**AWS**

**In the AWS for the Authentication and Authorization purpose we use IAM Services:**

**It contains:**

**Users**

**Groups**

**Policies**

**Roles**

**Authentication means having the login credentials to enter into the Aws Account.**

**But,Autherization means permissions per accessing all the services for that IAM service come into picture**

**Users:**

**You can create user with out policies you cant access any service for that u need to add policies accessing services**

**Groups:**

**You can create group for multiple users by the the advantage is you can give any types of permissions to all the users instead of giving individually so efficiency will increase.**

**Policies:**

**Through these policies you can access services in the AWS.**

**Roles:**

* + **Roles are used to grant permissions for the services which are run outside of the AWS, roles also have username and password.**
* **Communicate between two AWS accounts**
* **It is like userbut not 100%,roles create for temporary purpose only**

**Day -2:**

**Ec2 🡪what &why & types-🡪 aws provides 750 hours free for instance running in a month.**

**What are Regions and Available zones**

**EC2🡪Elastic Cloud Compute**

**->Elastic means scale up & scale down**

**Only few things have elastic as prefix like**

**Elastic load balancer**

**Elastic Kubernetes Service etc..**

**->cloud menas it is cloud platform**

**>Compute means asking for virtual machine 🡪for CPU,RAM,Disk**

**So final meaning is…you are AWS give virtual machine on cloud platform**

**Why means:**

**Maintaining physical server is too difficult--->timelyupgrade,securityissues,server maintain.**

* **Management headache will reduce by choosing public cloud as well as cost also reduce because it follows pay-as -you go method**

**Different Types:**

* **General Ec2 Instance**
* **Memory optimized Ec2 Instances**
* **Compute Optimized Ec2 Instances**
* **Accelarated Ec2 Instances**
* **Storage optimized Ec2 Instances**
* **Based on our requirement we will go for that instance like**
* **If you want for machine learning models,gaming servers you can go for Compute Instances.**
* **when want to use high performance,big data applications then you go for memory instances.**

**Regions And Availability zones:**

* **at the time of working on AWS you can see different regions it means aws have different physical servers across world**
* **the main reason the users who are using aws sometimes they don’t want to store their data outside of their country and the other reason is to reduce the latency.**
* **latency means that it takes time to take request from customer and give the server in response back.**
* **If the region is near the latency is very low**
* **Delay means high latency**
* **No delay means low latency🡪 prefer everyone**

**Avilability zones:**

**Ex: in india they have different zones like Mumbai(N)or (S),Kolkata,delhi**

**The need means whenever the one zone goes downtime the application goes down so in the region you have different zones it is easy to put application in two reason for safety purpose.**

**Virtual private Cloud: (VPC)**

**Understand Scenario:**

**There are three lazy fellows who suffered for buildibg their house at that time one wise came and think wisely said with them he owned that land he said I will build house for in return you can give money to me back.the person accepted this decision after they raised concern regarding to (privacy Breach) (if hacker comes their three member will suffer)at that time the wise person build one security wall for them for allowing authorized ones through gateways and some security groups.**

**The Real Scenario Regarding AWS:**

**The wise person is Aws and The lazy means persons who suffered with maintaining their own(data centers) physical severs so at that time AWS came into picture it said I built my own data centers I will provide Instances and everything in return you pay to me.In this case The security concept is VPC.(to host the applications of customers)**

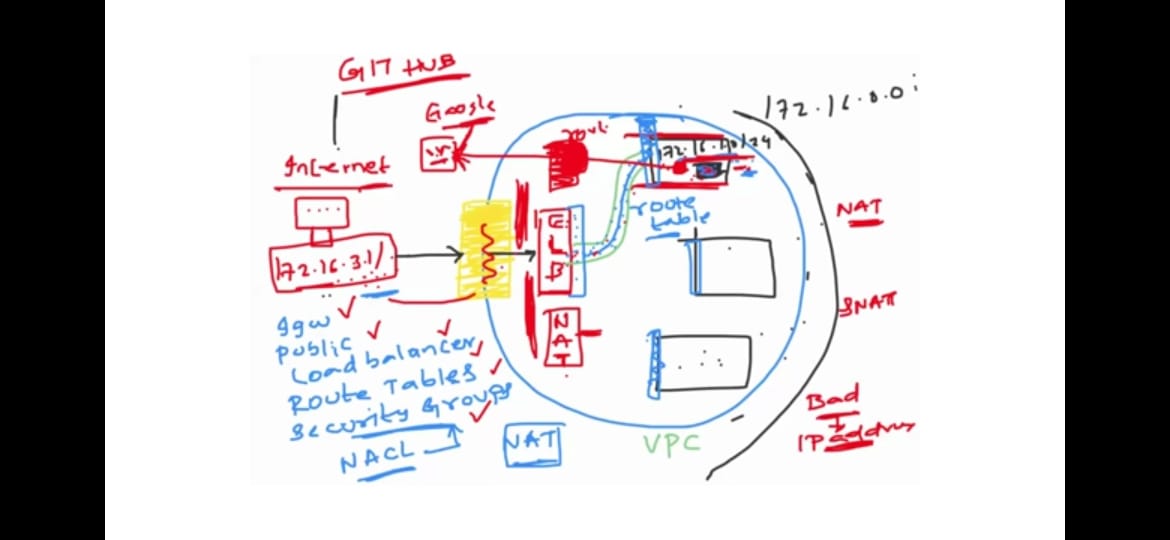
* **The DEvops Engineers Responsible to configure the VPC everything.**
* **In the VPC you defined space by using IP address Range**
* **For projects there will be a sub projects for that The Entire Range Space split into some spaces is called Subnet,deploying how many application subnet is their wish.**

