TRANSIT GATEWAY

A transit gateway is a network transit hub that you can use to interconnect your virtual private clouds (VPCs) and on-premises networks. As your cloud infrastructure expands globally, inter-Region peering connects transit gateways together using the AWS Global Infrastructure. All network traffic between AWS data centers is automatically encrypted at the physical layer.

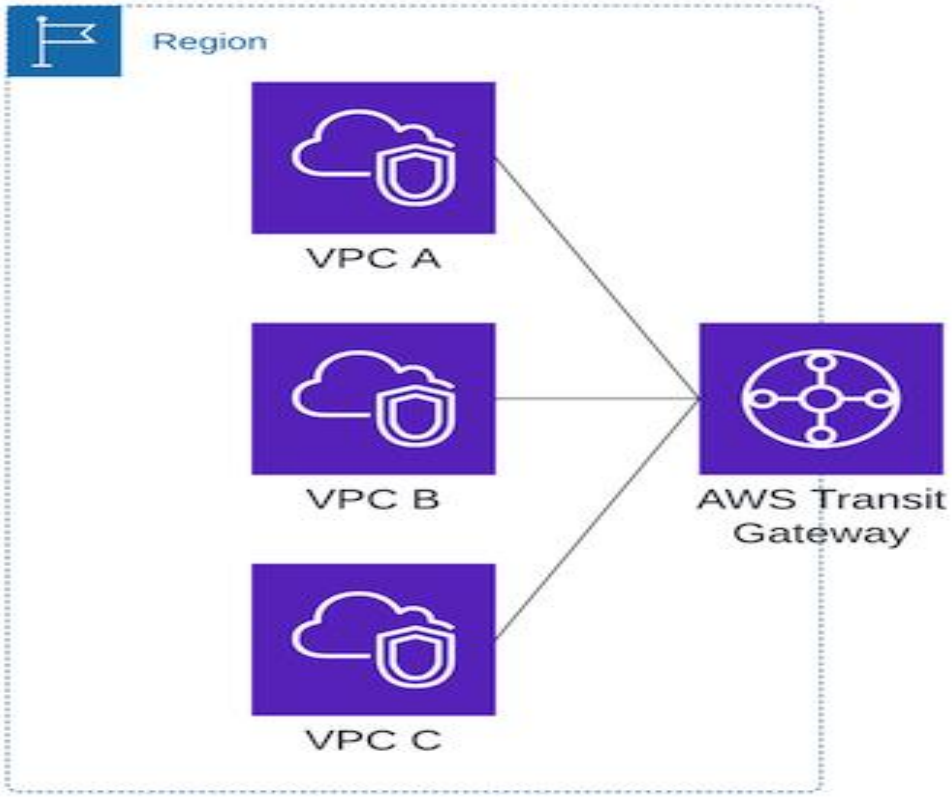


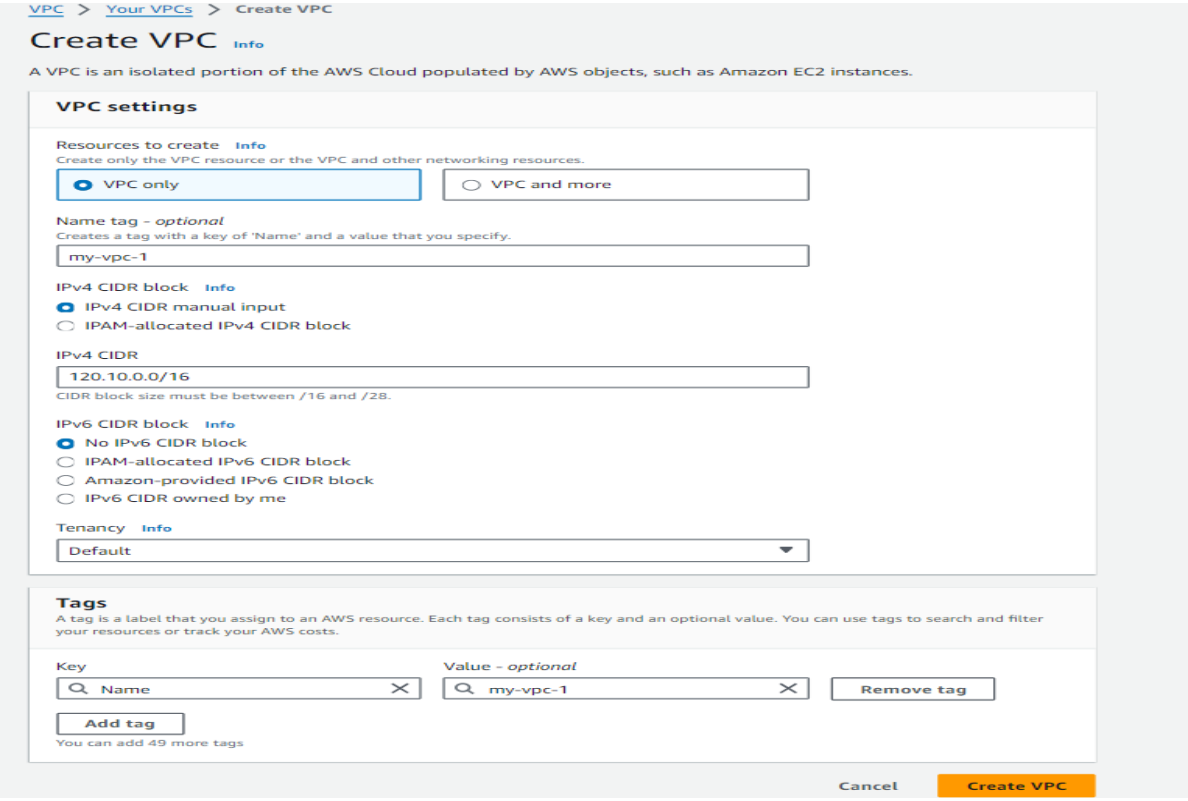
Fig1:Transit Gateway

Components of Transit Gateway:

