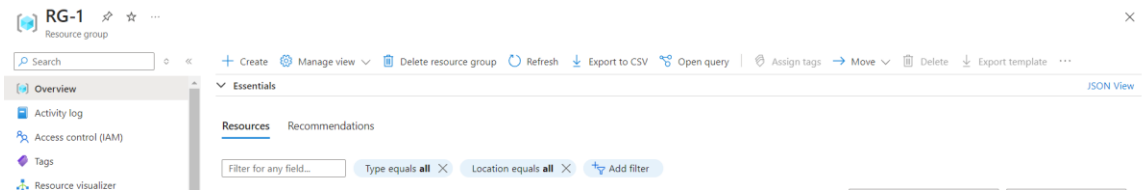
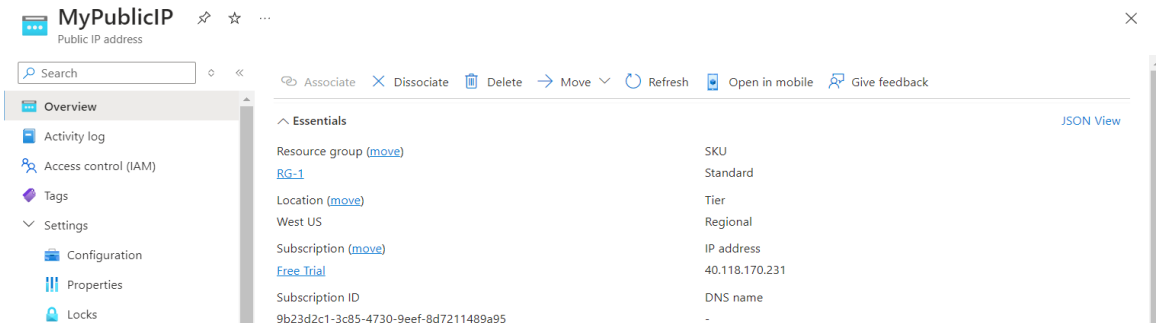
Step 1: Create a Resource Group

1. Go to the Azure Portal.
2. Search for and select "Resource groups".
3. Click on "Create".
4. Fill in the details:
    - **Resource group name**: RG-1
    - **Region**: Select "West US"
5. Click "Review + Create" and then "Create".



Step 2: Create a Public IP Address for the Load Balancer

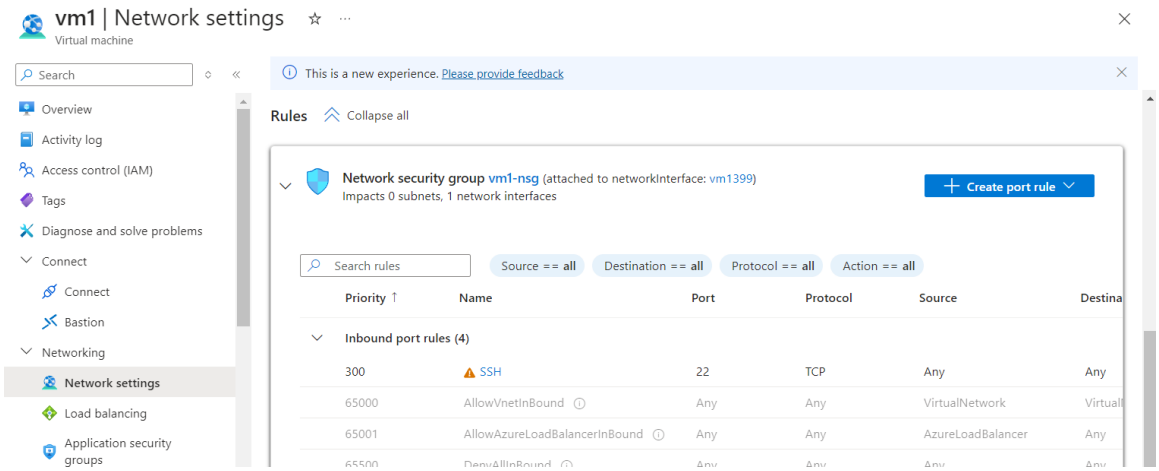
1. Go to the Azure Portal.
2. Search for and select "Public IP addresses".
3. Click on "Create".
4. Fill in the details:
    - **Name**: MyPublicIP
    - **SKU**: Standard
5. Click "Review + Create" and then "Create".



