

# **Lab 06 - Implement Traffic Management Report**

**Yunis Mohamed**

MICROSOFT AZURE LAB 06

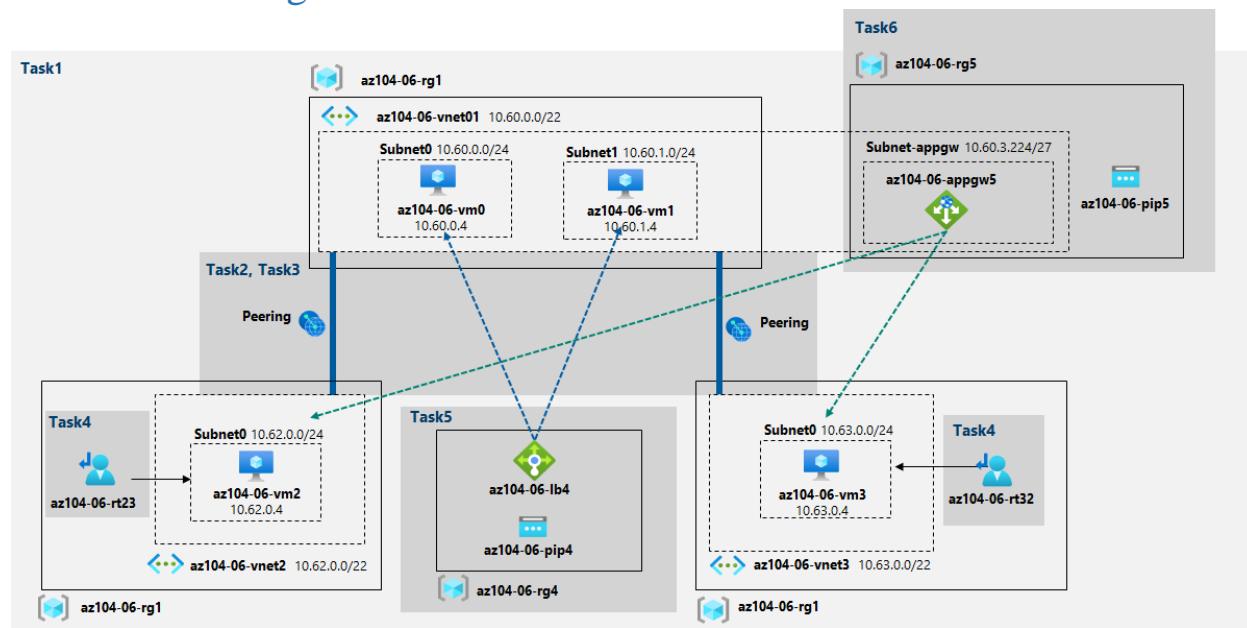
## Table of Contents

Introduction.....	2
Architecture diagram .....	2
Task 1: Provision the lab environment .....	2
Task 2: Configure the hub and spoke network topology .....	4
Task 3: Test transitivity of virtual network peering.....	8
Task 4: Configure routing in the hub and spoke topology.....	11
Task 5: Implement Azure Load Balancer .....	18
Task 6: Implement Azure Application Gateway .....	21
Conclusion .....	26

# Introduction

The Implement Traffic Management Lab 06 in Azure comprehensive lab provides an environment to practice and gain knowledge on cloud Traffic Management. Through a series of tasks designed in the lab I will be able to complete the Lab's objective. Task 1 focuses on provisioning the lab environment, ensuring I have everything necessary to proceed. Task 2 involves configuring the hub and spoke network topology to establish a robust and scalable network foundation. Task 3 tests the connectivity between different virtual networks using virtual network peering. Task 4 involves optimizing the flow of network traffic by configuring routing in the hub and spoke topology. Task 5 provides an opportunity to implement Azure Load Balancer, a high-performance load balancing solution. Finally, in Task 6, I will explore Azure Application Gateway, a powerful application delivery controller. By completing these tasks, I hope to gain valuable knowledge and hands-on experience in implementing traffic management techniques in Azure.

