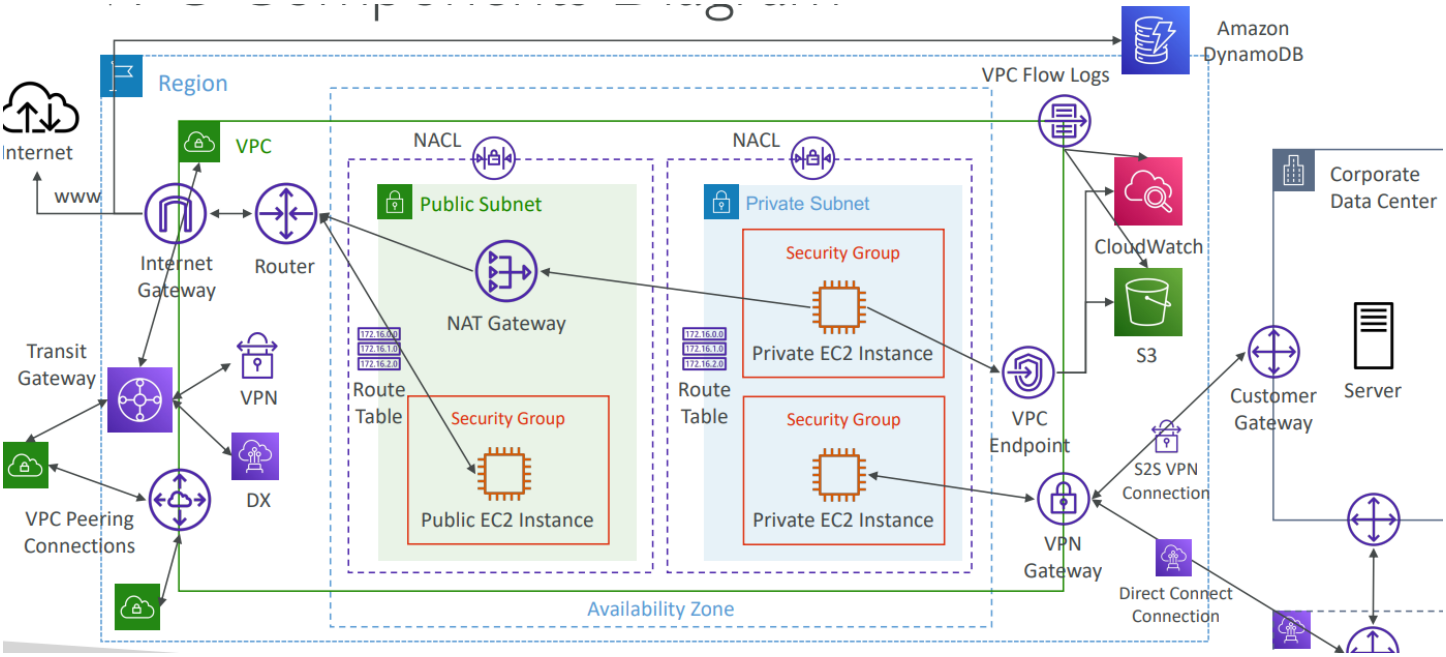
Creating an AWS VPC from Scratch
(EnterPrise Grade)

~Ritvik Kant



This is the final VPC which I will construct step by step

VPC

Stands for Virtual Private Cloud
In this guide, I'll create a custom VPC in AWS from scratch, including subnets, route tables, Internet connectivity, and essential components. This provides better control and security than using the default VPC.
This VPC WILL BE an Enterprise grade VPC with options and features which ensure