Step 3: Deploy VM1

1. Go to the Azure Portal.
2. Search for and select "Virtual machines".
3. Click on "Add" and select "Virtual machine".
4. Fill in the details:
    - **VM name**: VM1
    - **Region**: RG-1
    - **Image**: Ubuntu LTS
    - **Admin username**: azureuser
    - **Authentication type**: SSH public key
5. Click on "Review + Create" and then "Create".
6. After VM1 is created, navigate to the "VM1" blade, select "Run command", and then "RunShellScript". Enter the command to install Apache2:

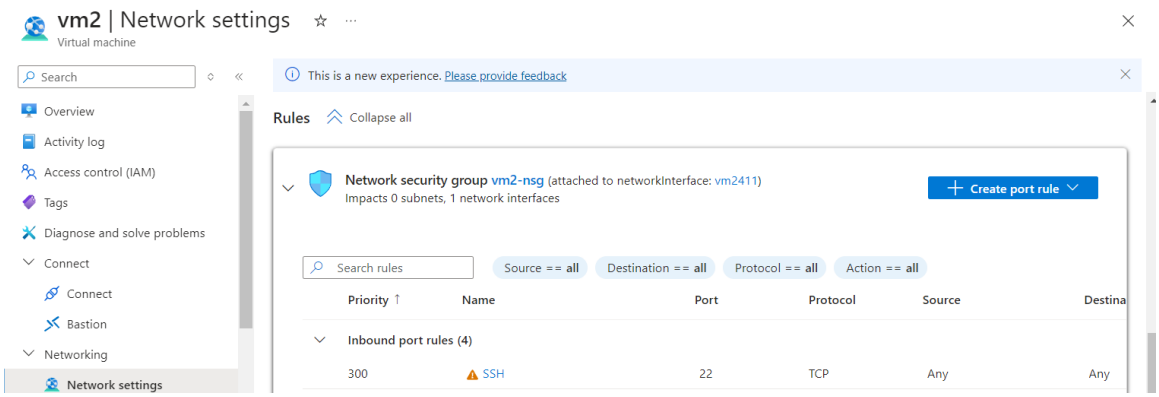
    sudo apt update && sudo apt install -y apache2



Step 4: Deploy VM2

1. Go to the Azure Portal.
2. Search for and select "Virtual machines".
3. Click on "Add" and select "Virtual machine".
4. Fill in the details:
    - **VM name**: VM2
    - **Region**: RG-1
    - **Image**: Ubuntu LTS
    - **Admin username**: azureuser
    - **Authentication type**: SSH public key
5. Click on "Review + Create" and then "Create".
6. After VM2 is created, navigate to the "VM2" blade, select "Run command", and then "RunShellScript". Enter the command to install Apache2:

sudo apt update && sudo apt install -y apache2



We do not require to allow port 80 in NSG because we are using App Gateway. It will not be accessible directly.

Step 5: Update index.html on VM1

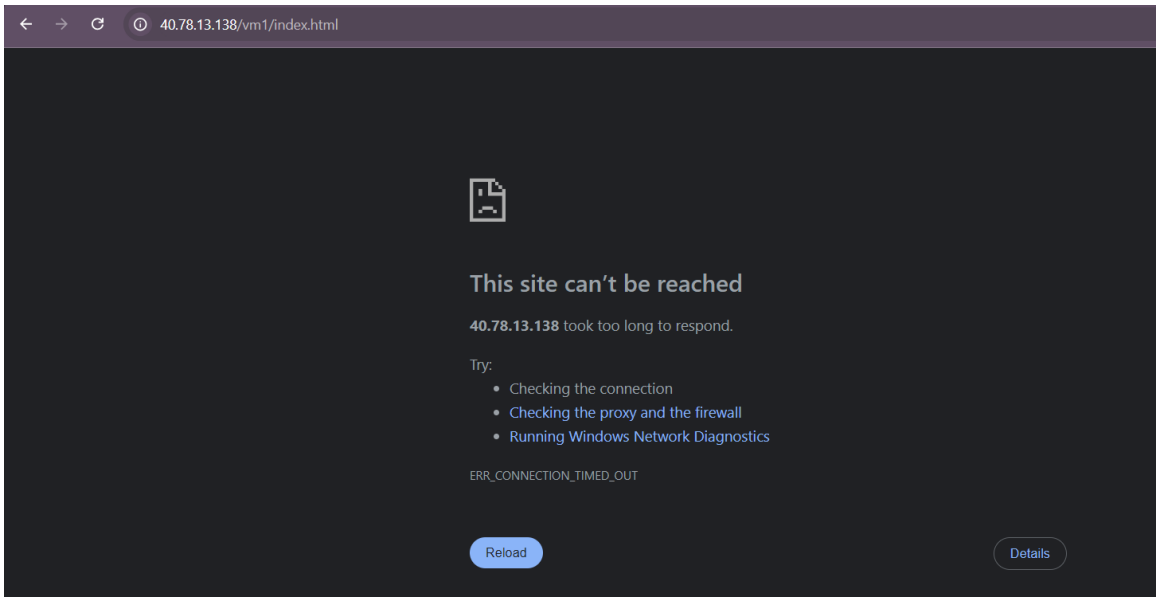
1. SSH into VM1:

    ssh azureuser@<VM1_Public_IP>

2. Change the index.html file:

echo "This is VM1" | sudo tee /var/www/html/vm1/index.html

```
root@Linux-VM:/home/azureuser# echo "This is VM1" | sudo tee /var/www/html/index.html
This is VM1
root@Linux-VM:/home/azureuser# exit
```



Step 6: Update index.html on VM2

Using Azure CLI:

1. SSH into VM2:

ssh azureuser@<VM2_Public_IP>

2. Change the index.html file:

echo "This is VM2" | sudo tee /var/www/html/vm2/index.html

```
azureuser@vm2:~$ sudo su
root@vm2:/home/azureuser# echo "This is VM2" | sudo tee /var/www/html/index.html
tee: /var/www/html/index.html: No such file or directory
This is VM2
```

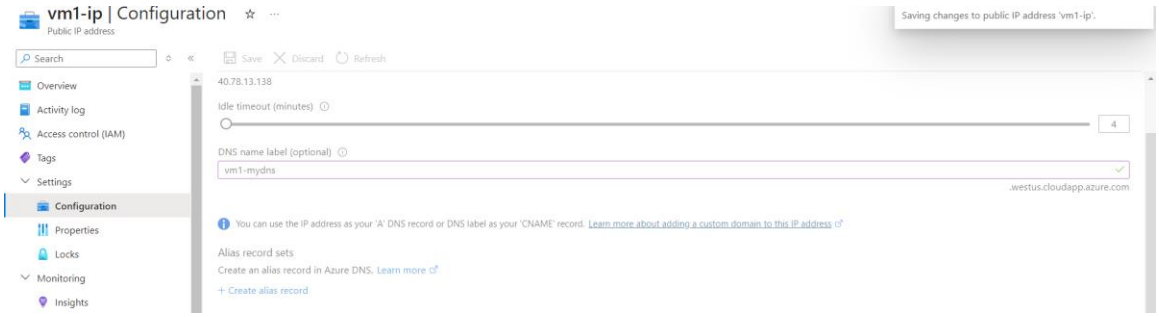
Step 7: Locate Public IPs:

- Inside the resource group, find your two VMs (myVM1 and myVM2).
- For each VM, click on the **Networking** section on the left menu.

- Under **Public IP addresses**, click on the **Public IP** resource linked to each VM.

Step 8: Configure DNS Settings

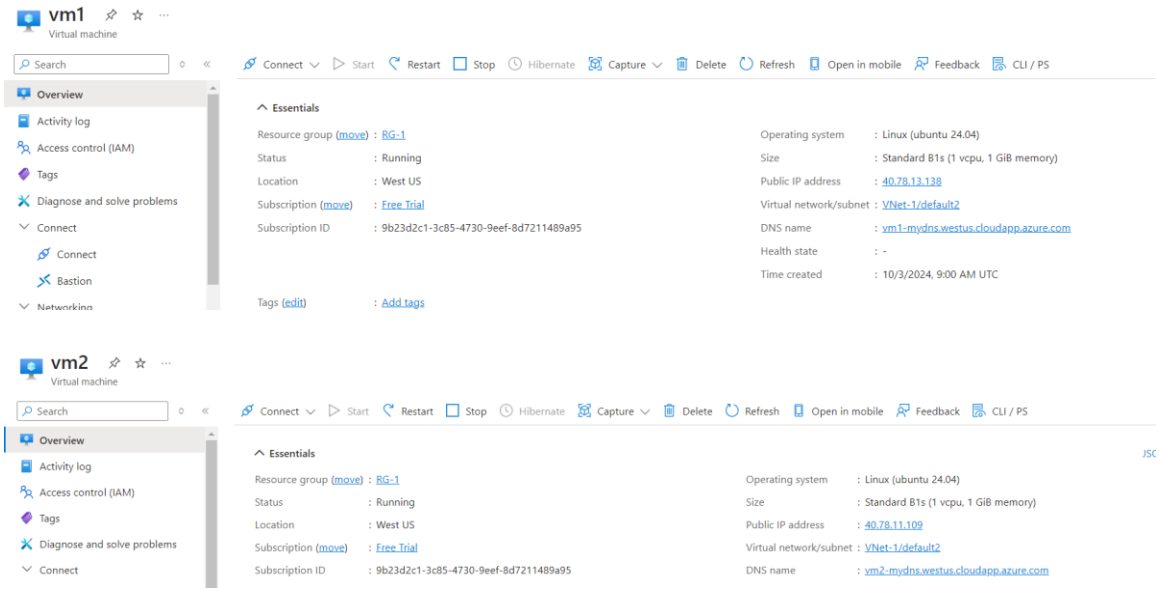
1. **Go to the Public IP Configuration**:
    - Once you're on the Public IP address page, look for the **Configuration** option under the **Settings** section in the left menu.
2. **Set the DNS Name Label**:
    - In the **Configuration** page, you'll find a section for **DNS name label**.
    - Enter a unique DNS name for each VM:
        - For VM1, you might use something like **vm1-mydns**(which would create the DNS name **vm1-mydns.westus.cloudapp.azure.com**).



