

Assignment 1: Implement Network Traffic Management

Report by: Tonny Odhiambo, CS-CNS06-24028

Introduction

In this lab, I explored the implementation of traffic management solutions using Azure services. Specifically, I configured and tested both a public Azure Load Balancer and an Azure Application Gateway. These tools are essential for distributing incoming public requests across multiple virtual machines, thus ensuring efficient resource utilization and enhanced application performance.

The lab scenario involved deploying a public website, distributing images and videos from different virtual machines, and utilizing Azure's load balancing capabilities to manage traffic effectively. Through hands-on tasks, I gained practical experience in setting up a load balancer, creating virtual networks, deploying backend servers, and configuring an application gateway to handle complex web traffic.

Task 1: Used a Template to Provision an Infrastructure

I started by downloading the required template and parameter files from the provided lab resources. After signing into the Azure portal at <https://portal.azure.com>, I searched for "Deploy a custom template," selected "Build your own template in the editor," and loaded the az104-06-vms-template.json file. After saving the template, I loaded the corresponding parameters file (az104-06-vms-parameters.json). I then completed the deployment form with the following settings: my Azure subscription, a resource group named az104-rg6 (which I created as necessary), and a secure password for the virtual machines. I reviewed and created the deployment, ensuring it included one virtual network with three subnets and two virtual machines.

The screenshot displays the Microsoft Azure portal interface. At the top, the navigation bar shows 'Microsoft Azure' and 'Upgrade' buttons. The main header indicates the deployment is in progress for 'Microsoft.Template-20240704194626'. The left sidebar contains navigation links for 'Overview', 'Inputs', 'Outputs', and 'Template'. The central pane shows the deployment details, including the deployment name, subscription, resource group, start time, and correlation ID. A table lists the resources created, their types, and their statuses. The right sidebar provides links to Microsoft Defender for Cloud, Free Microsoft tutorials, and Work with an expert. The bottom status bar shows the current date and time as Thursday, 4 July 2024, 19:52 (Local time).

Resource	Type	Status	Operation details
az104-06-vm0/customScriptExtension	Microsoft.Compute/virtualMach	Created	Operation details
az104-06-vm2/customScriptExtension	Microsoft.Compute/virtualMach	Created	Operation details
az104-06-vm1/customScriptExtension	Microsoft.Compute/virtualMach	Created	Operation details
az104-06-vm0	Virtual machine	OK	Operation details
az104-06-vm1	Virtual machine	OK	Operation details
az104-06-vm2	Virtual machine	OK	Operation details
az104-06-nic2	Network interface	Created	Operation details
az104-06-nic1	Network interface	Created	Operation details
az104-06-nic0	Network interface	Created	Operation details
az104-06-vnet1	Virtual network	OK	Operation details
az104-06-nsg1	Network security group	OK	Operation details

Microsoft Azure

Upgrade

Search resources, services, and docs (G+)

Home >

Microsoft.Template-20240704194626 | Overview

Deployment

Search

Delete

Cancel

Redeploy

Download

Refresh

Overview

Inputs

Outputs

Template

✓ Your deployment is complete

Deployment name : Microsoft.Template-20240704194626

Subscription : Azure Free Tier

Resource group : az104-rg6

Start time : 7/4/2024, 7:46:56 PM

Correlation ID : 8ae4f6bc-0762-4e5e-939c-52184e1f0091

> Deployment details

< Next steps

Go to resource group

Give feedback

Tell us about your experience with deployment

Task 2: Configure an Azure Load Balancer

I implemented a public Azure Load Balancer to distribute traffic across two virtual machines.

Microsoft Azure

Search resources, services, and docs (G+)

odhiamboyano@gmail....

DEFAULT DIRECTORY (ODHIAMBOYANO@GMAIL.COM)

Home >

Create load balancer

Basics

Frontend IP configuration

Backend pools

A frontend IP configuration is an IP address used for inbound rules.

+ Add a frontend IP configuration

Name ↑↓

Add a frontend IP to get started

Add frontend IP configuration

az104-lb

Name *

az104-fe

IP version

IPv4

IPv6

IP type

IP address

IP prefix

Public IP address *

(new) az104-lbpip

Create new

Gateway Load balancer ⓘ

None

Give feedback

zm

Chrome

VS Code

in

W

ENG INTL

20:10

04/07/2024

Created a Load Balancer and configured frontend IP and Backend pool.


