPC, you can use a **virtual private gateway**.

Here’s an example of how a virtual private gateway works. You can think of the internet as the road between your home and the coffee shop. Suppose that you are traveling on this road with a bodyguard to protect you. You are still using the same road as other customers, but with an extra layer of protection.

The bodyguard is like a virtual private network (VPN) connection that encrypts (or protects) your internet traffic from all the other requests around it.

The virtual private gateway is the component that allows protected internet traffic to enter into the VPC. Even though your connection to the coffee shop has extra protection, traffic jams are possible because you’re using the same road as other customers.

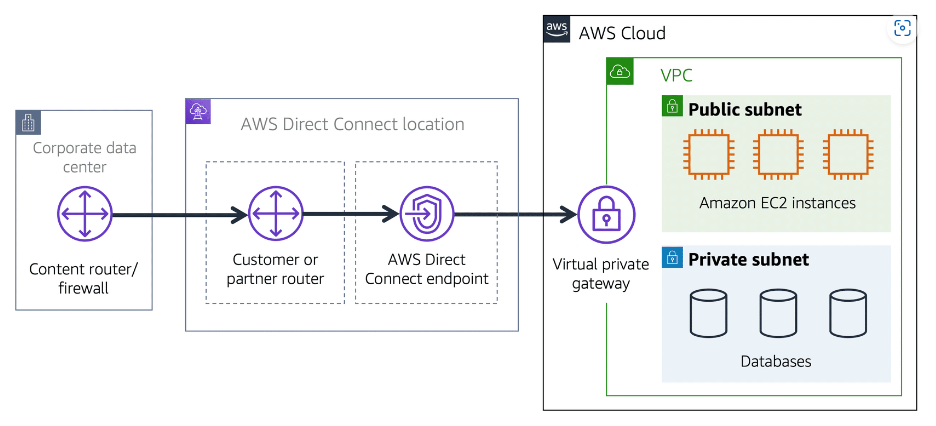
A virtual private gateway enables you to establish a virtual private network (VPN) connection between your VPC and a private network, such as an on-premises data center or internal corporate network. A virtual private gateway allows traffic into the VPC only if it is coming from an approved network.

[**AWS Direct Connect**](https://aws.amazon.com/directconnect/) is a service that enables you to establish a dedicated private connection between your data center and a VPC.

Suppose that there is an apartment building with a hallway directly linking the building to the coffee shop. Only the residents of the apartment building can travel through this hallway.

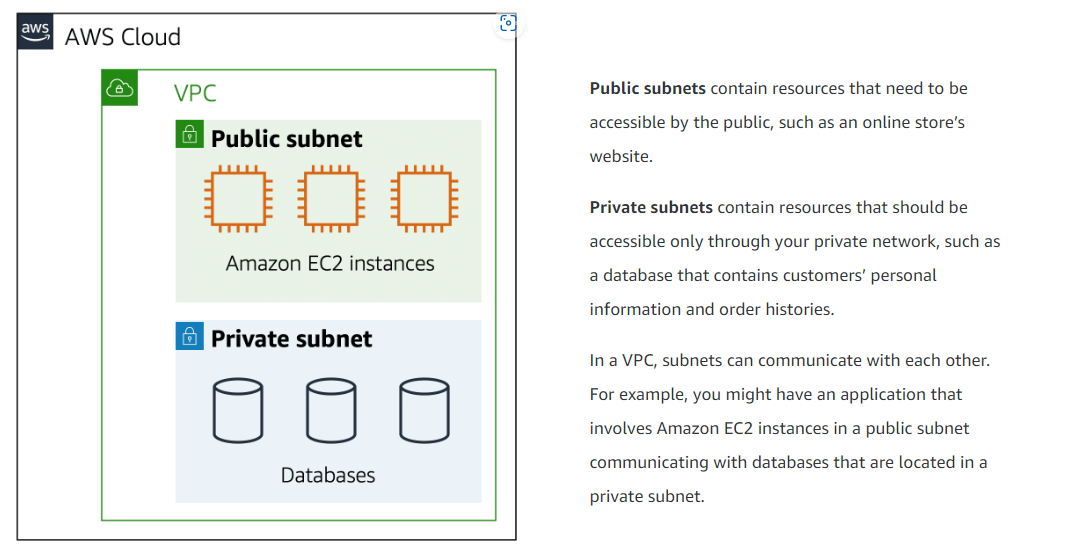
This private hallway provides the same type of dedicated connection as AWS Direct Connect. Residents are able to get into the coffee shop without needing to use the public road shared with other customers.

The private connection that AWS Direct Connect provides helps you to reduce network costs and increase the amount of bandwidth that can travel through your network.



* Subnets and Network access control lists

A subnet is a section of a VPC in which you can group resources based on security or operational needs. Subnets can be public or private.



When a customer requests data from an application hosted in the AWS Cloud, this request is sent as a packet. A packet is a unit of data sent over the internet or a network.

It enters into a VPC through an internet gateway. Before a packet can enter into a subnet or exit from a subnet, it checks for permissions. These permissions indicate who sent the packet and how the packet is trying to communicate with the resources in a subnet.

The VPC component that checks packet permissions for subnets is a [**network access control list (ACL)**](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html)**.**

[**network access control list (ACL)**](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html)is a virtual firewall that controls **inbound and outbound traffic at the subnet level.**

Each AWS account includes a default network ACL. When configuring your VPC, you can use your account’s default network ACL or create custom network ACLs.

By default, your account’s **default network ACL allows all inbound and outbound traffic**, but you can modify it by adding your own rules**. For custom network ACLs, all inbound and outbound traffic is denied until you add rules to specify which traffic to allow**. Additionally, all network ACLs have an explicit deny rule. This rule ensures that if a packet doesn’t match any of the other rules on the list, the packet is denied.

Network ACLs perform **stateless** packet filtering. They remember nothing and check packets that cross the subnet border each way: inbound and outbound.

A network ACL has inbound rules and outbound rules. Each rule can either allow or deny traffic. Each rule has a number 1 from to 32766. We evaluate the rules in order, starting with the lowest numbered rule, when deciding whether allow or deny traffic. If the traffic matches a rule, the rule is applied and we do not evaluate any additional rules. We recommend that you start by creating rules in increments (for example, increments of 10 or 100) so that you can insert new rules later on, if needed.

**After a packet has entered a subnet**, it must have its permissions evaluated for resources within the subnet, such as Amazon EC2 instances.

