Prepared by: Nandini Goyal
Intern at Celebal Technologies
Department: Cloud Infra & Security
Date: 18 July 2025

R&D Document

Azure Load Balancer Implementation

Table of Contents

Serial Number	Content	Page Number
1	Objective	3
2	Overview of Azure Load Balancer 2.1 What is Load Balancing? 2.2 Types of Load Balancer in Azure 2.2.1 Public (External) Load Balancer 2.2.2 Internal Load Balancer 2.3 Key Components	3
3	Objective	4
4	Implementation Steps 4.1 Resource Group and Virtual Network Setup 4.2 Virtual Machine Creation 4.3 Backend Pool Configuration 4.4 Internal Load Balancer Configuration 4.5 Health Probe Setup 4.6 Load Balancing Rule Creation 4.7 Testing Internal Load Balancing	4
5	Issues with External Load Balancer	11
9	Conclusion	17
10	References	17

1. Introduction

In modern cloud infrastructures, distributing network traffic efficiently across multiple virtual machines (VMs) ensures high availability, scalability, and fault tolerance. This report demonstrates the setup and working of an **Azure Load Balancer**, specifically focusing on **Internal Load Balancer**, and explains the challenge encountered during **External Load Balancer** configuration due to limited feature availability.

2. Overview of Azure Load Balancer

2.1 What is Load Balancing?

Load balancing is the process of distributing incoming network traffic across multiple servers (or VMs). It helps in:

- Avoiding overload on a single server
- Increasing responsiveness
- Ensuring high availability

2.2 Types of Load Balancer in Azure

2.2.1 Public Load Balancer (External)

Distributes traffic from the internet to Azure VMs via a public IP address.

2.2.2 Internal Load Balancer (Private)

Distributes traffic **within a virtual network** using a private IP address. Commonly used for intra-application traffic.

2.3 Key Components

- **Frontend IP Configuration**: IP exposed to clients
- **Backend Pool**: Set of VMs to distribute traffic
- **Health Probe**: Checks availability of VMs
- **Load Balancing Rule**: Maps frontend IP to backend pool
- **Inbound NAT Rule**: For directing specific port traffic to a VM

3. Objective

To configure and demonstrate the working of **Internal Load Balancer** using Azure VMs, and understand the limitations in configuring the **External Load Balancer** due to subscription constraints.

4. Implementation Steps

4.1 Resource Group and Virtual Network Setup

- Create a Resource Group (e.g., `Week7Group`)
- Create a Virtual Network `Week7VNet` with subnet `Week7Subnet`

