

## VPC- Challenge1

Use Case: Setting up Transit Gateway and VPC Endpoints for a Multi-VPC Architecture

### **Scenario:**

A large organization is migrating its on-premises infrastructure to the AWS cloud. The organization's architecture involves multiple VPCs for different departments and applications, each requiring secure communication with centralized services and external resources. The IT team needs to design and implement a scalable and efficient network architecture to accommodate the organization's growth and ensure robust connectivity between VPCs and external services.

### **Objectives:**

- Design and deploy a scalable network architecture using AWS Transit Gateway to simplify network connectivity between multiple VPCs.
- Configure VPC endpoints to securely access AWS services without internet gateways or NAT gateways, ensuring data privacy and minimizing exposure to external threats.

# VPC- Challenge1

## Objective

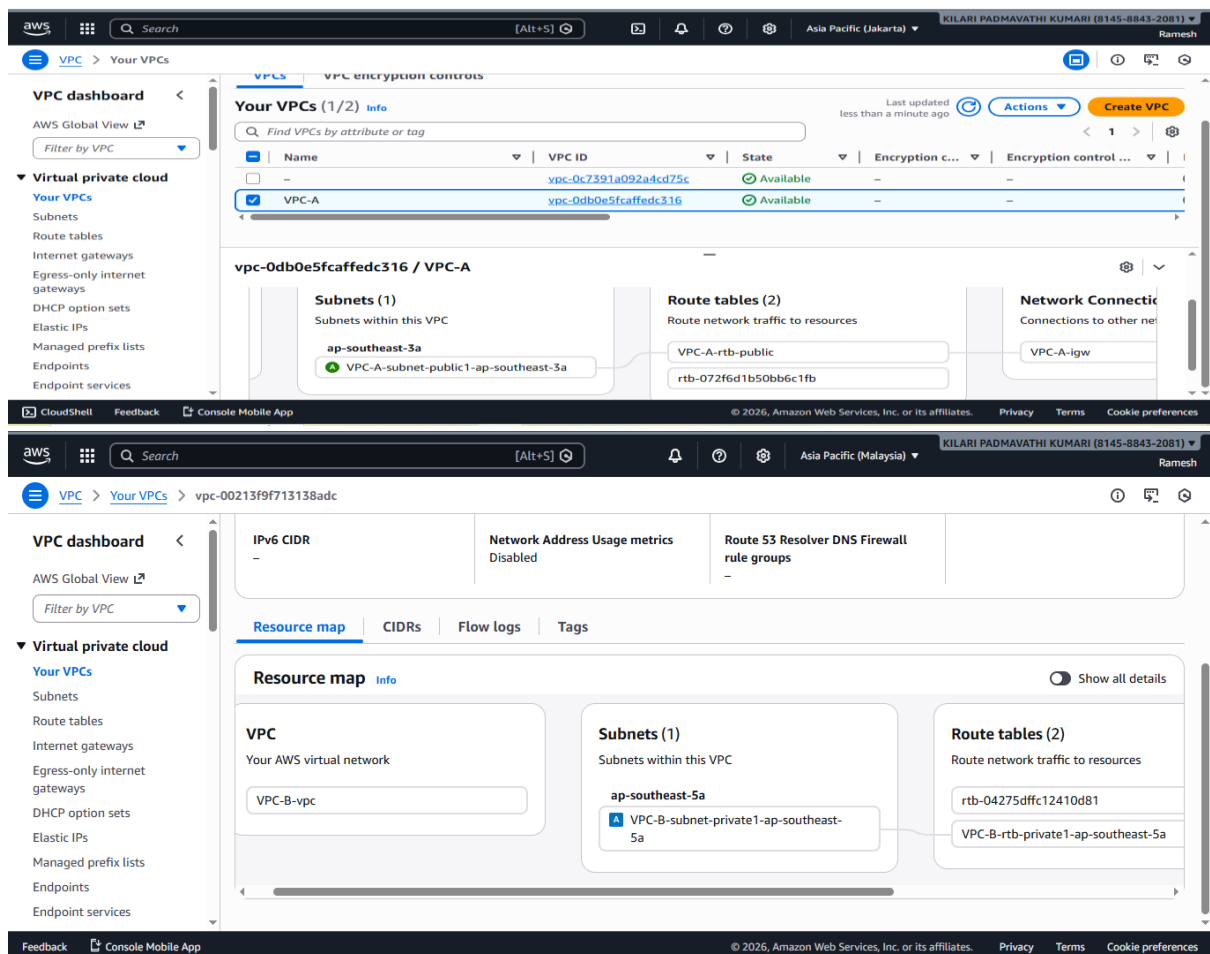
- To design a **centralized, scalable, and secure network architecture** that simplifies connectivity between multiple VPCs using **AWS Transit Gateway**, reducing operational complexity and enabling easy future expansion.

