Configuration Manual

MSc Research Project
MSc Cybersecurity

Vishak Anandha Kumar
Student ID: x23206055

School of Computing
National College of Ireland

Supervisor: Prof: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Vishak Anandha Kumar
Student ID:	23206055
Programme:	MSc Cybersecurity **Year:** 2024
Module:	MSc Research Project
Lecturer:	Jawad Salahuddin
Submission Due Date:	12/12/2024
Project Title:	Multi-cloud Infrastructure Provisioning with Auto Scaling
Word Count:	819 **Page Count:** 07

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Vishak Anandha Kumar
Date:	12/12/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Vishak Anandha Kumar
Student ID: x23206055

1. Introduction

This configuration manual contains step by step configuration screenshots of the tools which are included in the deployment of Multi cloud Auto scaling. It aims to provide detailed guidance on setting up and integrating resources across AWS and Azure for seamless, scalable, and secure workload management.

2. System Requirements

Hardware Specifications

- Device: Acer Aspire 3 15
- Processor: AMD RYZEN 7000 series 5
- ROM: 512GB
- RAM: 8GB

Software Specification

- Windows 11
- Terraform and Notepad
- Windows Powershell, AWS and Azure CLI
- Amazon Web Services
- Azure

3. Configuration

For auto-scaling group mlcl-asg, the desired capacity is set at 2 instances; the capability to scale the configuration allows an upper limit of 4 instances and a lower limit of 2 instances. It uses a launch template (mlcl-tmp) specifying an AMI and t2.micro instance type. The ASG is integrated with the target group (mlcl-tg) by directing traffic through load balancing to ensure high availability and resource-efficient distribution. This setup is essential for handling variable workloads dynamically while maintaining fault tolerance and scalability.

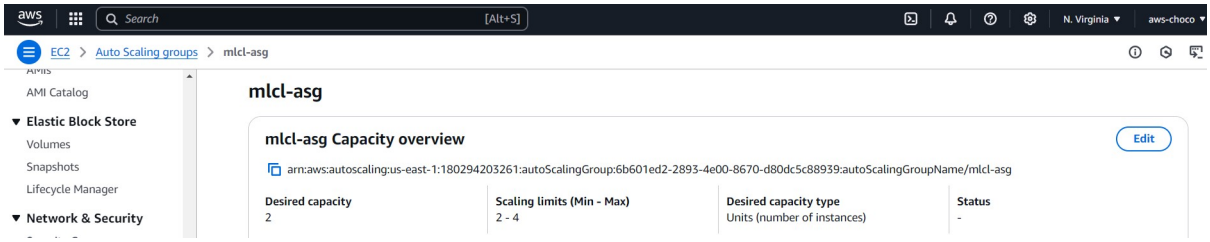


Fig 1

Details | Integrations - *new* | Automatic scaling | Instance management | Instance refresh

Launch template

Launch template	AMI ID	Instance type
lt-0329d7a319dab64ef mlcl-tmp	ami-0453ec754f44f9a4a	t2.micro
Version	**Security groups**	**Security group IDs**
Default	-	sg-0a1a611d8f5330d5c

Fig 1.1

Details | Integrations - *new* | Automatic scaling | Instance management

Load balancing

Load balancer target groups	Classic Load Balancers
mlcl-tg	-

Fig 1.2

- Terraform Installed and associated with my local to run code for aws and azure using cli.

```
Windows PowerShell

Terraform v1.9.8
on windows_386
PS C:\Users\VISHAK> terraform init
Terraform initialized in an empty directory!

The directory has no Terraform configuration files. You may begin working
with Terraform immediately by creating Terraform configuration files.
PS C:\Users\VISHAK> notepad main.tf
PS C:\Users\VISHAK> notepad main.tf
PS C:\Users\VISHAK> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.74.0...
- Installed hashicorp/aws v5.74.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\VISHAK>
```

Fig 2

- IAM has created just to get the access and secret key to connect azure cli with terraform

