Step-by-Step Solution for Azure Deployment

1. Set Up Azure Infrastructure

1.1. Create Resource Groups

- Create two resource groups for the two regions:
    - **Central US**: ResourceGroupCentralUS
    - **West US**: ResourceGroupWestUS

Using Azure CLI

az group create --name ResourceGroupCentralUS --location centralus

```
new [ ~ ]$ az group create --name ResourceGroupCentralUS --location centralus
{
  "id": "/subscriptions/9b23d2c1-3c85-4730-9eef-8d7211489a95/resourceGroups/ResourceGroupCentralUS",
  "location": "centralus",
  "managedBy": null,
  "name": "ResourceGroupCentralUS",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
```

az group create --name ResourceGroupWestUS --location westus

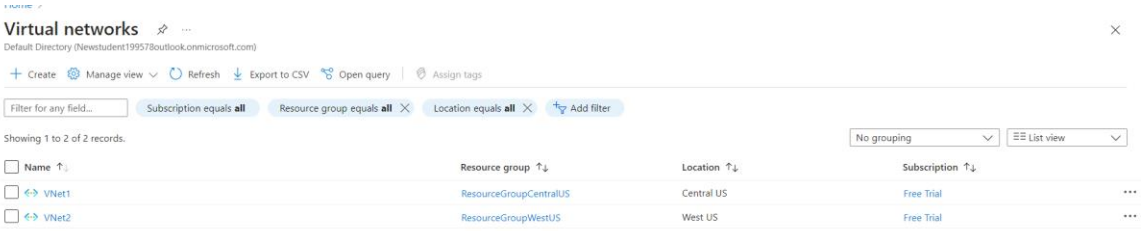
```
new [ ~ ]$ az group create --name ResourceGroupWestUS --location westus
{
  "id": "/subscriptions/9b23d2c1-3c85-4730-9eef-8d7211489a95/resourceGroups/ResourceGroupWestUS",
  "location": "westus",
  "managedBy": null,
  "name": "ResourceGroupWestUS",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

1.2. Create Virtual Networks

- Create two virtual networks, one for each region:

az network vnet create --resource-group ResourceGroupCentralUS --name VNet1 --subnet-name Subnet1

az network vnet create --resource-group ResourceGroupWestUS --name VNet2 --subnet-name Subnet2

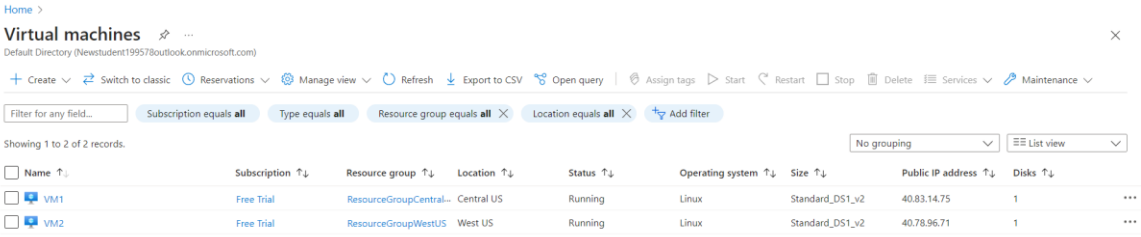


1.3. Create Virtual Machines

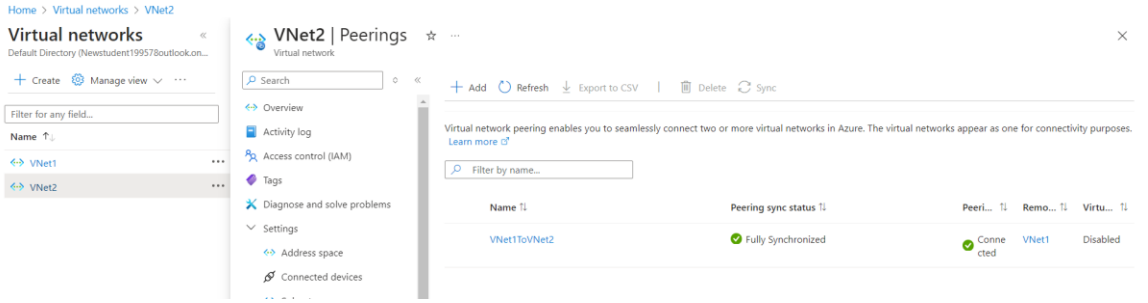
- Deploy VM1 in Central US and VM2 in West US.