1.High Availability and scalibility

2. Resilience

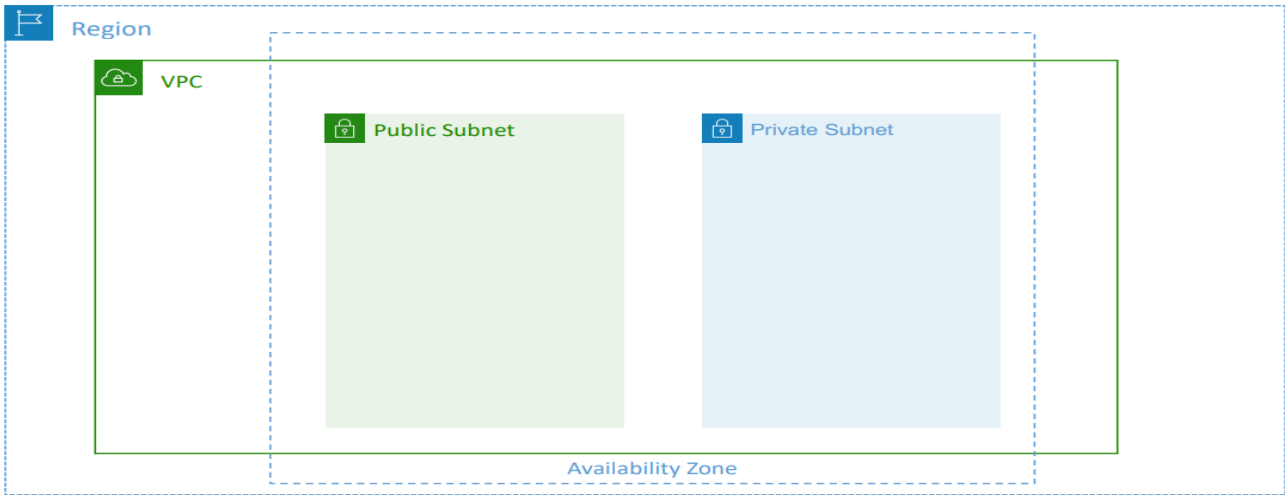
3. Security

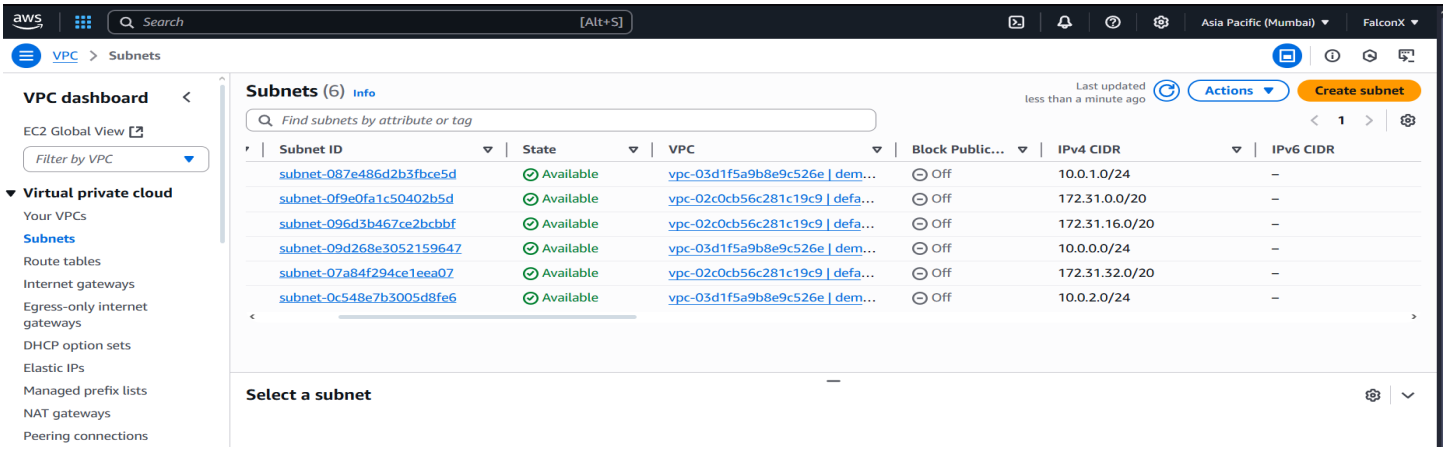
4. High performance and traffic spiking workloads

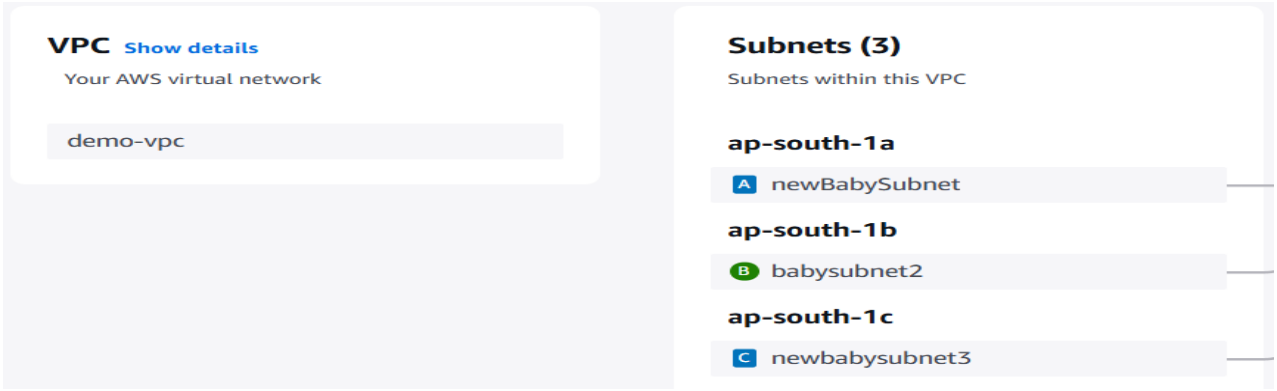
This is Ritvik and here is my take of a fault proof and an all rounder cost optimized and ENTERPRISE GRADE VPC
Let's make it from scratch

Step1. Creating a VPC and establishing subnets in a **region**

VPC as stated is a virtual private cloud in short like an independant society with residants not dependant on the outside for any resources
Imagine Subnets like the houses in the Neighbourhood Some our public like movie halls and parks
Rest are private like residential Homes







Step-2

Internet GateWay

Now that the neighbourhodis created ,we require a connection for import exports or in cloud terms Internet Access to the instacnes containing the app

I created an internet Gateway now the diagram is like this

Note:

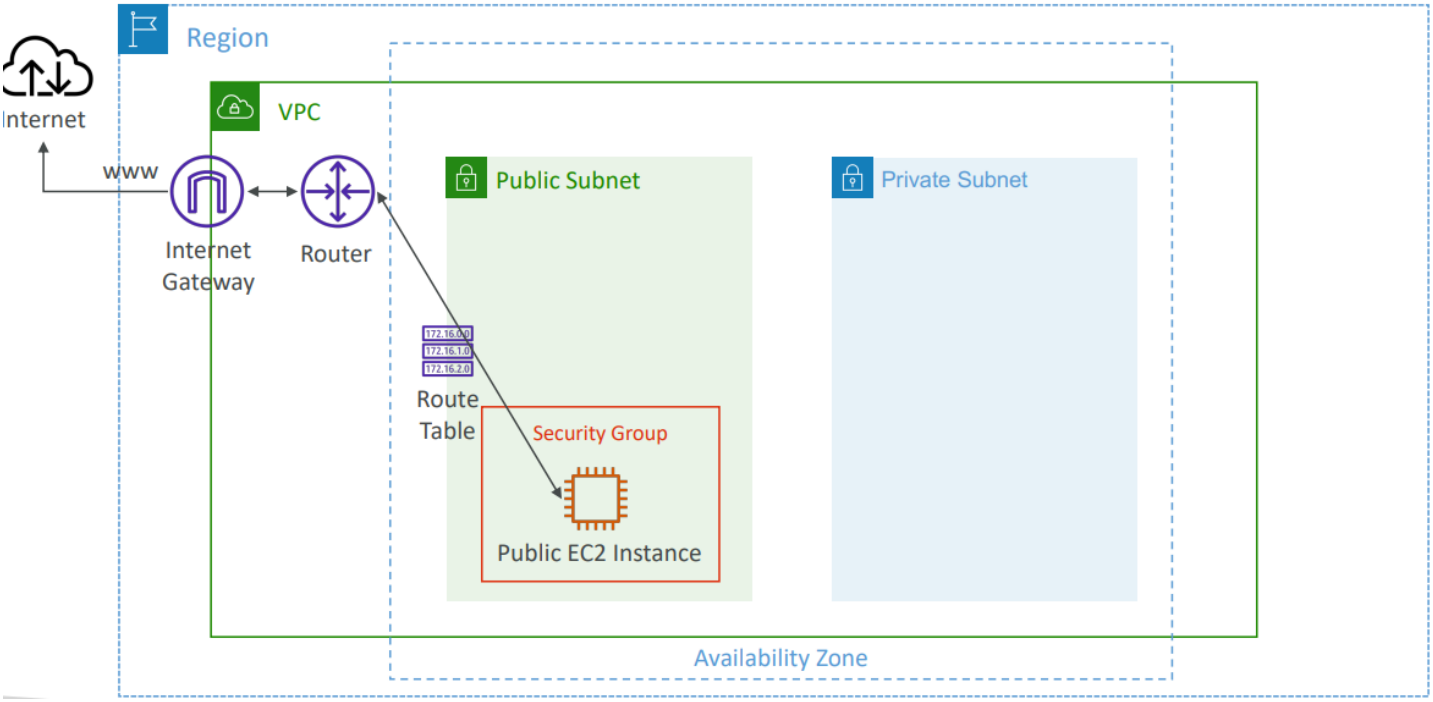
• It scales horizontally and is highly available and redundant

