EC2

Elastic Compute Cloud (Ec2):

Amazon Elastic Compute Cloud is a web service offered by Amazon web serivces (AWS) that provides resizable compute capacity in the cloud.

- Resizable means, you can quickly scanle out and scale in or scale up and scale down to meet the demands of your application.

> Horizontal Scaling means we are increasing the number of instances.

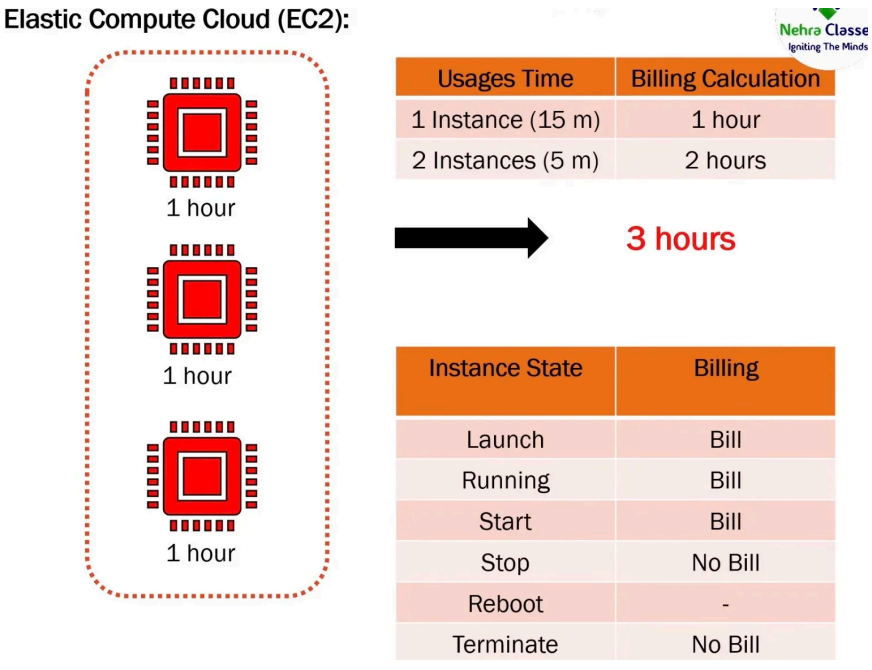
> Vertical Scaling means we are increasing the capacity (CPU,RAM...)of particular instances

- EC2 is a regional service, which works on pay as you go.
- EC2 instances are virtual servers in the cloud that you can use to run your application.
- In Ec2, you can use an Amazon Machine Image (IAM) which is pre-configured virtual machine image that includes an Operating system and often additional software.
- In EC2, we use security Groups which act as virtual firewalls for your instances to control inbound and outbound traffic.

- Key pairs are used to securely SSH into your EC2 instances.
- ELB is a Service that automatically distributes incoming application traffic across multiple EC2 instances.
- Auto Scaling helps you maintain application availability by automatically adjusting the number of EC2 instances based on changes in demand.
- EBS provides BLOCK-LEVEL Storage volumes for use with EC2 instances.
- EC2 instances are billed on an hourly or pre-second basis, depending on the instances type. Pricing can vary based on the instance type, region, and other factors.
- In a free tier account compute capacity is calculated on hourly basis.

> if you run 1 instance for 1 hours then amazon calculate on 1 hours basis and if you run 3 instances for 1 hours then amazon calculate total 3 hours.

> 1 instances for 10,15,19... mins which means it run for 1 hours according to amazon



EC2 Pricing Models:

On-Demand Instances: Pay for Compute capacity by the hour or second with no upfront payments (advanced payments) or long-term commitments. Ideal for short-term, unpredictable workloads where flexibility is crucial.

Reserved Instances:Reserved capacity for a specific term (1 or 3 years) with significant discounts compared to on-demand pricing. Requires an upfront payment (advanced payment) or a higher hourly fee. Suitable for stable ,predictable workloads with a commitment to a specific instance type.

Spot Instances: Bid for un-used EC2 capacity at potentially lower prices. Instances can be terminated if the capacity is needed by on-demand or reserved instances. Best for fault-tolerant and flexible application.

Dedicated Hosts: Physical servers dedicated for your use. Can be purchased on demand or reserved for 1 or 3 years. Provides full control over the placement of instances. Ideal for compliance requirements or software licensing.

Saving plans: Commit to a consistent amount of usage (measured in $/hr) for a 1 or 3-year period. Offers significant savings over on-demand pricing.provides flexibility across a wide range of instances types and families.

Ec2 Instances Types:

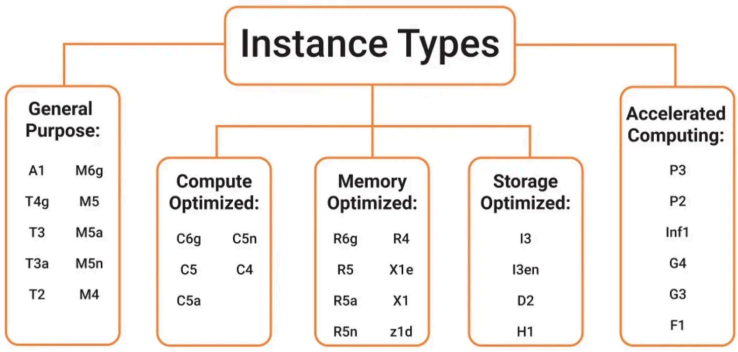
General Purpose: Suitable for general purpose usages.

Compute Optimized: Suitable for applications requiring more CPU.

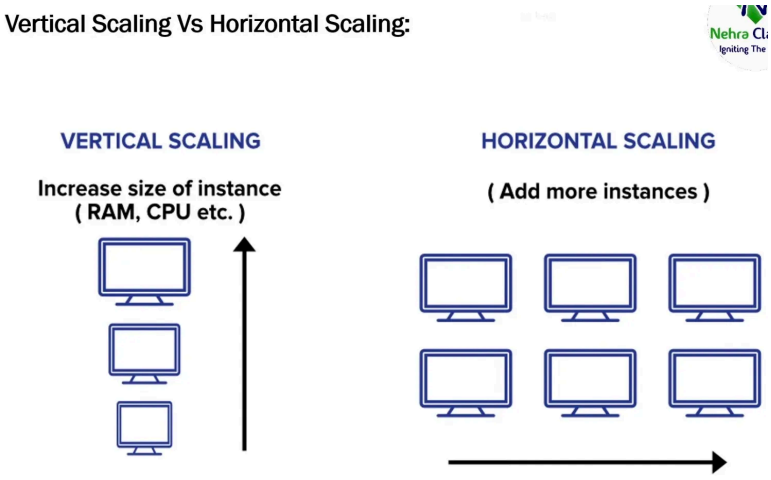
Memory Optimized: Suitable for applications requiring more RAM.