**For Vpc There will be a Internet Gateway,inside Vpc there is Public Subnet and also Elastic Load balancer, to reach application request from load balancer to private subnet ther is need of router(Route table in AWS) through these we can reach subnet but there is security group in subnet after confirming credential it allows request.load balancer is attached to the Public subnet and it hastargeted group,route table defines the path once security group allows you the request reached to application>**

**The VPC components:Internet gate way, Public subnet,load balancer,Route tables,security groups.,NACL s🡪 it automate the defining of security group for each subnet instead defining same thing for eachone.**

**NAT gate way:**

**The private subnet want access something (info) from google or some other in internet but your Application Ip address should not show to someone it is not good practice for that purpose we use NAT gate way for asking IP address, if we apply masking in Load balancer is called SNAT and applied in router is called NAT gate way 🡪 these help to mask the IP address and accessing resources from internet or download require packages in internet.**





**AWS VPC [security group & NACL]**

* **We define security group at the time of instance create security group is used for only allow purpose but there is no concept of deny access.**
* **When we send request that can pass through the internet gate way and load balancer to subnet(application) ther if you want add additional security then the concept of NACL came into picture .**
* **So, we can say NACL is the additional security for security group.NACL has the power to allow and deny of any connection/access.It can be handled by devops Engineers/Aws admins etc..**
* **BY default AWS provides default vpc if you want you can customize.**
  + **Security group-**🡪 **instance level**
  + **NACL-**🡪**subnet level**
  + **NACL allows/Denies particular port.**

**Inbound Outbound**

**User--------🡪APP-----------🡪**

**The app is amazon.com,the user want access this using inbound traffic but the app amazon.com wants access with another application like amazon pay these called outbound rules.**

**In the default AWS outbound port 25 won’t allow🡪 because it related to Email spam activity it blocks that port.**

**NACL-🡪Network Access Control List**

**Let us understand scenario in the security group it allows all traffic but it could find someone make wrong it wants stop traffic for some ports then at that time at subnet level the devops engineers or Aws admins they block unwanted port/Traffics.By using NACL we can automately add security group for all subnet instances.**

**Practical:**

**1.**login into Aws

2.create vpc in the Vpc services(search for it,defaultly aws provide all things route tables loadbalancers everything)

3.create Ec2 instance inthat u have to edit Network settings choose your vpc instead default ,and then choose public tier one ,enable IP address launch instance)

4.connect to terminal install python3

5.launch simple python server with🡪**python3 -m http.server 8000**(ur wishing port number)

6. try to access it from the browser using Public IP address

7.here u need to enable port 8000 in security group and in you need to check in Vpc whether all traffic allow or not in NACL. (see it in VPC services Network ACL,NACL allows then only your request enter into subnet otherwise you can’t access your server).

**ROUTE53:**

* **Route53 provides DNS service.(Domain Name System)**
* **It will be easy to use domain names instead IP address,because remembering IP address is difficult practice.**
* **Whenever we create Load balancer or Applicationther wil be IP address allocated for that by AWS.but we never use IPaddress.**
* **DNA that converts domain names into IP address internally**
* **The main challenges using IP Address:**
  + **Remembering IP Adress is difficult for everything**
  + **Ip address change everytime so it can be static or it may be dynamic.**

**For geeting Domain Name:**

**First you should buy domain from Go daddy or some where**

**You also need hosting solution and need to maintain DNS records**

**For that AWS giving service called Route53**

**Amazon-🡪IGW🡪R53🡪LB🡪you can access subnets here(subnets)**

**Domain registration purchase 🡪you can host it in the inside aws or outside somewhere both the case you need maintain Hosted Zones🡪in these the record is ther which contain**

