TP2 Report

VIRTUAL NETWORKS

By: Nadia FRIKHA and Rania MIDAOUI, RT4-G2

Introduction:

Azure Cloud offers a wide range of resources and services that can be used together to implement different solutions on the Cloud.
One of the most important services offered by Azure is Azure Virtual Network (VNET). It's used whenever you need to run virtual machines (VMs) and applications in the cloud. Once Azure services and virtual machines are built, they communicate securely in the Azure VNet.

We will have a look during the upcoming tasks on how we can build efficient cloud applications by using Azure Virtual Network and some networking components, which provide a wide range of functionalities.

Azure virtual network :

A virtual network (VNet) in Azure is a representation of the network in the cloud. It is a logical isolation of the Azure cloud dedicated to a subscription . Azure virtual network enables Azure resources to securely communicate with each other, the internet, and on-premises networks.

Each VNet you create has its own CIDR block and can be associated with other VNets and on-premises networks as long as the CIDR blocks don't overlap. You also have control over DNS server settings for VNets and the segmentation of the VNet into subnets.

All resources in a VNet can communicate outbound to the internet, by default. You can communicate inbound to a resource by assigning a public IP address or a public Load Balancer. You can also use public IP or public Load Balancer to manage your outbound connections.
Azure resources communicate securely with each other in one of the following ways:

- Through a virtual network
- Through a virtual network service endpoint
- Through VNet Peering

Azure Virtual Networks documentation: https://learn.microsoft.com/en-us/azure/virtual-network/

Questions:

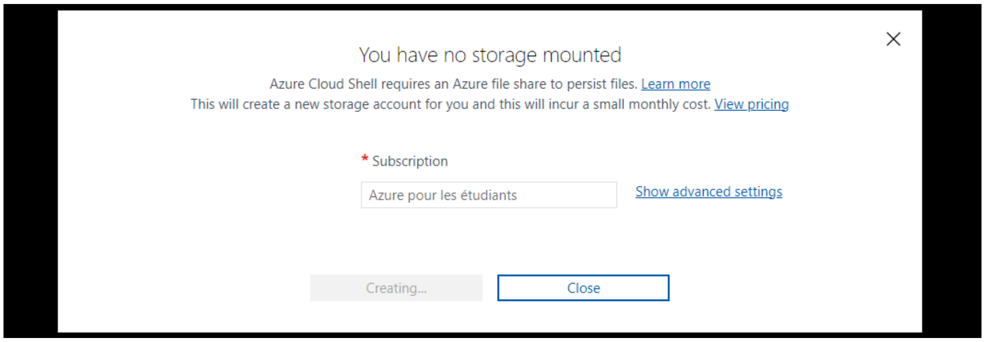
Task 1:

1- We open the Azure Cloud Shell and select PowerShell.

We were given two files to upload, the Parameters and Template files:

http://www.imenemami.com/cloud/tp2/vnetvm-parameters.json

http://www.imenemami.com/cloud/tp2/vnetvm-template.json



You have no storage mounted

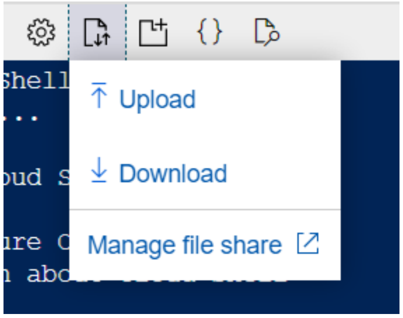
Azure Cloud Shell requires an Azure file share to persist files. Learn more

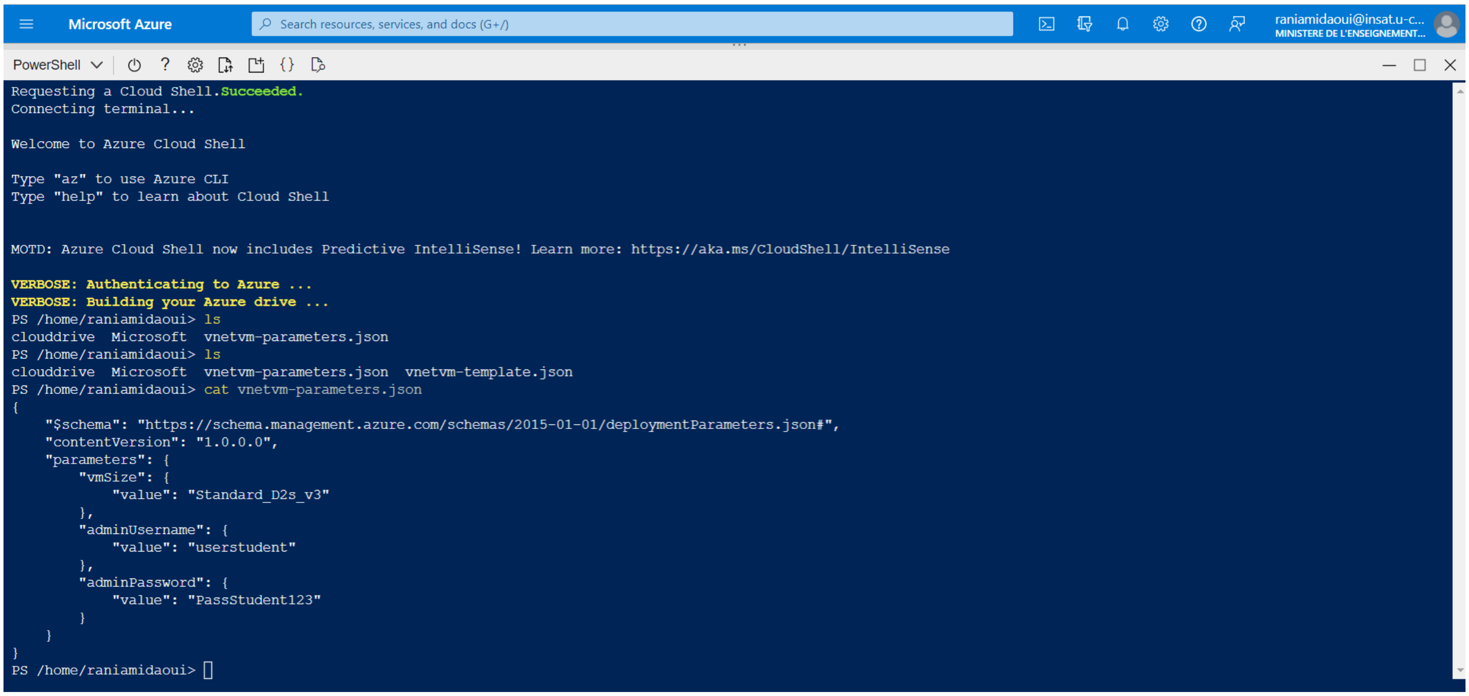
This will create a new storage account for you and this will incur a small monthly cost. View pricing