• Must be created separately from a VPC

• One VPC can only be attached to one IGW and vice versa

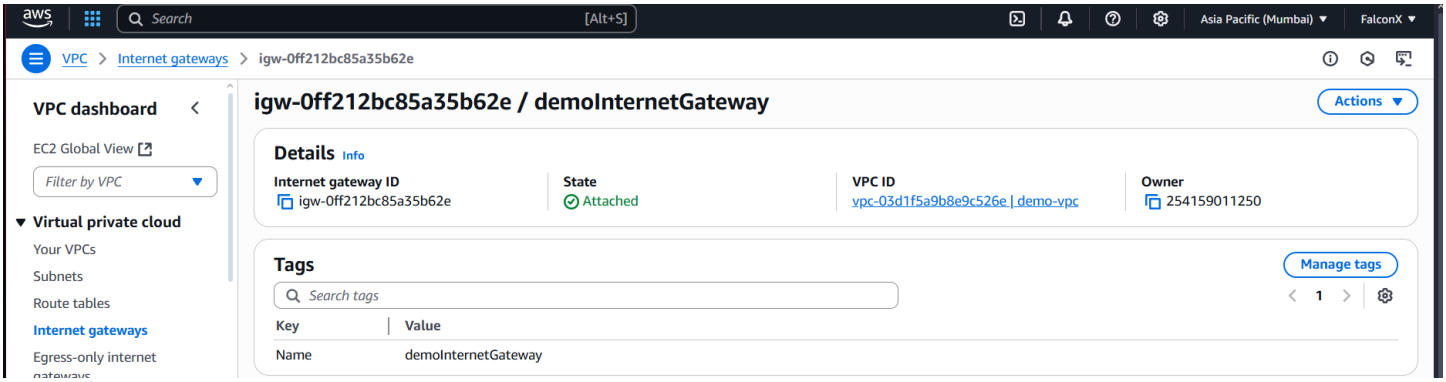
• Internet Gateways on their own do not allow Internet access…

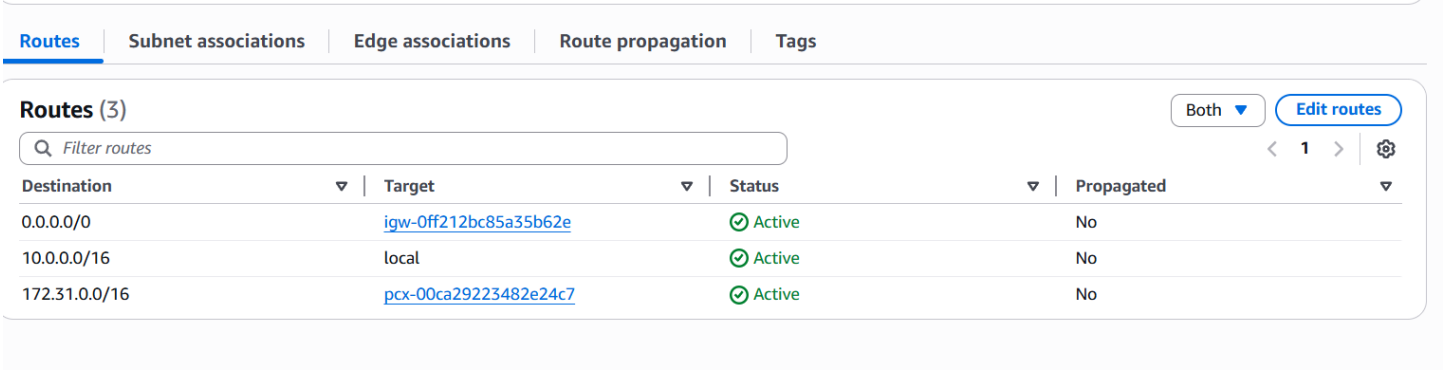
• Route tables must also be edited!



Below shows the attached Internet Gateway
and the route table records being attached to the Internet Gateway

PUBLIC SUBNET	PRIVATE SUBNET
Its personal Route table is created pointing the Internet Gateway	Its personal Route table is created either pointing to a Bastion Host or private networks only





Now Ritvik You just talked About how public subnets are visible but Private subnets are not

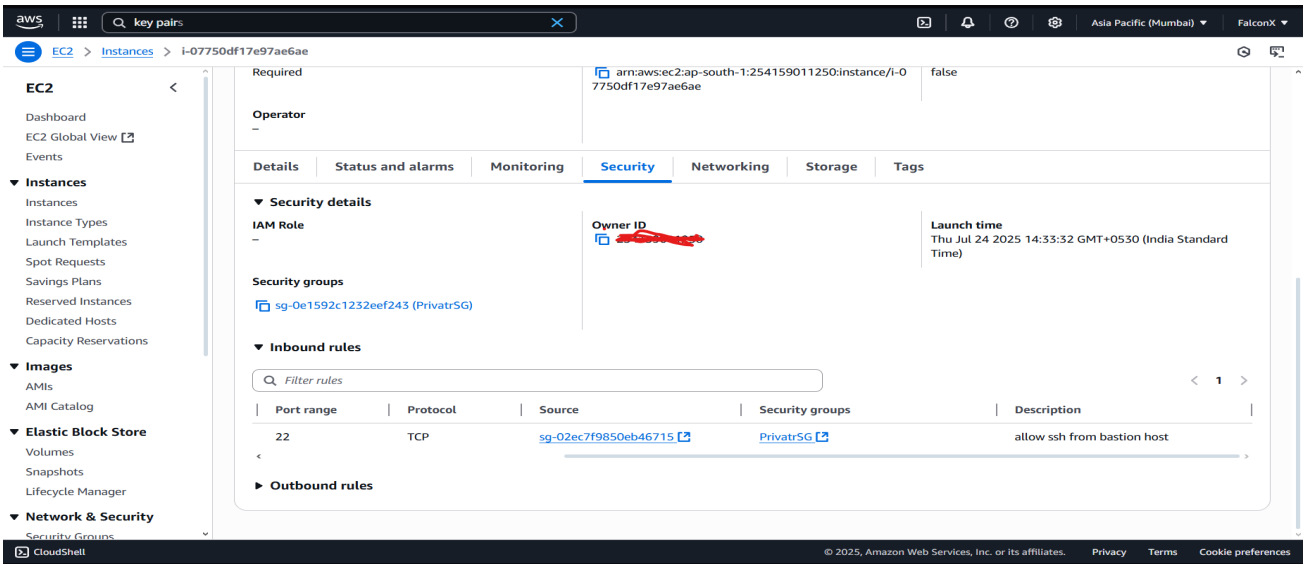
So what if I want to access my instaces /App on the private instance for the public but I don t want to show Them #Principle of OOPs-->Abstraction

