**AWS**

**1.VPC**

control how your network and the EC2 resources inside your network are exposed to the Internet.

By Default one VPC will be existed

First, we need to create a VPC by giving a subnet range eg:10.1.0.0/16 (by default per account we can create only 5 VPC, we need to request for more)

After creating VPC right click on it and enable DNS host name on it

* **SUBNET:**

here we need to create subnet by giving above vpc and provide availability zone 1a and provide the subnets between the vpc subnets e.g.: 10.1.1.0/24

create one more subnet in 1b zone with same VPC (1b -10.1.2.0/24)

and one more subnet in 1c zone with same VPC (1c -10.1.3.0/24)

after creation of all subnet zone we should right click on each subnet and enable auto assign ip for each subnet

* **Internet Gateway:**

Create an internet gateway and after creation we need to attach the vpc for that gateway, for 1 vpc we can attach 1 IGW

* **Route Table**

here it will enable the connection between two subnets and we can enable the routes what to connect

create the route table by provide the VPC and here

1. Here below we have Subnet Association in route table here we need to assign the 1a,1b,1c availability zones of this same VPC to the route table

2. here below we have routes here we can assign routes to internet gateway, NATgatway, vpc peering…etc here if we need internet connection, we can give destination as 0.0.0.0/0 and target as internet gateway

(By default we route table will be created for a vpc but for best practice we should create our own route table for variating public and private subnets in the vpc)

* **NAT Gateway**: (Network Address Translation)

NAT gateway is used for security for private subnets, which it will not allow public to access private subnets but through NAT gateway servers can access the internet

here we can create two subnets named public and private,

and create two Routing table and name it as public RT and private RT

here public subnet should be linked to public RT and allow all traffic and in Target select internet gateway

But in private subnet we need to link to the NAT gateway and NAT should be linked to the public subnet

First go to NAT gateway and create the NAT gateway, here subnet – public subnet and allocate the elastic ip give the name-NATGW1(here we will give input of Nat gateway with public subnet)

After creation NAT gateway we need to come to route table here private subnet should be added to private route table(which we named as private RT)and allow all traffic and in Target we need to select NAT gateway

If we need to connect to any private server, we need to go to public server and in that we need to login to private server through ssh connection through private key

* **VPC Peering**

It is a component it will make a connection between two vpc

Ip Address Ranges must not overlap in any vpc

here peering will be done one vpc to another vpc but series connection will not be done

e.g.: let us take three VPC over here

1. green VPC in 1a -10.1.0.0/16
2. red vpc in 1b -192.168.0.0/16
3. Blue VPC in some other region -172.16.0.0/16

1.Green to red peering

here we are going to connect two vpc within the region and different availability

create a peering connection and give requested vpc as green and select my account and this region, Vpc accepter as red vpc

after creating in above we need to accept peering request

Now let’s go to route table select green vpc and go to below routs and give the red vpc cidr of 192.168.0.0/16 and in target select peering- in that select green-red peering

now again go to route table and select red vpc and select cidr of green vpc 10.1.0.0/16 and in target select same peering

Now connecting of two vpc has done

2.Green to Blue peering

here we are going to connect two vpc with other regions

create a peering connection and give requested vpc as green and select my account and another region, vpc accepter as blue vpc id( blue VPC id should be copied)

after creating let’s go blue vpc and accept the request of vpc peering

now in blue vpc region we need to go for route table in routs add the cidr of green 10.1.0.0/16 and in targets select peering – in that select green blue peering

now again come back to green vpc region go to route table select green vpc and below routes add the cidr of blue vpc 172.16.0.0/16 and in target select peering

Now connecting of two vpc has done

3.AWS one account to another account :

let’s take A and B

here every thing is same but while creating a peering select other account, it will ask the id of another account (here we need to go to B account in the route table select specific vpc in below we have summary here we will get the id… of the account B) and everything is same

if we need to connect specific subnet, we need to use subnets cidr in routes

if we need to connect specific server or instance, we need to provide private ip in routes