* Subscription

Azure pour les étudiants

Show advanced settings

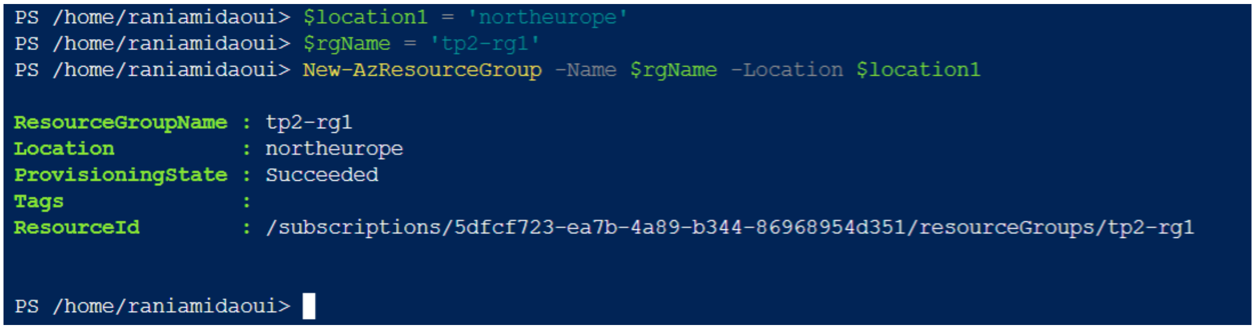




Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
PowerShell

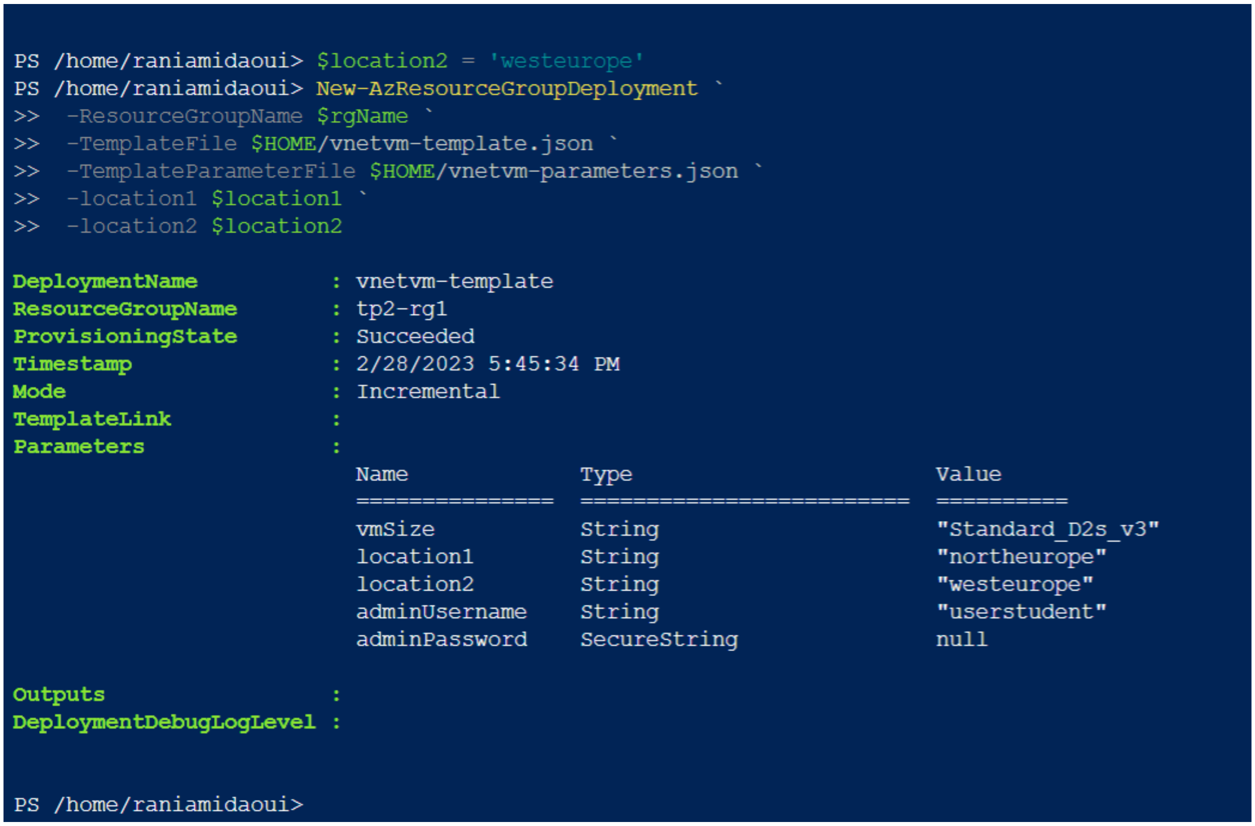
```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell


MOTD: Azure Cloud Shell now includes Predictive IntelliSense! Learn more: https://aka.ms/CloudShell/IntelliSense

VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
PS /home/raniamidaoui> ls
clouddrive  Microsoft  vnetvm-parameters.json
PS /home/raniamidaoui> ls
clouddrive  Microsoft  vnetvm-parameters.json  vnetvm-template.json
PS /home/raniamidaoui> cat vnetvm-parameters.json
{
    "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "vmSize": {
            "value": "Standard_D2s_v3"
        },
        "adminUsername": {
            "value": "userstudent"
        },
        "adminPassword": {
            "value": "PassStudent123"
        }
    }
}
PS /home/raniamidaoui> 
```

2- In the Cloud Shell pane, we run the following commands to create the resource group in the North Europe region, that will be hosting the lab environment:



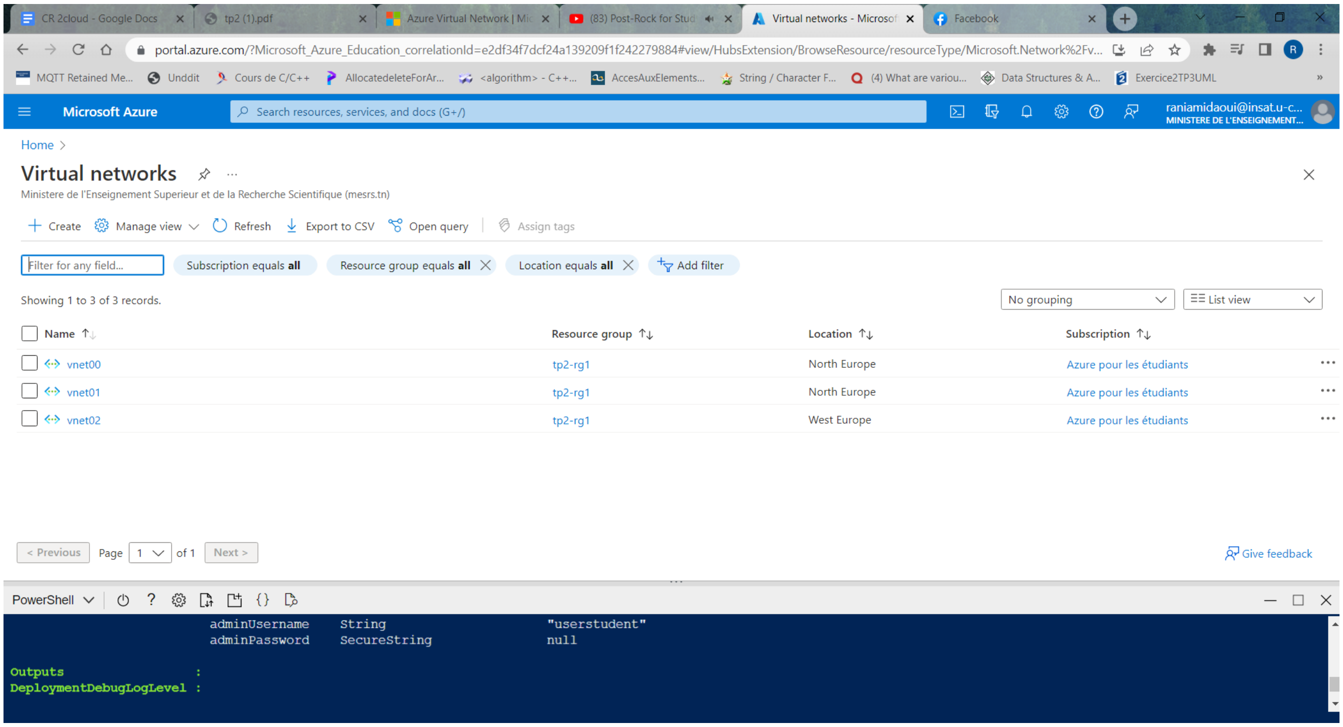
```
PS /home/raniamidaoui> $location1 = 'northeurope'
PS /home/raniamidaoui> $rgName = 'tp2-rg1'
PS /home/raniamidaoui> New-AzResourceGroup -Name $rgName -Location $location1

ResourceGroupName : tp2-rg1
Location          : northeurope
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1


PS /home/raniamidaoui> 
```

