

Homework 11 – Lab 06 - Implement Traffic Management

Velibor Stanisic

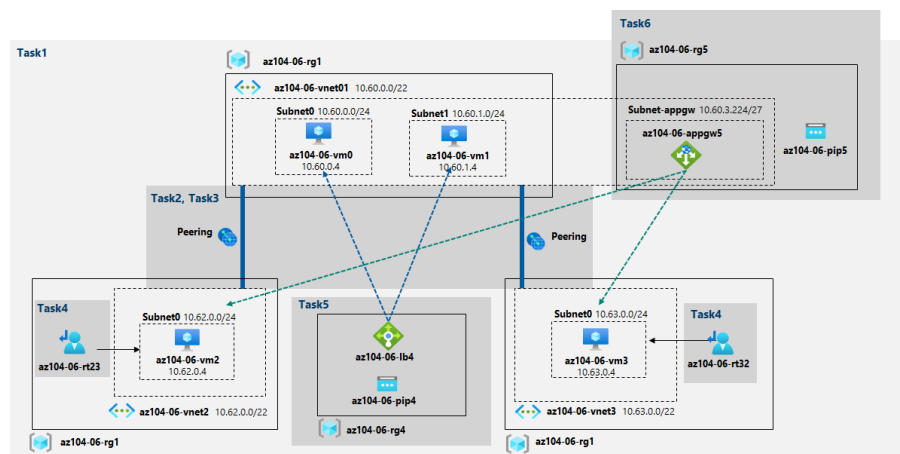
Lab scenario

You were tasked with testing managing network traffic targeting Azure virtual machines in the hub and spoke network topology, which Contoso considers implementing in its Azure environment (instead of creating the mesh topology, which you tested in the previous lab). This testing needs to include implementing connectivity between spokes by relying on user defined routes that force traffic to flow via the hub, as well as traffic distribution across virtual machines by using layer 4 and layer 7 load balancers. For this purpose, you intend to use Azure Load Balancer (layer 4) and Azure Application Gateway (layer 7).

Objectives

In this lab, you will:

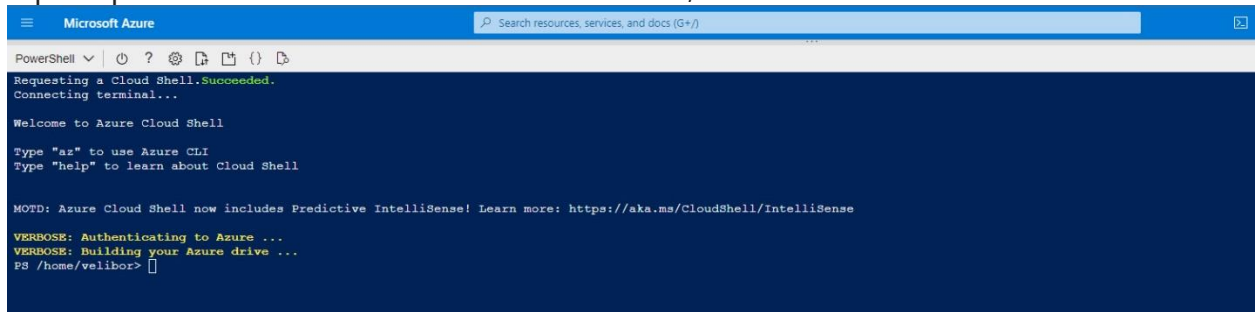
- Task 1: Provision the lab environment
- Task 2: Configure the hub and spoke network topology
- Task 3: Test transitivity of virtual network peering
- Task 4: Configure routing in the hub and spoke topology
- Task 5: Implement Azure Load Balancer
- Task 6: Implement Azure Application Gateway