Microsoft Azure

Search resources, services, and docs (G+)

6

odhiamboyano@gm
DEFAULT DIRECTORY (ODH

Home >

 Microsoft.LoadBalancer-20240704200631 | Overview

Deployment

Search

×

«

Delete

Cancel

Redeploy

Download

Refresh

Overview


Inputs

Outputs

Template

✓

Your deployment is complete



Deployment name : Microsoft.LoadBalancer-20240704200631

Subscription : Azure Free Tier

Resource group : az104-rg6

Start time : 7/4/2024, 8:14:42 PM

Correlation ID : 3d93e78f-d4ca-476b-93c8-d7208bec0122

Deployment details


Resource	Type	Status
<div>✓</div> NicUpdate-76edcf	<div>↑</div> Deployment	OK
<div>✓</div> NicUpdate-5951d7	<div>↑</div> Deployment	OK
<div>✓</div> az104-lb	<div>⬢</div> Load balancer	Created
<div>✓</div> az104-lbpip	<div>🌐</div> Public IP address	OK

Next steps

Go to resource

Give feedback

Tell us about your experience with deployment



Cost management

zm

chrome

edge

teams

in

mail

W

ENG
INTL

20:20
04/07/2024

I then added a Load Balancing Rule

Microsoft Azure

Search resources, services, and docs (G+)

odhiamboyano@gmail....

DEFAULT DIRECTORY (ODHIAMB...

Home > Microsoft.LoadBalancer-20240704200631 | Overview > az104-lb

az104-lb | Load balancing rules

Load balancer

Search

+ Add Refresh | Delete

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

- Frontend IP configuration
- Backend pools
- Health probes
- Load balancing rules**
- Inbound NAT rules
- Outbound rules
- Properties
- Locks

Monitoring

Automation

Help

A load balancer rule is used to define how incoming traffic is distributed to the all the instances within the backend pool. A load-balancing rule maps a given frontend IP configuration and port to multiple backend IP addresses and ports. An example would be a rule created on port 80 to load balance web traffic. [Learn more.](#)

Filter by name...

Name	Protocol	Backend pool	Health probe
az104-lbrule	TCP/80 to TCP...	az104-be	az104-hp

zm Chrome VS Code Figma LinkedIn Telegram Word

ENG INTL 20:29 04/07/2024

Task 3: Configure an Azure Application Gateway

I set up an Azure Application Gateway to manage web traffic with path-based routing.

First, I created a new subnet to be used by the Azure Application Gateway

The screenshot shows the 'Edit subnet' configuration page in the Azure portal. The left-hand navigation pane includes links for Home, Overview, Activity log, Access control, Tags, Diagnose, Settings, Address space, Connections, Subnets, Bastion, and DDoS. The 'Subnets' link is selected. The main content area is titled 'Edit subnet' and contains the following configuration details:

- Subnet purpose:** Default
- Name:** subnet-appgw
- IPv4 configuration:**
 - Include an IPv4 address space:** ☒
 - Choose a starting address and size within your IPv4 address range:** 10.60.0.0/22 (10.60.0.0 - 10.60.3.255)
 - Starting address:** 10.60.3.224
 - Size:** /27 (32 addresses)
 - Subnet address range:** 10.60.3.224 - 10.60.3.255
- IPv6 configuration:**
 - Include an IPv6 address space:** ☐ This virtual network has no IPv6 address ranges.

Next, I created an application gateway

The screenshot shows the 'Create application gateway' page in the Azure portal. The breadcrumb navigation is 'Home > Load balancing | Application Gateway >'. The title is 'Create application gateway'. Below the title is a description: 'An application gateway is a web traffic load balancer that enables you to manage traffic to your web application. [Learn about creating application gateway](#)'. The page is divided into two main sections: 'Project details' and 'Instance details'.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Resource group *
[Create new](#)

Instance details

Application gateway name *

Region *

Tier

Enable autoscaling ☐ Yes ☒ No

Instance count *

Availability zone *

HTTP2 ☒ Disabled ☐ Enabled

IP address type ☒ IPv4 only ☐ Dual stack (IPv4 & IPv6)

Configure virtual network

Virtual network *
[Create new](#)

Subnet *
[Manage subnet configuration](#)

The bottom of the screenshot shows a Windows taskbar with icons for Zoom, Google Chrome, File Explorer, OneDrive, LinkedIn, Telegram, and Word. The system tray on the right shows network and volume icons, and the language is set to 'ENG INTL'.