3- We run the following to create three virtual machines:



```
PS /home/raniamidaoui> $location2 = 'westeurope'
PS /home/raniamidaoui> New-AzResourceGroupDeployment `
>>  -ResourceGroupName $rgName `
>>  -TemplateFile $HOME/vnetvm-template.json `
>>  -TemplateParameterFile $HOME/vnetvm-parameters.json `
>>  -location1 $location1 `
>>  -location2 $location2

DeploymentName          : vnetvm-template
ResourceGroupName       : tp2-rg1
ProvisioningState       : Succeeded
Timestamp               : 2/28/2023 5:45:34 PM
Mode                    : Incremental
TemplateLink            :
Parameters              :
                          Name             Type                       Value
                          ===============  =========================  ==========
                          vmSize           String                     "Standard_D2s_v3"
                          location1        String                     "northeurope"
                          location2        String                     "westeurope"
                          adminUsername    String                     "userstudent"
                          adminPassword    SecureString               null

Outputs                 :
DeploymentDebugLogLevel :


PS /home/raniamidaoui>
```

4 - We search in the Azure portal for virtual networks you created in the previous task and click vnet00:



Virtual networks

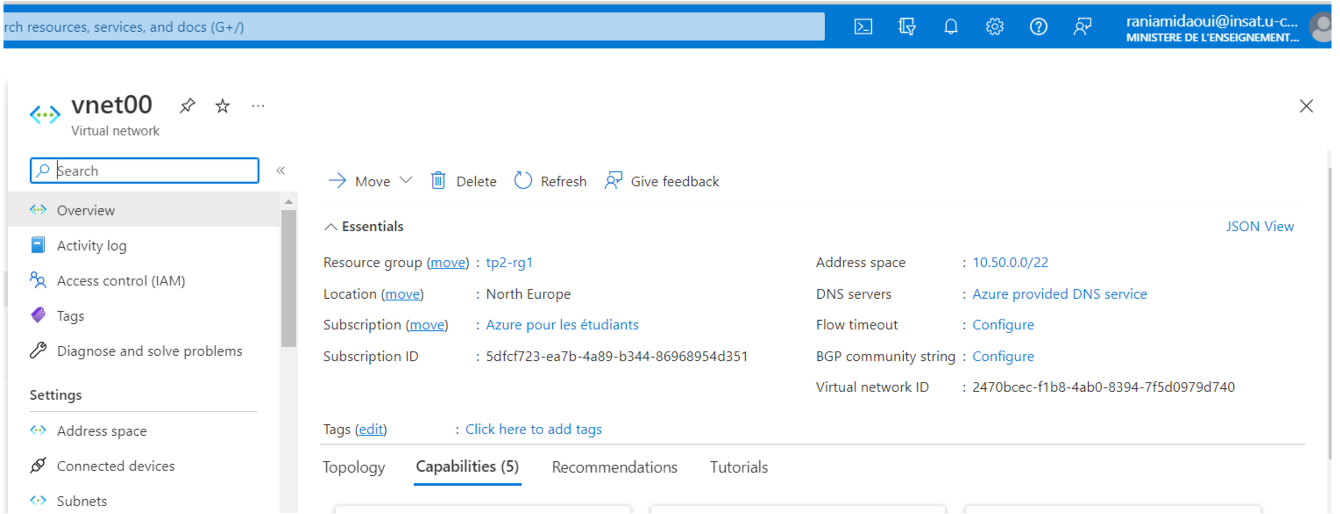
Ministere de l'Enseignement Superieur et de la Recherche Scientifique (mesrs.tn)

Subscription equals **all** | Resource group equals **all** | Location equals **all**

Showing 1 to 3 of 3 records.

Name	Resource group	Location	Subscription
vnet00	tp2-rg1	North Europe	Azure pour les étudiants
vnet01	tp2-rg1	North Europe	Azure pour les étudiants
vnet02	tp2-rg1	West Europe	Azure pour les étudiants

```
                        adminUsername    String                 "userstudent"
                        adminPassword    SecureString           null

Outputs                 :
DeploymentDebugLogLevel :
```



vnet00

Virtual network

Essentials

Resource group (move) : tp2-rg1

Location (move) : North Europe

Subscription (move) : Azure pour les étudiants

Subscription ID : 5dfcf723-ea7b-4a89-b344-86968954d351

Address space : 10.50.0.0/22

DNS servers : Azure provided DNS service

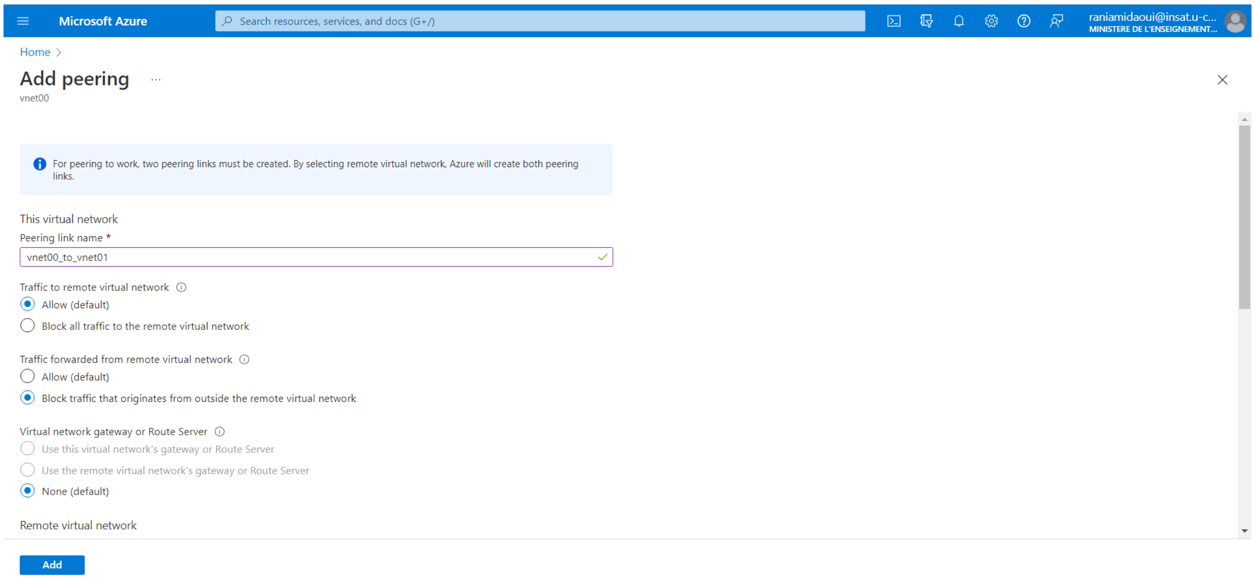
Flow timeout : Configure

BGP community string : Configure

Virtual network ID : 2470bcec-f1b8-4ab0-8394-7f5d0979d740

Tags (edit) : Click here to add tags

5- We add a peering on the vnet00 virtual network blade:


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home

Add peering ···

vnet00

For peering to work, two peering links must be created. By selecting remote virtual network, Azure will create both peering links.

This virtual network

Peering link name * vnet00_to_vnet01

Traffic to remote virtual network

● Allow (default)
○ Block all traffic to the remote virtual network

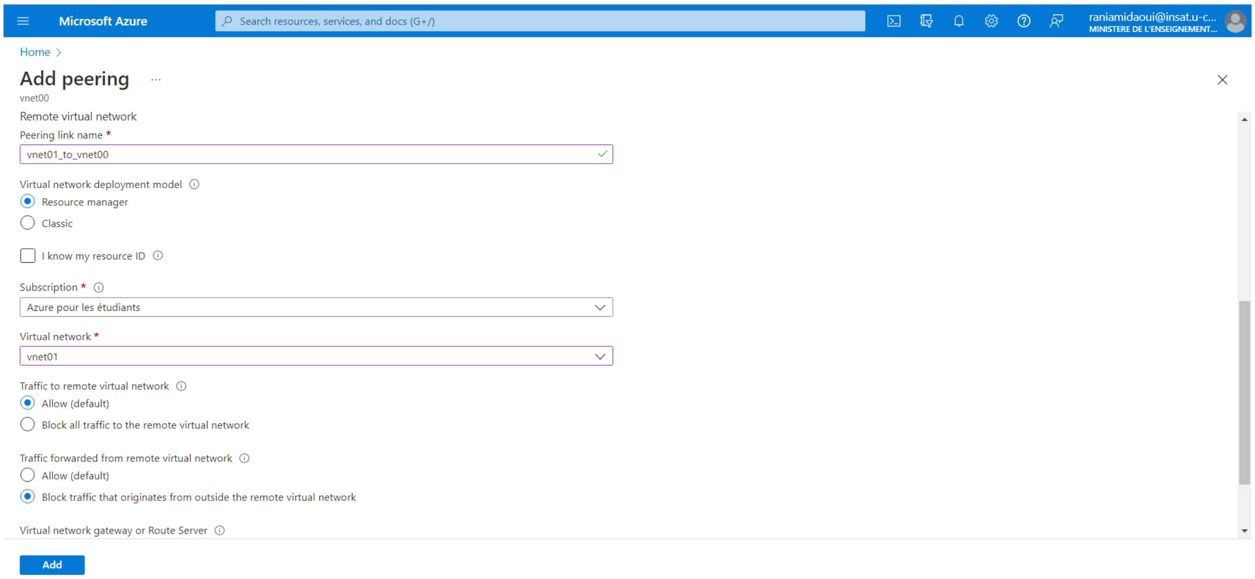
Traffic forwarded from remote virtual network