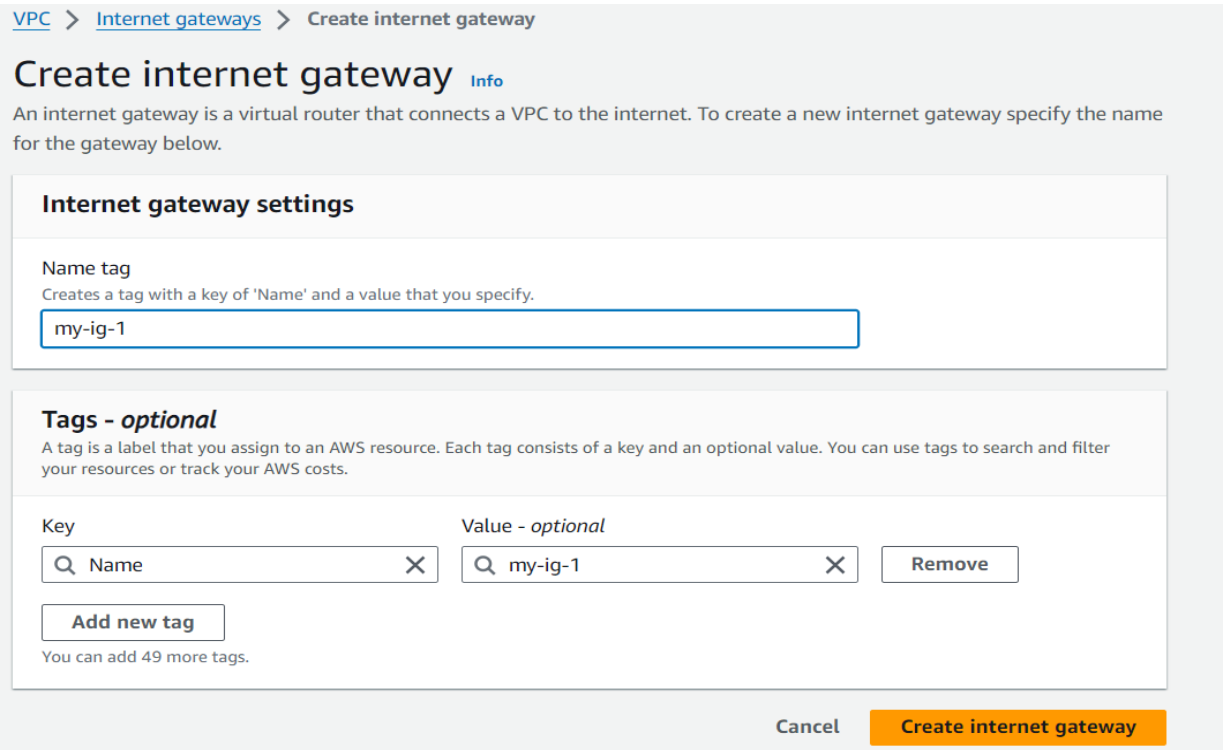
- **Transit Gateway :** Central hub that connects multiple VPCs and on-premises networks.
- **Attachments :** Connection between Transit Gateway and vcp's.
- **Route Tables :** Specifies how traffic should be routed between them.

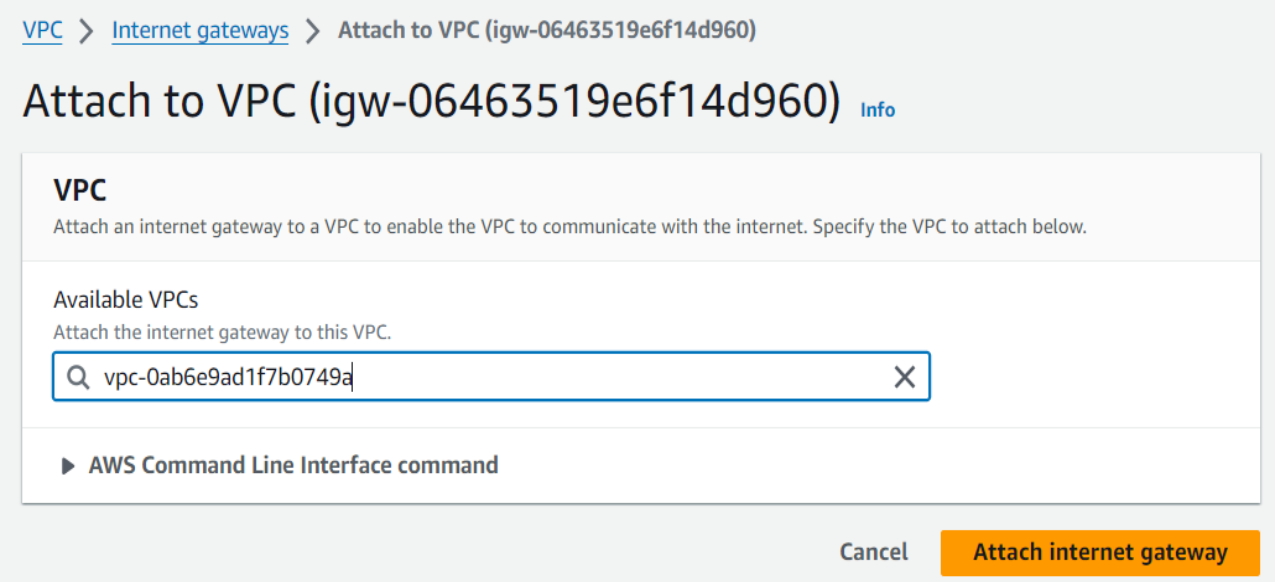
Steps to connect three vpc's to Transit Gateway :

