

# **Project Summary: Multi-Account Network Integration with AWS Transit Gateway & RAM**

**In this lab, I designed and implemented a centralized multi-account network architecture across three AWS accounts using AWS Transit Gateway (TGW) and AWS Resource Access Manager (RAM).**

**A total of six VPCs (two in each account) were deployed with isolated networks, public/private subnets, route tables, and EC2 instances for connectivity validation.**

**A single TGW was created in the management account, and then securely shared with the other two accounts using AWS RAM. Each account attached its VPCs to this shared TGW, forming a hub-and-spoke architecture for cross-account communication.**

**Routing was configured at both TGW level (propagation) and VPC level (route tables), enabling seamless private network connectivity between all VPCs across accounts without any VPC peering or public internet exposure.**

## **Key Outcomes**

- Successfully built a multi-account, multi-VPC private network using AWS Transit Gateway.**
- Enabled secure cross-account connectivity using AWS Resource Access Manager (RAM).**

- **Established a full-mesh private network across:**
  - **3 AWS accounts**
  - **6 VPCs**
  - **6 EC2 instances**
- **Verified end-to-end connectivity via private IP between all instances.**
- **Demonstrated TGW features:**
  - **VPC attachments**
  - **Route table propagation**
  - **Cross-account sharing**
  - **Centralized hub-and-spoke routing**
- **Eliminated the complexity of VPC peering (no need for 15+ peering links).**
- **Gained hands-on experience designing scalable, enterprise-grade network architecture.**
- **Validated routing behavior, security groups, and isolation before/after TGW integration.**

**Services Used:**

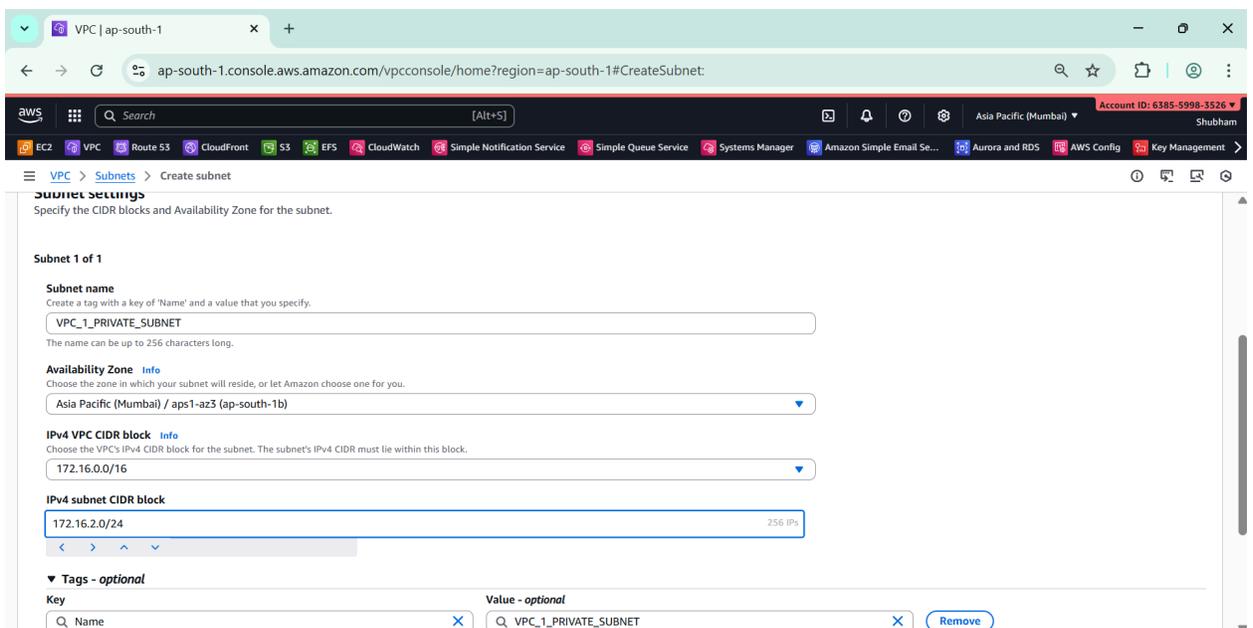
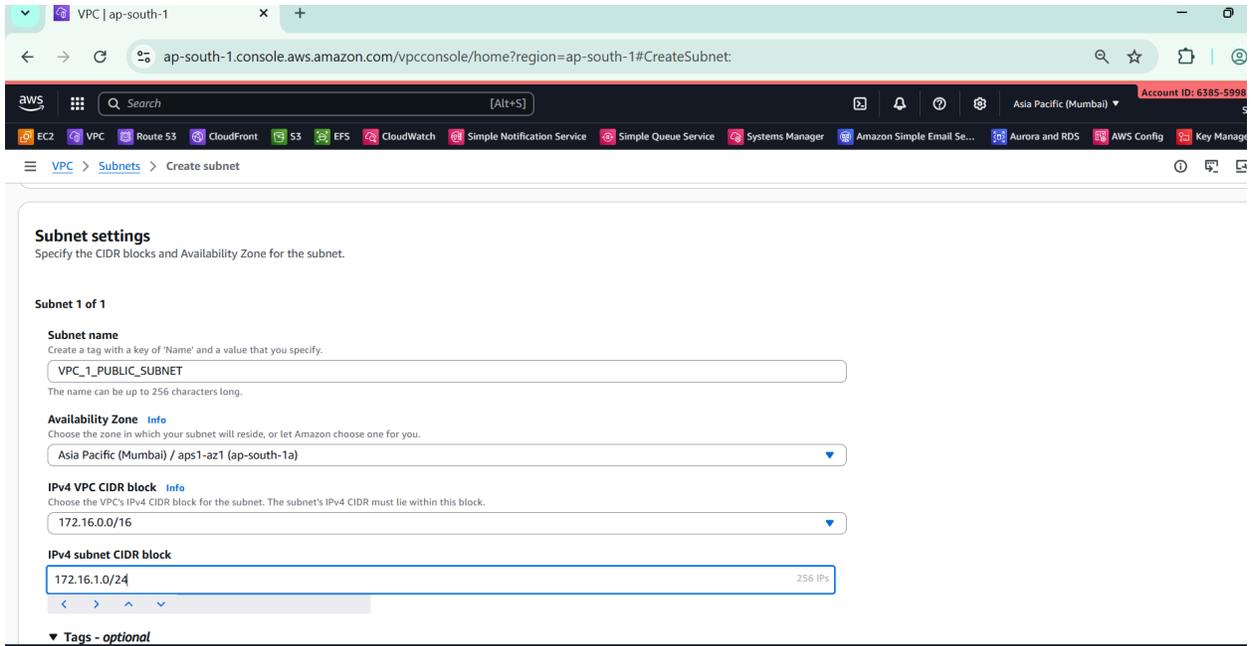
***VPC, Subnets, Route Tables, IGW, EC2, Security Groups, Transit Gateway, RAM, AWS Organizations, IAM.***

## Step 1) We will create Two Vpcs in Mumbai region with cidr 172.16.0.0/16 and 172.17.0.0/16

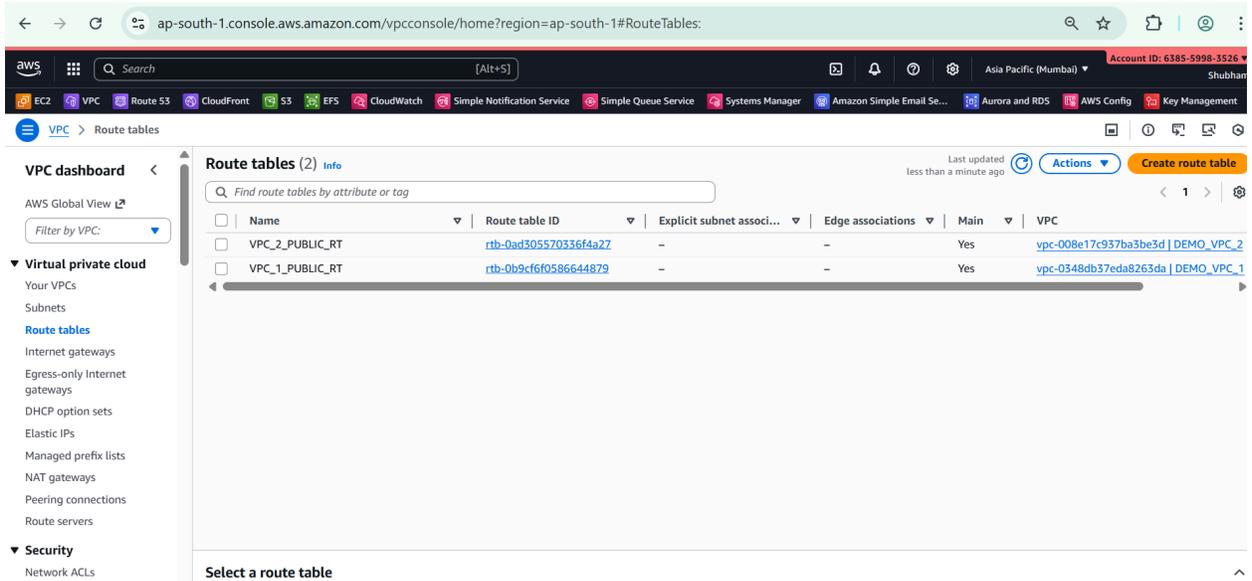
The screenshot shows the AWS console 'Create VPC' page for the first VPC. The browser address bar shows the URL: `ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#CreateVpc:createMode=vpcOnly`. The page title is 'Create VPC'. The 'Name tag - optional' field contains 'DEMO\_VPC\_1'. Under 'IPv4 CIDR block', the 'IPv4 CIDR manual input' radio button is selected, and the 'IPv4 CIDR' field contains '172.16.0.0/16'. Under 'IPv6 CIDR block', the 'No IPv6 CIDR block' radio button is selected. The 'Tenancy' dropdown is set to 'Default'. Under 'VPC encryption control (\$)', the 'None' radio button is selected.

The screenshot shows the AWS console 'Create VPC' page for the second VPC. The browser address bar shows the URL: `ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#CreateVpc:createMode=vpcOnly`. The page title is 'Create VPC'. The 'Resources to create' section has 'VPC only' selected. The 'Name tag - optional' field contains 'DEMO\_VPC\_2'. Under 'IPv4 CIDR block', the 'IPv4 CIDR manual input' radio button is selected, and the 'IPv4 CIDR' field contains '172.17.0.0/16'. Under 'IPv6 CIDR block', the 'No IPv6 CIDR block' radio button is selected. The 'Tenancy' dropdown is set to 'Default'.

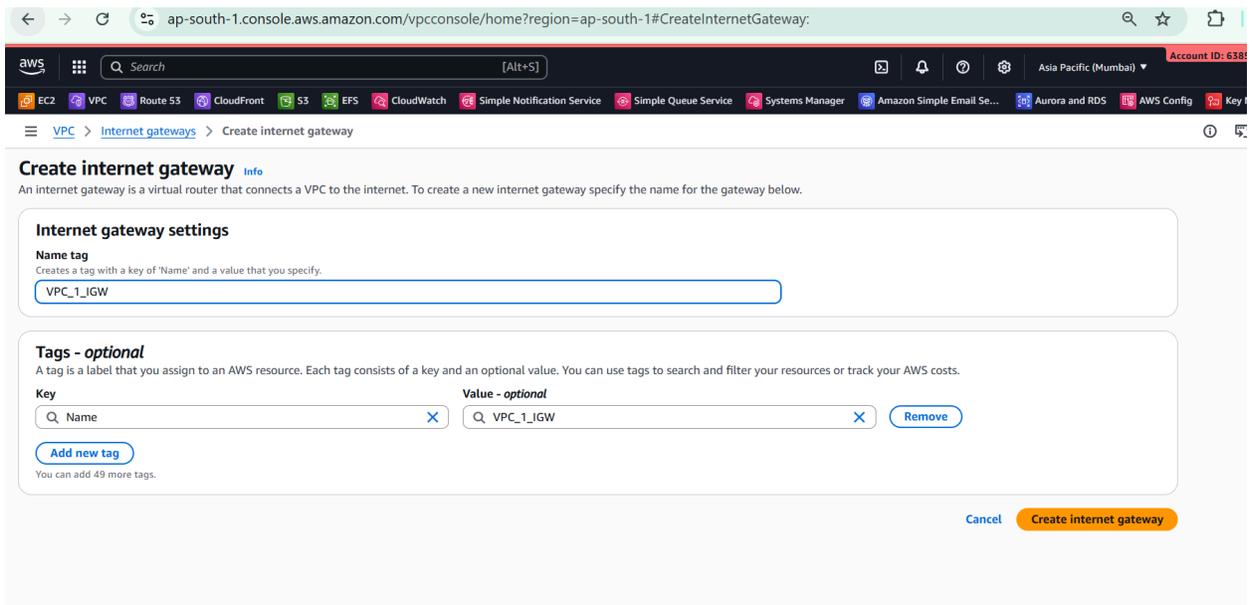
## Step 2) We will create 1 public and 1 private subnets in Vpc 1 with cidr block as 172.16.1.0/24 and 172.16.2.0/24.