○ 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


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home

Add peering ···

vnet00

Remote virtual network

Peering link name * vnet01_to_vnet00

Virtual network deployment model

● Resource manager
○ Classic

□ I know my resource ID

Subscription * Azure pour les étudiants

Virtual network * vnet01

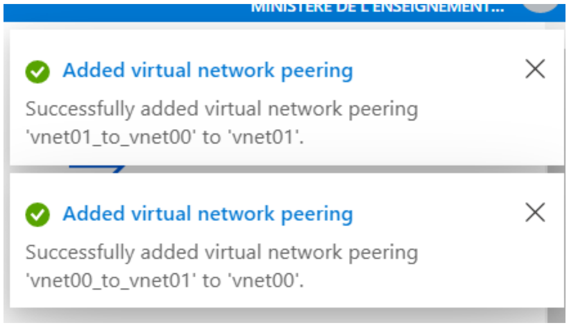
Traffic to remote virtual network

● 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



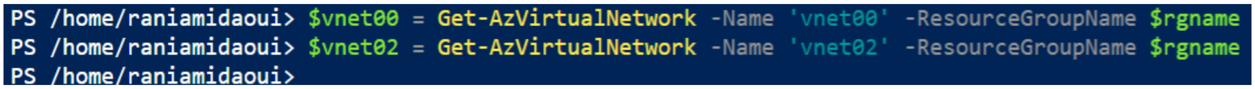
✓ Added virtual network peering

Successfully added virtual network peering 'vnet01_to_vnet00' to 'vnet01'.

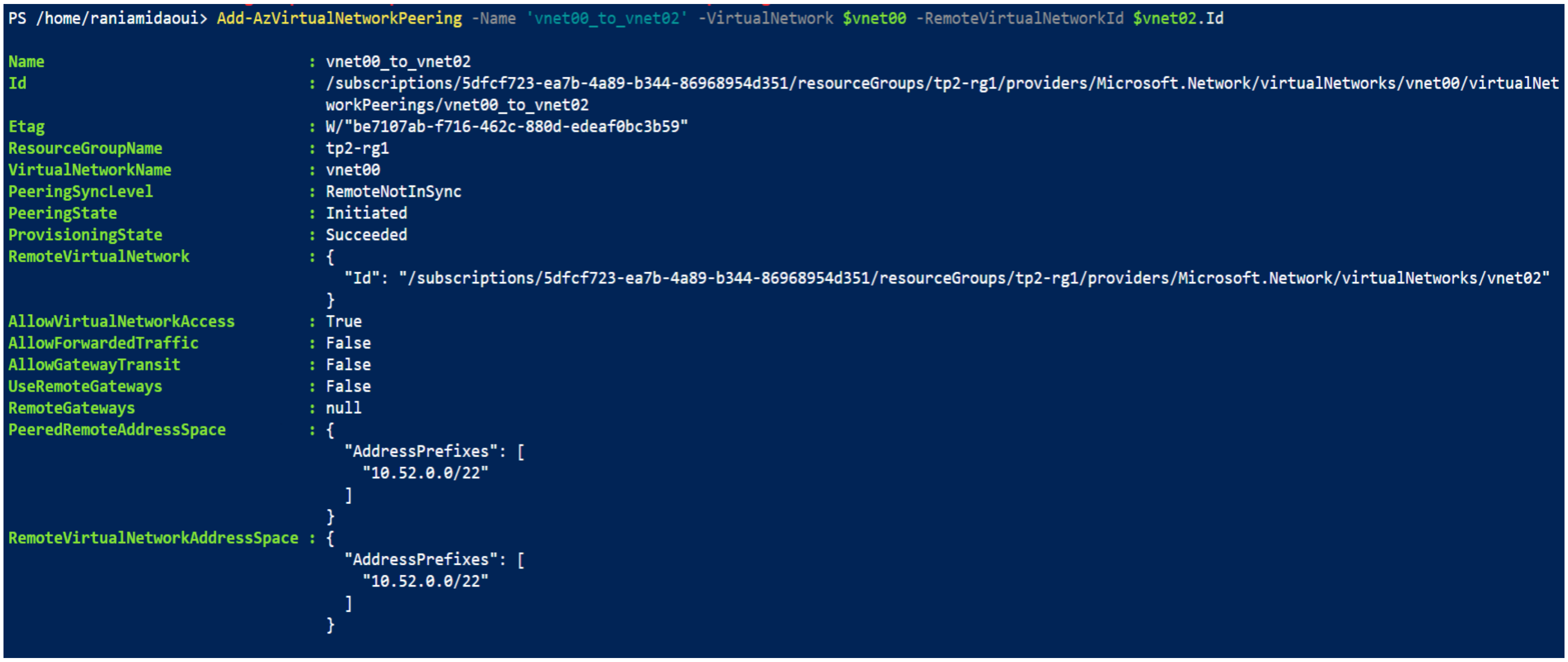
✓ Added virtual network peering

Successfully added virtual network peering 'vnet00_to_vnet01' to 'vnet00'.

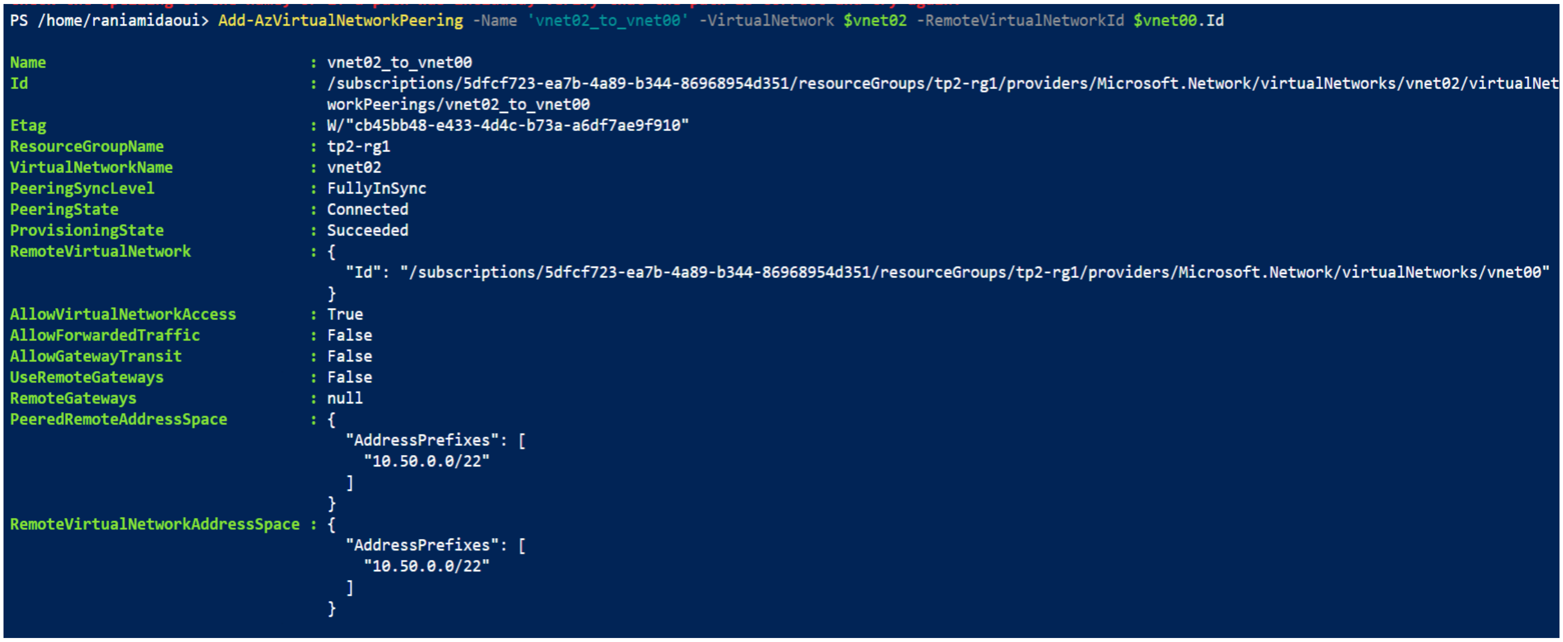
6- We configure peering vnet00_to_vnet02:



```
PS /home/raniamidaoui> $vnet00 = Get-AzVirtualNetwork -Name 'vnet00' -ResourceGroupName $rgname
PS /home/raniamidaoui> $vnet02 = Get-AzVirtualNetwork -Name 'vnet02' -ResourceGroupName $rgname
PS /home/raniamidaoui>
```

