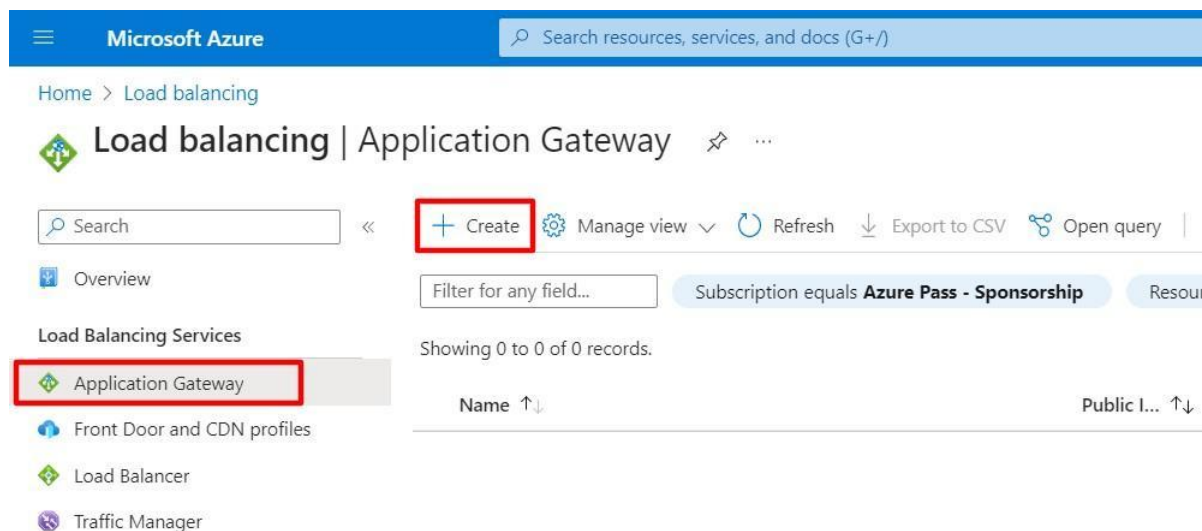




Module 7: Hands-On: Configure and Deploy Azure Application Gateway

Create two VMs first in the same VNet and then launch it.
Now go to Azure Portal and search for Application Gateway and then click on Create.




The screenshot shows the Microsoft Azure portal interface. At the top, there's a blue header with the Microsoft Azure logo and a search bar. Below the header, the breadcrumb navigation shows 'Home > Load balancing'. The main heading is 'Load balancing | Application Gateway'. On the left, there's a sidebar with 'Load Balancing Services' listed: 'Application Gateway' (highlighted with a red box), 'Front Door and CDN profiles', 'Load Balancer', and 'Traffic Manager'. In the main content area, there's a '+ Create' button (highlighted with a red box) and a 'Manage view' dropdown. Below this, there's a filter bar and a table showing 0 records. The table has columns for 'Name' and 'Public IP'.

Step 1: Fill in the basic details such as the resource group, auto-scaling details, and the virtual network. Create a new VNet subnet if it doesn't exist before and click on Frontends



[Home](#) > [Load balancing](#) | [Application Gateway](#) >

Create application gateway ...







 Changes you make on this tab may affect any configuration you've done on other tabs. Review all options prior to creating the application gateway.

Project details



Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources. [?](#)

Subscription * ?	<div>Azure Pass - Sponsorship (4362042a-1e87-43ca-83aa-3c7ebc545a78) </div>
Resource group * ?	<div>Az-104 </div> <div>Create new</div>

Instance details

Application gateway name *	<div>Apg1 </div>
Region *	<div>East US </div>
Tier ?	<div>Standard V2 </div>
Enable autoscaling	<div><input checked="" type="radio"/> Yes <input type="radio"/> No</div>
Minimum instance count * ?	<div>0 </div>
Maximum instance count	<div>10 </div>
Availability zone ?	<div>None </div>
HTTP2 ?	<div><input checked="" type="radio"/> Disabled <input type="radio"/> Enabled</div>

Configure virtual network

Virtual network * ?	<div>Vnet1 </div> <div>Create new</div>
Subnet * ?	<div>subnetapg1 (10.0.1.0/24) </div> <div>Manage subnet configuration</div>

[Previous](#)

[Next : Frontends >](#)

Step 2: Now fill in the public IP address name and then click on OK[Home](#) > [Load balancing | Application Gateway](#) >**Create application gateway** ...