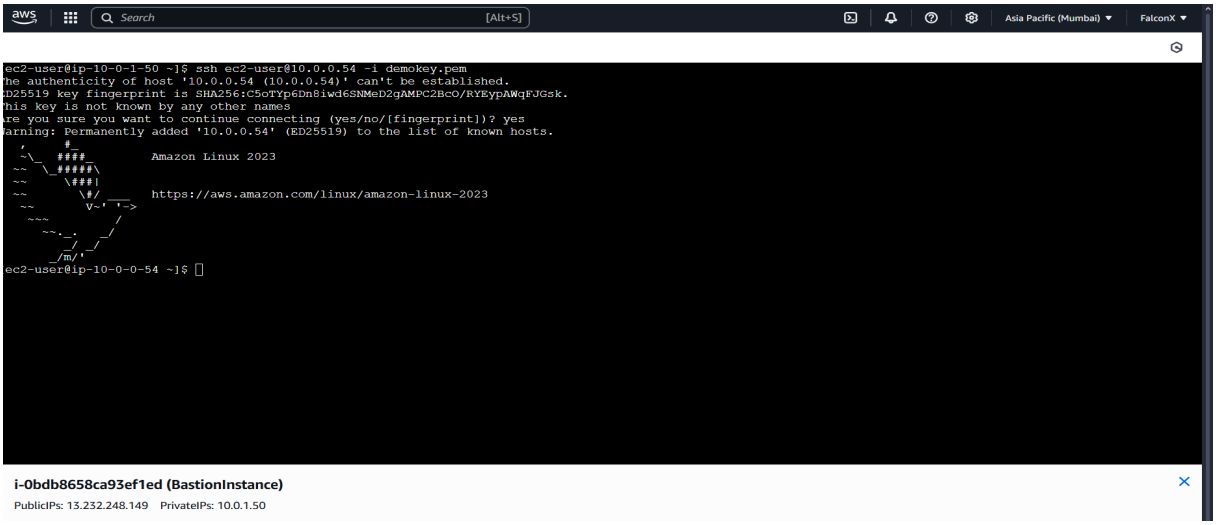
Bastion Hosts (Step-2.5)

These are the types of instances which are hosted on public subnets have access and are visible to the public and can operate resources to the private subnet instances simply but doing SSH into those instances

Here I have

1. Created instancen amed bastion host in public subnet and one in private subnet
2. Modified the security group inbound rules of private instance to allow traffic on prot 22(SSH Port) to point FROM THE SECURITY GROUP OF BASTION HOST
3. Now I ssh into the Bastion host
4. From here I ssh into the private instance – make a .pem file ,import it in the terminal itself and echo a “Hello” in Private instace
5. As soon as I curl cmd(For calling) it in the public instance it returns the desired output

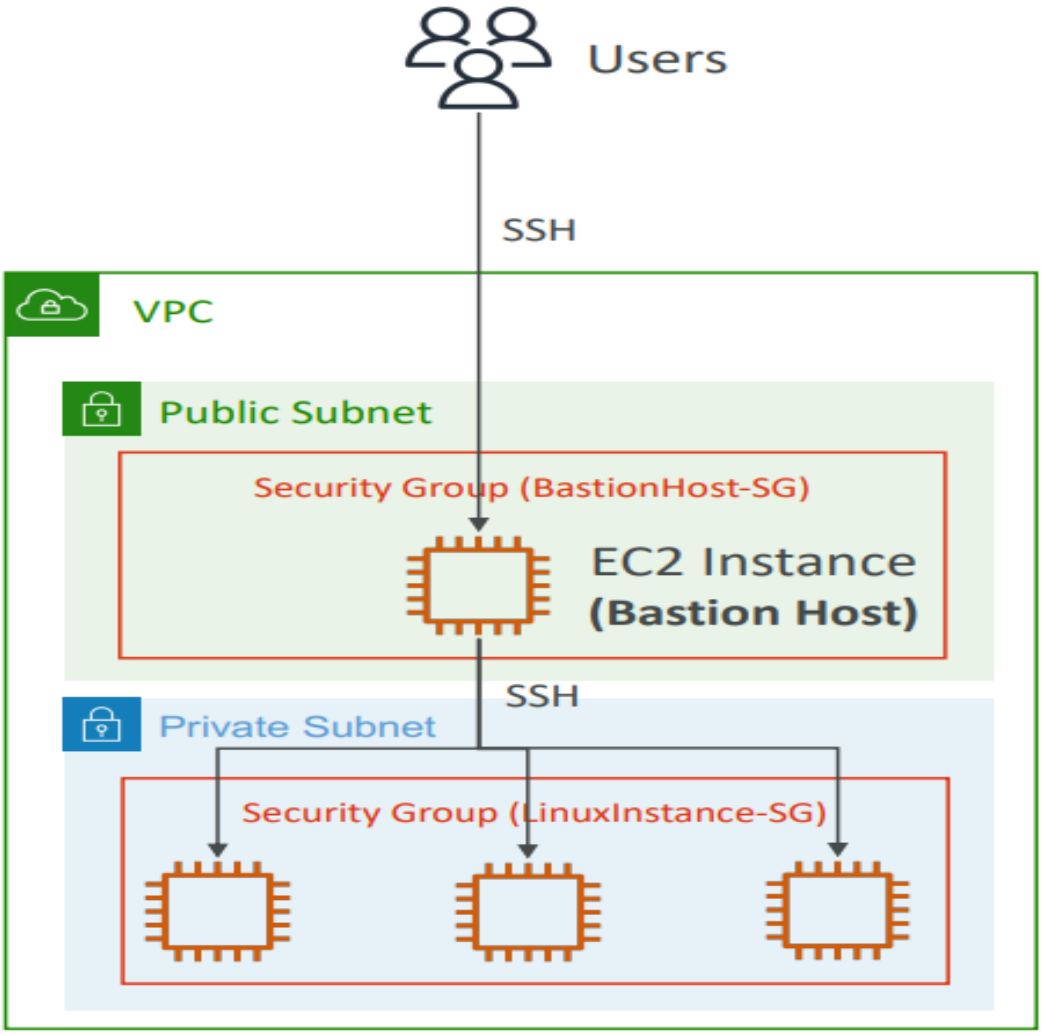




ExpectedOutput

```
[ec2-user@ip-172-31-3-7 ~]$ curl 10.0.1.50
hello
[ec2-user@ip-172-31-3-7 ~]$
```

NOW THE Structure LOOK LIKE THIS



Step 3

Increasing level [#########----]

This method is great but not reliable for bigger workloads so

NAT GATEWAY

This thing ensures a stable internet conenction both in and out be given to private subnets with a window for security cheks ofc!