Storage Optimized: Suitable for apps requiring more storage.

Accelerated Computing: Suitable for heavy usages like video editing, graphics, animations etc.



EC2 Scalability:



Amazon Elastic Compute Cloud (EC2) provides scalable compute capacity in the cloud. Scalability in the context of Ec2 refers to the ability to easily and efficiently increase or decrease the compute resources allocated to your application based on demand.

- **Scalability:** Changing the capacity of the server (hardware resources like no. of CPUs, RAM and storage) is called as scalability or **vertical scaling.** Increasing the number of hardware resources is know as scale up and decreasing is know as scale down.
- In EC2, changing the capacity of the server means changing the type of Ec2 instances.
- Scalability can be achieved by simply changing the instance type.

- Changing the capacity of an instances in AWS, specially when performing vertical scaling (resizing an instances), does not typically cause data loss.
- Vertical Scaling involves changing the instances type to one with different compute, memory, or storage capacity but it should not affect the data stored on the instance's storage volumes (EBS).
- You need to stop the EC2 instance to change the instance type (for Vertical Scaling) and downtime is required if HA (High Availability) is not configured.
- In case if there is only one instances which means there not no high availability and we don't want to have downtime during vertical scaling, we can go for burstable performance instances.
- Burstable instances are a type of Amazon Ec2 instance that provide a baseline level of CPU performance.
- It is a chargeable service of AWS, which works on pay as you go model.

Burstable Instances:

Burstable performance instances in Amazon Web Service refers to a category of instances that allow you to use additional CPU resources beyond your baseline level for temporary periods when needed.

- These instances are designed for workloads with varying level of CPU usage, providing flexibility and cost-effectiveness.
- Burstable instances mainly come in the **T family**. This includes standard ones like **T2 and T3**, and the latest version called **T4g**
- One CPU credit equals one vCPU running at 100% utilisation for one minute or an equivalent combination of vCPUs, utilisation, and time (for example, one vCPU running at 50% utilisation for two minutes or two vCPUs running at 25% utilisation for two minutes).
- An Ec2 instances will enter into burstable mode and give high performance for a limited period of time.
- CPU credits depends on the type of EC2 instances.

▼ Volume

Ec2 volumes come with some nifty features to make your storage game strong:

Elasticity: You can easily increase the size of your volumes or change the volume type without much hassle. Flexibility is the name of the game.

Snapshotting: Take snapshots of your volumes for data backup and replication. It's like capturing the essence of your data at a specific point in time.

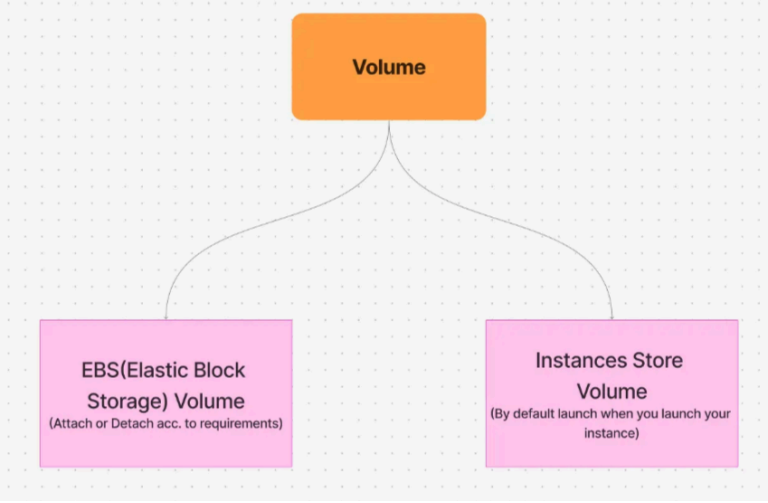
Encryption: AWS allows you to encrypt your volumes for added security. Keep those sensitive bits and bytes safe and sound.

Performance Options: Depending on your needs, you can choose different volume types like general purpose {SSD}, Provisioned IOPS{SSD}, or Throughput Optimized {HDD}. Tailor your performance to match your workload.

Attachment and Detachment: Attach Volumes to EC2 instances when you need them and detach when you are done. It's like plugging and unplugging an external hard drive, but in the cloud.

Multi-Attach: Some volumes support attaching to multiple instances simultaneously. Perfect for scenarios where shared access is needed.

Lifecycle Management: Set up lifecycle policies to automate the creation and deletion of snapshots. Let AWS handle the housekeeping.



Instance Store Volumes:

- Instance store volume are ephemeral (temporary) storage volumes. (data is lost on stop and start the instance, it's different than reboot).
- The maximum size of an instance store volume depends on the instance type you are using.
- Instance store volumes are free.
- Terminating the EC2 instance will by default delete the root volume because delete on termination is checked/enabled.

→In order to retain the root volume make sure you uncheck the delete on termination option.

→Terminating the Ec2 instance will by default not delete the additional volumes because delete on termination is unchecked/disabled.

→Instance store volumes are attached automatically based on the type of instance you are launching.

Stop & Start vs Reboot:

Stopping an Ec2 instance and starting it again is different from rebooting. When you stop an EC2 instance, it is completely terminated ***(process)***, and you lose the public and private Ip addresses.The data on the instance's root volume is preserved.

- ***Public IP and DNS:*** Stopping and starting an instance will cause the public ip and public DNS to change, unless you have assigned an Elastic Ip to it. However rebooting will not cause the Ip to change.
- ***Data:*** Stopping and restarting an instance will cause the loss of any temporary data stored in its ephemeral storage.
- ***Status check:*** Stopping and restarting an instance will initialize all status checks.
- ***Hardware:***When you stop your instance, AWS releases the physical server it was using. Your instance is no longer tied to that specific hardware.

- ***Migration:*** The “STOP” and “START” commands from the AWS console are the only way to allow an instance to migrate to another host on the backend.

Status Checks

In Amazon Web service (AWS) EC2, there are two types of status checks that are performed on instances: **System status checks** and **Instance status checks.**

1.System Status Checks: (Hardware check)

→**Purpose:** These checks monitor the health of the underlying infrastructure of your Ec2 instance, including the host computer and the network

→**Examples of Checks:** Verifying the correct functioning of the host computer’s hardware, network connectivity, and other infrastructure-related aspects.

→**Action Taken:** If a system status check fails, AWS automatically attempts t recover the instance by migrating it to a healthy host.

2. **Instance Status Check: (Software Check)**

→**Purpose:** These checks focus on the instance itself and ensure that operating system is functioning as expected.

→**Examples of Checks:** Verifying the correct boot process, Checking for any system level issues within the instances.

