AWS Certified

Cloud Practitioner

* Sony Shrestha

AWS allows you to share a single server with other AWS customers using virtualization.

Cloud computing services are delivered via the Internet and managed through web browser.

**Cloud Terminology**

1. Availability
2. Agility
3. Durability
4. Elasticity
5. Availability

* always up and running

1. Agility

* high speed

1. Durability

* data is not corrupted

1. Elasticity

* auto scaling, scale up and scale down

**6 advantages of Cloud Computing**

1. Go global in minutes
2. Stop guessing capacity
3. Stop spending money running and maintaining data centers
4. Increase speed and agility
5. Benefit from massive economic of scale
6. Trade capital expense for variable expense
7. Go global in minutes

* You can deploy your applications around the world at the click of a button.

1. Stop guessing capacity

* Start with what you want then scale-up or scale-down

1. Stop spending money running and maintaining data centers

* No need to manage servers
* Focus on your applications instead of managing hardware

1. Increase speed and agility

* High speed

1. Benefit from massive economic of scale

* Huge discount

1. Trade capital expense for variable expense

* Pay for what you use instead of making huge upfront investments

**Cloud Computing Models**

1. IaaS
2. PaaS
3. SaaS
4. IaaS

* Infrastructure as a Service
* It is the basic building block that you can rent like an EC2 instance.
* Eg: Web hosting

1. PaaS

* Platform as a Service
* Mostly used by developers to build application.
* Eg: Storefront website

1. SaaS

* Software as a Service
* It is complete product or application that you can rent.
* Eg: Email Provider

**Cloud Deployment Models**

1. Private Cloud
2. Public Cloud
3. Hybrid Cloud
4. Private Cloud

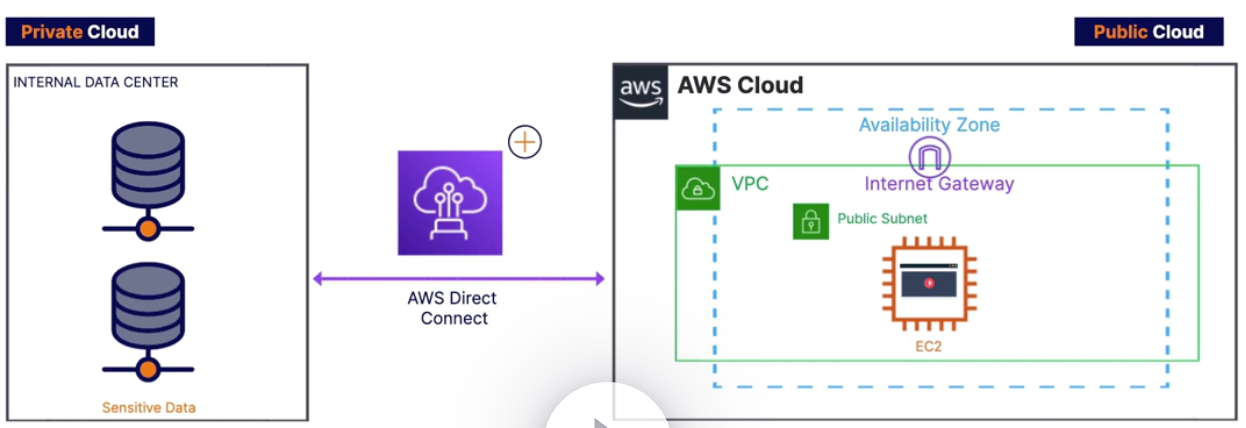
* Called on-premises
* Exists in your internal data center. Everything runs on internal data center
* Does not offer advantage of cloud computing
* There is increased level of security

1. Public Cloud

* Offered by AWS
* Offers advantages of cloud computing

1. Hybrid Cloud

* Combination of private and public cloud
* Highly sensitive data resides in private cloud, application that reads these data run in AWS and they communicate with each other using a service provided by AWS called AWS Direct Connect.



Note:

Hybrid deployments are supported by Direct Connect.

**AWS Global Infrastructure**

1. Availability Zone
2. Region
3. Edge Location

Note:

240 countries

80 availability zone

25 regions

Data Center

Collection of servers

1. Availability Zone

* Consists of one or more physically separated data centers, each with redundant power, networking, and connectivity, housed in separate facilities.
* Characteristics of Availability Zone
  + They are physically separated
  + They use separate power grids
  + They are connected among themselves within a region through low-latency links
  + Fault tolerance- If one Availability zone gets out of service, other should not get impact
  + Allows for high availability
* An availability zone is associated with a single region
* Availability Zone contains server you are renting, and it is where you deploy your applications

1. Region

* Physical location
* Collection of multiple availability zones
* When deploying your application, select region that is closest to users. It improves performance and availability
* Characteristics of Region
  + Fully independent and isolated- if one region gets impact, other will not be affected
  + Resource and service specific- resources are not automatically replicated across regions.