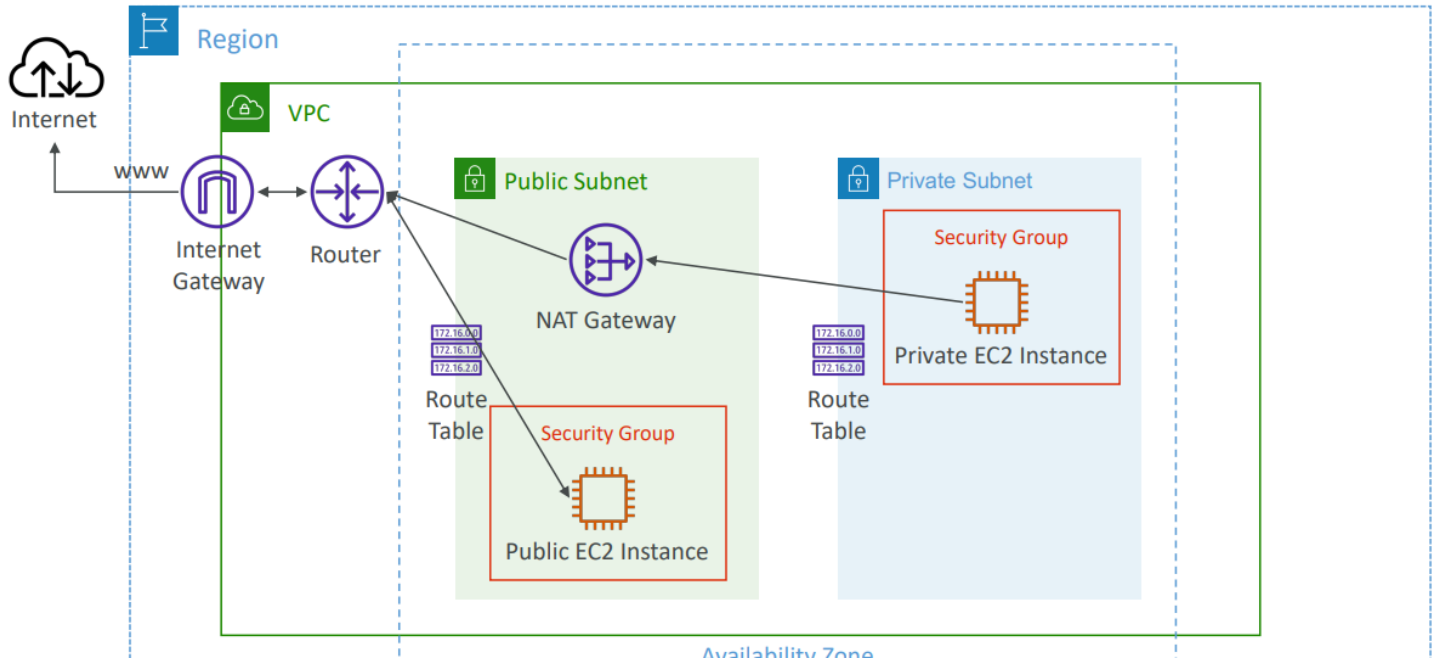
AWS-managed NAT, higher bandwidth, high availability, no administration

• NATGW is created in a specific Availability Zone, uses an Elastic IP

• Can’t be used by EC2 instance in the same subnet (only from other subnets)

• Requires an IGW (Private Subnet => NATGW => IGW)

• 5 Gbps of bandwidth with automatic scaling up to 100 Gbps



Now Comes the Security Part to secure the infrastructure for

1.Malicious Traffic

2. Misguided Traffic

Step-4

NACL(s) /Security Groups

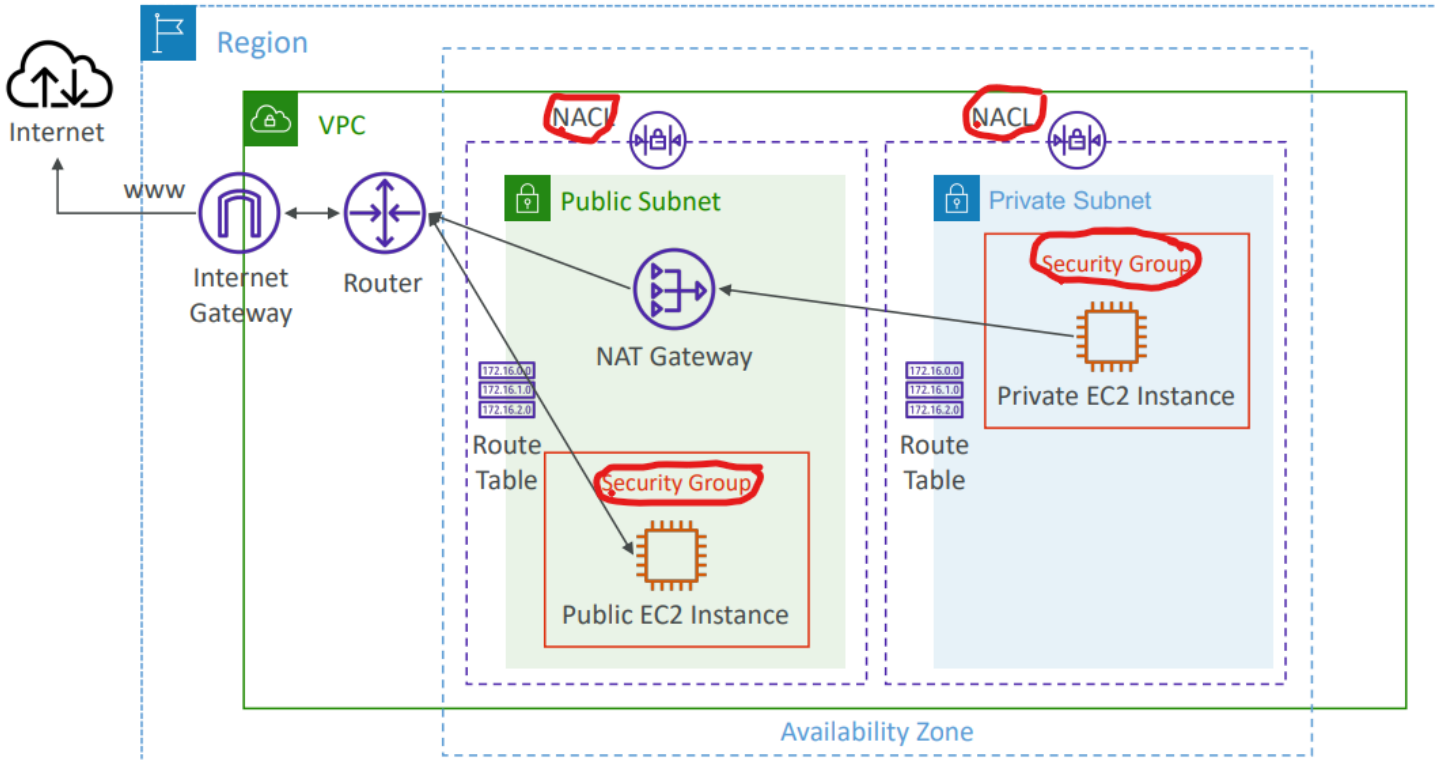
NACL(s) are Subnet Level completely stateless(Check both in and out traffic) and are based on Rules priority no.s

