ASSIGNMENT ONE: NETWORK TRAFFIC MANAGEMENT WRITEUP

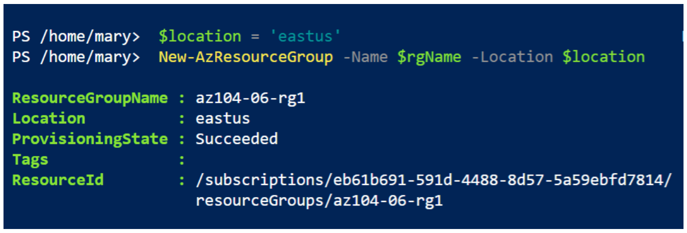
BY CS-CNS03-23082 – ABUOR ISABELLA MERCY

INTRODUCTION

Throughout this hands-on experience lab assignment, you'll master the setup, configuration, and testing of diverse networking components within Azure's dynamic ecosystem. By engaging with a series of tasks, including provisioning the lab environment, configuring hub and spoke network topology, testing virtual network peering transitivity, setting up routing, implementing Azure Load Balancer, and leveraging Azure Application Gateway. The following is a descriptive of a step-by-step procedure of how the tasks were achieved.

Task 1: Provision the lab environment

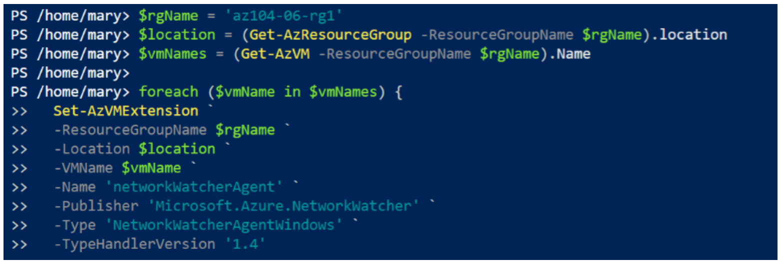
In this task, we deployed four virtual machines into the same Azure region on the Azure Cloud Shell and uploaded the az104-06-vms-loop template.json and \Allfiles\Labs\06\az104-06-vms-loop-parameters.json into the Cloud Shell home directory. From the Cloud Shell pane, run the following to create the first resource group that will be hosting the lab environment.



```
PS /home/mary>  $location = 'eastus'
PS /home/mary>  New-AzResourceGroup -Name $rgName -Location $location

ResourceGroupName : az104-06-rg1
Location          : eastus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/eb61b691-591d-4488-8d57-5a59ebfd7814/
                    resourceGroups/az104-06-rg1
```

Note: we created the three virtual networks and four Azure VMs into them by using the template and parameter files you uploaded using the custom template blade in azure portal.

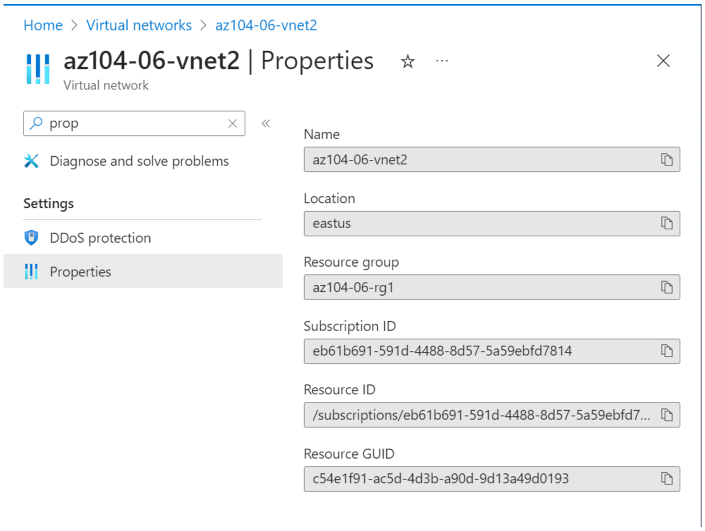
From the Cloud Shell pane, run the following to install the Network Watcher extension on the Azure VMs deployed in the previous step:



```
PS /home/mary> $rgName = 'az104-06-rg1'
PS /home/mary> $location = (Get-AzResourceGroup -ResourceGroupName $rgName).location
PS /home/mary> $vmNames = (Get-AzVM -ResourceGroupName $rgName).Name
PS /home/mary>
PS /home/mary> foreach ($vmName in $vmNames) {
>>   Set-AzVMExtension `
>>   -ResourceGroupName $rgName `
>>   -Location $location `
>>   -VMName $vmName `
>>   -Name 'networkWatcherAgent' `
>>   -Publisher 'Microsoft.Azure.NetworkWatcher' `
>>   -Type 'NetworkWatcherAgentWindows' `
>>   -TypeHandlerVersion '1.4'
```

Task 2: Configure the hub and spoke network topology

In this task, we configured local peering between the virtual networks you deployed in the previous tasks in order to create a hub and spoke network topology On the Virtual networks page, in the list of virtual networks, select az104-06-vnet2. On the az104-06-vnet2 blade, select Properties. On the az104-06-vnet2 | Properties blade, record the value of the Resource ID property. The same was done for select az104-06-vnet3.


Home > Virtual networks > az104-06-vnet2
prop
Diagnose and solve problems
Settings
DDoS protection
Properties

az104-06-vnet2 | Properties
Virtual network

Name: az104-06-vnet2

Location: eastus

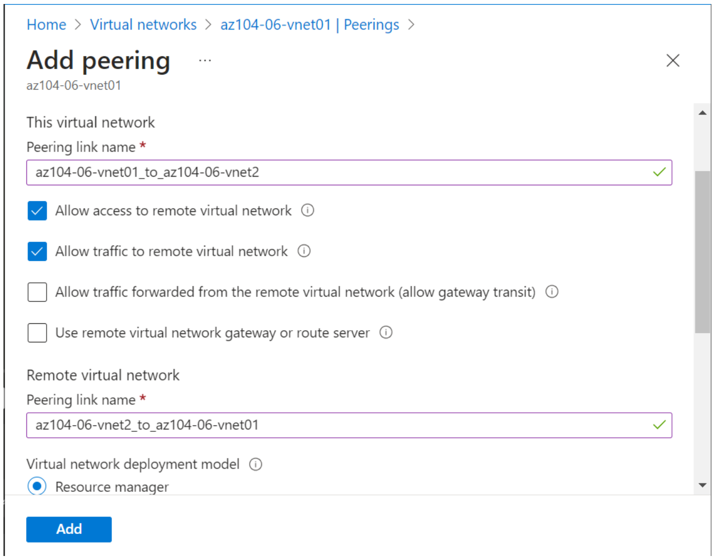
Resource group: az104-06-rg1

Subscription ID: eb61b691-591d-4488-8d57-5a59ebfd7814

Resource ID: /subscriptions/eb61b691-591d-4488-8d57-5a59ebfd7...

Resource GUID: c54e1f91-ac5d-4d3b-a90d-9d13a49d0193

In the list of virtual networks, click az104-06-vnet01. On the az104-06-vnet01 virtual network blade, in the Settings section, added the following settings:



Add peering
az104-06-vnet01

This virtual network

Peering link name *: az104-06-vnet01_to_az104-06-vnet2

☑ Allow access to remote virtual network

☑ Allow traffic to remote virtual network

□ Allow traffic forwarded from the remote virtual network (allow gateway transit)

□ Use remote virtual network gateway or route server

Remote virtual network

Peering link name *: az104-06-vnet2_to_az104-06-vnet01

Virtual network deployment model

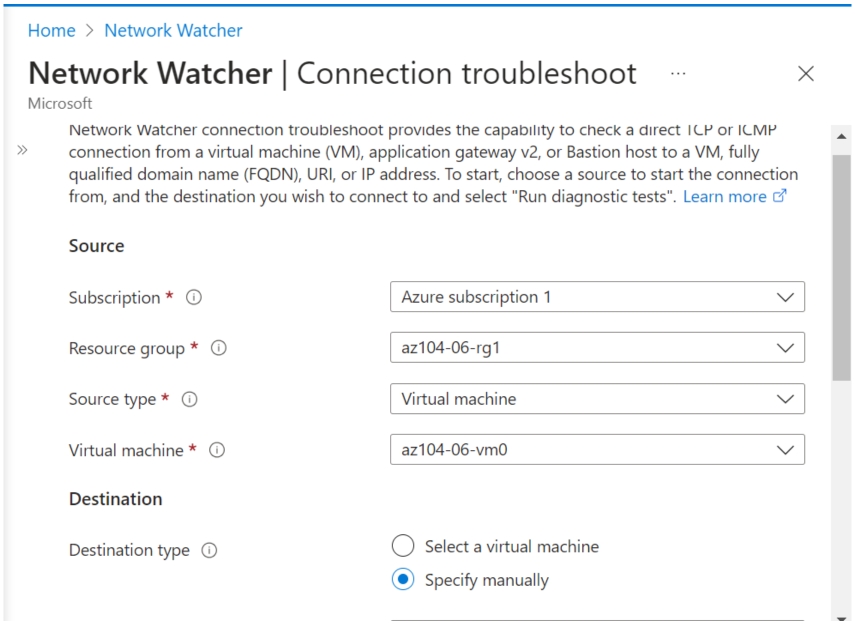
● Resource manager

The same was done for the az104-06-vnet01 virtual network.

