=> VPC ✓

-> EFS —
Cloudformation —
{ Terraform }
- Ansible }

{ Reread
Maven
Git }

- Tomcat
- Docker
- K8s
- Jenkins   projects
Sonarqube
nexus

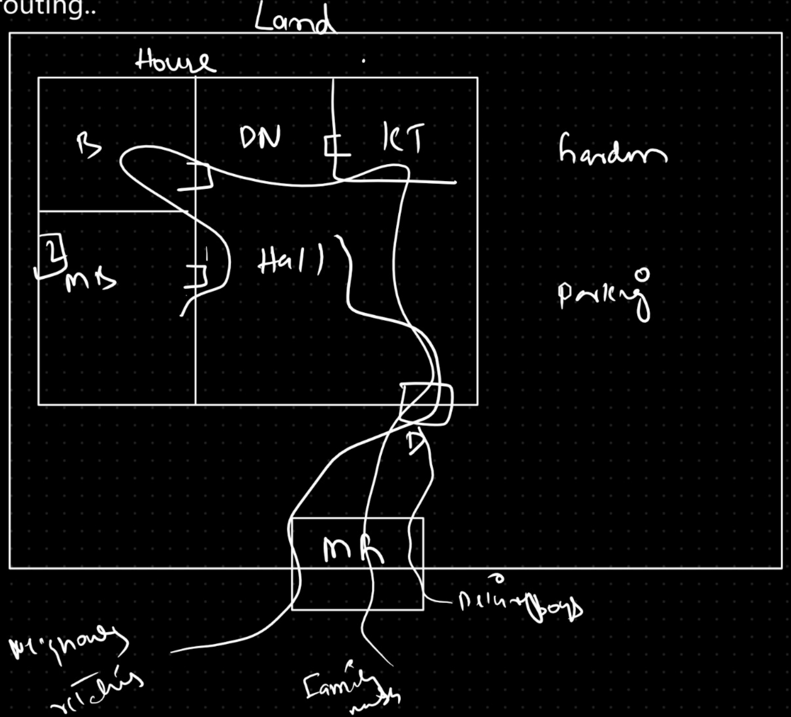
=> VPC (Virtual Private Cloud)

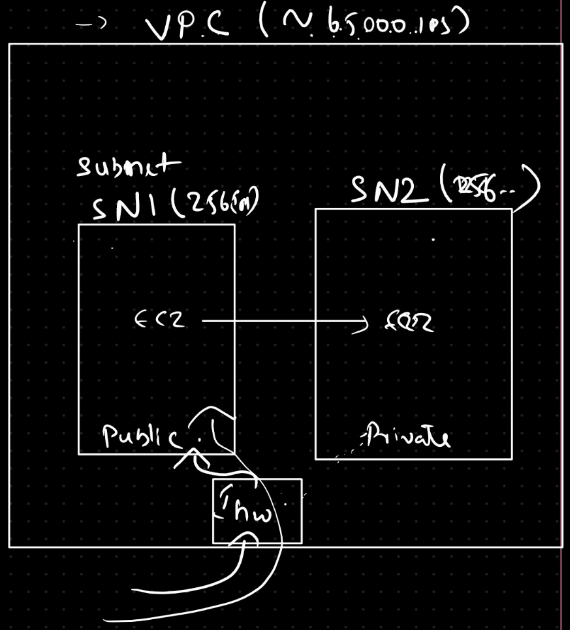
=> (Network & security)

--> VPC provides isolated network for resources in AWS Cloud

--> With the help of VPC we can protect our resources in AWS Cloud

-->VPC Provides flexible and secured network to maintain and manage our resources in AWS Cloud
--> In VPC , users can define their own IP addresses range, subnets, route table and network gateways. It provides control over network configuration such as setting up access control policies , firewall rules, and network traffic routing..





→ VPC Terminology {

① VPC.

② Subnet (Public & private)

③ CIDR Block (IP ranges)

④ Route Tables

⑧ Internet Gateway (IGW)

⑥ NAT Gateway

⑦ VPC Peering

⑧ Security Groups → Resource level → Rules (I/O) to allow

⑨ NACL → subnet level → Rules Allow & Deny }

IP's

There are several types of IPs (Internet Protocol) address used in a computer network
IPV4 --> 32 bits --> 4 sets of numbers
IPV6 --> 128 bits --> 8 sets of four hexadecimal digits
IPv4: Uses dotted decimal notation (e.g., 192.168.1.1).
IPv6: Uses hexadecimal notation with colons (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334)

Public IP
Private IP
Static IP address
Dynamic IP

VPC Sizing--> Process of allocating IPs

IP Ranges we will do with CIDR --> Class less inter domain range
VPC are regional Specific