- Open AWS Console and then open VPC.
- Click on Create VPC and select VPC .

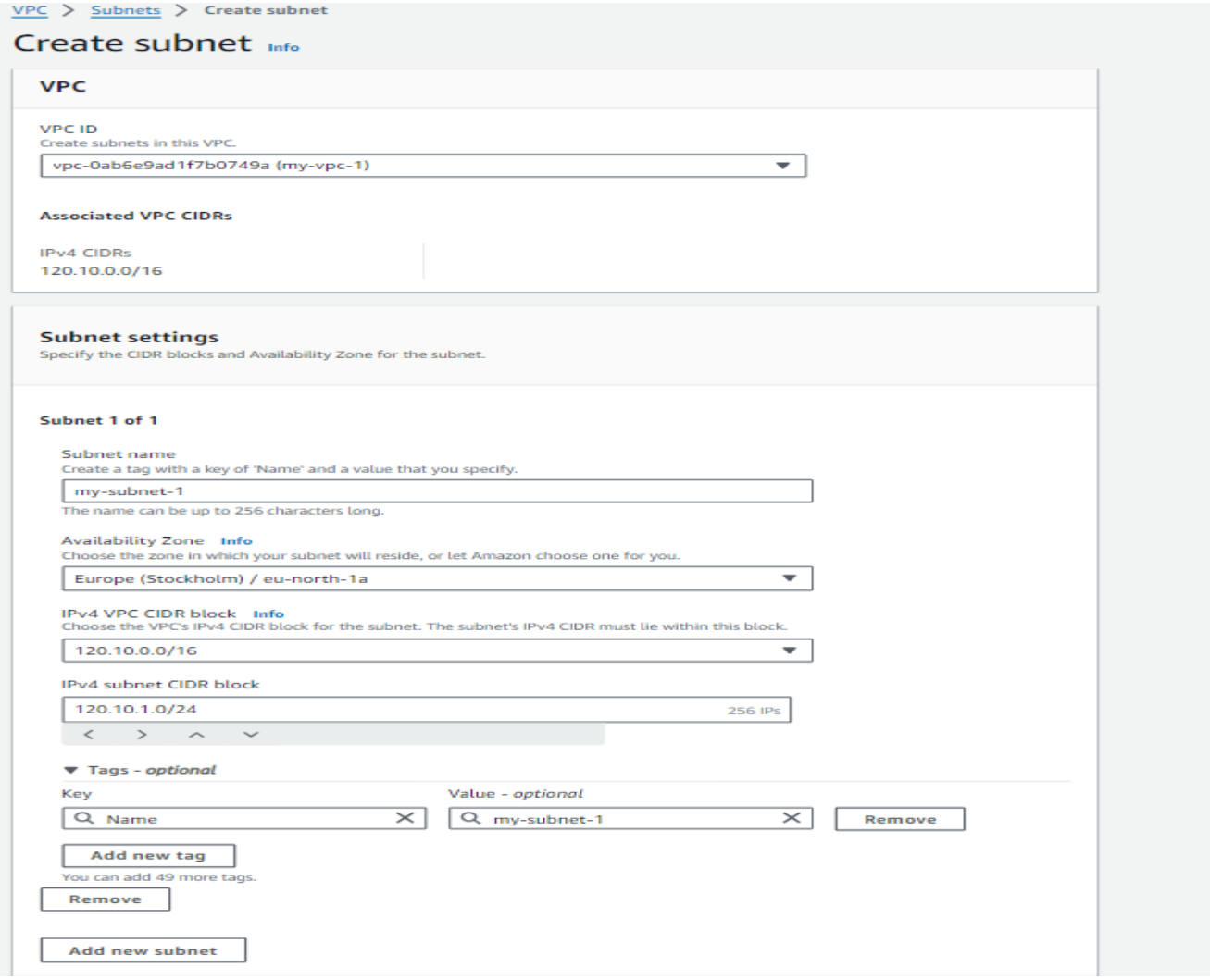


- Create internal gateway to VPC-1and attack to VPC-1





- Create subnet-1-public for VPC-1



- Now we need to add the route tables , Click on Route Tables from the LHS panel and click on create route table

VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
my-routetable-1

VPC
The VPC to use for this route table.
vpc-0ab6e9ad1f7b0749a (my-vpc-1)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	my-routetable-1	Remove

Add new tag
You can add 49 more tags.

Cancel | Create route table

Next, we have to associate the subnet with routing table. For that select VPC-1-Route -> Click on Subnet associations -> Edit subnet associations, then select VPC-1 Public-Subnet1 ->Save associations.

Edit subnet associations
Change which subnets are associated with this route table.