## Design Components:

- Transit Gateway
- VPC Attachments
- Transit Gateway Route Tables
- VPC Route Tables

## Deployment Steps

Firstly, create VPC , subnets and Route tables in 3 regions and CIDR should not overlap(Regions:-Jakarta, Malaysia, Melbourne)



# VPC- Challenge1

The screenshot shows the AWS VPC console dashboard for a specific VPC. The left sidebar contains navigation links for VPC, Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, and Endpoint services. The main content area displays the VPC dashboard with a resource map, CIDRs, Flow logs, Tags, and Integrations. The resource map shows the VPC-C-vpc, Subnets (1) including VPC-C-subnet-private1-ap-southeast-4a, and Route tables (2) including rtb-0c6c042c955a5ae64 and VPC-C-rtb-private1-ap-southeast-4a. The top bar shows the AWS logo, search bar, and user information.

## Step 1: Create a Transit Gateway

- Enable DNS support if required
- Disable auto-accept (recommended for security)

The screenshot shows the AWS Transit Gateway console. A green notification banner at the top states "You successfully created tgw-0e1df611f9755d849 / TGW-Mel." The left sidebar contains navigation links for VPC, Transit gateways, and various VPN options. The main content area displays the "Transit gateways (1)" section with a table listing the created gateway. Below the table, there is a "Select a transit gateway" section.

Name	Transit gateway ID	Owner ID	State
TGW-Mel	tgw-0e1df611f9755d849	814588432081	Pending

The screenshot shows the AWS Transit Gateway console. A green notification banner at the top states "You successfully created tgw-010c0c9e3fbac2d4 / TGW-Jakarta." The left sidebar contains navigation links for VPC, Transit gateways, and various VPN options. The main content area displays the "Transit gateways (1)" section with a table listing the created gateway. Below the table, there is a "Select a transit gateway" section.

Name	Transit gateway ID	Owner ID	State
TGW-Jakarta	tgw-010c0c9e3fbac2d4	814588432081	Pending

# VPC- Challenge1

**Introducing new feature: Metering Policy for Transit Gateway (TGW)**  
Previously, transit gateway data usage was metered solely to the source attachment owner. Now you can create custom metering policies to allocate data usage to source, destination, or central TGW accounts. [Learn more](#)

**Transit gateways (1/1)** Info

Find transit gateway by attribute or tag

<input checked="" type="checkbox"/>	Name	Transit gateway ID	Owner ID	State
<input checked="" type="checkbox"/>	TGW-Mal	tgw-0b89b9dd9a5b36ae4	814588432081	Available

**Transit gateway: tgw-0b89b9dd9a5b36ae4 / TGW-Mal**

Details | Flow logs | Sharing | Tags

**Details**

Transit gateway ID	State	Amazon ASN	DNS support
tgw-0b89b9dd9a5b36ae4	Available	64512	Enable

## Step 2: Create VPC Attachments

- Attach each VPC to the TGW
- Select private subnets only
- One attachment per VPC

You successfully created VPC attachment tgw-attach-0a66a523c0d00a776 / TG\_attach\_Mal.