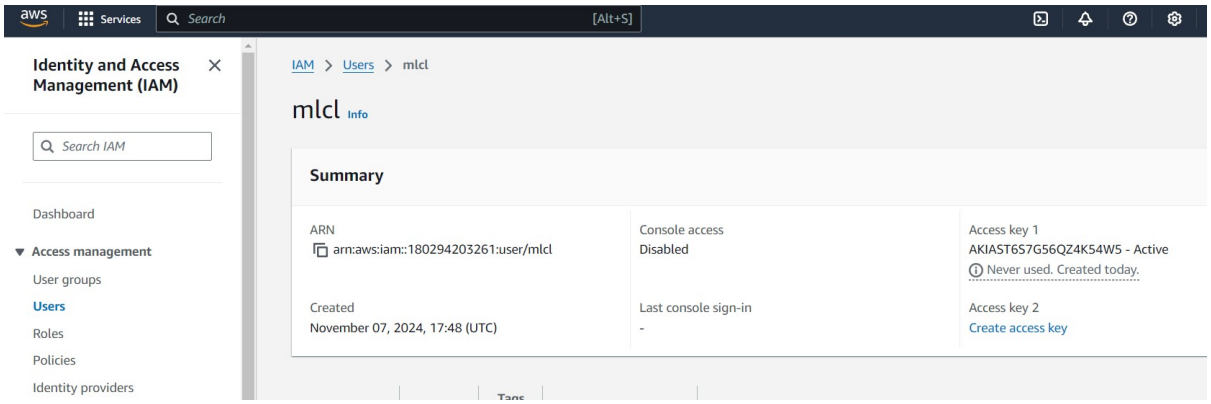
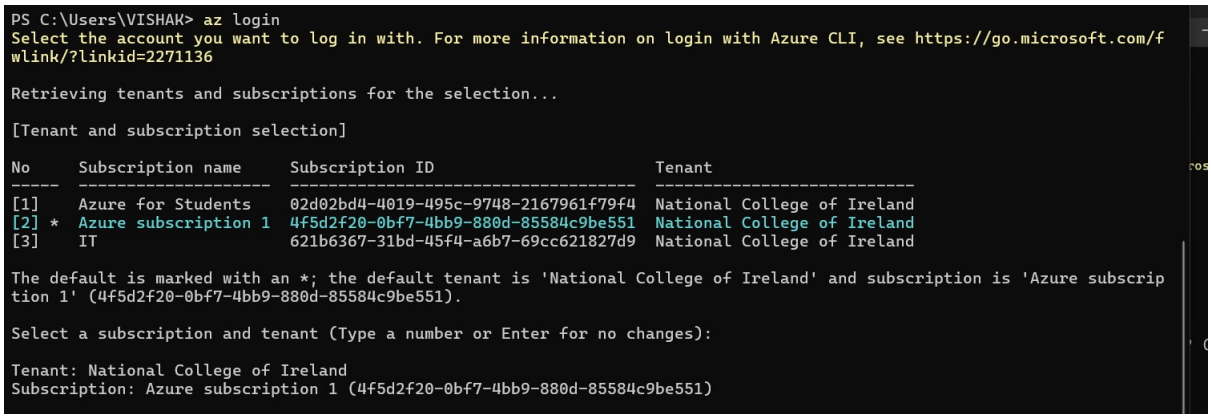


Fig 3

- Both aws and azure associated with terraform, meanwhile azure also get associated using subscription ID



```
PS C:\Users\VISHAK> az login
Select the account you want to log in with. For more information on login with Azure CLI, see https://go.microsoft.com/f
wlink/?linkid=2271136

Retrieving tenants and subscriptions for the selection...

[Tenant and subscription selection]

No     Subscription name     Subscription ID                       Tenant
-----  --------------------  ------------------------------------  -------------------------
[1]    Azure for Students    02d02bd4-4019-495c-9748-2167961f79f4  National College of Ireland
[2] *  Azure subscription 1  4f5d2f20-0bf7-4bb9-880d-85584c9be551  National College of Ireland
[3]    IT                    621b6367-31bd-45f4-a6b7-69cc621827d9  National College of Ireland

The default is marked with an *; the default tenant is 'National College of Ireland' and subscription is 'Azure subscrip
tion 1' (4f5d2f20-0bf7-4bb9-880d-85584c9be551).

Select a subscription and tenant (Type a number or Enter for no changes):

Tenant: National College of Ireland
Subscription: Azure subscription 1 (4f5d2f20-0bf7-4bb9-880d-85584c9be551)
```