Available subnets (1/1)
Filter subnet associations

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
my-subnet-1	subnet-09db6da59b940b4dc	120.10.1.0/24	–	rtb-01f951781bab97d69 / my-routetable-1

Selected subnets
subnet-09db6da59b940b4dc / my-subnet-1

Cancel | Save associations

Select the VPC-A-Route and go to Routes->Edit routes and add as per below , then click on save changes.

VPC > Route tables > rtb-01f951781bab97d69 > Edit routes

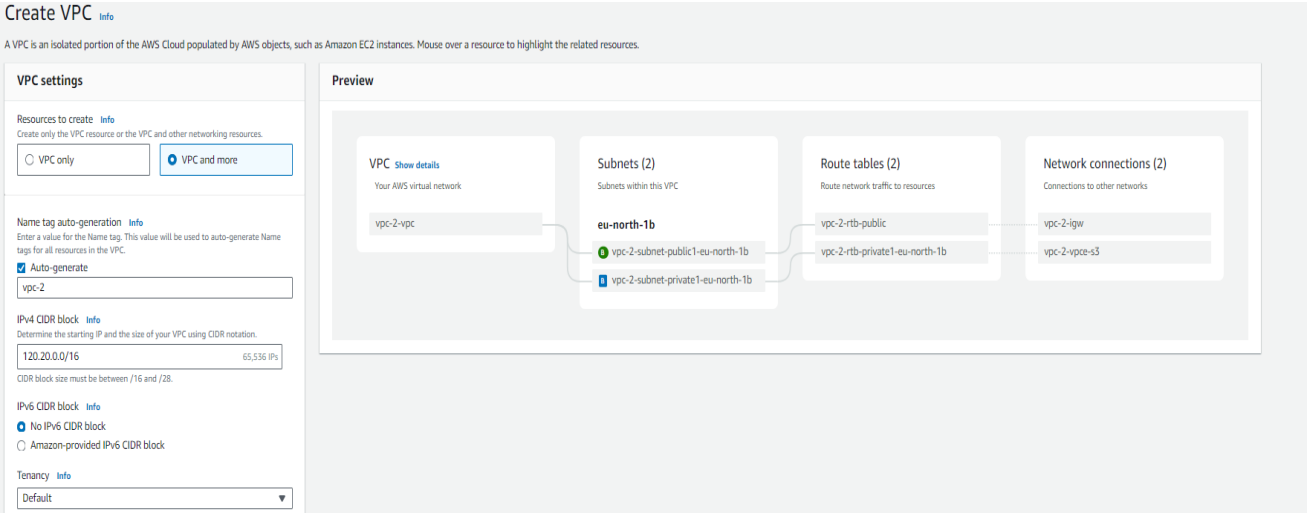
Edit routes

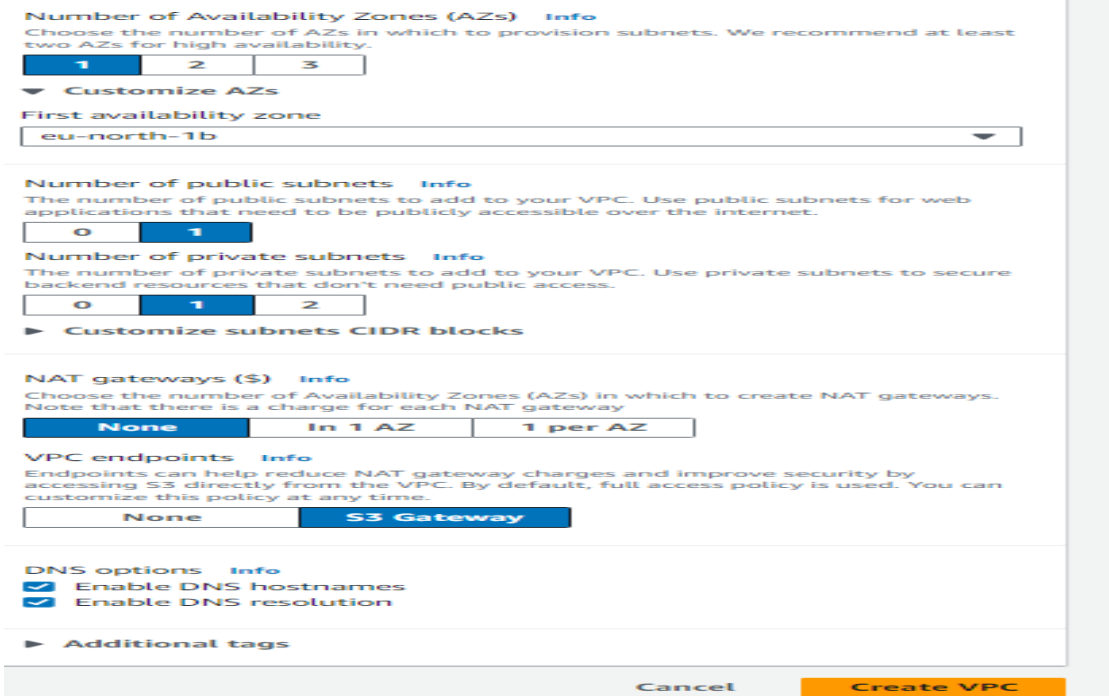
Destination	Target	Status	Propagated	
120.10.0.0/16	local	Active	No	
	local			
0.0.0.0/0	Internet Gateway	–	No	Remove
	igw-06463519e6f14d960			