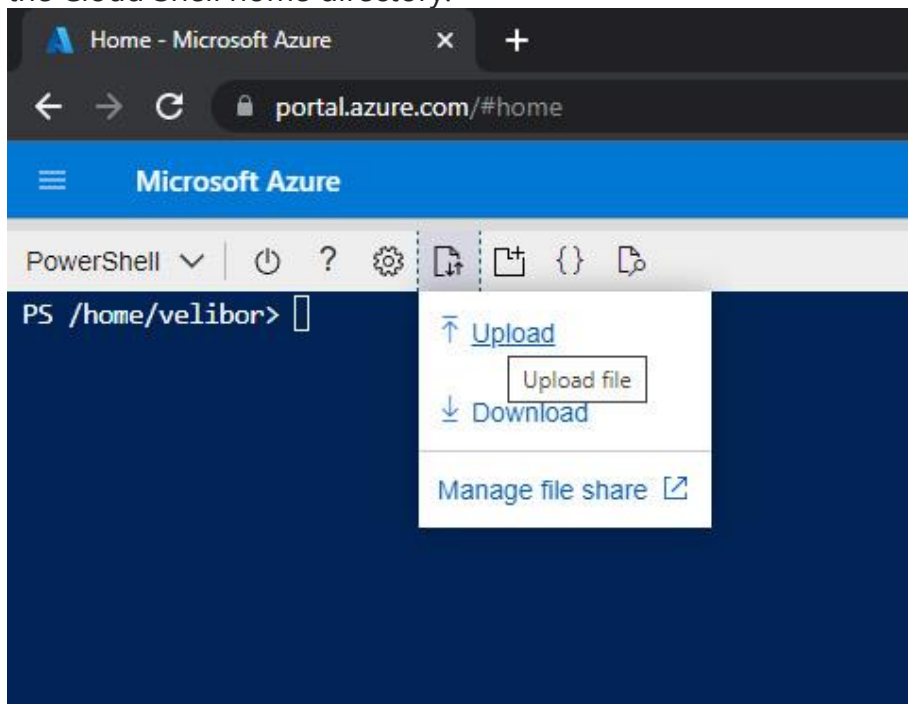
Task 1: Provision the lab environment

In this task, you will deploy four virtual machines into the same Azure region. The first two will reside in a hub virtual network, while each of the remaining two will reside in a separate spoke virtual network.

1. Sign in to the [Azure portal](#).
2. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.



4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\06\az104-06-vms-loop-template.json** and **\Allfiles\Labs\06\az104-06-vms-loop-parameters.json** into the Cloud Shell home directory.



5. Edit the **Parameters** file you just uploaded and change the password. If you need help editing the file in the Shell please ask your instructor for assistance. As a best practice, secrets, like passwords, should be more securely stored in the Key Vault.
6. From the Cloud Shell pane, run the following to create the first resource group that will be hosting the lab environment (replace the '[Azure_region]' placeholder with the name of an Azure region where you intend to deploy Azure virtual machines)(you can use the "(Get-AzLocation).Location" cmdlet to get the region list):

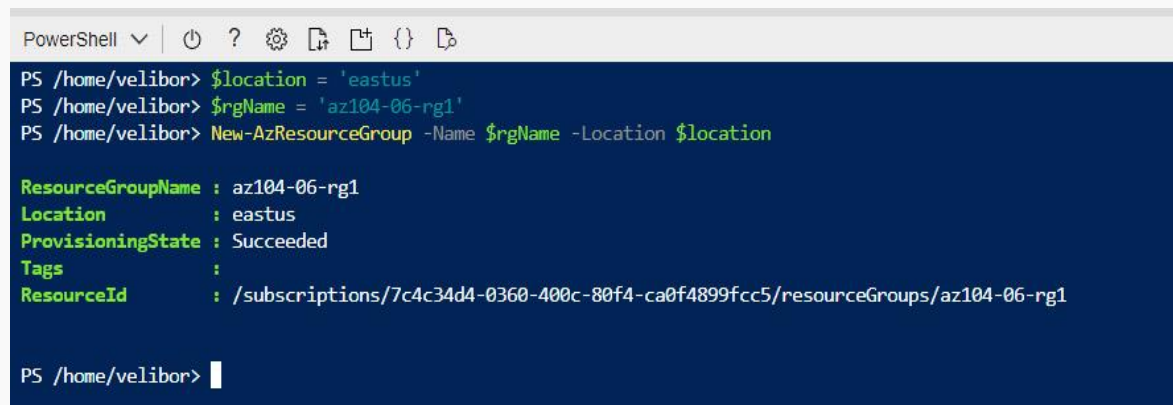
```
$location = 'eastus'
```

Now the resource group name:

```
$rgName = 'az104-06-rg1'
```

And finally create the resource group in your desired location:

```
New-AzResourceGroup -Name $rgName -Location $location
```



The screenshot shows a PowerShell terminal window with a dark blue background. The title bar reads "PowerShell" with a dropdown arrow and several icons. The terminal content shows the following commands and output:

```
PS /home/velibor> $location = 'eastus'
PS /home/velibor> $rgName = 'az104-06-rg1'
PS /home/velibor> New-AzResourceGroup -Name $rgName -Location $location

ResourceGroupName : az104-06-rg1
Location           : eastus
ProvisioningState  : Succeeded
Tags              :
ResourceId         : /subscriptions/7c4c34d4-0360-400c-80f4-ca0f4899fcc5/resourceGroups/az104-06-rg1

PS /home/velibor> 
```