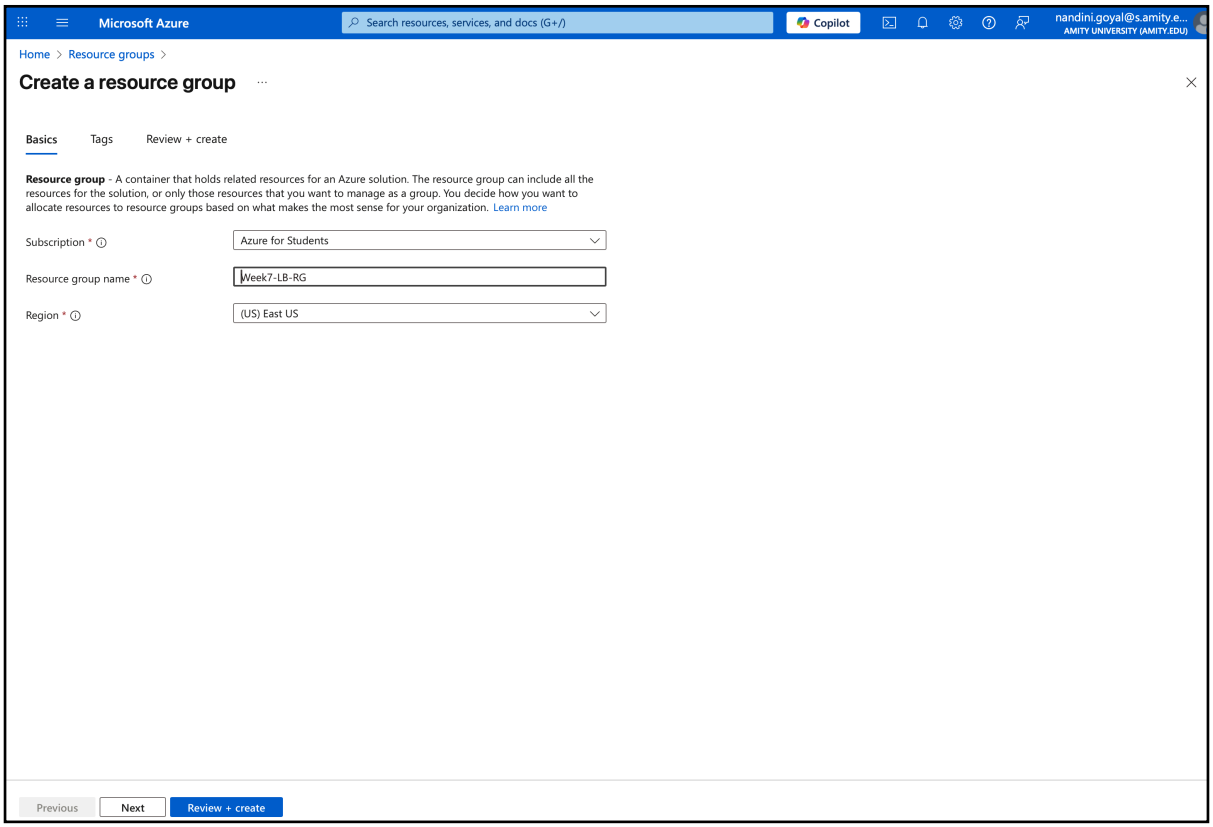


Figure 1: Resource Group creation

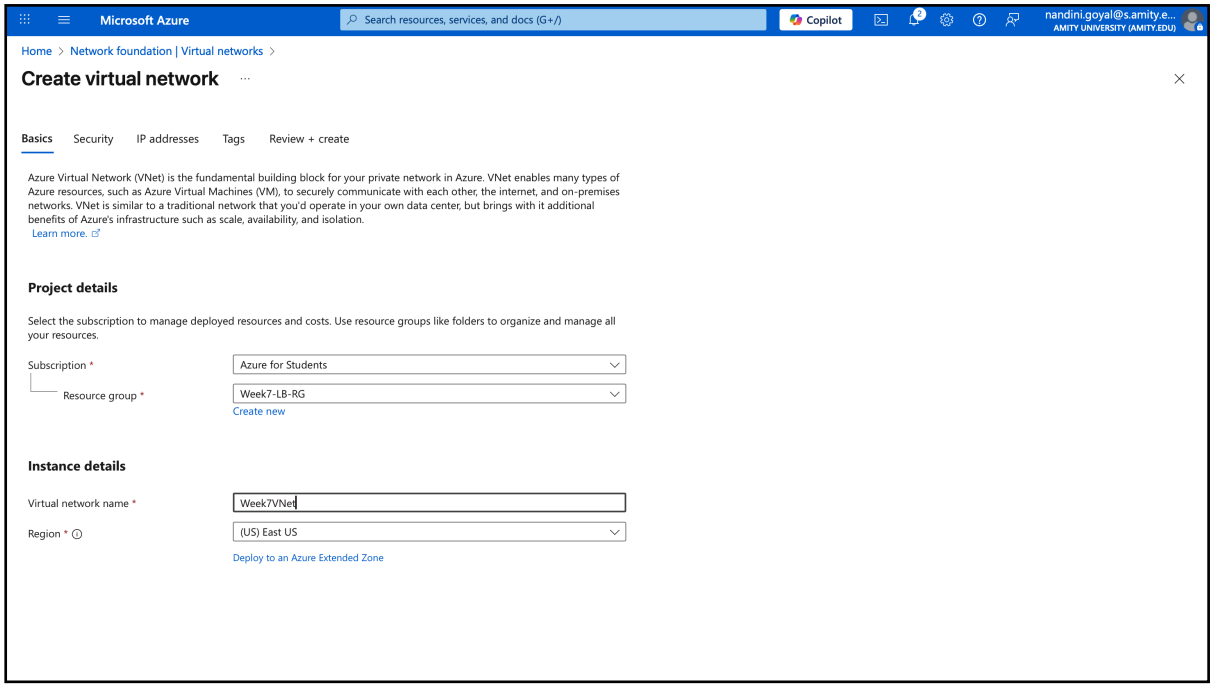


Figure 2: Creating Week7VNet

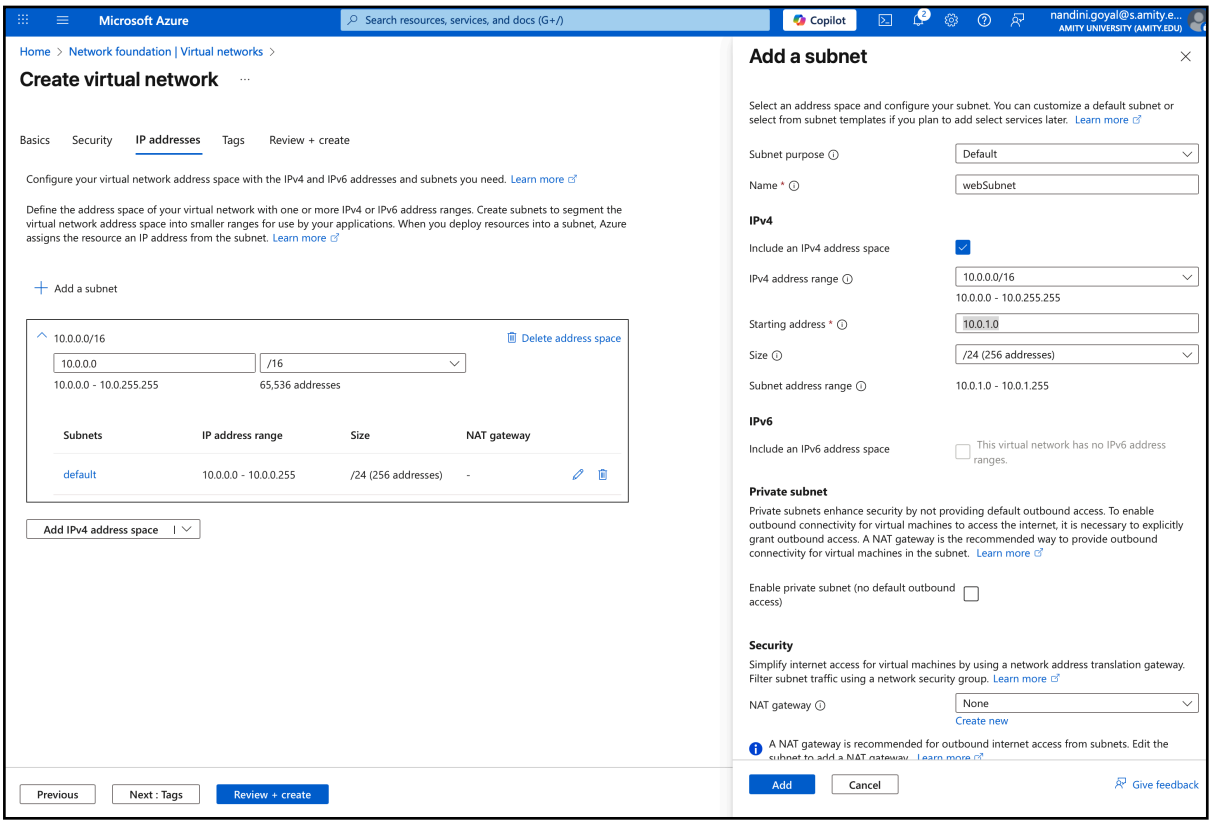


Figure 3: Creating WebSubnet in Week7VNet

4.2 Virtual Machine Creation

- Create three VMs: `VM1`, `VM2`, `VM3` in same region and subnet.
- Enable connectivity using Bastion or Public IP (for initial access).