Add route

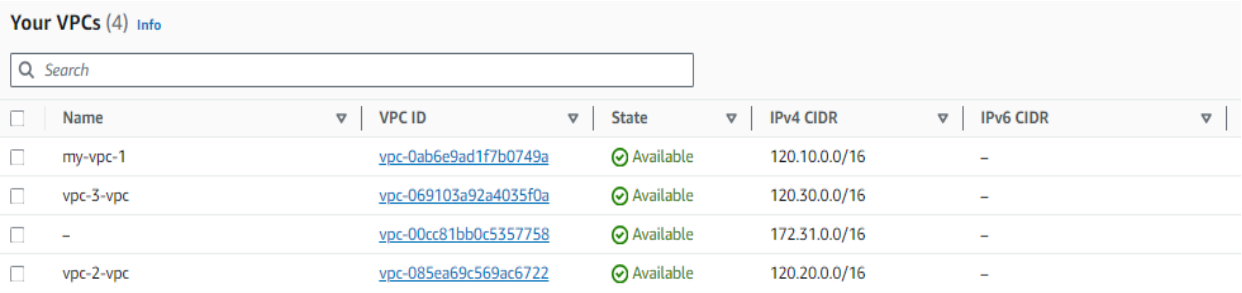
Cancel | Preview | Save changes

- Now create another VPC with using of VPC and more





- I Was Created VPC-2 in Stockholm (us north 1b) with 2 subnets (1 public, 1 Private) & 2 Route Tables(1 Public, 1 Private) with 1 internet gateway of IP (120.20.0.0/16).
- By the same process create VPC-3 with (120.30.0.0/16).