✓ Basics 2 Frontends 3 Backends 4 Configuration 5 Tags 6 Review + create

Traffic enters the application gateway via its frontend IP address(es). An application gateway can use a public IP address, private IP address, or one of each type. ⓘ

Frontend IP address type ⓘ ☒ Public ☐ Private ☐ Both

Public IP address *

Choose public IP address

Add new

Add a public IP

Name *

SKU ☐ Basic ☒ Standard

Assignment ☐ Dynamic ☒ Static

Availability zone

Step 3: Next go to backends, here add a backend pool for both the VMs (pool1 and pool2)

Home > Load balancing > Application Gateway >

Create application gateway

✓ Basics ✓ Frontends **Backends** ○ Configuration ○ Tags ○ Review + create

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, app services, IP addresses, or fully qualified domain names (FQDN).

Add a backend pool

Backend pool: No results

Targets

Previous Next: Configuration >

Add a backend pool.

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machines scale sets, IP addresses, domain names, or an App Service.

Name: pool1

Add backend pool without targets: Yes No

Backend targets: 1 item

Target type	Target
Virtual machine	vm1727 (10.0.0.76)
IP address or FQDN	

Add Cancel

Step 4: Now create a storage account

Microsoft Azure

Home > azstorage13mar, 16013829110791 | Overview >

azstorage13mar

Storage account

Upload Open in Explorer Delete Move Refresh Open in mobile CLI / PS Feedback

Overview

Activity log

Tags

Diagnose and solve problems

Access Control (IAM)

Data migration

Events

Storage browser

Data storage

- Containers
- File shares
- Queues
- Tables

Security + networking

- Networking
- Azure CDN
- Access keys
- Shared access signature
- Encryption
- Microsoft Defender for Cloud

Data management

- Redundancy

Essentials

Resource group: Az-104

Location: East US

Subscription: Azure Pass - Sponsorship

Subscription ID: 4362042e-1e87-43ca-83aa-3c7ebc345a78

Disk state: Available

Tags: Click here to add tags

Performance: Standard

Replication: Locally-redundant storage (LRS)

Account kind: StorageV2 (general purpose v2)

Provisioning state: Succeeded

Created: 4/13/2023, 6:02:05 PM

Properties Monitoring Capabilities (7) Recommendations (0) Tutorials Tools + SDKs

Blob service

Hierarchical namespace	Disabled
Default access tier	Hot
Blob public access	Enabled
Blob soft delete	Enabled (7 days)
Container soft delete	Enabled (7 days)
Versioning	Disabled
Change feed	Disabled
NFS v3	Disabled
Allow cross-tenant replication	Enabled

File service

Large file share	Disabled
Active Directory	Not configured
Default share-level permissions	Disabled
Soft delete	Enabled (7 days)

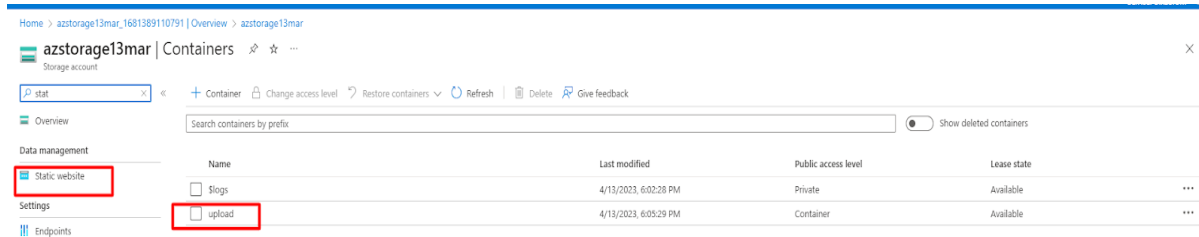
Security

Require secure transfer for REST API operations	Enabled
Storage account key access	Enabled
Minimum TLS version	Version 1.2
Infrastructure encryption	Disabled

Networking

Allow access from	All networks
Number of private endpoint connections	0
Network routing	Microsoft network routing
Access for trusted Microsoft services	Yes
Endpoint type	Standard

