

2025

ITHFA0-33 Project

QUESTION 2
EDUV7276134



Creating Private Virtual Switch

Adding VM1 on the switch

Adding VM2 on the switch

VM1 ARP IP Address

Locating Network Adapter

Changing VM1 IP Address to Static

New Static Address for VM1

VM2 ARP IP Address

Changing VM2 IP Address to Static

New Static Address for VM2

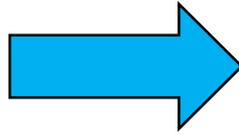
Setting Firewall Rule in VM2

Ping VM2 with VM1

Host Machine IP Address

Pinging Host Machine with VM1

START

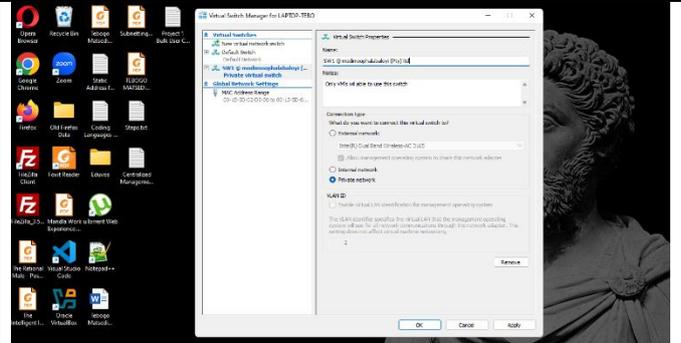


Introduction

In this project, we will create a private network for two virtual machines (VM1 and VM2) using Hyper-V. This will allow them to communicate with each other but not with the physical host computer. We'll also assign static IP addresses, test network connectivity using ping, and adjust firewall settings to ensure smooth communication.

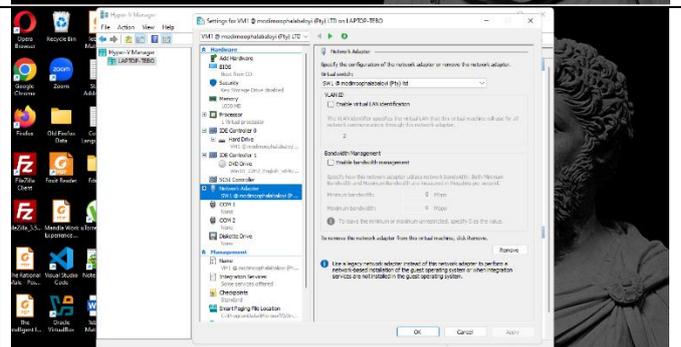
Creating Private Virtual Switch

We begin by creating a private virtual switch in Hyper-V. This type of switch allows virtual machines to talk to each other without connecting to the internet or the physical host. It's like setting up a separate, isolated network for just the VMs.



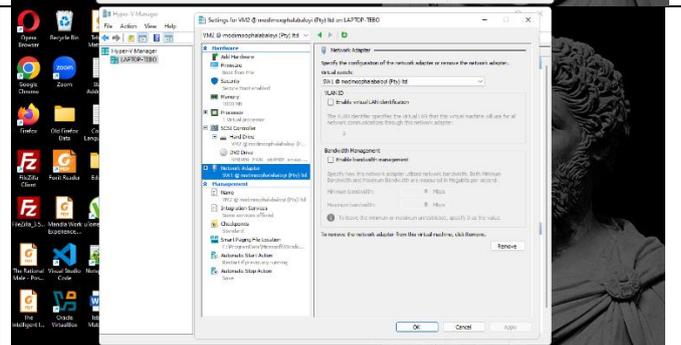
Adding VM1 on the switch

Next, we connect VM1 to the virtual switch. This step links VM1 to the private network we created, so it can later communicate with VM2.



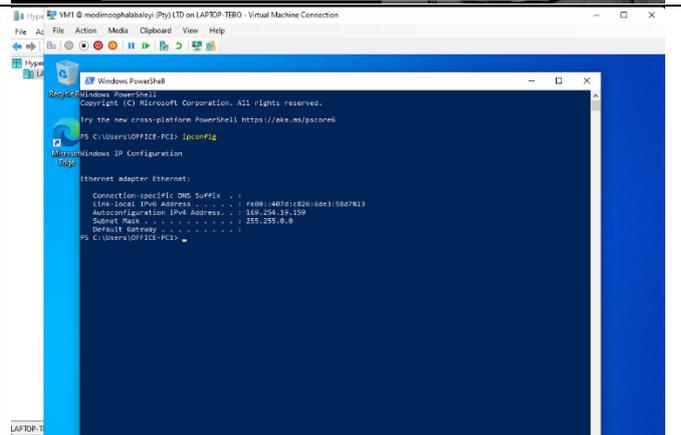
Adding VM2 on the switch

Similarly, we connect VM2 to the same virtual switch. Now both VMs are on the same network and can potentially communicate.