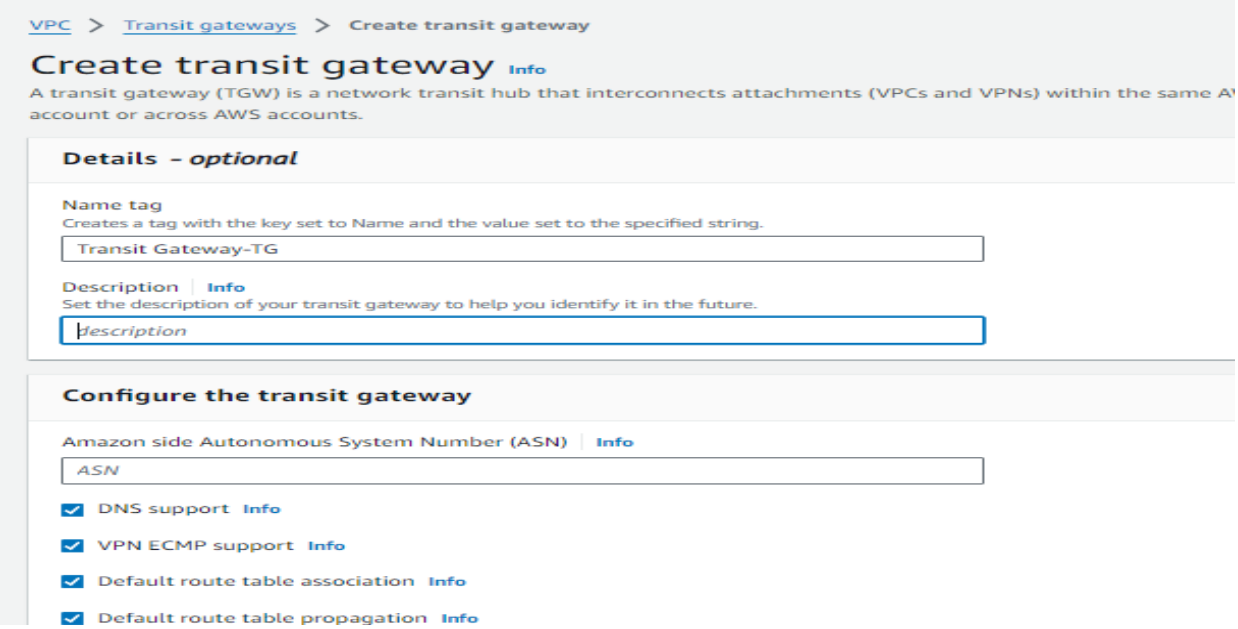


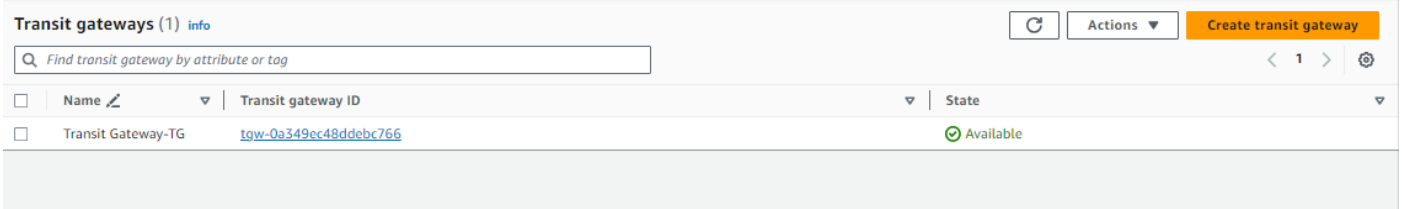
Your VPCs (4) Info

Search

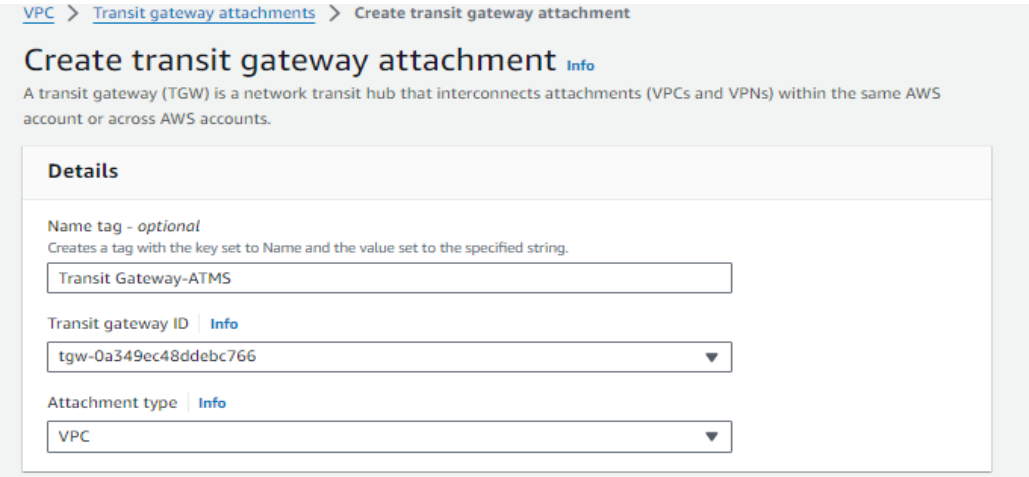
Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
my-vpc-1	vpc-0ab6e9ad1f7b0749a	Available	120.10.0.0/16	–
vpc-3-vpc	vpc-069103a92a4035f0a	Available	120.30.0.0/16	–
–	vpc-00cc81bb0c5357758	Available	172.31.0.0/16	–
vpc-2-vpc	vpc-085ea69c569ac6722	Available	120.20.0.0/16	–

- Next Click on Transit Gateway and Create Transit Gateway with name (Transit Gateway-TG).





- After creating Transit Gateway click on Transit Gateway Attachments as shown below.
- Create transit gateway attachments.



- Given name as TG-1.
- Select VPC-1 for attachmen.
- By the same process create attachments for TG-2 & TG-3 as Follow VPC-2 & VPC-3.

Transit gateway attachments (5) info

Find transit gateway attachment by attribute or tag

Name	Transit gateway attachment ID	Transit gateway ID	State	Resource type	Resou
transit-gatway-2	tgw-attach-0a4e4c7ddb9e4d258	tgw-0bad599222c11ba6b	Deleted	Peering	tgw-0
transit-gateway-3	tgw-attach-002a467e061193841	tgw-0bad599222c11ba6b	Available	VPC	vpc-0
transit-gateway-2	tgw-attach-044290778068f2e2c	tgw-0bad599222c11ba6b	Available	VPC	vpc-0
transitgateway-attac...	tgw-attach-073e54aa8eaa0f512	tgw-0bad599222c11ba6b	Deleted	VPC	vpc-0
transit-gateway-1	tgw-attach-0aa332da14166208e	tgw-0bad599222c11ba6b	Available	VPC	vpc-0

- Next go to route tables and click on VPC-1.
- Click on Add Routes and add IP's of VPC-2 & VPC-3.

Edit routes

Destination	Target	Status	Propagated	
120.10.0.0/16	local	Active	No	
120.20.0.0/16	Transit Gateway tgw-0bad599222c11ba6b	Active	No	Remove
120.30.0.0/16	Transit Gateway tgw-0bad599222c11ba6b	Active	No	Remove
0.0.0.0/0	Internet Gateway igw-06463519e6f14d960	Active	No	Remove

Add route

Cancel | Preview | Save changes

rtb-01f951781bab97d69 / my-routetable-1

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-01f951781bab97d69	No	subnet-09db6da59b940b4dc / my-subnet-1	–
VPC	**Owner ID**		
vpc-0ab6e9ad1f7b0749a | my-vpc-1	980186824762		