Fig 4

- VPC has been created

Fig 5

- In this we can see that required features associated with VPC

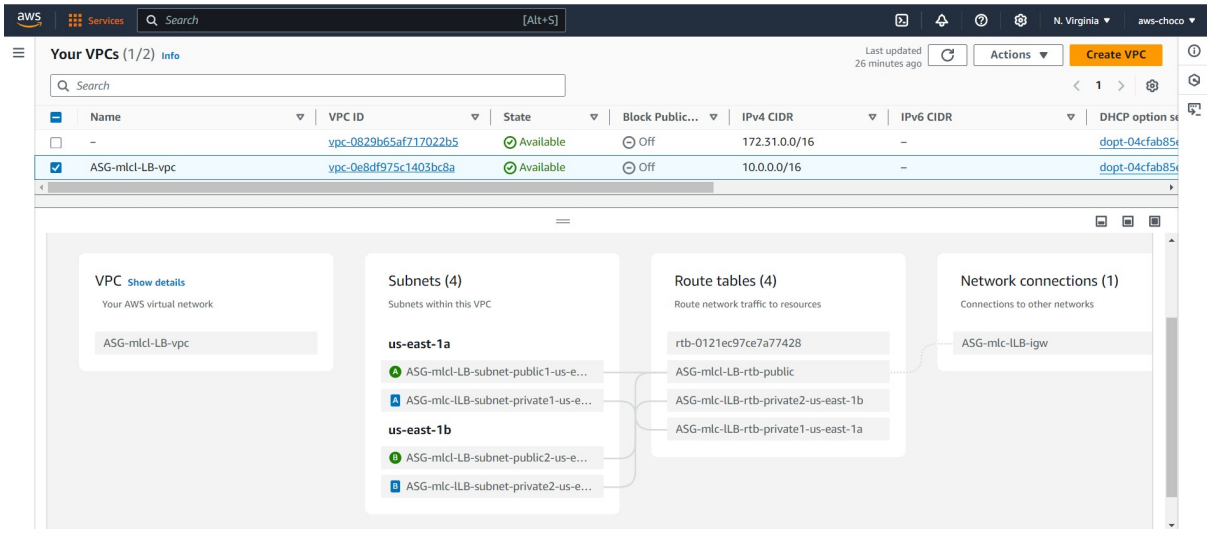


Fig 6

- The subnets are distributed across multiple availability zones (AZs) within the same region (us-east-1a and us-east-1b) to ensure fault tolerance. If one AZ becomes unavailable, resources in the other AZ can continue functioning without interruption, ensuring business continuity.