Task 3: Test transitivity of virtual network peering

In this task, we tested transitivity of virtual network peering by using Network Watcher.

On the Network Watcher blade, expand the listing of Azure regions and verify the service is enabled in region you are using. On the Network Watcher blade, navigate to the Connection troubleshoot. On the Network Watcher - Connection troubleshoot blade, initiate a check with the following settings:



Network Watcher | Connection troubleshoot

Microsoft

Network Watcher connection troubleshoot provides the capability to check a direct TCP or ICMP connection from a virtual machine (VM), application gateway v2, or Bastion host to a VM, fully qualified domain name (FQDN), URI, or IP address. To start, choose a source to start the connection from, and the destination you wish to connect to and select "Run diagnostic tests". Learn more

Source

Subscription * : Azure subscription 1

Resource group * : az104-06-rg1

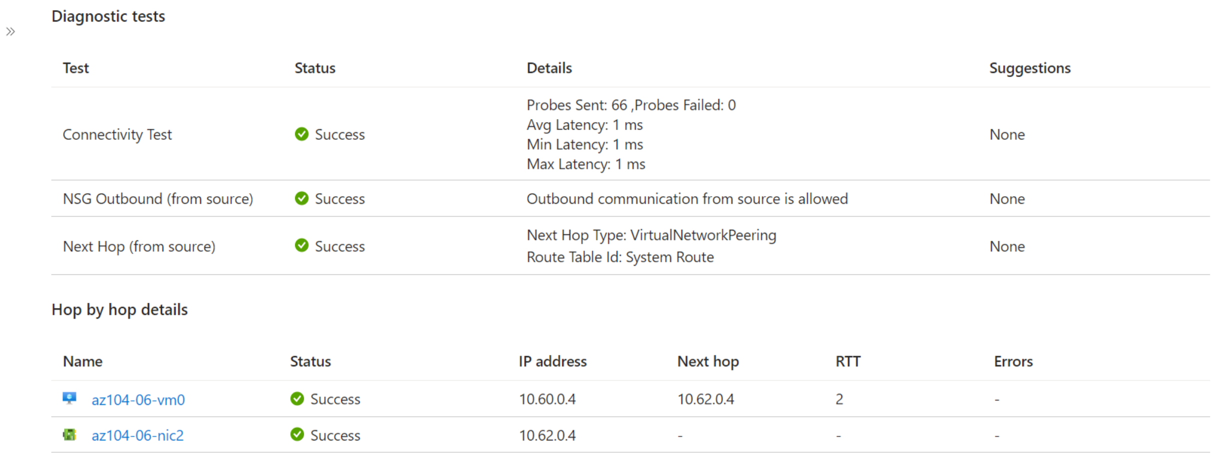
Source type * : Virtual machine

Virtual machine * : az104-06-vm0

Destination

Destination type : ○ Select a virtual machine ● Specify manually

Then clicked Run diagnostic tests and wait until results of the connectivity check are returned.



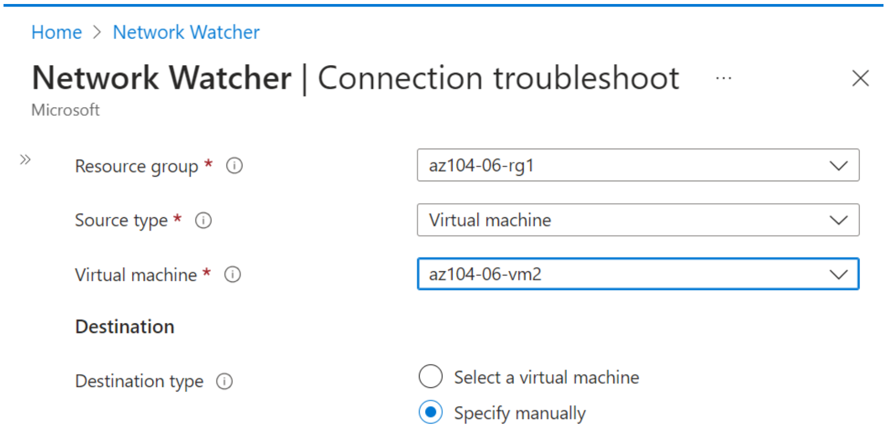
Diagnostic tests

Test	Status	Details	Suggestions
Connectivity Test	✓ Success	Probes Sent: 66 ,Probes Failed: 0 Avg Latency: 1 ms Min Latency: 1 ms Max Latency: 1 ms	None
NSG Outbound (from source)	✓ Success	Outbound communication from source is allowed	None
Next Hop (from source)	✓ Success	Next Hop Type: VirtualNetworkPeering Route Table Id: System Route	None

Hop by hop details

Name	Status	IP address	Next hop	RTT	Errors
az104-06-vm0	✓ Success	10.60.0.4	10.62.0.4	2	-
az104-06-nic2	✓ Success	10.62.0.4	-	-	-

On the Network Watcher - Connection troubleshoot blade again, initiate a check with the following settings:



Home > Network Watcher

Network Watcher | Connection troubleshoot

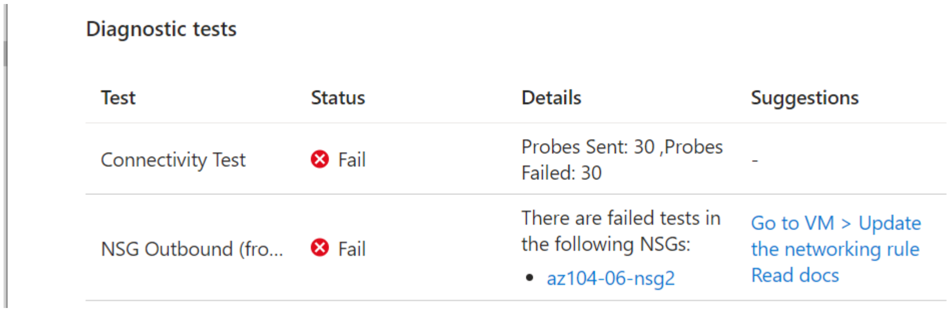
Microsoft

Resource group *	az104-06-rg1
Source type *	Virtual machine
Virtual machine *	az104-06-vm2

Destination

Destination type
○ Select a virtual machine
● Specify manually

Click Run diagnostic tests and wait until results of the connectivity check are returned. Note that the status is Fail.