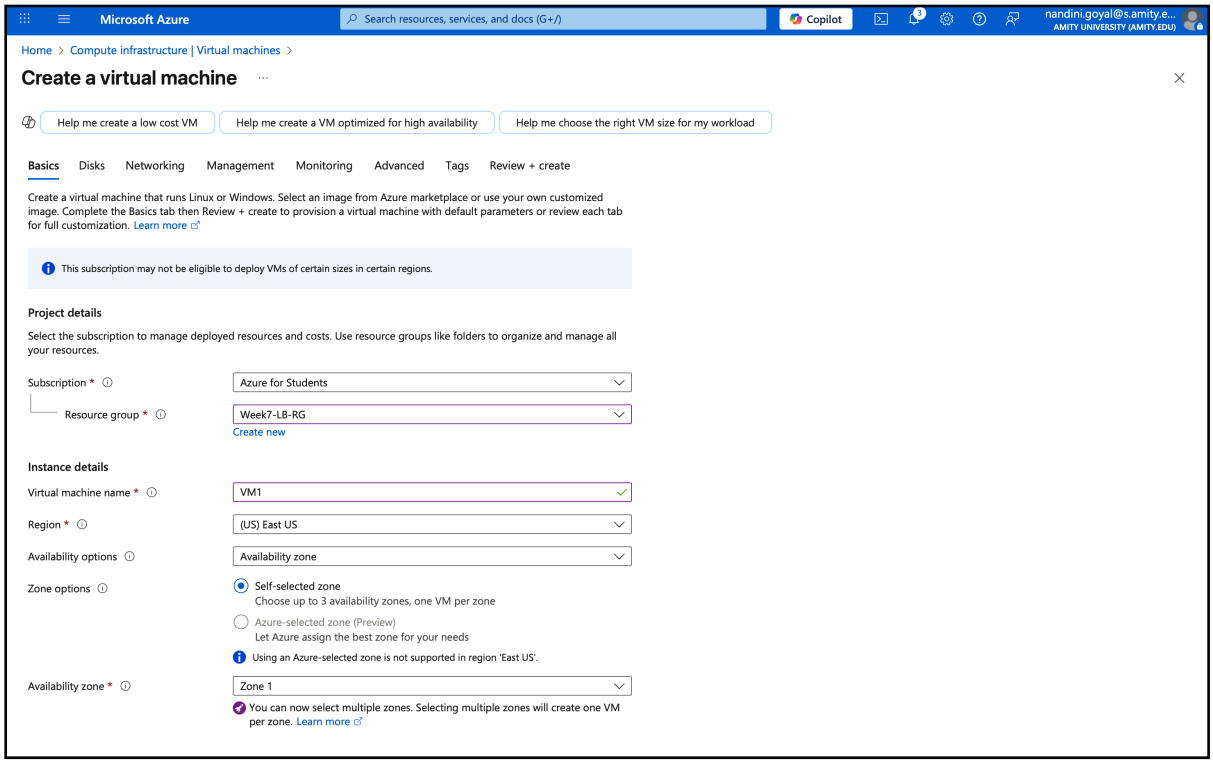


Figure 4: VM1 Basics Tab

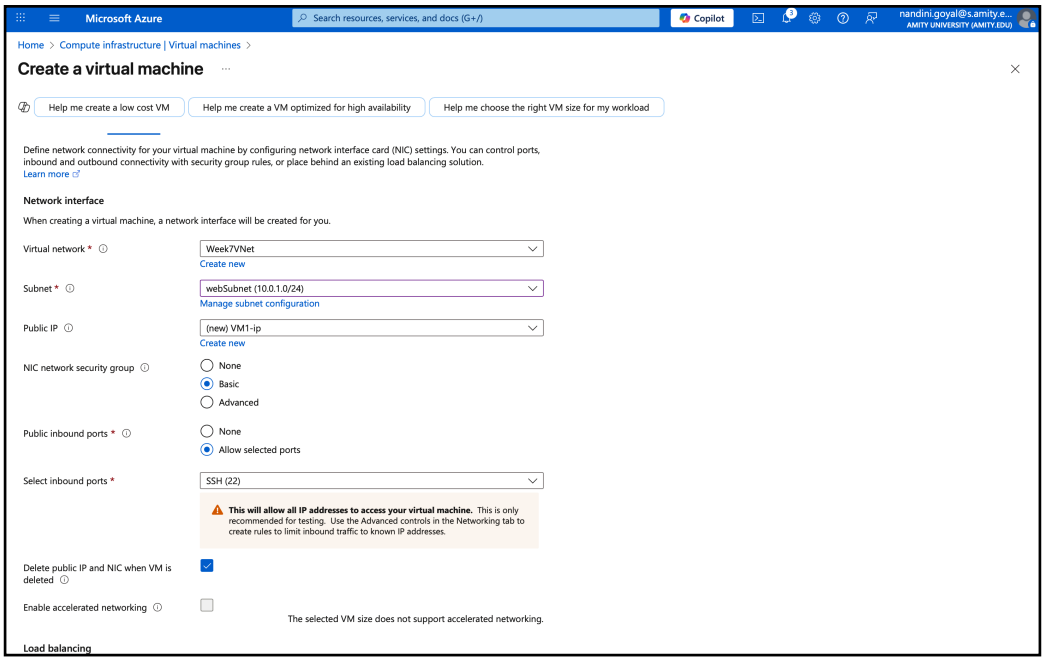


Figure 5: VM1 Networkings Tab

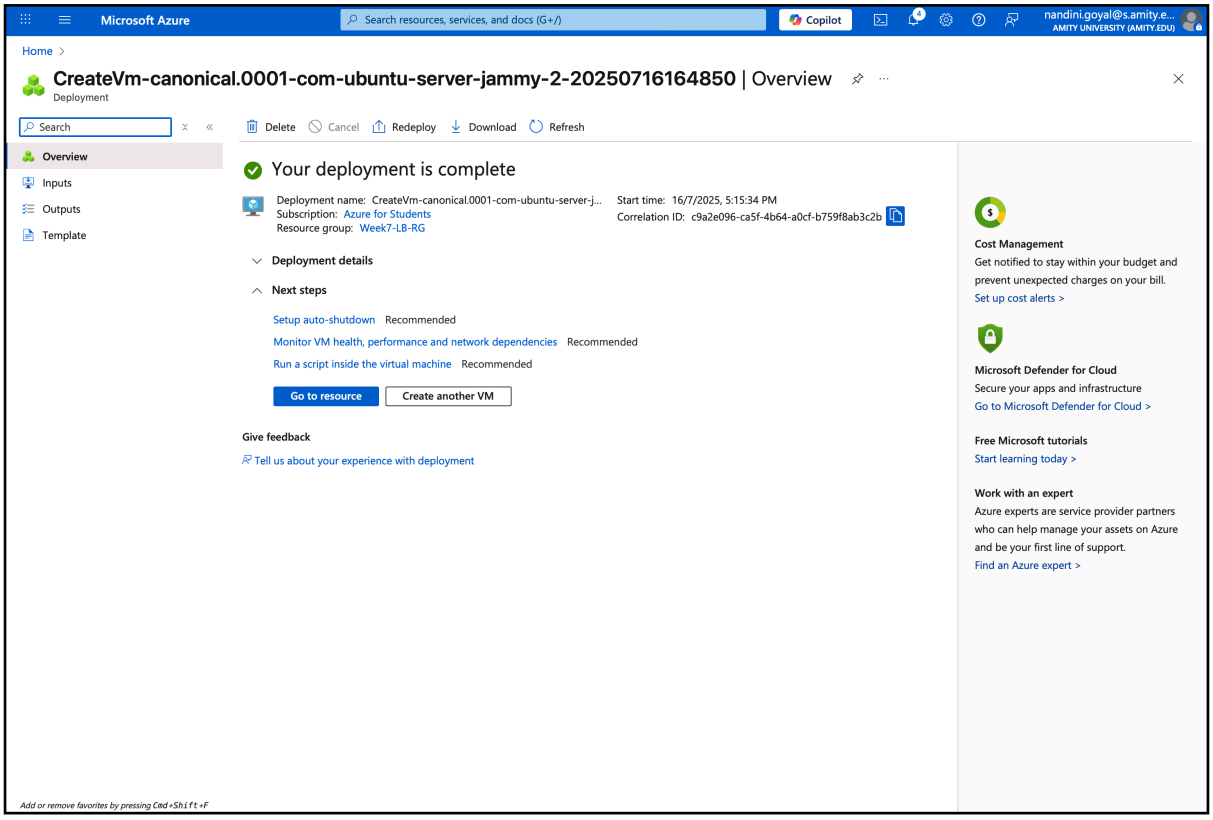


Figure 6: VM1 Deployed

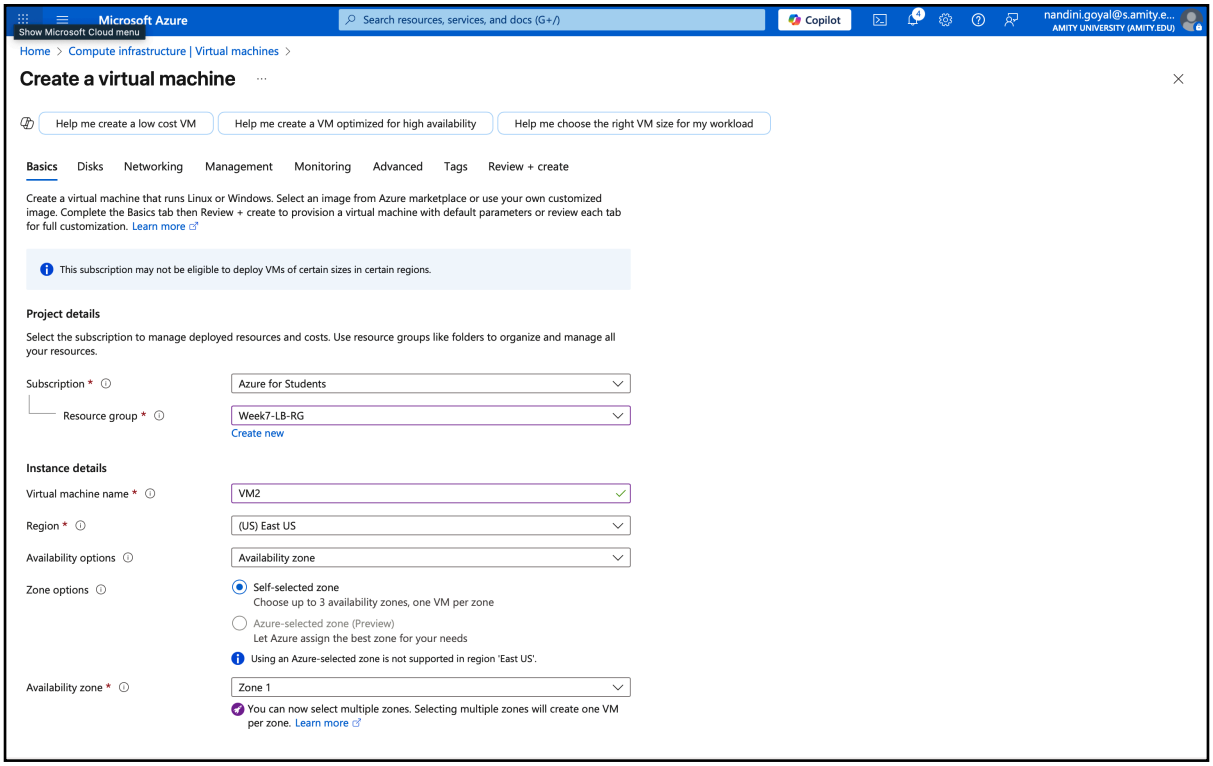