## Architecture diagram



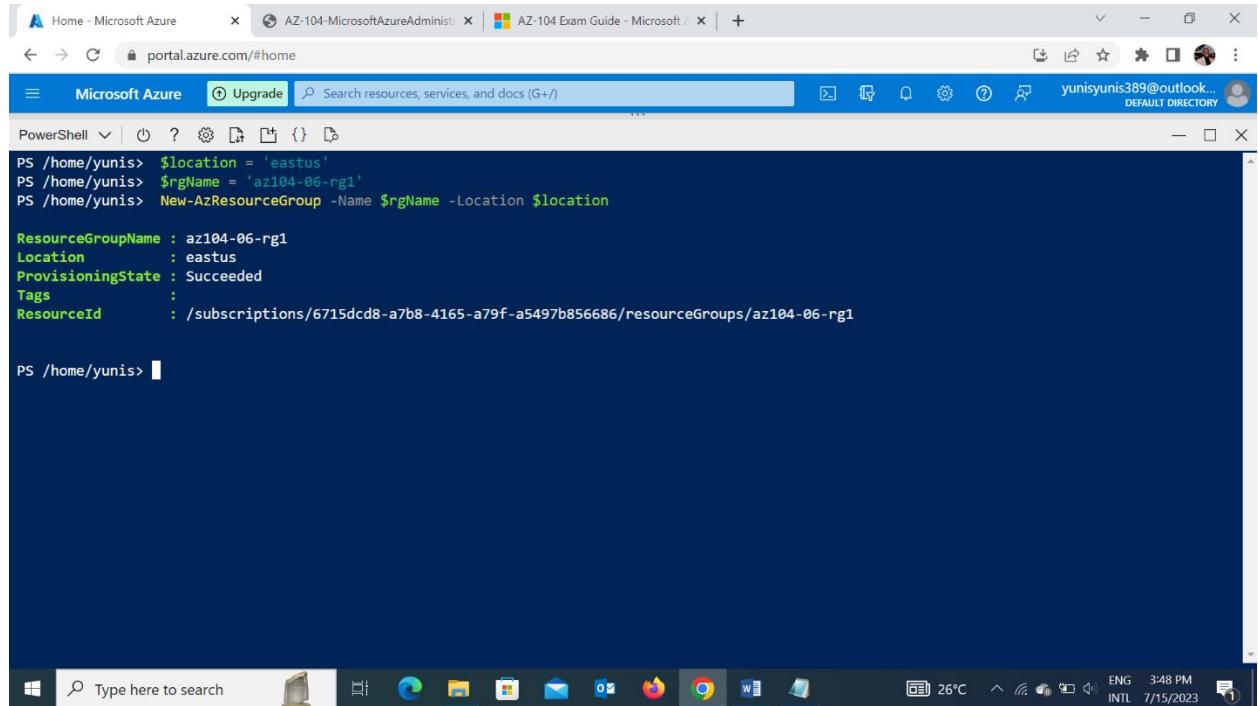
## Task 1: Provision the lab environment

In this task, I 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\Jobs\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 (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):

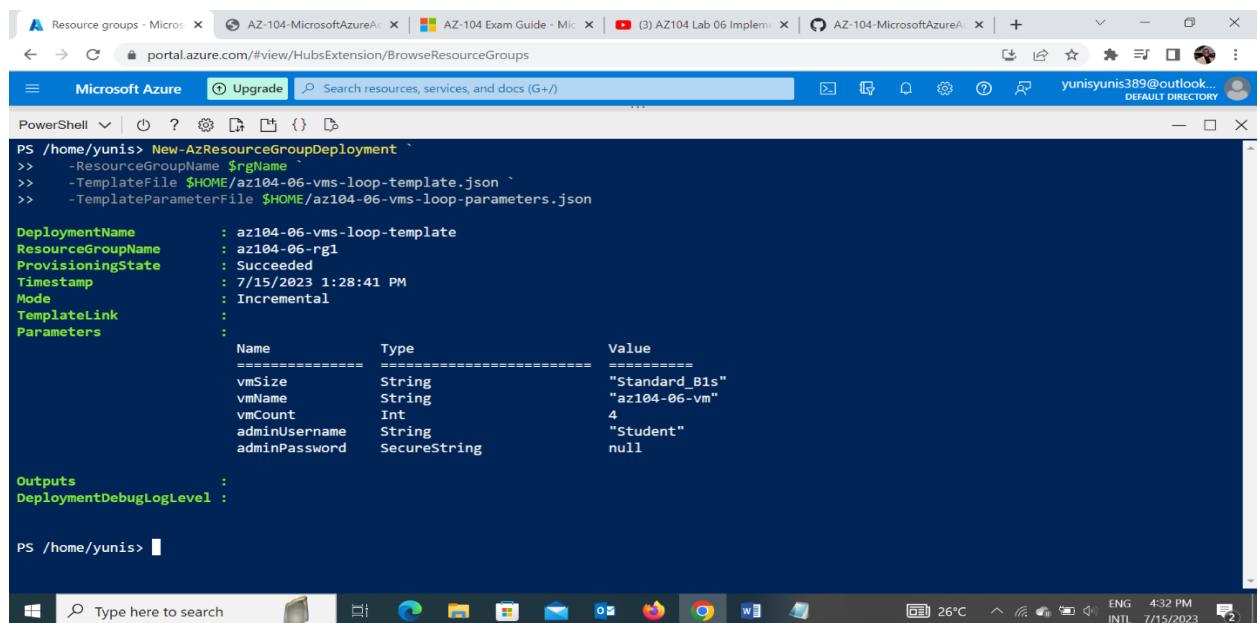


```
PS /home/yunis> $location = 'eastus'
PS /home/yunis> $rgName = 'az104-06-rg1'
PS /home/yunis> New-AzResourceGroup -Name $rgName -Location $location

ResourceGroupName : az104-06-rg1
Location         : eastus
ProvisioningState: Succeeded
Tags             :
ResourceId       : /subscriptions/6715dc8-a7b8-4165-a79f-a5497b856686/resourceGroups/az104-06-rg1

PS /home/yunis>
```