**Transit gateway attachments (1)** Info

Find transit gateway attachment by attribute or tag

<input type="checkbox"/>	Name	Transit gateway attachment ID	Transit gateway ID	State	Resourc...	Reso
<input type="checkbox"/>	TG_attach_Mal	tgw-attach-0a66a523c0d00a776	tgw-0b89b9dd9a5b36ae4	Pending	VPC	ypc-C

Select a transit gateway attachment

You successfully created VPC attachment tgw-attach-0c55c0ad1cd0939c0 / TG\_attach\_Mel.

**Transit gateway attachments (1)** Info

Find transit gateway attachment by attribute or tag

<input type="checkbox"/>	Name	Transit gateway attachment ID	Transit gateway ID	State	Resourc...	Reso
<input type="checkbox"/>	TG_attach_Mel	tgw-attach-0c55c0ad1cd0939c0	tgw-0e1df611f9755d849	Pending	VPC	ypc-C

Select a transit gateway attachment

# VPC- Challenge1

The screenshot shows the AWS Management Console interface for the 'Transit gateway attachments' page. A green notification banner at the top states: 'You successfully created VPC attachment tgw-attach-0ac9742ec291bc2cf / TG\_attach\_jakarta.' Below this, the 'Transit gateway attachments (1)' section displays a table with one entry:

Name	Transit gateway attachment ID	Transit gateway ID	State	Resource
TG_attach_jakarta	tgw-attach-0ac9742ec291bc2cf	tgw-010c0c9e3fbcac2d4	Pending	VPC

Below the table, there is a section titled 'Select a transit gateway attachment' with a dropdown menu. The left sidebar shows navigation options under 'VPC' and 'Traffic Mirroring'. The top bar includes the AWS logo, search bar, and user information.

## Step 3: Configure Transit Gateway Route Tables

The screenshot shows the AWS Management Console interface for the 'Transit gateway route tables' page. The 'Transit gateway route tables (1/1)' section displays a table with one entry:

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route table
TGW_RT_J	tgw-rtb-05029982b5e6b46b8	tgw-010c0c9e3fbcac2d4	Available	Yes

Below the table, the 'Details' section for 'Transit gateway route tables: tgw-rtb-05029982b5e6b46b8 / TGW\_RT\_J' is shown. It includes fields for 'Transit gateway route table ID' (tgw-rtb-05029982b5e6b46b8), 'State' (Available), 'Default association route table' (Yes), and 'Default propagation route table' (Yes).

The screenshot shows the AWS Management Console interface for the 'Transit gateway route tables' page in a different region (Asia Pacific (Malaysia)). The 'Transit gateway route tables (1/1)' section displays a table with one entry:

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route table
TGW_RT_Mal	tgw-rtb-018a5fd3c17c82f53	tgw-0b89b9dd9a5b36ae4	Available	Yes

Below the table, the 'Details' section for 'Transit gateway route tables: tgw-rtb-018a5fd3c17c82f53 / TGW\_RT\_Mal' is shown. It includes fields for 'Transit gateway route table ID' (tgw-rtb-018a5fd3c17c82f53), 'State' (Available), 'Default association route table' (Yes), and 'Default propagation route table' (Yes). The left sidebar shows navigation options under 'VPC dashboard' and 'Virtual private cloud'. The top bar includes the AWS logo, search bar, and user information.

# VPC- Challenge1

The screenshot shows the AWS Management Console interface for the 'Transit gateway route tables' page. The left sidebar contains the 'VPC dashboard' and 'Virtual private cloud' sections. The main content area displays a table of transit gateway route tables. The table has columns for Name, Transit gateway route table ID, Transit gateway ID, State, and Default association route table. One table is listed: TG\_RT\_melb, with ID tgw-rtb-09ef84589ad21b5bf, associated with transit gateway tgw-0e1df611f9755d849, and is in an 'Available' state. Below the table, the 'Details' tab is selected, showing the route table ID, state, default association route table, and default propagation route table.