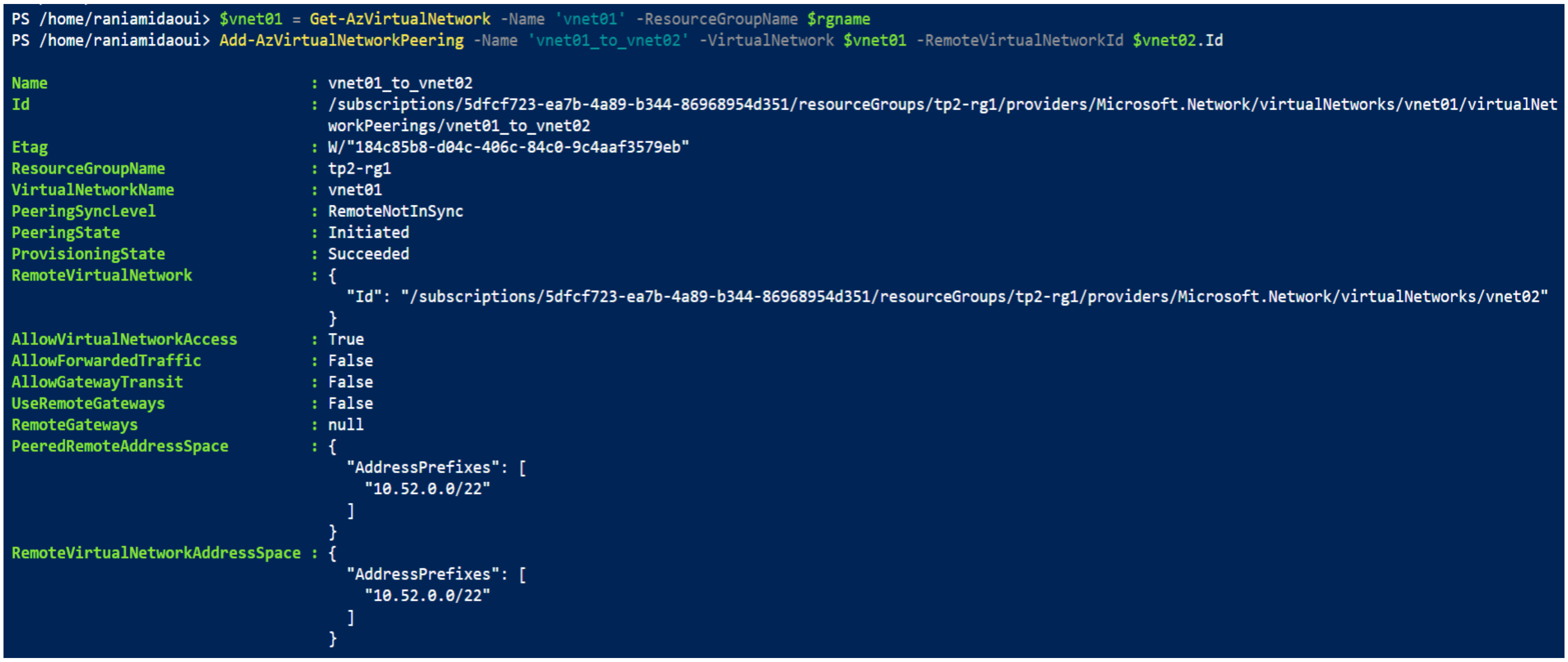


```
PS /home/raniamidaoui> Add-AzVirtualNetworkPeering -Name 'vnet00_to_vnet02' -VirtualNetwork $vnet00 -RemoteVirtualNetworkId $vnet02.Id

Name                             : vnet00_to_vnet02
Id                               : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet00/virtualNet
                                   workPeerings/vnet00_to_vnet02
Etag                             : W/"be7107ab-f716-462c-880d-edeaf0bc3b59"
ResourceGroupName                : tp2-rg1
VirtualNetworkName               : vnet00
PeeringSyncLevel                 : RemoteNotInSync
PeeringState                     : Initiated
ProvisioningState                : Succeeded
RemoteVirtualNetwork             : {
                                     "Id": "/subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet02"
                                   }
AllowVirtualNetworkAccess        : True
AllowForwardedTraffic            : False
AllowGatewayTransit              : False
UseRemoteGateways                : False
RemoteGateways                   : null
PeeredRemoteAddressSpace         : {
                                     "AddressPrefixes": [
                                       "10.52.0.0/22"
                                     ]
                                   }
RemoteVirtualNetworkAddressSpace : {
                                     "AddressPrefixes": [
                                       "10.52.0.0/22"
                                     ]
                                   }
```

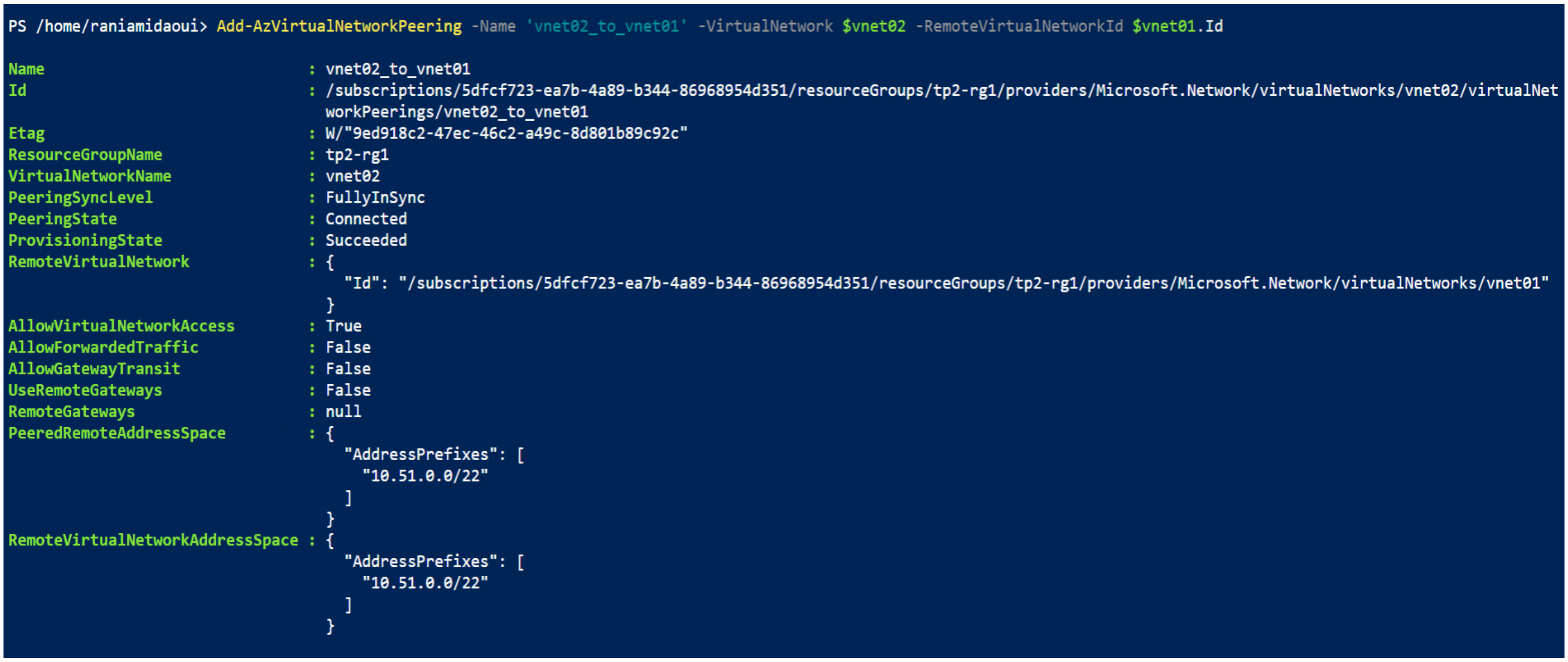


```
PS /home/raniamidaoui> Add-AzVirtualNetworkPeering -Name 'vnet02_to_vnet00' -VirtualNetwork $vnet02 -RemoteVirtualNetworkId $vnet00.Id

Name                             : vnet02_to_vnet00
Id                               : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet02/virtualNet
                                   workPeerings/vnet02_to_vnet00
Etag                             : W/"cb45bb48-e433-4d4c-b73a-a6df7ae9f910"
ResourceGroupName                : tp2-rg1
VirtualNetworkName               : vnet02
PeeringSyncLevel                 : FullyInSync
PeeringState                     : Connected
ProvisioningState                : Succeeded
RemoteVirtualNetwork             : {
                                     "Id": "/subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet00"
                                   }
AllowVirtualNetworkAccess        : True
AllowForwardedTraffic            : False
AllowGatewayTransit              : False
UseRemoteGateways                : False
RemoteGateways                   : null
PeeredRemoteAddressSpace         : {
                                     "AddressPrefixes": [
                                       "10.50.0.0/22"
                                     ]
                                   }
RemoteVirtualNetworkAddressSpace : {
                                     "AddressPrefixes": [
                                       "10.50.0.0/22"
                                     ]
                                   }
```

