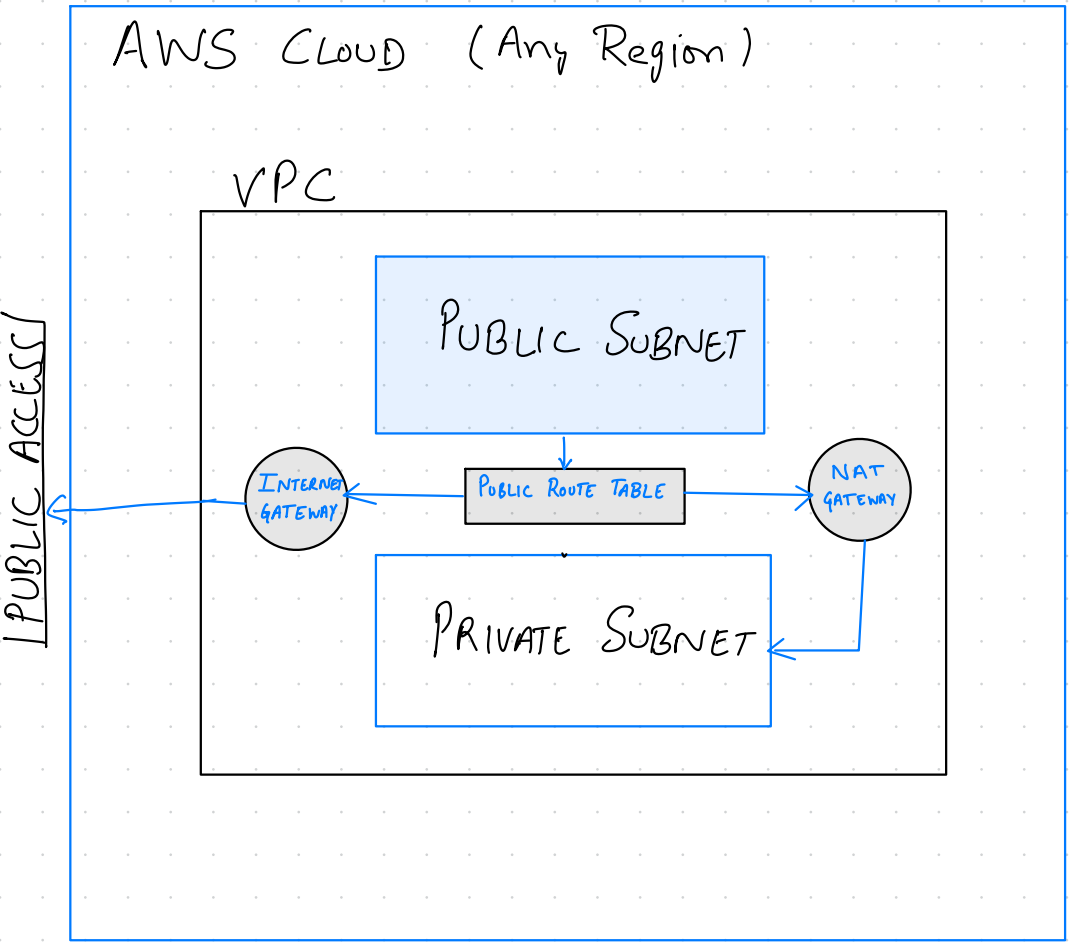
All about AWS VPC Service

By: Cheton Harmon

AWS VPC & VPC Peering



VPC → Virtual Private cloud

- Isolated environment to setup your application

SUBNET → A range of IP addresses in your VPC.

Route table → A set of rules called routes, that are used to determine where network traffic is directed

Internet Gateway → A gateway that you attach to your VPC to enable communicatio

between resources in your VPC and the internet.

CIDR block → classes inter domain routing.