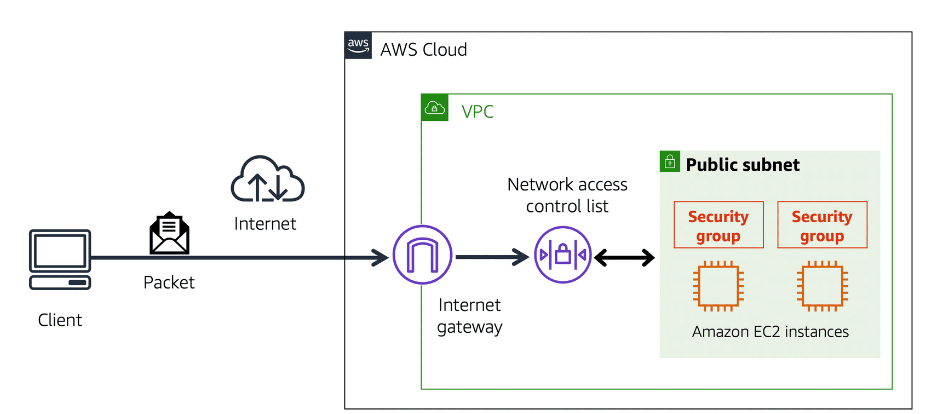
The VPC component that checks packet permissions for an **Amazon EC2 instance** is a [**security group**](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html).

**A security group** is a virtual firewall that controls inbound and outbound traffic for an Amazon EC2 instance.

By default, a security group **denies all inbound traffic** and **allows all outbound traffic**. You can add custom rules to configure which traffic to allow or deny.

If you have multiple Amazon EC2 instances within a subnet, you can associate them with the same security group or use different security groups for each instance.

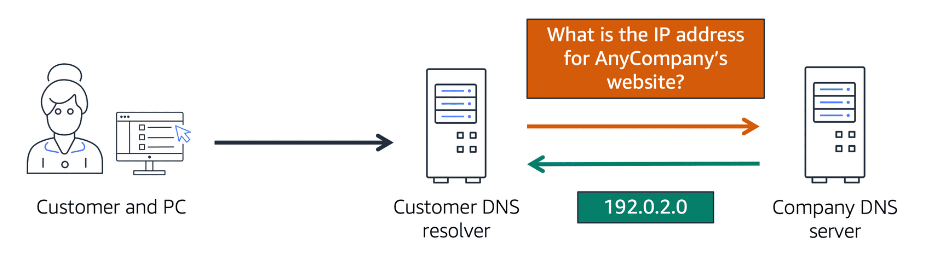
Security groups perform **stateful** packet filtering. They remember previous decisions made for incoming packets.



* Global Networking

Suppose that AnyCompany has a website hosted in the AWS Cloud. Customers enter the web address into their browser, and they are able to access the website. This happens because of Domain Name System (DNS) resolution. DNS resolution involves a customer DNS resolver communicating with a company DNS server.

You can think of DNS as being the phone book of the internet. DNS resolution is the process of translating a domain name to an IP address.



[**Amazon Route 53**](https://aws.amazon.com/route53) is a DNS web service. It gives developers and businesses a reliable way to route end users to internet applications hosted in AWS.

Amazon Route 53 connects user requests to infrastructure running in AWS (such as Amazon EC2 instances and load balancers). It can route users to infrastructure outside of AWS.

Another feature of Route 53 is the ability to manage the DNS records for domain names. You can register new domain names directly in Route 53. You can also transfer DNS records for existing domain names managed by other domain registrars. This enables you to manage all of your domain names within a single location.

**Module 5 : Storage and Databases**

* Instance and EBS (Elastic Block Store)

Block-level storage volumes behave like physical hard drives.

An [**instance store**](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/InstanceStorage.html) provides temporary block-level storage for an Amazon EC2 instance. An instance store is disk storage that is physically attached to the host computer for an EC2 instance, and therefore has the same lifespan as the instance. When the instance is terminated, you lose any data in the instance store.

[**Amazon Elastic Block Store (Amazon EBS)**](https://aws.amazon.com/ebs) is a service that provides block-level storage volumes that you can use with Amazon EC2 instances. If you stop or terminate an Amazon EC2 instance, all the data on the attached EBS volume remains available.

To create an EBS volume, you define the configuration (such as volume size and type) and provision it. After you create an EBS volume, it can attach to an Amazon EC2 instance.

Because EBS volumes are for data that needs to persist, it’s important to back up the data. **You can take incremental backups of EBS volumes by creating Amazon EBS snapshots.**

An [**EBS snapshot**](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSSnapshots.html) is an incremental backup. This means that the first backup taken of a volume copies all the data. For subsequent backups, only the blocks of data that have changed since the most recent snapshot are saved.

Incremental backups are different from full backups, in which all the data in a storage volume copies each time a backup occurs. The full backup includes data that has not changed since the most recent backup.

An Amazon EBS volume stores data in a **single** Availability Zone.

To attach an Amazon EC2 instance to an EBS volume, both the Amazon EC2 instance and the EBS volume must reside within the same Availability Zone.

* Amazon Simple Storage ( Amazon S3)

**In object storage**, each object consists of data, metadata, and a key.

The data might be an image, video, text document, or any other type of file. Metadata contains information about what the data is, how it is used, the object size, and so on. An object’s key is its unique identifier.

Recall that when you modify a file in block storage, only the pieces that are changed are updated. **When a file in object storage is modified, the entire object is updated**.

[**Amazon Simple Storage Service (Amazon S3)**](https://aws.amazon.com/s3/) is a service that provides object-level storage. Amazon S3 stores data as objects in buckets.

You can upload any type of file to Amazon S3, such as images, videos, text files, and so on. For example, you might use Amazon S3 to store backup files, media files for a website, or archived documents. Amazon S3 offers unlimited storage space. The maximum file size for an object in **Amazon S3 is 5 TB.**

When you upload a file to Amazon S3, you can set permissions to control visibility and access to it. You can also use the Amazon S3 versioning feature to track changes to your objects over time.

**Amazon S3 storage classes**

With Amazon S3, you pay only for what you use. You can choose from [a range of storage classes](https://aws.amazon.com/s3/storage-classes) to select a fit for your business and cost needs. When selecting an Amazon S3 storage class, consider these two factors:

* How often you plan to retrieve your data
* How available you need your data to be

**Amazon S3 Standard** provides high availability for objects. This makes it a good choice for a wide range of use cases, such as websites, content distribution, and data analytics. Amazon S3 Standard has a higher cost than other storage classes intended for infrequently accessed data and archival storage.

* Designed for frequently accessed data
* Stores data in a minimum of three Availability Zones

**Amazon S3 Standard-IA (Infrequent Access)** is ideal for data infrequently accessed but requires high availability when needed. Both Amazon S3 Standard and Amazon S3 Standard-IA store data in a minimum of three Availability Zones. Amazon S3 Standard-IA provides the same level of availability as Amazon S3 Standard but with a lower storage price and a higher retrieval price.

* Ideal for infrequently accessed data
* Similar to Amazon S3 Standard but has a lower storage price and higher retrieval price

**Amazon S3 one zone -infrequent access (S3 One Zone -IA)**

Compared to Amazon S3 Standard and Amazon S3 Standard-IA, which store data in a minimum of three Availability Zones, Amazon S3 One Zone-IA stores data in a single Availability Zone.