Figure 7: VM2 Basics Tab

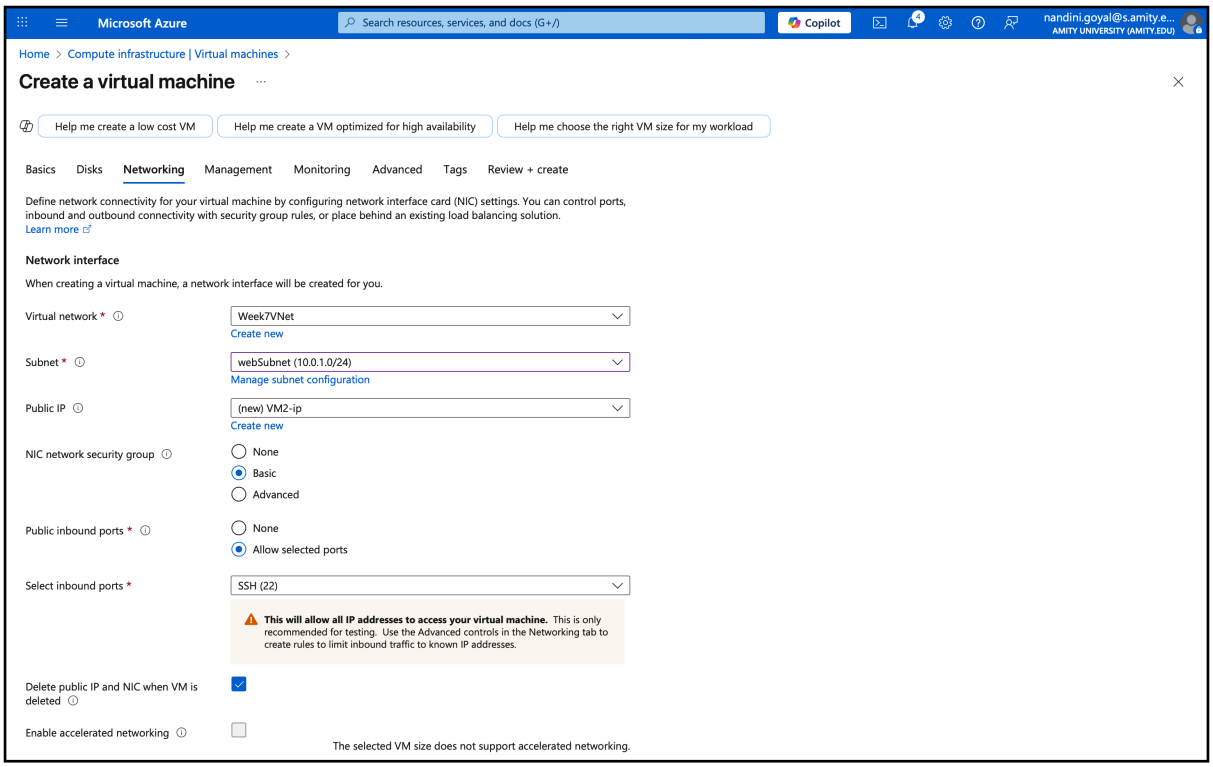


Figure 8: VM2 Networkings Tab

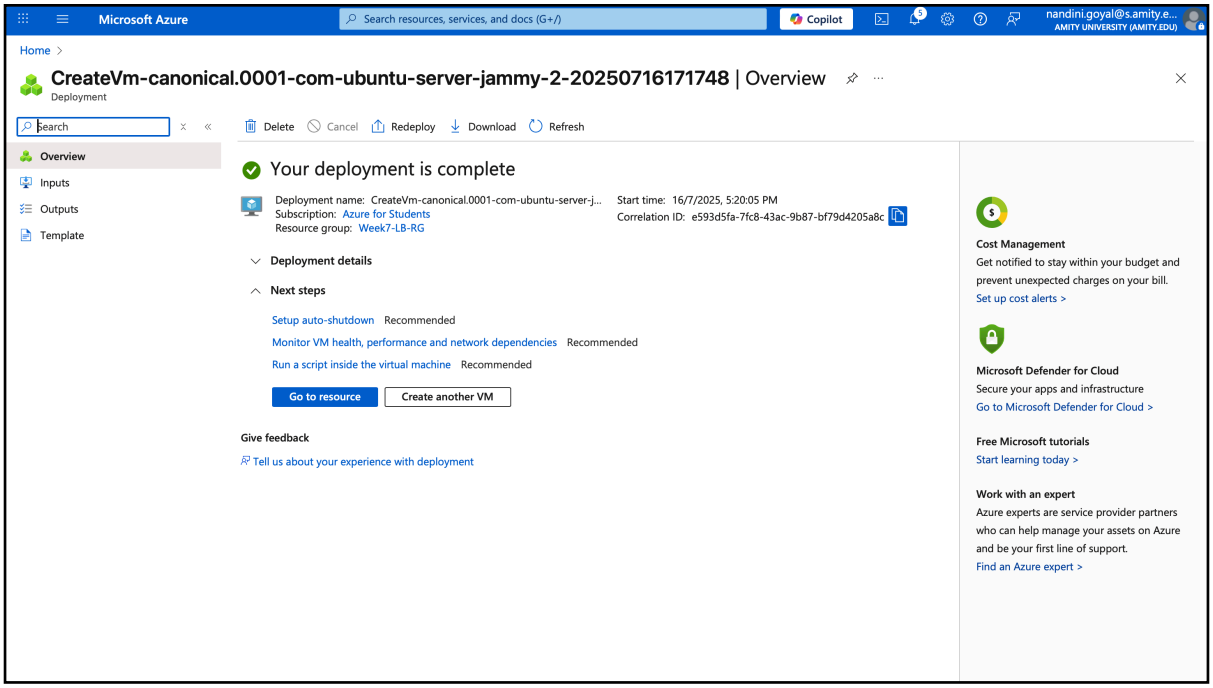


Figure 9: VM2 Deployed

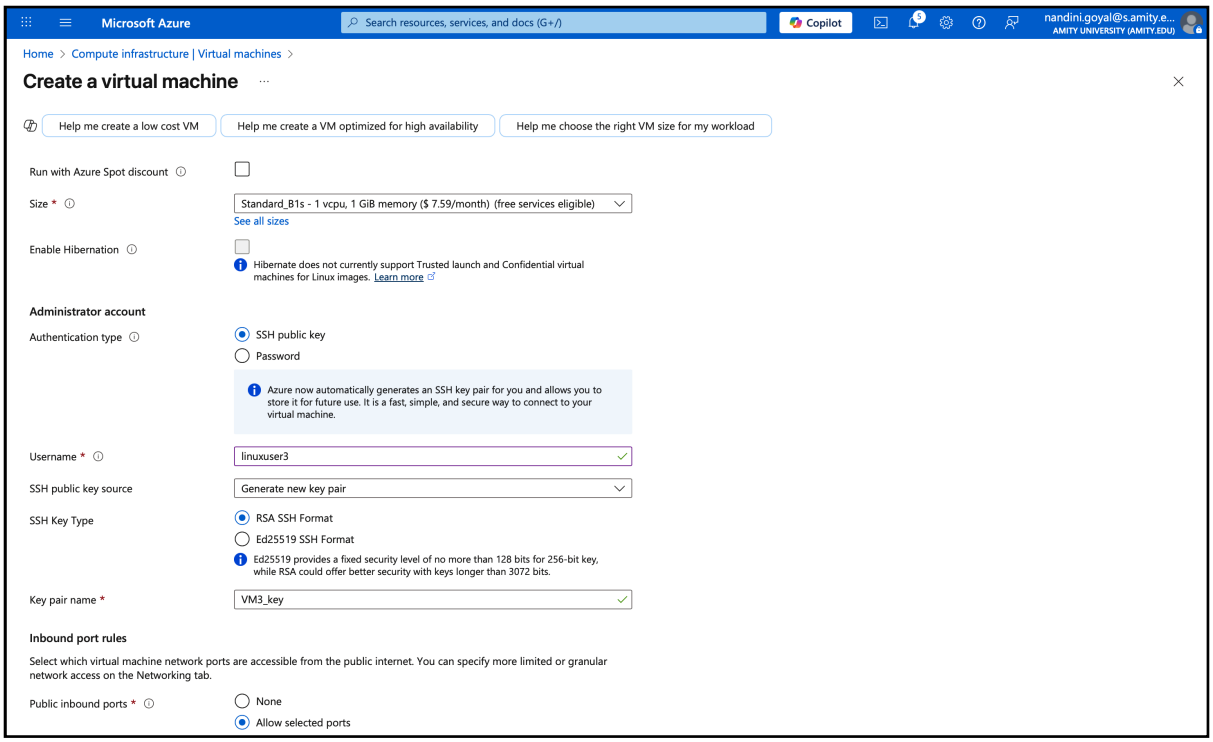


Figure 10: VM3 Basics Tab

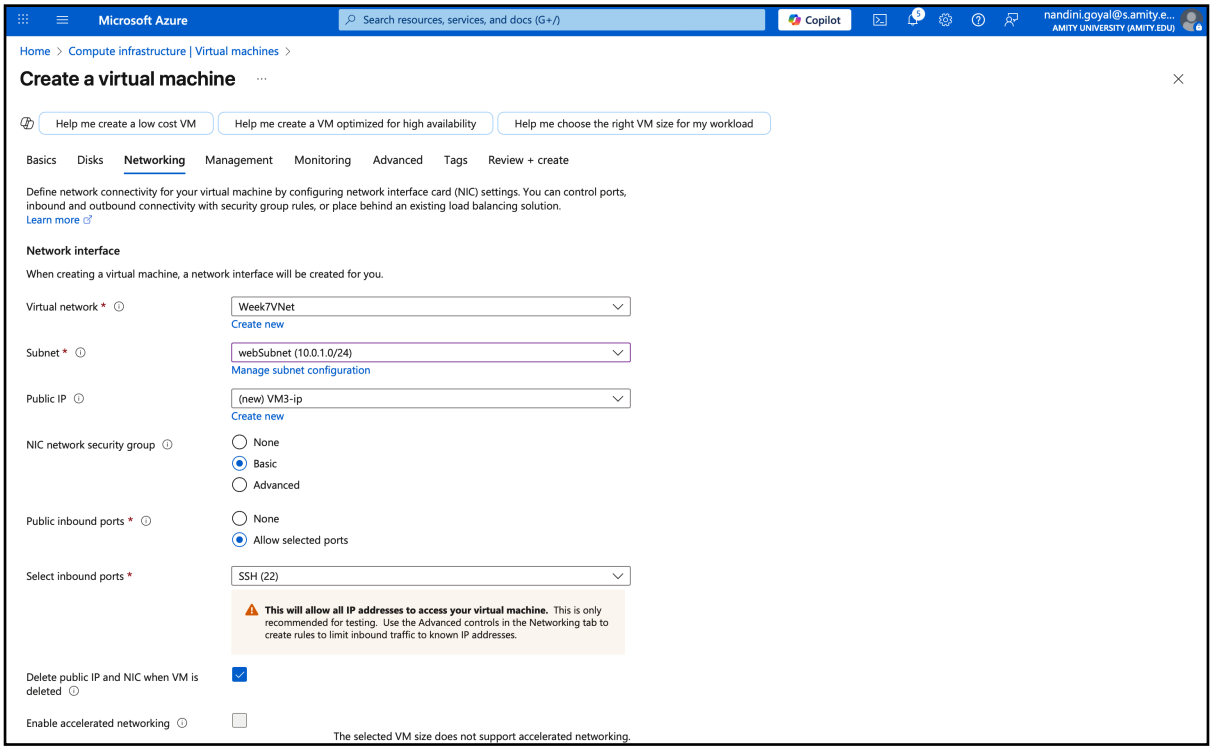