An internet protocol address allocation and route aggregation methodology

Default VPC Components :-

- Create a VPC with a size /16 IPv4 CIDR block 172.0.0.1/16. This provides upto 65,536 private IPv4 addresses.
- Create a size of /20 default subnet in each availability zone. This provide upto 4,096 address per subnet. A few of

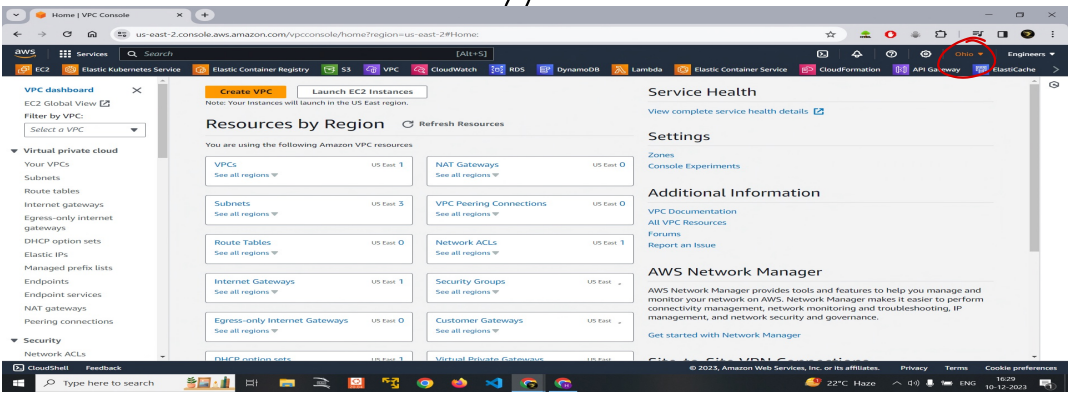
which are reserved for our use.

- Create an Internet gateway and connect it to your VPC.
- add a route to main route table that points all traffic (0.0.0.0/0) to the internet gateway

- Create a security group and associate it with your VPC.

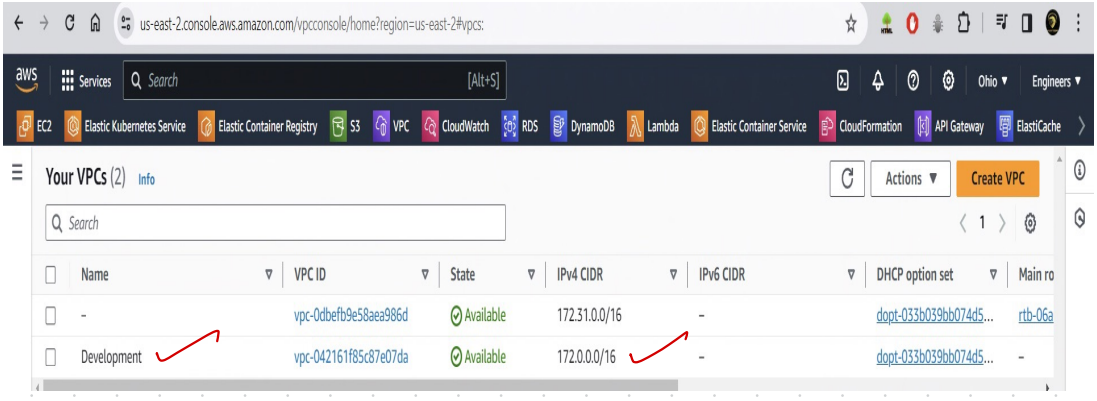
Steps to create VPC :-

1. Select any particular region. In this case we will go with ohio region.



2. Select your VPCs. There we can see the default VPC. Now hit on create VPC.

- give name to your VPC
- Select IPv4 CIDR block
- Set the IPv4 CIDR range as 172.0.0.0/16
- and then hit create VPC



3. Next we will create Subnet. In Subnet create window.

- Select your VPC.
- Set Subnet name (Public Subnet)
- Choose any availability Zone, out of three.
- to set IPv4 CIDR block, go to

https://www.davidc.net/sites/default/subnets/subnets.html

on this site set

the VPC CIDR block range and set block as /16 & hit update Now divide the CIDR range in such that public subnet attain 256 IP's.