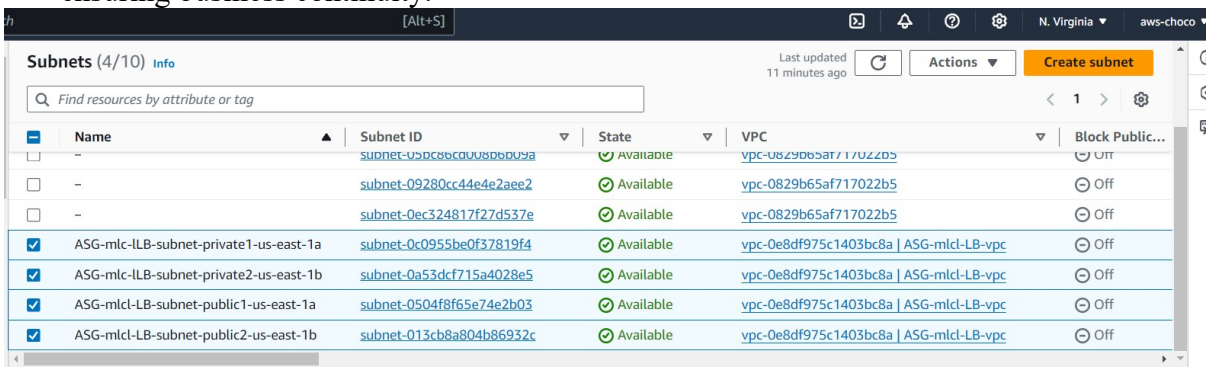


Fig 7

- VPC associated with subnets has been attached to Internet gateway

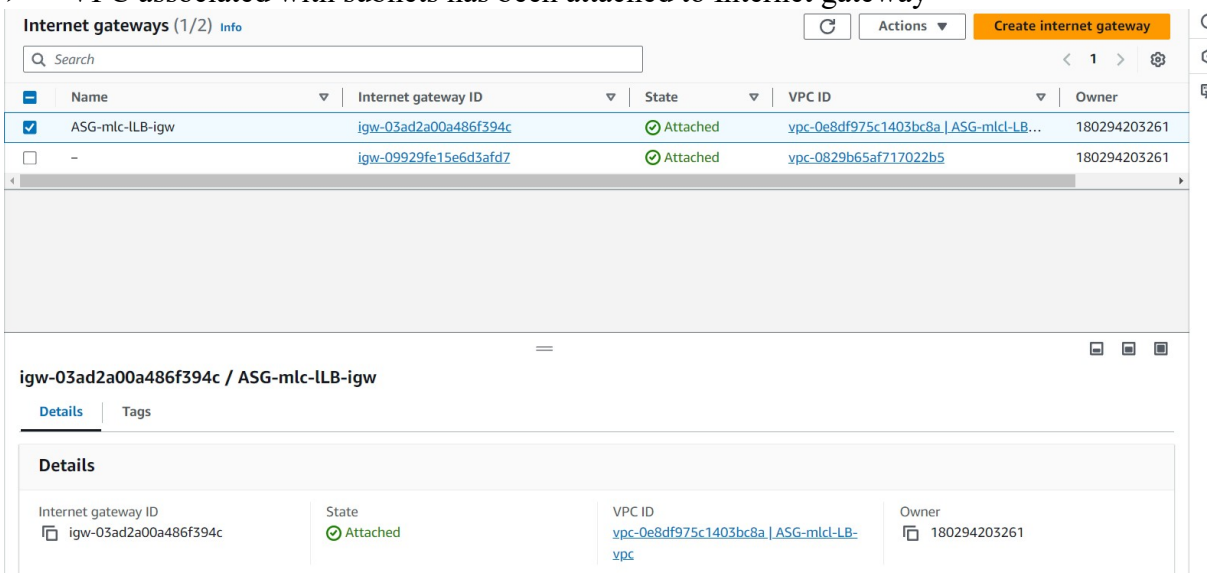


Fig 8

- The customer gateway here is the part of VPN which is to connect Azure with AWS.

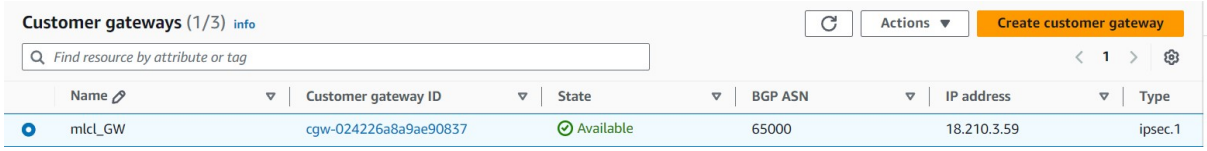


Fig 9

- The virtual private gateway creates secure connection between AWS VPC and Azure VN, so the VPC ID is attached here.