* **Endpoints**

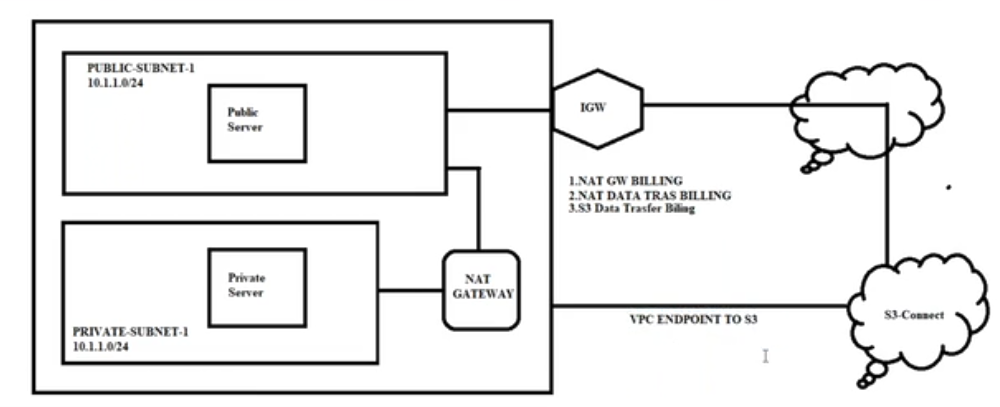
It is a service which it is used to connect the some aws services directly to the vpc

here for e.g. if private server want to connect to s3… here flow will be like this

Before end point connection (private subnet -NAT -Public subnet -internet -s3)

here endpoint will connect the s3 to the specific vpc… so private subnet can connect directly to s3

After end point connection(private subnet – s3)



first need to create endpoint by selecting the service what to access (e.g. s3) and select the specific vpc for it and select specific private-RT. so it will create endpoint

after creating endpoint in route table in routes it will create one route for default

* **Endpoints**
  1. **Security Groups**

Create a Security group by giving specified VPC, here we have to control the traffics like inbound and outbound

security groups can be given to servers(ec2) not for subnets

here this is stateful connection.

for e.g.: if we give inbound connection, it automatically allow outbound for the same connection

* 1. **Network ACL’s**

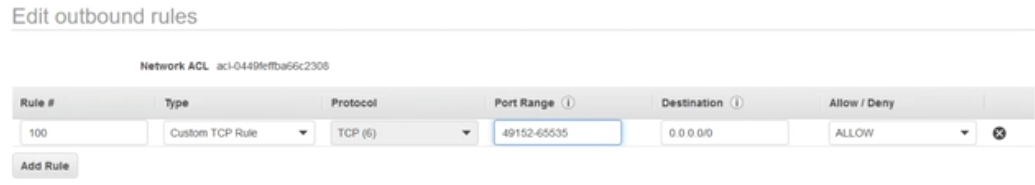
here default one network ACL was created, when vpc was created

For best practice let’s create our own NACL by giving VPC and attach subnets for a NACL

Network ACL’s can be given to subnet levels

basically it will deny all the inbound or outbound rules, we can provide the traffic what we need allow traffic in inbound and outbound both way i.e. it is stateless

if both the traffic is not allowed, it will not allow traffic



here in outbound we should allow Rule-100; type-custom TCP rule; Port range- 32768–60999(Ephemeral ports)

here for example if u given same traffic two times, if you allow in the first column and deny in second column it will take the first column as main priority

better avoid changes in Network ACL, it may lead to many risks

**2.CLOUD FORMATION:**

AWS CloudFormation provides users with a simple way to create and manage a collection of Amazon Web Services (AWS) resources by provisioning and updating them in a predictable way. AWS CloudFormation enables you to manage your complete infrastructure or AWS resources in a text file.

* Create a new template or use an existing CloudFormation template using the JSON or YAML format.
* Save your code template locally or in an S3 bucket.
* Use AWS CloudFormation to build a stack on your template.
* AWS CloudFormation constructs and configures the stack resources that you have specified in your template.

AWS CloudFormation Concepts

* Format version
* Description
* Metadata
* Parameters
* Mappings
* Conditions
* Transform
* Resources
* Output