Visual Subnet Calculator

Enter the network you wish to subnet:

Network Address: 172.0.0.0 / Mask bits: 16 [Update] [Reset]

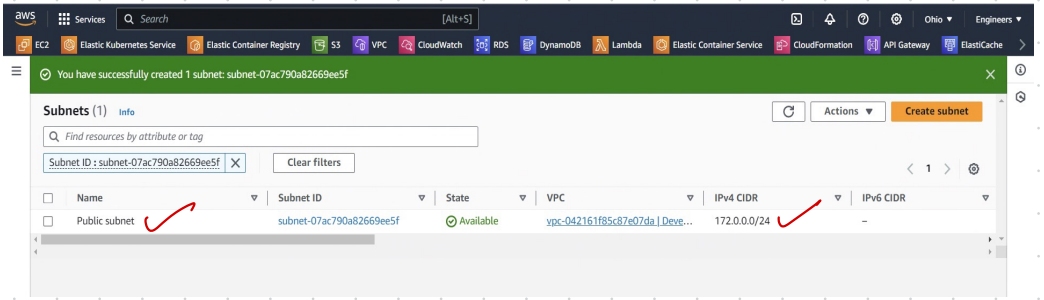
Show columns: ☑ Subnet address ☐ Netmask ☑ Range of addresses ☑ Useable IPs ☑ Hosts ☑ Divide ☑ Join

Click below to split and join subnets.
If you wish to save this subnetting for later, bookmark this hyperlink.

Subnet address	Range of addresses	Useable IPs	Hosts	Divide	Join
172.0.0.0/24	172.0.0.0 - 172.0.0.255	172.0.0.1 - 172.0.0.254	254	Divide	/24 /24
172.0.1.0/24	172.0.1.0 - 172.0.1.255	172.0.1.1 - 172.0.1.254	254	Divide	/23
172.0.2.0/23	172.0.2.0 - 172.0.3.255	172.0.2.1 - 172.0.3.254	510	Divide	/22 /23
172.0.4.0/22	172.0.4.0 - 172.0.7.255	172.0.4.1 - 172.0.7.254	1022	Divide	/21 /22
172.0.8.0/21	172.0.8.0 - 172.0.15.255	172.0.8.1 - 172.0.15.254	2046	Divide	/20 /21
172.0.16.0/20	172.0.16.0 - 172.0.31.255	172.0.16.1 - 172.0.31.254	4094	Divide	/19 /20
172.0.32.0/19	172.0.32.0 - 172.0.63.255	172.0.32.1 - 172.0.63.254	8190	Divide	/18 /19
172.0.64.0/18	172.0.64.0 - 172.0.127.255	172.0.64.1 - 172.0.127.254	16382	Divide	/17 /18
172.0.128.0/17	172.0.128.0 - 172.0.255.255	172.0.128.1 - 172.0.255.254	32766	Divide	/16 /17

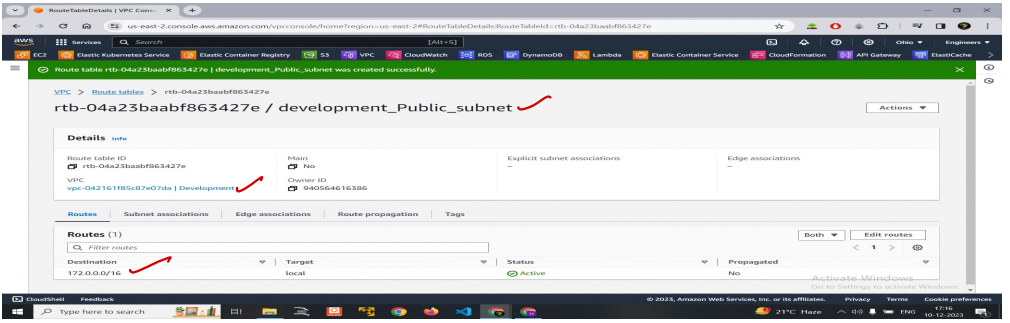
hit divide to train IP range.

— Now create the Subnet.



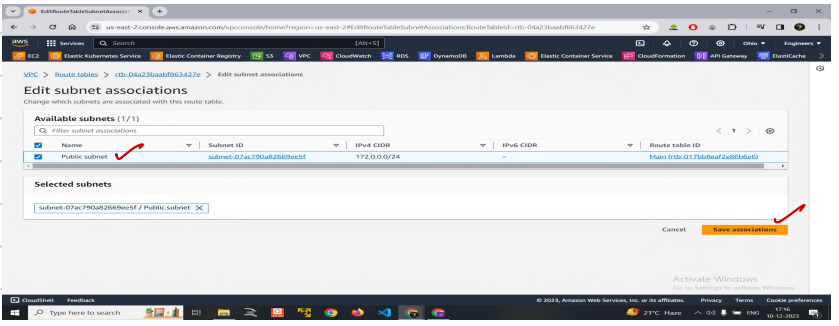
4. Now create Route table that will associate with this subnet. In route table window select :-

- Give name to route table
- Select your VPC (development)
- Create route table.