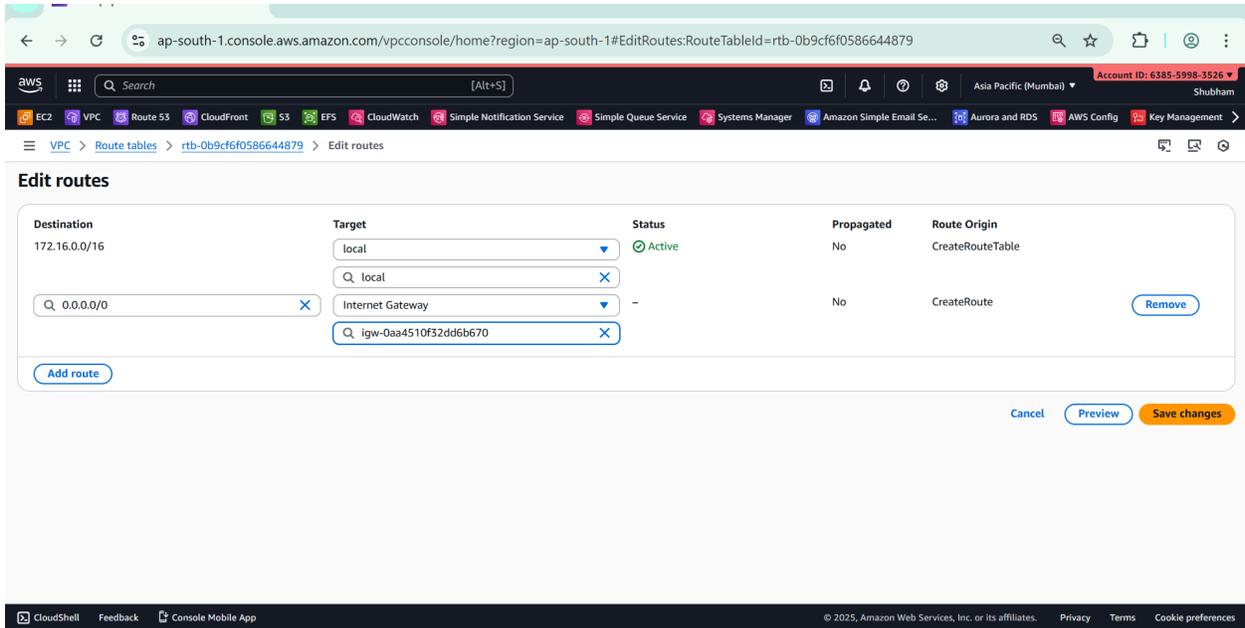
**Step 3) We can see public route tables for both vpc 1 and vpc 2.**



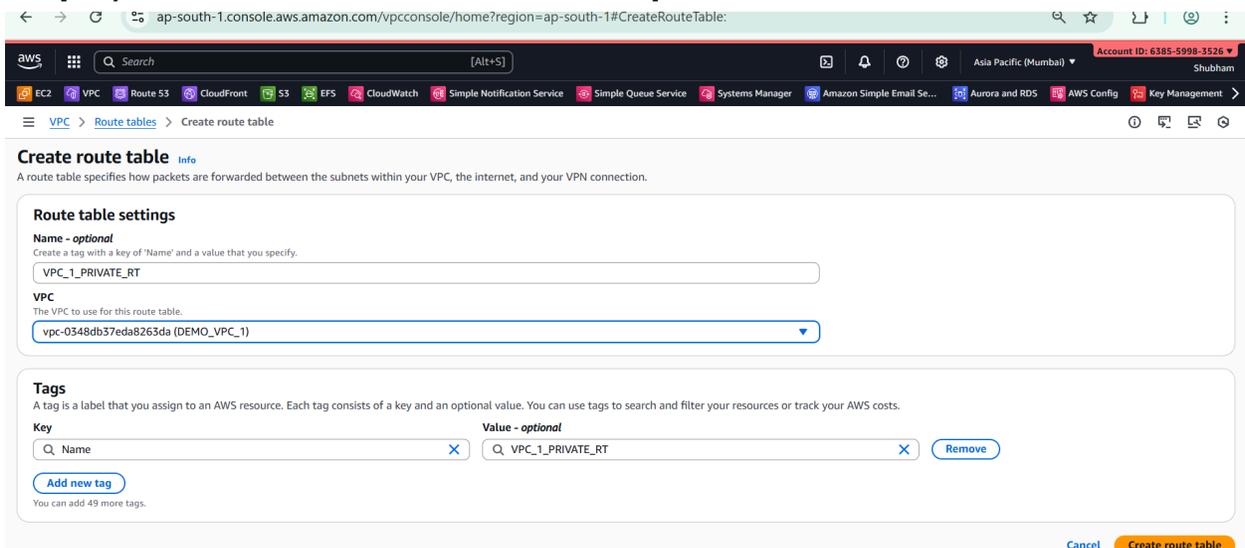
**Step 4) We will create internet gateway and attach it to Vpc1.**



**Step 5) We will add route for internet in vpc1 public route table**

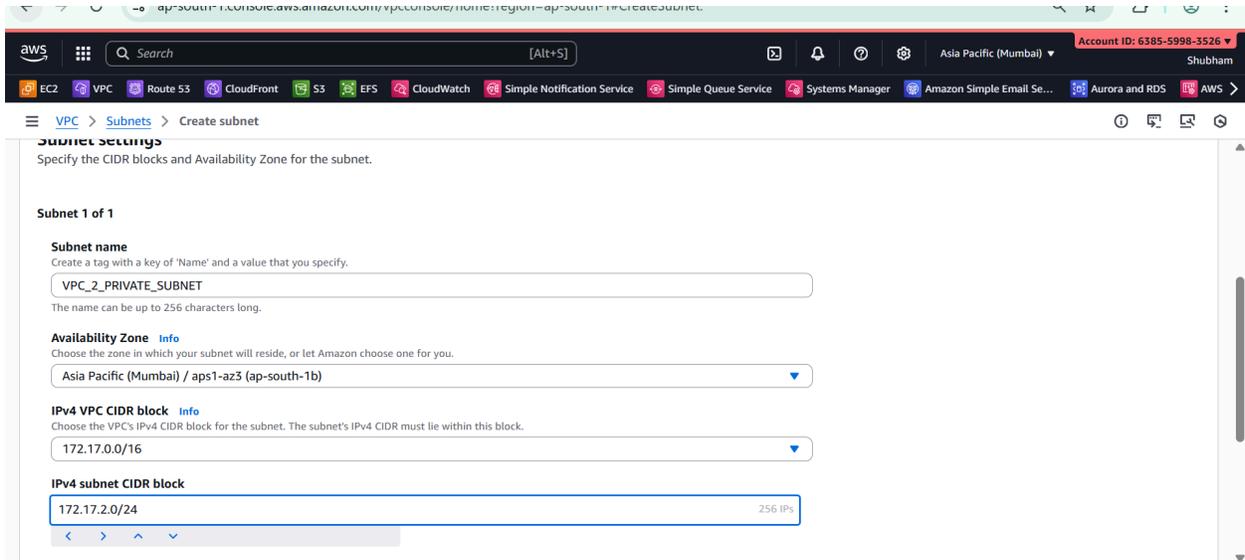


## Step 6) We will create route table for private subnet.

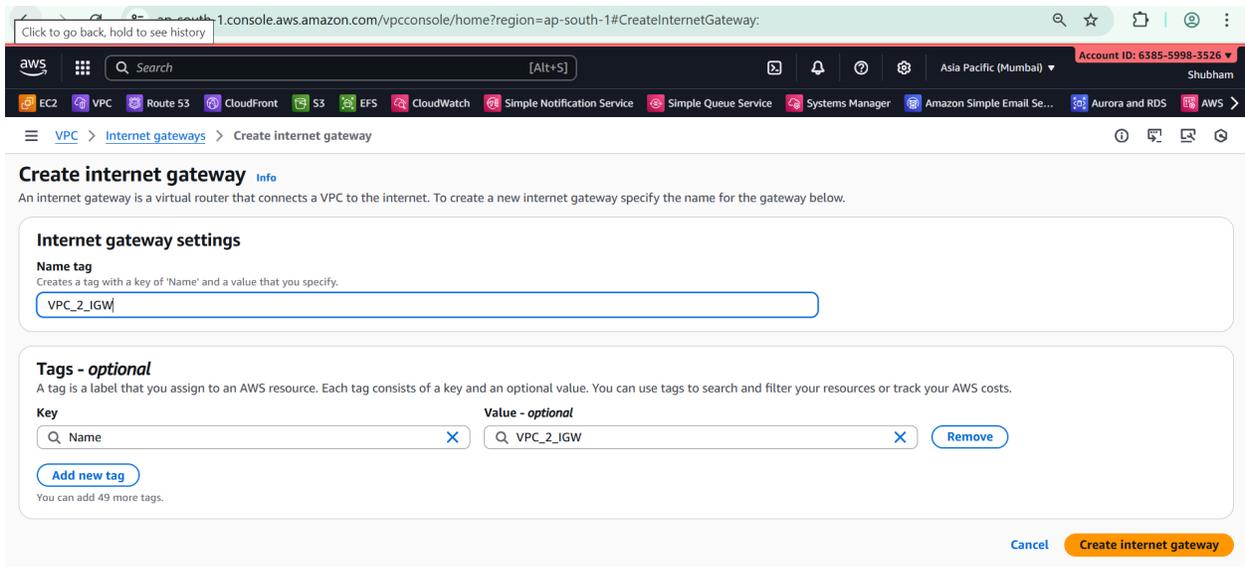


## Step 7) We can see all resources for vpc1 has been configured below.

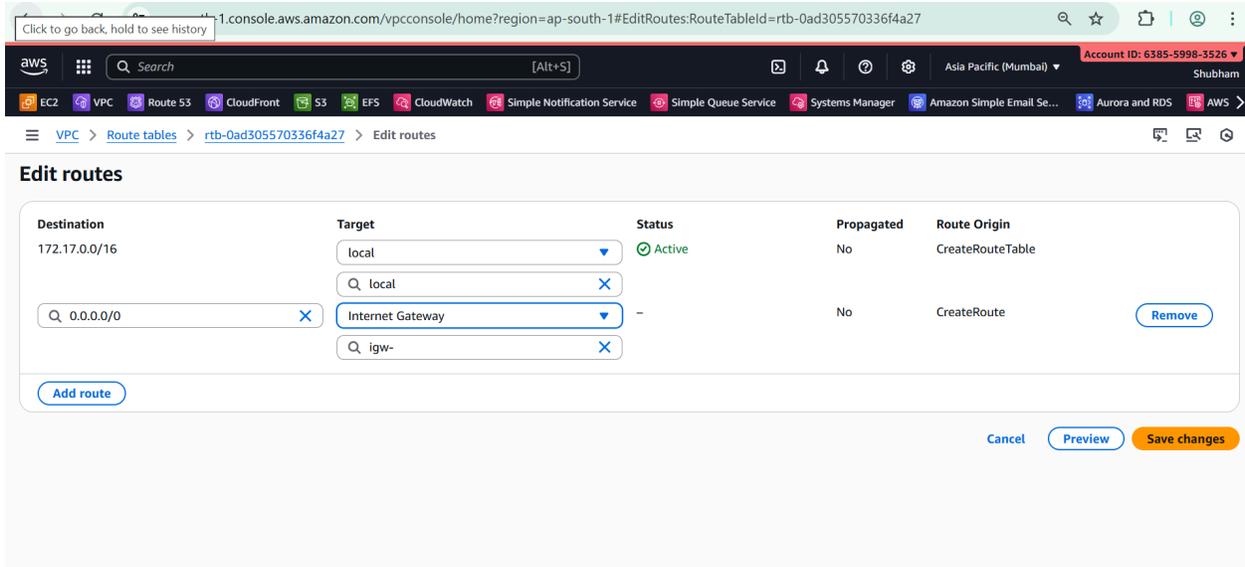
**Step 8) Similarly we will create subnets for vpc2 now with cidr block as 172.17.1.0/16 and 172.17.2.0/16**



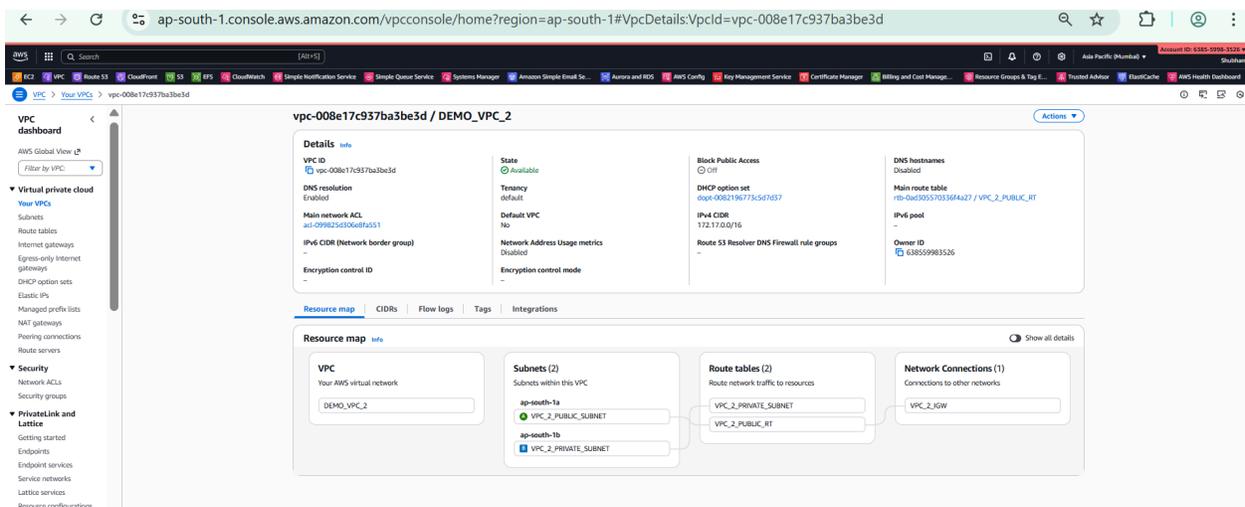
**Step 9) We will create again Internet gateway and attach it to vpc2.**



**Step 10) We will add this internet route in public route table and we will create private route table as well.**



**Step 11) We can confirm again all resources for vpc2 has been setup correctly.**



**Step 12) Similarly we will create two more vpcs in account 2.**

The screenshot shows the AWS VPC console for VPC\_1 (vpc-00af422bf693dfec9). The interface includes a left-hand navigation menu with categories like Virtual private cloud, Security, and PrivateLink and Lattice. The main content area is divided into several sections:

- Details:** Shows VPC ID, State (Available), DNS resolution (Enabled), Main network ACL, IPv4 CIDR (192.168.0.0/16), and Encryption control ID.
- State:** Shows Tenancy (default), Default VPC (No), Network Address Usage metrics (Disabled), and Encryption control mode.
- Block Public Access:** Shows as OFF.
- DNS hostnames:** Shows as Disabled.
- DHCP option set:** Shows as default (dopt-058b70b7b9994:329).
- Main route table:** Shows as rtb-0c65017f70965af7f / VPC\_1\_PUBLIC\_RT.
- IPv6 pool:** Shows as none.
- Owner ID:** Shows as 390503781838.