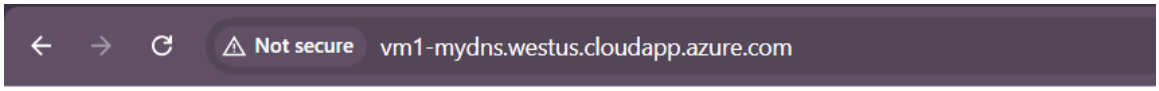
        - For VM2, you might use **vm2-mydns** (which would create the DNS name **vm2-mydns.westus.cloudapp.azure.com**).
3. **Save the Configuration**:
    - Click on the **Save** button at the top to apply the DNS name settings.

Step 9: Verify the DNS Configuration

1. **Check the Public DNS Names**:
    - After saving, navigate back to the Public IP address overview page.
    - You should see the new DNS name under **DNS Name**.

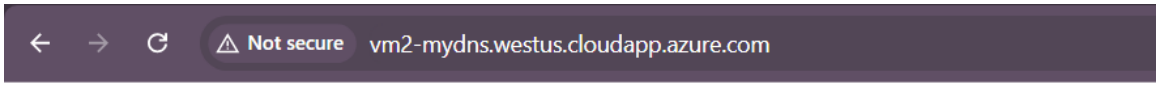


2. **Test the DNS Configuration**:
    - Open a web browser and enter the following URLs:
        - For myVM1: **vm1-mydns.westus.cloudapp.azure.com**