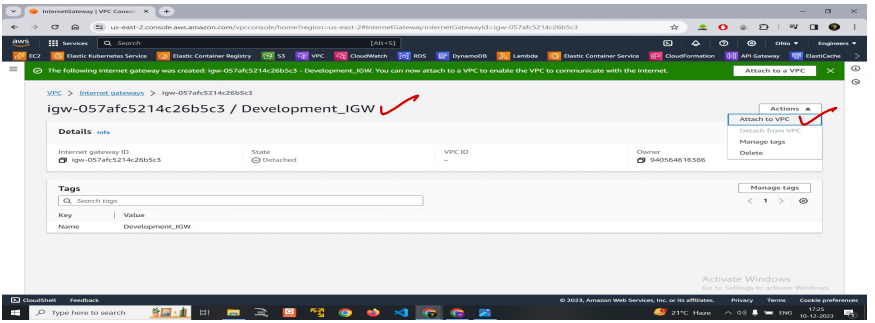
5. Now Associate this to public Subnet.

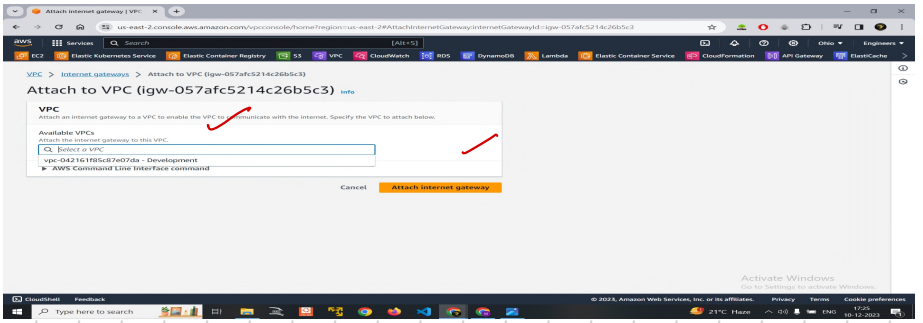
- Select on Subnet Association & hit edit subnet association. Over there select the subnet which we created.



6. Now our Subnet has Route table. But it isn't public subnet yet. for that we need to attach

internet gateway to this.
So, now create Internet gateway and attach it to VPC.