1. Edge Location

* Mini data center
* Cache content for faster delivery of content to users
* There are more edge locations than availability zones and regions
* It is not used to launch resources like EC2 instance. It is just used to cache content.
* Reduces latency and speeds up delivery of your application
* Related to Content Delivery Network (CDN) and Amazon CloudFront

**Ways to access AWS**

1. AWS Management Console
2. AWS Command Line Interface (CLI)
3. AWS Software Development Kit (SDK)
4. AWS Management Console

* Allows you to access your AWS account and manage resources using web browser
* Root user is created when we initially sign up for your account. He has unrestricted access to your account, and it cannot be restricted.
* For best practice, use root user for first time access. After that create separate users for day-to-day activities.
* For best practice, protect root user with MFA (Multi-Factor Authentication) This is where a code is sent to your phone that you then have to enter in order to officially sign into your accounts. MFA is sometimes called 2-factor authentication because you have to enter two things your password and code sent to your phone.
* Following tasks can only be performed by root user:
  + Deleting account
  + Changing email address
  + Changing service plan

1. AWS Command Line Interface (CLI)

* It is called programmatic way to access your AWS account
* Allows you to access AWS account through a terminal or command window on your local laptop
* Mostly used by developers
* Some new features are available via command line before console
* When you set up CLI locally, you will have to do some configurations like generating secret key and access key and then using that access keys locally.

1. AWS Software Development Kit (SDK)

* Access AWS account through application code
* Allows you to access AWS Services from popular programming languages like Java, Python, C# and many more

**Compute Services**

1. EC2 (Elastic Cloud Compute)

**EC2 (Elastic Cloud Compute)**

**Introduction**

* Elastic Cloud Compute
* Virtual server in the cloud
* Allows you to rent and manage virtual servers in the cloud
* Has elastic compute power: It can grow and shrink based on load in your application
* Are not serverless
* You are able to provision an EC2 instance at the click of a button
* You can also use pre-configured template called AMI (Amazon Machine Image) to launch your resources.
* You will receive 750 compute hours per month on Free Tier Plan
* You can deploy your applications directly to EC2 instances.

**EC2 in real world**

1. Deploy a database

Deploy a database to EC2 gives you full control over the database.

1. Deploy a web application

Deploy web application to multiple availability zones to make web application highly available.

**Methods to access EC2 instance**

1. AWS Management Console
2. Secure Shell (SSH)
3. EC2 Instance Connect (EIC)
4. AWS Systems Manager
5. AWS Management Console

* You are able to configure and manage instances via a web browser.

1. Secure Shell (SSH)

* Allows you to connect to your instance from local laptop using SSH Client and keys
* Puttygen converts .pem(privacy enhanced mail) file to ..ppk (putty private key)

1. EC2 Instance Connect

* Allows you to connect to instance without using SSH client and keys and directly using terminal in web browser
* For this, you must grant IAM users permission to push public key to the instance.

1. AWS System Manager

* Allows you to manage EC2 instances via a web browser or CLI

Note

The must common way to connect to Linux EC2 instance is via Secure Shell (SSH)

For this key pair is generated.

A key pair, consists of a private key and a public key, which proves your identity when connecting to an EC2 instance.

When user connects to SSH Client laptop, he uses private key. When he connects to EC2 instance, he uses public key.

**EC2 Pricing Model**

1. On demand
2. Spot
3. Reserved Instance
4. Dedicated Hosts
5. Savings Plans
6. On demand

* Fixed price in which you are billed based on instance type used
* You pay for what you use
* No contract
* Use on-demand instances when
  + You cannot make upfront payment or long-term commitment
  + Your applications have unpredictable workloads that cannot be interrupted
  + Your applications are under-development
  + Your applications will not run for more than a year

1. Spot

* Spot instance lets you take advantage of unused EC2 capacity
* Your request is available only if capacity is available
* Use Spot instance when
  + Your application is not concerned about start and stop time of execution
  + Your workload can be interrupted
* You can save upto 90% off on-demand prices
* You pay for spot that is in effect at the beginning of each hour

1. Reserved Instance

* Allows you to commit to a specific instance type for 1 or 3 years
* Use reserved instance when
  + Your application has steady state usage and you can commit for 1 or 3 years
  + You can make upfront payment
  + Your application requires capacity reservation
* You can save upto 75% off on-demand prices
* You are required to sign a contract of 1 or 3 years
* You can pay all-upfront, partial upfront or no upfront. All upfront for maximum term earns the highest discount

1. Dedicated Hosts

* Allows you to pay for physical server that is dedicated to running your instances. No other AWS customers is going to have applications running on this server. The server basically belongs to you
* Use dedicated hosts when
  + You want to bring your own server-bound software license from vendors like Microsoft and Oracle
  + You have regulatory or corporate compliance requirements around tenancy model
* You can save upto 70% off on-demand prices
* You can bring your existing per-core, per-socket and per-VM software licenses.

1. Savings Plan

* You do not commit for specific instance type instead you commit to compute usage (measured per hour) for 1 to 3 years
* Use savings plan when
  + You want to lower your bill across multiple compute services
  + You want flexibility to change compute services, instance types, OS or regions
* You can save upto 72% off on-demand prices
* Savings can be shared across various compute services like EC2, Fargate and lambda

Features

1. Elastic Load Balancing
2. EC2 Autoscaling
3. Elastic Load Balancing

* Automatically distributes your incoming traffic across multiple EC2 instances
* Types
  + Classic Load Balancer
  + Application Load Balancer
  + Gateway Load Balancer
  + Network Load Balancer

1. EC2 Autoscaling

* Scale up or scale down based on workload in your application
* Reduces impact of system failure and improves availability of your applications
* You can also use autoscaling with Aurora and DynamoDB

Note

Horizontal Scaling vs Vertical Scaling

Horizontal Scaling (Scaling out)- adds or replaces instances

Vertical Scaling (Scaling up)- upgrades an existing instance

Tag

Allows you to track your instance by adding key/value pair

Eg: We can add a tag to specify “This instance is dedicated for Management purpose.”