→**Action Taken:** If an instances status checks fail, AWS doesn’t automatically recover the instances. You may need to troubleshoot and manually address the issus.

Status check:

Meaning of status checks and troubleshooting:

→2/2 checks are passed: Everything is OK, we can login into the machine and use it.

→1/2 or 0/2 checks are passed: Something is wrong, we can't login into the machine.

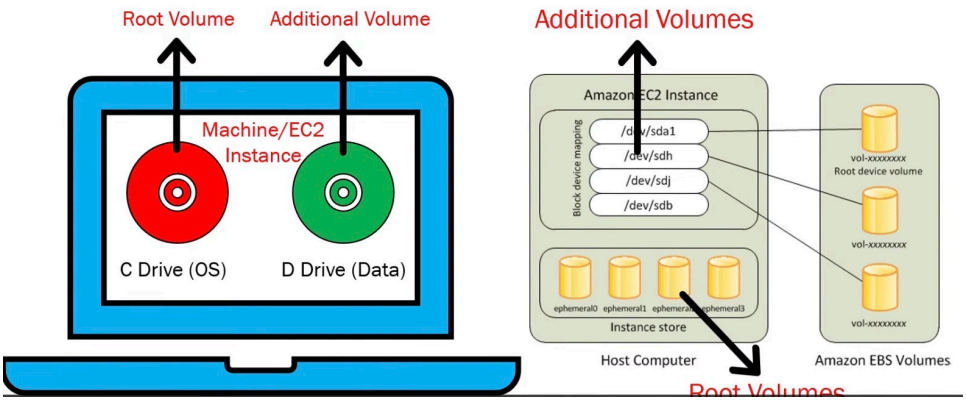
Troubleshooting:

- If it is a newly created machine, we can terminate it and launch a fresh one.
- If it is an old machine, we can stop it and start it again so that it can migrate to another host machine.
- If you still can't resolve the issue after these steps, you may need to gather more information from logs and error messages or reach out to AWS support for assistance.

▼ Snapshot

A snapshot is a point-in-time copy of a data volume or file system. In context of cloud computing, such as Amazon Web services (AWS) or Google Cloud Platform (GCP), a snapshot typically refers to a backup mechanism for storage volumes.

Creating snapshots of Amazon Elastic Block Store (EBS) volumes in Amazon Ec2 is common practice for data backup and disaster recovery purposes.



We don't take snapshots of Instance Store because it is **temporary storage**. If data is **crucial**, it should be kept on **EBS**. AWS provides a built-in snapshot tool for EBS, but for Instance Store, there is no easy way to take a 'point-in-time' copy because the data is lost as soon as the instance stops.

"Important data Instance Store mein rakhna hi galat architecture hai, use EBS par move karna chahiye."

▼ Speed (IOPS and Latency)

- **Instance Store (Super Fast):** Ye physical server ke andar **seedha CPU se juda hota hai** (locally attached). Isliye iski speed bahut zyada hoti hai aur latency (delay) minimal hoti hai. Ye SSD ki tarah fast kaam karta hai.
- **EBS (Fast, but through Network):** EBS ek alag network-attached storage hai. Matlab data ko server se EBS tak ek **"network wire"** ke zariye jana padta hai. Is wajah se ye Instance Store ke muqable thoda slow hota hai.

Ec2 Volumes Snapshot:

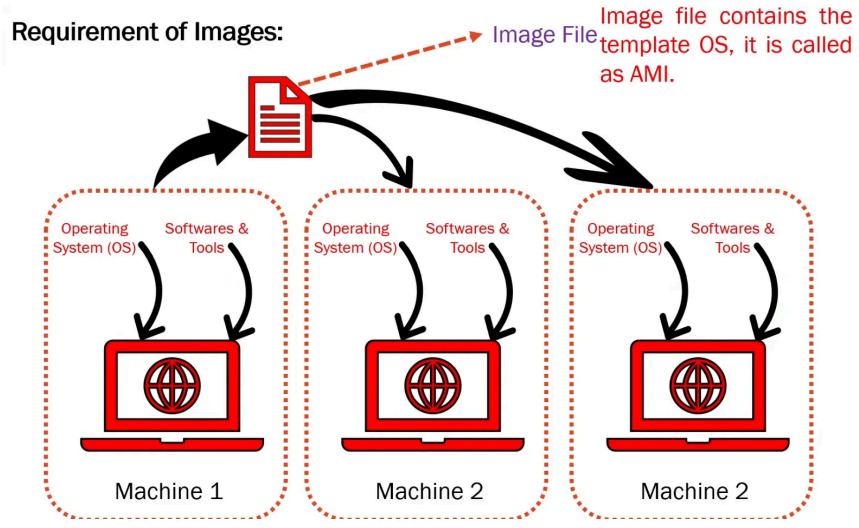
1. Snapshot is a point in time copy of the volume which means it will contain all the data which was available on the volume at when the snapshot was created.
2. Snapshots are used for backup, replication and recovery purpose.
3. EBS snapshots are created from the EBS volume to backup the data.
4. We create the snapshot from the volume for backup purpose and we create the volume from the snapshot for the recovery purpose.
5. We can’t directly attach a snapshot to the EC2 instance, we have to create a volume and then we can attach this volume to the instance.
6. It is not possible to login into the snapshot directly in AWS.
7. Snapshots are stored in AWS S3, when you create an EBS snapshot it is stored as an Amazon S3 object in the AWS region where the snapshot was created.
8. Snapshots are visible to the user from the Ec2 console.
9. Snapshots don’t have any availability zones in AWS.
10. Snapshots are regional in AWS.

11. Snapshots are private by default because of security reasons but we can make them public if we need.
12. Snapshots can be copied from one region to another in the same AWS account.
13. Snapshots can be shared from one AWS account to another privately.
14. EBS volumes can’t be moved from one Az to another directly, this can be done using snapshots only.
15. Instance store volumes (root volumes) are created from a template stored in AWS S3.
16. Snapshots can be created on a running Ec2 instance, no need to stop the machine in order to create a snapshot.
17. We can manually take the snapshot at any tie and we can also use AWS data LIFECYCLE MANAGER to automatically create the snapshots.
18. AWS Life Cycle Manager can manage the snapshots with the help of tags, we cn schedule the snapshots creation and deletion easily.
19. Snapshots not in used can be moved from the standard tier (stored in Amazon S3) to archive tier (Stored in Amazon S3 GLACIER) to save the costing upto 75%.
20. Snapshots in the archive tier can’t be used directly, they are required to be moved to standard tier first.
21. It may take up to 72 hours to restore the volume from the snapshot present in the archive tier.
22. Fast Snapshot Restore (FSR) enables you to quickly provision volumes from your EBS snapshots, allowing you to meet changing workload demands without having to wait for lengthy data transfers from the snapshots to the volumes.
23. FSR feature can be particularly useful in scenarios where you need to rapidly scale your infrastructure or recover from failures, there may be additional charges for using FSR, and you’ll also incur standard EBS volume usage charges.