- 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:



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

DeploymentName      : az104-06-vms-loop-template
ResourceGroupName   : az104-06-rg1
ProvisioningState   : Succeeded
Timestamp          : 7/15/2023 1:28:41 PM
Mode               : Incremental
TemplateLink       :
Parameters          :
    Name           Type           Value
    ======        ======        ======
    vmSize         String        "Standard_B1s"
    vmName         String        "az104-06-vm"
    vmCount        Int          4
    adminUsername  String        "Student"
    adminPassword  SecureString null

Outputs            :
DeploymentLogLevel :
```

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

- #### 8. Close the Cloud Shell pane.

## Task 2: Configure the hub and spoke network topology

In this task, I 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.
  3. In the list of virtual networks, select **az104-06-vnet2**.

In the list of Virtual networks, select az104-06-vnet01.

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes tabs for Virtual networks - Micros... (selected), AZ-104-MicrosoftAzure..., AZ-104 Exam Guide - Mi..., (3) AZ104 Lab 06 Implement..., and AZ-104-MicrosoftAzure... . Below the navigation bar is a search bar with placeholder text "Search resources, services, and docs (G+)" and a "Microsoft Azure" logo. The main content area is titled "Virtual networks" with a "Default Directory" link. It features a toolbar with "Create", "Manage view", "Refresh", "Export to CSV", "Open query", and "Assign tags" buttons. Below the toolbar are filter options: "Subscription equals all", "Resource group equals all", "Location equals all", and a "Add filter" button. The main table displays three records:

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

At the bottom, there are navigation links for < Previous, Page 1 of 1, Next >, and a "Give feedback" link. The status bar at the bottom right shows the date and time as 5:14 PM INTL 7/15/2023.

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.

The screenshot shows the Microsoft Azure portal interface. The left sidebar lists 'Virtual networks' with items: az104-06-vnet0, az104-06-vnet2, and az104-06-vnet3. The right pane displays the properties for 'az104-06-vnet2'. The 'Properties' tab is selected. The 'Resource ID' field contains the value: /subscriptions/6715dcdb8-a7b8-4165-a79f-a5497b856686/resourceGroups/az104-06-rg1/providers/Microsoft... . A 'Copy' button is visible next to it, with a 'Copied' message displayed. Other fields include Name (az104-06-vnet2), Location (eastus), Resource group (az104-06-rg1), Subscription ID (6715dcdb8-a7b8-4165-a79f-a5497b856686), and Resource GUID (a0f3b75f-9849-43db-9f24-415438e955da).

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

The screenshot shows the Microsoft Azure portal interface. The left sidebar lists 'Virtual networks' with items: az104-06-vnet0, az104-06-vnet2, and az104-06-vnet3. The right pane displays the properties for 'az104-06-vnet3'. The 'Properties' tab is selected. The 'Resource ID' field contains the value: /subscriptions/6715dcdb8-a7b8-4165-a79f-a5497b856686/resourceGroups/az104-06-rg1/providers/Microsoft... . A 'Copy' button is visible next to it, with a 'Copied' message displayed. Other fields include Name (az104-06-vnet3), Location (eastus), Resource group (az104-06-rg1), Subscription ID (6715dcdb8-a7b8-4165-a79f-a5497b856686), and Resource GUID (ccc3f0c9-7432-4640-8bc9-10e9aa37340e).

9. In the list of virtual networks, click **az104-06-vnet01**.

10. On the **az104-06-vnet01** virtual network blade, in the Settings section, click Peerings and then click + Add.

11. Add a peering with the following settings (leave others with their default values) and click Add:

Add peering

This virtual network

Peering link name \*

az104-06-vnet01\_to\_az104-06-vnet2

Traffic to remote virtual network ⓘ

Allow (default)

Block all traffic to the remote virtual network

Traffic forwarded from remote virtual network ⓘ

Allow (default)

Block traffic that originates from outside the remote virtual network

Virtual network gateway or Route Server ⓘ

Use this virtual network's gateway or Route Server

Use the remote virtual network's gateway or Route Server

None (default)

Remote virtual network

Add

Type here to search

25°C 5:24 PM INTL 7/15/2023

Add peering

Virtual network deployment model ⓘ

Resource manager

Classic

I know my resource ID ⓘ

Resource ID \*

7b8-4165-a79f-a5497b856686/resourceGroups/az104-06-rg1/providers/Microsoft.Network/virtualNetworks/az104-06-vne

Traffic to remote virtual network ⓘ

Allow (default)

Block all traffic to the remote virtual network

Traffic forwarded from remote virtual network ⓘ

Allow (default)

Block traffic that originates from outside the remote virtual network

Virtual network gateway or Route Server ⓘ

Use this virtual network's gateway or Route Server

Add

Type here to search

25°C 5:24 PM INTL 7/15/2023

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:

The screenshot shows the Azure portal interface for adding a peering link. The top navigation bar includes tabs for 'Add peering - Microsoft', 'AZ-104-MicrosoftAzureA...', 'AZ-104 Exam Guide - Mic...', '(3) AZ104 Lab 06 Implement...', 'AZ-104-MicrosoftAzureA...', and a search bar. The user is signed in as 'yunisyunis389@outlook...'. The main content area is titled 'Add peering' under 'az104-06-vnet01'. It has sections for '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 the remote virtual network), 'Virtual network gateway or Route Server' (None (default)), and 'Remote virtual network' (Peer ID: '/subscriptions/6715dcdb-a7b8-4165-a79f-a5497b856686/resourceGroups/az104-06-rg1/providers/Microsoft.Network/virtualNetworks/az104-06-vnet3'). At the bottom is a blue 'Add' button.

## Task 3: Test transitivity of virtual network peering

In this task, I 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.

Name	Subscription	Location
NetworkWatcher_eastus	free Trial	East US
NetworkWatcher_francecentral	free Trial	France Central
NetworkWatcher_norwayeast	free Trial	Norway East

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): **10.62.0.4** represents the private IP address of **az104-06-vm2**

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

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

6. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with the following settings (leave others with their default values): **10.63.0.4** represents the private IP address of **az104-06-vm3**

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

Test	Status	Details	Suggestions
Connectivity Test	Success	Probes Sent: 66 , Probes Failed: 0 Avg Latency: 1 ms Min Latency: 1 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: VirtualNetworkPeering Route Table Id: System Route	None

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

- Click **Run diagnostic tests** and wait until results of the connectivity check are returned.  
Note that the status is **Fail**. *This is expected, since the two spoke virtual networks are not peered with each other (virtual network peering is not transitive).*

**Diagnostic details**

Source	Destination
az104-06-vm2	10.63.0.4

**Diagnostic tests**

Test	Status	Details	Suggestions
Connectivity Test	Fail	Probes Sent: 30 ,Probes Failed: 30	-
NSG Outbound (from source)	Fail	There are failed tests in the following NSGs: • az104-06-nsg2	<a href="#">Go to VM &gt; Update the networking rule</a> <a href="#">Read docs</a>
Next Hop (from source)	Success	Next Hop Type: None Route Table Id: System Route	None