* Stores data in a single Availability Zone
* Has a lower storage price than Amazon S3 Standard-IA

This makes it a good storage class to consider if the following conditions apply:

* You want to save costs on storage.
* You can easily reproduce your data in the event of an Availability Zone failure.

In the **Amazon S3 Intelligent-Tiering** storage class, Amazon S3 monitors objects’ access patterns. If you haven’t accessed an object for 30 consecutive days, Amazon S3 automatically moves it to the infrequent access tier, Amazon S3 Standard-IA. If you access an object in the infrequent access tier, Amazon S3 automatically moves it to the frequent access tier, Amazon S3 Standard.

* Ideal for data with unknown or changing access patterns
* Requires a small monthly monitoring and automation fee per object

**Amazon S3 Glacier Instant Retrieval**

When you decide between the options for archival storage, consider how quickly you must retrieve the archived objects. You can retrieve objects stored in the Amazon S3 Glacier Instant Retrieval storage class within milliseconds, with the same performance as Amazon S3 Standard.

* Works well for archived data that requires immediate access
* Can retrieve objects within a few milliseconds

**Amazon S3 Glacier Flexible Retrieval**

Amazon S3 Glacier Flexible Retrieval is a low-cost storage class that is ideal for data archiving. For example, you might use this storage class to store archived customer records or older photos and video files.

* Low-cost storage designed for data archiving
* Able to retrieve objects within a few minutes to hours

**Amazon S3 Glacier Deep Archive**

Amazon S3 Deep Archive supports long-term retention and digital preservation for data that might be accessed once or twice in a year. This storage class is the lowest-cost storage in the AWS Cloud, with data retrieval from 12 to 48 hours. All objects from this storage class are replicated and stored across at least three geographically dispersed Availability Zones.

* Lowest-cost object storage class ideal for archiving
* Able to retrieve objects within 12 hours

**Amazon S3 Outposts**

Amazon S3 Outposts delivers object storage to your on-premises AWS Outposts environment. Amazon S3 Outposts is designed to store data durably and redundantly across multiple devices and servers on your Outposts. It works well for workloads with local data residency requirements that must satisfy demanding performance needs by keeping data close to on-premises applications.

* Creates S3 buckets on Amazon S3 Outposts
* Makes it easier to retrieve, store, and access data on AWS Outposts
* Amazon Elastic File System ( Amazon EFS)

In **file storage**, multiple clients (such as users, applications, servers, and so on) can access data that is stored in shared file folders. In this approach, a storage server uses block storage with a local file system to organize files. Clients access data through file paths.

Compared to block storage and object storage, file storage is ideal for use cases in which a large number of services and resources need to access the same data at the same time.

[**Amazon Elastic File System (Amazon EFS)**](https://aws.amazon.com/efs/) is a scalable file system used with AWS Cloud services and on-premises resources. As you add and remove files, Amazon EFS grows and shrinks automatically. **It can scale on demand to petabytes without disrupting applications.**

Amazon EFS is a regional service. It stores data in and across **multiple** Availability Zones.

The duplicate storage enables you to access data concurrently from all the Availability Zones in the Region where a file system is located. Additionally, on-premises servers can access Amazon EFS using AWS Direct Connect.

* Amazon Relation database service

Relational databases use **structured query language (SQL)** to store and query data. This approach allows data to be stored in an easily understandable, consistent, and scalable way. For example, the coffee shop owners can write a SQL query to identify all the customers whose most frequently purchased drink is a medium latte.

[**Amazon Relational Database Service (Amazon RDS)**](https://aws.amazon.com/rds/) is a service that enables you to run relational databases in the AWS Cloud.

Amazon RDS is a managed service that automates tasks such as hardware provisioning, database setup, patching, and backups. With these capabilities, you can spend less time completing administrative tasks and more time using data to innovate your applications. You can integrate Amazon RDS with other services to fulfill your business and operational needs, such as using AWS Lambda to query your database from a serverless application.

Amazon RDS provides a number of different security options. Many Amazon RDS database engines offer encryption at rest (protecting data while it is stored) and encryption in transit (protecting data while it is being sent and received).

Amazon RDS is available on **six database engines,** which optimize for memory, performance, or input/output (I/O). Supported database engines include:

* Amazon Aurora
* PostgreSQL
* MySQL
* MariaDB
* Oracle Database
* Microsoft SQL Server

[**Amazon Aurora**](https://aws.amazon.com/rds/aurora/) is an enterprise-class relational database. It is compatible with MySQL and PostgreSQL relational databases. It is up to five times faster than standard MySQL databases and up to three times faster than standard PostgreSQL databases.

Amazon Aurora helps to reduce your database costs by reducing unnecessary input/output (I/O) operations, while ensuring that your database resources remain reliable and available.

Consider Amazon Aurora if your workloads require high availability. **It replicates six copies of your data across three Availability Zones and continuously backs up your data to Amazon S3.**

* Amazon DynamoDB

[**Amazon DynamoDB**](https://aws.amazon.com/dynamodb/) is a key-value database service. It delivers single-digit millisecond performance at any scale.

In a **nonrelational database**, you create tables. A table is a place where you can store and query data.

Nonrelational databases are sometimes referred to as “NoSQL databases” because they use structures other than rows and columns to organize data. One type of structural approach for nonrelational databases is key-value pairs.

DynamoDB is serverless, which means that you do not have to provision, patch, or manage servers.

You also do not have to install, maintain, or operate software.

As the size of your database shrinks or grows, DynamoDB automatically scales to adjust for changes in capacity while maintaining consistent performance.

This makes it a suitable choice for use cases that require high performance while scaling. Scaling up to 10 trillion requests per day

* Amazon Redshift

[**Amazon Redshift**](https://aws.amazon.com/redshift) is a data warehousing service that you can use for big data analytics. It offers the ability to collect data from many sources and helps you to understand relationships and trends across your data

* AWS Database migration service

[**AWS Database Migration Service (AWS DMS)**](https://aws.amazon.com/dms/) enables you to migrate relational databases, nonrelational databases, and other types of data stores.

With AWS DMS, you move data between a source database and a target database. [The source and target databases](https://aws.amazon.com/dms/resources) can be of the same type or different types. During the migration, your source database remains operational, reducing downtime for any applications that rely on the database.