24. By default volumes and snapshots are not encrypted but we can use encryption if we need using KMS in AWS.
25. EBS Volume **(Not Encrypted)** → EBS Snapshot **(Not Encrypted)**
26. EBS Volume **(Encrypted)** → EBS Snapshot **(Encrypted)**
27. EBS Volume **(Not Encrypted)** → EBS Snapshot **(Encrypted)** ← **copy Option**
28. Encryption and decryption is completely managed by AWS.
29. All the encryption keys are stored in key management service (KMS) of AWS.
30. Access and Secret keys are for the access purpose while the encryption keys are for encryption or security purpose.
31. We can't share the encrypted Snapshot.

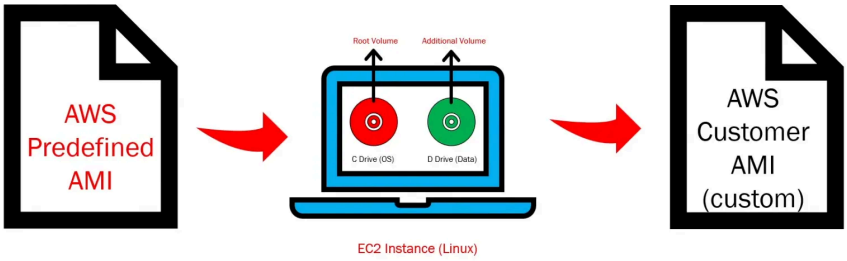
▼ Image(AMI)

Image is a file which contains the copy of the Operating System along with all other additional software and tools.



Ec2 Images(Amazon Machine Images):

We can launch an Ec2 instance using AWS predefined AMI and with the help of this instance we can create our own custom AMI which can have all the data present on this machine like volumes, OS, softwares, tools and other setting present on the machine.



Ec2 images, also know as Amazon Machine Images(AMIs), are templates used to create virtual servers in Amazon Web Services' Elastic Compute Cloud (Ec2). These images are pre-configured with operating systems, software configuration, and other settings, allowing users to quickly launch instances (virtual server) without manually setting up each component.

- AMI may either contain the operating system only or the operating system with other tools and applications.
- AMI contains all the data and setting which were available on the original instances.
- AMI ⇒ Ec2 instance ⇒ AMI ⇒ Ec2 instance.
- A single AMI can be used to launch multiple Ec2 instance.
- AMIs are reusable and doesn't have any availability zones.
- By default, AMI are private, but can be made public if needed.
- AMIs are original.
- AMIs can be copied from one region to another.
- AMIs can be shared from the one AWS account to another using AWS account ID.
- All public images (AMIs) are available at AWS market place.
- You can customize your instance (OS environment settings) and create the AMIs manually which are called as **Custom AMIs.**

- If you want to create the custom images automatically you can use Ec2 image builder and these images are called **golden AWS.**
- You don't need to shutdown the Ec2 instance while capturing the image but it is not recommended. Ideally you should always stop the instance first before you create its image.
- AMIs contain OS, Volumes, tools and settings. Which means AMI can have all the volumes which are present in your instance whose AMI you created earlier.
- Images are backed by either EBS volumes or instance store volumes.

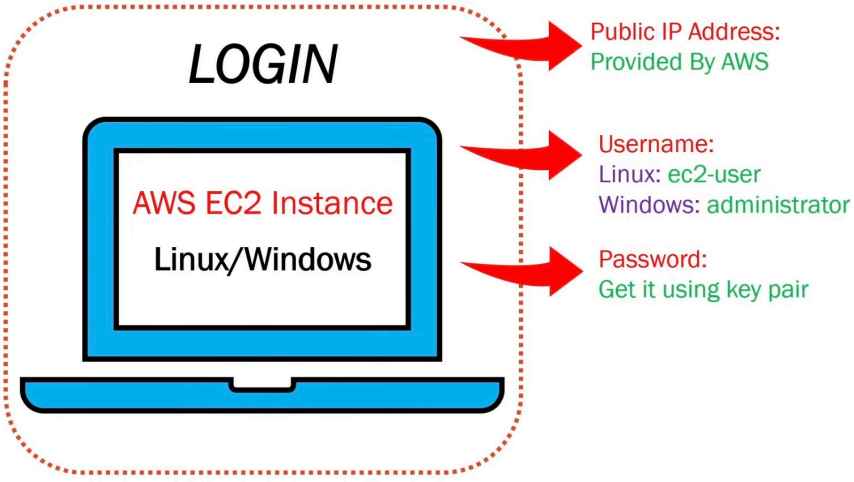
Using EC2 images offers several benefits, including:

1. Efficiency: AMIs streamline the process of launching new instances by providing pre-configured templates, reducing setup time and effort.
2. Consistency: Using standardized images ensures that all instances launched from the same AMI have consistent configurations, which can simplify management and troubleshooting.
3. Flexibility: Users can create custom AMIs tailored to their specific requirements, including specific software configurations, security settings, and other customizations.
4. Reproducibility: AMIs can be easily shared ad replicated, allowing users to reproduce complex setups reliably across multiple instances or environment.

Overall, Ec2 images play a crucial role in the scalability, flexibility and efficiency of AWS Ec2 instances, making it easier for users to deploy and manage virtual server in cloud.

▼ Key pair

An AWS Ec2 key pair consists of a public key and a private key. when you launch an Ec2 instance, you specify the key pair to use, and AWS stores the public key while you retain the private key.