Below the details is a **Resource map** section with tabs for Resource map, CIDRs, Flow logs, Tags, and Integrations. The Resource map shows a diagram of the VPC structure:

- VPC:** vpc-00af422bf693dfec9
- Subnets (2):** ap-south-1a (VPC\_1\_PUBLIC\_SUBNET, VPC\_1\_PRIVATE\_SUBNET)
- Route tables (2):** VPC\_1\_PRIVATE\_RT, VPC\_1\_PUBLIC\_RT
- Network Connections (1):** IGW\_VPC\_1

The screenshot shows the AWS VPC console for VPC\_2 (vpc-004f3898c528641f9). The interface is similar to the first screenshot, showing details and a resource map for VPC\_2:

- Details:** Shows VPC ID, State (Available), DNS resolution (Enabled), Main network ACL, IPv4 CIDR (192.169.0.0/16), and Encryption control ID.
- State:** Shows Tenancy (default), Default VPC (No), Network Address Usage metrics (Disabled), and Encryption control mode.
- Block Public Access:** Shows as OFF.
- DNS hostnames:** Shows as Disabled.
- DHCP option set:** Shows as default (dopt-058b70b7b9994:329).
- Main route table:** Shows as rtb-01f5fb527d5515676 / VPC\_2\_PUBLIC\_RT.
- IPv6 pool:** Shows as none.
- Owner ID:** Shows as 390503781838.

The **Resource map** section shows the following structure:

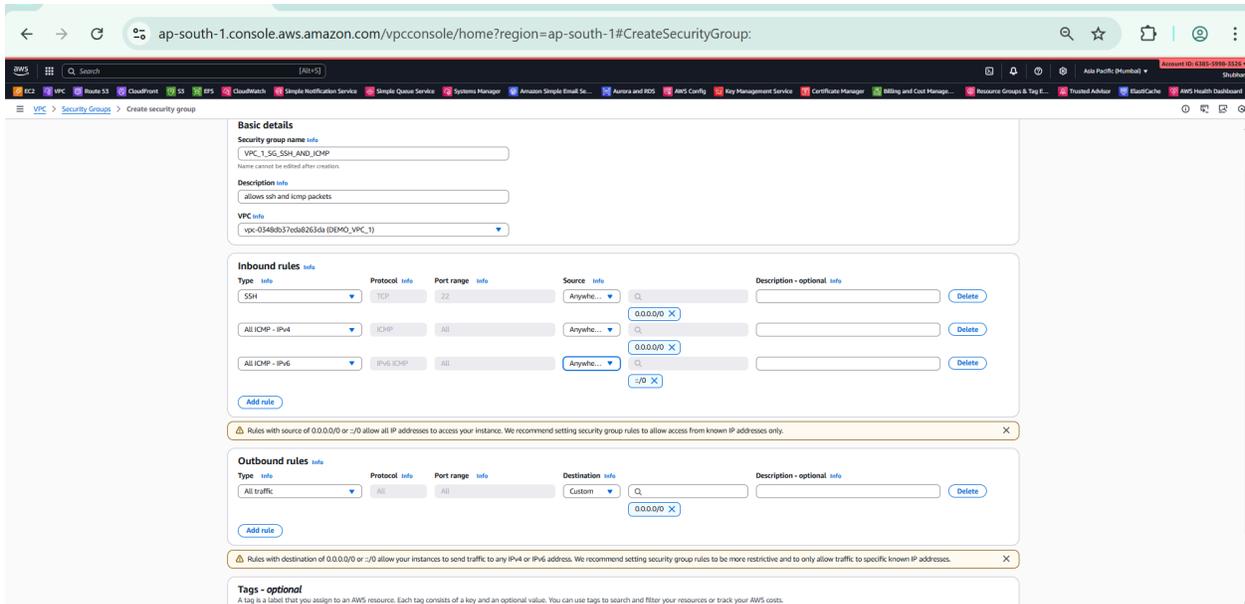
- VPC:** vpc-004f3898c528641f9
- Subnets (2):** ap-south-1a (VPC\_2\_PUBLIC\_SUBNET, VPC\_2\_PRIVATE\_SUBNET), ap-south-1b (VPC\_2\_PRIVATE\_SUBNET)
- Route tables (2):** VPC\_2\_PRIVATE\_RT, VPC\_2\_PUBLIC\_RT
- Network Connections (1):** IGW\_VPC\_2

**Step 13) Similarly we will create another two vpc in Account 3.**

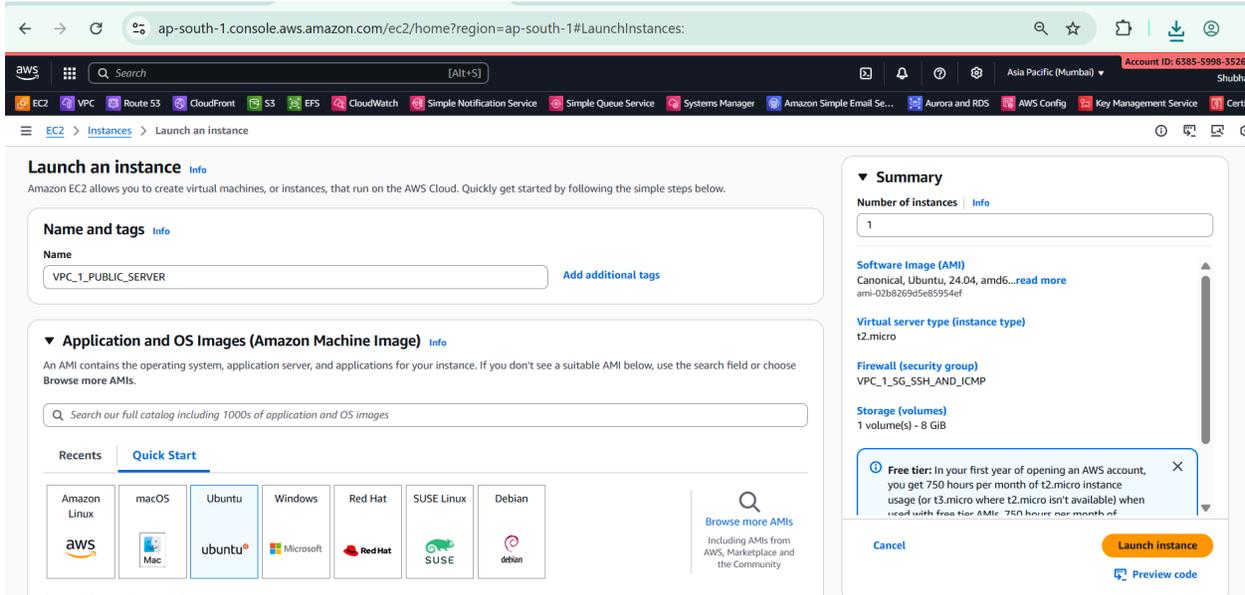
The screenshot shows the AWS VPC console for VPC\_1 (vpc-0ba6f64147b03782a). The interface includes a left-hand navigation menu with categories like Virtual private cloud, Security, and PrivateLink and Lattice. The main content area is divided into a 'Details' section and a 'Resource map' section. The 'Details' section provides information such as VPC ID, State (Available), DNS resolution (Enabled), Main network ACL, IPv6 CIDR, Encryption control ID, Block Public Access (Off), DHCP option set, IPv4 CIDR, Route 53 Resolver DNS Firewall rule groups, DNS hostnames (Disabled), Main route table, IPv6 pool, and Owner ID. The 'Resource map' section shows a diagram of the VPC resources, including two subnets (PUBLIC\_SUBNET\_VPC\_1 and PRIVATE\_SUBNET\_VPC\_1), two route tables (VPC\_1\_PUBLIC\_RT and VPC\_1\_PRIVATE\_RT), and one network connection (IGW\_VPC\_1).

The screenshot shows the AWS VPC console for VPC\_2 (vpc-075603526ae60ba2d). The interface is similar to the first screenshot, showing the 'Details' and 'Resource map' sections. The 'Details' section for VPC\_2 includes VPC ID, State (Available), DNS resolution (Enabled), Main network ACL, IPv6 CIDR, Encryption control ID, Block Public Access (Off), DHCP option set, IPv4 CIDR, Route 53 Resolver DNS Firewall rule groups, DNS hostnames (Disabled), Main route table, IPv6 pool, and Owner ID. The 'Resource map' section shows a diagram of the VPC resources, including two subnets (PUBLIC\_SUBNET\_VPC\_2 and PRIVATE\_SUBNET\_VPC\_2), two route tables (VPC\_2\_PRIVATE\_RT and VPC\_2\_PUBLIC\_RT), and one network connection (IGW\_VPC\_2).

**Step 14) We will create now Security group in Account A allowing ssh and icmp packers from anywhere to check our connectivity later. Also we will create similar Sg in Account 2 and Account 3 as well.**



**Step 15) Now we will launch and ec2 server in Account 1 and attach this security group created .**



Ubuntu Server 24.04 LTS (HVM), SSD Volume Type Free tier eligible

ami-02b8269d5e85954ef (64-bit (x86)) / ami-027308df79a86d22c (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

**Description**  
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture	AMI ID	Publish Date	Username	Verified provider
64-bit (x86)	ami-02b8269d5e85954ef	2025-10-22	ubuntu	<span style="color: green;">Verified provider</span>

**Instance type** [Info](#) [Get advice](#)

**Instance type**  
t2.micro  
Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand Windows base pricing: 0.017 USD per Hour Free tier eligible  
On-Demand RHEL base pricing: 0.0268 USD per Hour On-Demand Linux base pricing: 0.0124 USD per Hour All generations  
On-Demand Ubuntu Pro base pricing: 0.0142 USD per Hour On-Demand SUSE base pricing: 0.0124 USD per Hour [Compare instance types](#)

**Key pair (login)** [Info](#)  
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

**Summary**  
Number of instances [Info](#)  
1

**Firewall (security group)**  
VPC\_1\_SG\_SSH\_AND\_ICMP

**Storage (volumes)**  
1 volume(s) - 8 GiB

**Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

[Cancel](#) [Launch instance](#) [Preview code](#)

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Account ID: 6385-5998-3526 Shubham

**Network settings** [Info](#)

**VPC - required** [Info](#)  
vpc-0348db37eda8263da (DEMO\_VPC\_1)  
172.16.0.0/16

**Subnet** [Info](#)  
subnet-03243dad05d4ef2db VPC\_1\_PUBLIC\_SUBNET  
VPC: vpc-0348db37eda8263da Owner: 638559983526 Availability Zone: ap-south-1a (aps-1-az-1)  
Zone type: Availability Zone IP addresses available: 251 CIDR: 172.16.1.0/24 [Create new subnet](#)

**Auto-assign public IP** [Info](#)  
Enable  
Additional charges apply when outside of free tier allowance

**Firewall (security groups)** [Info](#)  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.  
 Create security group  Select existing security group

**Common security groups** [Info](#)  
Select security groups  
VPC\_1\_SG\_SSH\_AND\_ICMP sg-021f94e488e89977d [Compare security group rules](#)  
VPC: vpc-0348db37eda8263da

**Summary**  
Number of instances [Info](#)  
1

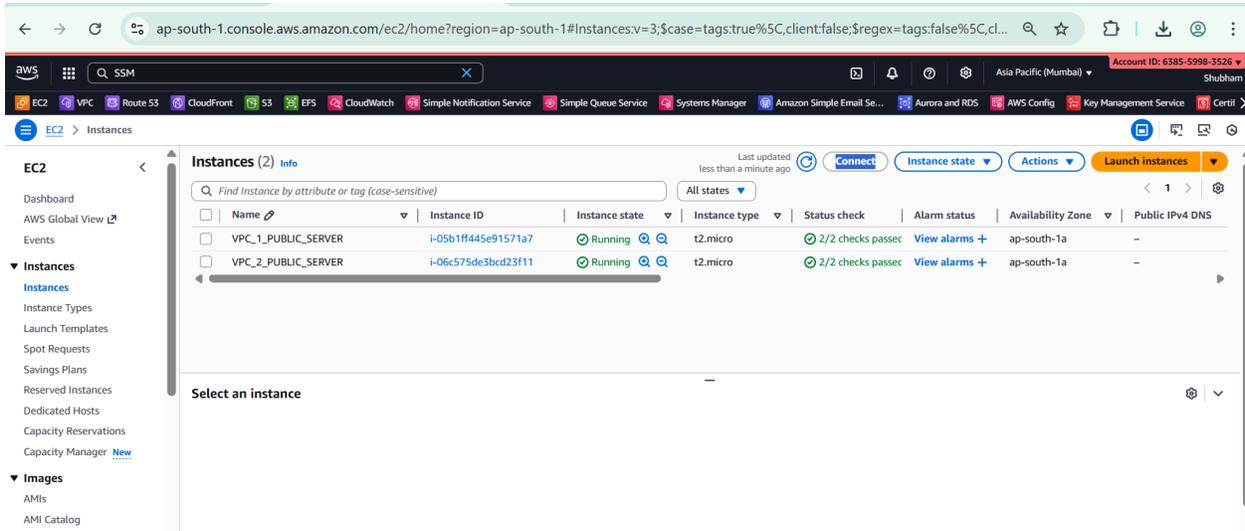
**Firewall (security group)**  
VPC\_1\_SG\_SSH\_AND\_ICMP

**Storage (volumes)**  
1 volume(s) - 8 GiB

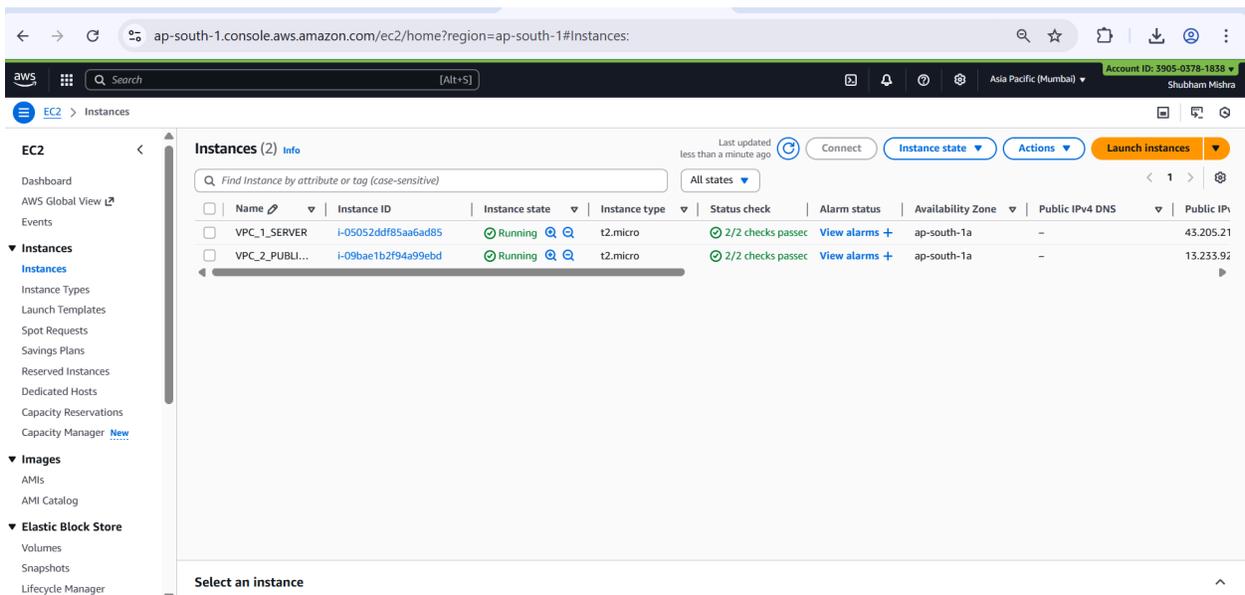
**Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