Configured Frontend IP: I used a public IP address for the frontend named az104-gwpip.

The screenshot shows the 'Create application gateway' page in the Azure portal, specifically the 'Frontends' step. The breadcrumb navigation is 'Home > Load balancing | Application Gateway >'. The title is 'Create application gateway'. Below the title is a progress bar with six steps: 1. Basics, 2. Frontends (selected), 3. Backends, 4. Configuration, 5. Tags, and 6. Review + create. Below the progress bar is a description: '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.' The 'Frontend IP address type' is set to 'Public'. The 'Public IPv4 address' is set to '(New) az104-gwpip'. There is a link to 'Add new' below the dropdown menu.

✓ 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 IPv4 address *
[Add new](#)

Configured Backend Pools: I created three backend pools:

- az104-appgwbe with both VMs.
- az104-imagebe with az104-06-nic1.
- az104-videobe with az104-06-nic2.

Micros... Upg Search resources, services, and docs (G+/)

Home > Load balancing | Application Gateway >

Create application gateway

Application gateway needs at least one valid Backend pool. Click 'Add a backend pool' to create a new backend pool.

✓ Basics ✓ Frontends ① Backends ④ Configuration ⑤ Tags ⑥ 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	Targets
az104-appgwbe	2 targets az104-06-nic1 az104-06-nic2

Added another backend pool for images.

Micros... Upg Search resources, services, and docs (G+/)

Home > Load balancing | Application Gateway >

Create application gateway

Changes you make on this tab may affect any configuration

✓ Basics ✓ Frontends ✓ Backends ④ Configuration

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, app services, IP addresses, or an App Service.

Add a backend pool

Backend pool
az104-appgwbe

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 * az104-imagebe

Add backend pool without targets Yes No

Backend targets 1 item

Target type	Target
Virtual machine	az104-06-nic1 (10.60.1.4)
IP address or FQDN	

Added another backend pool for video

The screenshot shows the 'Add a backend pool' page in the Azure portal. The left sidebar has tabs for 'Basics', 'Frontends', 'Backends', and 'Connections'. The 'Backends' tab is selected. The main content area has a title 'Add a backend pool.' and a description: '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.' Below this, there are fields for 'Name *' (az104-videobe), 'Add backend pool without targets' (Yes/No buttons), and 'Backend targets' (1 item). The 'Backend targets' section has a table with columns 'Target type' and 'Target'. The first row shows 'Virtual machine' as the target type and 'az104-06-nic2 (10.60.2.4)' as the target. Below this, there is a section for 'IP address or FQDN'.

Home > Load balancing | Application Gateway >

Create application gateway

Changes you make on this tab may affect any configuration

✓ Basics ✓ Frontends ✓ **Backends** (4) Connections

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, app services, IP addresses, domain names, or an App Service.

[Add a backend pool](#)

Backend pool

- az104-appgwbe
- az104-imagebe

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 * az104-videobe

Add backend pool without targets Yes No

Backend targets

1 item

Target type	Target
Virtual machine	az104-06-nic2 (10.60.2.4)
IP address or FQDN	

Addition of Routing Rule: I added a routing rule named az104-gwrule with path-based targets:

- /image/* to az104-imagebe.
- /video/* to az104-videobe.

The screenshot shows the 'Add a routing rule' page in the Azure portal. The left sidebar has tabs for 'Basics', 'Frontends', 'Backends', and 'Connections'. The 'Basics' tab is selected. The main content area has a title 'Add a routing rule' and a description: '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.' Below this, there are fields for 'Rule name *' (az104-gwrule), 'Priority *' (10), and 'Listener *'. The 'Listener *' section has fields for 'Listener name *' (az104-listener), 'Frontend IP *' (Public IPv4), 'Protocol' (HTTP/HTTPS), 'Port *' (80), and 'Listener type' (Basic/Multi site). Below this, there is a section for 'Custom error pages' with fields for 'Bad Gateway - 502' and 'Forbidden - 403'.

