

Problem Statement:

You work for XYZ Corporation and based on the expansion requirements of your corporation you have been asked to create and set up a distinct Amazon VPC for the production and development team.

You are expected to perform the following tasks for the respective VPCs:

Production Network:

1. Design and build a 4-tier architecture.
 2. Create 5 subnets out of which 4 should be private named app1, app2, dbcache and db and one should be public, named web.
 3. Launch instances in all subnets and name them as per the subnet that they have been launched in.
 4. Allow dbcache instance and app1 subnet to send internet requests
 5. Manage security groups and NA
- 1.Create Production VPC (cidr - 10.10.0.0/16)

Development Network:

1. Design and build 2-tier architecture with two subnets named web and db and launch instances in both subnets and name them as per the subnet names.
2. Make sure only the web subnet can send internet requests. ‘
3. Create peering connection between production network and development network.
4. Setup connection between db subnets of both production network and development network respectively.

Step by step demonstration:

Note: I have performed all the tasks through AWS console. You can use Cloud formation/Terraform/CLI for resource creation.

There are EC2 instances which have been created on private subnets so directly we can't perform SSH. For making connections I used the Jump server/Bastion host concept. You may refer to any youtube video for explanation:

- 1.Create Production VPC (cidr - 10.10.0.0/16)

VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Production VPC

IPv4 CIDR block Info

☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.10.0.0/16

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

IPv6 CIDR block Info

☒ No IPv6 CIDR block
☐ IPAM-allocated IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block
☐ IPv6 CIDR owned by me

Tenancy Info

Default

2. Create 5 Subnets in it

- web subnet [public] (cidr - 10.10.1.0/24)
- app1 subnet [private] (cidr - 10.10.2.0/24)
- app2 subnet [private] (cidr - 10.10.3.0/24)
- db subnet [private] (cidr - 10.10.4.0/24)
- dbcache subnet [private] (cidr - 10.10.5.0/24)

You have successfully created 5 subnets: subnet-0c457d9f67a4678ad, subnet-002c6f151f0f78d28, subnet-0a4e57a1f51762741, subnet-0f677afdd7926d9f2, subnet-07c31412818d04a27

Subnets (5) Info

Find resources by attribute or tag

Subnet ID : subnet-0c457d9f67a4678ad X Subnet ID : subnet-002c6f151f0f78d28 X Subnet ID : subnet-0a4e57a1f51762741 X Show more (+3) Clear filters

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	Availat
app2 subnet	subnet-0a4e57a1f51762741	Available	vpc-06c4fd5a10ee79642 Production VPC	10.10.3.0/24	-	251
web subnet	subnet-0c457d9f67a4678ad	Available	vpc-06c4fd5a10ee79642 Production VPC	10.10.1.0/24	-	251
app1 subnet	subnet-002c6f151f0f78d28	Available	vpc-06c4fd5a10ee79642 Production VPC	10.10.2.0/24	-	251
dbcache subnet	subnet-07c31412818d04a27	Available	vpc-06c4fd5a10ee79642 Production VPC	10.10.5.0/24	-	251
db subnet	subnet-0f677afdd7926d9f2	Available	vpc-06c4fd5a10ee79642 Production VPC	10.10.4.0/24	-	251

- Select web subnet > Go to Action > Edit Subnet setting > Enable Auto Assign IPv4.

VPC > Subnets > subnet-0c457d9f67a4678ad > Edit subnet settings

Edit subnet settings [Info](#)

Subnet

Subnet ID	Name
subnet-0c457d9f67a4678ad	web subnet

Auto-assign IP settings [Info](#)

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

☒ Enable auto-assign public IPv4 address [Info](#)

☐ Enable auto-assign customer-owned IPv4 address [Info](#)
 Option disabled because no customer owned pools found.