**Hop by hop details**

Name	Status	IP address	Next hop	RTT	Errors

## Task 4: Configure routing in the hub and spoke topology

In this task, I 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.

- In the Azure portal, search and select **Virtual machines**.
- 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. *This setting is required in order for az104-06-vm0 to function as a router, which will route traffic between two spoke virtual networks.*

az104-06-nic0 - Microsoft Edge

portal.azure.com/@yunisunis389outlook.onmicrosoft.com/resource/subscriptions/6715dcdb-a7b8-4165-a79f-a5497b856686/resourceGroups...

Microsoft Azure

az104-06-vm0 | Networking

az104-06-nic0 | IP configurations

Network interface

IP Settings

Enable IP forwarding

Virtual network: az104-06-vnet01

Subnet: subnet0 (10.60.0.0/24) 250 free IP addresses

Private and public IP addresses can be assigned to a virtual machine's network interface controller. You can add as many private and public IPv4 addresses as necessary to a network interface, within the limits listed in the Azure limits article. [Learn more](#)

Add Make primary Delete

Name	IP Version	Type	Private IP Address	Public IP Address
az104-06-nic0	IPv4	Ethernet	10.60.0.2	

Apply Discard changes

Type here to search

25°C ENG INTL 6:31 PM 7/15/2023

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.

Run Command Script - Microsoft Edge

portal.azure.com/@yunisunis389outlook.onmicrosoft.com/resource/subscriptions/6715dcdb-a7b8-4165-a79f-a5497b856686/resourceGroups...

Microsoft Azure

az104-06-vm0 | Networking

Run Command Script

RunPowerShellScript

PowerShell Script

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

Run

Virtual machines

az104-06-vm0

az104-06-vm1

az104-06-vm2

az104-06-vm3

Filter for any field...

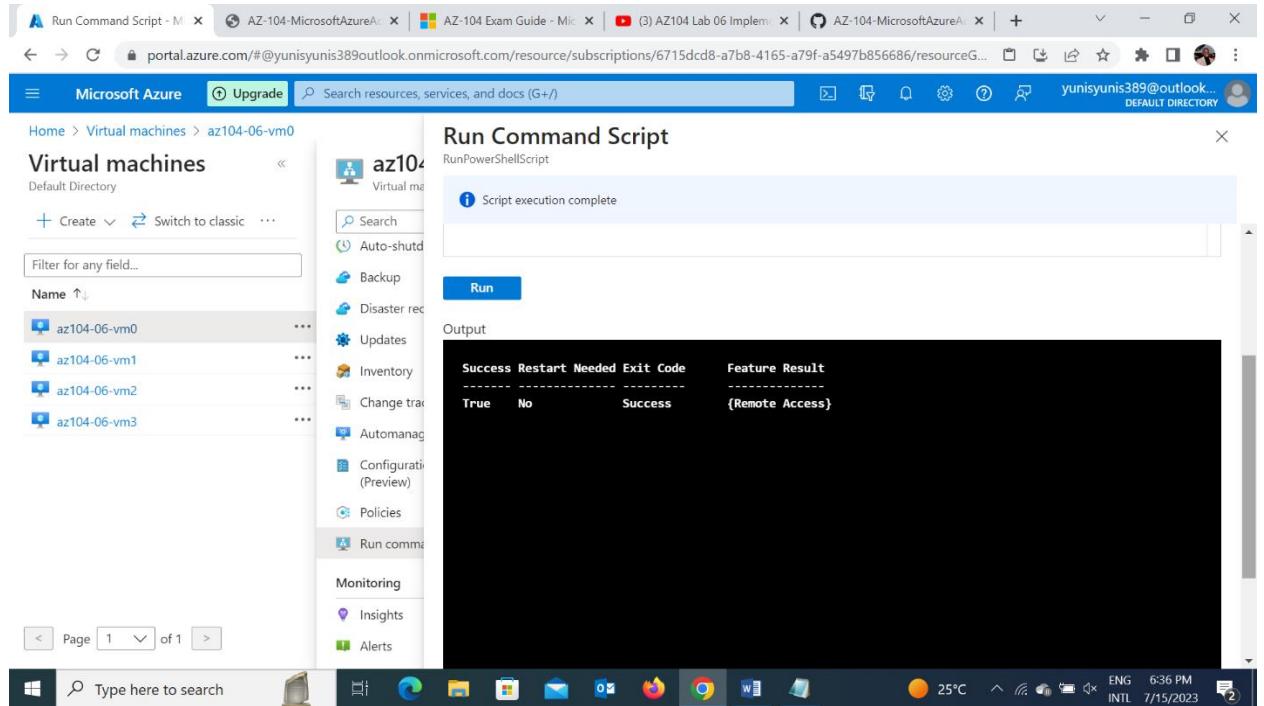
Name

Page 1 of 1

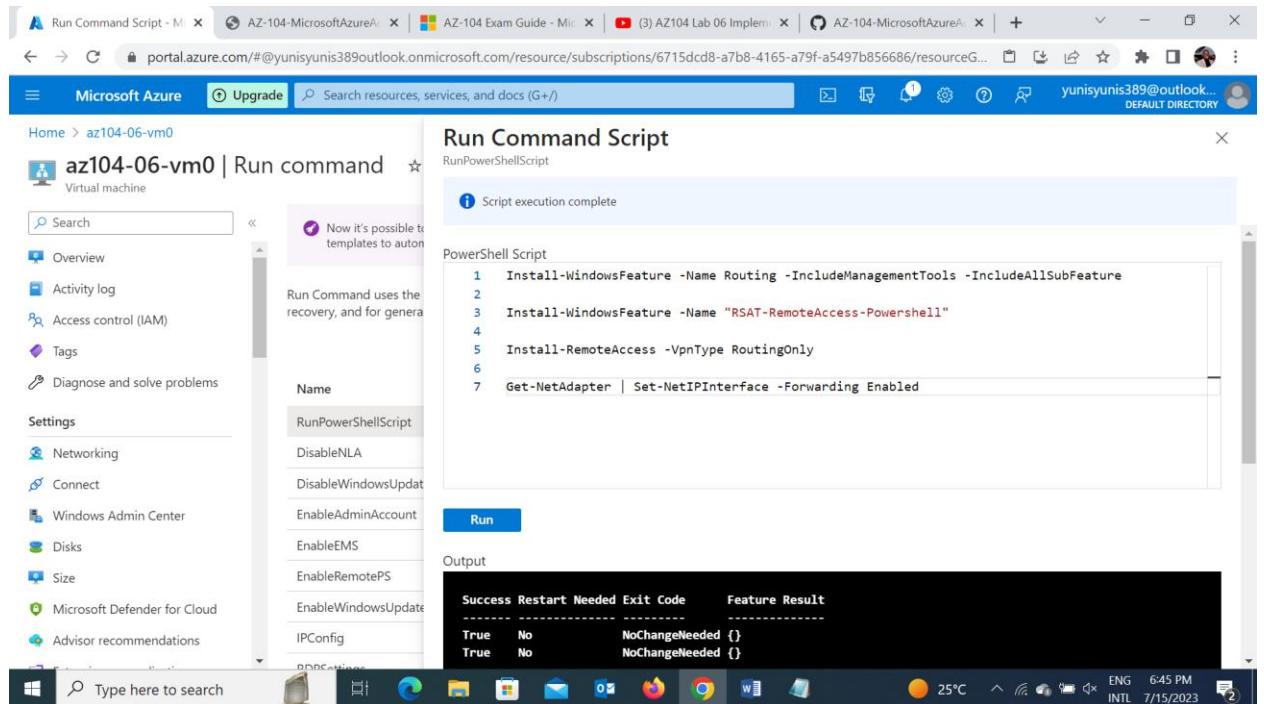
Type here to search

25°C ENG INTL 6:35 PM 7/15/2023

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



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



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

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

Region \*

Name \*

Propagate gateway routes \*  Yes  No

[Previous](#) [Next](#) [Review + create](#)

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

13. Click Go to resource.

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 Add

Home > Microsoft.RouteTable-20230715184617 | Overview > az104-06-rt23

Route table

Search

+ Add Refresh Give feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings Configuration Routes Subnets Properties Locks

Monitoring Alerts

Add route

az104-06-rt23

A user defined route (UDR) is a static route that overrides Azure's default system routes, or adds a route to a subnet's route table. [Learn more](#)

Route name \*

Destination type \*

Destination IP addresses/CIDR ranges \*

Next hop type \*

Next hop address \*

Add

17. Back on the az104-06-rt23 route table blade, in the Settings section, click Subnets, and then click + Associate.

18. Associate the route table az104-06-rt23 with the following subnet:

19. Click Add

The screenshot shows the Microsoft Azure portal interface. The left sidebar has 'Subnets' selected under 'Route table'. The main area is titled 'Associate subnet' with 'az104-06-rt23' selected. A dropdown for 'Virtual network' shows 'az104-06-vnet01 (az104-06-rg1)'. A dropdown for 'Subnet' shows 'subnet0'. At the bottom right is an 'OK' button.

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):