az vm create --resource-group ResourceGroupCentralUS --name VM1 --image Ubuntu2204 --vnet-name VNet1 --subnet Subnet1 --admin-username azureuser --generate-ssh-keys

az vm create --resource-group ResourceGroupWestUS --name VM2 --image Ubuntu2204 --vnet-name VNet2 --subnet Subnet2 --admin-username azureuser --generate-ssh-keys



1.4. Create V-net Peering



2. Configure Storage Account

2.1. Create a Storage Account

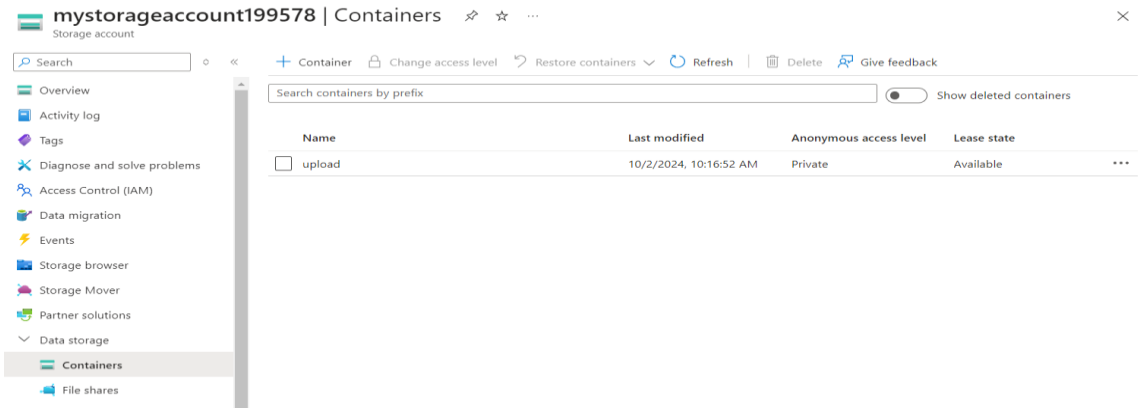
- Create a storage account in Central US for hosting the error.html file.

az storage account create --name mystorageaccount199578 --resource-group ResourceGroupCentralUS --location centralus --sku Standard_LRS

2.2. Create a Blob Container

- Create a container named upload.

az storage container create --name upload --account-name mystorageaccount199578



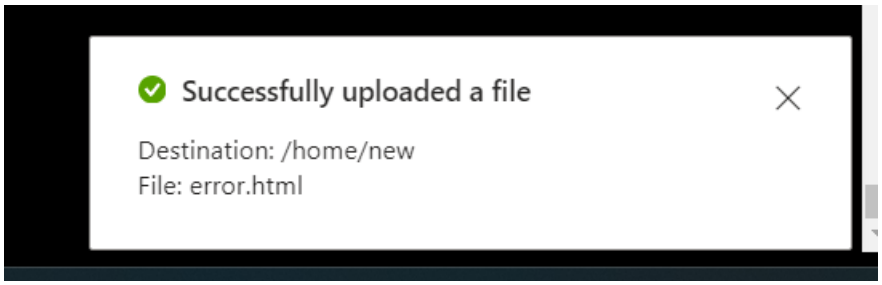
2.3. Enable Static Website Hosting

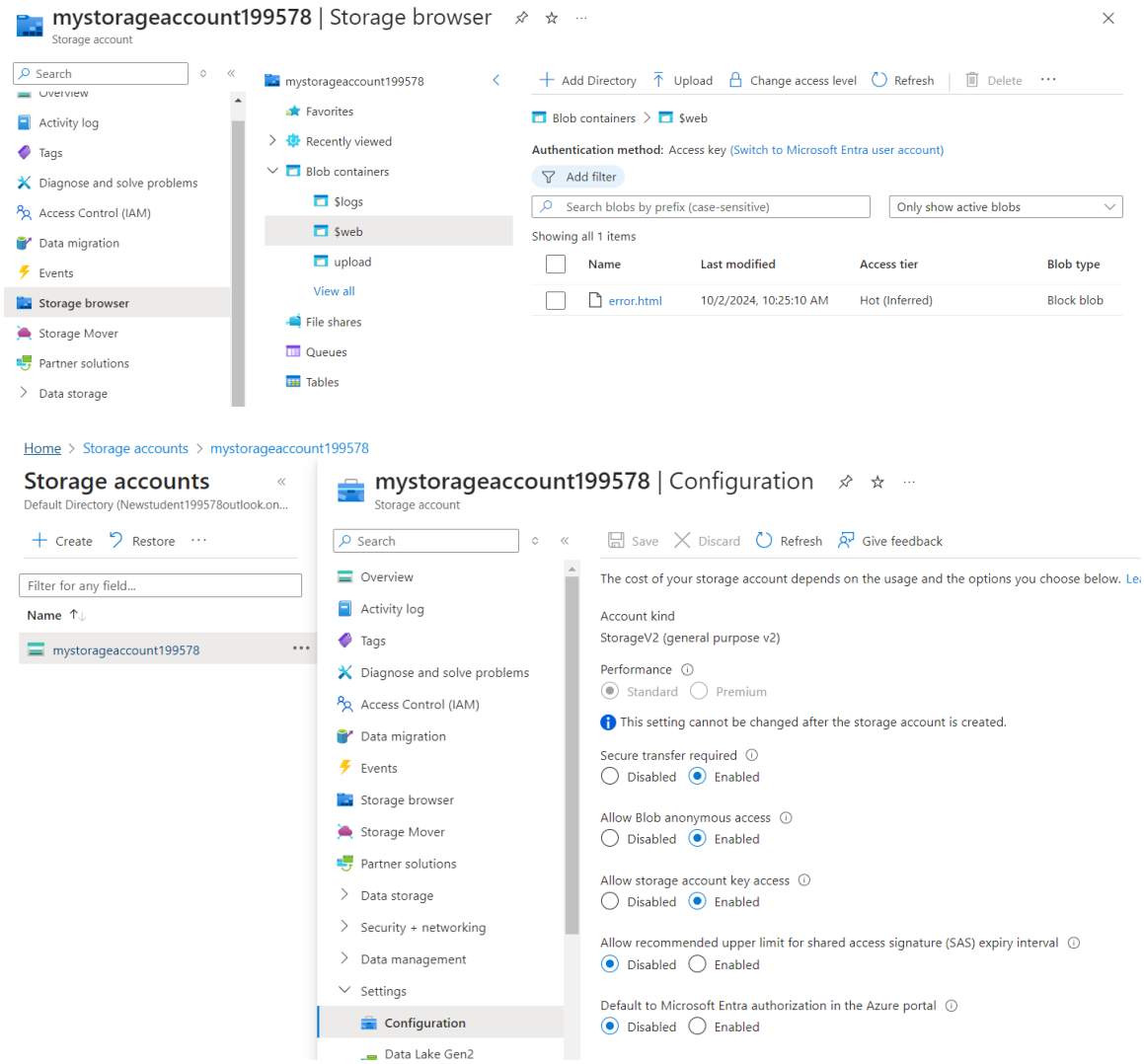
- Enable static website hosting and upload the error.html.

az storage blob service-properties update --account-name mystorageaccount199578 --static-website true

az storage blob service-properties update --account-name mystorageaccount199578 --static-website --index-document index.html --404-document error.html