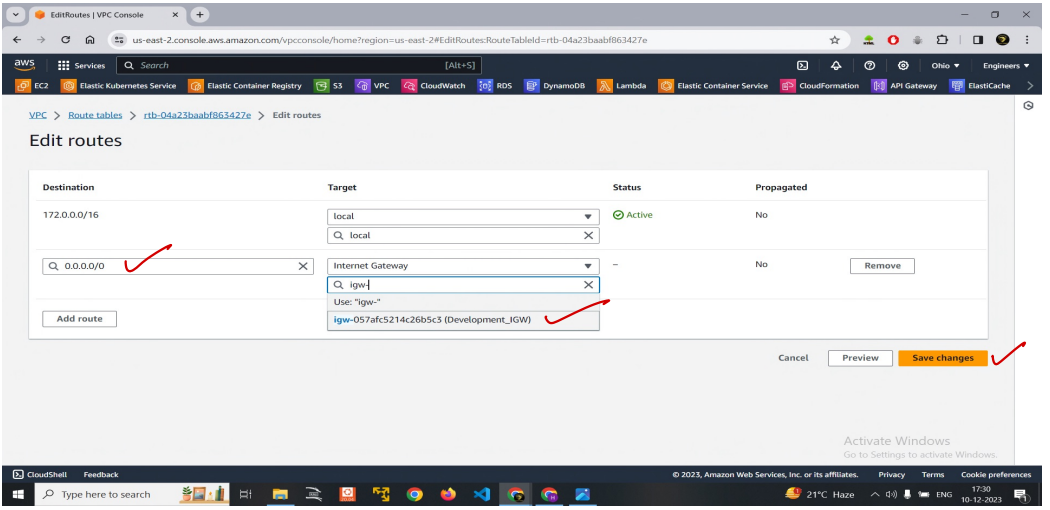




7. Now we will set route of all traffic (0.0.0.0/0) to this internet gateway that will make the subnet

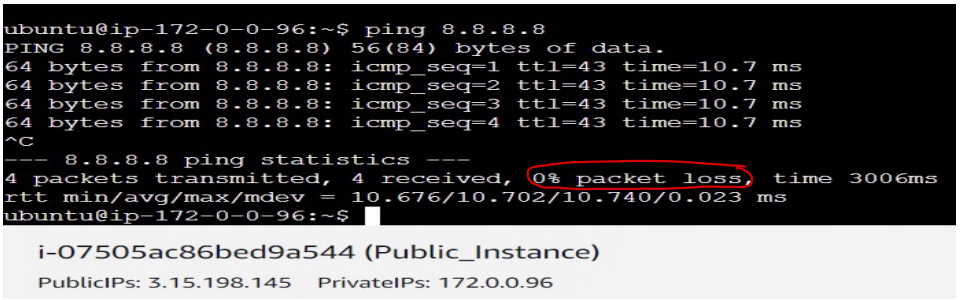
public

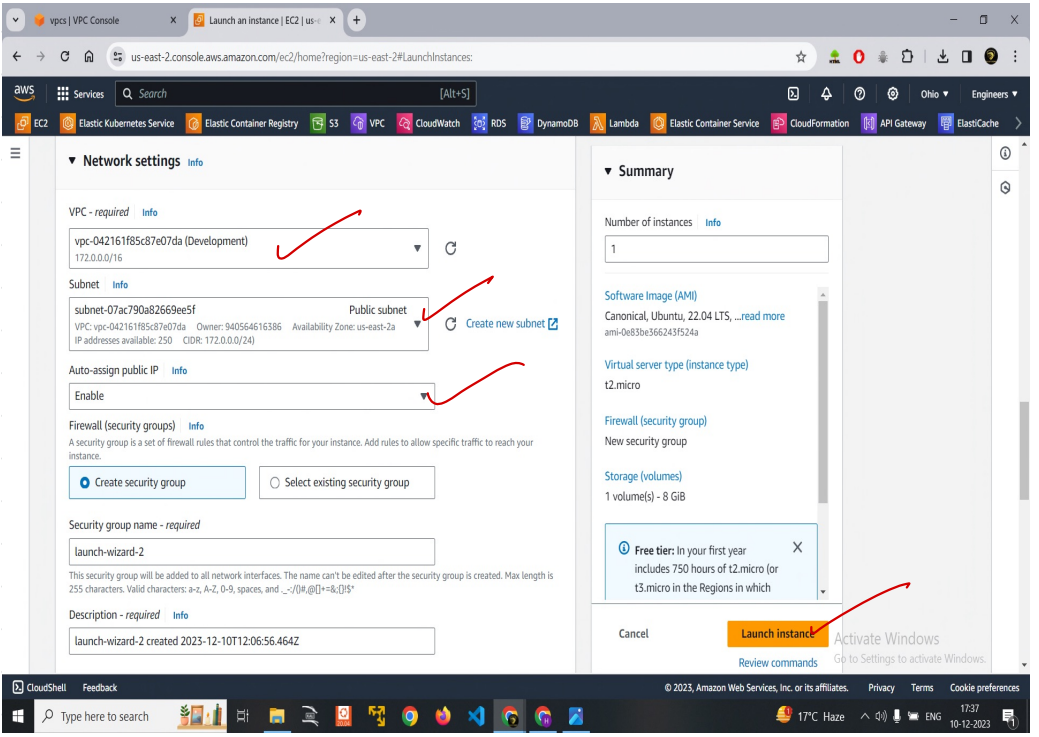
- go to route table i.e. attached to subnet in routes choose edit routes. over there add route.