**Domain name-🡪IP address**

**Domain Name-🡪 Ip address…etc**

**AWS CLI🡪command line Interface**

**The main use of Aws CLI is for quick access**

**->used for manage &create Infrastructre**

**AWS API🡪Application Programming Interface**

**🡪reach programmatically**

**AWS API are:**

* **AWS CLI**
* **Terraform**
* **CloudFormation**
* **CDK**

**The first three are under Infrastucture as a code(IAC)**

**--. At the time of reaching application through program instead of UI you need to mention parameters like name,versioning etc..**

**The formats are JSON,YAML**

**The simple conversion of program done by aws cli like**

**./aws s3 parameters**

**User->cli->->API🡪user simply mention requirement api create resources ex:aws - -**

**Difference of all 4 types:**

**CLI🡪 quicker usecase like list the s3 buckets🡪aws s3 ls mention on cli**

**CFT/Terraform🡪 for creation of multiple combination resource creation**

**CFT🡪cloudformation Template**

* **It is implemented IaC but Cli cannot**
* **The principles of IaC are:** 
  + **Declarative🡪what u sees is what you have 🡪means by seeing the template u need to understand what are the resources are available to you**
  + **Versioned🡪 we can say GIT /s3 also versioned**

**The cloud provider understand language is API calls**

**CFT acts as middle man & also IaC🡪 it will ask user provide json/yaml file I will convert into API calls.**

**CFT:**

**Yaml vs json**

* **Yaml have advantage of commenting which is useful for understanding to others**
* **Readability is there and indendation is there for dividing block of code for flexible**
* **But in json it is complex🡪because it have brackets**

**🡪the one more advantage of Drift Detection🡪it means after creating s3 bucket using CFT with enabling version.**

**->but someone went s3 bucket and disabled the versioning then u can see/track where changes done how using drift detection it show everything**

**🡪the one more thingis creating Stack by creating stock u can import the yaml file from personel laptop also.**

**The mandatory things in yaml/json file:**

* **Version**
* **Description**
* **Metadata**
* **Parameters**
* **Rules**
* **Mapping**
* **Resources🡪is very very important even u don’t mention above things**

**[go through aws CFT documentation🡪user guide🡪working with templates🡪template formats/template anatomy**

**After 🡪go template reference🡪template property🡪choose EC2 or S3 whatever see syntax for learning]**

**AWS CI/CD:**

**In these four things:**

* **Aws code commit**
* **Aws code pipeline**
* **Aws code build**
* **Aws code deploy**

**Aws code commit Advantages:**

* + **Managed Git**🡪 **like github,gitlab:but it is private git**
  + **Scalability**
  + **Reliability**

**Aws Disadvantages:**

* **Features**🡪 **provide less features than git open source**
* **Aws Restricted**🡪**not open source because it is private git and also not compatible to other private gits like git lab,git hub enterprise so org don’t use more**
* **Less Integration with services outside AWS**
* **User interface is not that much flexible**

**And another one we can upload files by using AWS UI but it allows only one file at atime for upload etc..**

* **If u use terminal it is flexible to upload more files at a time**
* **Before codecommit service members istall git on their virtual machines &manage them by scaling but amazon make it is to provide these service with buy as much we want.**



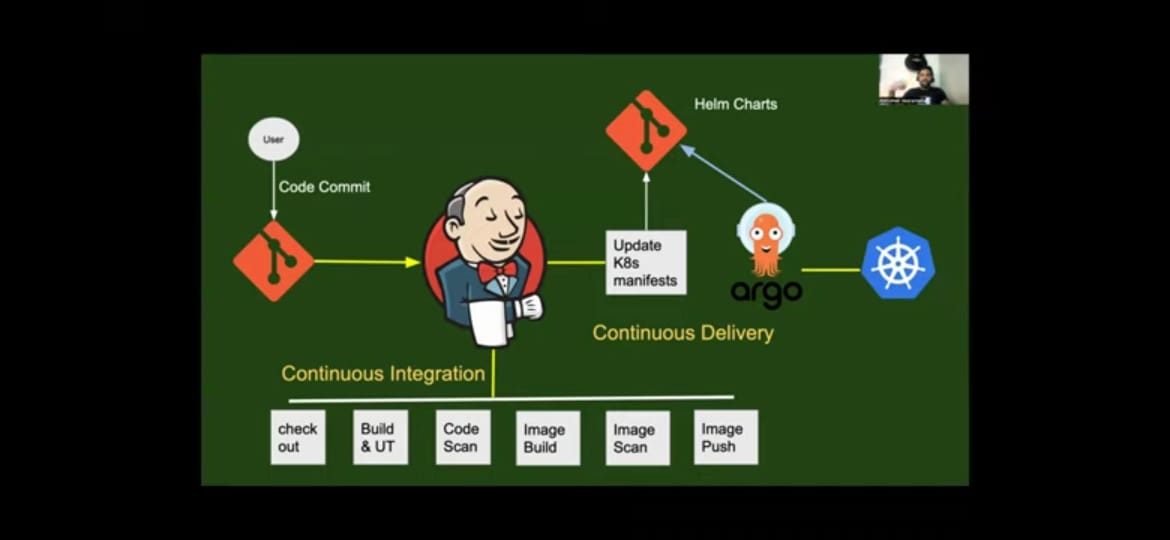
**We should connect to git by using IAM user only we should nt use root user.**