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route table
TG_RT_melb	tgw-rtb-09ef84589ad21b5bf	tgw-0e1df611f9755d849	Available	Yes

## Step 4: Update VPC Route Tables

Add routes in private subnet route tables:

Destination: 10.0.1.0/16 (Remote VPC)

Target: Transit Gateway

The screenshot shows the AWS Management Console interface for the 'Transit gateway route tables' page. The left sidebar contains the 'AWS Verified Access' and 'Transit gateways' sections. The main content area displays a table of transit gateway route tables. The table has columns for Name, Transit gateway route table ID, Transit gateway ID, State, and Default association route table. One table is listed: TGW\_RT\_J, with ID tgw-rtb-05029982b5e6b46b8, associated with transit gateway tgw-010c0c9e3fbcac2d4, and is in an 'Available' state. Below the table, the 'Details' tab is selected, showing the route table ID, state, default association route table, and default propagation route table. The 'Routes' tab is also visible, showing a list of routes with columns for CIDR, Attachment ID, Resource ID, Resource type, Route type, and Route status. Two routes are listed: 10.0.0.0/16 (Propagated, Active) and 11.0.0.0/16 (Static, Active).

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route table
TGW_RT_J	tgw-rtb-05029982b5e6b46b8	tgw-010c0c9e3fbcac2d4	Available	Yes

CIDR	Attachment ID	Resource ID	Resource type	Route type	Route status
10.0.0.0/16	tgw-attach-0ac9742ec291bc2cf	vpc-0db0e5fcafedc316	VPC	Propagated	Active
11.0.0.0/16	tgw-attach-001e1dd7d5d059b85	tgw-0b89b9dd9a5b36ae4	Peering	Static	Active

# VPC- Challenge1

The screenshot shows the AWS Management Console interface for the Asia Pacific (Melbourne) region. The left sidebar displays the VPC dashboard with a filter by VPC. The main content area shows the 'Transit gateway route tables (1/1)' page. A table lists the route tables, with 'TG\_RT\_melb' selected. Below the table, the 'Routes (2)' section shows two routes: one for CIDR 11.0.0/16 attached to 'tgw-attach-070f05ec06353deb9' and another for CIDR 12.0.0/16 attached to 'tgw-attach-0c55c0ad1cd0939c0'. Both routes are in an 'Active' state.

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route
TG_RT_melb	tgw-rtb-09ef84589ad21b5bf	tgw-0e1df611f9755d849	Available	Yes

CIDR	Attachment ID	Resource ID	Resource t...	Route type	Route stat
11.0.0/16	tgw-attach-070f05ec06353deb9	tgw-0b89b9dd9a5b36ae4	Peering	Static	Active
12.0.0/16	tgw-attach-0c55c0ad1cd0939c0	vpc-047027957b46d3605	VPC	Propagated	Active

The screenshot shows the AWS Management Console interface for the Asia Pacific (Malaysia) region. The left sidebar displays the VPC dashboard with a filter by VPC. The main content area shows the 'Transit gateway route tables (1/1)' page. A table lists the route tables, with 'TGW\_RT\_Mal' selected. Below the table, the 'Routes (2)' section shows three routes: one for CIDR 10.0.0/16 attached to 'tgw-attach-001e1dd7d5d059b85', another for CIDR 11.0.0/16 attached to 'tgw-attach-0a66a523c0d00a776', and a third for CIDR 12.0.0/16 attached to 'tgw-attach-070f05ec06353deb9'. All routes are in an 'Active' state.