3. Create Internet Gateway (Prod-IGW) and attach it to Prod VPC.

VPC > Internet gateways > igw-04ca5e970e81e2d53

igw-04ca5e970e81e2d53 / Prod-IGW

Details [Info](#)

Internet gateway ID	State	VPC ID
igw-04ca5e970e81e2d53	Attached	vpc-06c4fd5a10ee79642 Production VPC

Tags

Search tags

Key	Value
Name	Prod-IGW

4. Create Public Route Table

- edit routes
- destination - 0.0.0.0/0
- target - internet gateway (Prod-IGW)
- subnet association - web subnet

rtb-0762520e05b34cb30 / PublicRouteProd

Details [Info](#)

Route table ID rtb-0762520e05b34cb30	Main No	Explicit subnet associations -	Edge associations -
VPC vpc-06c4fd5a10ee79642 Production VPC	Owner ID 892543032022		

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

Routes (2)

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	igw-04ca5e970e81e2d53	Active	No
10.10.0.0/16	local	Active	No

Subnet associations | Edge associations | Route propagation | Tags

Explicit subnet associations (1) [Edit subnet associations](#)

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
web subnet	subnet-0c457d9f67a4678ad	10.10.1.0/24	-

5. Create NAT Gateway [in web subnet] (Prod-NAT)

Create NAT gateway [Info](#)

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

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

Prod-NAT

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

subnet-0c457d9f67a4678ad (web subnet)

Connectivity type
Select a connectivity type for the NAT gateway.

☒ Public

☐ Private

6. Create Private Route Table

- edit routes
- destination - 0.0.0.0/0
- target - nat gateway (Prod-NAT)
- subnet association - app1 subnet and dbcache subnet

rtb-05a84ddaa4011c7f9 / PrivateRouteProd

Details Info

Route table ID rtb-05a84ddaa4011c7f9	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-06c4fd5a10ee79642 Production VPC	Owner ID 892543032022		

Routes Subnet associations Edge associations Route propagation Tags

Routes (2) Both Edit routes

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0555f8352efdb2701	Active	No
10.10.0.0/16	local	Active	No

Routes Subnet associations Edge associations Route propagation Tags

Explicit subnet associations (2) Edit subnet associations

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
app1 subnet	subnet-002c6f151f0f78d28	10.10.2.0/24	-
dbcache subnet	subnet-07c31412818d04a27	10.10.5.0/24	-

7. Launch ec2 instances in all subnets of Production VPC and name them as per name of your subnets.

Instances (5) Info Refresh Connect Instance state

Find Instance by attribute or tag (case-sensitive) All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	db instance	i-0a88368d39ff5aec2	Running	t2.micro	Initializing	View alarms +	ap-south-1a	-
<input type="checkbox"/>	dbcache instance	i-053066dd1aec8b5ad	Running	t2.micro	Initializing	View alarms +	ap-south-1a	-
<input type="checkbox"/>	app2 instance	i-0811638700a38608d	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a	-
<input type="checkbox"/>	app1 instance	i-0ad1ed0a08d127abf	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a	-
<input type="checkbox"/>	web instance	i-0cfdd560eb37cb2f5	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a	-

8. Connect to your web instance and check the internet connectivity (ping google.com)

```
root@ip-10-10-1-192:/home/ubuntu# ping google.com
PING google.com (142.250.183.174) 56(84) bytes of data.
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=1 ttl=54 time=1.52 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=2 ttl=54 time=1.66 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=3 ttl=54 time=1.58 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=4 ttl=54 time=1.60 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=5 ttl=54 time=1.57 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=6 ttl=54 time=1.63 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=7 ttl=54 time=1.65 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=8 ttl=54 time=1.65 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=9 ttl=54 time=1.63 ms
64 bytes from bom07s32-in-f14.1e100.net (142.250.183.174): icmp_seq=10 ttl=54 time=1.60 ms
```

i-0cfdd560eb37cb2f5 (web instance)

PublicIPs: 13.235.8.93 PrivateIPs: 10.10.1.192

9. Connect to your app1/dbcache instance and check the internet connectivity [connect by bastion/jump server method] (ping google.com)
App1 instance:

```
ubuntu@ip-10-10-2-72:~$ ping google.com
PING google.com (142.250.183.46) 56(84) bytes of data.
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=1 ttl=53 time=2.35 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=2 ttl=53 time=2.05 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=3 ttl=53 time=2.10 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=4 ttl=53 time=2.07 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=5 ttl=53 time=2.11 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=6 ttl=53 time=2.13 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=7 ttl=53 time=2.15 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=8 ttl=53 time=2.10 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=9 ttl=53 time=2.18 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=10 ttl=53 time=2.10 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=11 ttl=53 time=2.12 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=12 ttl=53 time=2.06 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=13 ttl=53 time=2.08 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=14 ttl=53 time=2.12 ms
64 bytes from bom12s11-in-f14.1e100.net (142.250.183.46): icmp_seq=15 ttl=53 time=2.12 ms
```