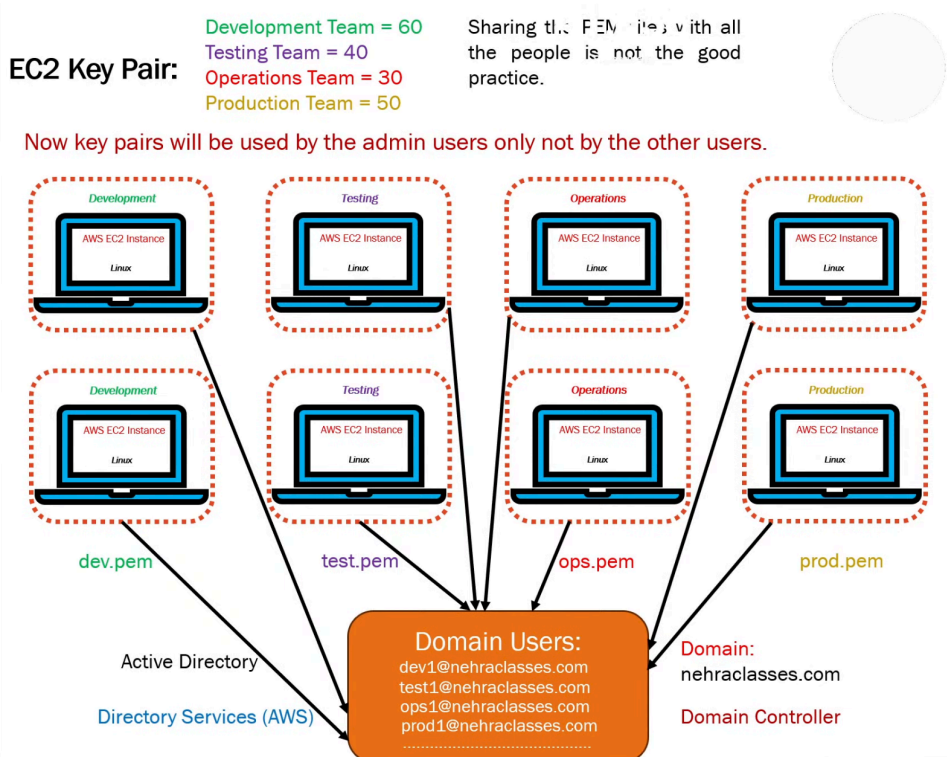


- In AWS, a key pair is a set of cryptographic keys used to authenticate with instances (virtual servers) running on the AWS platform.
- This key pair consists of a public key that AWS stores, and a private key file that you download to your local machine.
- By default there is no key pair present, we create it manually.
- When you create an EC2 instance (Elastic Compute Cloud, AWS's virtual server service), you typically specify a key pair.
- AWS uses this key pair to encrypt the password that allows you to connect to your instance securely.
- The extension of the key pair file downloaded from AWS is typically '.pem' which stands for Privacy Enhanced Mail. This file format is commonly used for storing cryptographic keys.

Question (Based on your scenario):

"Ek company mein **180+ employees** hain (60 Dev, 40 Testing, 30 Ops, 50 Production). Agar hum sabhi ko unke respective servers ke liye `.pem` files distribute karte hain, toh isse kya security risks honge? Iska professional

solution kya hai jisme hum **AWS Directory Services** aur **Active Directory** ka use karein?"



Answer in Hinglish:

Aapka sochna bilkul sahi hai ki itne saare logo ko private keys dena **"Not a good practice"** hai. Iske piche main reasons ye hain:

1. Security Risk (Key Leakage):
Agar 180 logo ke paas `.pem` files hongi, toh kisi na kisi se wo file leak ho sakti hai, galti se GitHub par upload ho sakti hai, ya laptop chori hone par server ka access chala jayega.

2. No Central Control:
Agar koi employee company chhod kar jata hai, toh aapko saare servers ki keys badalni padegi, jo ki impossible kaam hai.

Professional Solution: AWS Directory Service (Active Directory)

Jaisa ki aapke diagram mein dikhaya gaya hai, badi companies **Domain-based login** use karti hain.

- **Centralized Login:** Bajaye `.pem` file ke, har user (jaise `dev1@nehraclasses.com`) apne khud ke **Corporate Username aur Password** se login karta hai.
- **Active Directory (AD):** Ye ek database ki tarah kaam karta hai jahan saare users ki details hoti hain.
- **Role-Based Access:** Hum AD mein groups bana dete hain.
  - Dev team ke 60 log sirf **dev.pem** wale environment mein apne username se login kar payenge.
  - Production team ke 50 log hi **prod.pem** wale servers ko touch kar payenge.
- **Admin Access:** Sirf **Admin users** ke paas hi asli `.pem` files hoti hain (emergency ke liye). Baki normal users ko key ki zarurat hi nahi padti.

Summary of your Diagram:

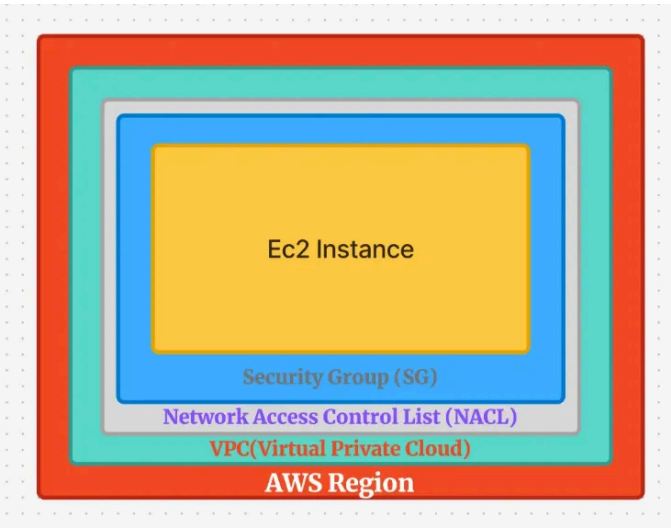
Aapke image ke mutabiq:

1. **dev.pem, test.pem, ops.pem, prod.pem** sirf Admin ke paas rahegi.
2. Baaki saare users (**Domain Users**) AWS Directory Service ke through apne email-id se login karenge.
3. Isse security bani rehti hai aur kisi ko bhi key share karne ki tension nahi hoti.

▼ Security Groups

Security Groups is nothing but similar to firewall that we use in our devices, machines or networks for securing the device/network.

In AWS, security groups prevent unauthorised access to our Ec2 instances.



- Security groups control the incoming as well as outgoing traffic to and fro from an AWS EC2 instance on the basis of the inbound and outbound rules respectively.
- Inbound rule in SG decides which traffic is allowed to come in the EC2 instance.
- Outbound rule in SG decides which traffic is allowed to go out of the EC2 instance.
- By default inbound rules are denied which means there is no inbound rule present in SG by default which means no traffic is allowed to come in to the EC2 instance.
- By default outbound rules are allowed which means all the traffic allowed to go out of the EC2 instance.
- In security groups, we can only allow the protocols (traffic from the ports) not deny, because the default action for the incoming traffic in security group is deny.
- In order to deny the traffic which are allowed earlier (means inbound rule was added for the same) just remove the rule from the SG.
- Every EC2 instance in AWS should have at least one security group attached with it.

- You can assign multiple security groups to an Amazon Elastic Compute Cloud (EC2) instance.
- You can specify one or more security groups for each EC2 instance, with a maximum of five per network interface.
- You can also attach a security group to multiple instances in a VPC.

When configuring security group rules, the "source type" refers to the type of entity that is allowed to send inbound traffic to the associated AWS resource.