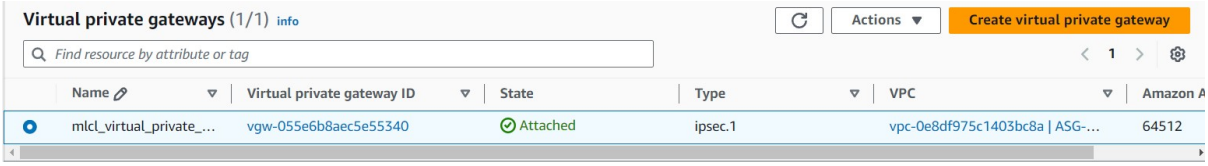


Fig 10

- Finally VPN connection established in AWS

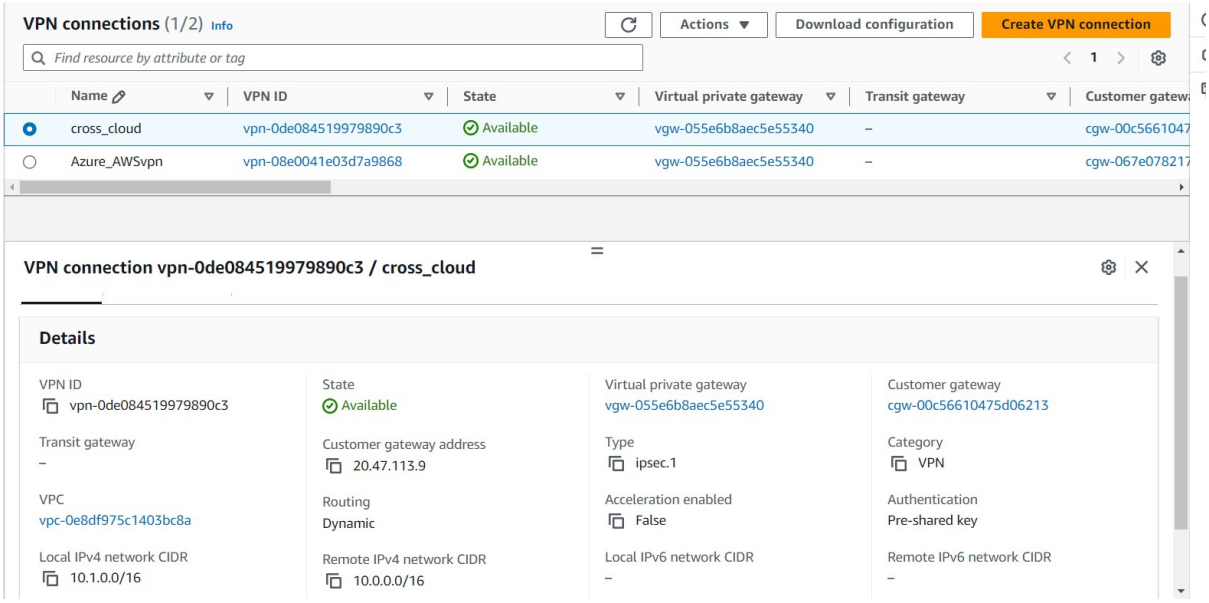


Fig 11

- In Azure, resource group ahs been created and necessary resources were also deployed.

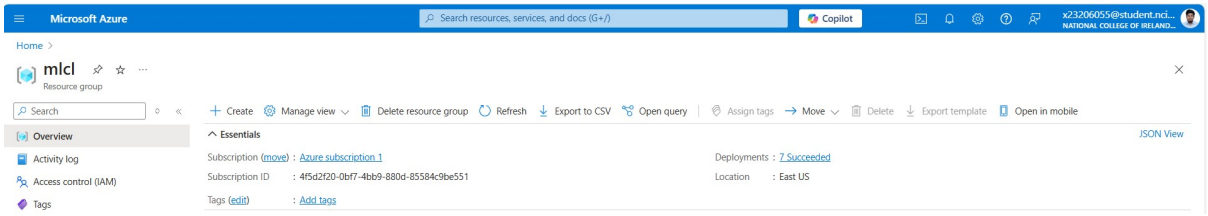


Fig 12

- Establish an Azure virtual network (mlclVN) with the right address space, for example, 10.1.0.0/16, to avoid overlapping IP address space with the AWS VPC and connect the Azure VNet to the VPN Gateway to communicate securely.