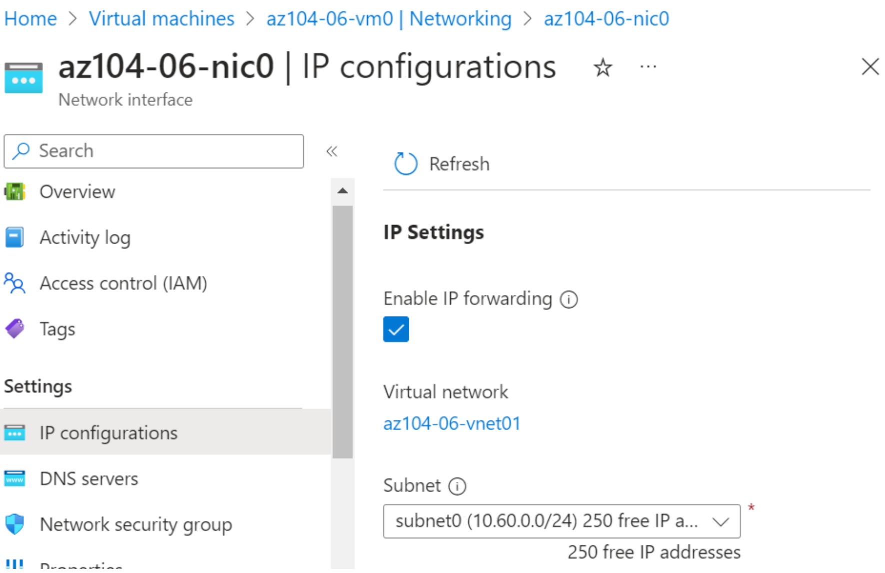
Diagnostic tests

Test	Status	Details	Suggestions
Connectivity Test	☒ Fail	Probes Sent: 30 ,Probes Failed: 30	-
NSG Outbound (fro...	☒ Fail	There are failed tests in the following NSGs: • az104-06-nsg2	Go to VM > Update the networking rule Read docs

Task 4: Configure routing in the hub and spoke topology

In this task, we configured and tested routing between the two spoke virtual networks by enabling IP forwarding on the network interface of the az104-06-vm0 virtual machine, enabling routing within its operating system, and configuring user-defined routes on the spoke virtual network.

On the Virtual machines blade, in the list of virtual machines, click az104-06-vm0. On the az104-06-vm0 virtual machine blade, in the Settings section, click Networking. Click the az104-06-nic0 link next to the Network interface label, and then, on the az104-06-nic0 network interface blade, in the Settings section, click IP configurations. Set IP forwarding to Enabled and save the change like the figure below:



az104-06-nic0 | IP configurations

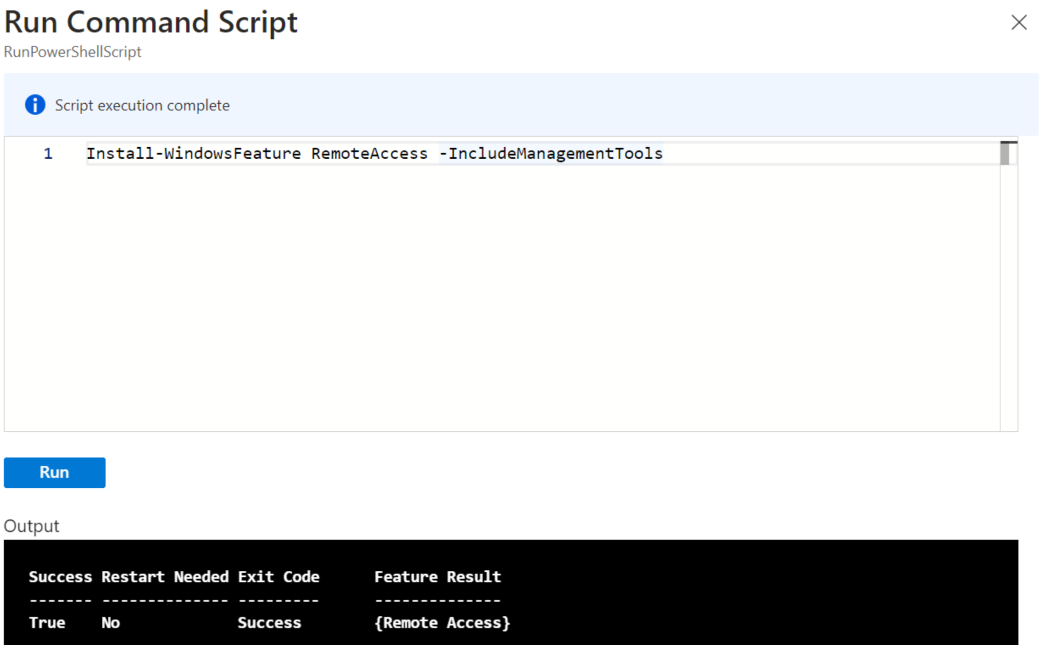
IP Settings

Enable IP forwarding ☑

Virtual network
az104-06-vnet01

Subnet *
subnet0 (10.60.0.0/24) 250 free IP a...
250 free IP addresses

In the Azure portal, navigate back to the az104-06-vm0 Azure virtual machine blade and click Overview. On the az104-06-vm0 blade, in the Operations section, click Run command, and, in the list of commands, click RunPowerShellScript. On the Run Command Script blade, type the following and click Run to install the Remote Access Windows Server role.



Run Command Script
RunPowerShellScript

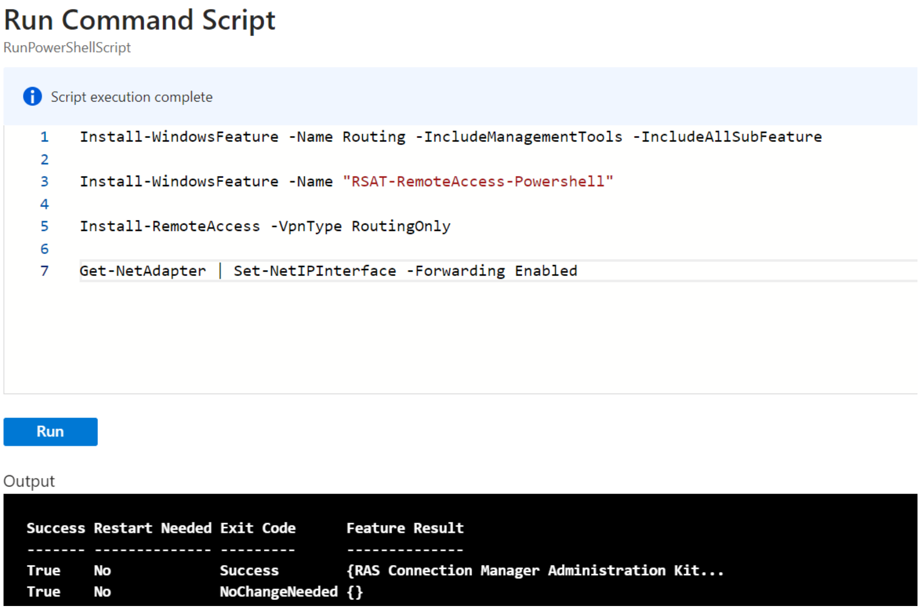
Script execution complete

```
Install-WindowsFeature RemoteAccess -IncludeManagementTools
```

Output

```
Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Remote Access}
```

On the Run Command Script blade, type the following and click Run to install the Routing role service.



Run Command Script

RunPowerShellScript

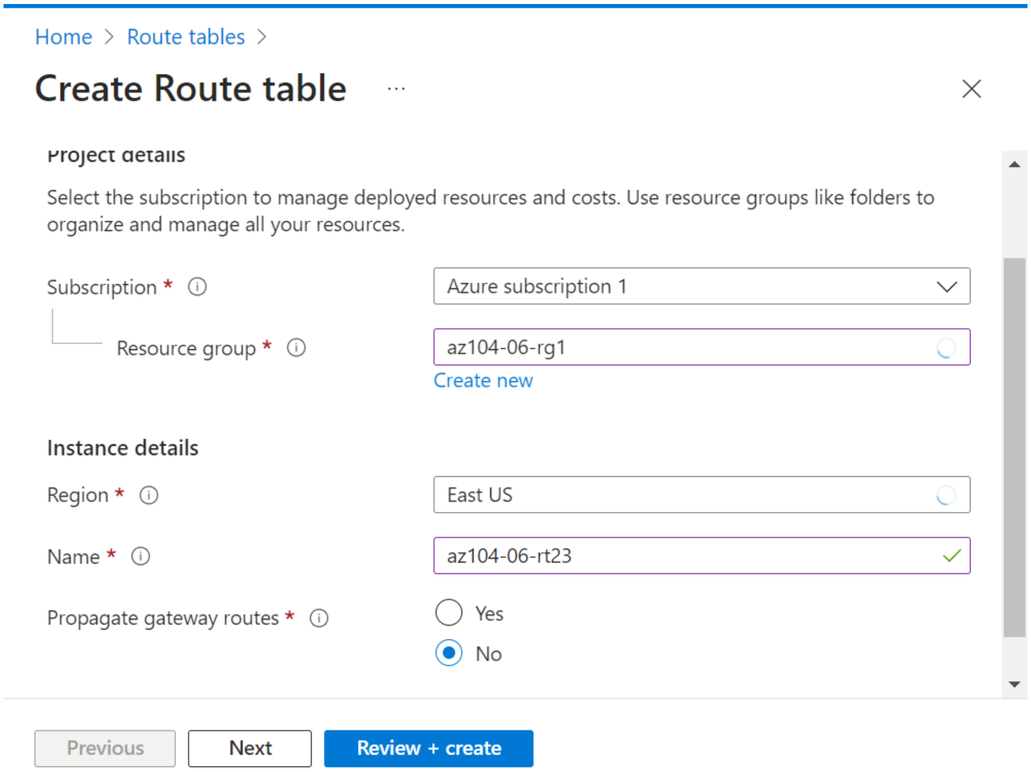
Script execution complete

```
Install-WindowsFeature -Name Routing -IncludeManagementTools -IncludeAllSubFeature

Install-WindowsFeature -Name "RSAT-RemoteAccess-Powershell"

Install-RemoteAccess -VpnType RoutingOnly

Get-NetAdapter | Set-NetIPInterface -Forwarding Enabled
```