Figure 11: VM3 Networkings Tab

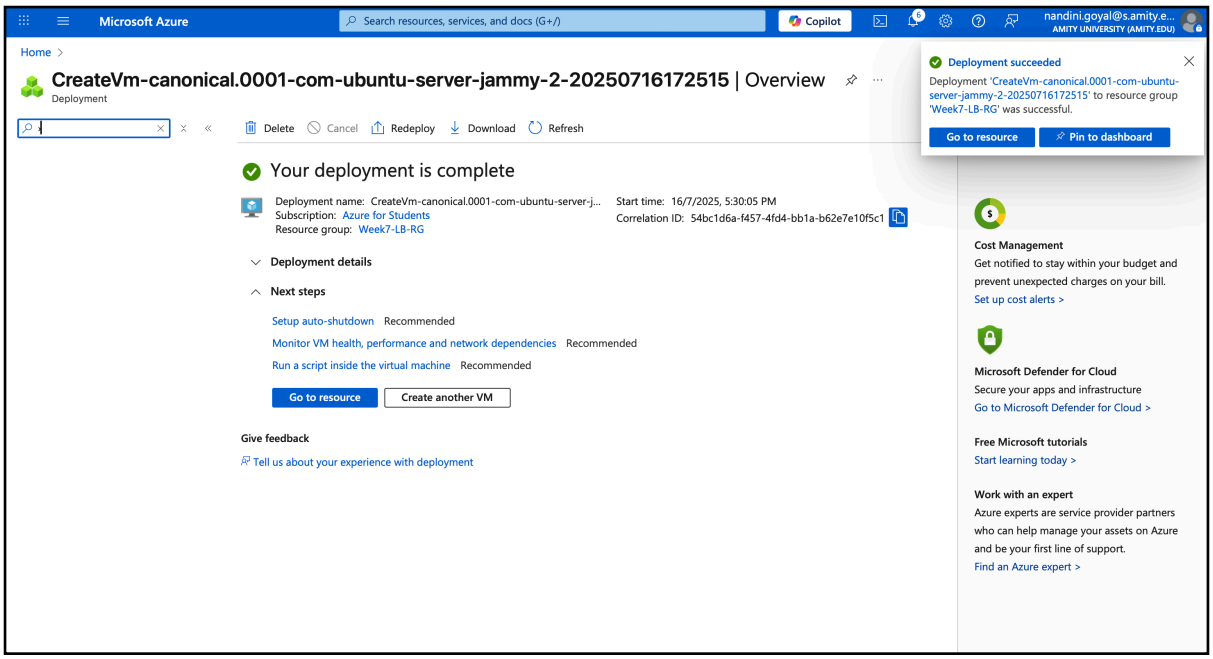


Figure 12: VM3 Deployed

4.3 Backend Pool Configuration

- Create Internal Load Balancer `Week7-Internal-LB`
- Assign a **private IP address** (e.g., `10.0.1.10`)
- Add VMs to backend pool