Other use cases

* + Development and test database migrations

Enabling developers to test applications against production data without affecting production users

* + Database consolidation

Combining several databases into a single database

* + Continuous replication

Sending ongoing copies of your data to other target sources instead of doing a one-time migration

* Additional Database Services
  + [**Amazon DocumentDB**](https://aws.amazon.com/documentdb) is a document database service that supports MongoDB workloads. (MongoDB is a document database program.)
  + [**Amazon Neptune**](https://aws.amazon.com/neptune) is a graph database service. You can use Amazon Neptune to build and run applications that work with highly connected datasets, such as recommendation engines, fraud detection, and knowledge graphs.
  + [**Amazon Quantum Ledger Database (Amazon QLDB)**](https://aws.amazon.com/qldb) is a ledger database service.  You can use Amazon QLDB to review a complete history of all the changes that have been made to your application data.
  + [**Amazon Managed Blockchain**](https://aws.amazon.com/managed-blockchain) is a service that you can use to create and manage blockchain networks with open-source frameworks. Blockchain is a distributed ledger system that lets multiple parties run transactions and share data without a central authority.
  + [**Amazon ElastiCache**](https://aws.amazon.com/elasticache) is a service that adds caching layers on top of your databases to help improve the read times of common requests. It supports two types of data stores: **Redis and Memcached.**
* [**Amazon DynamoDB Accelerator (DAX)**](https://aws.amazon.com/dynamodb/dax/) is an in-memory cache for DynamoDB.

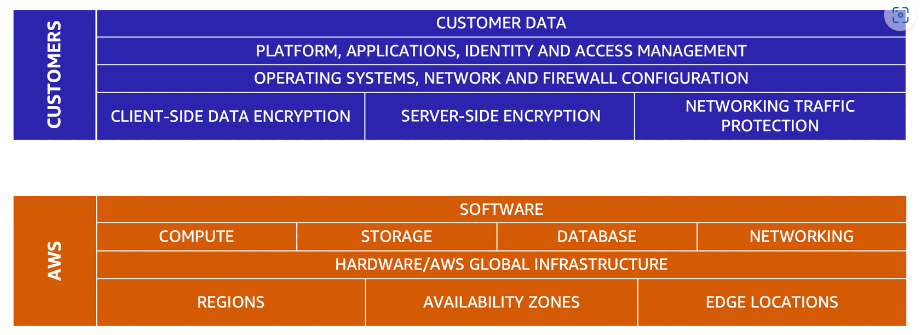
It helps improve response times from single-digit milliseconds to microseconds.

**Module 6 : Security**

* AWS Shared responsibility Model

AWS is responsible for some parts of your environment and you (the customer) are responsible for other parts. This concept is known as the [**shared responsibility model**](https://aws.amazon.com/compliance/shared-responsibility-model/).

The shared responsibility model divides into customer responsibilities (commonly referred to as “security in the cloud”) and AWS responsibilities (commonly referred to as “security of the cloud”).



**Customers: Security in the cloud**

Customers are responsible for the security of everything that they create and put *in*the AWS Cloud.

When using AWS services, you, the customer, maintain complete control over your content. You are responsible for managing security requirements for your content, including which content you choose to store on AWS, which AWS services you use, and who has access to that content. You also control how access rights are granted, managed, and revoked.

The security steps that you take will depend on factors such as the services that you use, the complexity of your systems, and your company’s specific operational and security needs. Steps include selecting, configuring, and patching the operating systems that will run on Amazon EC2 instances, configuring security groups, and managing user accounts.

E.g

* + Patching software on Amazon EC2 instances
  + Setting permissions for Amazon S3 objects

**AWS: Security of the cloud**

AWS is responsible for security *of*the cloud.

AWS operates, manages, and controls the components at all layers of infrastructure. This includes areas such as the host operating system, the virtualization layer, and even the physical security of the data centers from which services operate.

AWS is responsible for protecting the global infrastructure that runs all of the services offered in the AWS Cloud. This infrastructure includes AWS Regions, Availability Zones, and edge locations.

AWS manages the security of the cloud, specifically the physical infrastructure that hosts your resources, which include:

* Physical security of data centers
* Hardware and software infrastructure
* Network infrastructure
* Virtualization infrastructure

Although you cannot visit AWS data centers to see this protection firsthand, AWS provides several reports from third-party auditors. These auditors have verified its compliance with a variety of computer security standards and regulations.

* User Permission and Access

[**AWS Identity and Access Management (IAM)**](https://aws.amazon.com/iam/) enables you to manage access to AWS services and resources securely.

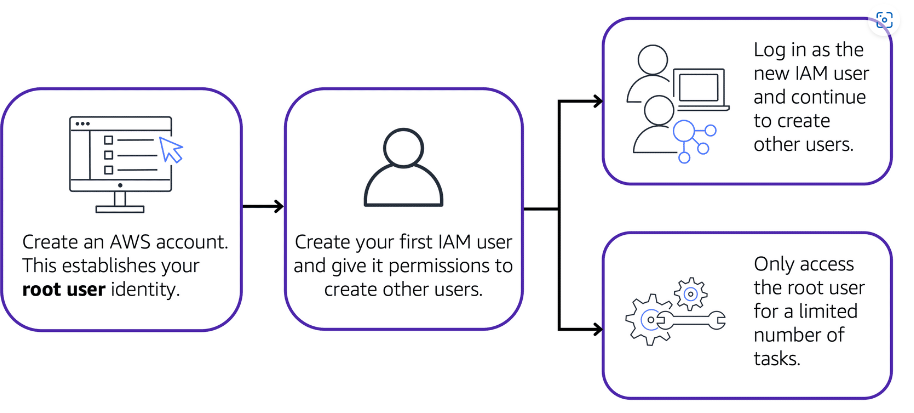
IAM gives you the flexibility to configure access based on your company’s specific operational and security needs. You do this by using a combination of IAM features, which are explored in detail in this lesson:

* IAM users, groups, and roles
* IAM policies
* Multi-factor authentication

**AWS account root user**

When you first create an AWS account, you begin with an identity known as the [**root user**](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_root-user.html).

The root user is accessed by signing in with the email address and password that you used to create your AWS account. You can think of the root user as being similar to the owner of the coffee shop. It has complete access to all the AWS services and resources in the account.



Best practice:

Do **not** use the root user for everyday tasks.

Instead, use the root user to create your first IAM user and assign it permissions to create other users.

Then, continue to create other IAM users, and access those identities for performing regular tasks throughout AWS. Only use the root user when you need to perform a limited number of tasks that are only available to the root user. **Examples of these tasks include changing your root user email address and changing your AWS support plan**.

An **IAM user** is an identity that you create in AWS. It represents the person or application that interacts with AWS services and resources. It consists of a name and credentials.

By default, when you create a new IAM user in AWS, it has no permissions associated with it. To allow the IAM user to perform specific actions in AWS, such as launching an Amazon EC2 instance or creating an Amazon S3 bucket, you must grant the IAM user the necessary permissions.

Best practice:

We recommend that you create individual IAM users for each person who needs to access AWS.

An **IAM policy** is a document that allows or denies permissions to AWS services and resources.

IAM policies enable you to customize users’ levels of access to resources. For example, you can allow users to access all of the Amazon S3 buckets within your AWS account, or only a specific bucket.

Follow the security principle of **least privilege** when granting permissions.

By following this principle, you help to prevent users or roles from having more permissions than needed to perform their tasks.

For example, if an employee needs access to only a specific bucket, specify the bucket in the IAM policy. Do this instead of granting the employee access to all of the buckets in your AWS account.