Name	Transit gateway route table ID	Transit gateway ID	State	Default association route
TGW_RT_Mal	tgw-rtb-018a5fd3c17c82f53	tgw-0b89b9dd9a5b36ae4	Available	Yes

CIDR	Attachment ID	Resource ID	Resource t...	Route type	Route stat
10.0.0/16	tgw-attach-001e1dd7d5d059b85	tgw-010c0c9e3fbcac2d4	Peering	Static	Active
11.0.0/16	tgw-attach-0a66a523c0d00a776	vpc-00213f9f713138adc	VPC	Propagated	Active
12.0.0/16	tgw-attach-070f05ec06353deb9	tgw-0e1df611f9755d849	Peering	Static	Active

## Step 5: Security Configuration

- Use **Security Groups** and **NACLs**(follow these rules for all 3 regions)
- Restrict TGW routes to required CIDRs

The screenshot shows the AWS Management Console interface for the Asia Pacific (Malaysia) region, specifically the 'Edit inbound rules' page for a security group. The page displays a table of inbound rules. The first rule is for SSH traffic (Type: SSH, Protocol: TCP, Port range: 22) with source 'Cust...'. The second rule is for All ICMP - IPv4 (Type: All ICMP - IPv4, Protocol: ICMP, Port range: All) with source 'Cust...'. The third rule is for All traffic (Type: All traffic, Protocol: All, Port range: All) with source 'Cust...'. Each rule has a 'Delete' button. A new rule is being added with source 'sg-0c955e9bf02a2a3bd'.

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-07822103d229b9c21	SSH	TCP	22	Cust...	
sgr-08969c27efc2a04e3	All ICMP - IPv4	ICMP	All	Cust...	
sgr-06c2deddaa828c7ea	All traffic	All	All	Cust...	

# VPC- Challenge1

**Edit inbound rules** [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the VPC.

Rule number <a href="#">Info</a>	Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Allow/Deny <a href="#">Info</a>	
100	All traffic	All	All	0.0.0.0/0	Allow	<a href="#">Remove</a>
*	All traffic	All	All	0.0.0.0/0	Deny	

[Add new rule](#) [Sort by rule number](#)

[Cancel](#) [Preview changes](#) [Save changes](#)

## Step 6: High Availability

- TGW is **regionally highly available**
- Use multiple subnets (AZs) per attachment
- For multi-region:
  - Use **TGW Peering**

**Transit gateway attachments (2)** [Info](#)

[Find transit gateway attachment by attribute or tag](#)

<input type="checkbox"/>	Name	Transit gateway attachment ID	Transit gateway ID	State	Resource...	Re
<input type="checkbox"/>	TGA_peerJakarta	tgw-attach-001e1dd7d5d059b85	tgw-010c0c9e3fbac2d4	Available	Peering	tgw
<input type="checkbox"/>	TG_attach_jakarta	tgw-attach-0ac9742ec291bc2cf	tgw-010c0c9e3fbac2d4	Available	VPC	yp

**Select a transit gateway attachment**



# VPC- Challenge1

This screenshot shows the AWS Management Console for the Asia Pacific (Malaysia) region. The left-hand navigation pane is expanded to 'VPC', and the 'Transit gateway attachments' page is selected. The main content area displays a table of three transit gateway attachments, all in an 'Available' state. Below the table is a section titled 'Select a transit gateway attachment'.

Name	Transit gateway attachment ID	Transit gateway ID	State	Resource	Resource ID
Peer_jak-mal	tgw-attach-001e1dd7d5d059b85	tgw-0b89b9dd9a5b36ae4	Available	Peering	tgw-0b89b9dd9a5b36ae4
TGA-peerMalaysia	tgw-attach-070f05ec06353deb9	tgw-0b89b9dd9a5b36ae4	Available	Peering	tgw-0b89b9dd9a5b36ae4
TG_attach_Mal	tgw-attach-0a66a523cd00a776	tgw-0b89b9dd9a5b36ae4	Available	VPC	ypc-0a66a523cd00a776

This screenshot shows the AWS Management Console for the Asia Pacific (Melbourne) region. The left-hand navigation pane is expanded to 'VPC', and the 'Transit gateway attachments' page is selected. The main content area displays a table of two transit gateway attachments, both in an 'Available' state. Below the table is a section titled 'Select a transit gateway attachment'.