az storage blob upload --account-name mystorageaccount199578 --container-name \$web --name error.html --file /home/new/error.html





3. SSH into Each VM, Clone the Repository and **Run Deployment Scripts**

On VM1

git clone https://github.com/azcloudberg/azproject.git

cd azproject

./vm1.sh

```
root@VM1:/home/azureuser# git clone https://github.com/azcloudberg/azproject.git
cd azproject
Cloning into 'azproject'...
remote: Enumerating objects: 229, done.
remote: Counting objects: 100% (26/26), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 229 (delta 21), reused 14 (delta 14), pack-reused 203 (from 1)
Receiving objects: 100% (229/229), 52.16 KiB | 1.58 MiB/s, done.
Resolving deltas: 100% (108/108), done.
root@VM1:/home/azureuser/azproject# ./vm1.sh
Rules updated
Rules updated (v6)
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
```

On VM2

git clone https://github.com/azcloudberg/azproject.git

cd azproject

./vm2.sh

```
root@VM2:/home/azureuser# git clone https://github.com/azcloudberg/azproject.git
cd azproject
./vm2.sh
Cloning into 'azproject'...
remote: Enumerating objects: 229, done.
```

5. Configure the Application

5.1. Edit Configuration Files

- On VM1, open the config.py file and update the storage account details.