[Cancel](#) [Launch instance](#) [Preview code](#)

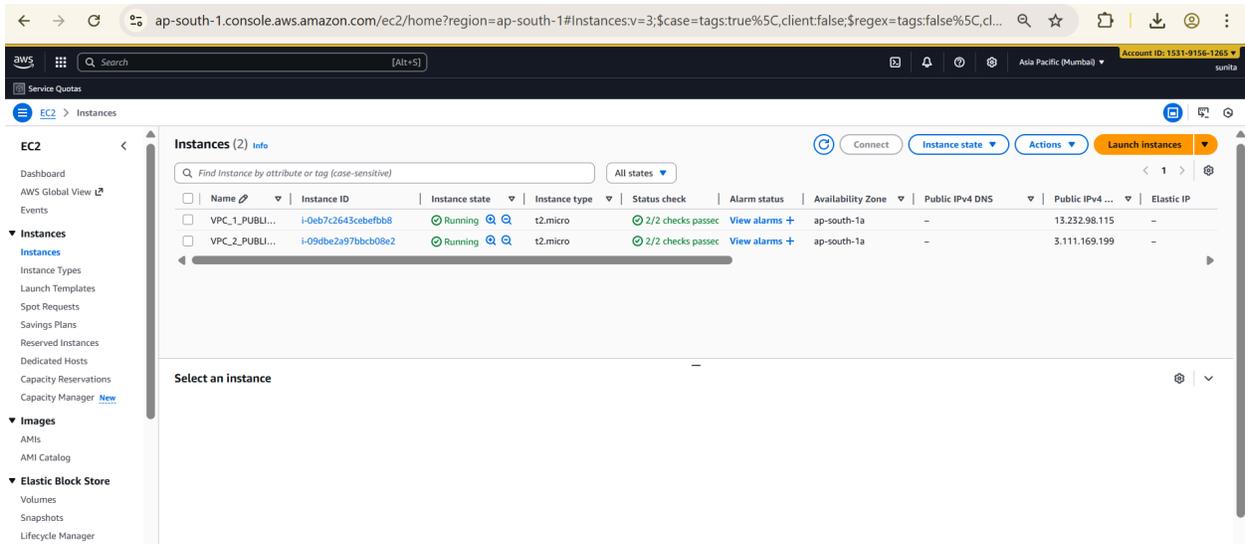
**Step 16) We launched similar ec2 in another vpc of Account 1. We can confirm below now both ec2 instances in both vpcs in Account 1 are running successfully.**



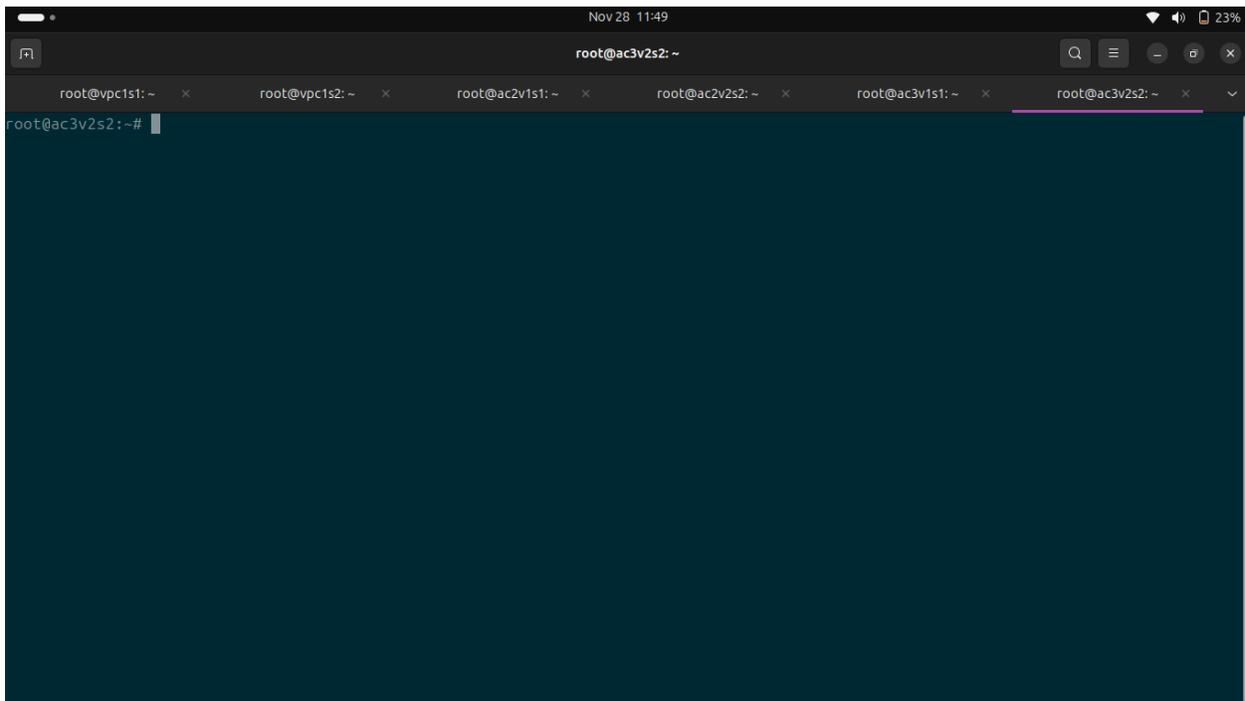
**Step 17) Similarly we will launch two instances in to two different vpc in Account 2 and we can see it's successful.**



**Step 18) Similarly we launched another two instances in to 2 different vpcs in Account 3 as well.**



**Step 19) We will ssh into all 6 instances of different vpcs across three different account.**



**Step 20) We will try to ping all ec2 instances across different vpcs across different account and it got failed as can be seen below.**

```
root@vpc1s1:~# ping 172.17.1.231
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
^C
--- 172.17.1.231 ping statistics ---
17 packets transmitted, 0 received, 100% packet loss, time 16398ms

root@vpc1s1:~# ping 192.169.0.95
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
^C
--- 192.169.0.95 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1044ms

root@vpc1s1:~# ping 192.168.0.242
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
^C
--- 192.168.0.242 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4125ms

root@vpc1s1:~# ping 10.0.0.92
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
^C
--- 10.0.0.92 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1052ms

root@vpc1s1:~# ping 10.1.0.43
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
^C
--- 10.1.0.43 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4075ms

root@vpc1s1:~# S
```

**Step 21) Now we will create Transit gateway in Account 1.**

**tgw-0ae31632c154c8fc6 / TGW-ACCOUNT\_1\_MAIN\_ACCOUNT**

Property	Value
<b>Transit gateway ID</b>	tgw-0ae31632c154c8fc6
<b>State</b>	Available
<b>Amazon ASN</b>	420000000
<b>DNS support</b>	Enable
<b>Transit gateway ARN</b>	arn:aws:ec2:ap-south-1:638559983526:transit-gateway/tgw-0ae31632c154c8fc6
<b>Default association route table</b>	Enable
<b>Association route table ID</b>	tgw-rtb-063bed3fbc7105826
<b>Auto accept shared attachments</b>	Disable
<b>Owner ID</b>	638559983526
<b>Default propagation route table</b>	Enable
<b>Propagation route table ID</b>	tgw-rtb-063bed3fbc7105826
<b>VPN ECMP support</b>	Enable
<b>Description</b>	this tgw will be shared to other aws accounts using ram
<b>Transit gateway CIDR blocks</b>	-
<b>Security Group Referencing support</b>	Disable
<b>Metering policy ID</b>	-
<b>Multicast support</b>	Disable
<b>Encryption support</b>	Disabled
<b>Encryption state message</b>	-

**Step 22) Now we will create Transit gateway attachment for both vpc in Account A.**

**Create transit gateway attachment**

A transit gateway (TGW) is a network transit hub that interconnects attachments (VPCs and VPNs) within the same AWS account or across AWS accounts.

**Details**

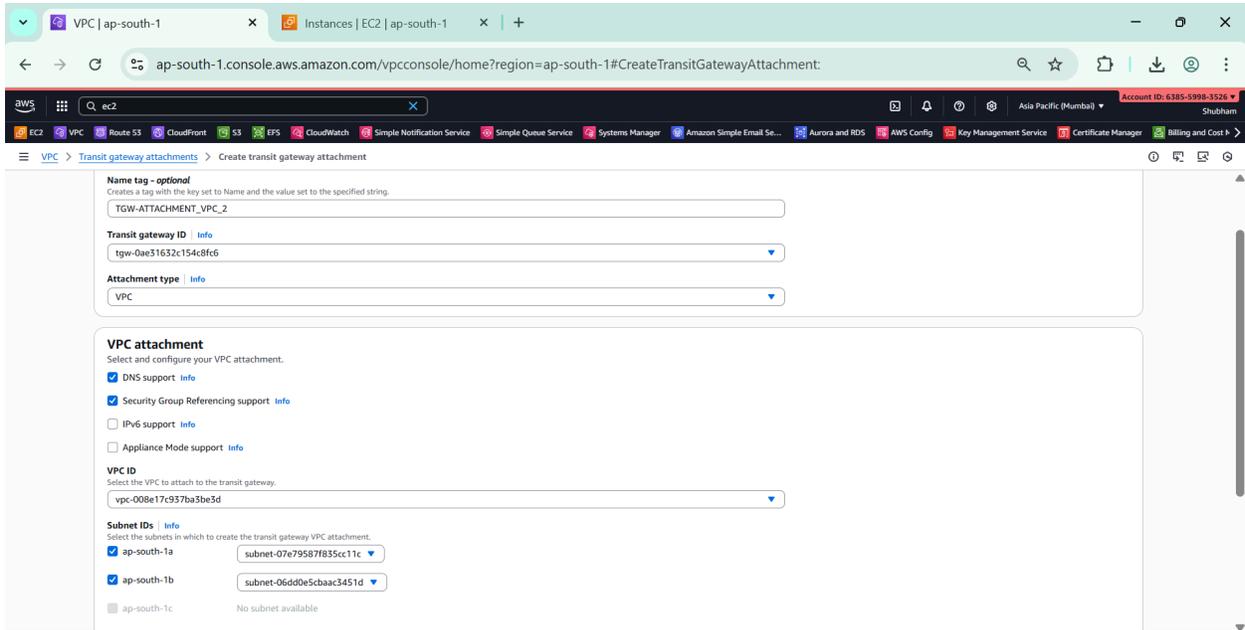
- Name tag - optional**  
Creates a tag with the key set to Name and the value set to the specified string.  
TGW-ATTACHMENT-VPC1
- Transit gateway ID**  
tgw-0ae31632c154c8fc6
- Attachment type**  
VPC

**VPC attachment**  
Select and configure your VPC attachment.

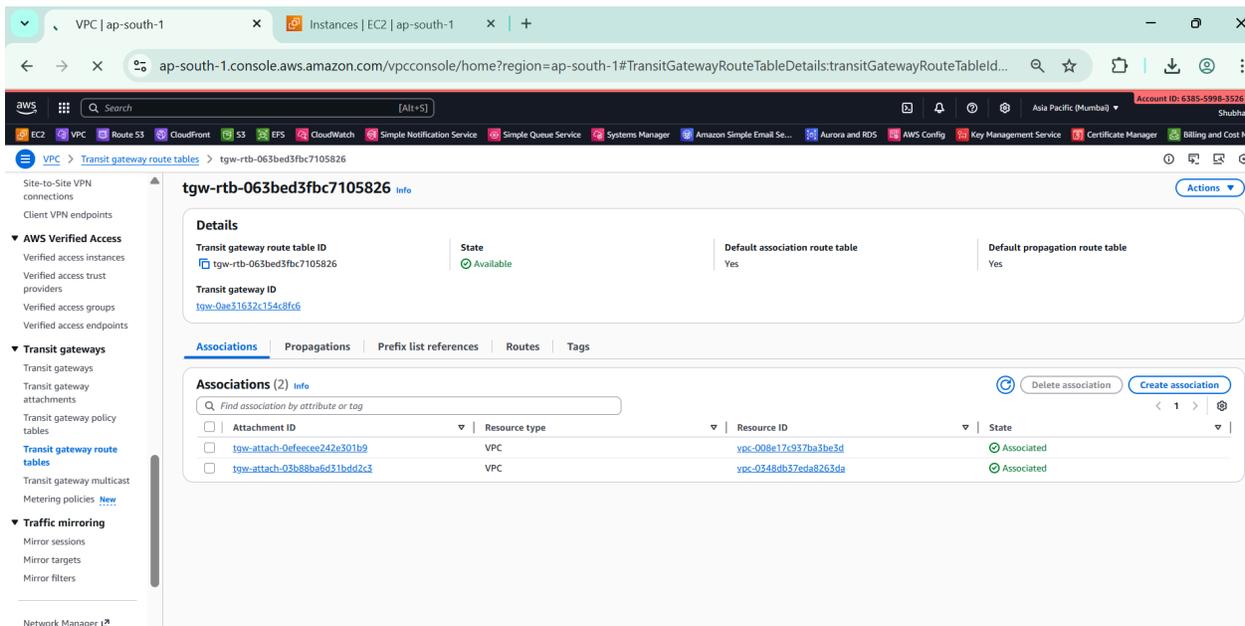
- DNS support**
- Security Group Referencing support**
- IPv6 support**
- Appliance Mode support**