The screenshot shows the Microsoft Azure portal interface. The left sidebar has 'Route tables' selected. The main area is titled 'Create Route table'. The 'Project details' section shows 'Subscription' set to 'free Trial' and 'Resource group' set to 'az104-06-rg1'. The 'Instance details' section shows 'Region' set to 'East US' and 'Name' set to 'az104-06-rt32'. Under 'Propagate gateway routes', the 'No' option is selected. At the bottom are 'Previous', 'Next', and 'Review + create' buttons.

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:

26. Click OK

The screenshot shows the Microsoft Azure portal interface. The left sidebar is open, showing 'az104-06-rt32 | Routes'. The main content area is titled 'Add route' for 'az104-06-rt32'. The 'Route name' field contains 'az104-06-route-vnet3-to-vnet2'. The 'Destination type' is set to 'IP Addresses', and the 'Destination IP addresses/CIDR ranges' field contains '10.62.0.0/20'. The 'Next hop type' is 'Virtual appliance', and the 'Next hop address' is '10.60.0.4'. A large blue 'Add' button is at the bottom right of the form.

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:

29. Click OK

The screenshot shows the Microsoft Azure portal interface. The left sidebar is open, showing 'az104-06-rt32 | Subnets'. The main content area is titled 'Associate subnet' for 'az104-06-rt32'. The 'Virtual network' dropdown is set to 'az104-06-vnet3 (az104-06-rg1)'. The 'Subnet' dropdown is set to 'subnet0'. A large blue 'OK' button is at the bottom right of the dialog.

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

31. On the Network Watcher - Connection troubleshoot blade, use the following settings (leave others with their default values):

Network Watcher | Connection troubleshoot

Source type \* Virtual machine  
az104-06-vm0

Destination type Select a virtual machine  
Specify manually  
URI, FQDN or IP address \* 10.63.0.4

Protocol TCP  
Destination port \* 3389

Diagnostic tests \* 4 selected

32. Click **Run diagnostic tests** and wait until results of the connectivity check are returned.

Verify that the status is **Success**. 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 **Fail**, you should stop and then start az104-06-vm0.

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

Name	Status	IP address	Next hop	RTT	Errors
az104-06-vm0	Success	10.60.0.4	10.63.0.4	3	-
az104-06-nic3	Success	10.63.0.4	-	-	-

## Task 5: Implement Azure Load Balancer

In this task, I 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**:

Project details

Subscription \* free Trial

Resource group \* (New) az104-06-rg4

Create new

Instance details

Name \* az104-06-lb4

Region \* East US

SKU \* Standard

Gateway

Basic

Type \* Public

Internal

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

3. 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

IP type IP address

Public IP address \* Choose public IP address

Gateway Load balancer None

Review + create < Previous Next : Backend pools > Download a template for automation Give feedback

4. On the **Add a public IP address** popup, use the following settings before clicking **OK** and then **Add**. When completed click **Next: Backend pools**.

The screenshot shows the Azure portal interface for creating a load balancer. The main page has tabs for Basics, Frontend IP configuration, Backend pools, Inbound rules, Outbound rules, Tags, and Review + create. The 'Frontend IP configuration' tab is selected. Below it, there's a note about what a frontend IP configuration is and a button to 'Add a frontend IP configuration'. A modal window titled 'Add a public IP address' is open on the right, containing the following fields:

Name *	az104-06-pip4
SKU	<input checked="" type="radio"/> Standard <input type="radio"/> Basic
Tier	<input checked="" type="radio"/> Regional <input type="radio"/> Global
Assignment	<input type="radio"/> Dynamic <input checked="" type="radio"/> Static
Availability zone *	Zone-redundant
Routing	<input checked="" type="radio"/> Microsoft network