Home > Load balancing | Application Gateway >

Create application gateway

Basics

Create routing rule already, or create a new one

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 * az104-gwrule

Priority * 10

* Listener * Backend targets

A listener "listens" on a specified port and IP address for traffic that uses a specified protocol. If the listener criteria are met, the application gateway will apply this routing rule.

Listener name * az104-listener

Frontend IP * Public IPv4

Protocol HTTP HTTPS

Port * 80

Listener type Basic Multi site

Custom error pages

Show customized error pages for different response codes generated by Application Gateway. This section lets you configure Listener-specific error pages. [Learn more](#)

Please verify that the url(s) being added here is reachable from your application gateway using the [connection troubleshoot](#) tool to prevent any deployment error.

Bad Gateway - 502 Enter Html file URL

Forbidden - 403 Enter Html file URL

[Show more status codes](#)

Added multiple targets to create a path-based rule.

Micros...

Upd

Search resources, services, and docs (G+)

3

odhiamboyano@gmail...

DEFAULT DIRECTORY

Home > Load balancers

Create a new routing rule

Close

Basics

Create routing rule

already, or edit an existing rule

Public: (new)

Rule name *

az104-gwrule

Priority *

10

* 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 *

az104-appgwbe

Add new

Backend settings *

az104-http

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	
/image/*	images	az104-http	az104-imagebe	...
/video/*	videos	az104-http	az104-videobe	...

Add multiple targets to create a path-based rule

zm

in

W

ENG

INTL

21:31

04/07/2024

PRE

The application gateway deployed successfully

Microsoft Azure | Upgrade | Search resources, services, and docs (G+)

Home > Microsoft.ApplicationGateway-20240704210650 | Overview

Deployment

Search | Delete | Cancel | Redeploy | Download | Refresh

Overview

Inputs

Outputs

Template

✓ Your deployment is complete

Deployment name : Microsoft.ApplicationGateway-20240704210650 Start time : 7/4/2024 9:35:01 PM

Subscription : Azure Free Tier Correlation ID : 8b703221-ace4-4cee-a1fc-4e47e967e522

Resource group : az104-rg5

Deployment details

Resource	Type	Status	Operation details
az104-06-nic2-20240704213445	Deployment	OK	Operation details
az104-06-nic1-20240704213445	Deployment	OK	Operation details
az104-appgw	Application gateway	OK	Operation details
az104-gwip	Public IP address	OK	Operation details

Next steps

Go to resource group

Give feedback

Tell us about your experience with deployment

Cost management

Get notified to stay within your budget and prevent unexpected charges on your bill. Set up cost alerts >

Microsoft Defender for Cloud

Secure your apps and infrastructure. Go to Microsoft Defender for Cloud >

Free Microsoft tutorials

Start learning today >

Work with an expert

Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support. Find an Azure expert >

Testing of Application Gateway

I verified backend health and tested the public IP address with /image/ and /video/ paths in a browser.

Microsoft Azure | Search resources, services, and docs (G+)

Home > Microsoft.ApplicationGateway-20240704210650 | Overview > az104-appgw

♥ az104-appgw | Backend health ☆ ...

Application gateway

Search | Refresh | Feedback

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Monitoring

Alerts

Metrics

Diagnostic settings

Logs

Insights

Backend health

Connection troubleshoot

Automation

Help

Backend health

By default, Azure Application Gateway probes backend servers to check their health and whether they're ready to serve requests. You can also create custom **Health Probes** to mention a specific hostname and path to be probed or a response code to be accepted as Healthy.

The Backend health report is updated based on the respective probe's refresh interval and doesn't depend on the page refresh.

All 4 out of 4

Healthy 4 out of 4

Search backend health

Server (backend...↑↓	Status	↑↓	Port (Backend s...↑↓	Protocol	↑↓	Details
10.60.2.4 (az104-app...	✓ Healthy		80 (az104-http)	Http		Success. Recei code
10.60.1.4 (az104-app...	✓ Healthy		80 (az104-http)	Http		Success. Recei code
10.60.2.4 (az104-vid...	✓ Healthy		80 (az104-http)	Http		Success. Recei code
10.60.1.4 (az104-ima...	✓ Healthy		80 (az104-http)	Http		Success. Recei code

I opened the Overview blade of my Azure Firewall, copied the Frontend public IP address, and used it to test the URL **<http://191.235.121.3/image/>** in a browser to verify I was directed to the image server (vm1).