**VPC ID**  
Select the VPC to attach to the transit gateway.  
vpc-0348db57eda8265da

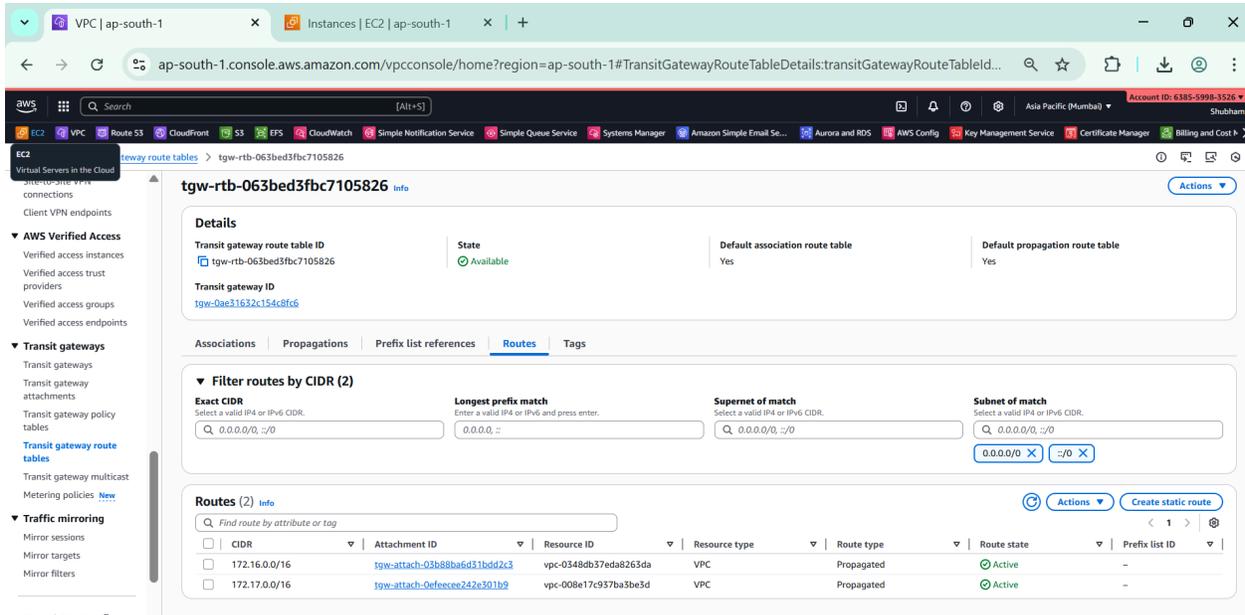
**Subnet IDs**  
Select the subnets in which to create the transit gateway VPC attachment.  
 ap-south-1a  
subnet-03243dad05d4ef2db



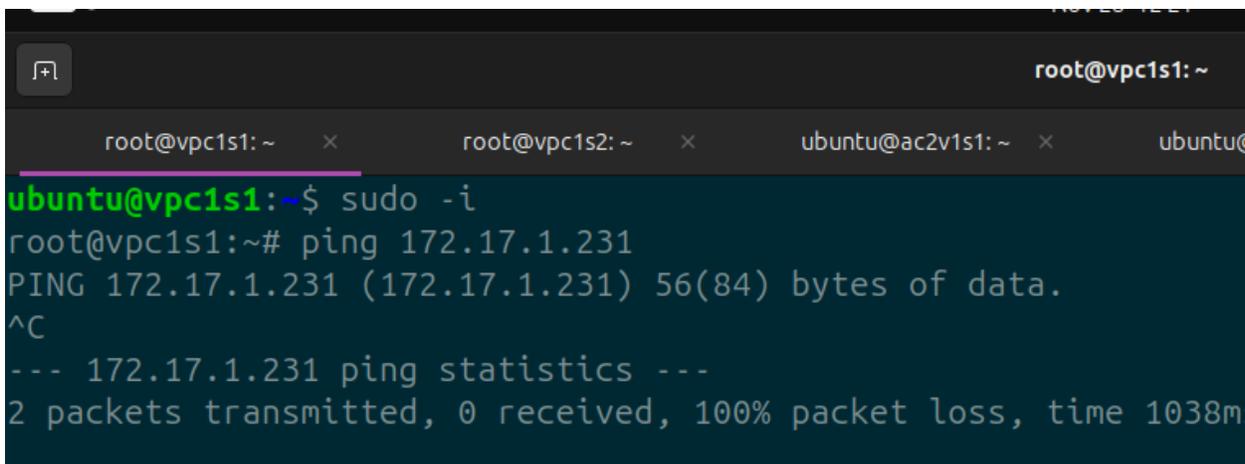
**Step 23) Now we can see as soon as TGW attachments are made it gets attached to TGW.**



**Step 24) Also routes gets added automatically fo both vpc once attachment is successful.**



**Step 25) Now still both servers in different vpcs in Account 1 cannot ping.**



**Step 26) Now we will update route table and add routes in both vpc's route table in Account 1.**

**rtb-0b9cf6f0586644879 / VPC\_1\_PUBLIC\_RT**

**Details**

- Route table ID: rtb-0b9cf6f0586644879
- Main: Yes
- Explicit subnet associations: subnet-03243dad05d4ef2db / VPC\_1\_PUBLIC\_SUBNET
- Edge associations: -
- VPC: vpc-0348db37eda8263da | DEMO\_VPC\_1
- Owner ID: 638559983526

**Routes (3)**

Destination	Target	Status	Propagated	Route Origin
0.0.0/0	igw-0aa4510f32dd6b670	Active	No	Create Route
172.16.0.0/16	local	Active	No	Create Route Table
172.17.0.0/16	tgw-0ae31632c154c8f66	Active	No	Create Route

**rtb-0ad305570336f4a27 / VPC\_2\_PUBLIC\_RT**

**Details**

- Route table ID: rtb-0ad305570336f4a27
- Main: Yes
- Explicit subnet associations: subnet-07e795587f935cc11c / VPC\_2\_PUBLIC\_SUBNET
- Edge associations: -
- VPC: vpc-008e17e937ba3be3d | DEMO\_VPC\_2
- Owner ID: 638559983526

**Routes (3)**

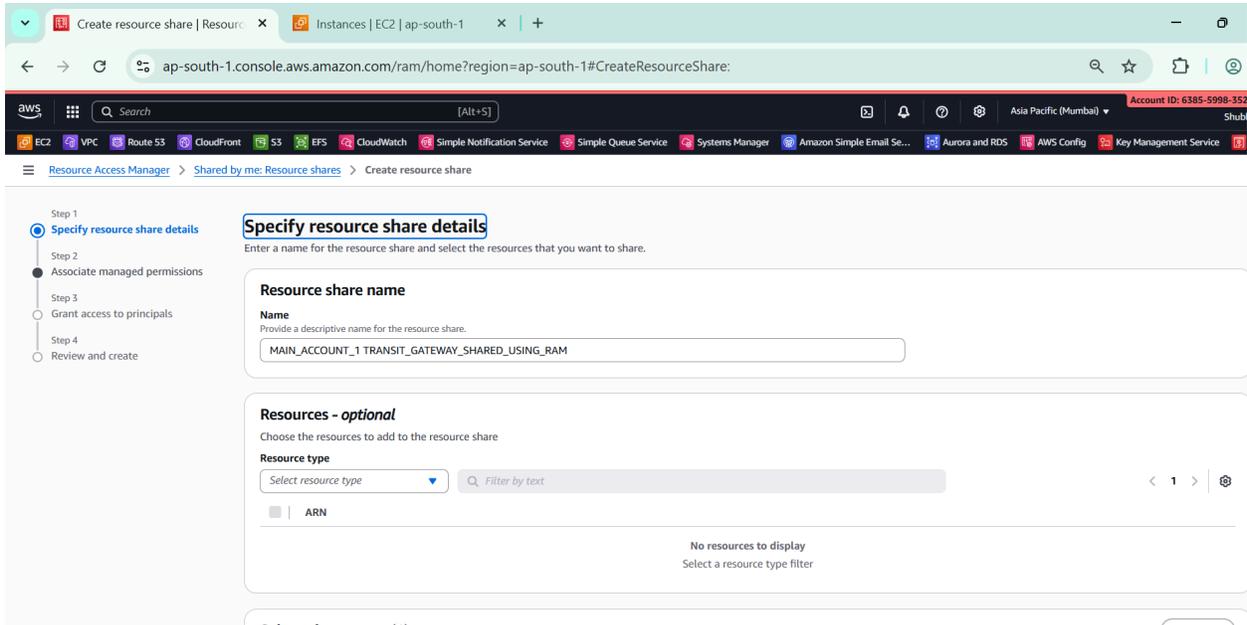
Destination	Target	Status	Propagated	Route Origin
0.0.0/0	igw-002eb6baa8d8d82ac	Active	No	Create Route
172.16.0.0/16	tgw-0ae31632c154c8f66	Active	No	Create Route
172.17.0.0/16	local	Active	No	Create Route Table

**Step 27) After successfully updating route table we can see that both servers in different vpc across account 1 can ping on private ip.**

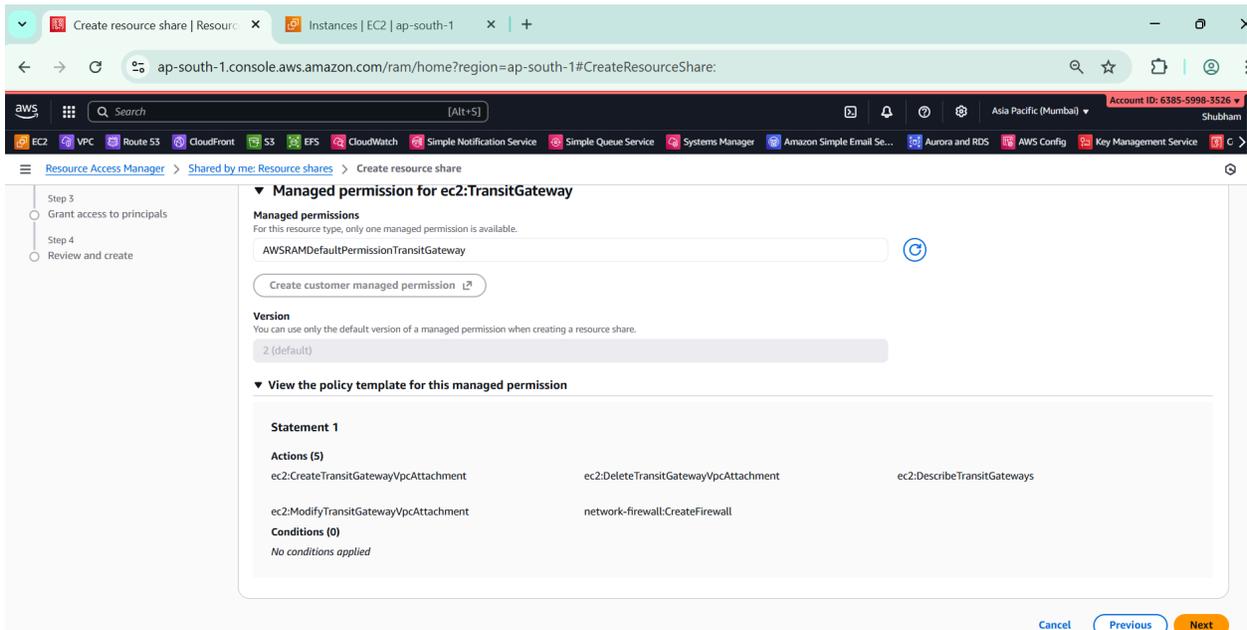
```
root@vpc1s1: ~  
root@vpc1s1:~# ping 172.17.1.231  
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.  
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=2.11 ms  
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=1.15 ms  
64 bytes from 172.17.1.231: icmp_seq=3 ttl=63 time=1.13 ms  
64 bytes from 172.17.1.231: icmp_seq=4 ttl=63 time=0.659 ms  
^C  
--- 172.17.1.231 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3004ms  
rtt min/avg/max/mdev = 0.659/1.263/2.108/0.526 ms  
root@vpc1s1:~#
```

```
root@vpc1s2: ~  
root@vpc1s2:~# ping 172.16.1.64  
PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.  
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=0.852 ms  
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=1.00 ms  
64 bytes from 172.16.1.64: icmp_seq=3 ttl=63 time=0.423 ms  
64 bytes from 172.16.1.64: icmp_seq=4 ttl=63 time=1.06 ms  
64 bytes from 172.16.1.64: icmp_seq=5 ttl=63 time=0.895 ms  
64 bytes from 172.16.1.64: icmp_seq=6 ttl=63 time=0.793 ms  
64 bytes from 172.16.1.64: icmp_seq=7 ttl=63 time=0.417 ms  
64 bytes from 172.16.1.64: icmp_seq=8 ttl=63 time=0.828 ms  
64 bytes from 172.16.1.64: icmp_seq=9 ttl=63 time=1.02 ms  
64 bytes from 172.16.1.64: icmp_seq=10 ttl=63 time=0.771 ms  
64 bytes from 172.16.1.64: icmp_seq=11 ttl=63 time=1.28 ms  
64 bytes from 172.16.1.64: icmp_seq=12 ttl=63 time=0.573 ms  
64 bytes from 172.16.1.64: icmp_seq=13 ttl=63 time=0.742 ms
```