Security groups are instance level to further prevent the misguided traffic to reach its correct path and may not activate a wrong instance these are stateful(logical people if in then out and vice versa types) and have no priority no. System just declarations

There are many palces wehre I used NACLs to block certain CIDRs

And off those Which enter inside to guide through security groups

So here is our progress



NOTE:

EPHIMERAL PORTS- these are on spot usable ports used instead of defined ports for commmunication for added security

Now If the app is Enterprise grade and the infrastructure is also Enterprise grde
It can not rely on only 1 VPC

We have to have multiple VPC but that is not the point that is simple the point is

WE NEED SYNC and updated Information between all the VPCs for the companies

Also an Advantage of this will be to share info across VPC ,,,,next Feature

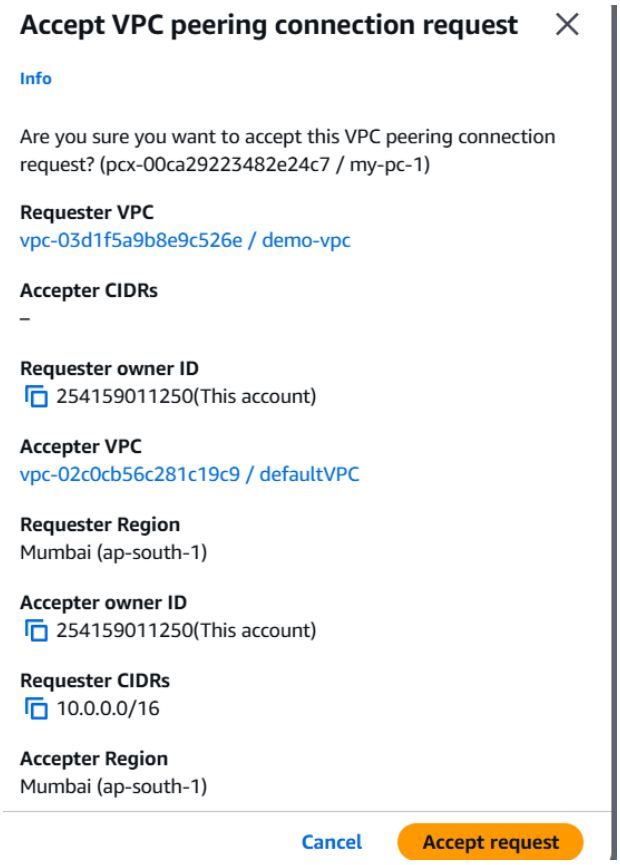
Step 4

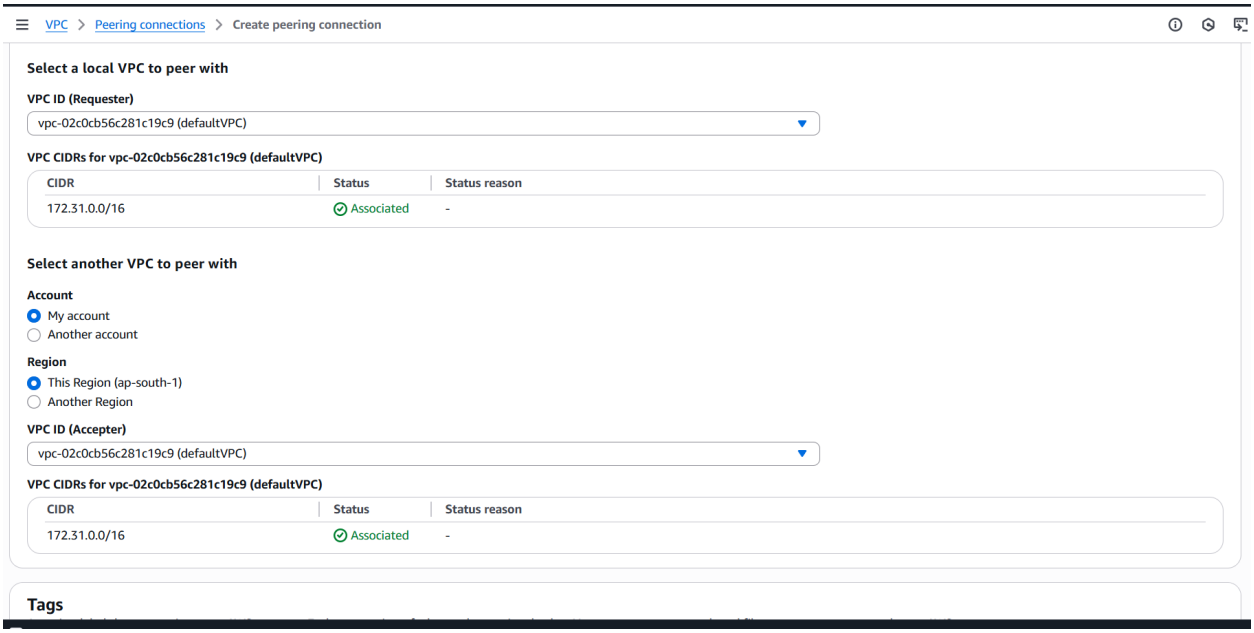
VPC Peering

It simply means establishing a peering connection or simply a connection between VPCs CROSS ACCOUNT & CROSS REGION