Below these settings, there's a 'Create new' button and a 'Gateway Load balancer' dropdown set to 'None'. At the bottom of the modal is an 'Add' button.

5. 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**.

The screenshot shows the 'Add IP configurations to backend pool' dialog. On the left, there's a sidebar with 'IP configurations' and a note that they must be in the same location and virtual network as the load balancer. Below this are buttons for '+ Add' and 'Resource Name'. The main area shows a table with columns: Resource Name, Resource group, Type, IP configuration, IP Address, Availability set, and Tags. Two entries are listed under 'Virtual machine (2)':

Resource Name	Resource group	Type	IP configuration	IP Address	Availability set	Tags
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	-	-

At the bottom of the dialog are 'Save', 'Cancel', and 'Add' buttons, along with a 'Give feedback' link.

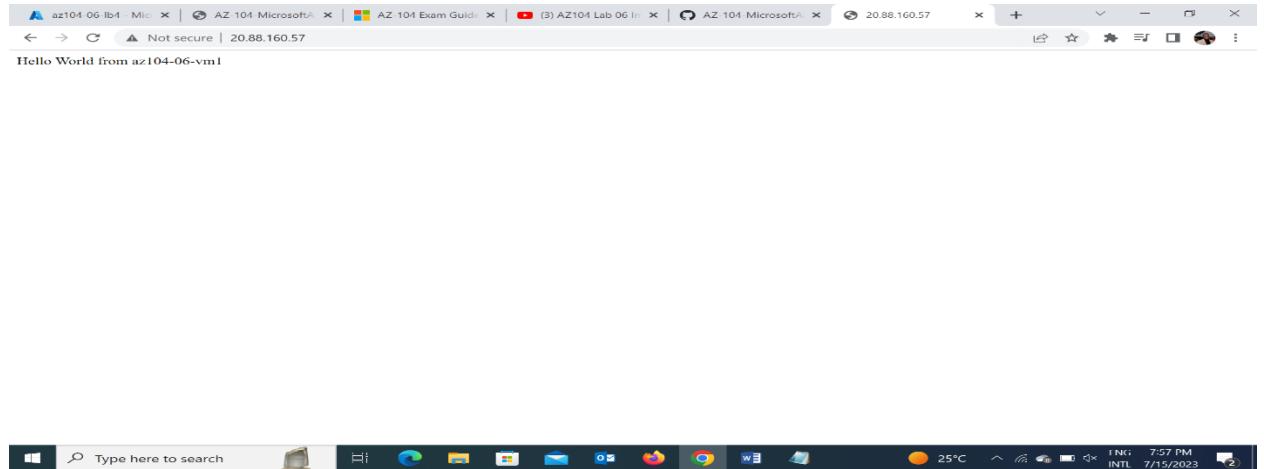
6. On the **Inbound rules** tab, click **Add a load balancing rule**. Add a load balancing rule with the following settings (leave others with their default values). When completed click **Add**.
7. As you have time, review the other tabs, then click Review and create. Ensure there are no validation errors, then click Create.
8. Wait for the load balancer to deploy then click Go to resource.
9. Select Frontend IP configuration from the Load Balancer resource page. Copy the IP address.

The screenshot shows the Microsoft Azure portal interface. The user is viewing the 'Frontend IP configuration' for the load balancer 'az104-06-lb4'. The main pane displays a table with one row, showing the name 'az104-06-fe4' and the IP address '20.88.160.57 (az104-06-pip4)'. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, and Diagnose and solve problems. A settings menu is open, with 'Frontend IP configuration' selected. The bottom taskbar shows various application icons and system status.

10. 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.

The screenshot shows a web browser window with the URL '20.88.160.57'. The page content is 'Hello World from az104-06-vm0'. The browser's address bar also shows 'Not secure | 20.88.160.57'. The bottom taskbar shows various application icons and system status.

11. 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, I 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**

The screenshot shows the Azure portal interface for adding a subnet to the 'az104-06-vnet01' virtual network. The 'Add subnet' dialog is open, displaying the following configuration:

- 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:** (unchecked)
- NAT gateway:** None
- Network security group:** None
- Route table:** None

At the bottom of the dialog, there are 'Save' and 'Cancel' buttons, with 'Save' being the active button.

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**

Subscription \*  Resource group \*  [Create new](#)

**Instance details**

Application gateway name \*  Region \*  Tier  Enable autoscaling  Yes  No  
Instance count  Availability zone  HTTP2  Enabled

[Previous](#) [Next : Frontends >](#)

**Create application gateway**

Region \*  Tier  Enable autoscaling  Yes  No  
Instance count  Availability zone  HTTP2  Enabled

**Configure virtual network**

Virtual network \*  [Create new](#) Subnet \*  [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**.

The screenshot shows the Microsoft Azure portal with the URL [portal.azure.com/#create/Microsoft.ApplicationGateway-ARM](https://portal.azure.com/#create/Microsoft.ApplicationGateway-ARM). The page title is "Create application gateway". The navigation bar includes "Microsoft Azure", "Upgrade", and a search bar. The main content area is titled "Create application gateway" with a sub-section "Frontends". The "Frontend IP address type" is set to "Public". A dropdown menu shows "(New) az104-06-pip5" selected. Navigation buttons at the bottom include "Previous" and "Next : Backends >". The taskbar at the bottom shows various pinned icons.