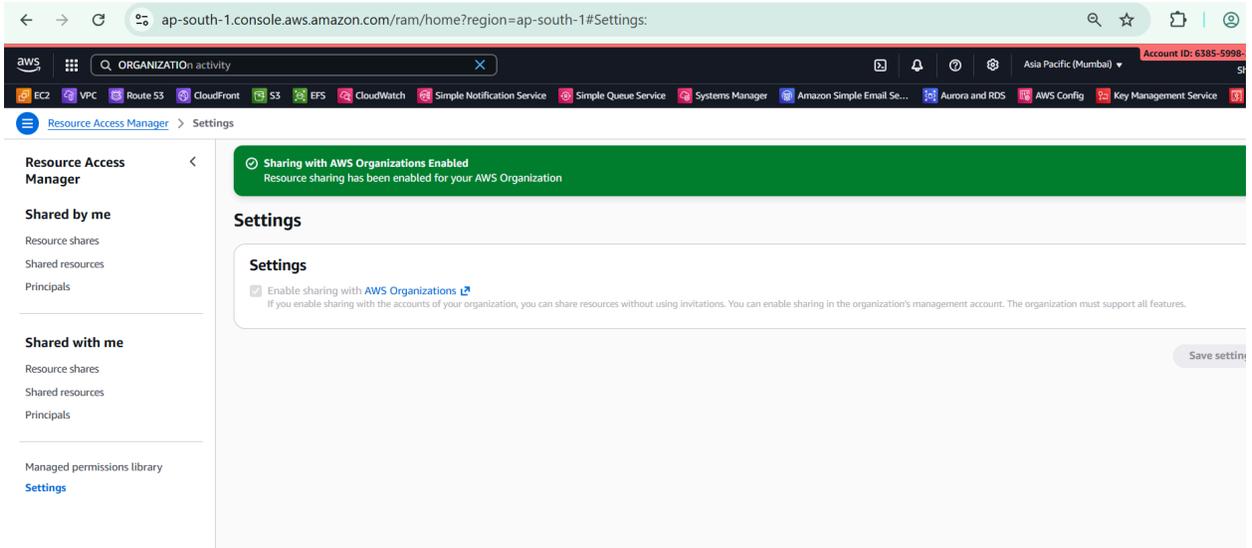
**Step 28) Now we will share the TGW created in Account 1 with Account 2 and Account 3 using RAM service in Account 1.**



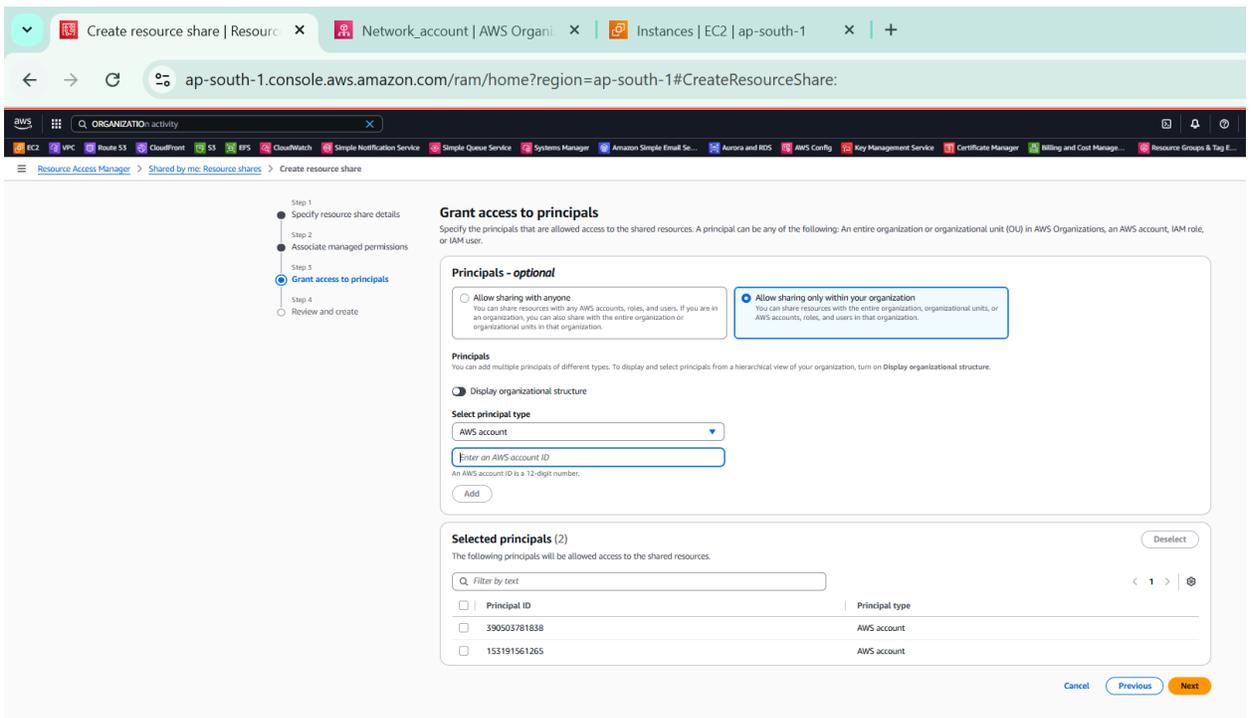
**Step 29) Here in RAM we will select the TGW we need to share with Account 2 and Account 3.**

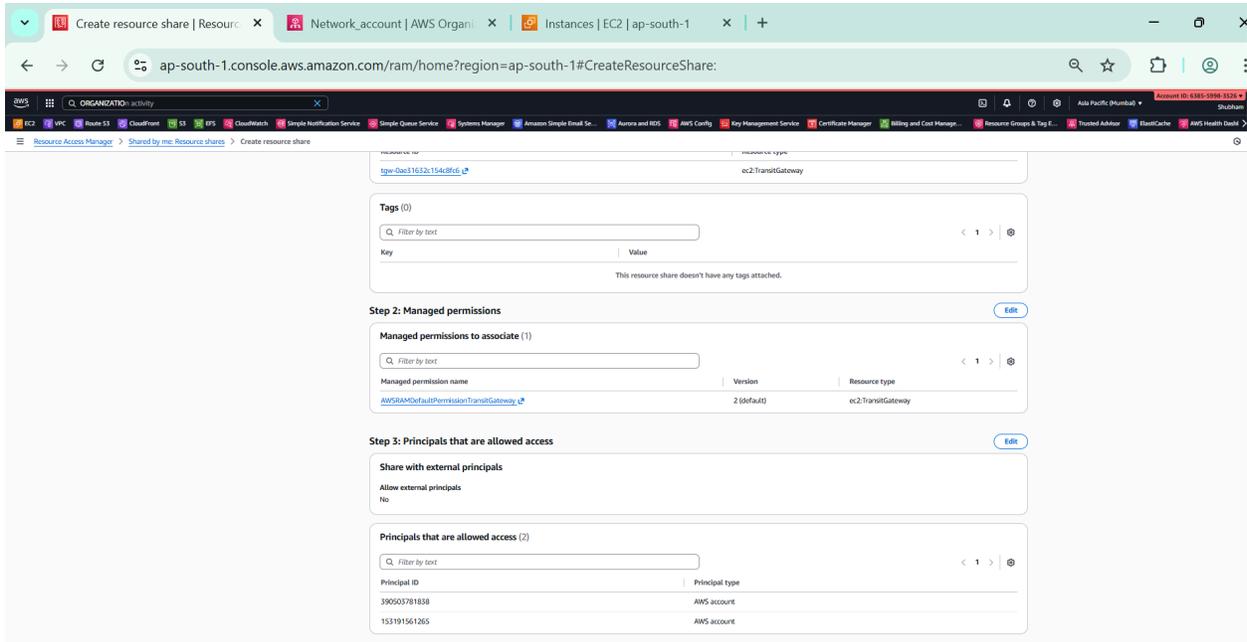


**Step 30) Also make sure that first in setting of RAM it allows sharing of resources in organization. AWS organization was already created and Account 1 is management account and Account 2 and Account 3 have joined that AWS organization.**

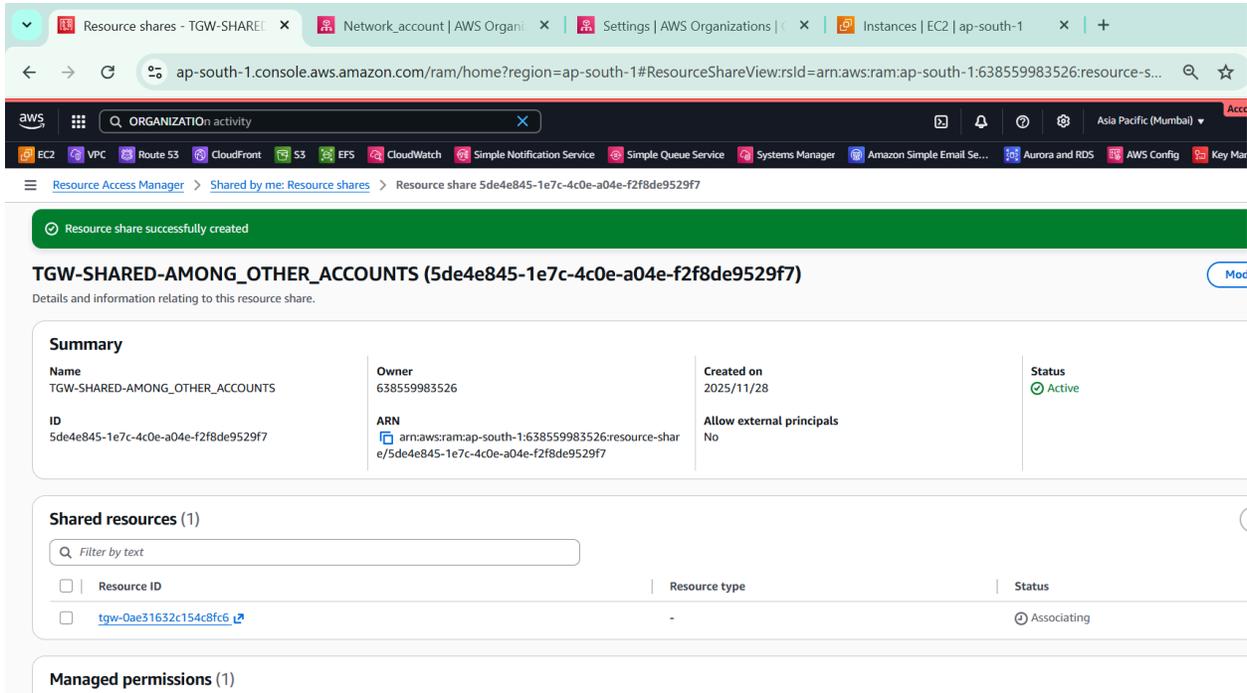


**Step 31) Now herw we will select the AWS accounts with whomTGW is to be shared.**





**Step 32) Now we can see the resource has been shared successfully with Account 2 and Account 3.**



**STEP 33) We can see the shared TGW in account 2 and account 3 below.**

The screenshot shows the AWS Management Console for account ID 3905-0378-1838 in the Asia Pacific (Mumbai) region. The page title is "Transit gateways | VPC | ap-sou...". The breadcrumb navigation is "VPC > Transit gateways". The main content area is titled "Transit gateways (1) info" and contains a search bar with the placeholder "Find transit gateway by attribute or tag". Below the search bar is a table with the following data:

<input type="checkbox"/>	Name	Transit gateway ID	Owner ID	State
<input type="checkbox"/>		<a href="#">tgw-0ae31632c154c8fc6</a>	638559983526	Available

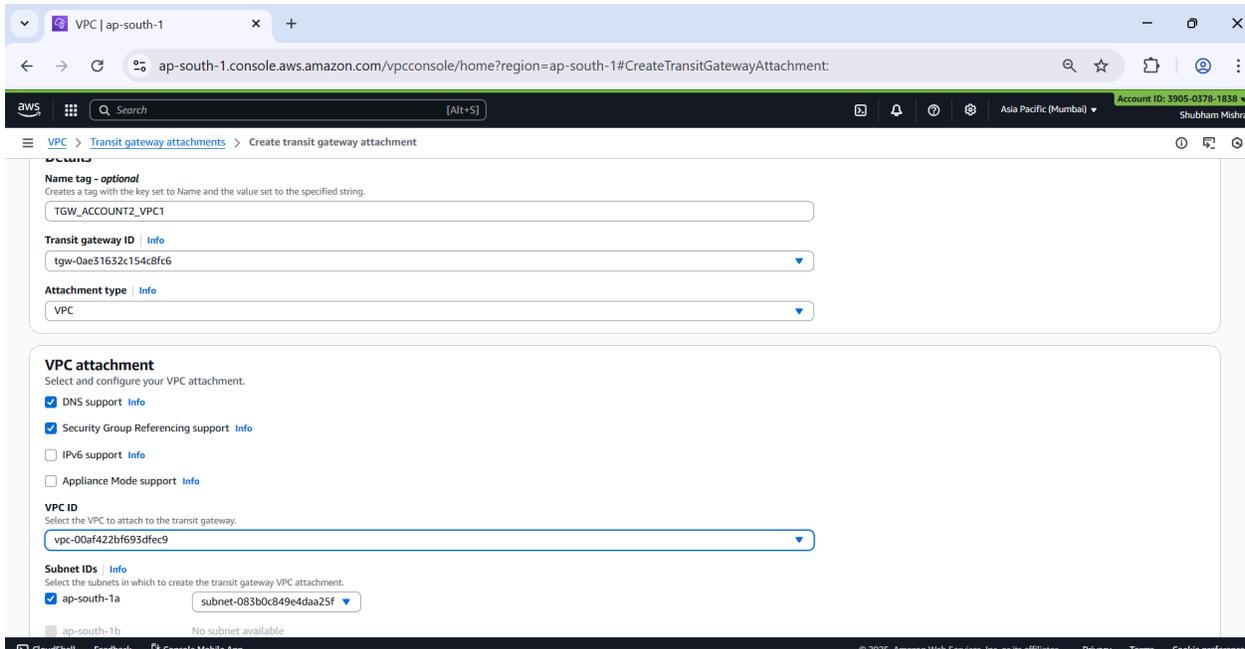
On the left sidebar, the "Transit gateways" section is expanded, showing sub-items like "Transit gateway attachments", "Transit gateway policy tables", "Transit gateway route tables", "Transit gateway multicast", and "Metering policies". Below this, the "Traffic mirroring" section is also expanded, showing "Mirror sessions" and "Mirror targets".