nano config.py

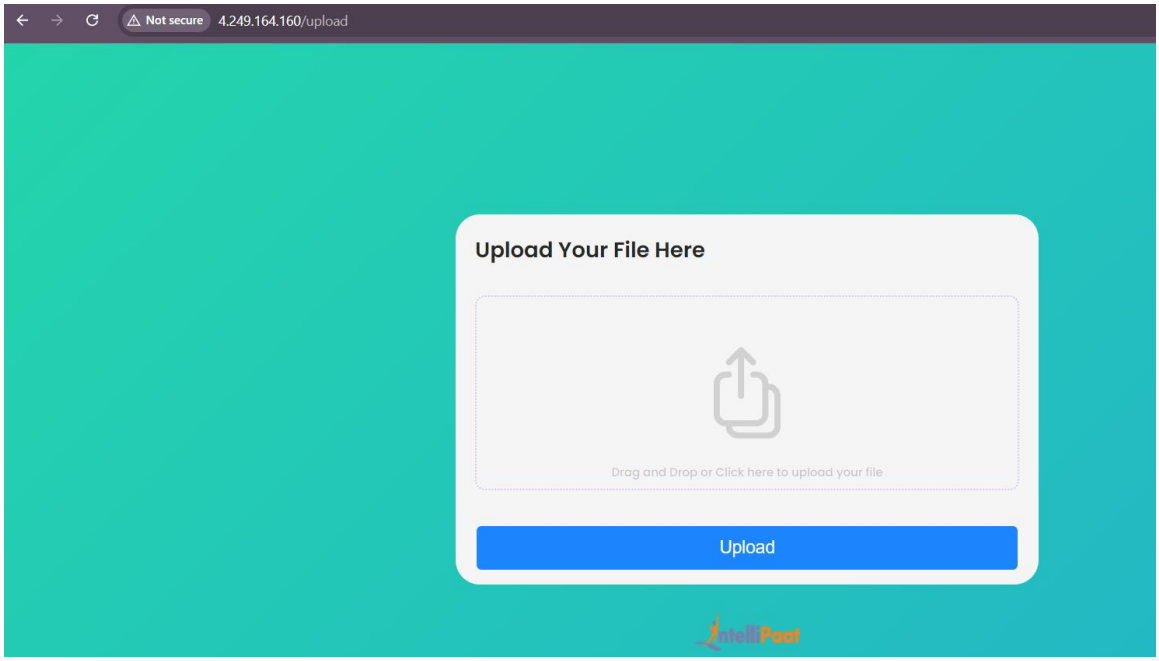
Update the storage account information

5.2. Run the Application

On VM1

sudo python3 app.py

```
root@VM1:/home/azureuser/azproject# sudo python3 app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production dep
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:80
 * Running on http://10.0.0.4:80
Press CTRL+C to quit
```



6. Configure Application Gateway

6.1. Create Application Gateways

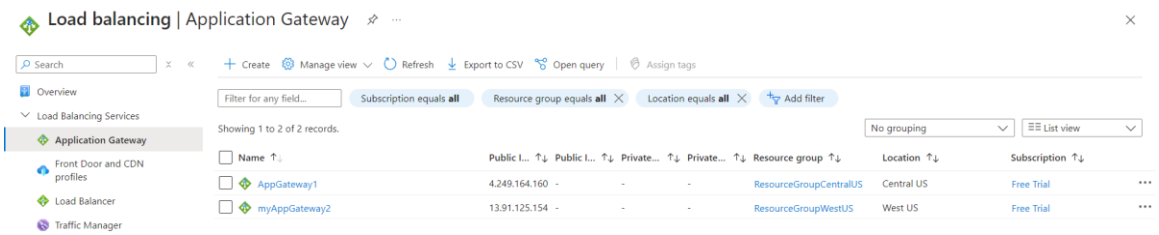
- Create Application Gateways in both regions and set up the routing rules.

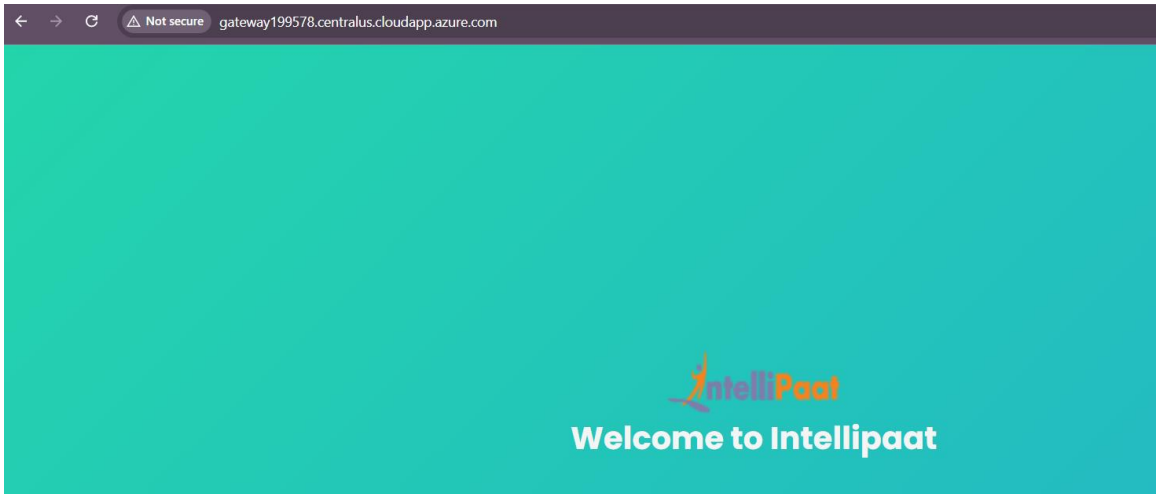
Application Gateway for Central US