Output

```
Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {RAS Connection Manager Administration Kit...
True    No             NoChangeNeeded {}
```

In the Azure portal, search and select Route tables and create a new Route table with the following settings:



Create Route table

Project details

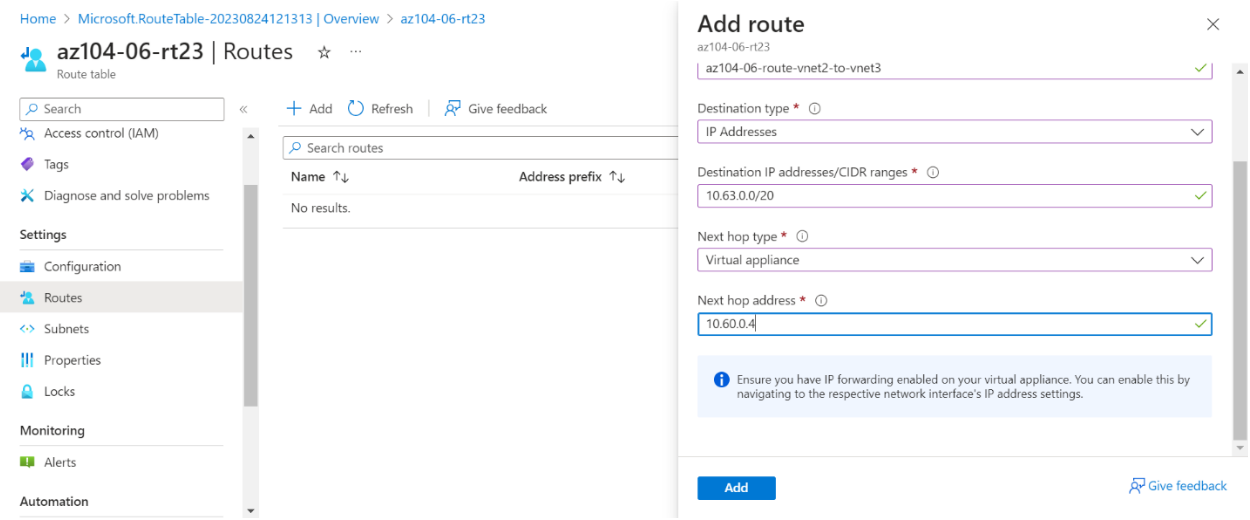
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Azure subscription 1
Resource group *	az104-06-rg1

Instance details

Region *	East US
Name *	az104-06-rt23
Propagate gateway routes *	○ Yes ● No

On resource, On the az104-06-rt23 route table blade, in the Settings section, add a new route with the following settings:



Add route

az104-06-rt23

az104-06-route-vnet2-to-vnet3

Destination type *: IP Addresses

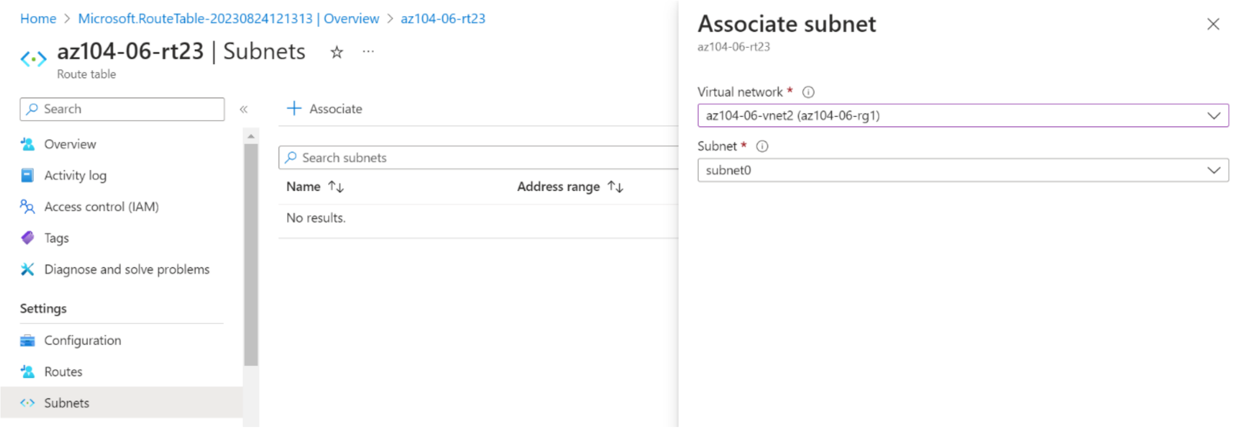
Destination IP addresses/CIDR ranges *: 10.63.0.0/20

Next hop type *: Virtual appliance

Next hop address *: 10.60.0.4

Ensure you have IP forwarding enabled on your virtual appliance. You can enable this by navigating to the respective network interface's IP address settings.

Back on the az104-06-rt23 route table blade, in the Settings section, click Subnets, and then click + Associate and associate the route table az104-06-rt23 with the following subnet:



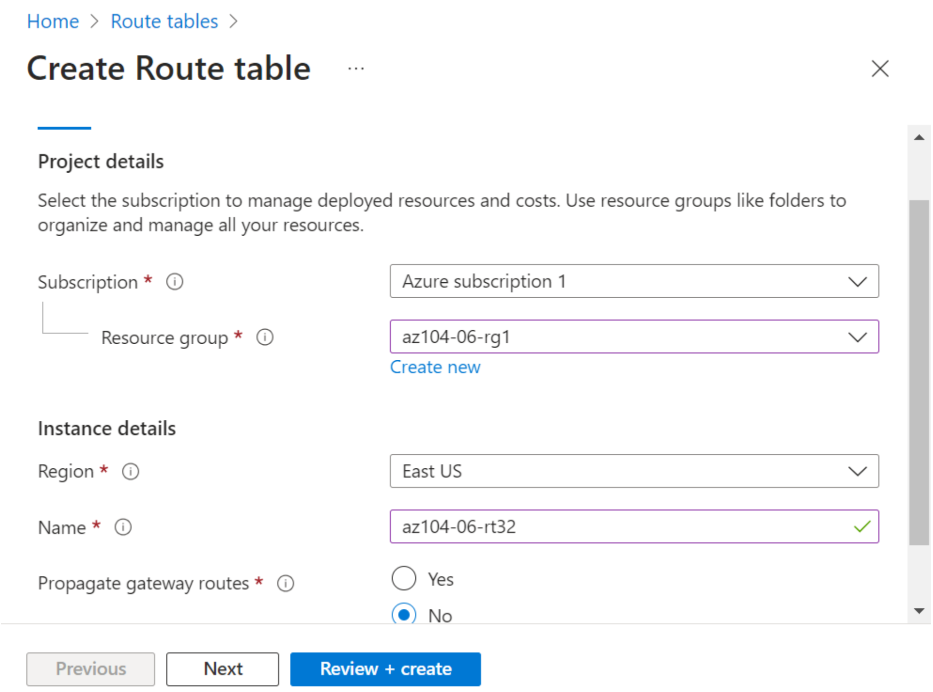
Associate subnet

az104-06-rt23

Virtual network *: az104-06-vnet2 (az104-06-rg1)