An **IAM group** is a collection of IAM users. When you assign an IAM policy to a group, all users in the group are granted permissions specified by the policy.

An **IAM role** is an identity that you can assume to gain temporary access to permissions.

Before an IAM user, application, or service can assume an IAM role, they must be granted permissions to switch to the role. When someone assumes an IAM role, they abandon all previous permissions that they had under a previous role and assume the permissions of the new role.

Best practice:

IAM roles are ideal for situations in which access to services or resources needs to be granted temporarily, instead of long-term.

In IAM, **multi-factor authentication (MFA)** provides an extra layer of security for your AWS account.

* AWS organisation

Suppose that your company has multiple AWS accounts. You can use [**AWS Organizations**](https://aws.amazon.com/organizations)to consolidate and manage multiple AWS accounts within a central location.

When you create an organization, AWS Organizations automatically creates a **root**, which is the parent container for all the accounts in your organization.

In AWS Organizations, you can centrally control permissions for the accounts in your organization by using [**service control policies (SCPs)**](https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps.html). SCPs enable you to place restrictions on the AWS services, resources, and individual API actions that users and roles in each account can access.

Consolidated billing is another feature of AWS Organizations. You will learn about consolidated billing in a later module.

In AWS Organizations, you **can group accounts into** **organizational units (OUs**) to make it easier to manage accounts with similar business or security requirements. When you apply a policy to an OU, all the accounts in the OU automatically inherit the permissions specified in the policy.

By organizing separate accounts into OUs, you can more easily isolate workloads or applications that have specific security requirements. For instance, if your company has accounts that can access only the AWS services that meet certain regulatory requirements, you can put these accounts into one OU. Then, you can attach a policy to the OU that blocks access to all other AWS services that do not meet the regulatory requirements.

* Compliance

Depending on your company’s industry, you may need to uphold specific standards. An audit or inspection will ensure that the company has met those standards.

[**AWS Artifact**](https://aws.amazon.com/artifact) is a service that provides on-demand access to AWS security and compliance reports and select online agreements. AWS Artifact consists of two main sections: **AWS Artifact Agreements and AWS Artifact Reports.**

**AWS Artifact Agreements**

Suppose that your company needs to sign an agreement with AWS regarding your use of certain types of information throughout AWS services. You can do this through **AWS Artifact Agreements**.

In AWS Artifact Agreements, you can review, accept, and manage agreements for an individual account and for all your accounts in AWS Organizations. Different types of agreements are offered to address the needs of customers who are subject to specific regulations, such as the Health Insurance Portability and Accountability Act (HIPAA).

**AWS Artifact Reports**

–

Next, suppose that a member of your company’s development team is building an application and needs more information about their responsibility for complying with certain regulatory standards. You can advise them to access this information in **AWS Artifact Reports**.

AWS Artifact Reports provide compliance reports from third-party auditors. These auditors have tested and verified that AWS is compliant with a variety of global, regional, and industry-specific security standards and regulations. AWS Artifact Reports remains up to date with the latest reports released. You can provide the AWS audit artifacts to your auditors or regulators as evidence of AWS security controls.

AWS Artifact is a web service that enables you to download AWS security and compliance documents such as ISO certifications and SOC reports

The [**Customer Compliance Center**](https://aws.amazon.com/compliance/customer-center/) contains resources to help you learn more about AWS compliance.

In the Customer Compliance Center, you can read customer compliance stories to discover how companies in regulated industries have solved various compliance, governance, and audit challenges.

You can also access compliance whitepapers and documentation on topics such as:

* AWS answers to key compliance questions
* An overview of AWS risk and compliance
* An auditing security checklist

Additionally, the Customer Compliance Center includes an auditor learning path. This learning path is designed for individuals in auditing, compliance, and legal roles who want to learn more about how their internal operations can demonstrate compliance using the AWS Cloud.

* Denial of service attacks

A **denial-of-service (DoS) attack** is a deliberate attempt to make a website or application unavailable to users.

In a **distributed denial-of-service (DDoS) attack**, multiple sources are used to start an attack that aims to make a website or application unavailable. This can come from a group of attackers, or even a single attacker. The single attacker can use multiple infected computers (also known as “bots”) to send excessive traffic to a website or application.

AWS Shield is a service that protects applications against DDoS attacks. AWS Shield provides two levels of protection: Standard and Advanced.

**AWS Shield Standard** automatically protects all AWS customers at no cost. It protects your AWS resources from the most common, frequently occurring types of DDoS attacks.

As network traffic comes into your applications, AWS Shield Standard uses a variety of analysis techniques to detect malicious traffic in real time and automatically mitigates it.

**AWS Shield Advanced** is a paid service that provides detailed attack diagnostics and the ability to detect and mitigate sophisticated DDoS attacks.

It also integrates with other services such as Amazon CloudFront, Amazon Route 53, and Elastic Load Balancing. Additionally, you can integrate AWS Shield with AWS WAF by writing custom rules to mitigate complex DDoS attacks.

* Additional Security Services

[**AWS Key Management Service (AWS KMS)**](https://aws.amazon.com/kms) enables you to perform encryption operations through the use of **cryptographic keys**. A cryptographic key is a random string of digits used for locking (encrypting) and unlocking (decrypting) data. You can use AWS KMS to create, manage, and use cryptographic keys. You can also control the use of keys across a wide range of services and in your applications.

With AWS KMS, you can choose the specific levels of access control that you need for your keys. For example, you can specify which IAM users and roles are able to manage keys. Alternatively, you can temporarily disable keys so that they are no longer in use by anyone. Your keys never leave AWS KMS, and you are always in control of them.

[**AWS WAF**](https://aws.amazon.com/waf) is a **web application firewall** that lets you monitor network requests that come into your web applications.

AWS WAF works together with Amazon CloudFront and an Application Load Balancer. Recall the network access control lists that you learned about in an earlier module. AWS WAF works in a similar way to block or allow traffic. However, it does this by using a [**web access control list (ACL)**](https://docs.aws.amazon.com/waf/latest/developerguide/web-acl.html) to protect your AWS resources.

When a request comes into AWS WAF, it checks against the list of rules that you have configured in the web ACL. If a request did not come from one of the blocked IP addresses, it allows access to the application.

**Amazon Inspector** helps to improve the security and compliance of applications by running automated security assessments. It checks applications for security vulnerabilities and deviations from security best practices, such as open access to Amazon EC2 instances and installations of vulnerable software versions.

After Amazon Inspector has performed an assessment, it provides you with a list of security findings. The list prioritizes by severity level, including a detailed description of each security issue and a recommendation for how to fix it. However, AWS does not guarantee that following the provided recommendations resolves every potential security issue. Under the shared responsibility model, customers are responsible for the security of their applications, processes, and tools that run on AWS services.

[**Amazon GuardDuty**](https://aws.amazon.com/guardduty) is a service that provides intelligent threat detection for your AWS infrastructure and resources. It identifies threats by continuously monitoring the network activity and account behavior within your AWS environment.