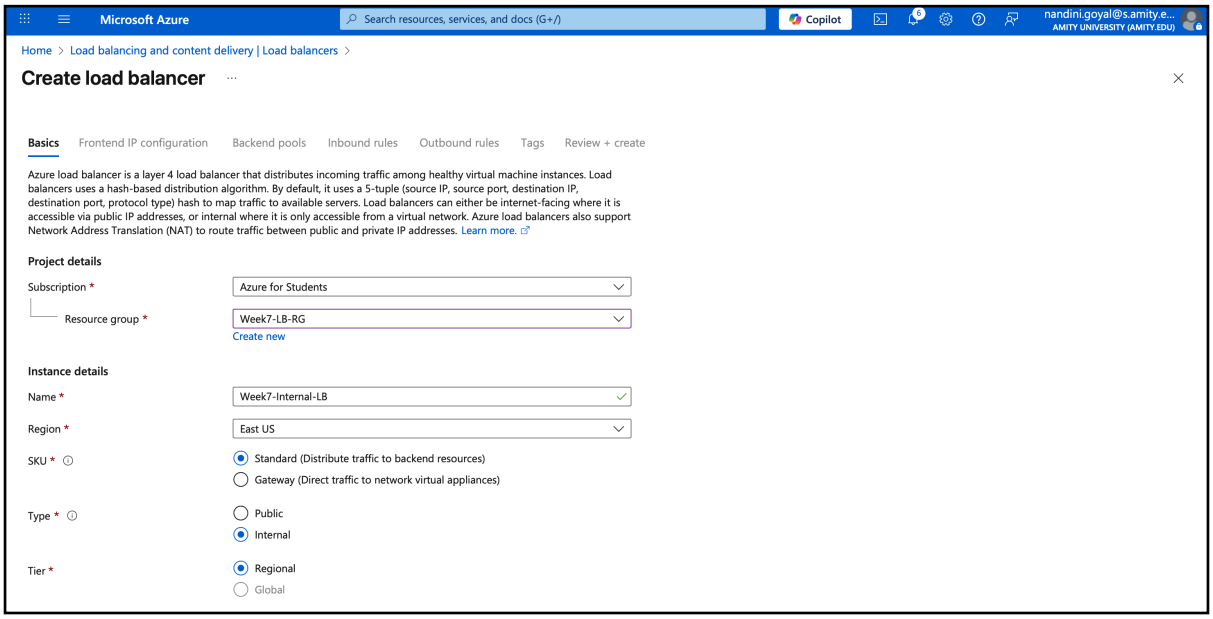


Figure 13: Internal Load Balancer Basics Tab

4.4 Internal Load Balancer Configuration

- Select IP version: IPv4
- Use default high availability ports
- Add frontend configuration with private IP

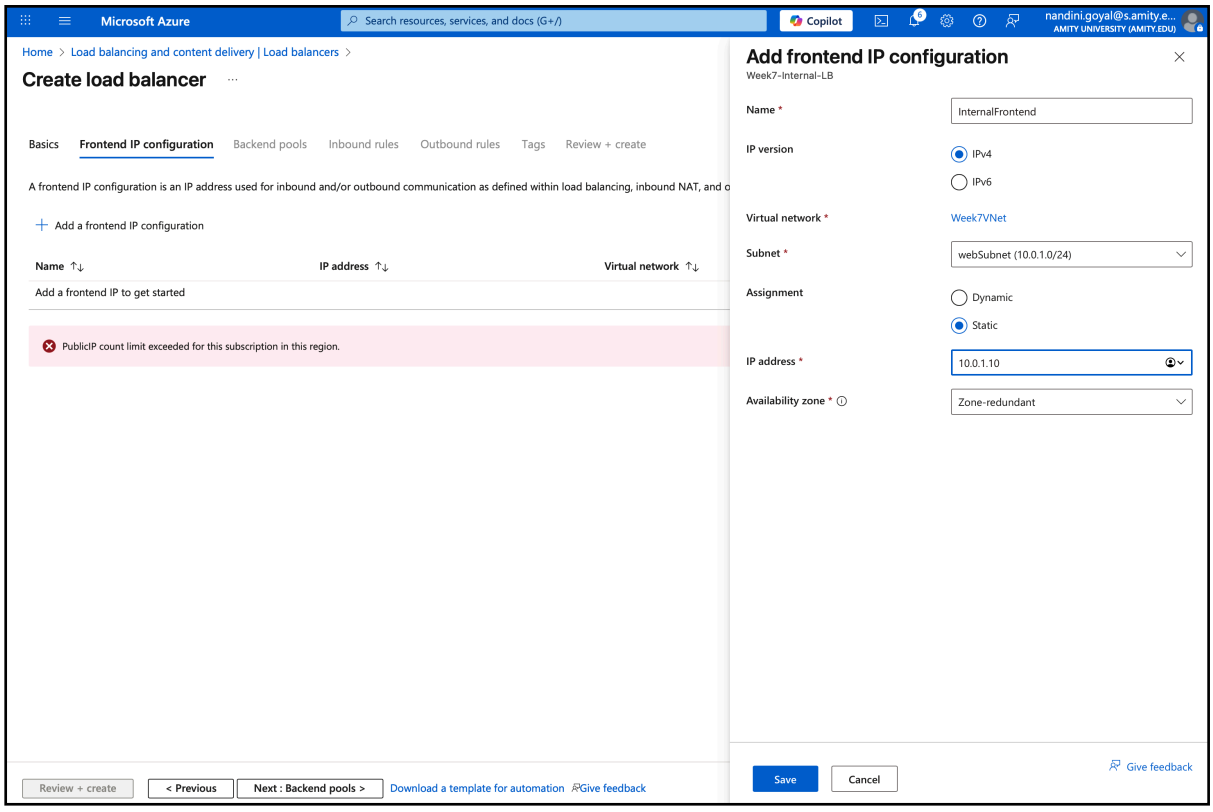


Figure 14: Internal Load Balancer - Frontend IP Configuration

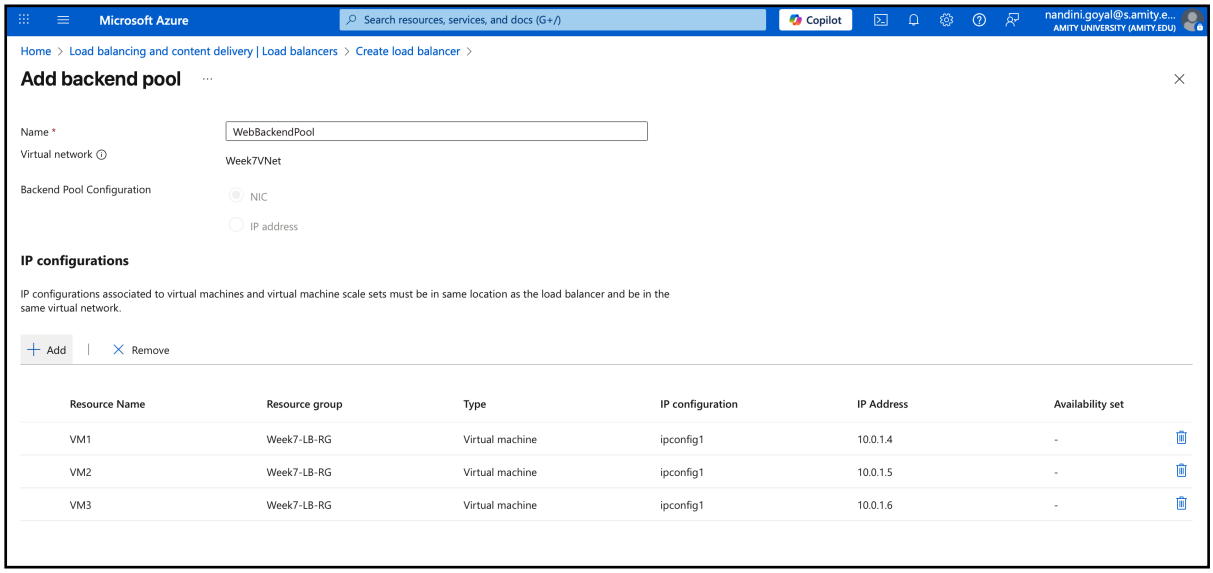


Figure 15: Internal Load Balancer - Backend pool

4.5 Health Probe Setup

- Create health probe for port 80 (TCP)
- Set interval and unhealthy threshold

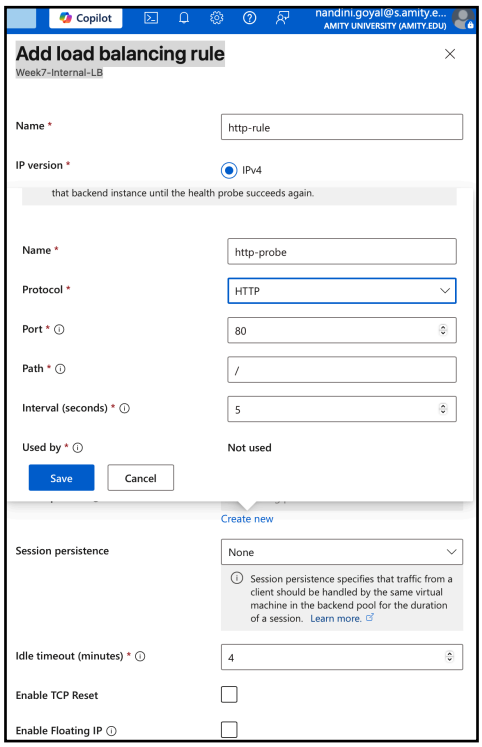


Figure 16: Health Probe Settings

4.6 Load Balancing Rule Creation

- Frontend IP: Load Balancer Private IP
- Backend port: 80
- Protocol: TCP
- Enable session persistence (Client IP or None)
- Idle timeout: default or set as required