- **CIDR IP:** You can specify a range of IP addresses in CIDR notation as the source of inbound traffic.” `Kisi specific IP address ya puri range (jaise aapka ghar ka internet IP) ko permission dena.` ”
- **Security Group:** You can specify another security group within the same AWS account as the source of inbound traffic.

`Jab hum ek Security Group ko dusre Security Group ke andar "Source" ki tarah allow karte hain`

- `Logic:` Iska matlab hai ki aap specific IP address ke bajaye pura ka pura `Security Group allow` kar rahe ho.
- `Usage:` Ye zyada tar `Multi-tier architecture` mein kaam aata hai (jaise Web Server ko Database se connect karna).
- `Fayda:` Agar aap Web-SG mein naye instances add karte ho, toh aapko Database ke Security Group mein baar-baar changes nahi karne padte. Wo automatically allow ho jate hain.
- **Prefix List:** AWS Prefix Lists are sets of CIDR blocks maintained by AWS. This is useful when you need to allow inbound traffic from AWS services or regions.
- **Self:** This option allows inbound traffic from the same security group. It's commonly used when applications on the same instance need to communicate with each other.

`Jab hum ek Security Group ko apne hi andar "Source" ki tarah allow karte hain.`

- `Logic:` Iska matlab hai ki `usi same Security Group` ke saare instances aapas mein ek-dusre se baat kar sakte hain.
- `Usage:` Ye tab use hota hai jab ek hi team ke servers ya ek hi application ke multiple nodes ko aapas mein `internal communication` karna ho.

- `Fayda:` `Aapko har instance ka private IP manually allow karne ki zarurat nahi padti. Group ka member hona hi kafi hai.`
- **Load Balancer:** For traffic originating from an Elastic Load Balancer (ELB), you can specify the ELB as the source type.
- **Any:** This option allows inbound traffic from any source. It's the least restrictive option and should be used cautiously.
- If you allow any protocol in the inbound rule, you don't need to allow the same in the outbound rule. This is known as stateful rule in AWS.
- If you allow any protocol in the inbound rule and there a requirement of allowing the same in the outbound rule. This is known as stateless rule in AWS.
- **Stateful Rules:** By default, all rules in an AWS security group are stateful. This means that when you allow inbound traffic for a specific protocol and port, the corresponding outbound traffic for the same protocol and port is automatically allowed, regardless of any explicit outbound rules. AWS automatically tracks the state of connections and allows return traffic.
- **Stateless Rules:** In contrast, stateless rules are rules where inbound and outbound traffic must be explicitly defined. If you create a stateless rule to allow inbound traffic on a specific port, outbound traffic for that same port is not automatically allowed. You would need to explicitly define an outbound rule to permit the return traffic.
- In AWS security groups, rules are by default stateful.
- In AWS Network Access Control List (NACL), rules are by default stateless.
-

▼ AUTO SCALING AND LOAD BALANCER

🤑 Auto Scaling and Load Balancing

▼ IP Addresses

An Internet Protocol Address, is a numerical label assigned to each device connected to a computer network that uses the internet protocol for

communication. It serves two main purposes: identifying the host or network interface and providing the location of the device in the network.

- Ip addresses come into two primary versions : IPV4 and IPV6.

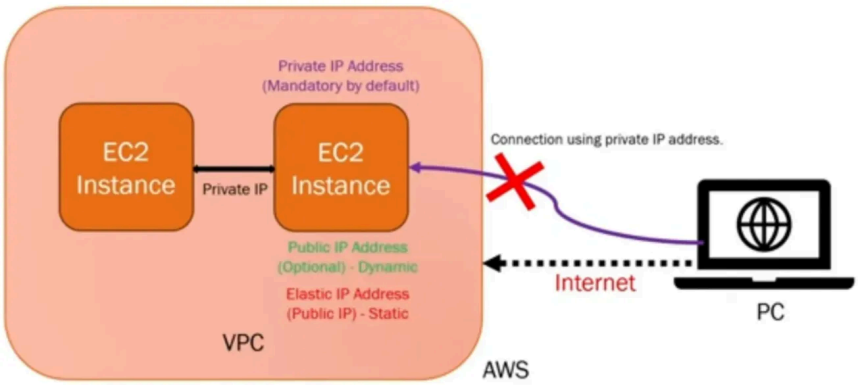
→**IPv4 (Internet Protocol version 4):** This is the most commonly used IP version. IPv4 addresses are 32-bit numerical addresses expressed as four octets separated by periods (e.g., 192.168.0.1). However, due to the limited number of available IPv4 addresses, IPv6 was introduced.

→**IPv6 (Internet Protocol version 6):** IPv6 was developed to address the exhaustion of IPv4 addresses. It uses a 128-bit address scheme, offering a vastly larger pool of addresses. IPv6 addresses are written in hexadecimal & separated by colons (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334).

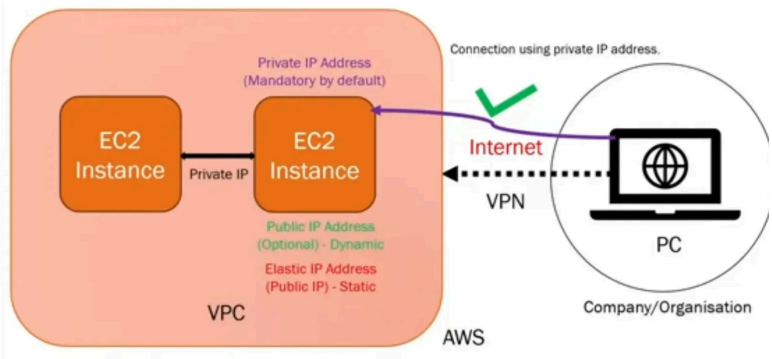
- IP addresses are used for various purposes, including identifying devices on a network, routing traffic across the internet, and enabling communication between devices.
- They can be static (unchanging) or dynamic (assigned).
- **1. Based on capacity:**
  - IPv4 Address - 32 bit number (192.168.0.1)
  - IPv6 Address - 128 bit number (2001:0db8:85a3:0000:0000:8a2e:0370:7334)
- **2. Based on networks:**
  - Private IP Address - used in a private network such as LAN.
  - Public IP Address - used on internet.
- **Based on allocation:**
  - Static IP Address (unchanging/fix) - manual allocation.
  - Dynamic IP Address (changing) - DHCP allocation.
- **Elastic IP Address (EIP):** static IPv4 address (Cloud Computing).
- **Loopback IP Address:** (127.0.0.1 in IPv4 and ::1 in IPv6).