az network application-gateway create --name myAppGateway2 --location westus --resource-group ResourceGroupWestUS --capacity 2 --sku Standard_v2 --public-ip-address MyAppGateway2PublicIp --vnet-name VNet1 --subnet default2 --servers "<private ip of vm>" --priority 100

Application Gateway for West US

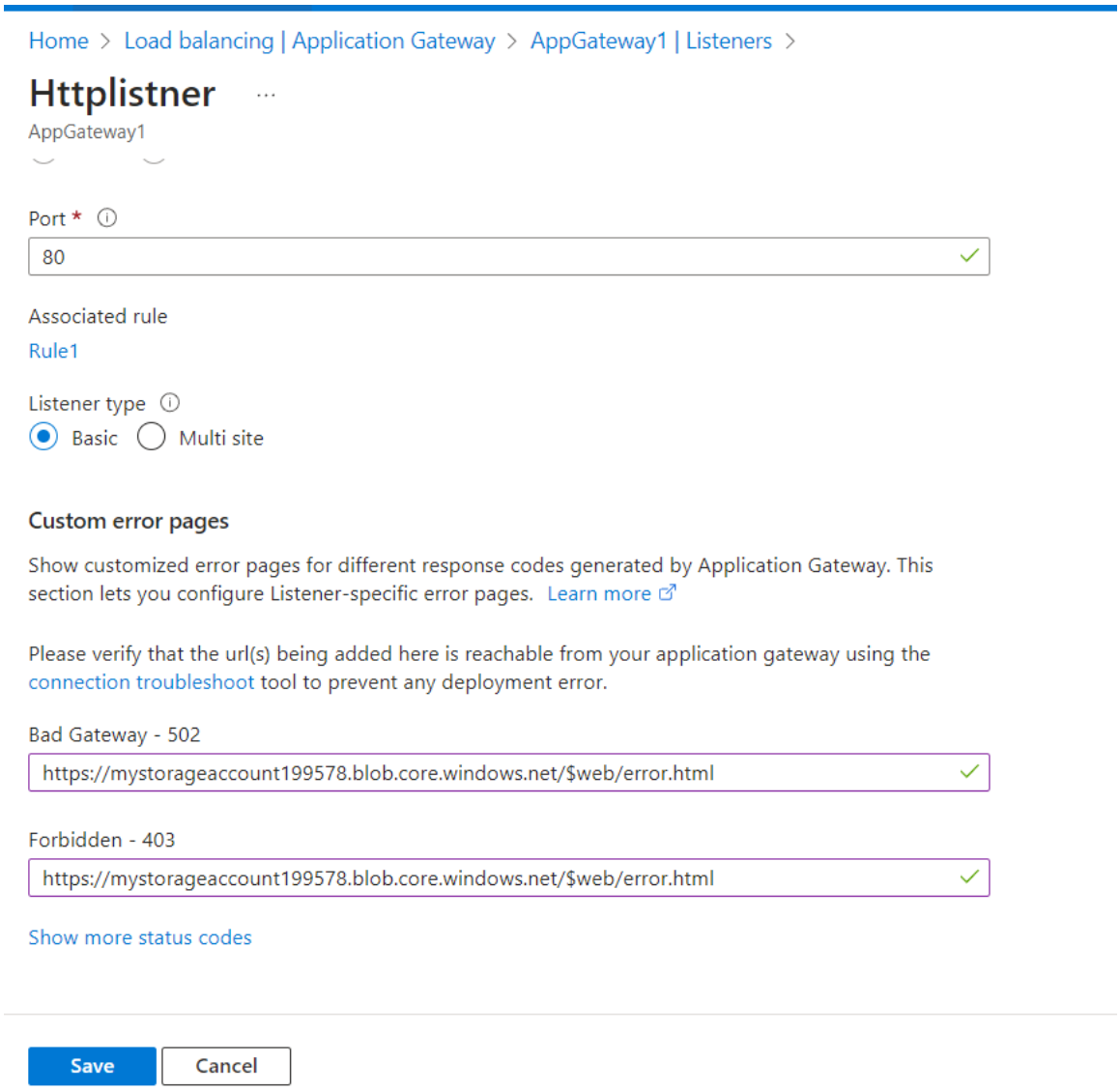
az network application-gateway create --name myAppGateway2 --location westus --resource-group ResourceGroupWestUS --capacity 2 --sku Standard_v2 --public-ip-address MyAppGateway2PublicIp --vnet-name VNet2 --subnet default --servers ""<private ip of vm>" --priority 100





6.2. Configure HTTP Settings and Routing Rules

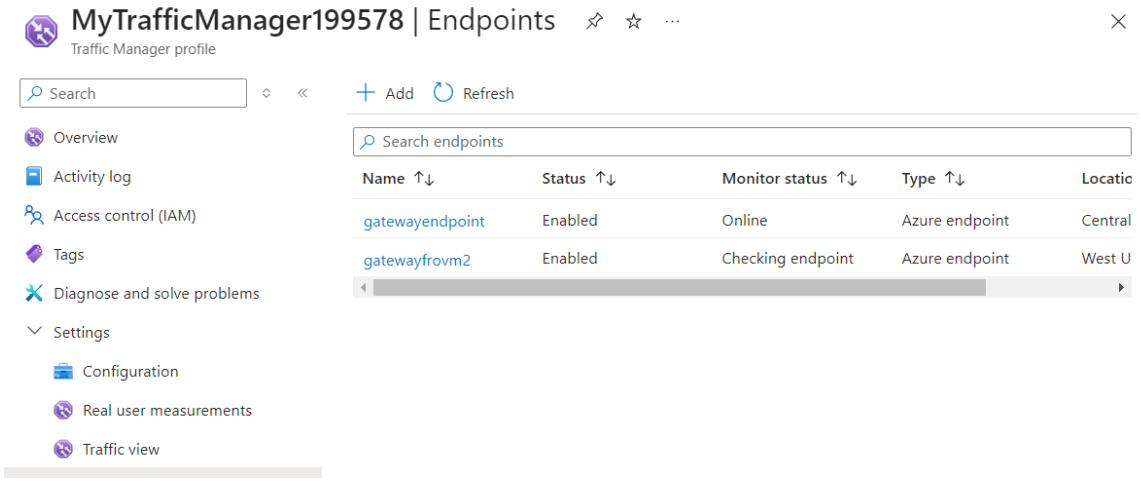
- Under **Settings** in the left-hand menu, select **HTTP settings**.
- Click on your existing HTTP setting or create a new one if required.