7. From the Cloud Shell pane, run the following to create the three virtual networks and four Azure VMs into them by using the template and parameter files you uploaded:

```
New-AzResourceGroupDeployment `
-ResourceGroupName $rgName `
-TemplateFile $HOME/az104-06-vms-loop-template.json `
-TemplateParameterFile $HOME/az104-06-vms-loop-parameters.json
```

```
PowerShell | ? | [ ] | { } | [ ]
PS /home/velibor> New-AzResourceGroupDeployment `
>> -ResourceGroupName $rgName `
>> -TemplateFile $HOME/az104-06-vm-loop-template.json `
>> -TemplateParameterFile $HOME/az104-06-vm-loop-parameters.json

DeploymentName      : az104-06-vm-loop-template
ResourceGroupName   : az104-06-rg1
ProvisioningState    : Succeeded
Timestamp           : 3/22/2023 11:11:33 PM
Mode                 : Incremental
TemplateLink         :
Parameters           :
                    Name                Type                Value
                    =====
                    vmSize               String              "Standard_D2s_v3"
                    vmName               String              "az104-06-vm"
                    vmCount              Int                 4
                    adminUsername         String              "Student"
                    adminPassword         SecureString         null

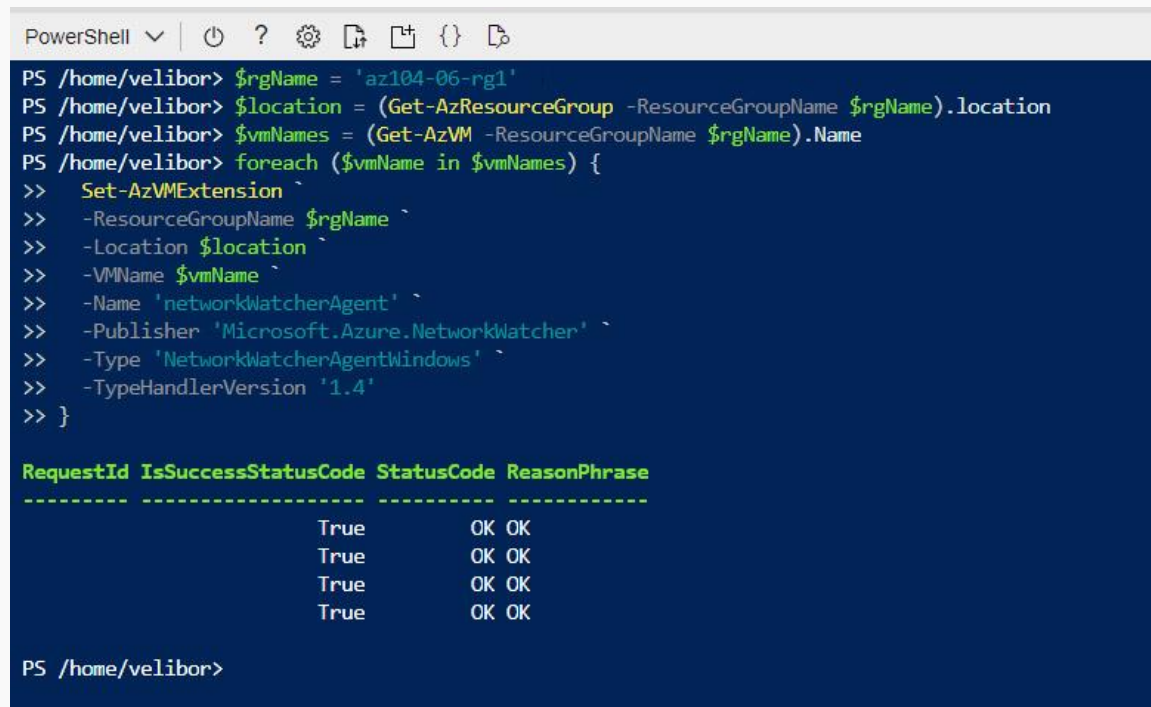
Outputs             :
DeploymentDebugLogLevel :

PS /home/velibor> 
```

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

```
$rgName = 'az104-06-rg1'
$location = (Get-AzResourceGroup -ResourceGroupName $rgName).location
$vmNames = (Get-AzVM -ResourceGroupName $rgName).Name

foreach ($vmName in $vmNames) {
    Set-AzVMExtension `
        -ResourceGroupName $rgName `
        -Location $location `
        -VMName $vmName `
        -Name 'networkWatcherAgent' `
        -Publisher 'Microsoft.Azure.NetworkWatcher' `
        -Type 'NetworkWatcherAgentWindows' `
        -TypeHandlerVersion '1.4'
}
```



```
PowerShell | [PowerShell Icon] [Help Icon] [Settings Icon] [Copy Icon] [Paste Icon] [Find Icon] [Close Icon]

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

RequestId IsSuccess StatusCode ReasonPhrase
-----
True OK OK
True OK OK
True OK OK
True OK OK

PS /home/velibor>
```

9. Close the Cloud Shell pane.

Task 2: Configure the hub and spoke network topology

In this task, you will configure local peering between the virtual networks you deployed in the previous tasks in order to create a hub and spoke network topology.

1. In the Azure portal, search for and select **Virtual networks**.
2. Review the virtual networks you created in the previous task.

Home >

Virtual networks

Default Directory

+ Create Manage view Refresh Export to CSV Open query Assign tags

Filter for any field... Subscription equals all Resource group equals all Location equals all Add filter

Showing 1 to 3 of 3 records.

Name	Resource group	Location
az104-06-vnet01	az104-06-rg1	East US
az104-06-vnet2	az104-06-rg1	East US
az104-06-vnet3	az104-06-rg1	East US

3. In the list of virtual networks, select **az104-06-vnet2**.
4. On the **az104-06-vnet2** blade, select **Properties**.
5. On the **az104-06-vnet2 | Properties** blade, record the value of the **Resource ID** property.

Home > Virtual networks > az104-06-vnet2

Virtual networks

Default Directory

+ Create Manage view

Filter for any field...