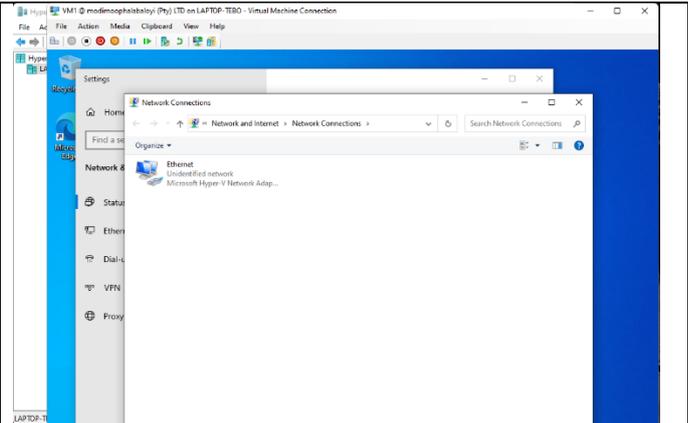


VM1 ARP IP Address

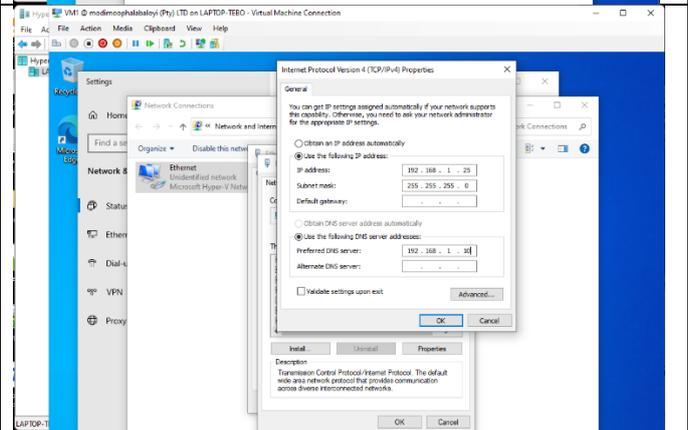
We then check VM1's current IP address details. This helps us understand how it is currently connected and prepares us for assigning it a proper static IP address.



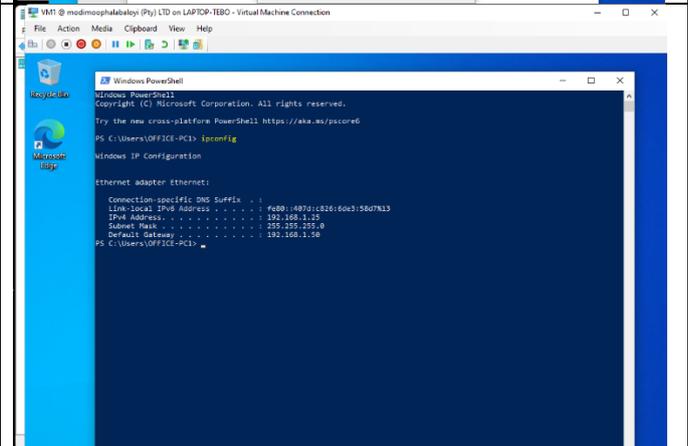
Locating Network Adapter



Changing VM1 IP Address to Static
Here, we set a static IP address for VM1. Unlike an automatic (dynamic) IP, a static IP doesn't change, which makes it easier to test and troubleshoot connections between the two VMs.

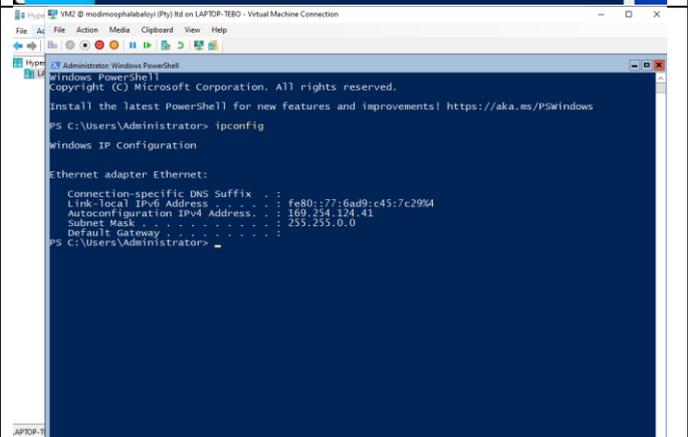


New Static Address for VM1
After changing the settings, we confirm VM1's new static IP address, ensuring it's properly assigned and ready for network communication.