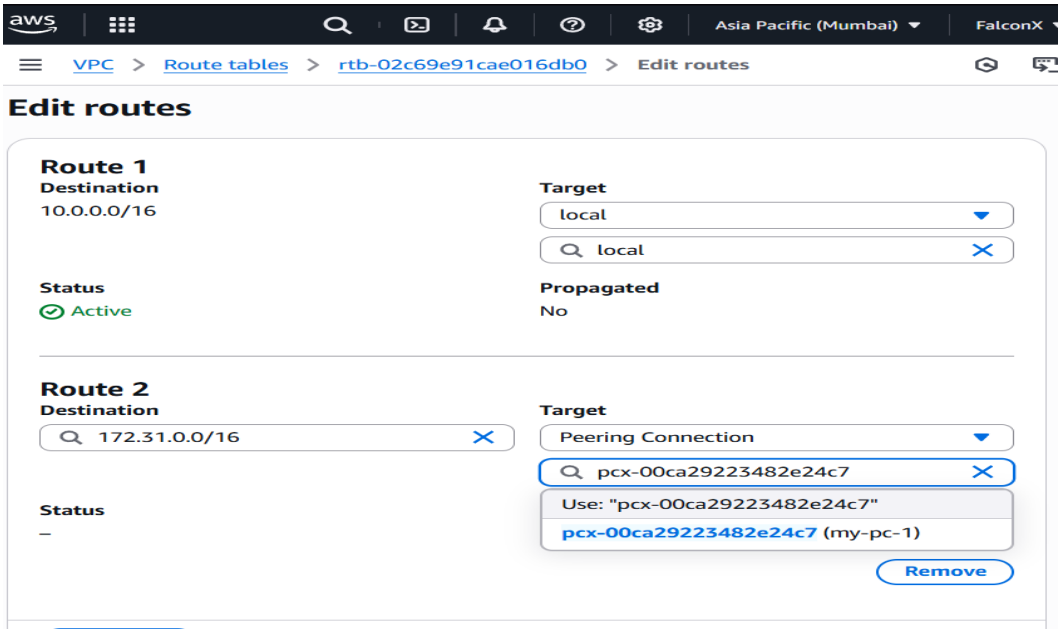
Here I set up

1. **TWO VPCs in different region**
2. **Accept the request of peering**
3. **MOST IMP. ALWAYS update it in the route tableswhich I will see in common once both VPCs are connected as shown**





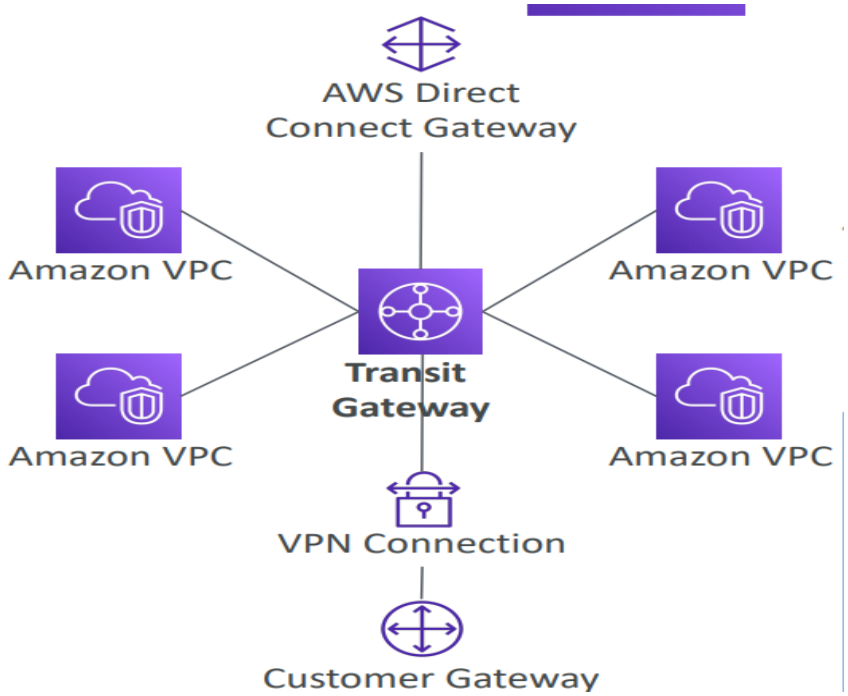
Changing the route Table



ADDITIONAL STEP FOR ENTERPRISE GRADE

Setup a Transit Gateway for incase of connecting multiple VPCs together using peerings

Reason:Network Topology becomes **complicated** and difficult to maintain



Some Cost Optimzing approaches....

Okay So Now the thing is I want my Private Instance to access AWS services or my private DATA Center

Why should i incurr such costs of setting up a NAT gateway then accessing those services through public