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

The screenshot shows the Microsoft Azure portal with the URL [portal.azure.com/#view/Microsoft\\_Azure\\_Network/LoadBalancingHubMenuBlade/~/applicationgateways](https://portal.azure.com/#view/Microsoft_Azure_Network/LoadBalancingHubMenuBlade/~/applicationgateways). The page title is "Add a backend pool". The main content area is titled "Add a backend pool". The "Name" field is set to "az104-06-appgw5-be1". The "Target" section lists two items: "IP address or FQDN" with target "10.62.0.4" and "IP address or FQDN" with target "10.63.0.4". Navigation buttons at the bottom include "Previous" and "Next : Configuration >". The taskbar at the bottom shows various pinned icons.

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

**Add a routing rule**

listener and at least one backend target.

Rule name \* az104-06-appgw5-rl1

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.<sup>3</sup>

Listener name \* az104-06-appgw5-rl1

Frontend IP \* Public

Protocol HTTP

Port 80

Additional settings

Listener type Basic

Error page url No

Previous Next : Tags > Add Cancel

11. Switch to the **Backend targets** tab and specify the following settings (leave others with their default values). When completed click **Add** (twice).

**Add Backend setting**

← Discard changes and go back to routing rules

Backend settings name \* az104-06-appgw5-http

Backend protocol HTTP

Backend port 80

Additional settings

Cookie-based affinity Disable

Connection draining

Request time-out (seconds) \* 20

Override backend path

Host name

By default, the Application Gateway sends the same HTTP host header to the backend as it receives from the client. If your backend application/service requires a specific host value, you can override it using this setting.

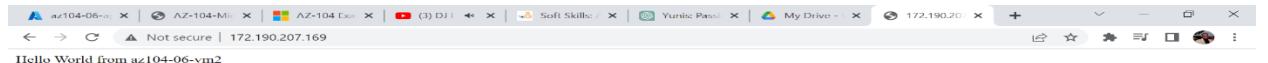
Override with new host name No

Create custom probes

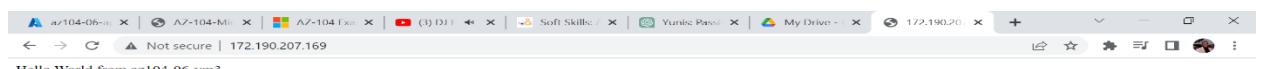
Previous Next : Tags > Add Cancel

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.



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



## Conclusion

In conclusion, Lab 06 - Implement Traffic Management in Azure has been an enlightening and enriching experience, providing me with valuable insights and practical skills in managing traffic within our Azure environment. Through a series of tasks including provisioning the lab environment, configuring network topologies, testing connectivity, optimizing routing, and implementing load balancing and application delivery controllers, I have gained a comprehensive understanding of traffic management techniques. I have learned the importance of distributing traffic effectively, enhancing application availability and performance, and ensuring fault tolerance. The Azure Application Gateway has the ability to intelligently route traffic to backend resources based on various criteria, such as URL path, host header, or HTTP header. This enables administrators to implement sophisticated routing rules and efficiently direct traffic to different applications or services based on specific requirements. The Azure Load Balancer ensures that each resource receives an optimal share of traffic, minimizing the chances of overloading any single resource and optimizing overall performance. This lab has emphasized the significance of proactive monitoring and continuous optimization to deliver seamless user experiences. Overall, this experience has equipped me with the knowledge and tools to tackle the challenges of traffic management in Azure.