Routes | Subnet associations | Edge associations | Route propagation | Tags

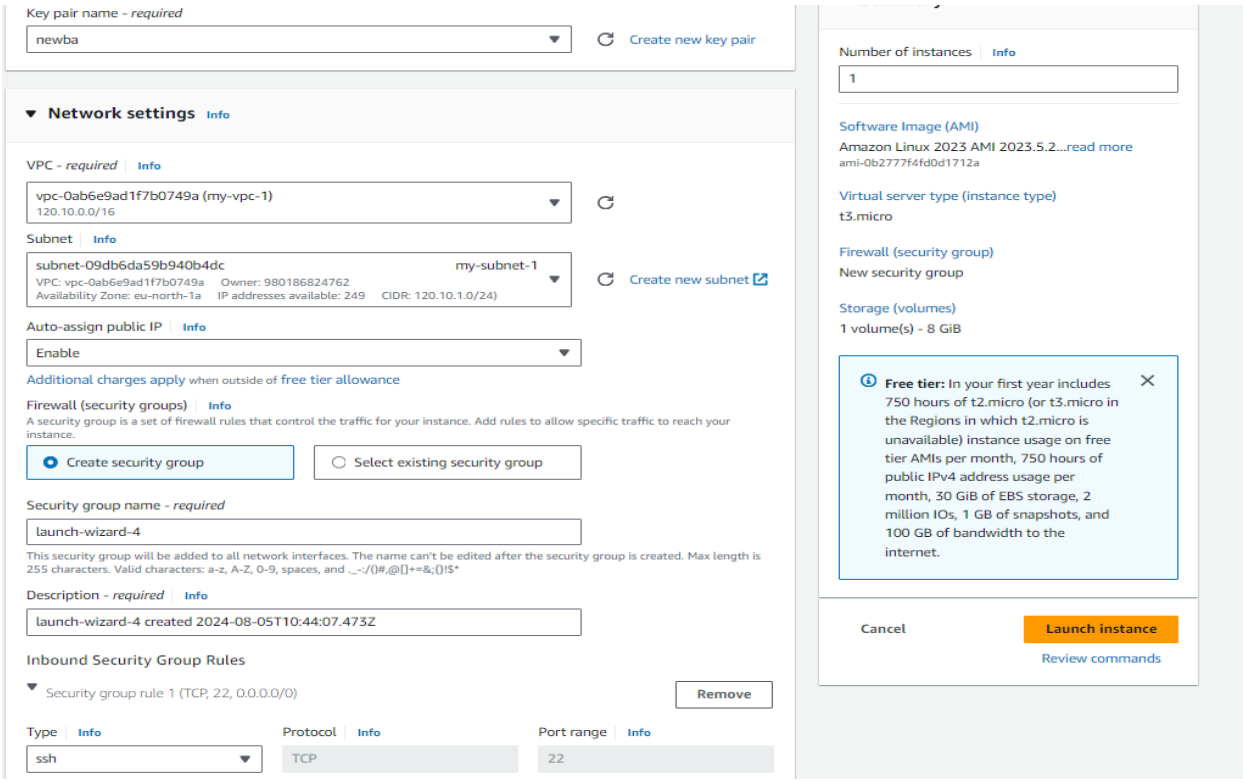
Routes (4) Both

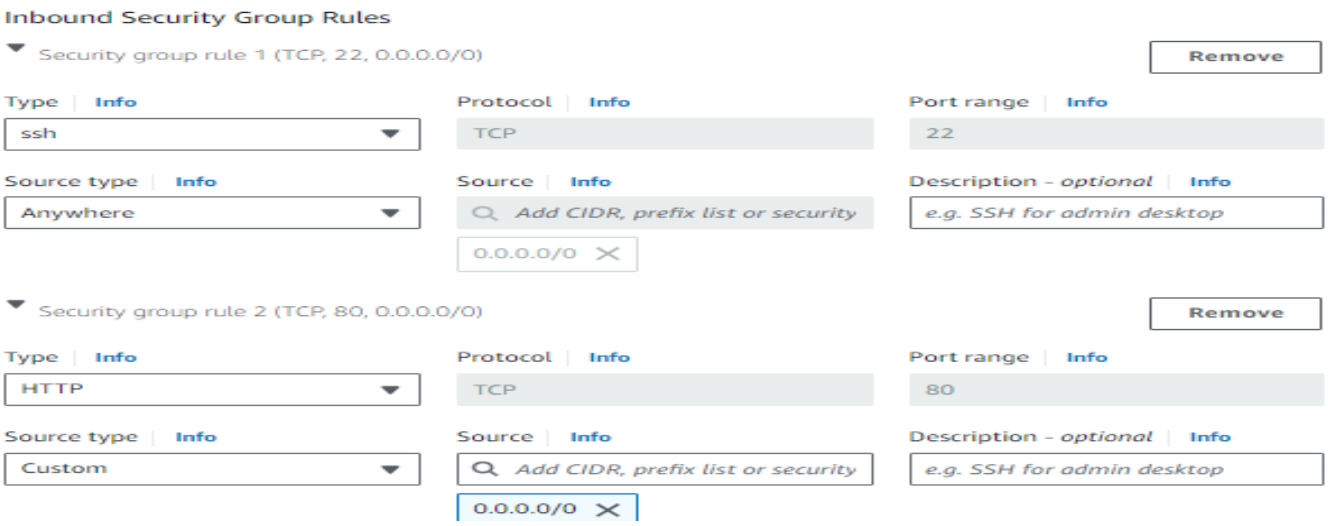
Filter routes

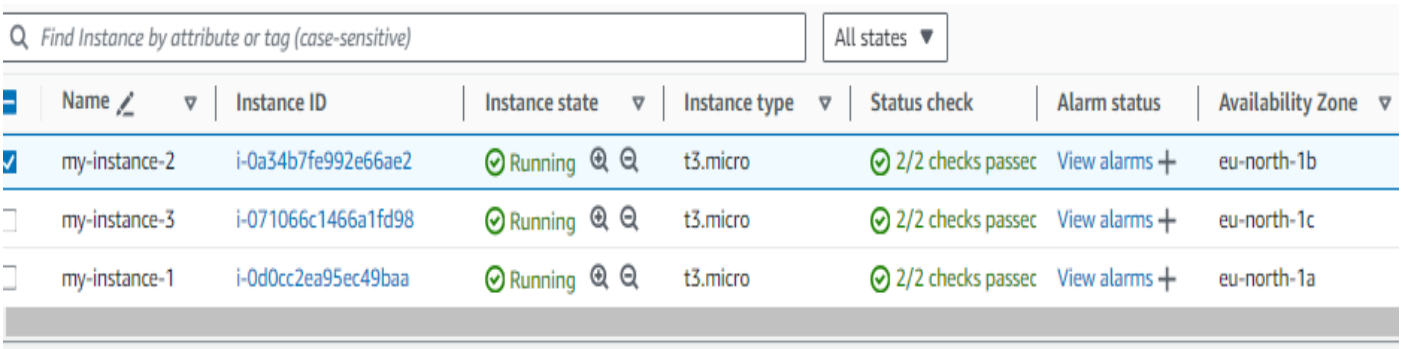
Destination	Target	Status	Propagated
0.0.0.0/0	igw-06463519e6f14d960	Active	No
120.10.0.0/16	local	Active	No
120.20.0.0/16	tgw-0bad599222c11ba6b	Active	No
120.30.0.0/16	tgw-0bad599222c11ba6b	Active	No