`“Loopback IP ka matlab hota hai` **`"Aapka apna computer"`** `. Iska use tab hota hai jab aapka computer khud se hi baat karna chahta hai.`

- **Self-Testing:** Agar aapne apne computer par koi website ya service banayi hai aur aap check karna chahte ho ki wo chal rahi hai ya nahi (bina internet use kiye), toh aap **127.0.0.1** use karte ho.
- **"Localhost" ka dost:** Is IP ko aksar **"localhost"** bhi kaha jata hai.
- **Kaise kaam karta hai:** Jab aap is IP par data bhejte ho, toh wo network ke bahar nahi jata. Wo aapke computer ke network card se hi "loop" hokar wapas aapke paas aa jata hai.
- **Example:** Jaise aap khud ko hi chitthi likh rahe ho. Aapko postman ya bahar ke kisi raste ki zarurat nahi hai, aapne khud likha aur khud padh liya."



- Public Ip address is required to access the EC2 instance using internet.
- Public Ip address is dynamic and gets changed on start and stop.
- So we need Elastic Ip address which remain static (do not change).



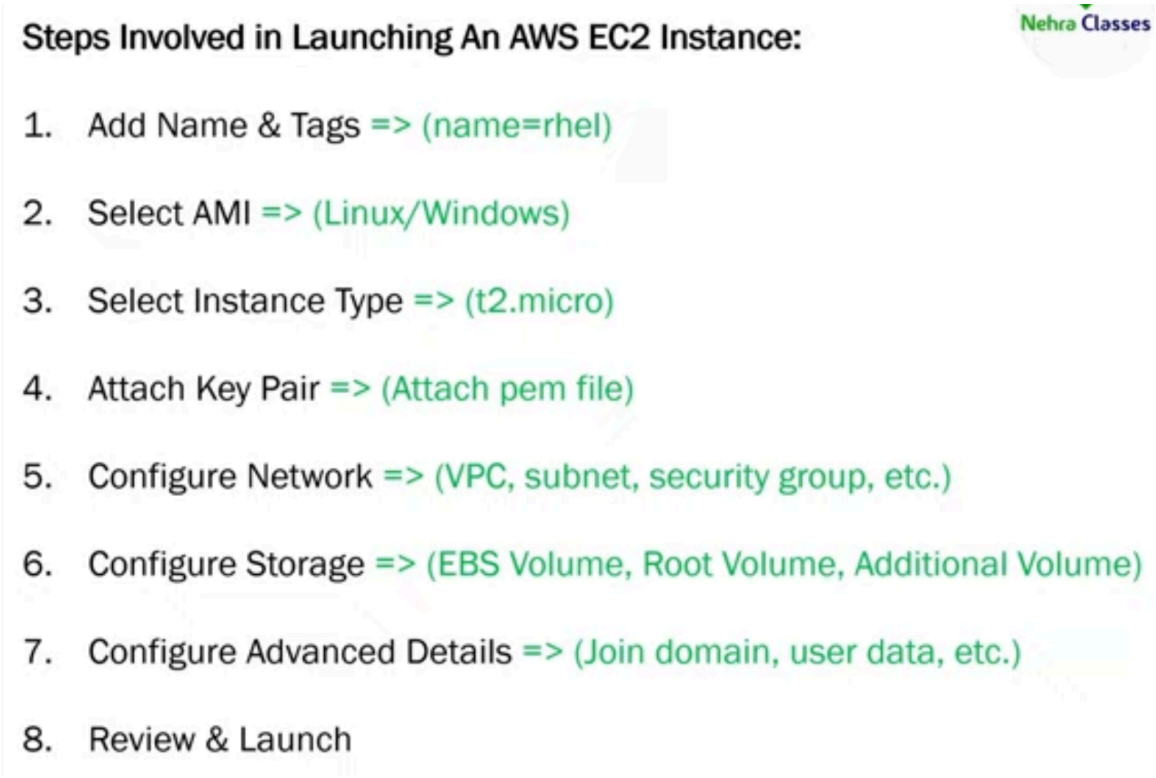
- We can directly connect to the our EC2 instance on AWS directly from the company network using the private IP address through VPN.
- Public Ip/ Elastic Ip address can also be used for this purpose.

•

VPN ke through access kaise hota hai? (Step-by-Step)

1. **Secure Tunnel:** Jab aap apne laptop par VPN on karte ho, toh woh aapke ghar/office ke internet aur AWS ke VPC (Virtual Private Cloud) ke beech ek **"Secret Tunnel"** bana deta hai.
2. **Part of the Network:** VPN connect hone ke baad, aapka laptop aisa behave karta hai jaise woh internet par nahi, balki AWS ke usi internal network ka hissa ho.
3. **Direct Communication:** Kyunki ab aapka laptop AWS ke network ke "andar" aa gaya hai, isliye ab aap EC2 instance ko uske **Private IP** se directly ping ya connect kar sakte ho.

Public IP Address	Private IP Address	Elastic IP Address
It can be accessed from anywhere from internet.	It cannot be accessed from internet (works on LAN only).	It can be accessed from anywhere from internet (like Public IP).
Public IP address is optional in AWS.	Private IP address is mandatory in AWS.	Elastic IP address is also optional like Public IP.
Public IP address is dynamic (change).	Private IP address is generally static (fix).	Elastic IP address is also static (fix/unchanged).
Stop & start will change the Public IP address.	Stop & start won't change the Private IP address.	Stop & start won't change the Elastic IP address.
Works globally on internet.	Works in VPC and through VPN only.	Works globally on internet.
AWS charges $0.005 per IP per hour for all public IPv4 addresses.	Private IP addresses are in AWS.	AWS charges $0.005 per IP per hour for all public IPv4 addresses.



▼ Global Accelerator

1. Core Networking Definitions (English)

- **Unicast:** A standard networking method where data is sent from one specific source to one specific destination. Each device has a unique IP address, and the traffic follows a fixed path to that single receiver.
- **Anycast:** A routing technique where multiple servers (nodes) across different geographic locations share the exact same IP address. The network automatically directs user traffic to the node that is "closest" in terms of network hops or latency.
- **AWS Global Accelerator:** A service that provides you with static Anycast IP addresses that act as a fixed entry point to your application hosted in one or more AWS Regions. It uses the AWS global private network to accelerate your users' traffic.

1. Unicast (One-to-One)

Unicast internet par communication ka sabse purana aur basic tarika hai.

- **Technical Logic:** Isme sender aur receiver ke beech ek **unique path** hota hai. Agar ek server Indore mein hai aur aap Delhi mein ho, toh aapki request sirf Indore wale server ke IP par hi jayegi.
- **IP Binding:** Ek IP address sirf ek hi physical interface ya device se bind hota hai.
- **Example (DevOps Context):** Jab aap kisi EC2 instance ko uske **Public IP** se SSH karte ho, toh wo Unicast hai. Aap ek specific machine se baat kar rahe ho.