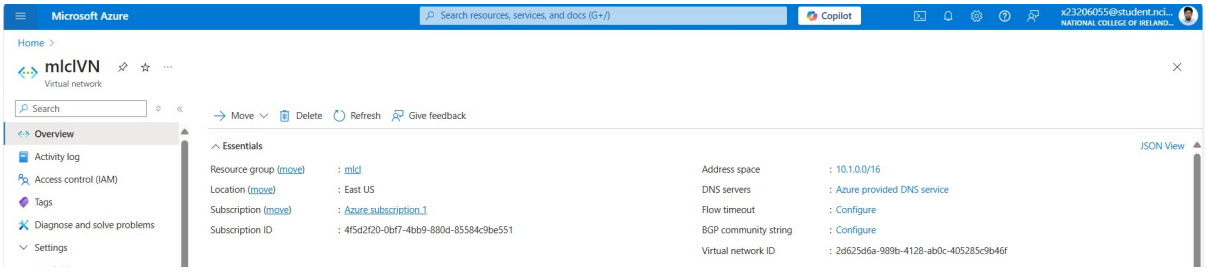


Fig 13

- Create a route-based VPN Gateway (VpnGw2AZ) in Azure and connect it to the AWS VGW. Configure the BGP settings and Azure route table to manage traffic between Azure and AWS seamlessly.

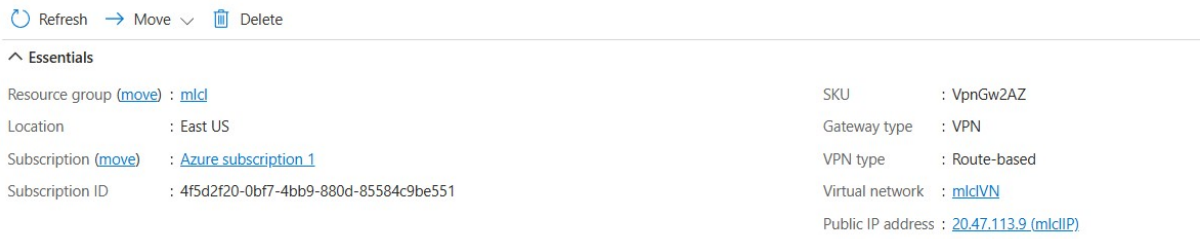


Fig 14

- Local Network Gateway is the Azure endpoint managing the VPN connection to the AWS Virtual Private Gateway. Ensure proper pairing and routing to bridge the networks so Azure can communicate across cloud with AWS.

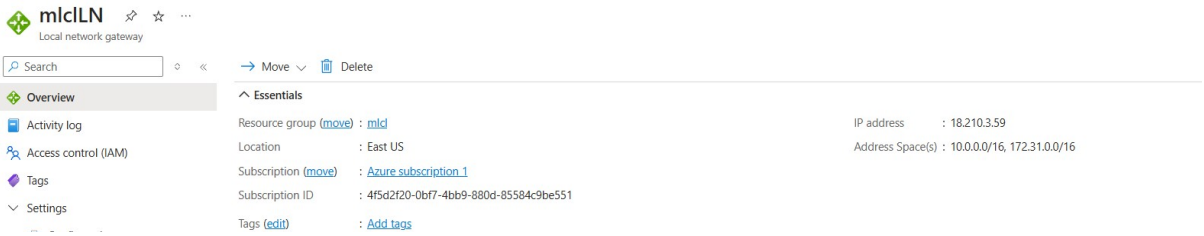


Fig 15

- Allows UDP traffic on port 4500 to Azure virtual network (10.1.0.0/16) from AWS CIDR block (10.0.0.0/16). Enables traffic from Azure virtual network (10.1.0.0/16) to AWS CIDR block.