Name	Transit gateway attachment ID	Transit gateway ID	State	Resource	Resource ID
TGA_peerMel	tgw-attach-070f05ec06353deb9	tgw-0e1df611f9755d849	Available	Peering	tgw-0e1df611f9755d849
TG_attach_Mel	tgw-attach-0c55c0ad1cd0939c0	tgw-0e1df611f9755d849	Available	VPC	ypc-0c55c0ad1cd0939c0

Create EC2 instance for checking connectivity.

This screenshot shows the AWS Management Console for the Asia Pacific (Jakarta) region. The left-hand navigation pane is expanded to 'EC2', and the 'Instances' page is selected. A green banner at the top indicates 'Successfully initiated starting of i-07caf15af3726e4e2'. The main content area displays a table of one EC2 instance, 'EC2-jakarta', which is in a 'Running' state. Below the table, the 'Details' tab is selected, showing the instance summary.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
EC2-jakarta	i-07caf15af3726e4e2	Running	t3.micro	Initializing	View alarms +	ap-southeast-1

**i-07caf15af3726e4e2 (EC2-jakarta)**

**Instance summary**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-07caf15af3726e4e2	16.78.77.248   <a href="#">open address</a>	10.0.5.169

# VPC- Challenge1

The screenshot shows the AWS Management Console for the Asia Pacific (Malaysia) region. The left sidebar displays the navigation menu with 'EC2' selected. The main content area shows a notification 'Successfully initiated starting of i-075b8faa5920c9b24'. Below this, the 'Instances (1/1)' table lists one instance: EC2-Malaysia (i-075b8faa5920c9b24) in the 'Running' state, using the 't3.micro' instance type. The 'Instance summary' section shows the instance ID, public IPv4 address, and private IPv4 addresses (11.0.139.60).

**Instances (1/1)**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
EC2-Malaysia	i-075b8faa5920c9b24	Running	t3.micro	Initializing	View alarms +	ap-southeast-5

**i-075b8faa5920c9b24 (EC2-Malaysia)**

**Instance summary**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-075b8faa5920c9b24	-	11.0.139.60

The screenshot shows the AWS Management Console for the Asia Pacific (Melbourne) region. The left sidebar displays the navigation menu with 'EC2' selected. The main content area shows a notification 'Successfully initiated starting of i-07dda7623e82107e7'. Below this, the 'Instances (1/1)' table lists one instance: EC2-Melb (i-07dda7623e82107e7) in the 'Running' state, using the 't3.micro' instance type. The 'Instance summary' section shows the instance ID, public IPv4 address, and private IPv4 addresses (12.0.132.250).

**Instances (1/1)**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
EC2-Melb	i-07dda7623e82107e7	Running	t3.micro	Initializing	View alarms +	ap-southeast-4