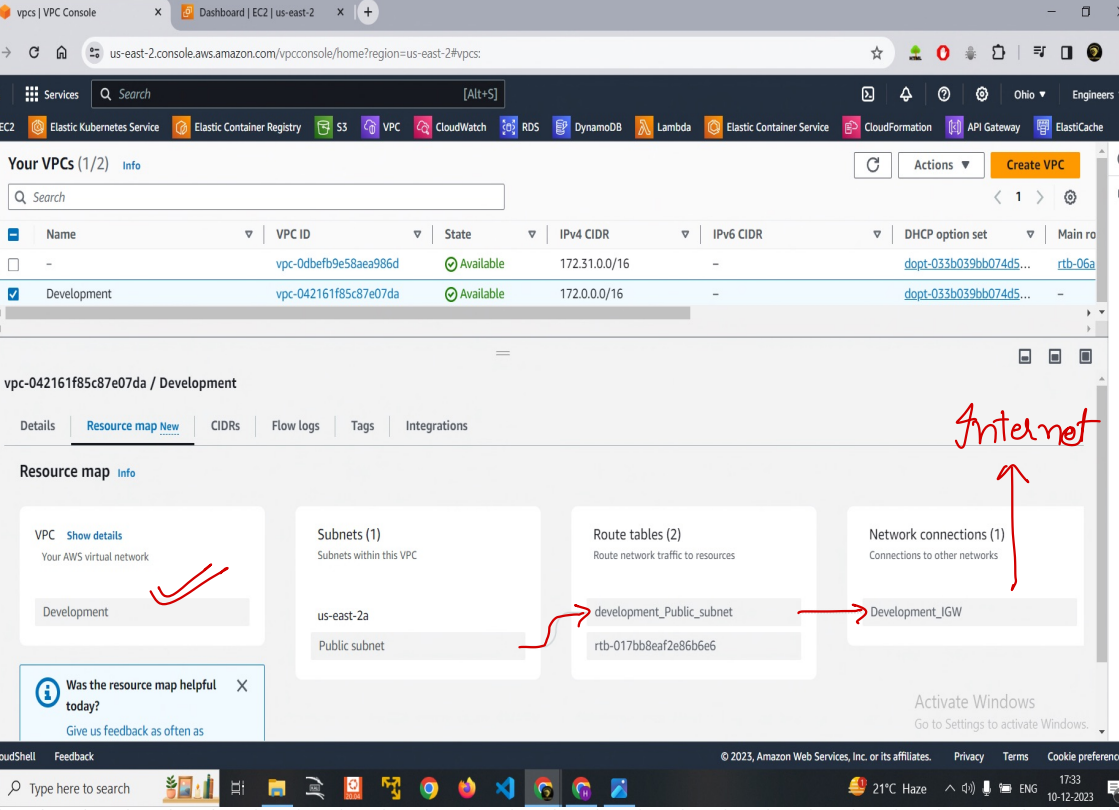
8. Now our subnet is public subnet.

9. To test go to EC2 and over there create instance inside this VPC in this public subnet. Connect to instance and ping to 8.8.8.8. If the package recieves data means instance has internet access.





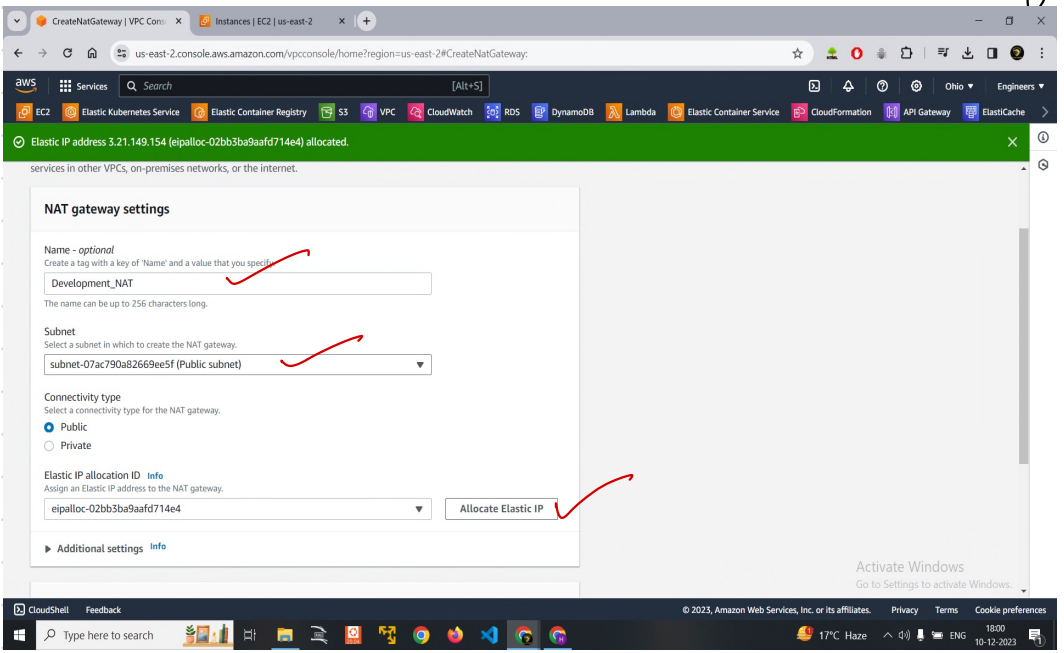
10. Till now we have achieved one public Subnet with internet access.