Subnet *: subnet0

Navigate back to Route tables blade and created a new route with the following settings:



Create Route table

Project details

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

Subscription * : Azure subscription 1

Resource group * : az104-06-rg1

Create new

Instance details

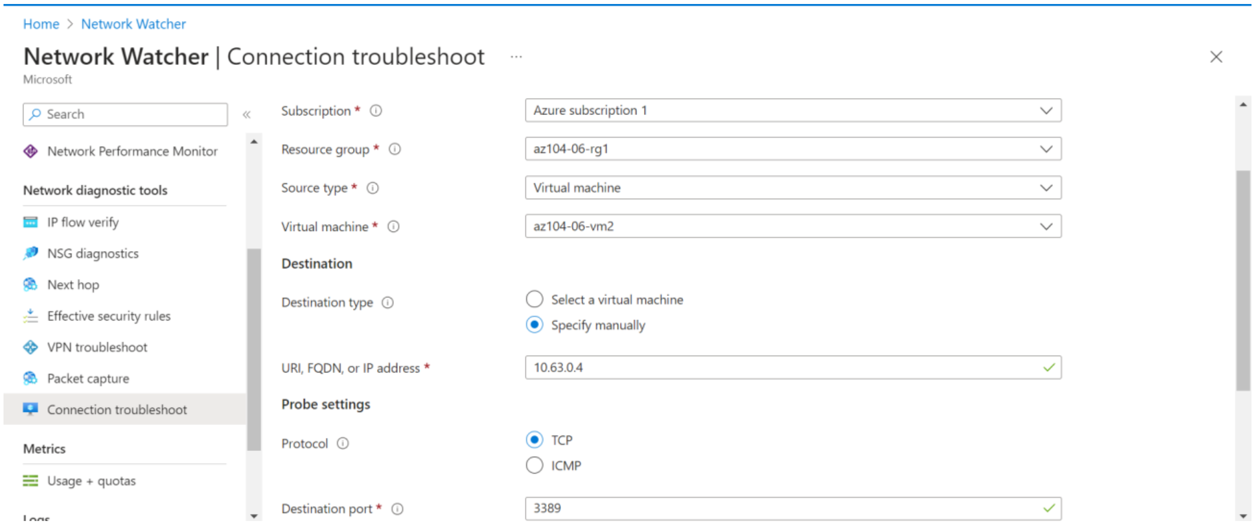
Region * : East US

Name * : az104-06-rt32

Propagate gateway routes * : ○ Yes ● No

And the same step was on the az104-06-rt23 route table.

In the Azure portal, navigate back to the Network Watcher - Connection troubleshoot blade. On the Network Watcher - Connection troubleshoot blade, use the following settings:



Network Watcher | Connection troubleshoot

Microsoft

Subscription * : Azure subscription 1

Resource group * : az104-06-rg1

Source type * : Virtual machine

Virtual machine * : az104-06-vm2

Destination

Destination type: ○ Select a virtual machine ● Specify manually

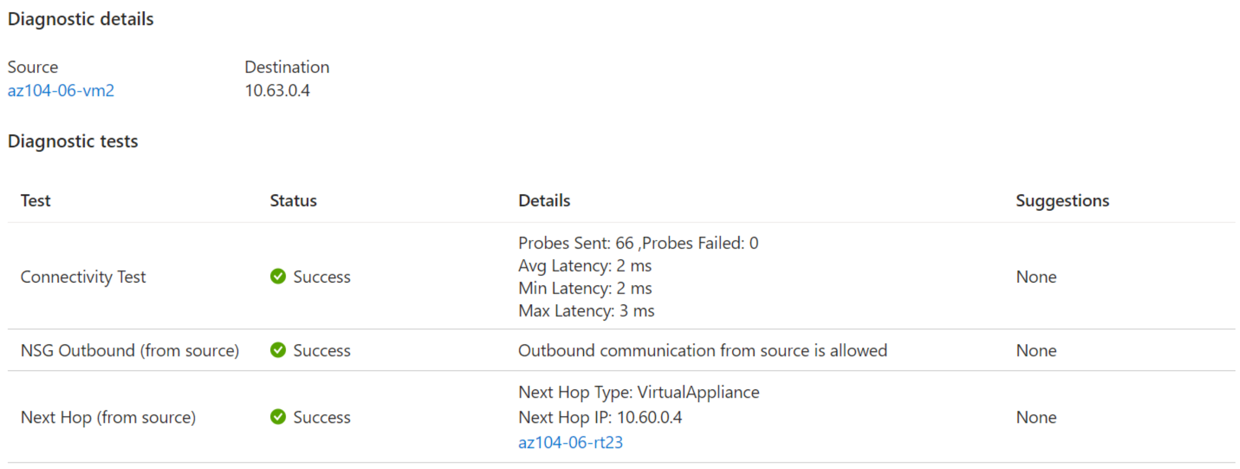
URI, FQDN, or IP address * : 10.63.0.4

Probe settings

Protocol: ● TCP ○ ICMP

Destination port * : 3389

Click Run diagnostic tests and waited until results of the connectivity check are returned.



Diagnostic details

Source
az104-06-vm2

Destination
10.63.0.4

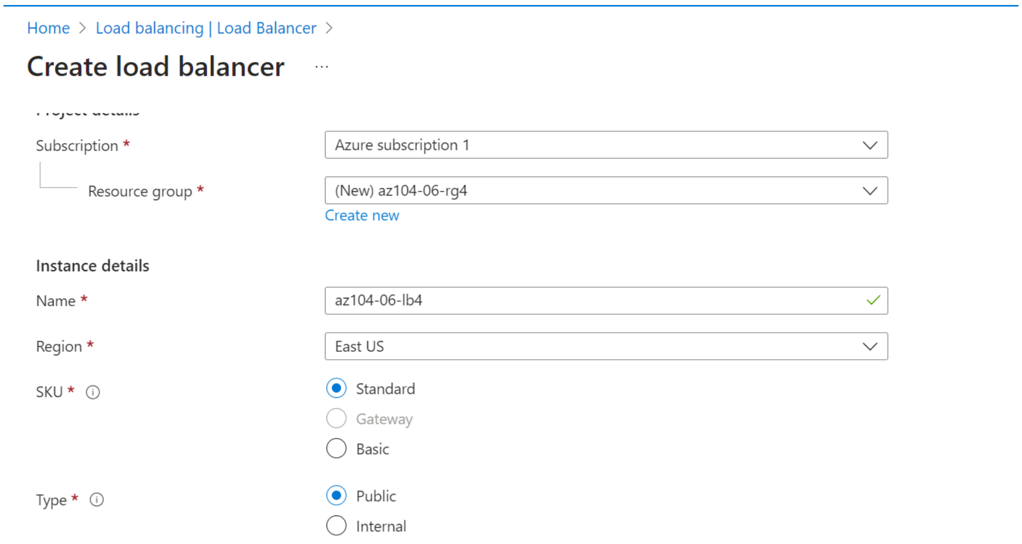
Diagnostic tests

Test	Status	Details	Suggestions
Connectivity Test	✓ Success	Probes Sent: 66 ,Probes Failed: 0 Avg Latency: 2 ms Min Latency: 2 ms Max Latency: 3 ms	None
NSG Outbound (from source)	✓ Success	Outbound communication from source is allowed	None
Next Hop (from source)	✓ Success	Next Hop Type: VirtualAppliance Next Hop IP: 10.60.0.4 az104-06-rt23	None

Task 5: Implement Azure Load Balancer

In this task, we implemented an Azure Load Balancer in front of the two Azure virtual machines in the hub virtual network.

On the Load balancers page and created a new load balancer with the following settings then click Next: Frontend IP configuration:



Home > Load balancing | Load Balancer >

Create load balancer ...