DB cache instance:

```
ubuntu@ip-10-10-5-149:~$ ping google.com
PING google.com (142.250.70.110) 56(84) bytes of data.
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=1 ttl=53 time=2.88 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=2 ttl=53 time=2.56 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=3 ttl=53 time=2.60 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=4 ttl=53 time=2.57 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=5 ttl=53 time=3.69 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=6 ttl=53 time=2.59 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=7 ttl=53 time=2.62 ms
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=8 ttl=53 time=2.58 ms
```

Note - make sure in the security group of the resources enabled ALL TRAFFIC from ANYWHERE source.

10. Create Development VPC (cidr - 20.20.0.0/16)

vpc-0abe06384fe22a944 / Development VPC			
Details	Info		
VPC ID vpc-0abe06384fe22a944	State Available	DNS hostnames Disabled	DNS resolution Enabled
Tenancy Default	DHCP option set dopt-0224512f7f1a4689e	Main route table rtb-02eb749e78e6be817	Main network ACL acl-0b161a013239bdbc7
Default VPC No	IPv4 CIDR 20.20.0.0/16	IPv6 pool -	IPv6 CIDR (Network border group) -
Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups -	Owner ID 892543032022	

11. Create 2 Subnets in it

- web2 subnet [public] (cidr - 20.20.1.0/24)
- db2 subnet [private] (cidr - 20.20.2.0/24)

Subnets (2) [Info](#)

Find resources by attribute or tag

Subnet ID : subnet-060ac8d4297fb0950 X Subnet ID : subnet-0fec4a2147af560e X VPC : vpc-0abe06384fe22a944 X Clear filters

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR
<input type="checkbox"/>	web2 subnet	subnet-060ac8d4297fb0950	Available	vpc-0abe06384fe22a944 Development VPC	20.20.1.0/24
<input type="checkbox"/>	db2 subnet	subnet-0fec4a2147af560e	Available	vpc-0abe06384fe22a944 Development VPC	20.20.2.0/24

- Select web2 subnet > Go to Action > Edit Subnet setting > Enable Auto Assign IPv4.

[VPC](#) > [Subnets](#) > [subnet-060ac8d4297fb0950](#) > **Edit subnet settings**

Edit subnet settings [Info](#)

Subnet

Subnet ID	Name
subnet-060ac8d4297fb0950	web2 subnet

Auto-assign IP settings [Info](#)

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

☒ **Enable auto-assign public IPv4 address** [Info](#)

☐ **Enable auto-assign customer-owned IPv4 address** [Info](#)
 Option disabled because no customer owned pools found.

12. Create Internet Gateway (Dev-IGW) and attach it to Dev VPC.

[VPC](#) > [Internet gateways](#) > [igw-05e6b514452570462](#)

igw-05e6b514452570462 / Dev-IGW

Details [Info](#)

Internet gateway ID	State	VPC ID
igw-05e6b514452570462	Attached	vpc-0abe06384fe22a944 Development VPC

Tags

Search tags

Key	Value
Name	Dev-IGW

13. Create Public Route Table

- edit routes
- destination - 0.0.0.0/0
- target - internet gateway (Dev-IGW)
- subnet association - web2 subnet

VPC > Route tables > rtb-06a20df739af3a2b0

rtb-06a20df739af3a2b0 / PublicRouteDev

Details Info

Route table ID

rtb-06a20df739af3a2b0

Main

No

Explicit subnet associations

subnet-060ac8d4297fb0950 / web2 subnet

VPC

vpc-0abe06384fe22a944 | Development VPC

Owner ID

892543032022

Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Filter routes

Destination	Target	Status
0.0.0.0/0	igw-05e6b514452570462	Active
20.20.0.0/16	local	Active