11. Now we will crete one private Subnet jimila to public subnet, but it will not be attached to Internet gateway.

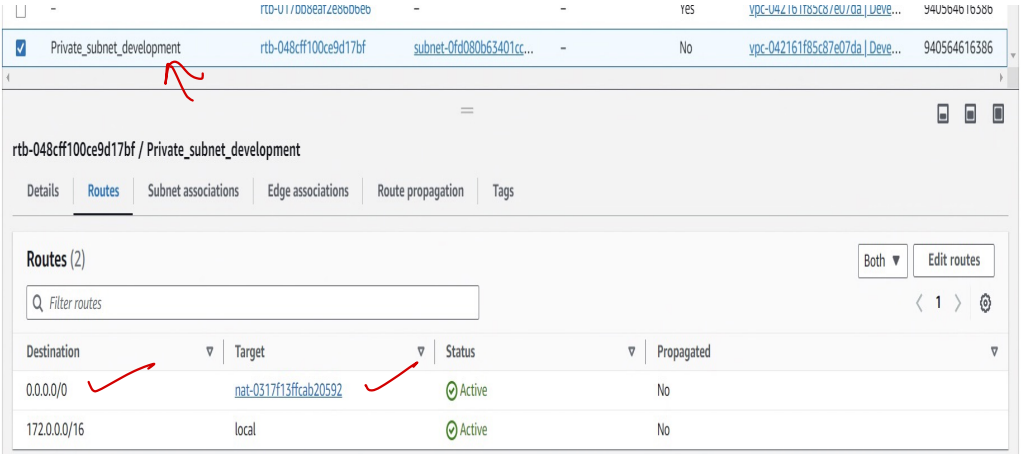
12. So, we will crete

a NAT Gateway in public subnet also allocate elastic IP to this NAT Gateway



13. Now we will give IPv4 CIDR range of

all traffic(0.0.0.0/0) to route table of private subnet & route that to NAT gateway.



14. Now create instance in the private subnet and don't give

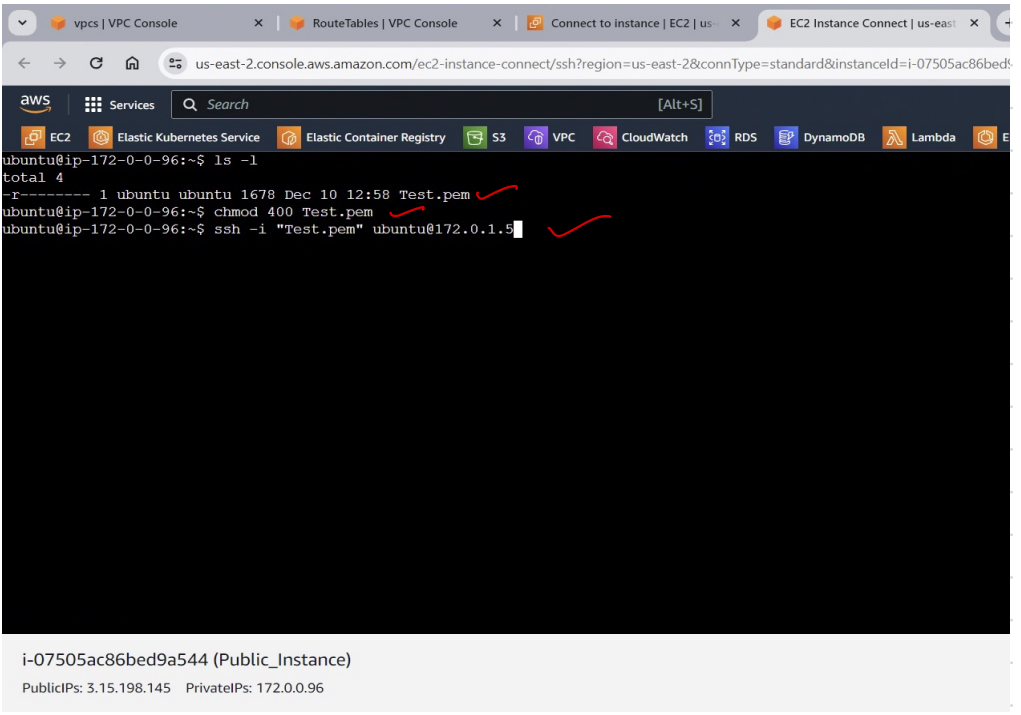
public Ip to it. Try to ping 8.8.8.8 from there.