Name

- az104-06-vnet01
- az104-06-vnet2
- az104-06-vnet3

az104-06-vnet2 | Properties

Virtual network

Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
 - Address space
 - Connected devices
 - Subnets
 - Bastion
 - DDoS protection
 - Firewall
 - Microsoft Defender for Cloud
 - Network manager
 - DNS servers
 - Peerings
 - Service endpoints
 - Private endpoints
 - Properties

Name: az104-06-vnet2

Location: eastus

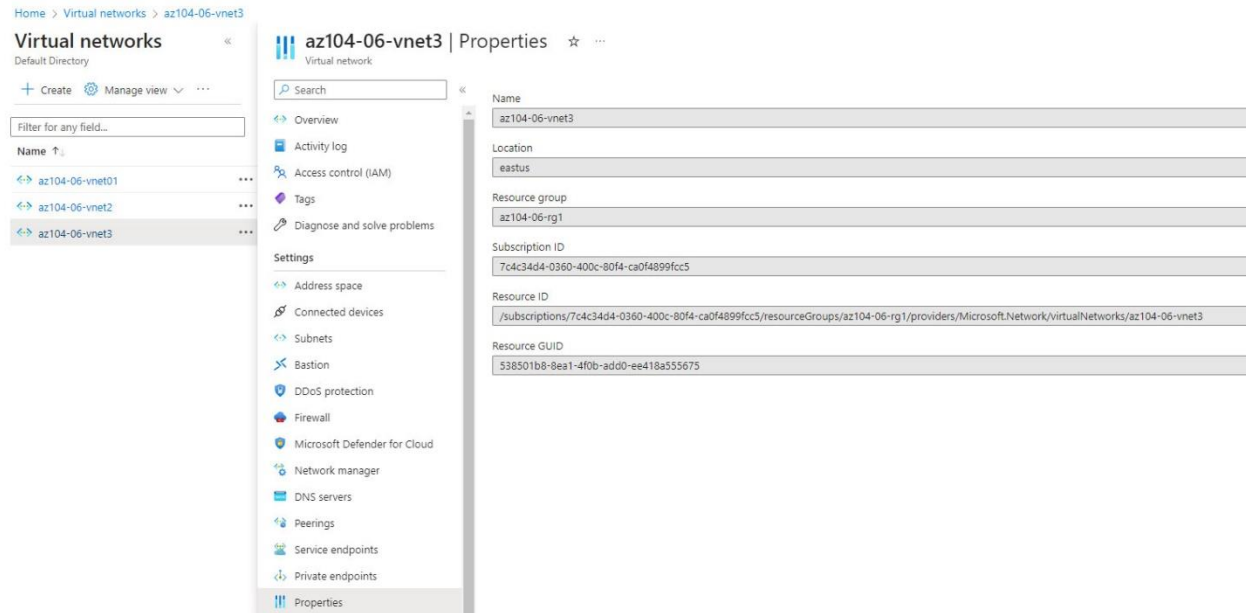
Resource group: az104-06-rg1

Subscription ID: 7c4c34d4-0360-400c-80f4-ca0f4899fcc5

Resource ID: /subscriptions/7c4c34d4-0360-400c-80f4-ca0f4899fcc5/resourceGroups/az104-06-rg1/providers/Microsoft.Network/virtualNetworks/az104-06-vnet2

Resource GUID: e83b28e6-40a4-4898-b5ce-aa12873ba155

- Navigate back to the list of virtual networks and select **az104-06-vnet3**.
- On the **az104-06-vnet3** blade, select **Properties**.
- On the **az104-06-vnet3 | Properties** blade, record the value of the **Resource ID** property.



- In the list of virtual networks, click **az104-06-vnet01**.
- On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
- Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	az104-06-vnet01_to_az104-06-vnet2
Traffic to remote virtual network	Allow (default)
Traffic forwarded from remote virtual network	Block traffic that originates from outside this virtual network
Virtual network gateway	None (default)
Remote virtual network: Peering link name	az104-06-vnet2_to_az104-06-vnet01
Virtual network deployment model	Resource manager
I know my resource ID	enabled
Resource ID	the value of resourceID parameter of az104-06-vnet2 you recorded earlier in this task
Traffic to remote virtual network	Allow (default)
Traffic forwarded from remote virtual network	Allow (default)
Virtual network gateway	None (default)

12. On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
13. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	az104-06-vnet01_to_az104-06-vnet3
Traffic to remote virtual network	Allow (default)
Traffic forwarded from remote virtual network	Block traffic that originates from outside this virtual network
Virtual network gateway	None (default)
Remote virtual network: Peering link name	az104-06-vnet3_to_az104-06-vnet01
Virtual network deployment model	Resource manager
I know my resource ID	enabled
Resource ID	the value of resourceID parameter of az104-06-vnet3 you recorded earlier in this task
Traffic to remote virtual network	Allow (default)
Traffic forwarded from remote virtual network	Allow (default)
Virtual network gateway	None (default)

Output:

The screenshot shows the Azure portal interface for the virtual network **az104-06-vnet01**. The **Peerings** section is selected in the left-hand navigation pane. The main area displays a table of peerings with the following data:

Name	Peering status	Peer	Gateway transit
az104-06-vnet01_to_az104-06-vnet2	Connected	az104-06-vnet2	Disabled
az104-06-vnet01_to_az104-06-vnet3	Connected	az104-06-vnet3	Disabled

The interface also includes a search bar at the top, a filter for "Peering status == all", and a list of settings on the left side of the main area.

Task 3: Test transitivity of virtual network peering

In this task, you will test transitivity of virtual network peering by using Network Watcher.