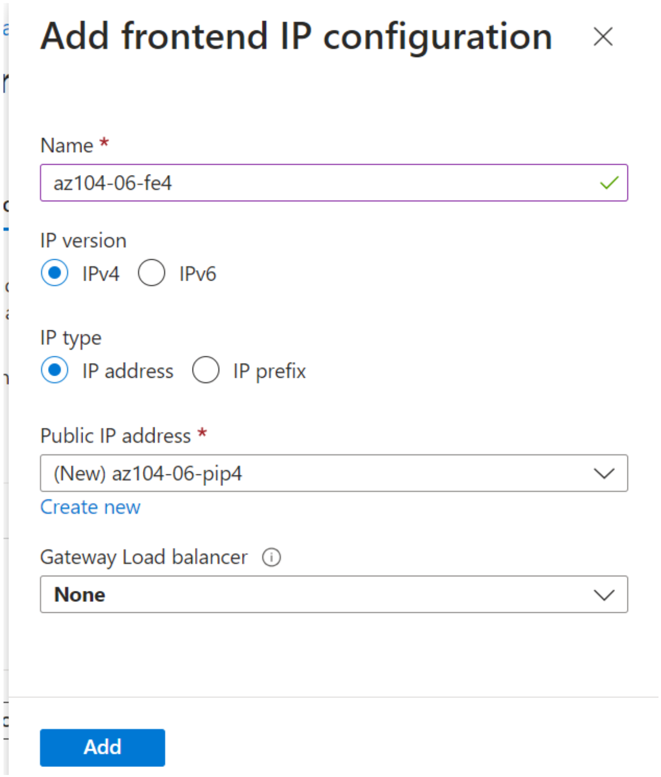
Project details

Subscription *	Azure subscription 1
Resource group *	(New) az104-06-rg4 Create new

Instance details

Name *	az104-06-lb4 ✓
Region *	East US
SKU * ⓘ	● Standard ○ Gateway ○ Basic
Type * ⓘ	● Public ○ Internal

On the Frontend IP configuration tab, click Add a frontend IP configuration and use the following settings:



Add frontend IP configuration

Name * az104-06-fe4

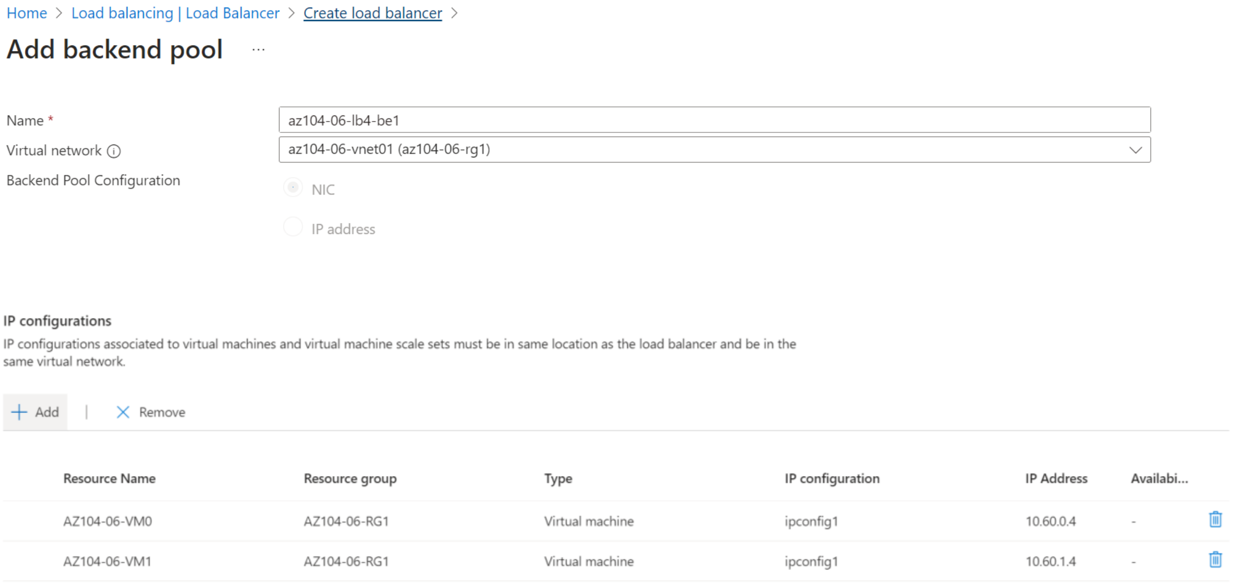
IP version ● IPv4 ○ IPv6

IP type ● IP address ○ IP prefix

Public IP address * (New) az104-06-pip4

Gateway Load balancer None

On the Add a public IP address popup, use the following settings before clicking OK and then Add. When completed click Next: Backend pools. On the Backend pools tab, click Add a backend pool with the following settings (leave others with their default values). Click + Add (twice) and then click Next:Inbound rules.



Home > Load balancing | Load Balancer > Create load balancer >

Add backend pool

Name * az104-06-lb4-be1

Virtual network az104-06-vnet01 (az104-06-rg1)

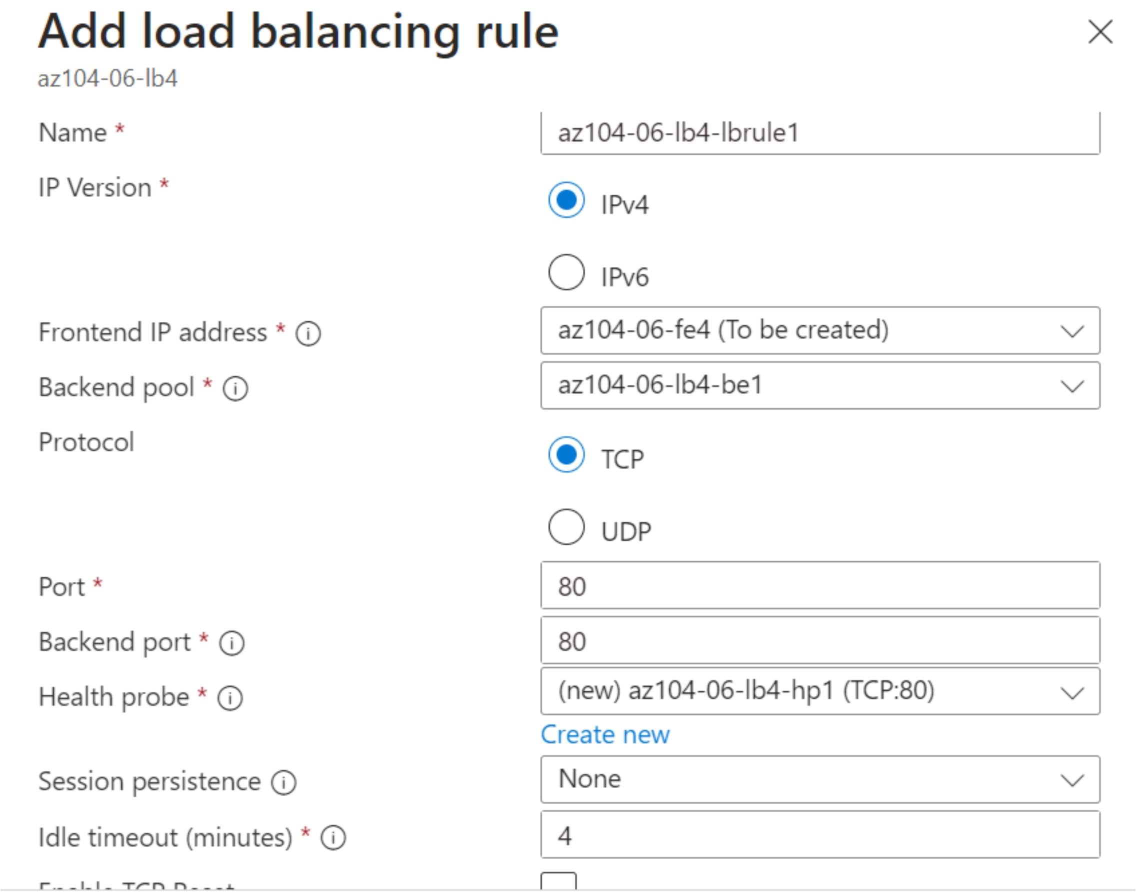
Backend Pool Configuration ● NIC ○ IP address

IP configurations

IP configurations associated to virtual machines and virtual machine scale sets must be in same location as the load balancer and be in the same virtual network.