15. After creating instance copy the .pem file to connect from public instance for that from local shell run command

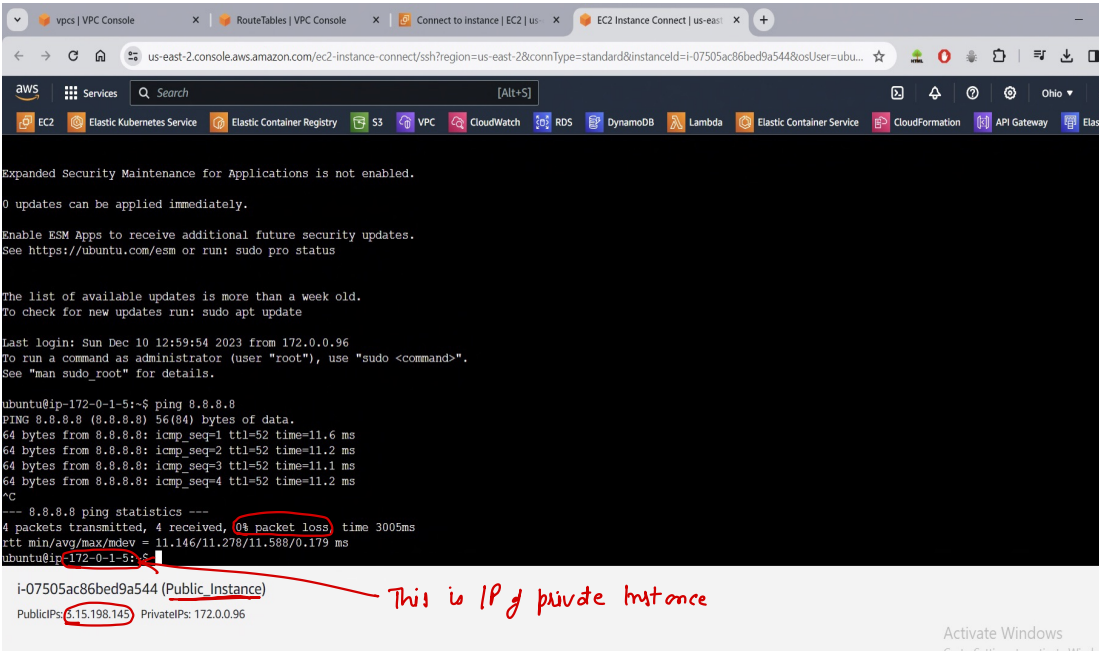
scp -i <filenametocopy> username@ipadd:<pathnamewheretocopy>

```
hardy@DESKTOP-R359UB4 MINGW64 ~/Downloads
$ scp -i Test.pem "Test.pem" ubuntu@3.15.198.145:/home/ubuntu
The authenticity of host '3.15.198.145 (3.15.198.145)' can't be established.
ED25519 key fingerprint is SHA256:Ex5kLpqC42ruzgUuBlbGMKPRkFSvvGwcystl2u10MIk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.15.198.145' (ED25519) to the list of known hosts.
Test.pem
```

16. After this Connect Public instance and on that look for the .pem file & change permissions



17. And then shell login to the private subnet. And ping 8.8.8.8 from there



Reach me on:

1. Github-> github.com/Hrmn97
2. Twitter -> twitter.com/Harman9765
3. LinkedIn -> linkedin.com/in/chetan-harman-56310424a
4. Website -> devhrmn.netlify.app