1. In the Azure portal, search for and select **Network Watcher**.
2. On the **Network Watcher** blade, expand the listing of Azure regions and verify the service is enabled in region you are using.
3. On the **Network Watcher** blade, navigate to the **Connection troubleshoot**.
4. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with the following settings (leave others with their default values):
5. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the connection was direct, with no intermediate hops in between the VMs.

The screenshot displays the Azure Network Watcher 'Connection troubleshoot' interface. On the left is a navigation pane with sections: Get started, Monitoring (Topology, Connection monitor (classic), Connection monitor, Network Performance Monitor), Network diagnostic tools (IP flow verify, NSG diagnostics, Next hop, Effective security rules, VPN troubleshoot, Packet capture, Connection troubleshoot), Metrics (Usage + quotas), and Logs (Flow logs, Diagnostic logs, Traffic Analytics). The main area shows test results for a connectivity check between two VMs.

Test	Status	Details	Suggestions
Connectivity Test	Success	Probes Sent: 66, Probes Failed: 0 Avg Latency: 1 ms Min Latency: 1 ms Min Latency: 2 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	-	-	-

Topology view

```
graph LR; VM0[az104-06-vm0  
10.60.0.4] -- "RTT: 2" --> NIC2[az104-06-nic2  
10.62.0.4];
```

6. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with the following settings (leave others with their default values):

7. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the connection was direct, with no intermediate hops in between the VMs.

The screenshot displays the Azure Network Watcher Connection Troubleshoot interface. On the left is a navigation pane with sections: Overview, Get started, Monitoring (Topology, Connection monitor (classic), Connection monitor, Network Performance Monitor), Network diagnostic tools (IP flow verify, NSG diagnostics, Next hop, Effective security rules, VPN troubleshoot, Packet capture, Connection troubleshoot), Metrics (Usage + quotas), and Logs (Flow logs, Diagnostic logs). The main area shows the results of a connectivity test. At the top, a summary bar indicates 'Probes Sent: 66, Probes Failed: 0, Avg Latency: 1 ms, Min Latency: 1 ms'. Below this is a table of test results:

Test Name	Status	Details	Result
Connectivity Test	Success	Probes Sent: 66, Probes Failed: 0, Avg Latency: 1 ms, Min 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

Below the table is a section titled 'Hop by hop details' with a table showing the path:

Name	Status	IP address	Next hop	RTT	Errors
az104-06-vm0	Success	10.60.0.4	10.63.0.4	2	-
az104-06-nic3	Success	10.63.0.4	-	-	-

At the bottom, the 'Topology view' shows a diagram with two nodes: 'az104-06-vm0' (IP 10.60.0.4) and 'az104-06-nic3' (IP 10.63.0.4), connected by a line with an arrow. The RTT is noted as 2.

8. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with the following settings (leave others with their default values):

The screenshot shows the configuration page for the Network Watcher Connection Troubleshoot tool. The left navigation pane is identical to the previous screenshot. The main area contains a description of the tool and a form for configuring a test:

Network Watcher connection troubleshoot provides the capability to check a direct TCP or ICMP connection from a virtual machine (VM), application gateway, 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 Pass - Sponsorship
- Resource group: az104-06-rg1
- Source type: Virtual machine
- Virtual machine: az104-06-vm2

Destination

- Destination type: ☒ Specify manually
- URI, FQDN or IP address: 10.63.0.4

Probe settings

- Protocol: ☒ TCP, ☐ ICMP
- Destination port: 3389
- Source port (optional):

9. Click **Check** and wait until results of the connectivity check are returned. Note that the status is **Unreachable**.

Task 4: Configure routing in the hub and spoke topology

In this task, you will configure and test 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.

1. In the Azure portal, search and select **Virtual machines**.
2. On the **Virtual machines** blade, in the list of virtual machines, click **az104-06-vm0**.
3. On the **az104-06-vm0** virtual machine blade, in the **Settings** section, click **Networking**.
4. 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**.
5. Set **IP forwarding** to **Enabled** and save the change.

The screenshot shows the Azure portal interface for the **az104-06-nic0** network interface. The breadcrumb navigation is **Home > Virtual machines > az104-06-vm0 | Networking > az104-06-nic0**. The page title is **az104-06-nic0 | IP configurations**. The left sidebar shows the **Settings** section with **IP configurations** selected. The main content area shows the **IP forwarding settings** with **IP forwarding** set to **Enabled** (indicated by a red 'Enabled' button). Below this, the **Virtual network** is **az104-06-vnet01** and the **Subnet** is **subnet0**. A table of **IP configurations** is shown below:

Name	IP Version	Type	Private IP address	Public IP address
ipconfig1	IPv4	Primary	10.60.0.4 (Dynamic)	-

6. In the Azure portal, navigate back to the **az104-06-vm0** Azure virtual machine blade and click **Overview**.
7. On the **az104-06-vm0** blade, in the **Operations** section, click **Run command**, and, in the list of commands, click **RunPowerShellScript**.

8. On the **Run Command Script** blade, type the following and click **Run** to install the Remote Access Windows Server role.