- Scroll down to the **Custom error pages** section.
- Set up the **Error page URLs** for 403 and 502 errors:
- For **403 error**, use the URL of your error.html hosted in the Azure Storage Static Website (e.g., https://<storage-account-name>.z13.web.core.windows.net/error.html).
- For **502 error**, use the same URL if desired.
- After entering the custom error page URLs, click **Save**.



7. Implement Traffic Manager

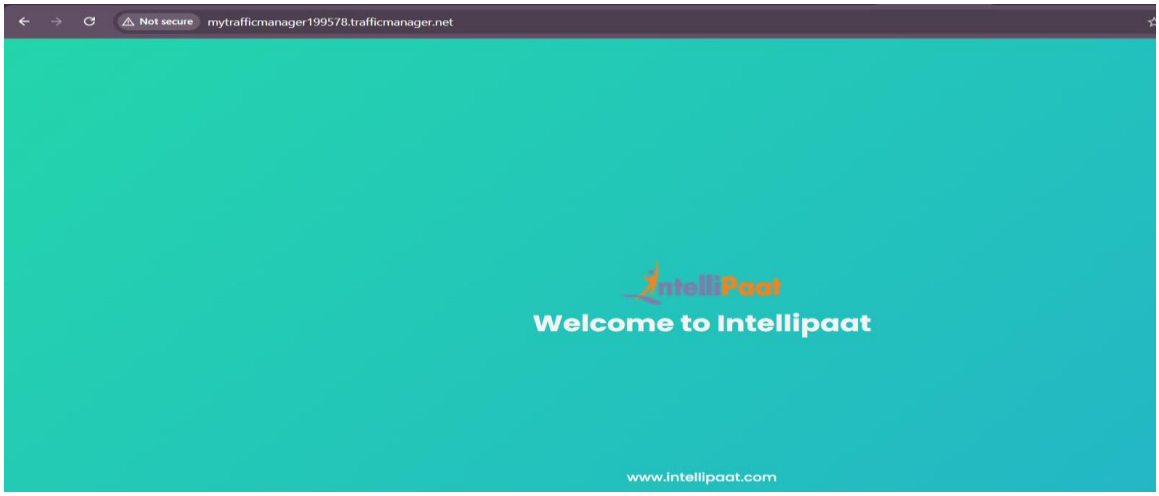
1. **Create Traffic Manager Profile**:
    - Click on **Create a resource** and select **Networking** > **Traffic Manager profile**.
    - Fill in the details (name: MyTrafficManager199578, routing method: Performance).
    - Click **Create**.