**Create IAM user by providing policy as codecommit powershell.login with these credentials.**

**Process:**

* **Go to codecommit service -> create repository->**
* **After login with IAM user**
* **Go to Iam services** 🡪 **goto user**🡪**there security credentials**🡪 **there u cansee http codecommit credential** 🡪 **generate them**
* **Goto git**
* **Git clone <url>**
* **U can get url in code commit repo** 🡪 **there u can see http copy and paste**
* **Then it will ask credential paste them previously get from iam user**
* **Then use**🡪 **cd <repo name>**
* **then do like normal git**🡪 **use file** 🡪 **git add file**🡪 **git commit -m”msg”**🡪 **use git push-**🡪 **see in repo u will pushed file there.**

**AWS code pipeline:**



The main difference b/w Jenkins and code pipeline is🡪 git used in Jenkins andb code commit is used in code pipeline.

But,Jenkins is very famous becz it is open source and it act as **implementation of continuous Integration** **and Invoking of continuous delivery** but not responsible for full continuous delivery

It have some process we can see above like check out

Jenkins is very famous for cI/CD-🡪 it can create webhooks to trigger code from github in the groove script.

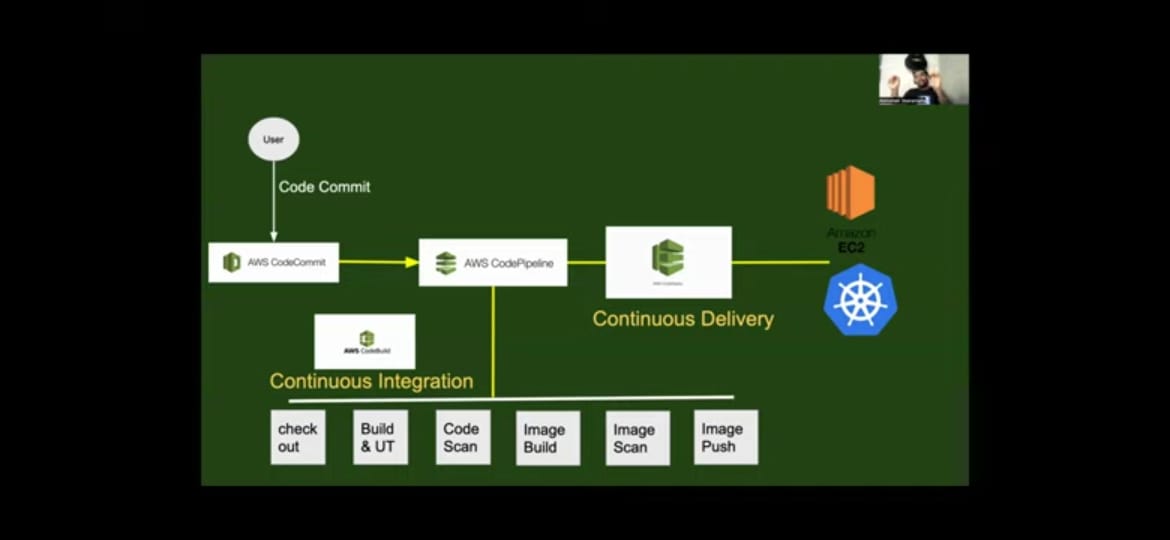
For continuous delivery we can use ansible or shell scripts but they are out dated

Right now famous is Argo CD which is git ops tool

By using helm charts you can push docker images every thing

Here, Jenkins works as **orchestrator.**

**Aws code pipeline:**



Code commit don’t have more features

Here, code pipeline **is invoking continuous integration(code build) and invoking continuous delivery**

**Some org prefer code pipeline because Jenkins install on vm 🡪 it is master- slave architecture**

**So if u want run 2 slaves you can but your org wants 20-30 then it is difficult to u to maintain all virual machines**

* **s**o in these case aws saw opportunity came with code pipeline & it said you can use as many workers I will responsible for manage just u need to pay as u go.
* **But why Jenkins famous 🡪 due to it can install on any where like ec2 instance or any where it will work 🡪 if u want scalability u can in Jenkins**
* **Aws code pipeline is unable to run outside aws**

**For launching your application most of them used the Kubernetes or ec2 instance.**