Run Command Script

RunPowerShellScript

i Script execution complete

PowerShell Script

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

Run

Output

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

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

Run Command Script

RunPowerShellScript

i Script execution complete

PowerShell Script

```
1 Install-WindowsFeature -Name Routing -IncludeManagementTools
2
3 Install-WindowsFeature -Name "RSAT-RemoteAccess-Powershell"
4
5 Install-RemoteAccess -VpnType RoutingOnly
6
7 Get-NetAdapter | Set-NetIPInterface -Forwarding Enabled
```

Run

Output

Success	Restart Needed	Exit Code	Feature Result
True	No	Success	{RAS Connection Manager Administr
True	No	NoChangeNeeded	{}

10. In the Azure portal, search and select **Route tables** and, on the **Route tables** blade, click **+ Create**.
11. Create a route table with the following settings (leave others with their default values):

[Home](#) > [Route tables](#) >

Create Route table

Basics Tags Review + create

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 Pass - Sponsorship

Resource group * ⓘ az104-06-rg1 [Create new](#)


Instance details

Region * ⓘ East US

Name * ⓘ az104-06-rt23 ✓

Propagate gateway routes * ⓘ ☐ Yes ☒ No

12. Click **Review and Create**. Let validation occur, and click **Create** to submit your deployment.
13. Click **Go to resource**.

 **Microsoft.RouteTable-20230323033545** | Overview ⓘ ...

Deployment

Search « Delete Cancel Redeploy Download Refresh

Overview Inputs Outputs Template

✓ Your deployment is complete

Deployment name: Microsoft.RouteTable-20230323033545
Subscription: [Azure Pass - Sponsorship](#)
Resource group: [az104-06-rg1](#)

Start time: 3/23/2023, 3:36:34 AM
Correlation ID: e9fd0c81-2f7d-4290-b633-04b2d0d8161d ⓘ

Deployment details

Next steps

[Go to resource](#)

Give feedback
[Tell us about your experience with deployment](#)

14. On the **az104-06-rt23** route table blade, in the **Settings** section, click **Routes**, and then click **+ Add**.

15. Add a new route with the following settings:
16. Click **Go to resource**.
17. On the **az104-06-rt23** route table blade, in the **Settings** section, click **Routes**, and then click **+ Add**.
18. Add a new route with the following settings:
19. Click **Add**
20. Navigate back to **Route tables** blade and click **+ Create**.
21. Create a route table with the following settings (leave others with their default values):

[Home](#) > [Route tables](#) >

Create Route table ...

Basics Tags Review + create

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 Pass - Sponsorship

Resource group * ⓘ

az104-06-rg1

[Create new](#)

Instance details

Region * ⓘ

East US

Name * ⓘ

az104-06-rt32 ✓

Propagate gateway routes * ⓘ

☐ Yes

☒ No

22. Click Review and Create. Let validation occur, and hit Create to submit your deployment.
23. Click **Go to resource**.
24. On the **az104-06-rt32** route table blade, in the **Settings** section, click **Routes**, and then click **+ Add**.

25. Add a new route with the following settings:

Add route

az104-06-rt32

Route name *

az104-06-route-vnet3-to-vnet2

Destination address prefix *

IP Addresses

Destination IP addresses/CIDR ranges *

10.62.0.0/20

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 IP forwarding by navigating to the respective network interface's IP address settings.

26. Click **OK**

27. Back on the **az104-06-rt32** route table blade, in the **Settings** section, click **Subnets**, and then click **+ Associate**.

28. Associate the route table **az104-06-rt32** with the following subnet:

Home > Microsoft.RouteTable-20230323033924 | Overview > az104-06-rt32

<> az104-06-rt32 | Subnets ☆ ...

Route table

Search

+ Associate

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Configuration

Routes

<> Subnets

Properties

Locks

Monitoring

Alerts

Automation

Search subnets

Name ↑↓	Address range ↑↓	Virtual network ↑↓
subnet0	10.63.0.0/24	az104-06-vnet3

29. Click **OK**

30. In the Azure portal, navigate back to the **Network Watcher - Connection troubleshoot** blade.

31. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with the following settings (leave others with their default values):
32. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the traffic was routed via **10.60.0.4**, assigned to the **az104-06-nic0** network adapter. If status is **Unreachable**, you should stop and then start az104-06-vm0.

Get started

Monitoring

Topology

Connection monitor (classic)

Connection monitor

Network Performance Monitor

Network diagnostic tools

IP flow verify

NSG diagnostics

Next hop

Effective security rules

VPN troubleshoot

Packet capture

Connection troubleshoot

Metrics

Usage + quotas

Logs

Flow logs

URI, FQDN or IP address *

10.63.0.4

Probe settings

Protocol

TCP

ICMP

Destination port *

3389

Source port (optional)

Connection diagnostic

Diagnostics tests *

4 selected

Run diagnostic tests

Diagnostics details

Source

az104-06-vm2

Destination

10.63.0.4

Diagnostics tests

Test	Status	Details	Suggestions
Connectivity Test	Success	Probes Sent: 66 ,Probes Failed: 0 Avg Latency: 4 ms Min Latency: 3 ms	None

Task 5: Implement Azure Load Balancer

In this task, you will implement an Azure Load Balancer in front of the two Azure virtual machines in the hub virtual network.