2. **Add Endpoints**:

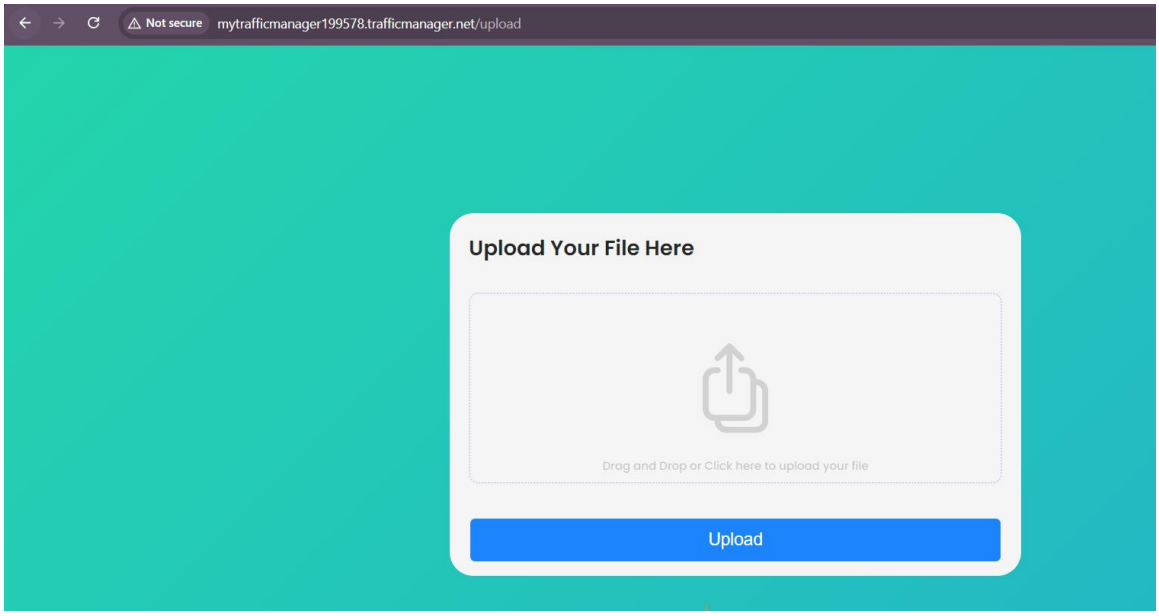
- In the Traffic Manager profile, click on **Endpoints** and add both application gateways as endpoints.

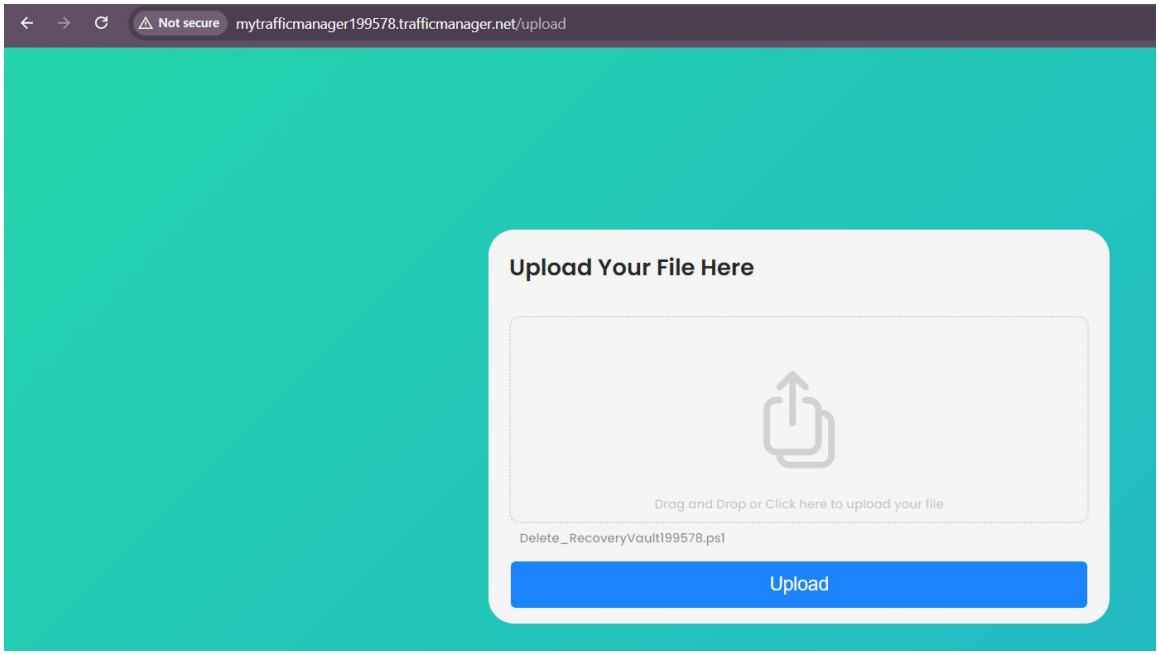


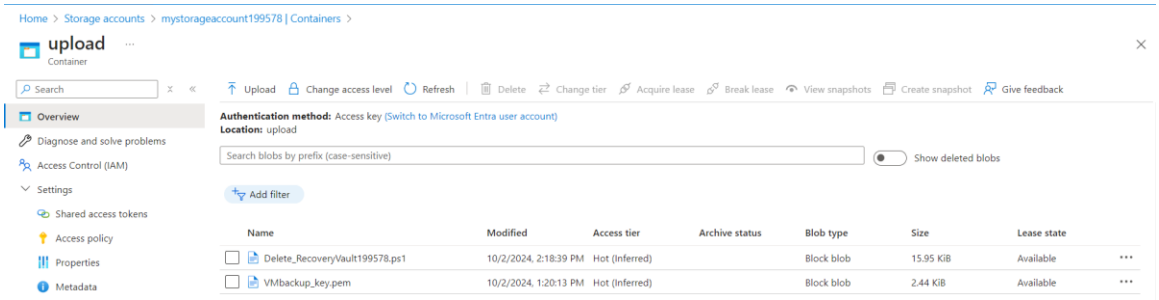
9. Validate the Setup

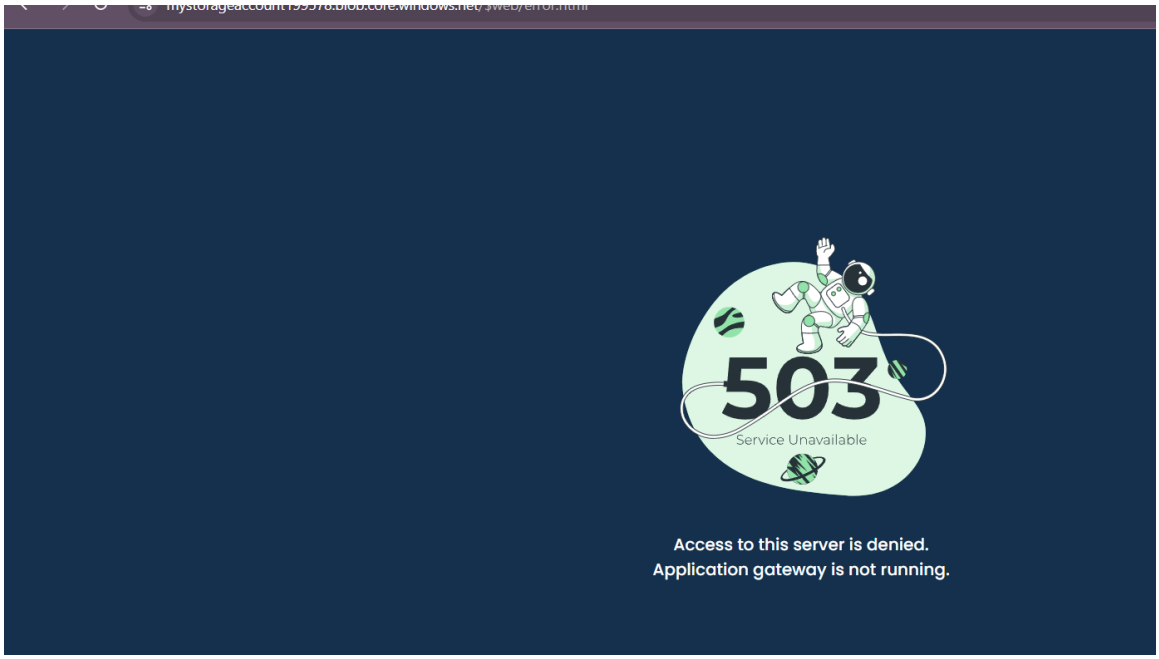
1. **Access the Application**:
    - Open a browser and go to http://mytrafficmanager199578.trafficmanager.net/ to test the application.




Not secure
mytrafficmanager199578.trafficmanager.net/upload
Upload Your File Here
Drag and Drop or Click here to upload your file
Upload


Not secure
mytrafficmanager199578.trafficmanager.net/upload
Upload Your File Here
Drag and Drop or Click here to upload your file
Delete_RecoveryVault199578.ps1
Upload


Home > Storage accounts > mystorageaccount199578 | Containers >
upload
Container
Search
Overview
Diagnose and solve problems
Access Control (IAM)
Settings
Shared access tokens
Access policy
Properties
Metadata
Upload
Change access level
Refresh
Delete
Change tier
Acquire lease
Break lease
View snapshots
Create snapshot
Give feedback
Authentication method: Access key (Switch to Microsoft Entra user account)
Location: upload
Search blobs by prefix (case-sensitive)
Show deleted blobs
Add filter
Name
Modified
Access tier
Archive status
Blob type
Size
Lease state
Delete_RecoveryVault199578.ps1
10/2/2024, 2:18:39 PM
Hot (Inferred)
Block blob
15.95 KiB
Available
VMbackup_key.pem
10/2/2024, 1:20:13 PM
Hot (Inferred)
Block blob
2.44 KiB
Available


503
Service Unavailable
Access to this server is denied.
Application gateway is not running.