That's expensive and not even secure

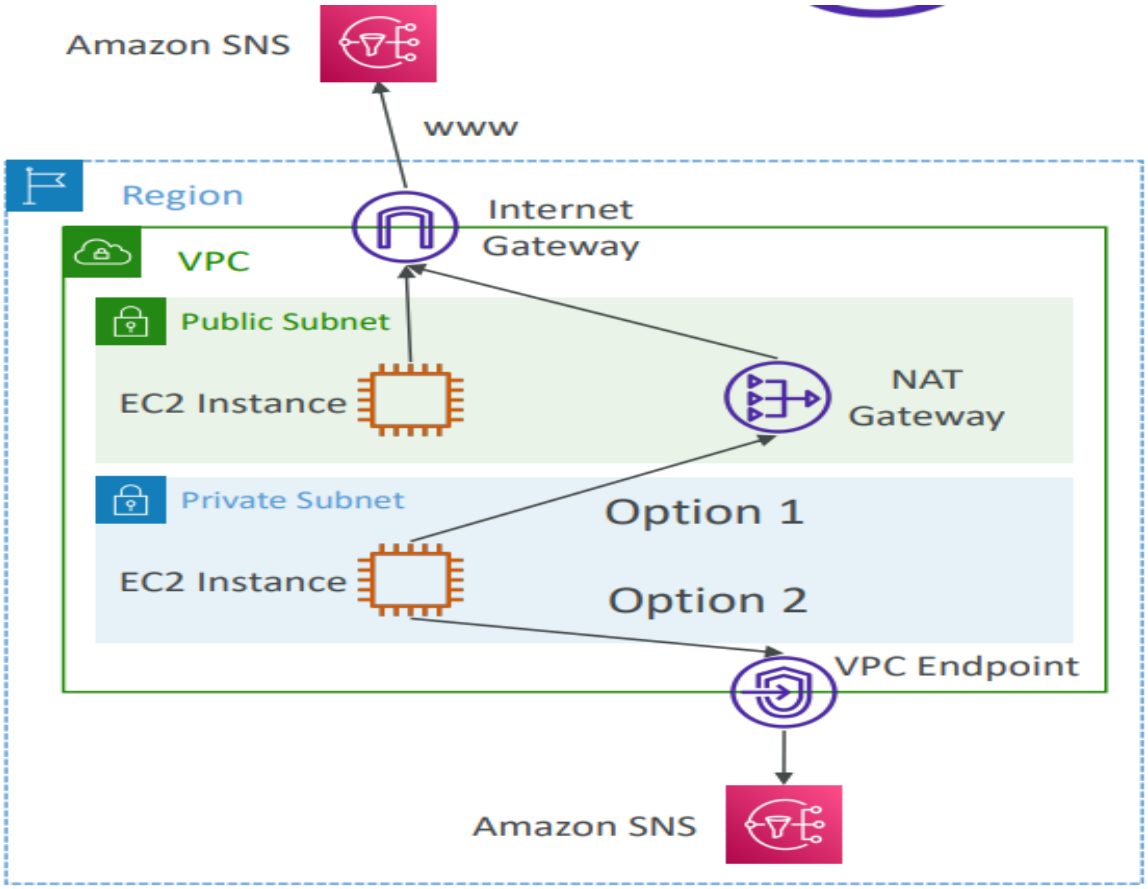
Hence

Step 5

VPC Endpoints

These are like metro connecting dots😅

VPC Endpoints (powered by AWS PrivateLink) allows you to connect to AWS services using a private network instead of using the public Internet



Now this is What I meant

These are also of two types one is a gatway endpoint supporting less services(FOR Small companies)

2. is Interface endpoints(Basically Endpoint premium subscription for Enterprises)

Finally.... So long but my VPC is all set !!!!

A final step remains.....

Imagine a city without police or fire Department

A school without rules and regulations

A college Without 75% criteria(Definitely mine😂)

A VPC sould also require to have a logging system to cath all the flow logs like how the traffic is flowing in and out and how is it behaving

STEP-Final

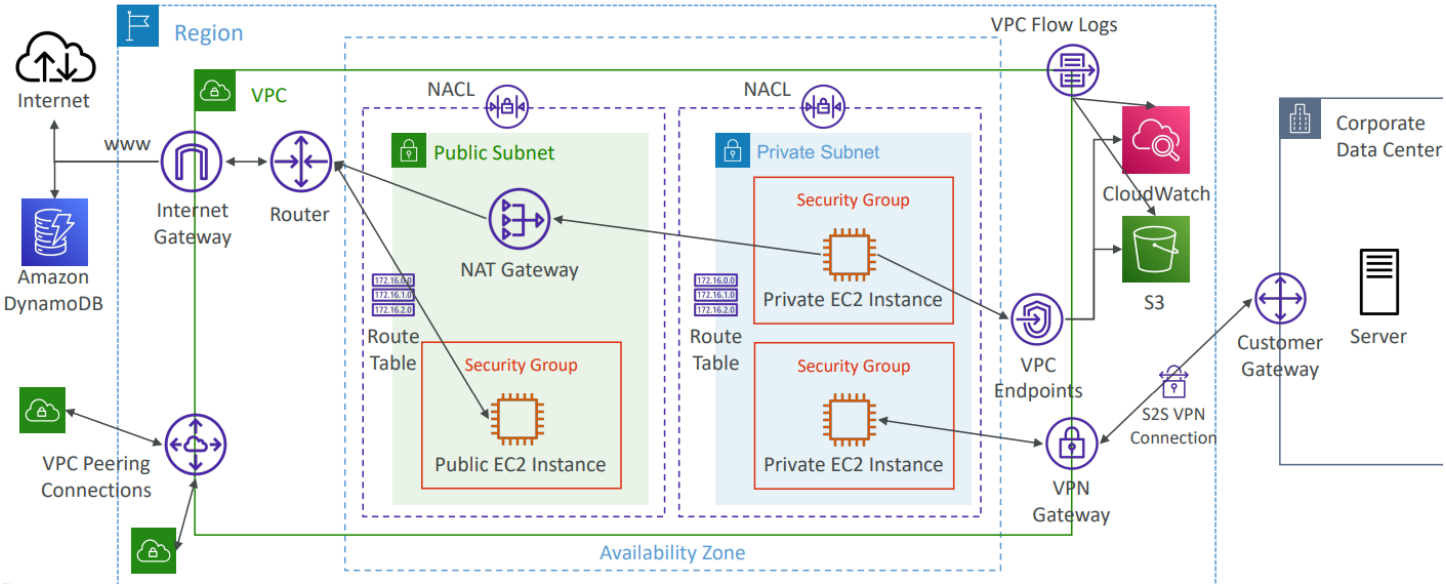
VPC FlowLogs

Flow logs data can go to S3, CloudWatch Logs, and Kinesis Data Firehose

Captures network information from AWS managed interfaces too: ELB, RDS, ElastiCache, Redshift, WorkSpaces, NATGW, Transit Gateway…

Some important things while viewing the log data from log groups

- srcaddr & dstaddr – help identify problematic IP
- srcport & dstport – help identity problematic ports
- Action – success or failure of the request due to Security Group / NACL
- Can be used for analytics on usage patterns, or malicious behavior
- Query VPC flow logs using Athena on S3 or CloudWatch Logs Insights



Some of the Parts like the direct connection to Corporate dta centers will be coverd later in the documentation

I am still writing it.. Figuring out how to document

IF YOU HAVE MADE THIS FAR💔💔

IF you liked my way of documenting things or would like to suggest changes feel free to message me and like the post

Hope you have understood my take of best practices in creatign a vpc for an enterprse grade or a startup grade web app

SEE YOU SOON