**i-07dda7623e82107e7 (EC2-Melb)**

**Instance summary**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-07dda7623e82107e7	-	12.0.132.250

# VPC- Challenge1

## Connection: To Verify

Ec2 instance for Jakarta region

ec2-user@ip-10-0-5-169:~

```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ ssh -i "Jakarta_keypair.pem" ec2-user@ec2-16-78-77-248.ap-southeast-3.compute.amazonaws.com
The authenticity of host 'ec2-16-78-77-248.ap-southeast-3.compute.amazonaws.com (16.78.77.248)' can't be established.
ED25519 key fingerprint is SHA256:P5EhVyr1bNcgE/VUfRNHX0orqB4sSyt5mRPXl19yr+Y.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-16-78-77-248.ap-southeast-3.compute.amazonaws.com' (ED25519) to the list of known hosts.

      #_
     ~\  #####_      Amazon Linux 2023
    ~ ~ \#####\
    ~ ~  \###|
    ~ ~   \#/_____ https://aws.amazon.com/linux/amazon-linux-2023
    ~ ~    V~' '->
    ~ ~
    ~ ~. _ . _ /
    ~ ~ _/_/_/_/_/
    ~ ~ _/m/'

Last login: Tue Jan 20 09:57:04 2026 from 43.218.193.65
[ec2-user@ip-10-0-5-169 ~]$ |
```

```
[ec2-user@ip-10-0-5-169 ~]$ sudo su -
Last login: Tue Jan 20 09:53:20 UTC 2026 on pts/3
[root@ip-10-0-5-169 ~]# ls
malaysikey.pem
[root@ip-10-0-5-169 ~]# ssh -i malaysikey.pem ec2-user@11.0.139.60

      #_
     ~\  #####_      Amazon Linux 2023
    ~ ~ \#####\
    ~ ~  \###|
    ~ ~   \#/_____ https://aws.amazon.com/linux/amazon-linux-2023
    ~ ~    V~' '->
    ~ ~
    ~ ~. _ . _ /
    ~ ~ _/_/_/_/_/
    ~ ~ _/m/'

Last login: Tue Jan 20 09:56:44 2026 from 10.0.5.169
[ec2-user@ip-11-0-139-60 ~]$ |
```

## VPC- Challenge1

```
[ec2-user@ip-11-0-139-60 ~]$ ls  
melbourne.pem  
[ec2-user@ip-11-0-139-60 ~]$ ssh -i melbourne.pem ec2-user@12.0.132.250  
The authenticity of host '12.0.132.250 (12.0.132.250)' can't be established.  
ED25519 key fingerprint is SHA256:QNSALiqP7i9ifiaqOm5z4EbgIXDYgusVaXUda9RBU8.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '12.0.132.250' (ED25519) to the list of known hosts.
```

```
, #_~\ #### Amazon Linux 2023  
~~ \ #####  
~~ \|###|  
~~ \|#/ https://aws.amazon.com/linux/amazon-linux-2023  
~~ V~' '~>  
~~~~~  
~~~~_.._./_____  
~~~~/_/_/'/_/
```

```
Last login: Tue Jan 20 09:43:48 2026 from 11.0.139.60  
[ec2-user@ip-12-0-132-250 ~]$ |
```

[illegible]

## Conclusion


- By using **AWS Transit Gateway**, we can design a **scalable, secure, and centralized network architecture** that simplifies VPC connectivity, reduces operational overhead, and supports future growth without re-architecting the network.

## VPC- Challenge1

- Configure VPC endpoints to securely access AWS services without internet gateways or NAT gateways, ensuring data privacy and minimizing exposure to external threats.

### Objective:

- Implemented **VPC Gateway and Interface Endpoints (PrivateLink)** to enable secure, private access to AWS services without Internet or NAT Gateways.
- Ensured **data privacy and reduced attack surface** by keeping all service traffic on the AWS private backbone using Private DNS and endpoint policies.
- Optimized **security and cost** by eliminating public IP dependencies and enforcing least-privilege access controls.

 MINGW64:/c/Users/user/Downloads

```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws --version
aws-cli/2.32.26 Python/3.13.11 Windows/10 exe/AMD64

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws configure
AWS Access Key ID [*****MTIY]: AKIA33KKBB3I6K456467
AWS Secret Access Key [*****FtU5]: 5wTuFgg9XNqt6cEUFN4PDdaITFXP8QbysW
YeylPQ
Default region name [ap-south-1]: us-west-2
Default output format [json]: json

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ neelimaranis3
bash: neelimaranis3: command not found

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws sts get-caller-identity

{
  "UserId": "AIDA33KKBB3IWEUYMZHID",
  "Account": "814588432081",
  "Arn": "arn:aws:iam::814588432081:user/Neelima_DevOps"
}

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```

```
user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ vi Hello

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ aws s3 cp Hello s3://neelimaranis3/
upload: .\Hello to s3://neelimaranis3/Hello

user@DESKTOP-3KH1IRE MINGW64 ~/Downloads (master)
$ |
```