If GuardDuty detects any threats, you can review detailed findings about them from the AWS Management Console. Findings include recommended steps for remediation. You can also configure AWS Lambda functions to take remediation steps automatically in response to GuardDuty’s security findings.

**Module 7 : Monitoring and Analytics**

* Amazon Cloud watch

[**Amazon CloudWatch**](https://aws.amazon.com/cloudwatch/) is a web service that enables you to monitor and manage various metrics and configure alarm actions based on data from those metrics.

CloudWatch uses [**metrics**](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/working_with_metrics.html) to represent the data points for your resources. AWS services send metrics to CloudWatch. CloudWatch then uses these metrics to create graphs automatically that show how performance has changed over time.

With CloudWatch, you can create [**alarms**](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/AlarmThatSendsEmail.html) that automatically perform actions if the value of your metric has gone above or below a predefined threshold.

The CloudWatch [**dashboard**](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/CloudWatch_Dashboards.html) feature enables you to access all the metrics for your resources from a single location. For example, you can use a CloudWatch dashboard to monitor the CPU utilization of an Amazon EC2 instance, the total number of requests made to an Amazon S3 bucket, and more. You can even customize separate dashboards for different business purposes, applications, or resources.

* Amazon Cloud Trail

[**AWS CloudTrail**](https://aws.amazon.com/cloudtrail/) records API calls for your account. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, and more.

With CloudTrail, you can view a complete history of user activity and API calls for your applications and resources. Events are typically updated in CloudTrail within **15 minutes after an API call**

Within CloudTrail, you can also enable [**CloudTrail Insights**](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-insights-events-with-cloudtrail.html). This optional feature allows CloudTrail to automatically detect unusual API activities in your AWS account.

* AWS Trusted Advisor

[**AWS Trusted Advisor**](https://aws.amazon.com/premiumsupport/technology/trusted-advisor/) is a web service that inspects your AWS environment and provides real-time recommendations in accordance with AWS best practices.

Trusted Advisor compares its findings to AWS best practices in five categories: **cost optimization, performance, security, fault tolerance, and service limits**.

When you access the Trusted Advisor dashboard on the AWS Management Console, you can review completed checks for cost optimization, performance, security, fault tolerance, and service limits.

For each category:

* + The green check indicates the number of items for which it detected **no problems**.
  + The orange triangle represents the number of recommended **investigations**.
  + The red circle represents the number of recommended **actions**.
* **Module 8 : Pricing & Support**
* **AWS Free Tier**

The [AWS Free Tier](https://aws.amazon.com/free/) enables you to begin using certain services without having to worry about incurring costs for the specified period.

Three types of offers are available:

* **Always Free**

These offers do not expire and are available to all AWS customers.

For example, AWS Lambda allows 1 million free requests and up to 3.2 million seconds of compute time per month. Amazon DynamoDB allows 25 GB of free storage per month.

* **12 Months Free**

These offers are free for 12 months following your initial sign-up date to AWS.

Examples include specific amounts of Amazon S3 Standard Storage, thresholds for monthly hours of Amazon EC2 compute time, and amounts of Amazon CloudFront data transfer out.

* **Trials**

Short-term free trial offers start from the date you activate a particular service. The length of each trial might vary by number of days or the amount of usage in the service.

For example, Amazon Inspector offers a 90-day free trial. Amazon Lightsail (a service that enables you to run virtual private servers) offers 750 free hours of usage over a 30-day period.

* **AWS Pricing Concepts**

**Pay for what you use.**

For each service, you pay for exactly the amount of resources that you actually use, without requiring long-term contracts or complex licensing.

**Pay less when you reserve.**

Some services offer reservation options that provide a significant discount compared to On-Demand Instance pricing.

For example, suppose that your company is using Amazon EC2 instances for a workload that needs to run continuously. You might choose to run this workload on Amazon EC2 Instance Savings Plans, because the plan allows you to save up to 72% over the equivalent On-Demand Instance capacity.

**Pay less with volume-based discounts when you use more.**

Some services offer tiered pricing, so the per-unit cost is incrementally lower with increased usage.

For example, the more Amazon S3 storage space you use, the less you pay for it per GB

The [**AWS Pricing Calculator**](https://calculator.aws/#/) lets you explore AWS services and create an estimate for the cost of your use cases on AWS. You can organize your AWS estimates by groups that you define. A group can reflect how your company is organized, such as providing estimates by cost center.

* **AWS Billing Dashboard**

Use the [**AWS Billing & Cost Management dashboard**](https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/billing-what-is.html) to pay your AWS bill, monitor your usage, and analyze and control your costs.

* Compare your current month-to-date balance with the previous month, and get a forecast of the next month based on current usage.
* View month-to-date spend by service.
* View Free Tier usage by service.
* Access Cost Explorer and create budgets.
* Purchase and manage Savings Plans.
* Publish [AWS Cost and Usage Reports](https://docs.aws.amazon.com/cur/latest/userguide/what-is-cur.html).
* **AWS Consolidated Billing**

The consolidated billing feature of AWS Organizations enables you to receive a single bill for all AWS accounts in your organization. By consolidating, you can easily track the combined costs of all the linked accounts in your organization. The default maximum number of accounts allowed for an organization is 4, but you can contact AWS Support to increase your quota, if needed.

Another benefit of consolidated billing is the ability to share bulk discount pricing, Savings Plans, and Reserved Instances across the accounts in your organization.

* **AWS Budgets**

In [**AWS Budgets**](https://aws.amazon.com/aws-cost-management/aws-budgets), you can create budgets to plan your service usage, service costs, and instance reservations.

The information in AWS Budgets updates three times a day. This helps you to accurately determine how close your usage is to your budgeted amounts or to the AWS Free Tier limits.

**In AWS Budgets, you can also set custom alerts when your usage exceeds (or is forecasted to exceed) the budgeted amount.**

* **AWS Cost Explorer**

[**AWS Cost Explorer**](https://aws.amazon.com/aws-cost-management/aws-cost-explorer/) is a tool that enables you to visualize, understand, and manage your AWS costs and usage over time.

AWS Cost Explorer includes a default report of the costs and usage for your top five cost-accruing AWS services. You can apply custom filters and groups to analyze your data. For example, you can view resource usage at the hourly level.

* **AWS Support Plan**

AWS offers four different [**Support plans**](https://aws.amazon.com/premiumsupport/plans/) to help you troubleshoot issues, lower costs, and efficiently use AWS services.

**Basic Support** is free for all AWS customers. It includes access to **whitepapers, documentation, and support communities.** With Basic Support, **you can also contact AWS for billing questions and service limit increases.**

With Basic Support, you have access to a limited selection of AWS Trusted Advisor checks. Additionally, you can use the **AWS Personal Health Dashboard**, a tool that provides alerts and remediation guidance when AWS is experiencing events that may affect you.