7- Configure peering vnet01_to_vnet02:

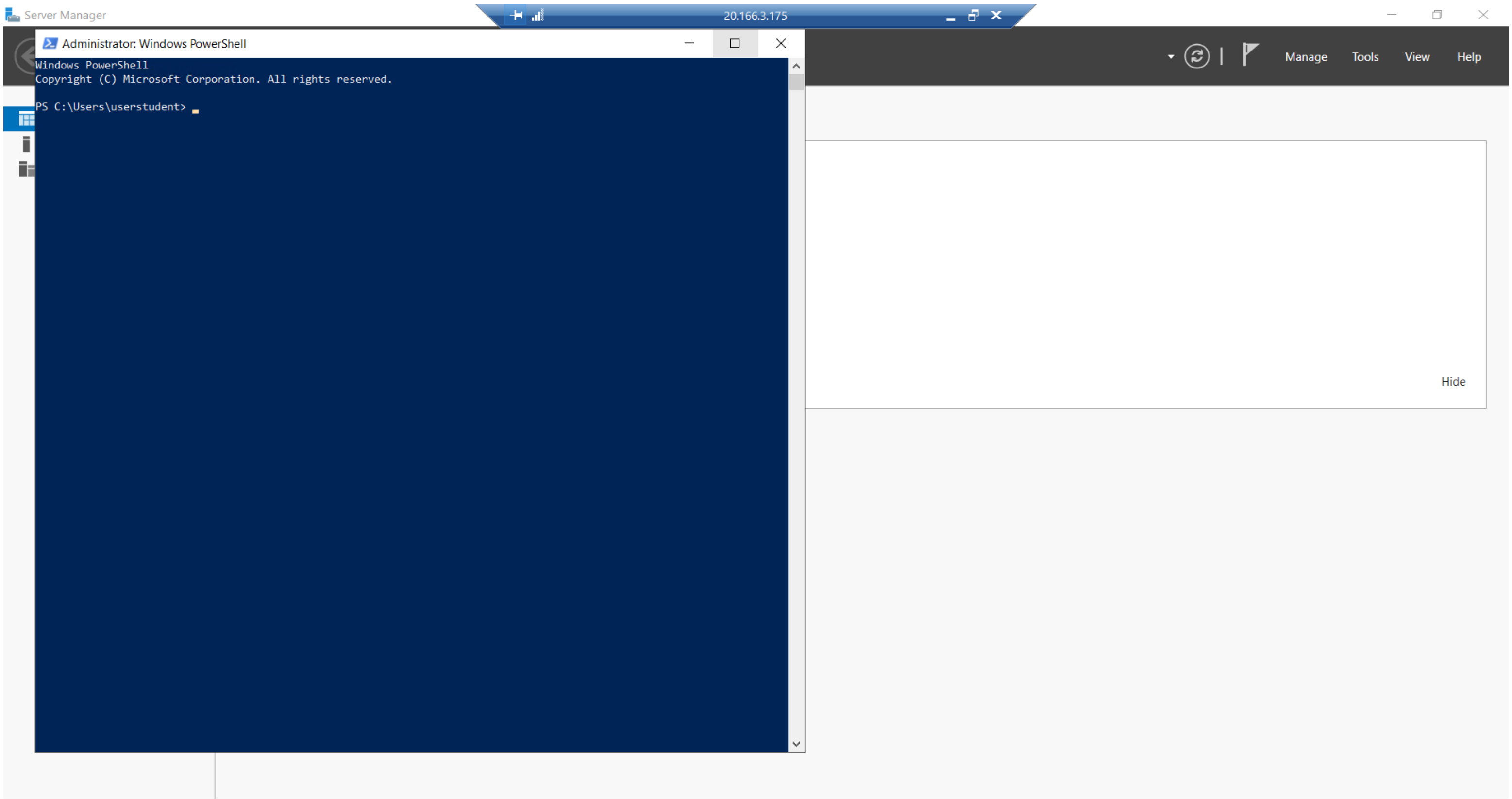


```
PS /home/raniamidaoui> $vnet01 = Get-AzVirtualNetwork -Name 'vnet01' -ResourceGroupName $rgname
PS /home/raniamidaoui> Add-AzVirtualNetworkPeering -Name 'vnet01_to_vnet02' -VirtualNetwork $vnet01 -RemoteVirtualNetworkId $vnet02.Id

Name                             : vnet01_to_vnet02
Id                               : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet01/virtualNet
                                   workPeerings/vnet01_to_vnet02
Etag                             : W/"184c85b8-d04c-406c-84c0-9c4aaf3579eb"
ResourceGroupName                : tp2-rg1
VirtualNetworkName               : vnet01
PeeringSyncLevel                 : RemoteNotInSync
PeeringState                     : Initiated
ProvisioningState                : Succeeded
RemoteVirtualNetwork             : {
                                     "Id": "/subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet02"
                                   }
AllowVirtualNetworkAccess        : True
AllowForwardedTraffic            : False
AllowGatewayTransit              : False
UseRemoteGateways                : False
RemoteGateways                   : null
PeeredRemoteAddressSpace         : {
                                     "AddressPrefixes": [
                                       "10.52.0.0/22"
                                     ]
                                   }
RemoteVirtualNetworkAddressSpace : {
                                     "AddressPrefixes": [
                                       "10.52.0.0/22"
                                     ]
                                   }
```