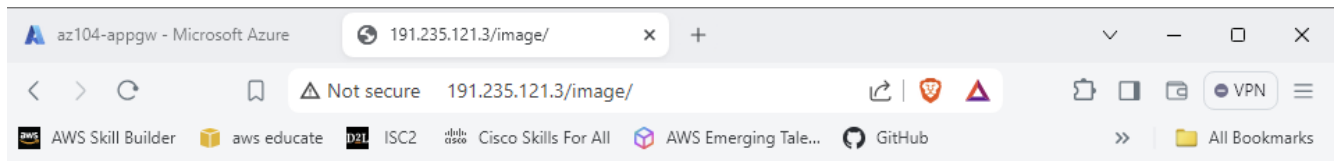
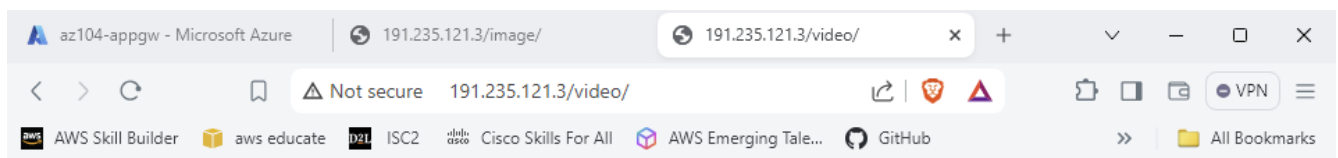


Image from: az104-06-vm1

In another browser window, I tested the URL **<http://191.235.121.3/video/>** and verified I was directed to the video server (vm2).

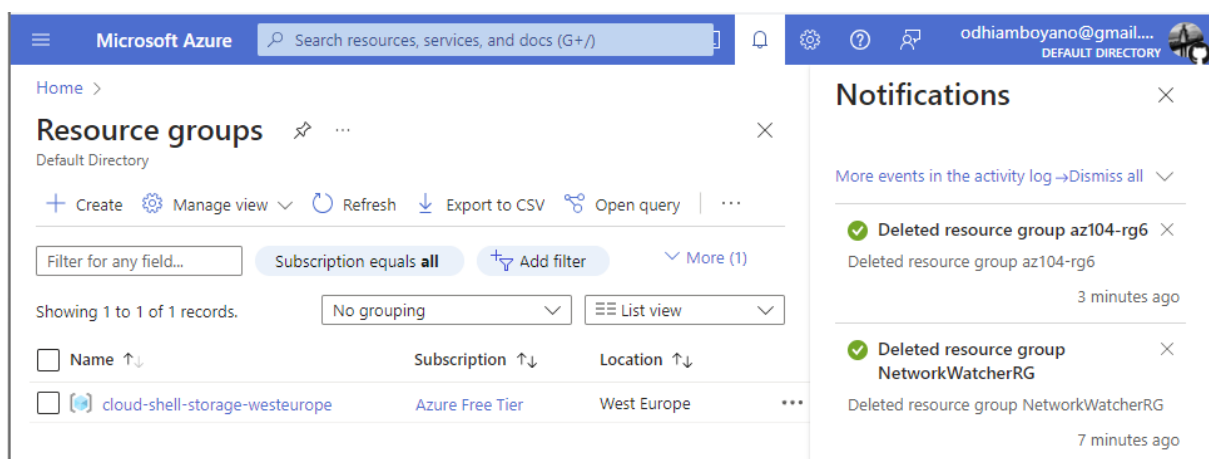


Video from: az104-06-vm2

Cleanup

I ensured that resources were cleaned up to prevent unnecessary costs.

In the Azure portal, I deleted the resource group az104-rg6 to remove all associated resources.



Conclusion

In this lab, I successfully implemented both an Azure Load Balancer and an Azure Application Gateway. These tools allowed me to efficiently manage and distribute web traffic across multiple virtual machines. The practical experience gained from this lab is highly valuable and applicable to real-world scenarios, particularly in optimizing web application performance and reliability.