This is VM1

        - For myVM2: **vm2-mydns.westus.cloudapp.azure.com**



This is VM2

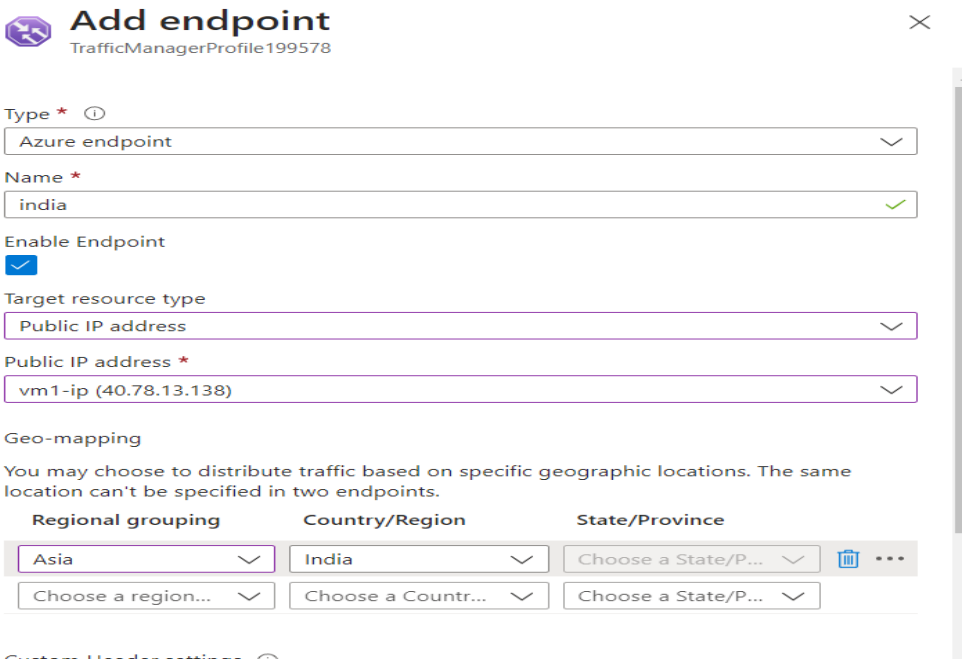
Step10: Set Up Azure Traffic Manager

1. **Create a Traffic Manager Profile**:
    - In the Azure Portal, search for **Traffic Manager profiles** and click **Create**.
    - Fill in the required details:

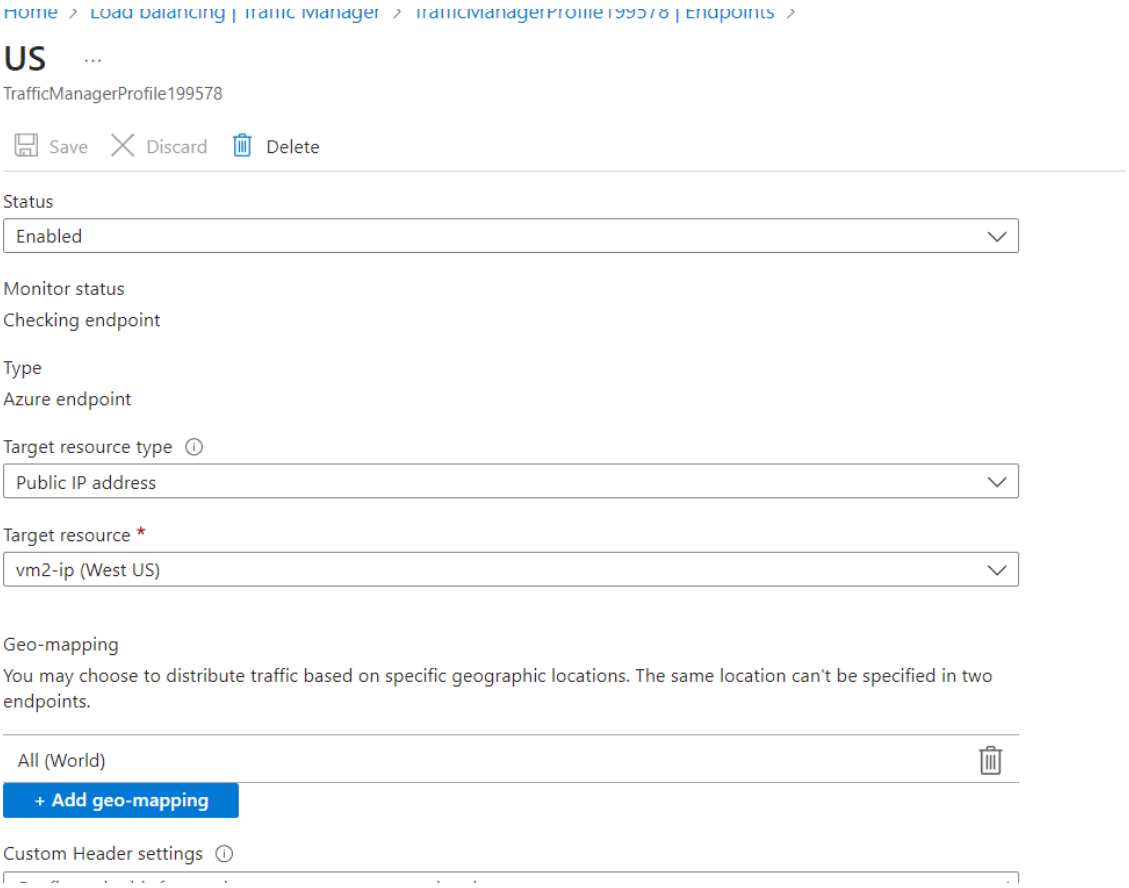
- **Name**: Give your Traffic Manager profile a name (e.g., TrafficManagerProfile199578).
- **Routing Method**: Select **Geographic** for geographic load balancing.
- **Resource Group**: Choose the same resource group as your VMs or create a new one.
- **Resource Group**: Select the desired location.

- Click **Review + Create**, then click **Create**.