14. Launch ec2 instances in all subnets of Development VPC and name them as per name of your subnets.

Instances (7) Info Refresh Connect

Find Instance by attribute or tag (case-sensitive) All states

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	db2 instance	i-0bb63e9ee1911d2da	Running	t2.micro	Initializing	View alarms	ap-south-1a
<input type="checkbox"/>	web2 instance	i-040450f419fa5c34c	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a

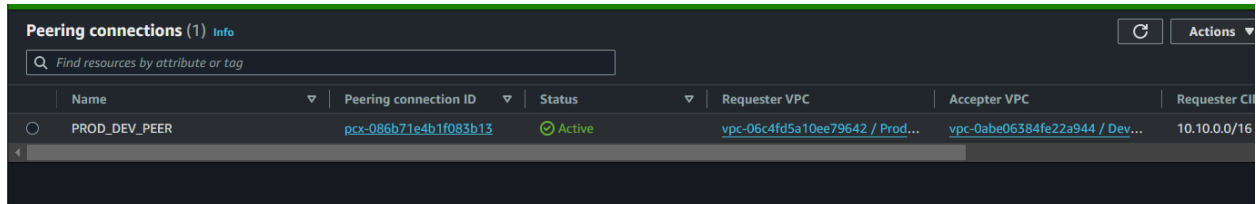
15. Connect to your web2 instance and check the internet connectivity (ping google.com)

```
ubuntu@ip-20-20-1-83:~$ sudo su
root@ip-20-20-1-83:/home/ubuntu# ping google.com
PING google.com (142.250.183.206) 56(84) bytes of data.
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=1 ttl=54 time=2.02 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=2 ttl=54 time=2.10 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=3 ttl=54 time=2.03 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=4 ttl=54 time=2.05 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=5 ttl=54 time=2.06 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=6 ttl=54 time=2.05 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=7 ttl=54 time=2.02 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=8 ttl=54 time=2.05 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=9 ttl=54 time=2.03 ms
64 bytes from bom07s33-in-f14.1e100.net (142.250.183.206): icmp_seq=10 ttl=54 time=2.04 ms
```

i-040450f419fa5c34c (web2 instance)
 PublicIPs: 13.201.40.232 PrivateIPs: 20.20.1.83

Note - make sure in the security group of the resources enabled ALL TRAFFIC from ANYWHERE source.

16. Create Peering connection (PROD_DEV_PEER) between Production VPC and Development VPC, also accept the peering request.



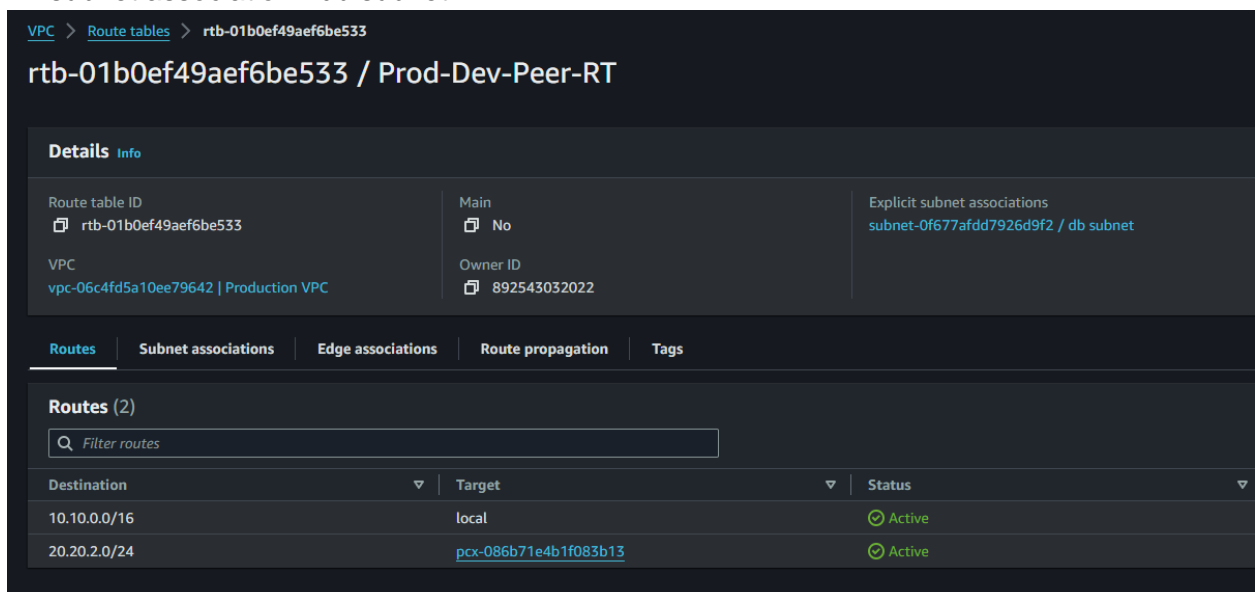
The screenshot shows the 'Peering connections (1)' page in the AWS console. A table lists the connection 'PROD_DEV_PEER' with ID 'pcx-086b71e4b1f083b13', status 'Active', requester VPC 'vpc-06c4fd5a10ee79642 / Prod...', and acceptor VPC 'vpc-0abe06384fe22a944 / Dev...'. The requester CIDR is '10.10.0.0/16'.