If your company needs support beyond the Basic level, you could consider purchasing **Developer, Business, Enterprise On-Ramp, and Enterprise Support**.

Customers in the **Developer Support** plan have access to features such as:

* Best practice guidance
* Client-side diagnostic tools
* Building-block architecture support, which consists of guidance for how to use AWS offerings, features, and services together

Customers with a **Business Support** plan have access to additional features, including:

* Use-case guidance to identify AWS offerings, features, and services that can best support your specific needs
* **All AWS Trusted Advisor checks**
* Limited support for third-party software, such as common operating systems and application stack components

In November 2021, AWS opened enrollment into AWS **Enterprise On-Ramp Support plan**. In addition to all the features included in the Basic, Developer, and Business Support plans, customers with an Enterprise On-Ramp Support plan have access to:

* A pool of **Technical Account** Managers to provide proactive guidance and coordinate access to programs and AWS experts
* A Cost Optimization workshop (one per year)
* A Concierge support team for billing and account assistance
* Tools to monitor costs and performance through Trusted Advisor and Health API/Dashboard

Enterprise On-Ramp Support plan also provides access to a specific set of proactive support services, which are provided by a pool of Technical Account Managers.

* Consultative review and architecture guidance (one per year)
* Infrastructure Event Management support (one per year)
* Support automation workflows
* 30 minutes or less response time for business-critical issues

In addition to all features included in the Basic, Developer, Business, and Enterprise On-Ramp support plans, customers with **Enterprise Support** have access to:

* A designated Technical Account Manager to provide proactive guidance and coordinate access to programs and AWS experts
* A Concierge support team for billing and account assistance
* Operations Reviews and tools to monitor health
* Training and Game Days to drive innovation
* Tools to monitor costs and performance through Trusted Advisor and Health API/Dashboard

The Enterprise plan also provides full access to proactive services, which are provided by a designated Technical Account Manager:

* Consultative review and architecture guidance
* Infrastructure Event Management support
* Cost Optimization Workshop and tools
* Support automation workflows
* 15 minutes or less response time for business-critical issues

The Enterprise On-Ramp and Enterprise Support plans include access to a **Technical Account Manager (TAM)**.

TAM educates, empowers, and evolves your cloud journey across the full range of AWS services. TAMs provide expert engineering guidance, help you design solutions that efficiently integrate AWS services, assist with cost-effective and resilient architectures, and provide direct access to AWS programs and a broad community of experts.

* **AWS Market Place**

[**AWS Marketplace**](https://aws.amazon.com/marketplace) is a digital catalog that includes thousands of software listings from independent software vendors. You can use AWS Marketplace to find, test, and buy software that runs on AWS.

AWS Marketplace offers products in several categories, such as Infrastructure Software, DevOps, Data Products, Professional Services, Business Applications, Machine Learning, Industries, and Internet of Things (IoT).

**Module 9 : Migration and Innovation**

* **AWS cloud adaption framework (AWS CAF)**

At the highest level, the [**AWS Cloud Adoption Framework (AWS CAF)**](https://d1.awsstatic.com/whitepapers/aws_cloud_adoption_framework.pdf) organizes guidance into six areas of focus, called **Perspectives**

n general, the **Business**, **People**, and **Governance** Perspectives focus on business capabilities, whereas the **Platform**, **Security**, and **Operations** Perspectives focus on technical capabilities.

The **Business Perspective** ensures that IT aligns with business needs and that IT investments link to key business results.

Use the Business Perspective to create a strong business case for cloud adoption and prioritize cloud adoption initiatives. Ensure that your business strategies and goals align with your IT strategies and goals.

Common roles in the Business Perspective include:

* Business managers
* Finance managers
* Budget owners
* Strategy stakeholders

The **People Perspective** supports development of an organization-wide change management strategy for successful cloud adoption.

Use the People Perspective to evaluate organizational structures and roles, new skill and process requirements, and identify gaps. This helps prioritize training, staffing, and organizational changes.

Common roles in the People Perspective include:

* Human resources
* Staffing
* People managers

he **Governance Perspective** focuses on the skills and processes to align IT strategy with business strategy. This ensures that you maximize the business value and minimize risks.

Use the Governance Perspective to understand how to update the staff skills and processes necessary to ensure business governance in the cloud. Manage and measure cloud investments to evaluate business outcomes.

Common roles in the Governance Perspective include:

* Chief Information Officer (CIO)
* Program managers
* Enterprise architects
* Business analysts
* Portfolio managers

The **Platform Perspective** includes principles and patterns for implementing new solutions on the cloud, and migrating on-premises workloads to the cloud.

Use a variety of architectural models to understand and communicate the structure of IT systems and their relationships. Describe the architecture of the target state environment in detail.

Common roles in the Platform Perspective include:

* Chief Technology Officer (CTO)
* IT managers
* Solutions architects

The **Security Perspective** ensures that the organization meets security objectives for visibility, auditability, control, and agility.

Use the AWS CAF to structure the selection and implementation of security controls that meet the organization’s needs.

Common roles in the Security Perspective include:

* Chief Information Security Officer (CISO)
* IT security managers
* IT security analysts

The **Operations Perspective** helps you to enable, run, use, operate, and recover IT workloads to the level agreed upon with your business stakeholders.

Define how day-to-day, quarter-to-quarter, and year-to-year business is conducted. Align with and support the operations of the business. The AWS CAF helps these stakeholders define current operating procedures and identify the process changes and training needed to implement successful cloud adoption.

Common roles in the Operations Perspective include:

* IT operations managers
* IT support managers
* **AWS Migration strategies**

When migrating applications to the cloud, six of the most common [migration strategies](https://aws.amazon.com/blogs/enterprise-strategy/6-strategies-for-migrating-applications-to-the-cloud/) that you can implement are:

* **Rehosting** – *Lift and shift*
* **Replatforming** *– Lift, twinker and shift*
* **Refactoring/re-architecting** - how an application is architected and developed by using cloud-native features. Refactoring is driven by a strong business need to add features, scale, or performance that would otherwise be difficult to achieve in the application’s existing environment.
* **Repurchasing** - involves moving from a traditional license to a software-as-a-service model. E.g moving from CRM to sales force
* **Retaining** - consists of keeping applications that are critical for the business in the source environment. This might include applications that require major refactoring before they can be migrated, or, work that can be postponed until a later time.
* **Retiring** - is the process of removing applications that are no longer needed.
* **AWS Snow family**

The [**AWS Snow Family**](https://aws.amazon.com/snow) is a collection of physical devices that help to physically transport up to exabytes of data into and out of AWS.

AWS Snow Family is composed of **AWS Snowcone**, **AWS Snowball**, and **AWS Snowmobile**.

[**AWS Snowcone**](https://aws.amazon.com/snowcone) is a small, rugged, and secure edge computing and data transfer device.

It features **2 CPUs, 4 GB of memory, and 8 TB** of usable storage.