VM2 ARP IP Address

We repeat the process on VM2 by first checking its current IP address details to prepare for setting a static address.



Changing VM2 IP Address to Static
 Next, we run a PowerShell command called New-NetIPAddress to assign a specific IP address and gateway to the VM. This is just another way to set a static IP using commands instead of menus.

```

PS C:\Users\Administrator> Get-NetAdapter

Name      InterfaceDescription      ifIndex Status      MacAddress
-----
Ethernet  Microsoft Hyper-V Network Adapter  4      Up          00-15-50-...

PS C:\Users\Administrator> New-NetIPAddress -InterfaceIndex 4 -IPAddress 192.168.1.10 -PrefixLength 4 -DefaultGateway 192.168.1.50

IPAddress      : 192.168.1.10
InterfaceIndex : 4
InterfaceAlias : Ethernet
AddressFamily  : IPv4
Type           : Unicast
PrefixLength   : 4
SuffixOrigin   : Manual
AddressState   : Tentative
ValidIfetime   : Infinite ([TimeSpan]::MaxValue)
PreferredIfetime : Infinite ([TimeSpan]::MaxValue)
SkipASSource   : False
PolicyStore    : ActiveStore

IPAddress      : 192.168.1.10
InterfaceIndex : 4
InterfaceAlias : Ethernet
AddressFamily  : IPv4
Type           : Unicast
PrefixLength   : 4
SuffixOrigin   : Manual
AddressState   : Invalid
ValidIfetime   : Infinite ([TimeSpan]::MaxValue)
PreferredIfetime : Infinite ([TimeSpan]::MaxValue)
SkipASSource   : False
PolicyStore    : ActiveStore
  
```

New Static Address for VM2
 We verify VM2's static IP is correctly assigned, making sure it matches the same network as VM1 (for example, both using addresses like 192.168.1.x).

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Administrator> ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::77:6ad9:c45:7c29%4
    IPv4 Address. . . . . : 192.168.1.10
    Subnet Mask . . . . . : 240.0.0.0
    Default Gateway . . . . . : 192.168.1.50

PS C:\Users\Administrator>
  
```

Setting Firewall Rule in VM2
 Sometimes, Windows Firewall can block network requests between the VMs. Here, we create a rule to **allow ping requests** so that VM1 and VM2 can communicate freely. We use a PowerShell command called New-NetFirewallRule to add a rule in VM2's firewall. This rule allows VM1 to send network pings (ICMP) to VM2 without being blocked.

```

Windows PowerShell

PS C:\Users\Administrator> New-NetFirewallRule -DisplayName "Allow VM's to communicate to the network" -Protocol ICMPv4 -Direction Inbound -RemoteAddresses 192.168.1.25 -Action Allow

Name      : {b2934fd8-5452-4d69-be4b-75869b6af970}
DisplayName : Allow VM's to communicate to the network
Description : Allow VM's to communicate to the network
DisplayGroup : 
Group      : 
Enabled    : True
Profile    : Any
Platform  : {}
Direction : Inbound
Action     : Allow
EdgeTraversalPolicy : Block
LooseSourceMapping : False
LocationMapping : False
Owner      : 
PrimaryStatus : OK
Status     : The rule was parsed successfully from the store. (65336)
EnforcementStatus : NotApplicable
PolicyStoreSource : PersistentStore
PolicyStoreSourceType : Local
RemoteDynamicKeywordAddresses : {}

PS C:\Users\Administrator>
  
```

Ping VM2 with VM1
 Now we test the connection by using the ping command from VM1 to VM2. If successful, it shows that the private network between the two VMs is working.

```

Windows PowerShell

PS C:\Users\OFFICE-PC1> ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:
Reply from 192.168.1.10: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PS C:\Users\OFFICE-PC1>
  
```


4. Compare and explain the output from the above ping results.

When pinging between both VMs the packet that was sent was able to reach the intended computer 1stly because their on the same network and 2ndly I added firewall rules to allow pinging from either devices whereas trying to ping the host machine it failed 1stly because they are on a separate network because of the private switch I made earlier therefore it would be impossible to reach each other and 2ndly because of firewall rules