**CLOUD: -** A network of remote servers hosted on the internet to store, manage, and process data, rather than using a local server or a personal computer

* **PUBLIC CLOUD: -** Services are delivered over the public internet and shared across organizations. Examples: -AWS, Google Cloud Platform, and Microsoft Azure.
* **PRIVATE CLOUD:** -Services are maintained on a private network, typically for a single organization, offering greater control and security. Examples: -VMware and OpenStack.

**AWS SERVICES:** -

* **IAM (Identity and Access Management): -** It provides centralized control over AWS accounts, allowing us to manage users, groups, roles, and permissions. we can access what resources the user is using within our AWS environment
* **USERS: -** Individual identities used for authentication when accessing AWS services and resources.
* **GROUPS: -** Collections of users that share the same set of permissions, making it easier to manage permissions for multiple users simultaneously.
* **POLICIES: -** permissions are attached to users, groups, or roles to specify what actions they can perform on which AWS resources.
* **ROLES: -** IAM identities that can be assumed by AWS resources, such as EC2 instances or Lambda functions, to access other AWS services securely.
* **EC2(Elastic Cloud Compute): -** A virtual machine which has the resources that a physical server has. Like memory, CPU, storage etc.
* We can install a hypervisor on top of the physical server to create a multiple virtual machine

1. **GENERAL PURPOSE EC2 INSTANCE**
2. **COMPUTE OPTIMIZED EC2 INSTANCE**
3. **MEMORY OPTIMIZED EC2 INSTANCE**
4. **STORAGE OPTIMIZED EC2 INSTANCE**
5. **ACCELERATED COMPUTE EC2 INSTANCE**

* Inorder to create an EC2-instance and connect, the following fields are mandatory

1. Click on **LAUNCH INSTANCE**
2. Give the **NAME**
3. Select **OPERATING SYSTEM**
4. Select **INSTANCE TYPE**
5. Give the **KEY-PAIR VALUE (Combination of public and the private key)**
6. Select **RSA as KEY PAIR TYPE and give the KEY PAIR NAME**
7. Then select **.PPK{Only for PUTTY} OR .PEM{For OpenSSH(**ex: - MobaXterm**)}**
8. Then click on **LAUNCH INSTANCE**
9. Use “**ssh -i example.pem ubuntu@public\_ip\_of\_ec2instance**” to connect to an ec2 instance through **OpenSSH.** For this we need to be in the location where this particular .pem file exists
10. If it shows “permissions are too open” then change the permissions of the .pem file using **chmod 777 filename**

* **VPC (Virtual Private Cloud): -** A Virtual network allowing us to launch AWS resources, such as Amazon EC2 instances, into a virtual network that we've defined.
* **INTERNET GATEWAY: -** A component of VPC that allows communication between instances in VPC and the internet. Public subnet is connected to the internet gateway
* **PUBLIC SUBNET: -** Component of VPC that divide the ip address range. Public subnet is accessible from the internet.
* **ELASTIC LOAD BALANCER: -** Elastic Load Balancing is a service by AWS that distributes incoming traffic across different resources
* **ROUTE TABLE: -**The route table contain rules that determined how traffic is diverted in VPC
* **SECURITY GROUPS: -**Act as virtual firewalls to control inbound and outbound traffic at the instance level and subnet level
* **PRIVATE SUBNETS: -** Component of VPC that divide the ip address range. Private subnet is not accessible from the internet.
* **NAT GATEWAY: -** Allow instances in a private subnet to connect to the internet or other AWS services, but prevent the internet from initiating connections with the instances. Masked ip will be used to connect to internet
* **SECURITY GROUPS: -** A virtual firewalls that control the inbound and outbound traffic to and from our AWS resources, such as EC2 instances. Allowing us to define the rules that permits the traffic based on our requirements
* **INBOUND RULES: -** The rules which are defined by us based on our requirements for which traffic needs to allow.
* **OUTBOUND RULES: -**The rules which are defined by the AWS itself. By default, it will allow all the ports in outbound rule **except port 25, because it is a mailing port.**
* **NACL (Network Access Control List): -**It is as same as the security groups but NACL has the special quality of blocking the traffic as well as allow the traffic as per our requirements
* **NACL acts as the first level of defense and security groups acts as the second level of defense.**
* **NACL is with the VPC and security groups are with ec2 instances**
* **ROUTE 53: -** It is a domain name system **(DNS).** With which we can use the domain names like **www.example.com** instead of **numeric IP address**
* Allows us to register new domain names or transfer existing domain names into Route 53.
* Continuously monitors the health of our application endpoints.