2. Anycast (One-to-Nearest)

Anycast thoda complex hai kyunki isme "Routing Logic" ka use hota hai.

- **Technical Logic:** Isme **ek hi IP address** duniya ke alag-alag data centers mein baithe multiple servers ko de diya jata hai. Jab aap request bhejte ho, toh "BGP" (Border Gateway Protocol) decide karta hai ki aapke network se sabse kam "hops" ya sabse chhota rasta kis server tak hai.
- **Latancy Benefit:** Anycast ka main maqsad latency (delay) kam karna hai. Agar server Mumbai aur New York dono jagah hai, toh India wale user ki request Mumbai wale server par hi jayegi, halaki dono ka IP same dikhega.
- **Failover (High Availability):** Agar Mumbai wala server down ho jata hai, toh Anycast apne aap user ko agle sabse paas wale server (maan lo Singapore) par bhej deta hai, bina IP change kiye.

> **Unicast:** "Aapne ek specific mobile number par call kiya."

> **Anycast:** "Aapne 100 number (Police) par call kiya. Aap kahin bhi ho, call hamesha aapke local police station hi jayegi."

3. Detailed Explanation: How it Works? (Hinglish)

Global Accelerator aur Anycast milkar ek **"VIP Express Way"** banate hain. Iske kaam karne ka tarika ye hai:

The "Anycast" Role (The Smart Entry)

Anycast ka main kaam user ko sabse pehle AWS ke network mein ghusana hai.

- Kyuki Global Accelerator ka IP **Anycast** hai, isliye wo IP puri duniya ke har AWS Edge Location (entry point) par dikhta hai.
- Jab koi user request bhejta hai, toh internet use kisi door wale server par bhejne ke bajaye, uske ghar ke sabse paas wale AWS Edge Location par bhej deta hai.

The "Accelerator" Role (The Private Highway)

Ek baar traffic AWS ke Edge Location mein enter ho gaya, toh asli magic shuru hota hai.

- Ab traffic normal "Public Internet" (jispe bahut traffic hota hai) par nahi chalta.
- Wo AWS ke apne **Private Fibre Network** ka use karta hai. Ye highway ki tarah hota hai jahan koi traffic signal nahi hota, isliye speed bahut badh jati hai.

4. Reason: Why 2 Static IP Addresses?

Aapne poocha ki 2 IP hi kyu? Iske peeche 3 bade reasons hain:

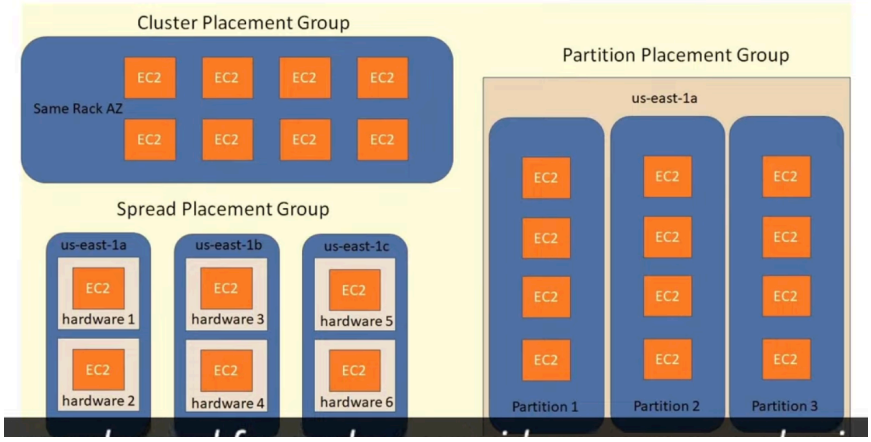
1. **Fault Tolerance (Backup Plan):** AWS aapko 2 IPs isliye deta hai taaki agar internet ke ek raste (Network Path) mein koi badi problem aa jaye ya wo block ho jaye, toh user automatically dusre IP ka rasta pakad le. Isse aapki application kabhi down nahi hoti.
2. **High Availability:** Ye dono IPs alag-alag infrastructure par chalte hain. Agar AWS ke ek system mein update ya maintenance chal raha ho, toh dusra IP hamesha traffic handle karne ke liye taiyar rehta hai.
3. **Firewall Whitelisting:** Badi companies apne security firewall mein IP addresses ko fix (whitelist) karna chahti hain. Kyuki ye dono IPs **Static** hain (kabhi nahi badalte), isliye companies ko baar-baar apni settings nahi badalni padti.

▼ Placement Group

- In Amazon's Elastic Compute Cloud (EC2), placement groups offer a way to influence the physical location of your launched instances on the underlying hardware.

⇒Launched instance

- They act as logical groupings ***(Logical grouping ka matlab yahan ek "Rule" ya "Policy"hai.)*** that define how EC2 positions your instances within an Availability Zone (AZ).



⇒Types of Placement group

1. **Cluster Placement Group:** These Groups aim for low latency network communication by placing all instances within the same AZ, close together on the physical hardware. This is ideal for high-performance computing (HPC) applications that rely on frequent inter-instance communication `.”Agar us ek rack ya hardware mein koi problem aayi, toh saare instances ek saath band ho sakte hain.”`
2. Partition Placement Group:These group spread your instance across logically separate partitions within the AZ. This ensures that instances in one partition don’t share the underlying hardware with instances in other partitions. This strategy is beneficial for fault tolerance and disaster recovery, especially for large-scale distributed applications like Hadoop, cassandra or kafka." `Fault tolerance is the ability of a system to continue operating without interruption even if one or more of its components fail.` ”

3. Spread Placement Group: These groups aim for maximum isolation by placing a small number of instances on distinct physical hardware within the AZ. This helps mitigate(**kisi cheez ke asar (impact) ko kam karna** ya **kisi khatre ko ghatana**.) the impact of hardware failures, as a failure in one piece of hardware wouldn't affect instances in other groups. This approach is suitable for critical applications where even a single instance failure can be detrimental.

⇒Key Points to remember about Placement Group:

1. They are optional. If you don't specify a placement group during launch, Ec2 will distribute your instances across the AZ by default.
2. There is no cost associated with creating or using placement groups.
3. You can launch instances into an existing placement group or create a new one during instance launch.
4. Placement group only control the placement within an AZ; they don't influence the AZ selection itself.

By understanding the different types of placement groups and their functionalities, you can effectively control the physical location of your Ec2 instances, optimizing their performance, fault tolerance, and overall deployment strategy.

▼ Summary

Cluster

Network Speed (Low Latency)High (Ek rack fail = sab fail)

Partition

Large Distributed AppsMedium (Partition level safety)

Spread

Max Safety (Isolation)Low (Sirf ek server ka khatra)