1. In the Azure portal, search for and select **Load balancers** and, on the **Load balancers** blade, click **+ Create**.
2. Create a load balancer with the following settings (leave others with their default values) then click **Next : Frontend IP configuration**:

Microsoft Azure

Search resources, services, and docs (G+/)

[Home](#) > [Load balancing](#) | [Load Balancer](#) >

Create load balancer

Basics

Frontend IP configuration

Backend pools

Inbound rules

Outbound rules

Tags

Review + create

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more](#).

Project details

Subscription *
Azure Pass - Sponsorship

Resource group *
(New) az104-06-rg4
[Create new](#)

Instance details

Name *

Region *
East US

SKU * ⓘ
☒ Standard
☐ Gateway
☐ Basic

i Microsoft recommends Standard SKU load balancer for production workloads. [Learn more about pricing differences between Standard and Basic SKU](#) ⓘ

Type * ⓘ
☒ Public
☐ Internal

Tier *
☒ Regional
☐ Global

Review + create

< Previous

Next : Frontend IP configuration >

[Download a template for automation](#) [Give feedback](#)

- On the **Frontend IP configuration** tab, click **Add a frontend IP configuration** and use the following settings before clicking **OK** and then **Add**. When completed click **Next: Backend pools**.

The screenshot shows the 'Create load balancer' page in the Microsoft Azure portal. The breadcrumb navigation indicates the path: 'Load balancing | Load Balancer >'. The page title is 'Create load balancer'. Below the title, there are tabs for 'Frontend IP configuration', 'Backend pools', 'Inbound rules', 'Outbound rules', 'Tags', and 'Review + create'. The 'Frontend IP configuration' tab is active. A descriptive text explains that a load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. It mentions a hash-based distribution algorithm and that load balancers can be internet-facing or internal. A link 'Learn more' is provided. The 'Configuration details' section includes a 'Description' field with the value 'Azure Pass - Sponsorship' and a 'Resource group' dropdown menu showing '(New) az104-06-rg4' with a 'Create new' link. The 'Location details' section includes a 'Region' dropdown menu showing 'East US'. Below the region, there are radio buttons for 'Standard', 'Gateway', and 'Basic', with 'Standard' selected. A blue information box states: 'Microsoft recommends Standard SKU load balancer for production workloads. Learn more about pricing differences between Standard and Basic SKU'. Below this, there are radio buttons for 'Public', 'Internal', 'Regional', and 'Global', with 'Regional' selected. At the bottom, there are buttons for 'Review + create', '< Previous', 'Next : Frontend IP configuration >', 'Download a template for automation', and 'Give feedback'.

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

> Load balancing | Load Balancer >

Create load balancer

Frontend IP configuration Backend pools Inbound rules Outbound rules Tags Review + create

A load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers use a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more.](#)

Configuration details

Description * Azure Pass - Sponsorship

Resource group * (New) az104-06-rg4 [Create new](#)

Location details

Region * East US

☒ Standard
☐ Gateway
☐ Basic

i Microsoft recommends Standard SKU load balancer for production workloads. [Learn more about pricing differences between Standard and Basic SKU](#)

☒ Public
☐ Internal
☒ Regional
☐ Global

Review + create < Previous Next : Frontend IP configuration > Download a template for automation Give feedback

- 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.**

Microsoft Azure

Search resources, services, and docs (G+ /)

Home > Load balancing | Load Balancer >

Create load balancer

...

Basics

Frontend IP configuration

Backend pools

Inbound rules

Outbound rules

Tags

Review + create

A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual m

+ Add a backend pool

Name	Virtual network	Resource Name
az104-06-lb4-be1		
az104-06-lb4-be1	az104-06-vnet01	az104-06-vm0
az104-06-lb4-be1	az104-06-vnet01	az104-06-vm1

Review + create

< Previous

Next : Inbound rules >

Download a template for automation

Give feedback

- On the **Inbound rules** tab, click **Add a load balancing rule**. Add a load balancing rule with the following settings (leave others with their default values). When completed click **Add**.

Create load balancer ...

Basics Frontend IP configuration Backend pools **Inbound rules** Outbound rules Tags Review + create




Load balancing rule
A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. The load balancing rule uses a health probe to determine which backend instances are eligible to receive traffic.

+ Add a load balancing rule






Name ↑	Frontend IP configuration ↑	Backend pool ↑	Health probe ↑	Frontend Port ↑	Backend port ↑
az104-06-lb4-lbrule1	az104-06-pip4	az104-06-lb4-be1	az104-06-lb4-hp1	80	80


Inbound NAT rule
An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.


- As you have time, review the other tabs, then click **Review and create**. Ensure there are no validation errors, then click **Create**.
- Wait for the load balancer to deploy then click **Go to resource**.


 **Microsoft.LoadBalancer-20230323040843** | Overview  


Deployment


<<  Delete  Cancel  Redeploy  Download  Refresh


 Overview

 Inputs

 Outputs

 Template

 **Your deployment is complete**


 Deployment name: Microsoft.LoadBalancer-20230323040843
Subscription: [Azure Pass - Sponsorship](#)
Resource group: [az104-06-rg4](#)

▼ Deployment details

^ Next steps

[Go to resource](#)

Give feedback

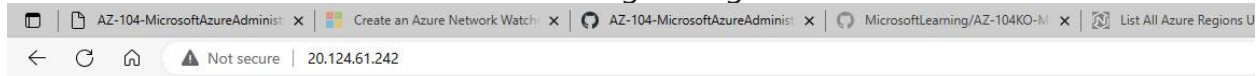
 [Tell us about your experience with deployment](#)

- Select **Frontend IP configuration** from the Load Balancer resource page. Copy the IP address.