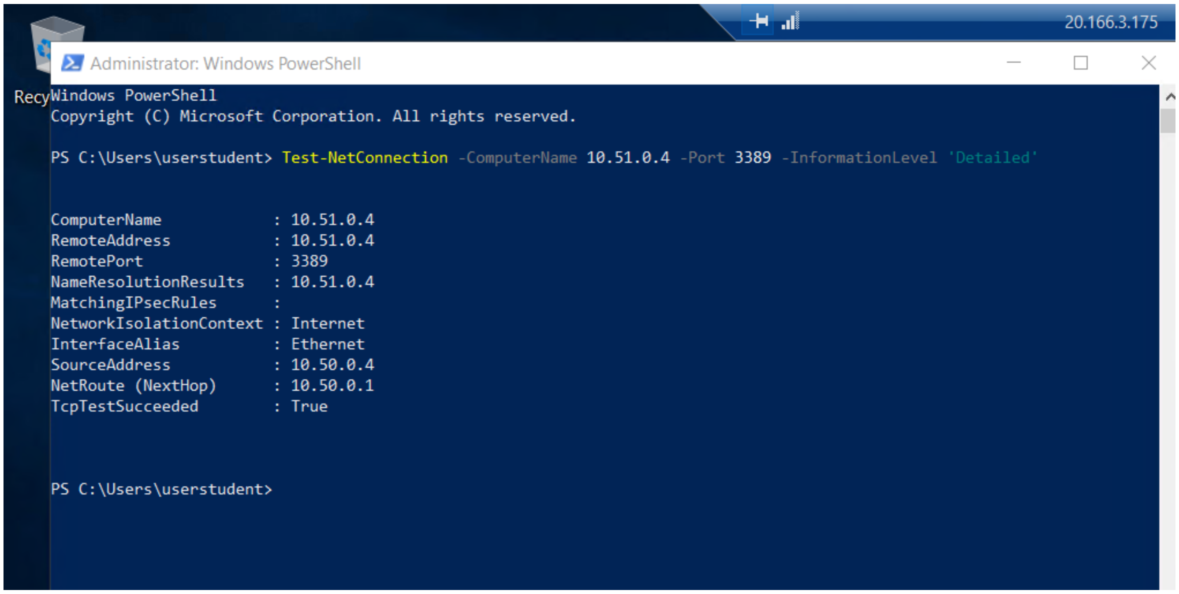
```
PS /home/raniamidaoui> Add-AzVirtualNetworkPeering -Name 'vnet02_to_vnet01' -VirtualNetwork $vnet02 -RemoteVirtualNetworkId $vnet01.Id

Name                             : vnet02_to_vnet01
Id                               : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet02/virtualNet
                                   workPeerings/vnet02_to_vnet01
Etag                             : W/"9ed918c2-47ec-46c2-a49c-8d801b89c92c"
ResourceGroupName                : tp2-rg1
VirtualNetworkName               : vnet02
PeeringSyncLevel                 : FullyInSync
PeeringState                     : Connected
ProvisioningState                : Succeeded
RemoteVirtualNetwork             : {
                                     "Id": "/subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg1/providers/Microsoft.Network/virtualNetworks/vnet01"
                                   }
AllowVirtualNetworkAccess        : True
AllowForwardedTraffic            : False
AllowGatewayTransit              : False
UseRemoteGateways                : False
RemoteGateways                   : null
PeeredRemoteAddressSpace         : {
                                     "AddressPrefixes": [
                                       "10.51.0.0/22"
                                     ]
                                   }
RemoteVirtualNetworkAddressSpace : {
                                     "AddressPrefixes": [
                                       "10.51.0.0/22"
                                     ]
                                   }
```

8- We connect to vm00 using RDP and launch PowerShell as admin:



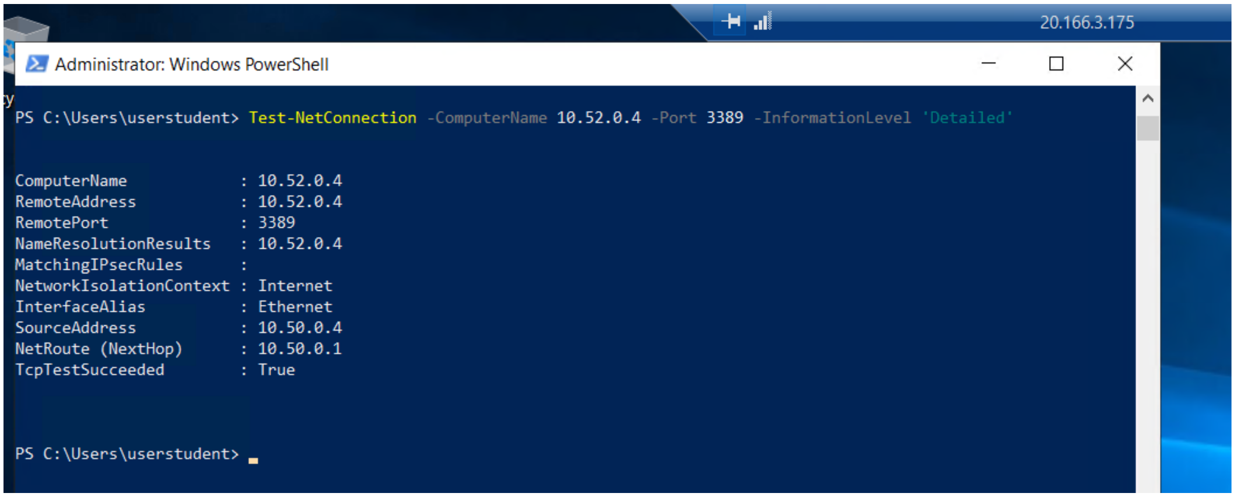
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\userstudent> _
```

9- We test the connectivity to vm01:



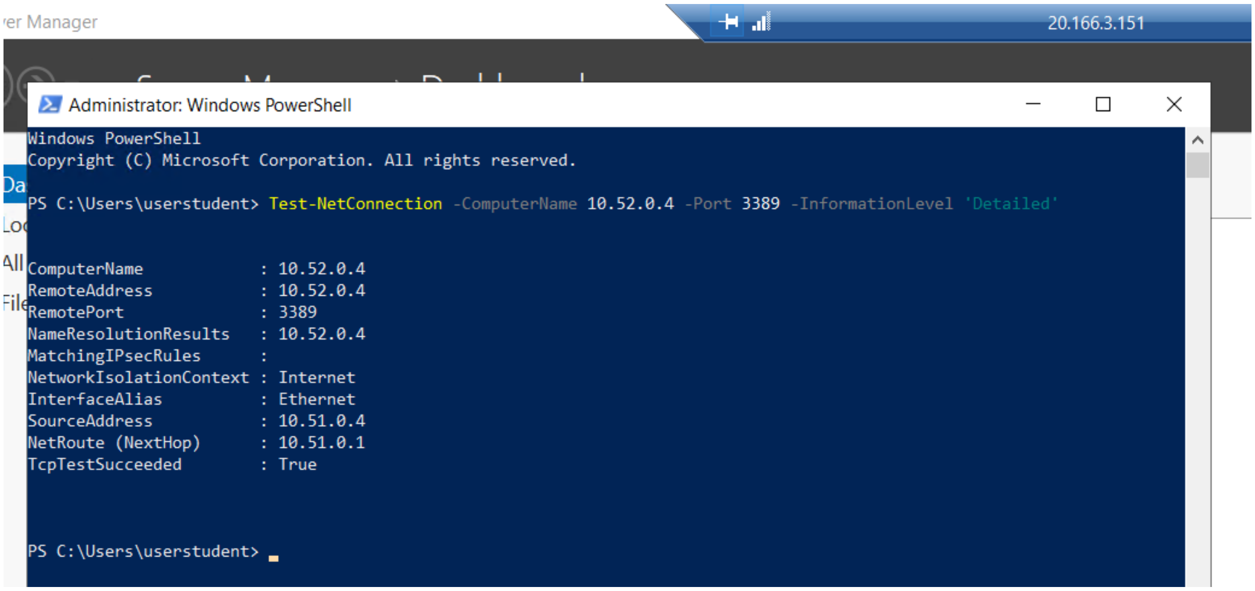
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\userstudent> Test-NetConnection -ComputerName 10.51.0.4 -Port 3389 -InformationLevel 'Detailed'


ComputerName            : 10.51.0.4
RemoteAddress           : 10.51.0.4
RemotePort              : 3389
NameResolutionResults   : 10.51.0.4
MatchingIPsecRules      :
NetworkIsolationContext : Internet
InterfaceAlias          : Ethernet
SourceAddress           : 10.50.0.4
NetRoute (NextHop)      : 10.50.0.1
TcpTestSucceeded        : True



PS C:\Users\userstudent>
```