2. **Add Endpoints to the Traffic Manager**:
   - After creating the Traffic Manager profile, navigate to it.
   - Click on **Endpoints** > **Add**.
   - For the first endpoint (VM1):
     - **Type**: Select **External endpoint**.
     - **Name**: Enter a name (india).
     - **URL**: Enter the public IP address
     - **Region** as Asia and country as **india**
     - Click **Add**.

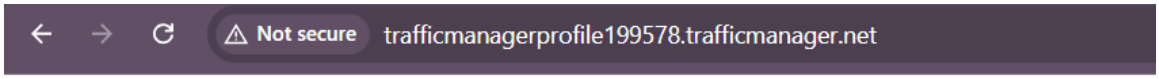


- Repeat the above steps to add VM2:
  - **Type**: Select **External endpoint**.
  - **Name**: Enter a name US.
  - **URL**: Enter the public IP address
  - **Region** as All World
  - Click **Add**.



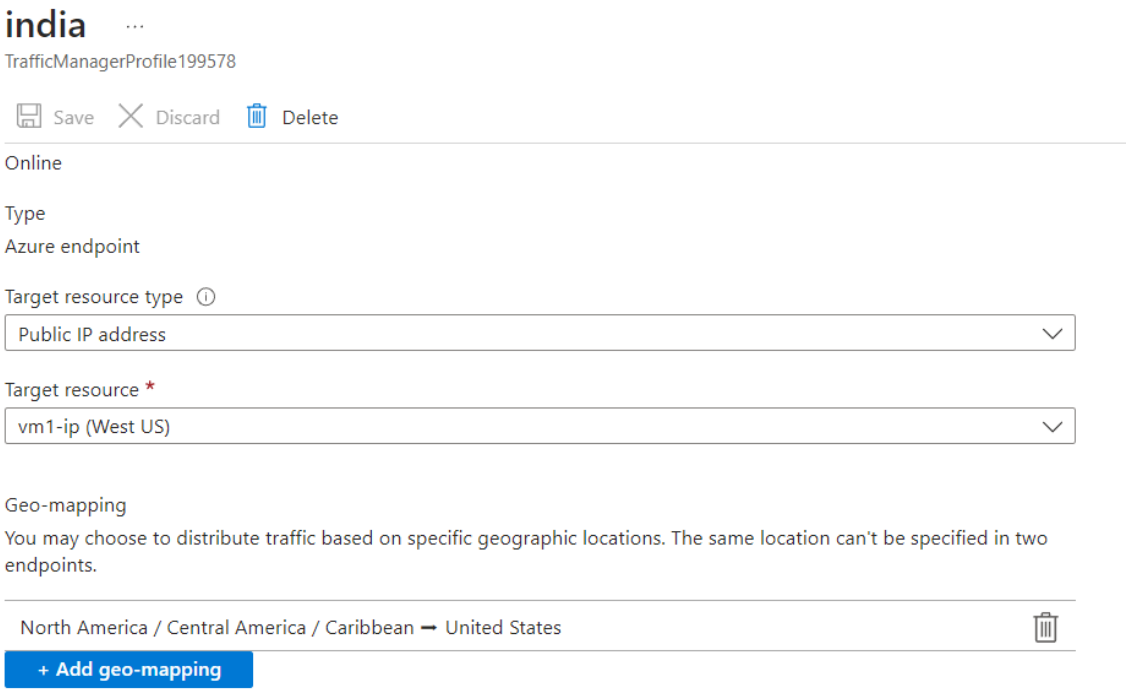
Step 11: Test the Configuration

1. **Obtain the Traffic Manager DNS Name**:
   - After the configuration is complete, you will get a DNS name for your Traffic Manager profile (**http://trafficmanagerprofile199578.trafficmanager.net**).
2. **Test the Traffic Manager**:
   - Open a web browser and navigate to the Traffic Manager DNS name (**http://trafficmanagerprofile199578.trafficmanager.net**).

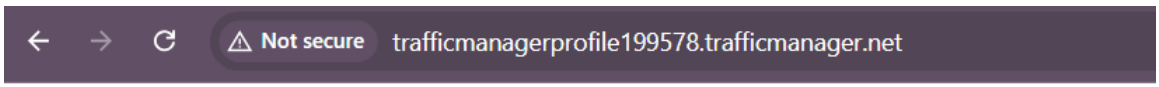


This is VM1

Now I changed India endpoint as North America



so I am getting response from VM2 because in vm2 Geo-mapping is all world



This is VM2