9. 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**.



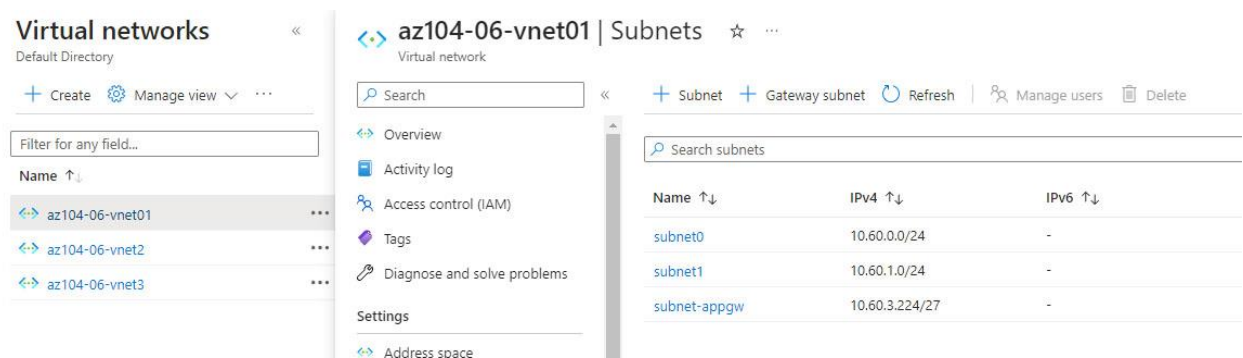
10. Refresh the window to verify the message changes to the other virtual machine. This demonstrates the load balancer rotating through the virtual machines.



Task 6: Implement Azure Application Gateway

In this task, you will implement an Azure Application Gateway in front of the two Azure virtual machines in the spoke virtual networks.

1. In the Azure portal, search and select **Virtual networks**.
2. On the **Virtual networks** blade, in the list of virtual networks, click **az104-06-vnet01**.
3. On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Subnets**, and then click **+ Subnet**.
4. Add a subnet with the following settings (leave others with their default values):
5. Click **Save**



6. In the Azure portal, search and select **Application Gateways** and, on the **Application Gateways** blade, click **+ Create**.

7. On the **Basics** tab, specify the following settings (leave others with their default values):

Create application gateway

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

An application gateway is a web traffic load balancer that enables you to manage traffic to your web application. [Learn more about 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 * ⓘ Azure Pass - Sponsorship ▼

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 ▼

HTTP2 ⓘ ☒ Disabled ☐ Enabled

Configure virtual network

Virtual network * ⓘ az104-06-vnet01 ▼
[Create new](#)

Subnet * ⓘ subnet-appgw (10.60.3.224/27) ▼
[Manage subnet configuration](#)

Previous Next : Frontends >

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

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 * (New) az104-06-pip5 ▼
[Add new](#)

9. 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 * ✓

Add backend pool without targets ☐ Yes ☒ No

Backend targets

2 items

Target type	Target	
IP address or FQDN	10.62.0.4	...
<input type="text" value="IP address or FQDN"/>	<input type="text" value="10.63.0.4"/> ✓	...
<input type="text" value="IP address or FQDN"/>	<input type="text"/>	

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

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 * ✓

Priority * ✓

* 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 * ✓

Frontend IP * ✓

Protocol ☒ HTTP ☐ HTTPS

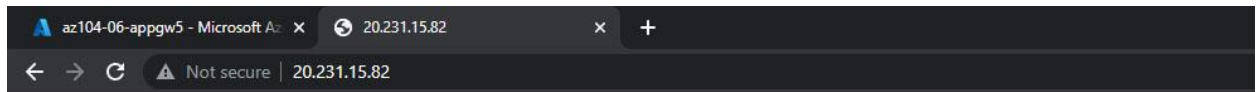
Port * ✓

Additional settings

Listener type ☒ Basic ☐ Multi site

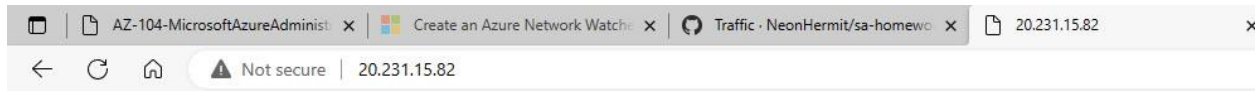
Error page url ☐ Yes ☒ No

11. Switch to the **Backend targets** tab and specify the following settings (leave others with their default values). When completed click **Add** (twice).
12. Click **Next: Tags >**, followed by **Next: Review + create >** and then click **Create**.
13. In the Azure portal, search and select **Application Gateways** and, on the **Application Gateways** blade, click **az104-06-appgw5**.
14. On the **az104-06-appgw5** Application Gateway blade, copy the value of the **Frontend public IP address**.
15. Start another browser window and navigate to the IP address you identified in the previous step.
16. 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

17. Refresh the window to verify the message changes to the other virtual machine.



Hello World from az104-06-vm3