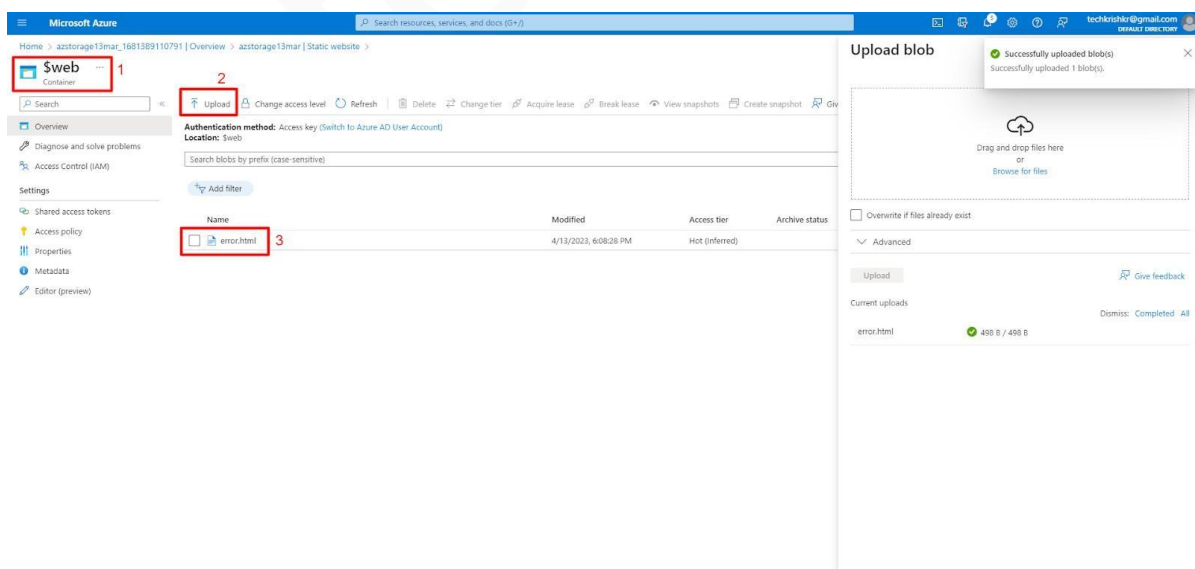
Step 5: Create a container inside it and upload the data through static website



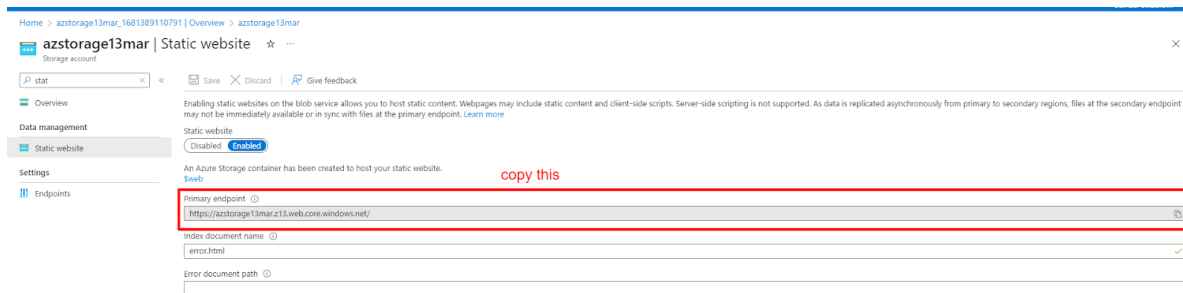
Step 6: Now click on the \$web and upload the data