- By configuring Route Tables Next we need to create 3 Instances for VPC-1, VPC-2 &VPC-3.
- Click on EC-2 instance. Then click on create instance.
- Give instance name as server-1 and select OS & Create Key Pair.
- Create Security Group & Add rules for SSH and HTTP as follows 22 & 80 port numbers.

- By the same process create instances for VPC-2 & VPC-3 with names as Server-2 & Server-3.







- After succefull creation of instances connect to instances and install nginx and create an .html file for our recognization.
- By the same process we need to do in 3 servers.
- First connect to the server 1 in git bash and type sudo -i for the root user.
- And then type yum update -y && yum install nginx -y && cd /usr/share/nginx/html in git bash.
- Remove index.html file and then create index.html file and insert data to the file as “hi this is bhargav from north 1a”.
- By the same do in server-2 and server-3.

Output From Server-1 :

```
[root@ip-120-10-1-155 html]# yum update -y && yum install nginx -y && cd /usr/share/nginx/html
Last metadata expiration check: 1:23:20 ago on Mon Aug  5 09:25:38 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 1:23:21 ago on Mon Aug  5 09:25:38 2024.
Package nginx-1:1.24.0-1.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-120-10-1-155 html]#
[root@ip-120-10-1-155 html]# cat index.html
hi this bhargav from north-1a
[root@ip-120-10-1-155 html]# systemctl restart nginx
[root@ip-120-10-1-155 html]# curl 120.10.1.155
hi this bhargav from north-1a
[root@ip-120-10-1-155 html]# curl 120.20.0.133:80
hi this is bhargav from north-1b
[root@ip-120-10-1-155 html]# curl 120.30.1.123:80
hi this is bhargav from north-1c
[root@ip-120-10-1-155 html]#
```

Output From Server-2 :

```
balineni bhargav@BHARGAV MINGW64 ~/OneDrive/Desktop (master)
$ ssh -i "newba.pem" ec2-user@ec2-16-171-165-84.eu-north-1.compute.amazonaws.com
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Aug  5 09:38:06 2024 from 103.160.27.100
[ec2-user@ip-120-20-0-133 ~]$ sudo su
[root@ip-120-20-0-133 ec2-user]# yum update -y && yum install nginx -y && cd /us
r/share/nginx/html
Last metadata expiration check: 0:29:11 ago on Mon Aug  5 09:36:09 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:29:12 ago on Mon Aug  5 09:36:09 2024.
Package nginx-1:1.24.0-1.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-120-20-0-133 html]# vi index.html
[root@ip-120-20-0-133 html]# systemctl restart nginx
[root@ip-120-20-0-133 html]# curl 120.20.0.133:80
hi this is bhargav from north-1b
[root@ip-120-20-0-133 html]# curl 120.10.1.155:80
hi this bhargav from north-1a
[root@ip-120-20-0-133 html]# curl 120.30.1.123:80
hi this is bhargav from north-1c
[root@ip-120-20-0-133 html]#
```

Output From Server-3 :

```
balineni bhargav@BHARGAV MINGW64 ~/OneDrive/Desktop (master)
$ ssh -i "newba.pem" ec2-user@ec2-13-53-172-204.eu-north-1.compute.amazonaws.com
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Aug  5 09:39:08 2024 from 103.160.27.100
[ec2-user@ip-120-30-1-123 ~]$ sudo su
[root@ip-120-30-1-123 ec2-user]# yum update -y && yum install nginx -y && cd /us
r/share/nginx/html
Last metadata expiration check: 0:30:26 ago on Mon Aug  5 09:36:57 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:30:27 ago on Mon Aug  5 09:36:57 2024.
Package nginx-1:1.24.0-1.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-120-30-1-123 html]# vi index.html
[root@ip-120-30-1-123 html]# systemctl restart nginx
[root@ip-120-30-1-123 html]# curl 120.30.1.123:80
hi this is bhargav from north-1c
[root@ip-120-30-1-123 html]# curl 120.20.0.133:80
hi this is bhargav from north-1b
[root@ip-120-30-1-123 html]# curl 120.10.1.155:80
hi this bhargav from north-1a
[root@ip-120-30-1-123 html]# |
```

Problems with Transit Vpc:

- Instance Based.
- Additional EC2 Cost.
- Software Licensing Cost.
- Availability Issues.
- Bandwidth Limitations of EC2.

Conclusion :

AWS Transit Gateway simplifies cloud network architectures by acting as a hub to interconnect your VPCs, VPNs, and data centers. It eliminates complex mesh topologies and provides easy scalability, centralized management, and secure network segmentation. As your cloud footprint grows, Transit Gateway is key to maintaining a simple, efficient, and secure network topology.

Balineni Bhargav

bargavrambalineni@gmail.com

7995693436

Batch No : 127