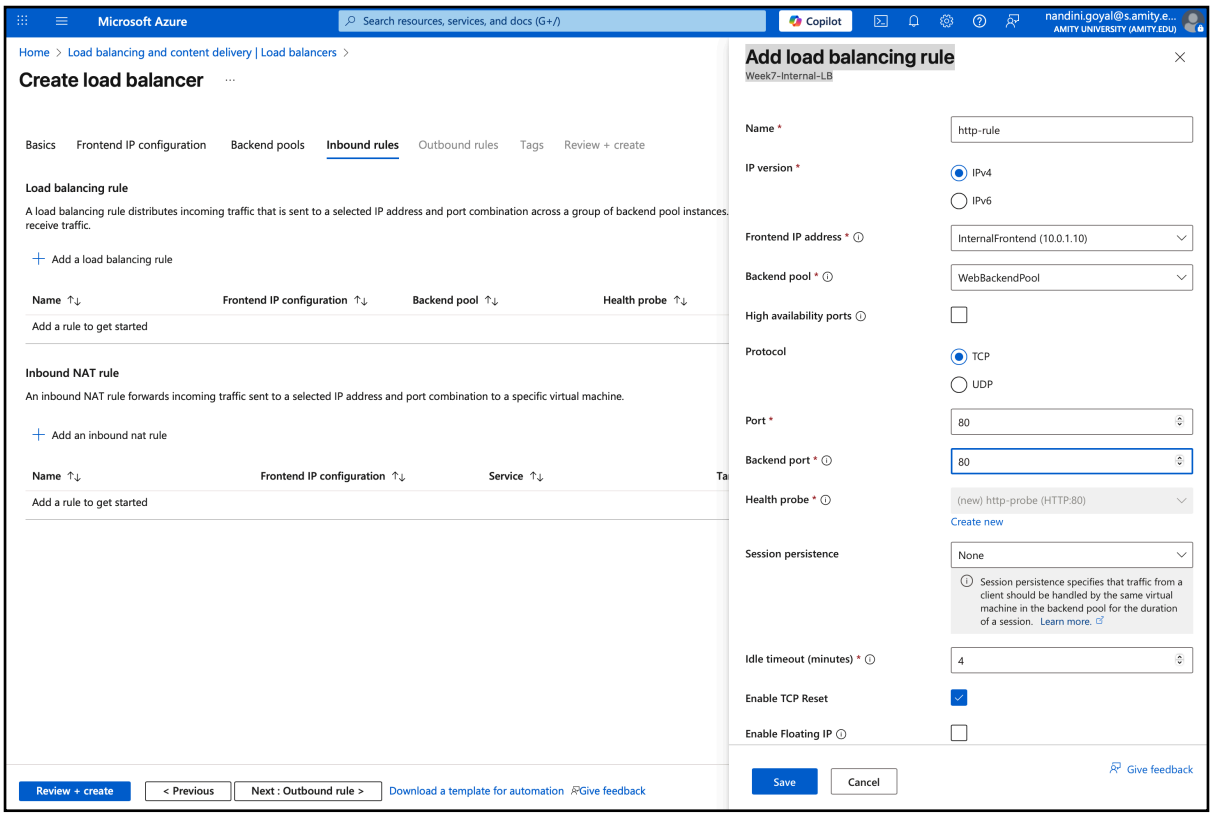


Figure 17: Inbound Rules(i)

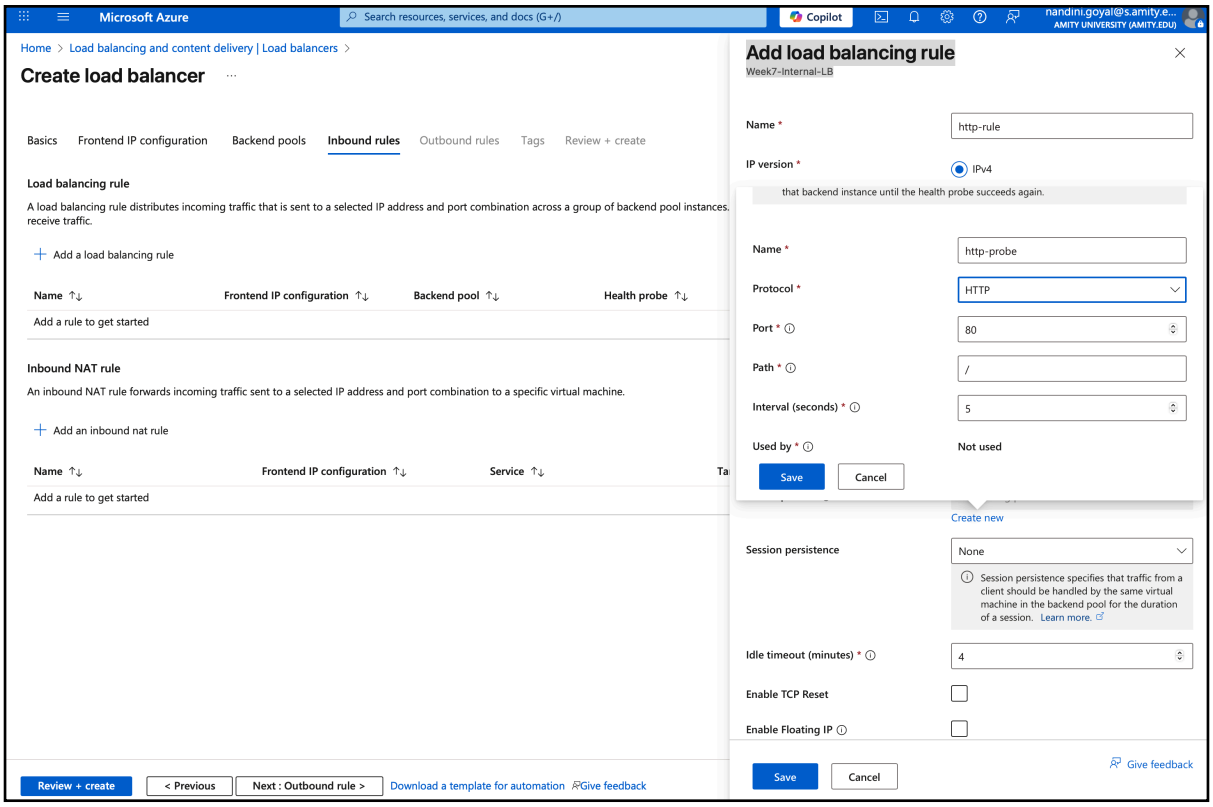


Figure 17: Inbound Rules(ii)