Resource Name	Resource group	Type	IP configuration	IP Address	Availabi...
AZ104-06-VM0	AZ104-06-RG1	Virtual machine	ipconfig1	10.60.0.4	-
AZ104-06-VM1	AZ104-06-RG1	Virtual machine	ipconfig1	10.60.1.4	-

On the Backend pools tab, click Add a backend pool with the following settings (leave others with their default values). Click + Add (twice) and then click Next:Inbound rules. On the Inbound rules tab, click Add a load balancing rule. Add a load balancing rule with the following settings. When completed click Add.

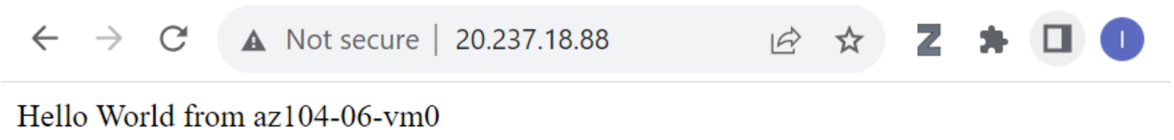


Add load balancing rule

az104-06-lb4

Name *	az104-06-lb4-lbrule1
IP Version *	● IPv4 ○ IPv6
Frontend IP address * ⓘ	az104-06-fe4 (To be created)
Backend pool * ⓘ	az104-06-lb4-be1
Protocol	● TCP ○ UDP
Port *	80
Backend port * ⓘ	80
Health probe * ⓘ	(new) az104-06-lb4-hp1 (TCP:80) Create new
Session persistence ⓘ	None
Idle timeout (minutes) * ⓘ	4
Enable TCP Reset	□

After the load balancer to deploy then click Go to resource. Select Frontend IP configuration from the Load Balancer resource page. Copy the IP address. Open another browser tab and navigate to the IP address. Verify that the browser window displays the message Hello World from az104-06-vm0 or Hello World from az104-06-vm1.

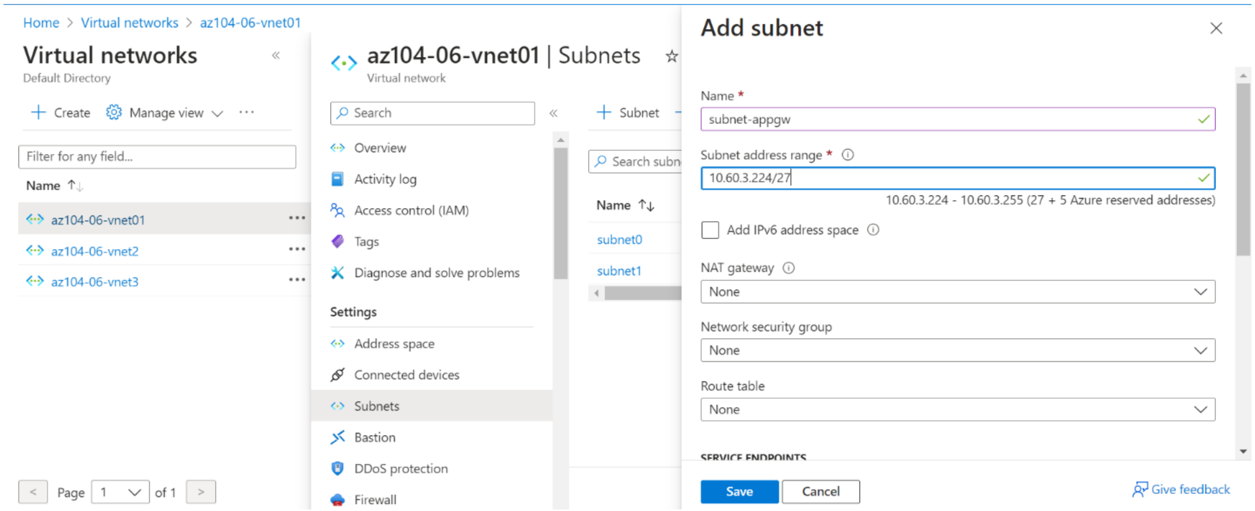


Hello World from az104-06-vm0

Task 6: Implement Azure Application Gateway

In this task, we implemented an Azure Application Gateway in front of the two Azure virtual machines in the spoke virtual networks.

On the Virtual networks blade, in the list of virtual networks, click az104-06-vnet01. On the az104-06-vnet01 virtual network blade, in the Settings section, click Subnets, and then Add a subnet with the following settings:



Name *: subnet-appgw

Subnet address range *: 10.60.3.224/27

10.60.3.224 - 10.60.3.255 (27 + 5 Azure reserved addresses)

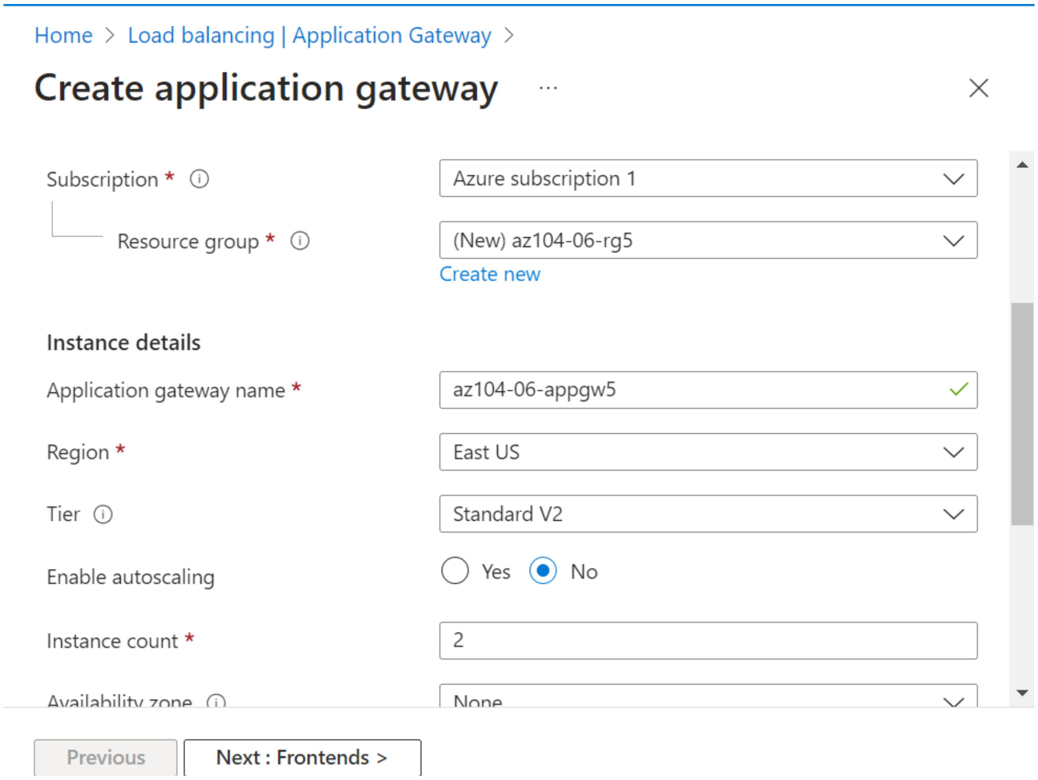
☐ Add IPv6 address space

NAT gateway: None

Network security group: None

Route table: None

In the Azure portal, search and select Application Gateways and, on the Application Gateways blade, click + Create. On the Basics tab, specify the following settings:



Home > Load balancing | Application Gateway >

Create application gateway ...

Subscription * ⓘ — Azure subscription 1

Resource group * ⓘ — (New) az104-06-rg5

Create new

Instance details

Application gateway name * — az104-06-appgw5 ✓

Region * — East US

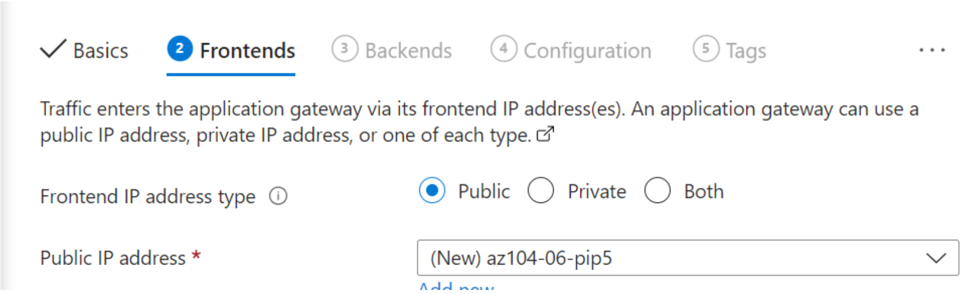
Tier ⓘ — Standard V2

Enable autoscaling — ○ Yes ● No

Instance count * — 2

Availability zone ⓘ — None

Click Next: Frontends > and specify the following settings (leave others with their default values). When complete, click OK.