The screenshot shows the AWS Management Console for account ID 1531-9156-1265 in the Asia Pacific (Mumbai) region. The page title is "Transit gateways | VPC Console". The breadcrumb navigation is "VPC > Transit gateways". The main content area is titled "Transit gateways (1) info" and contains a search bar with the placeholder "Find transit gateway by attribute or tag". Below the search bar is a table with the following data:

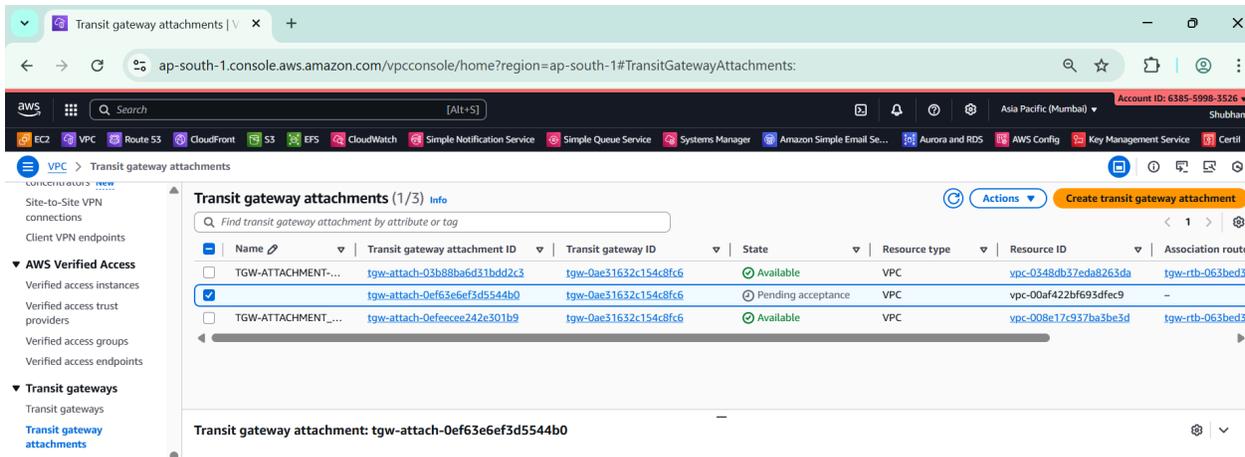
<input type="checkbox"/>	Name	Transit gateway ID	Owner ID	State
<input type="checkbox"/>		<a href="#">tgw-0ae31632c154c8fc6</a>	638559983526	Available

On the left sidebar, the "Virtual private network (VPN)" section is expanded, showing sub-items like "Customer gateways", "Virtual private gateways", "Site-to-site VPN concentrators", "Site-to-site VPN connections", and "Client VPN endpoints". Below this, the "AWS Verified Access" section is also expanded, showing "Verified access instances".

**Step 34) Now in account 2 we will create Tgw attachment for vpc1 and vpc2 and attach it to shared TGW.**



**Step 35) Now we can see that Tgw attachments from Account 2 and Account 3 are visible in Account 1.**



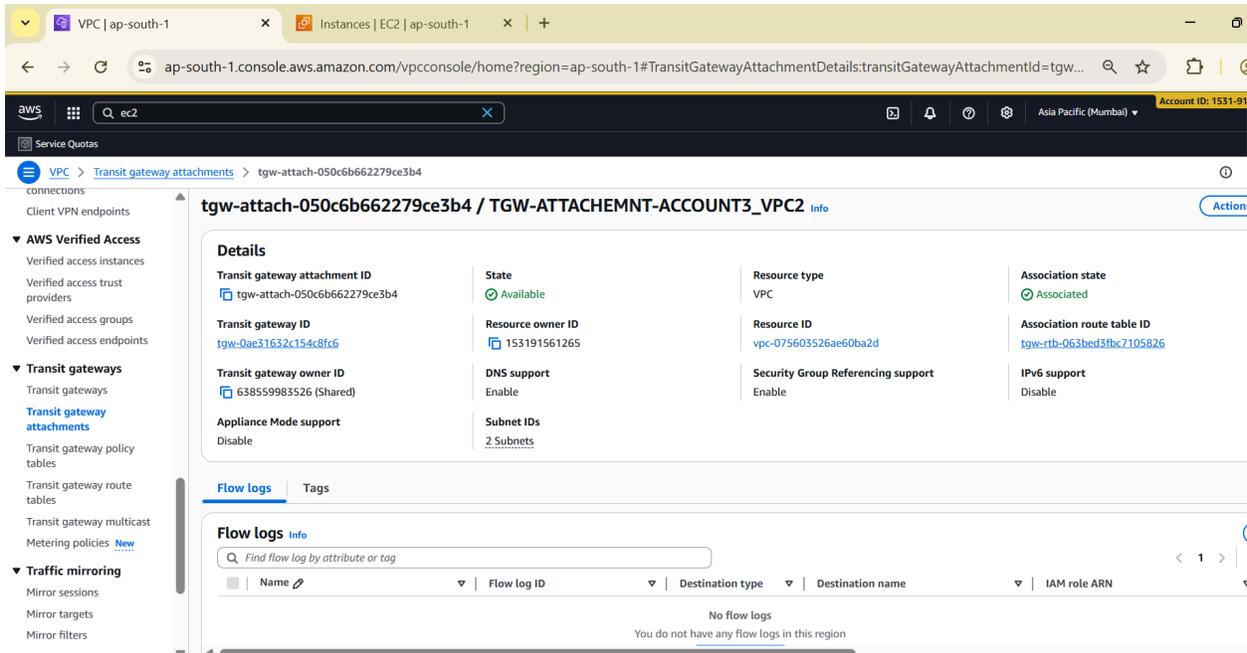
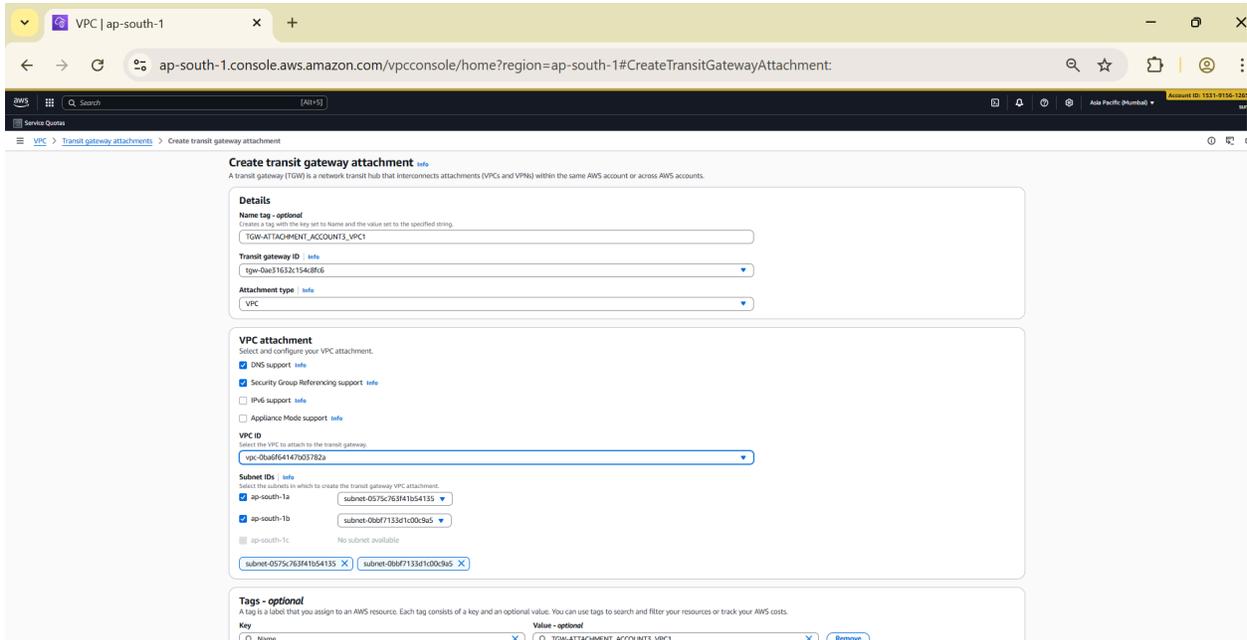
The screenshot shows the AWS Management Console interface for Transit gateway attachments. On the left, there is a navigation menu with 'VPC dashboard' and 'Virtual private cloud' sections. The main content area displays a table of three transit gateway attachments:

Name	Transit gateway attachment ID	Transit gateway ID	State	Resource type	Resource
TGW-ATTACHMENT_VPC1	tgw-attach-03b88ba6d51bdd2c3	tgw-0ae31632c154c8fc6	Available	VPC	vpc-0348
TGW-ATTACHMENT_ACCOUNT2_VPC1	tgw-attach-0ef63e6ef3d5544b0	tgw-0ae31632c154c8fc6	Available	VPC	vpc-00af4
TGW-ATTACHMENT_VPC_2	tgw-attach-0efeece242e301b9	tgw-0ae31632c154c8fc6	Available	VPC	vpc-008e

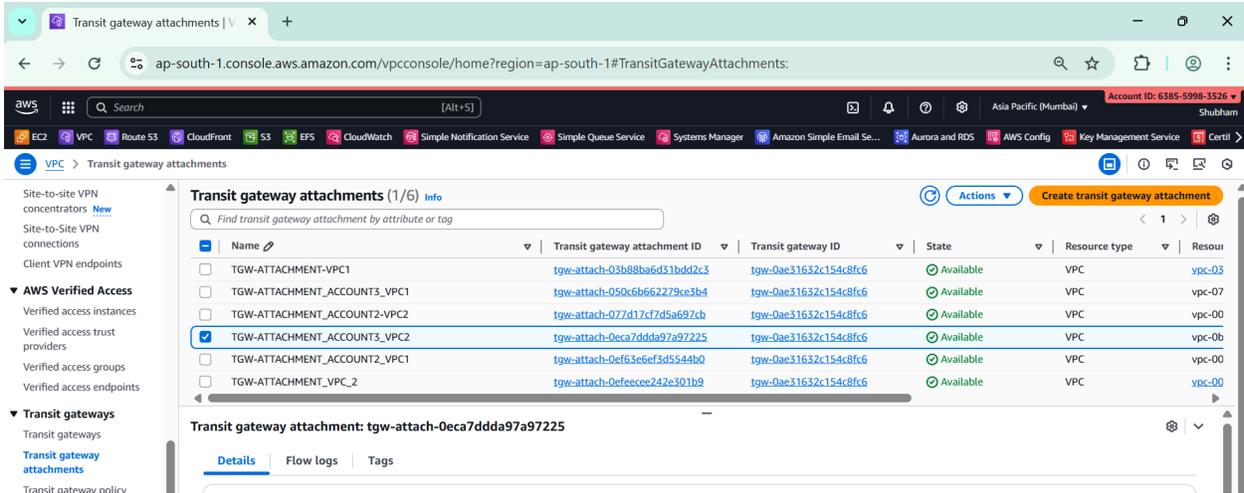
**Step 36) Similarly create Tgw attachment for vpc1 and vpc2 of account 3 to the shared TGW of account 1.**

The screenshot shows the 'Create transit gateway attachment' form in the AWS Management Console. The form is divided into several sections:

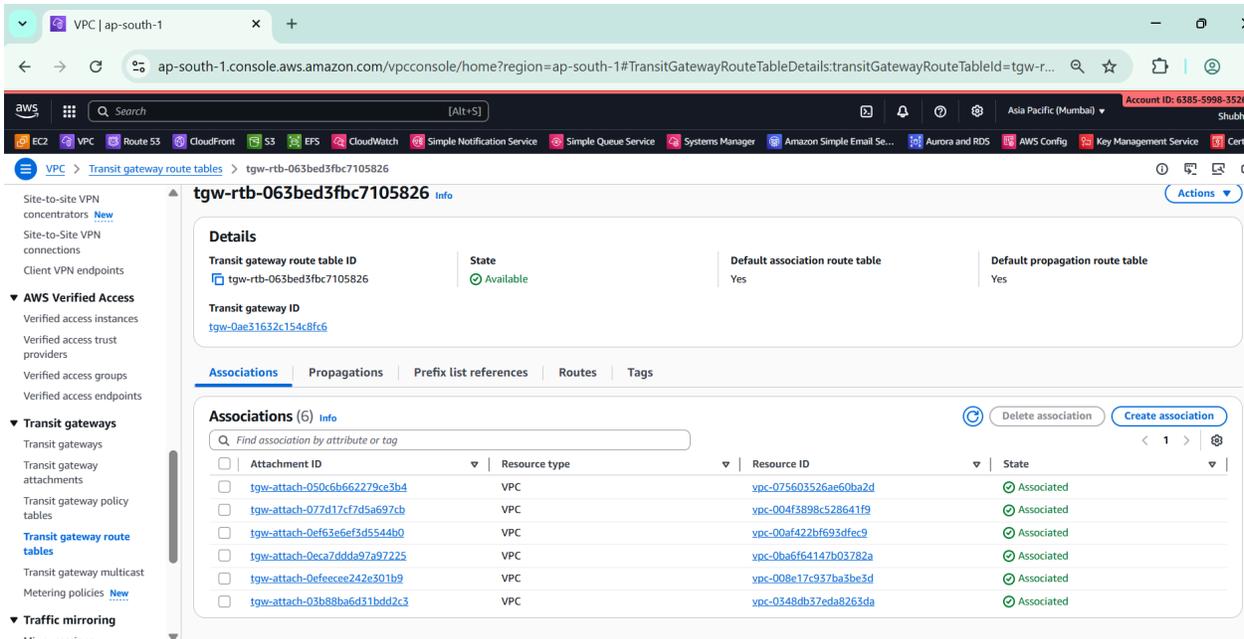
- Details:** Includes a text input for 'Name tag - optional' (value: TGW-ATTACHMENT\_ACCOUNT2\_VPC2), a dropdown for 'Transit gateway ID' (value: tgw-0ae31632c154c8fc6), a dropdown for 'Attachment type' (value: VPC), and a dropdown for 'VPC ID' (value: vpc-00af3896c528641f9).
- VPC attachment:** Includes checkboxes for 'DNS support', 'Security Group Referencing support', 'IPv6 support', and 'Appliance Mode support'. Below these are 'Subnet IDs' with checkboxes and dropdowns for 'ap-south-1a', 'ap-south-1b', and 'ap-south-1c'. The 'ap-south-1a' and 'ap-south-1b' options are selected.
- Tags - optional:** Includes a section for adding tags with a 'Key' (value: TGW-ATTACHMENT\_ACCOUNT2\_VPC2) and a 'Value' field.