[**AWS Snowball**](https://aws.amazon.com/snowball/) offers two types of devices:

* **Snowball Edge Storage Optimized**devices are well suited for large-scale data migrations and recurring transfer workflows, in addition to local computing with higher capacity needs.
  + **Storage: 80 TB of hard disk drive (HDD**) capacity for block volumes and Amazon S3 compatible object storage, and 1 TB of SATA solid state drive (SSD) for block volumes.
  + Compute: **40 vCPUs, and 80 GiB of memory** to support Amazon EC2 sbe1 instances (equivalent to C5).
* **Snowball Edge Compute Optimized**provides powerful computing resources for use cases such as machine learning, full motion video analysis, analytics, and local computing stacks.
  + Storage: **42-TB usable HDD** capacity for Amazon S3 compatible object storage or Amazon EBS compatible block volumes and 7.68 TB of usable NVMe SSD capacity for Amazon EBS compatible block volumes.
  + Compute: **52 vCPUs, 208 GiB** of memory, and an optional NVIDIA Tesla V100 GPU. Devices run Amazon EC2 sbe-c and sbe-g instances, which are equivalent to C5, M5a, G3, and P3 instances.

[**AWS Snowmobile**](https://aws.amazon.com/snowmobile) is an exabyte-scale data transfer service used to move large amounts of data to AWS.

You can transfer up to **100 petabytes** of data per Snowmobile, a 45-foot long ruggedized shipping container, pulled by a semi trailer truck.

* **Innovation with AWS**

You are properly equipped to drive innovation in the cloud if you can clearly articulate the following conditions:

* The current state
* The desired state
* The problems you are trying to solve

Consider some of the paths you might explore in the future as you continue on your cloud journey.

With AWS, **serverless** refers to applications that don’t require you to provision, maintain, or administer servers. You don’t need to worry about fault tolerance or availability. AWS handles these capabilities for you.

AWS Lambda is an example of a service that you can use to run serverless applications. If you design your architecture to trigger Lambda functions to run your code, you can bypass the need to manage a fleet of servers.

Building your architecture with serverless applications enables your developers to focus on their core product instead of managing and operating servers.

AWS offers a variety of services powered by **artificial intelligence (AI)**.

For example, you can perform the following tasks:

* Convert speech to text with Amazon Transcribe.
* Discover patterns in text with Amazon Comprehend.
* Identify potentially fraudulent online activities with Amazon Fraud Detector.
* Build voice and text chatbots with **Amazon Lex. (**A service that enables you to build conversational interfaces using voice and text)

Traditional **machine learning (ML)** development is complex, expensive, time consuming, and error prone. **AWS offers Amazon SageMaker** to remove the difficult work from the process and empower you to build, train, and deploy ML models quickly.

You can use ML to analyze data, solve complex problems, and predict outcomes before they happen.

**Module 9 : Cloud Journey**

* **The AWS Well Architected Framework**

The [**AWS Well-Architected Framework**](https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf) helps you understand how to design and operate reliable, secure, efficient, and cost-effective systems in the AWS Cloud. It provides a way for you to consistently measure your architecture against best practices and design principles and identify areas for improvement.

The Well-Architected Framework is based on six pillars:

**Operational excellence** is the ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures.

Design principles for operational excellence in the cloud include performing operations as code, annotating documentation, anticipating failure, and frequently making small, reversible changes.

The **Security** pillar is the ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies.

When considering the security of your architecture, apply these best practices:

* Automate security best practices when possible.
* Apply security at all layers.
* Protect data in transit and at rest.

**Reliability** is the ability of a system to do the following:

* Recover from infrastructure or service disruptions
* Dynamically acquire computing resources to meet demand
* Mitigate disruptions such as misconfigurations or transient network issues

Reliability includes testing recovery procedures, scaling horizontally to increase aggregate system availability, and automatically recovering from failure.

**Performance efficiency** is the ability to use computing resources efficiently to meet system requirements and to maintain that efficiency as demand changes and technologies evolve.

Evaluating the performance efficiency of your architecture includes experimenting more often, using serverless architectures, and designing systems to be able to go global in minutes.

**Cost optimization** is the ability to run systems to deliver business value at the lowest price point.

Cost optimization includes adopting a consumption model, analyzing and attributing expenditure, and using managed services to reduce the cost of ownership.

**Sustainability** is the ability to continually improve sustainability impacts by reducing energy consumption and increasing efficiency across all components of a workload by maximizing the benefits from the provisioned resources and minimizing the total resources required.

To facilitate good design for sustainability:

* Understand your impact
* Establish sustainability goals
* Maximize utilization
* Anticipate and adopt new, more efficient hardware and software offerings
* Use managed services
* Reduce the downstream impact of your cloud workloads
* **Advantages of Cloud Computing**

Operating in the AWS Cloud offers many benefits over computing in on-premises or hybrid environments.

In this section, you will learn about six advantages of cloud computing:

* Trade upfront expense for variable expense.
* Benefit from massive economies of scale.
* Stop guessing capacity.
* Increase speed and agility.
* Stop spending money running and maintaining data centers.
* Go global in minutes
* **Cost Allocation tags**

A tag is a label that you or AWS assigns to an AWS resource. Each tag consists of a key and a value. For each resource, each tag key must be unique, and each tag key can have only one value. You can use tags to organize your resources, and cost allocation tags to 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. AWS provides two types of cost allocation tags, an AWS generated tags and user-defined tags. AWS, or AWS Marketplace ISV defines, creates, and applies the AWS generated tags for you, and you define, create, and apply user-defined tags. You must activate both types of tags separately before they can appear in Cost Explorer or on a cost allocation report.

* **AWS Managed Services**

AWS Managed Services (AMS) helps you adopt AWS at scale and operate more efficiently and securely. We leverage standard AWS services and offer guidance and execution of operational best practices with specialized automations, skills, and experience that are contextual to your environment and applications. AMS provides proactive, preventative, and detective capabilities that raise the operational bar and help reduce risk without constraining agility, allowing you to focus on innovation. AMS extends your team with operational capabilities including monitoring, incident detection and management, security, patch, backup, and cost optimization.

* **AWS Consulting Partners**

APN Consulting Partners are **professional services firms that help customers design, architect, build, migrate, and manage their workloads and applications on AWS**. Consulting Partners include System Integrators, Strategic Consultancies, Agencies, Managed Service Providers, and Value-Added Resellers.

* **Service Quota**

Service Quotas enables you to manage your AWS service quotas from one central location. In addition to viewing service quota values, you can easily request and track quota increases. For supported services, you can proactively manage your quotas by configuring Amazon CloudWatch alarms that monitor usage and alert you to approaching quotas.

* **VPC Peering**

A *virtual private cloud* (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch AWS resources, such as Amazon EC2 instances, into your VPC.

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. Instances in either VPC can communicate with each other as if they are within the same network. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. The VPCs can be in different Regions (also known as an inter-Region VPC peering connection).


            A VPC peering connection
        