Name	Peering connection ID	Status	Requester VPC	Accepter VPC	Requester CIDR
PROD_DEV_PEER	pcx-086b71e4b1f083b13	Active	vpc-06c4fd5a10ee79642 / Prod...	vpc-0abe06384fe22a944 / Dev...	10.10.0.0/16

17. Create 2 Route Tables-

a. Prod-Dev-Peer-RT and attach to Production VPC.

- edit the routes
- destination - 20.20.2.0/24 [cidr of db2 subnet of dev network]
- target - peering connection (PROD_DEV_PEER)
- subnet association - db subnet



The screenshot shows the details for route table 'rtb-01b0ef49aef6be533' attached to 'vpc-06c4fd5a10ee79642 | Production VPC'. It lists two routes: a local route for '10.10.0.0/16' and a route for '20.20.2.0/24' targeting the peering connection 'pcx-086b71e4b1f083b13'. Both routes are 'Active'.

Destination	Target	Status
10.10.0.0/16	local	Active
20.20.2.0/24	pcx-086b71e4b1f083b13	Active

b. Dev-Prod-Peer-RT and attach to Development VPC.

- edit the routes
- destination - 10.10.4.0/24 [cidr of db subnet of prod network]
- target - peering connection (PROD_DEV_PEER)
- subnet association - db2 subnet

VPC > Route tables > rtb-03a4194dc87802c55

rtb-03a4194dc87802c55 / Dev-Prod-Peer-RT

Details Info

Route table ID rtb-03a4194dc87802c55	Main No	Explicit subnet associations subnet-0fec4a2147af560e / db2 subnet
VPC vpc-0abe06384fe22a944 Development VPC	Owner ID 892543032022	

Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Destination	Target	Status
10.10.4.0/24	pcx-086b71e4b1f083b13	Active
20.20.0.0/16	local	Active

18. Then connect to your db instance and then ping private ip of db2 instance (you will be able to see the network packet transfer)

```
ubuntu@ip-10-10-4-137:~$ ping 20.20.2.35
PING 20.20.2.35 (20.20.2.35) 56(84) bytes of data.
64 bytes from 20.20.2.35: icmp_seq=1 ttl=64 time=0.789 ms
64 bytes from 20.20.2.35: icmp_seq=2 ttl=64 time=0.380 ms
64 bytes from 20.20.2.35: icmp_seq=3 ttl=64 time=0.470 ms
64 bytes from 20.20.2.35: icmp_seq=4 ttl=64 time=0.400 ms
64 bytes from 20.20.2.35: icmp_seq=5 ttl=64 time=0.452 ms
64 bytes from 20.20.2.35: icmp_seq=6 ttl=64 time=0.415 ms
64 bytes from 20.20.2.35: icmp_seq=7 ttl=64 time=0.557 ms
64 bytes from 20.20.2.35: icmp_seq=8 ttl=64 time=0.468 ms
64 bytes from 20.20.2.35: icmp_seq=9 ttl=64 time=0.468 ms
```