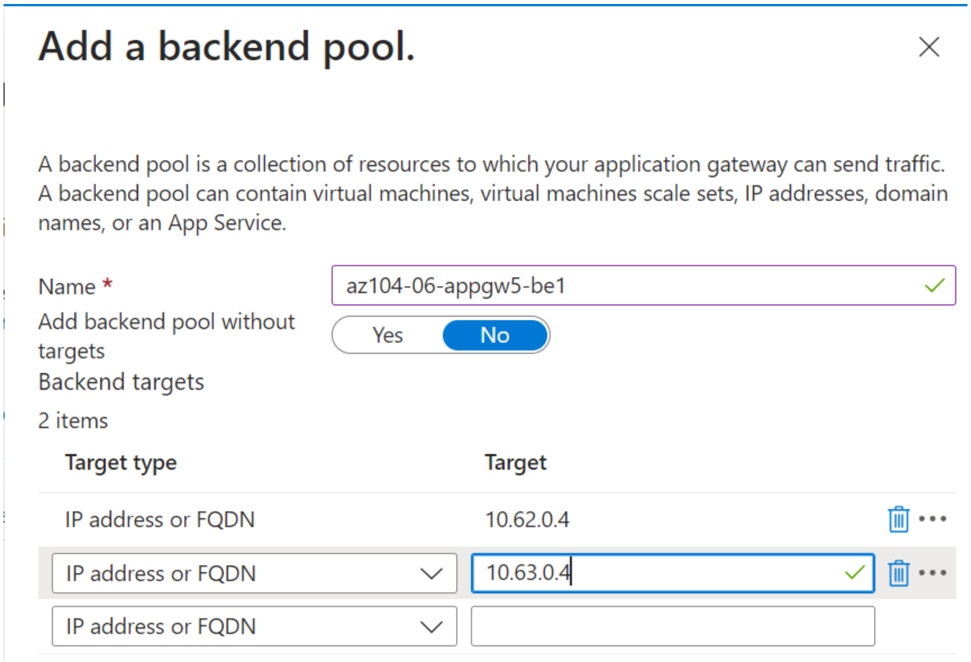
Traffic enters the application gateway via its frontend IP address(es). An application gateway can use a public IP address, private IP address, or one of each type.

Frontend IP address type ⓘ — ● Public ○ Private ○ Both

Public IP address * — (New) az104-06-pip5

Add new

Click Next: Backends > and then Add a backend pool. Specify the following settings (leave others with their default values). When completed click Add.



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 * az104-06-appgw5-be1 ✓

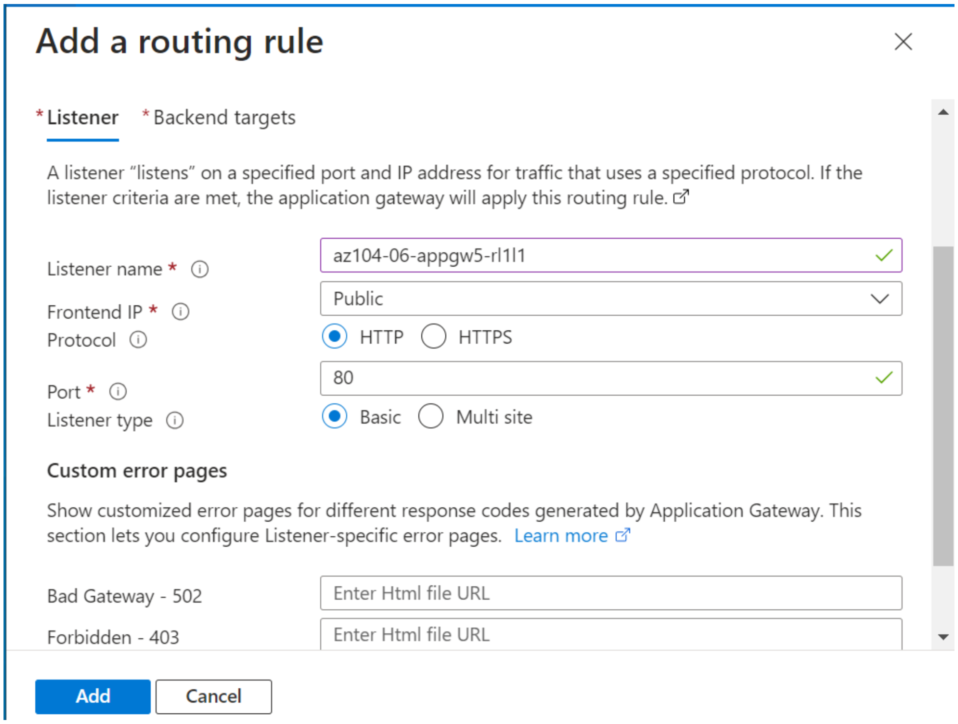
Add backend pool without targets ○ Yes ● No

Backend targets

2 items

Target type	Target
IP address or FQDN	10.62.0.4
IP address or FQDN	10.63.0.4 ✓
IP address or FQDN	

Click Next: Configuration > and then + Add a routing rule. Specify the following settings:



Add a routing rule

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 * az104-06-appgw5-rl1l1 ✓

Frontend IP * Public

Protocol ● HTTP ○ HTTPS

Port * 80 ✓

Listener type ● Basic ○ Multi site

Custom error pages

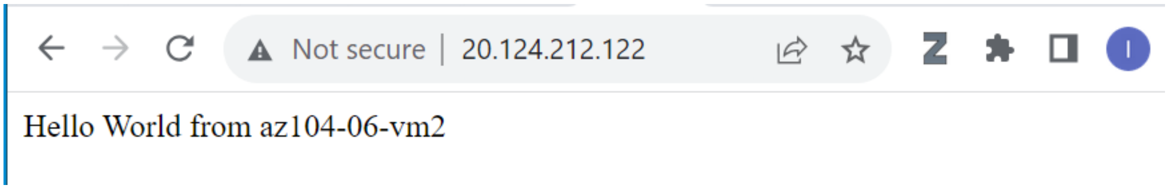
Show customized error pages for different response codes generated by Application Gateway. This section lets you configure Listener-specific error pages. Learn more

Bad Gateway - 502 Enter Html file URL

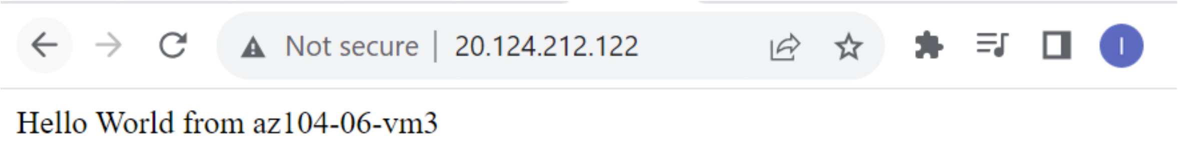
Forbidden - 403 Enter Html file URL

In the Azure portal, search and select Application Gateways and, on the Application Gateways blade, click az104-06-appgw5. On the az104-06-appgw5 Application Gateway blade, copy the value of the Frontend public IP address.

on another browser window and navigate to the IP address you identified in the previous step and verify that the browser window displays the message Hello World from az104-06-vm2 or Hello World from az104-06-vm3.



Hello World from az104-06-vm2



Hello World from az104-06-vm3

CONCLUSION

In conclusion, I gained practical experience in provisioning environments, configuring complex network topologies, testing connectivity, optimizing routing, and implementing advanced load balancing solutions. Additionally, I learned the use of custom deploy template method when I faced a challenge in task 1 step two. It was interesting to see the various methods of creating virtual machines.