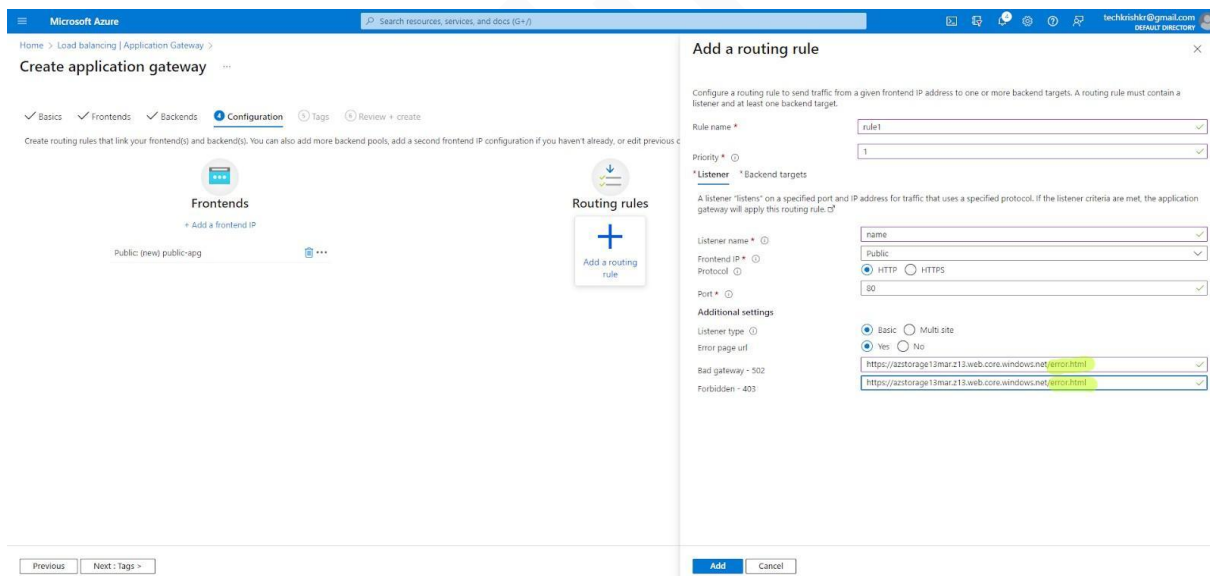
Step 7: Upload the data as mentioned in the diagram



Step 8: Now copy the Primary endpoint from the Static website



Step 9: Under Configuration, add a routing rule, enter the rule name and then specify the routing rules here along with the backend targets. Turn the error page URL to yes to add a link for the static web page to a storage account



Step 10: Now fill in the details of backend targets and also fill in the path-based rule. Click on Add

Add a routing rule ×

Configure a routing rule to send traffic from a given frontend IP address to one or more backend targets. A routing rule must contain a listener and at least one backend target.

Rule name *

rule1 ✓

Priority * ⓘ

1 ✓

* Listener

Backend targets

Choose a backend pool to which this routing rule will send traffic. You will also need to specify a set of Backend settings that define the behavior of the routing rule. ⓘ

Target type

☒ Backend pool
 ☐ Redirection

Backend target * ⓘ

pool1 ▼

Add new

Backend settings * ⓘ

default ▼

Add new

Path-based routing

You can route traffic from this rule's listener to different backend targets based on the URL path of the request. You can also apply a different set of Backend settings based on the URL path. ⓘ

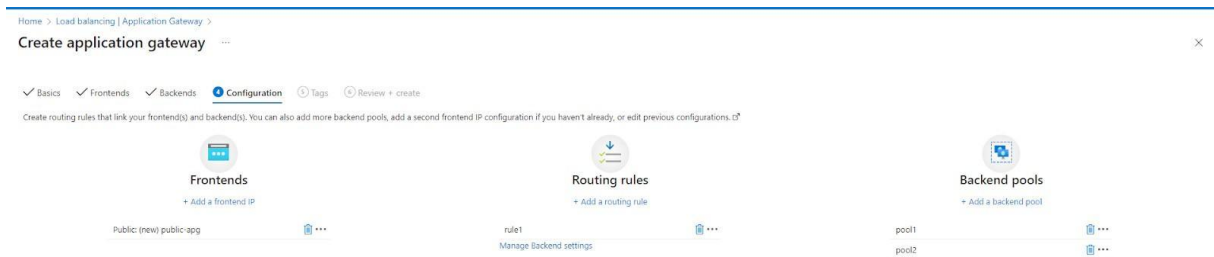
Path based rules				
Path	Target name	Backend setting name	Backend pool	
/index.html	vm1	default	pool1	...