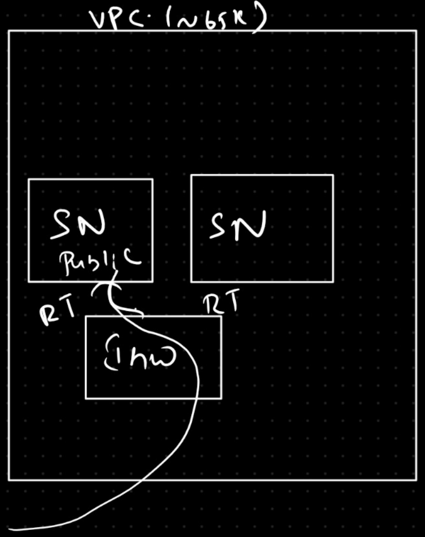
On regular basis number of new devices are rapidly increasing and are using internet and any device if it has to use internet then IP is mandatory and must and there might be possibility of running out of ips which has to be unique to overcome IPV6 came into picture

IPV4 --> 32 bits numeric addresses written in four sets of numbers separated by periods eg : 192.168.1.1

Its most widely used IP Version and supports approximately 4.3 billion unique addresses

IPV6 --> 128 bits alphanumeric addresses written in 8 sets of four hexadecimal digits separated by colons
e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334

Supports 340 undecillion unique addresses



VPC sizing

=> sizing will be calculated in 2 power ~~of~~ IPv4

10.0.0.0/16 => 2 pow (32-16) => 2 Pow 16

10.0.0.1/32 =>

2x2x2x2---16

=> $\boxed{65,536}$

2 power $^{(32-32)}$ => 2 Pow^{0} => 1

10.0.0.1/31 2^{1} -> 2

/30 2^{2} -> 4

/29 2^{3} -> 8

} AWS won't support /29 to /32

10.0.0.1 /28 => 2^{32-28} => 2^{4} => 16 ->

AWS supports from /28

/29 to /32 -> 0 1 2 4 8

16 to 65536

/16 to /28

~~10.0.0.1/15~~ X

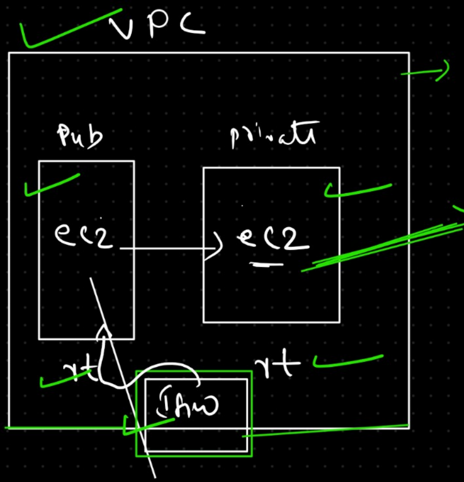
subnet ranges can be given

65536 IPs
(/16 to /28
↳ 16 IPs

CIDR block size must be between /16 and /28.

-> Recommended to use /24

10.0.0.1/24 => 2^{32-24} => 2^{8} => $\boxed{256}$



-> VPC

Natgateway is must

Practical task:

--> Create VPC

--> CIDR Block : 10.0.0.0/16
--> Select No IPV6 CIDR Block
--> Rest are default
--> Create VPC
Note : One Route Table will be created for VPC by default if needed rename it .

--> Create 2 Subnets

--->a) create subnet -1
----> name : telusko-public-sn
--> CIDR Block : 10.0.0.0/24 (It will take 256 ips)

---> b) Create subnet-2
---> name : telusko-private-sn
---> CIDR Block : 10.0.1.0/24 (It will take 256 ips)

Route Tables
The moment we create VPC default Route table will be created go to rout table section and rename it to private-telusko- rt
Create a new Route Table and name it to public- telusko-rt

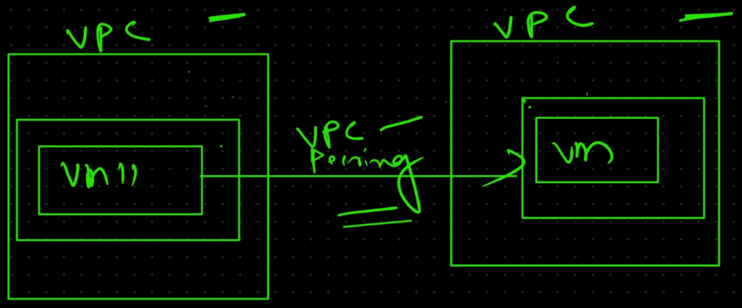
After Route table are created --> Edit and attach subnet association
public-telusko-rt --> public-telusko-sn
private-telusko- rt --> private-telusko- sn

Created new Internet gateway --> telusko-igw

After that click on public-telusko-rt --> routes and add new routre --> internet gateway and select the telusko-igw 0.0.0.0/0

--> Created VPC
--> Created 2 Subnets
-->Created IGW
-->Created Route Table
--> Associated RT to subnets
-->Attached IGW to RT
--> Create 2 EC2 VM our own VPC within 1 each public subnet and 1 in private subnet
--> Tested by connect to ec2vm of public subnet it worked and we couldn't able to connect to ec2vm of private subnet bcz no igw is attached

Connect with private ec2 of private subnet from public ec2 of public subnet using ssh connection


VPC
VPC
vm11
VPC Peering
vm