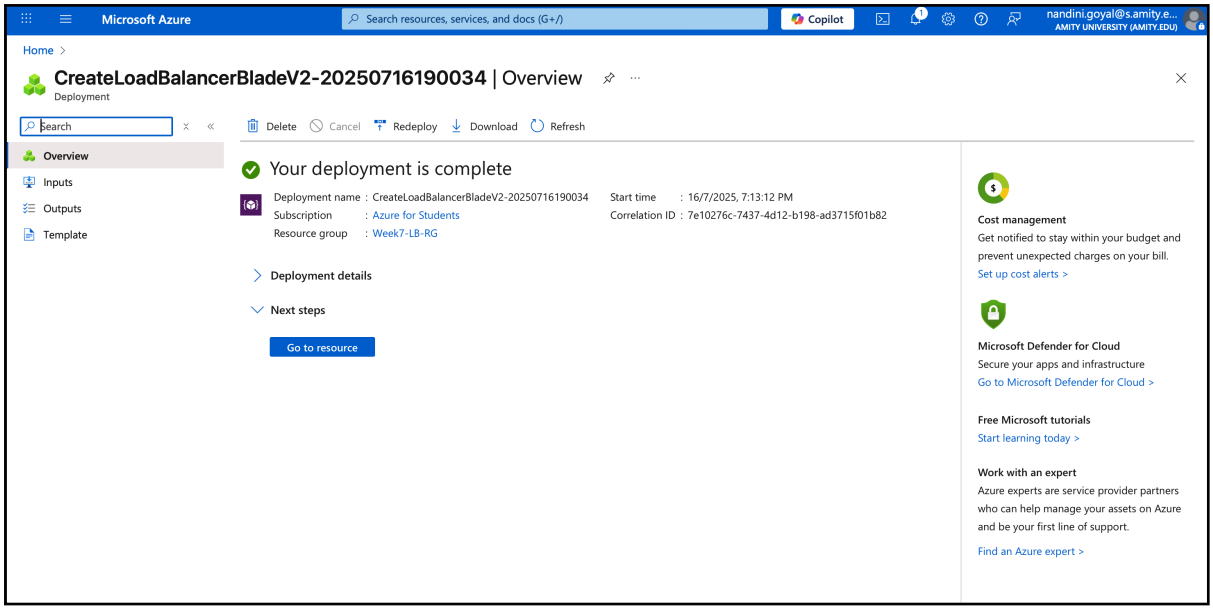


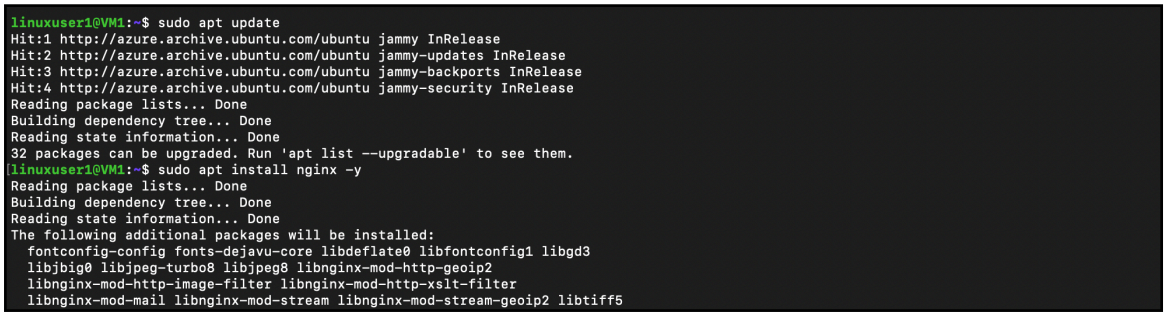
Figure 18: Internal Load Balancer Deployed

4.7 Testing Internal Load Balancing

- Install NGINX browser on each VM through terminal.
- SSH into `VM1`, `VM2`, `VM3`
- Set hostname in index.html of each VM (e.g., "VM1", "VM2"...)
- Use `curl http://10.0.1.10` command from any VM

Expected Output:

VM	Expected Response on Curl to 10.0.1.10
VM1	Randomly shows: VM1, VM2, or VM3
VM2	Randomly shows: VM1, VM2, or VM3
VM3	Randomly shows: VM1, VM2, or VM3



```
linuxuser1@VM1:~$ sudo apt update
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
32 packages can be upgraded. Run 'apt list --upgradable' to see them.
linuxuser1@VM1:~$ sudo apt install nginx -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3
  libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5
```

Figure 19: Installed NGINX on VM1

```
linuxuser2@VM2:~$ sudo apt install nginx -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3
  libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5
  libwebp7 libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3
  libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5
  libwebp7 libxpm4 nginx nginx-common nginx-core
```

Figure 20: Installed NGINX on VM2

```
linuxuser3@VM3:~$ sudo apt install nginx -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3
  libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5
  libwebp7 libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3
  libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2
```

Figure 21: Installed NGINX on VM3

```
linuxuser1@VM1:~$ curl http://10.0.1.10
VM3
linuxuser1@VM1:~$
```

Figure 22: Curl result output - 1

```
linuxuser2@VM2:~$ curl http://10.0.1.10
VM3
linuxuser2@VM2:~$
```

Figure 23: Curl result output - 2

```
linuxuser3@VM3:~$ curl http://10.0.1.10
VM2
linuxuser3@VM3:~$
```

Figure 24: Curl result output - 3

5. Issues with External Load Balancer

Attempted to set up **Public Load Balancer**, but encountered the following issue:

- The subscription **does not support standard SKU public IP**, which is **mandatory** for public load balancer configuration.

- Due to this, frontend IP could not be configured for external/public load balancing.

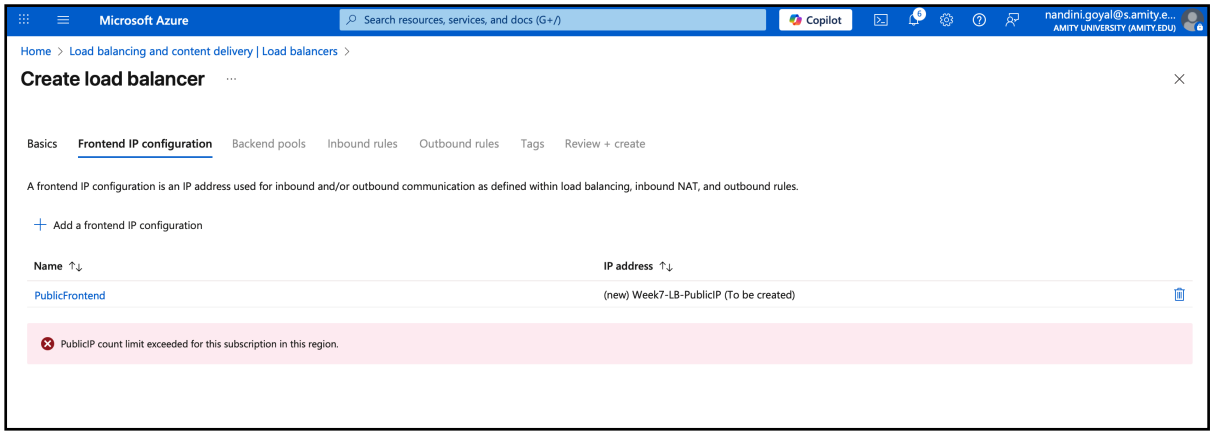


Figure 25: Public IP Limit Exceeded

6. Conclusion

The Internal Load Balancer was successfully configured using Azure VMs. It demonstrated how traffic is load-balanced across backend VMs using private IP addresses. However, the **Public Load Balancer** could not be completed due to subscription limitations. Understanding these differences and hands-on configuration helps in deploying real-world cloud architectures.

7. References

- Microsoft Azure Docs: Load Balancer
  https://learn.microsoft.com/en-us/azure/load-balancer/
- Microsoft Learn - Load Balancer Overview
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-overview
- Microsoft Learn - Configure internal load balancer for Azure VMs
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-get-started-ilb-portal
- Microsoft Learn - Load Balancer NAT Rules
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-inbound-nat-rule-overview
- Microsoft Learn - Load Balancer Backend Pool Configuration
  https://learn.microsoft.com/en-us/azure/load-balancer/backend-pool-management
- Microsoft Learn - Azure Load Balancer health probes
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview

- Microsoft Learn - Frontend IP configuration
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-front-end
- Microsoft Learn - Outbound rules for Azure Load Balancer
  https://learn.microsoft.com/en-us/azure/load-balancer/load-balancer-outbound-rules-overview