[Add multiple targets to create a path-based rule](#)

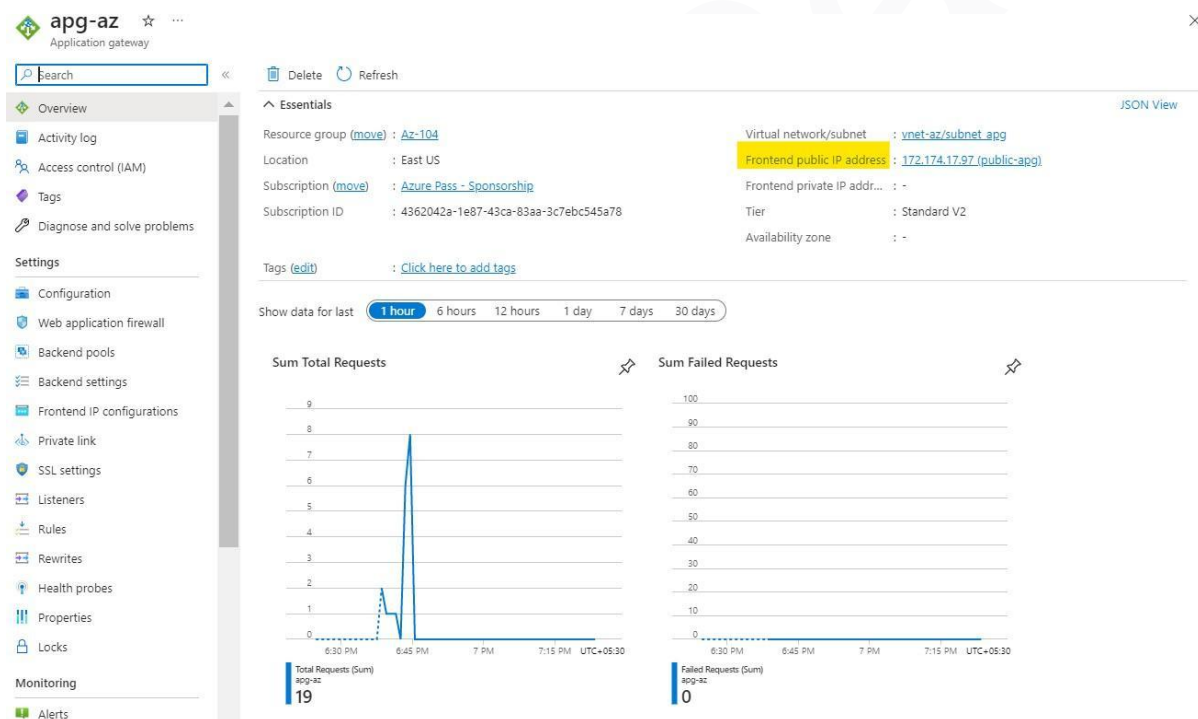
Add

Cancel

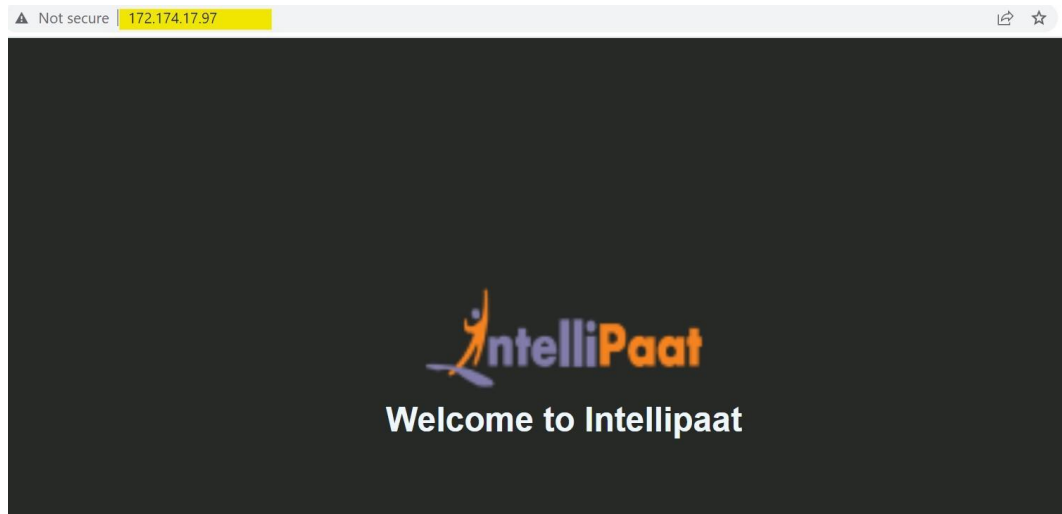
Step 11: Review all the details of frontends, backends and routing rules. Then Review + create



Step 12: After deployment, copy the frontend public IP address



Step 13: Paste the IP address in the browser and you can check the default page



Step 14: Also, change the path (IP address/index.html) and check it for the index.html page