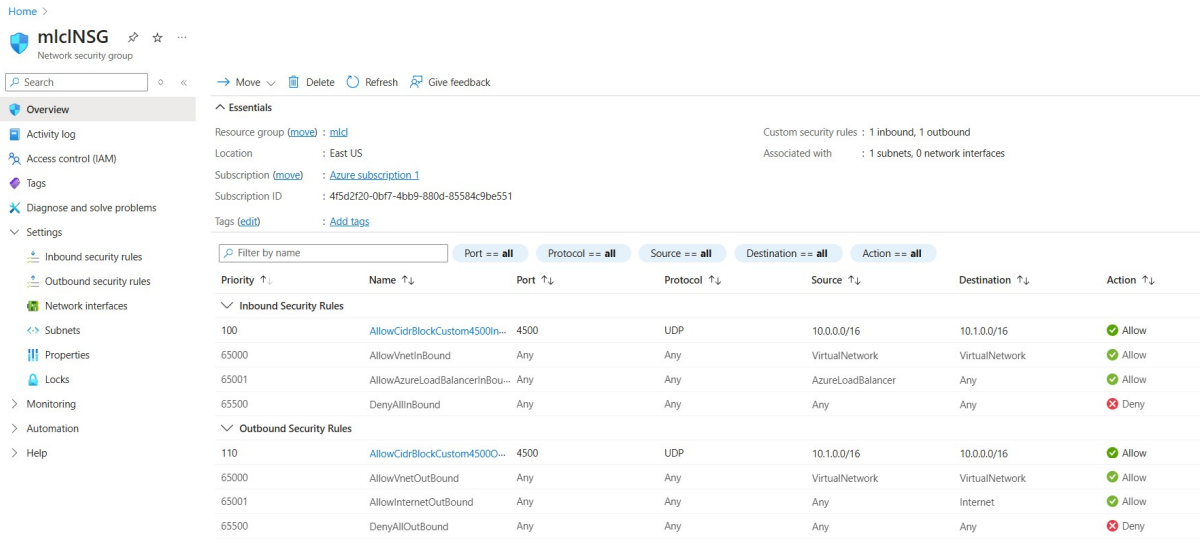


Fig 16

- A route will forward any traffic to AWS networks (10.0.0.0/16) through the Virtual Network Gateway. So, Azure resources in that subnet can now communicate with AWS resources over the VPN connection.

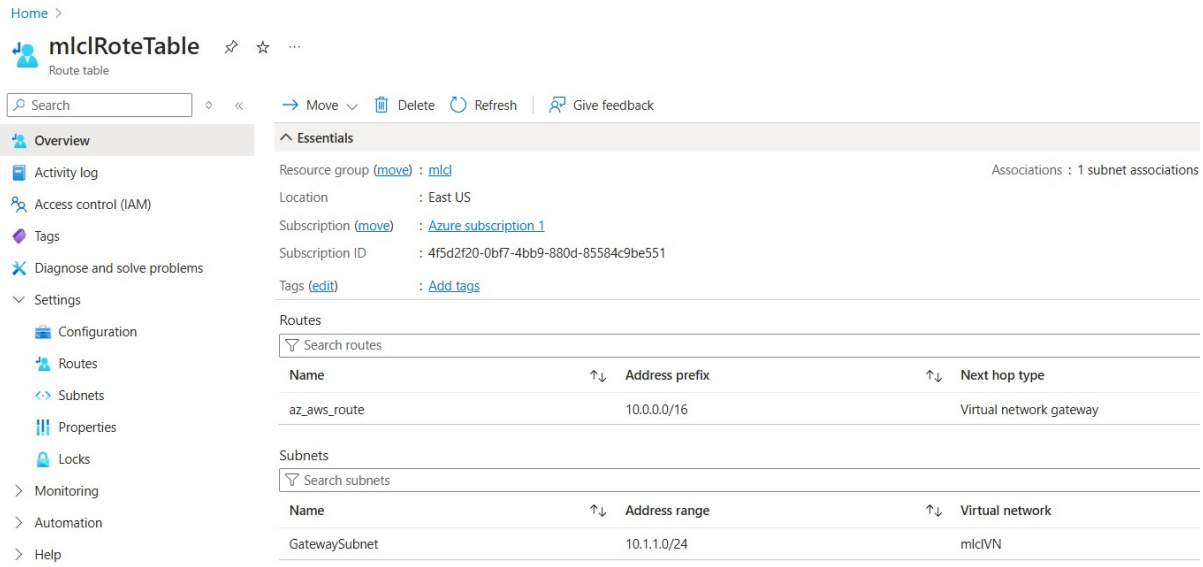


Fig 17

Reference

N/A.