**Step 37) Now we can see that Tgw attachments from Account 2 and Account 3 are visible in Account 1.**



**Step 38) Now we can see that all Tgw attachments to shared TGW route table below.**



**Step 39) We can also see propagations as well.**

**tgw-rtb-063bed3fbc7105826** info

**Details**

- Transit gateway route table ID: tgw-rtb-063bed3fbc7105826
- State: Available
- Default association route table: Yes
- Default propagation route table: Yes

**Propagations (6)** info

Attachment ID	Resource type	Resource ID	State
tgw-attach-03b88ba6d31bdd2c3	VPC	vpc-0348db37eda8263da	Enabled
tgw-attach-050c6b662279ce3b4	VPC	vpc-075603526ae60ba2d	Enabled
tgw-attach-077d17c7d5a697cb	VPC	vpc-004f3898c528641f9	Enabled
tgw-attach-0eca7ddda97a97225	VPC	vpc-0ba6f64147b03782a	Enabled
tgw-attach-0ef63e6ef3d5544b0	VPC	vpc-00af422bf693dfec9	Enabled
tgw-attach-0ef63e6ef3d5544b0	VPC	vpc-008e17c937ba3be3d	Enabled

**Step 40) After successful attachments to TGW routes are added automatically.**

**tgw-rtb-063bed3fbc7105826** info

**Routes (6)** info

**Filter routes by CIDR (2)**

- Exact CIDR: 0.0.0.0/0
- Longest prefix match: 0.0.0.0, ::
- Supernet of match: 0.0.0.0/0, ::/0
- Subnet of match: 0.0.0.0/0, ::/0

CIDR	Attachment ID	Resource ID	Resource type	Route type	Route state
10.0.0.0/16	tgw-attach-0eca7ddda97a97225	vpc-0ba6f64147b03782a	VPC	Propagated	Active
10.1.0.0/16	tgw-attach-050c6b662279ce3b4	vpc-075603526ae60ba2d	VPC	Propagated	Active
172.16.0.0/16	tgw-attach-03b88ba6d31bdd2c3	vpc-0348db37eda8263da	VPC	Propagated	Active
172.17.0.0/16	tgw-attach-0ef63e6ef3d5544b0	vpc-008e17c937ba3be3d	VPC	Propagated	Active
192.168.0.0/16	tgw-attach-0ef63e6ef3d5544b0	vpc-00af422bf693dfec9	VPC	Propagated	Active
192.169.0.0/16	tgw-attach-077d17c7d5a697cb	vpc-004f3898c528641f9	VPC	Propagated	Active

**Step 41) But now we will have to update each vpc route table across account 1, 2 and 3 to allow cidr block of other vpc across accounts.**

ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-0ad305570336f4a27

VPC dashboard <

AWS Global View

Filter by VPC:

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only Internet gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Route servers

Security

Network ACLs

Security groups

PrivateLink and Lattice

Getting started

Endpoints

Endpoint resolver

Details info

Route table ID: [rtb-0ad305570336f4a27](#)

Main: Yes

Explicit subnet associations: [subnet-07e79587f835cc11c](#) / [VPC\\_2\\_PUBLIC\\_SUBNET](#)

Edge associations: -

VPC: [vpc-008e17c937ba3be3d](#) | DEMO\_VPC\_2

Owner ID: [638559983526](#)

Routes (7)

Filter routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-002eb6baa8d8d82ac</a>	Active	No	Create Route
10.0.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
10.1.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.17.0.0/16	local	Active	No	Create Route Table
192.168.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.169.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route

ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-0b9cf6f0586644879

VPC dashboard <

AWS Global View

Filter by VPC:

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only Internet gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Route servers

Security

Network ACLs

Security groups

PrivateLink and Lattice

Getting started

Endpoints

Endpoint resolver

rtb-0b9cf6f0586644879 / VPC\_1\_PUBLIC\_RT

Actions

Details info

Route table ID: [rtb-0b9cf6f0586644879](#)

Main: Yes

Explicit subnet associations: [subnet-03243dad0564ef2db](#) / [VPC\\_1\\_PUBLIC\\_SUBNET](#)

Edge associations: -

VPC: [vpc-0348db37eda8263da](#) | DEMO\_VPC\_1

Owner ID: [638559983526](#)

Routes (7)

Filter routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-0aa4510f32d6b670</a>	Active	No	Create Route
10.0.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
10.1.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	local	Active	No	Create Route Table
172.17.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.168.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.169.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-0c63017f70965af7f

VPC > Route tables > rtb-0c63017f70965af7f

### VPC dashboard < rtb-0c63017f70965af7f / VPC\_1\_PUBLIC\_RT

Details info

Route table ID: [rtb-0c63017f70965af7f](#)

Main:  Yes

Owner ID: [390503781838](#)

Explicit subnet associations: [subnet-083b0c849e4daa25f / VPC\\_1\\_PUBLIC\\_SUBNET](#)

Edge associations: -

VPC: [vpc-00af422bf693dfec9 | VPC\\_1](#)

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

Routes (7)

Filter routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-043171f12530ada05</a>	Active	No	Create Route
10.0.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
10.1.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
172.17.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
192.168.0.0/16	local	Active	No	Create Route Table
192.169.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-01f5fb527d5515676

VPC > Route tables > rtb-01f5fb527d5515676

### VPC dashboard < rtb-01f5fb527d5515676 / VPC\_2\_PUBLIC\_RT

Details info

Route table ID: [rtb-01f5fb527d5515676](#)

Main:  Yes

Owner ID: [390503781838](#)

Explicit subnet associations: [subnet-0b37ed202ff4a5447 / VPC\\_2\\_PUBLIC\\_SUBNET](#)

Edge associations: -

VPC: [vpc-004f3898c528641f9 | VPC\\_2](#)

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

Routes (7)

Filter routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-05a2935157a163a37</a>	Active	No	Create Route
10.0.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
10.1.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
172.17.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
192.168.0.0/16	<a href="#">tgw-Oae31632c154c8fc6</a>	Active	No	Create Route
192.169.0.0/16	local	Active	No	Create Route Table

**rtb-0d72b6a61da0f2fb1 / VPC\_1\_PUBLIC\_RT**

**Details**

- Route table ID: [rtb-0d72b6a61da0f2fb1](#)
- Main:  Yes
- Explicit subnet associations: [subnet-0575c763f41b54135 / PUBLIC\\_SUBNET\\_VPC\\_1](#)
- Edge associations: -
- VPC: [vpc-0ba6f6417b03782a | VPC\\_1](#)
- Owner ID: [153191561265](#)

**Routes (7)**

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-0aa9dd2f80deceac</a>	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table
10.1.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.17.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.168.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.169.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route

**rtb-0d05678586a128051 / VPC\_2\_PUBLIC\_RT**

**Details**

- Route table ID: [rtb-0d05678586a128051](#)
- Main:  Yes
- Explicit subnet associations: [subnet-097d42d1514d620c / PUBLIC\\_SUBNET\\_VPC\\_2](#)
- Edge associations: -
- VPC: [vpc-075603526ae60ba2d | VPC\\_2](#)
- Owner ID: [153191561265](#)

**Routes (7)**

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	<a href="#">igw-02122b83adcd2790c</a>	Active	No	Create Route
10.1.0.0/16	local	Active	No	Create Route Table
10.2.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.16.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
172.17.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.168.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route
192.169.0.0/16	<a href="#">tgw-0ae31632c154c8fc6</a>	Active	No	Create Route

**Step 42) Now we can see that all vpc across three accounts can all ping each other over private ip address which confirms all the vpcs can reach other over internal network using TGW.**

```
root@vpc1s1: ~ x root@vpc1s2: ~ x
root@vpc1s1:~# ping 172.16.1.64 -c 2
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2
PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=64 time=0.019 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1035ms
rtt min/avg/max/mdev = 0.013/0.016/0.019/0.003 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=0.942 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=0.465 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.465/0.703/0.942/0.238 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=63 time=0.665 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=63 time=0.860 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1042ms
rtt min/avg/max/mdev = 0.665/0.762/0.860/0.097 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=63 time=0.653 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=63 time=0.504 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.504/0.578/0.653/0.074 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
64 bytes from 10.0.0.92: icmp_seq=1 ttl=63 time=0.622 ms
64 bytes from 10.0.0.92: icmp_seq=2 ttl=63 time=1.02 ms

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.622/0.821/1.021/0.199 ms
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
64 bytes from 10.1.0.43: icmp_seq=1 ttl=63 time=0.777 ms
64 bytes from 10.1.0.43: icmp_seq=2 ttl=63 time=0.823 ms

--- 10.1.0.43 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.777/0.800/0.823/0.023 ms
root@vpc1s1:~# S
```

```
root@vpc1s1: ~ x root@vpc1s2: ~ x
root@vpc1s2:~# ping 172.16.1.64 -c 2
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2

PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=0.906 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=0.439 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.439/0.672/0.906/0.233 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=64 time=0.011 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=64 time=0.021 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1029ms
rtt min/avg/max/mdev = 0.011/0.016/0.021/0.005 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=63 time=0.676 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=63 time=0.766 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1022ms
rtt min/avg/max/mdev = 0.676/0.721/0.766/0.045 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=63 time=1.50 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=63 time=0.422 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.422/0.958/1.495/0.536 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
64 bytes from 10.0.0.92: icmp_seq=1 ttl=63 time=1.34 ms
64 bytes from 10.0.0.92: icmp_seq=2 ttl=63 time=0.504 ms

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.504/0.924/1.344/0.420 ms
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
64 bytes from 10.1.0.43: icmp_seq=1 ttl=63 time=0.647 ms
64 bytes from 10.1.0.43: icmp_seq=2 ttl=63 time=0.983 ms

--- 10.1.0.43 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1060ms
rtt min/avg/max/mdev = 0.647/0.815/0.983/0.168 ms
root@vpc1s2:~#
```

```
root@vpc1s1: ~ × root@vpc1s2: ~ ×
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2

PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=1.54 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=0.587 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.587/1.064/1.542/0.477 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=0.644 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=0.579 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1026ms
rtt min/avg/max/mdev = 0.579/0.611/0.644/0.032 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=63 time=2.03 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=63 time=1.20 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.201/1.616/2.031/0.415 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=64 time=0.012 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=64 time=0.032 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1040ms
rtt min/avg/max/mdev = 0.012/0.022/0.032/0.010 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
64 bytes from 10.0.0.92: icmp_seq=1 ttl=63 time=1.66 ms
64 bytes from 10.0.0.92: icmp_seq=2 ttl=63 time=0.805 ms

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.805/1.230/1.656/0.425 ms
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
64 bytes from 10.1.0.43: icmp_seq=1 ttl=63 time=0.997 ms
64 bytes from 10.1.0.43: icmp_seq=2 ttl=63 time=0.772 ms

--- 10.1.0.43 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.772/0.884/0.997/0.112 ms
root@ac2v1s1:~#
```

```
root@vpc1s1: ~ × root@vpc1s2: ~ ×
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2

PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=0.995 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=0.645 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.645/0.820/0.995/0.175 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=0.758 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=0.804 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1009ms
rtt min/avg/max/mdev = 0.758/0.781/0.804/0.023 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=64 time=0.010 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=64 time=0.028 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.010/0.019/0.028/0.009 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=63 time=1.47 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=63 time=1.07 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.068/1.268/1.468/0.200 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
64 bytes from 10.0.0.92: icmp_seq=1 ttl=63 time=0.597 ms
64 bytes from 10.0.0.92: icmp_seq=2 ttl=63 time=1.19 ms

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1041ms
rtt min/avg/max/mdev = 0.597/0.891/1.186/0.294 ms
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
64 bytes from 10.1.0.43: icmp_seq=1 ttl=63 time=0.858 ms
64 bytes from 10.1.0.43: icmp_seq=2 ttl=63 time=0.663 ms

--- 10.1.0.43 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.663/0.760/0.858/0.097 ms
root@ac2v2s2:~#
```

```
root@ac3v1s1:~# ping 172.16.1.64 -c 2
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2

PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=0.880 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=0.430 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.430/0.655/0.880/0.225 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=1.20 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=0.885 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.885/1.044/1.204/0.159 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=63 time=0.909 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=63 time=0.656 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.656/0.782/0.909/0.126 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=63 time=1.73 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=63 time=0.779 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.779/1.255/1.732/0.476 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.
64 bytes from 10.0.0.92: icmp_seq=1 ttl=64 time=0.010 ms
64 bytes from 10.0.0.92: icmp_seq=2 ttl=64 time=0.026 ms

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1056ms
rtt min/avg/max/mdev = 0.010/0.018/0.026/0.008 ms
PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
```

```
root@ac3v2s2:~# ping 172.16.1.64 -c 2
ping 172.17.1.231 -c 2
ping 192.169.0.95 -c 2
ping 192.168.0.242 -c 2
ping 10.0.0.92 -c 2
ping 10.1.0.43 -c 2

PING 172.16.1.64 (172.16.1.64) 56(84) bytes of data.
64 bytes from 172.16.1.64: icmp_seq=1 ttl=63 time=0.747 ms
64 bytes from 172.16.1.64: icmp_seq=2 ttl=63 time=0.548 ms

--- 172.16.1.64 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.548/0.647/0.747/0.099 ms
PING 172.17.1.231 (172.17.1.231) 56(84) bytes of data.
64 bytes from 172.17.1.231: icmp_seq=1 ttl=63 time=0.643 ms
64 bytes from 172.17.1.231: icmp_seq=2 ttl=63 time=0.552 ms

--- 172.17.1.231 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1046ms
rtt min/avg/max/mdev = 0.552/0.597/0.643/0.045 ms
PING 192.169.0.95 (192.169.0.95) 56(84) bytes of data.
64 bytes from 192.169.0.95: icmp_seq=1 ttl=63 time=0.744 ms
64 bytes from 192.169.0.95: icmp_seq=2 ttl=63 time=0.660 ms

--- 192.169.0.95 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.660/0.702/0.744/0.042 ms
PING 192.168.0.242 (192.168.0.242) 56(84) bytes of data.
64 bytes from 192.168.0.242: icmp_seq=1 ttl=63 time=0.771 ms
64 bytes from 192.168.0.242: icmp_seq=2 ttl=63 time=0.736 ms

--- 192.168.0.242 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.736/0.753/0.771/0.017 ms
PING 10.0.0.92 (10.0.0.92) 56(84) bytes of data.

--- 10.0.0.92 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1021ms

PING 10.1.0.43 (10.1.0.43) 56(84) bytes of data.
64 bytes from 10.1.0.43: icmp_seq=1 ttl=64 time=0.034 ms
64 bytes from 10.1.0.43: icmp_seq=2 ttl=64 time=0.033 ms

--- 10.1.0.43 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1005ms
rtt min/avg/max/mdev = 0.033/0.033/0.034/0.000 ms
root@ac3v2s2:~#
```