10- We test the connectivity to vm02:



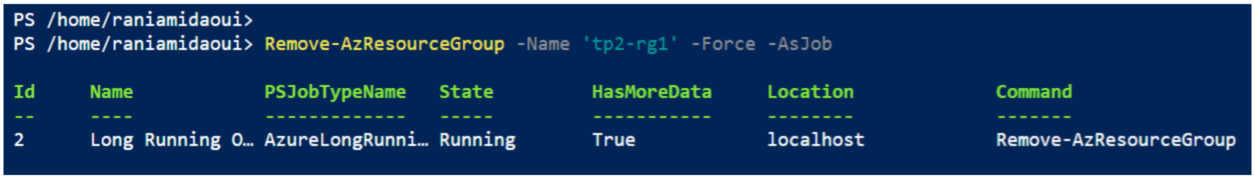
```
PS C:\Users\userstudent> Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'


ComputerName            : 10.52.0.4
RemoteAddress           : 10.52.0.4
RemotePort              : 3389
NameResolutionResults   : 10.52.0.4
MatchingIPsecRules      :
NetworkIsolationContext : Internet
InterfaceAlias          : Ethernet
SourceAddress           : 10.50.0.4
NetRoute (NextHop)      : 10.50.0.1
TcpTestSucceeded        : True



PS C:\Users\userstudent>
```

11- We connect to vm01 using RDP and launch PowerShell as admin, then test the connectivity to vm02:



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

PS C:\Users\userstudent> Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'


ComputerName            : 10.52.0.4
RemoteAddress           : 10.52.0.4
RemotePort              : 3389
NameResolutionResults   : 10.52.0.4
MatchingIPsecRules      :
NetworkIsolationContext : Internet
InterfaceAlias          : Ethernet
SourceAddress           : 10.51.0.4
NetRoute (NextHop)      : 10.51.0.1
TcpTestSucceeded        : True



PS C:\Users\userstudent>
```

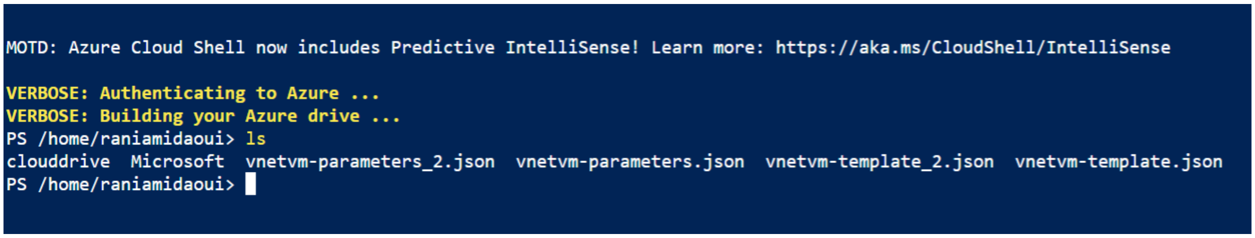
12- We delete the resource group:



```
PS /home/raniamidaoui>
PS /home/raniamidaoui> Remove-AzResourceGroup -Name 'tp2-rg1' -Force -AsJob

Id     Name             PSJobTypeName   State         HasMoreData     Location             Command
--     ----             -------------   -----         -----------     --------             -------
2      Long Running O…  AzureLongRunni… Running       True            localhost            Remove-AzResourceGroup
```

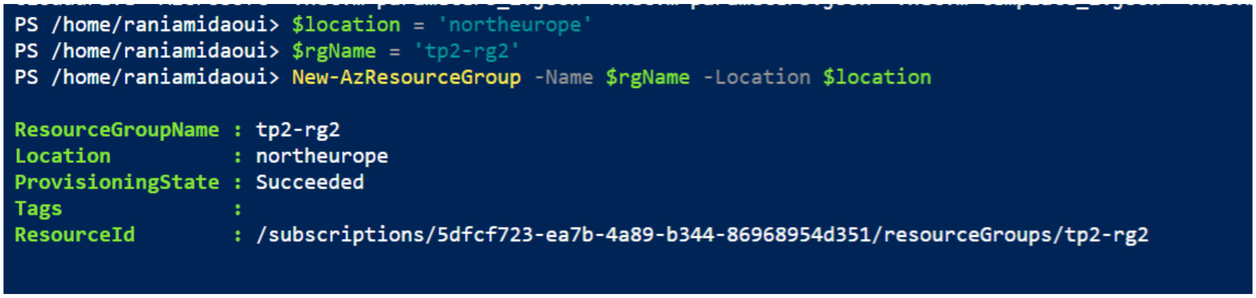
Task 2:

1- We open the Azure Cloud Shell and select PowerShell.
We were given two files to upload, the Parameters and Template files:
http://www.imenemami.com/cloud/tp2/vnetvm-parameters_2.json
http://www.imenemami.com/cloud/tp2/vnetvm-template_2.json



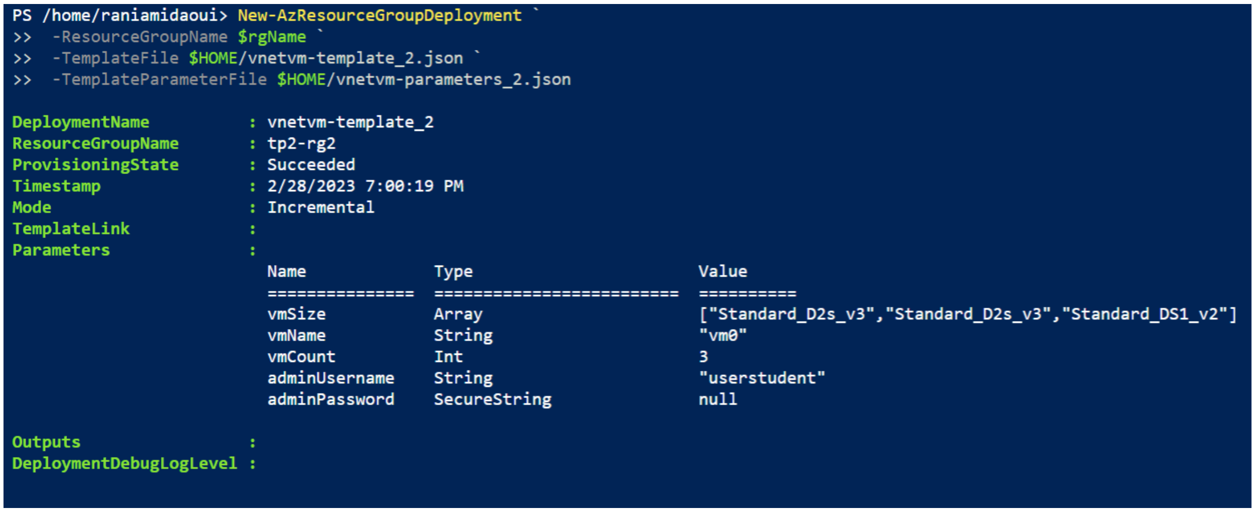
```
MOTD: Azure Cloud Shell now includes Predictive IntelliSense! Learn more: https://aka.ms/CloudShell/IntelliSense

VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
PS /home/raniamidaoui> ls
clouddrive  Microsoft  vnetvm-parameters_2.json  vnetvm-parameters.json  vnetvm-template_2.json  vnetvm-template.json
PS /home/raniamidaoui>
```

2- In the Cloud Shell pane, we run the following commands to create the resource group in the North Europe region, that will be hosting the lab environment:



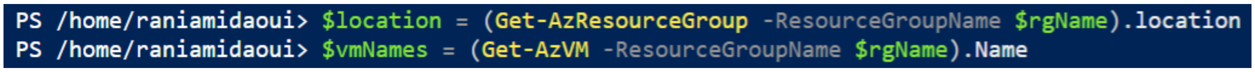
```
PS /home/raniamidaoui> $location = 'northeurope'
PS /home/raniamidaoui> $rgName = 'tp2-rg2'
PS /home/raniamidaoui> New-AzResourceGroup -Name $rgName -Location $location

ResourceGroupName : tp2-rg2
Location          : northeurope
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg2
```

3- We run the following to create three virtual networks and three virtual machines:

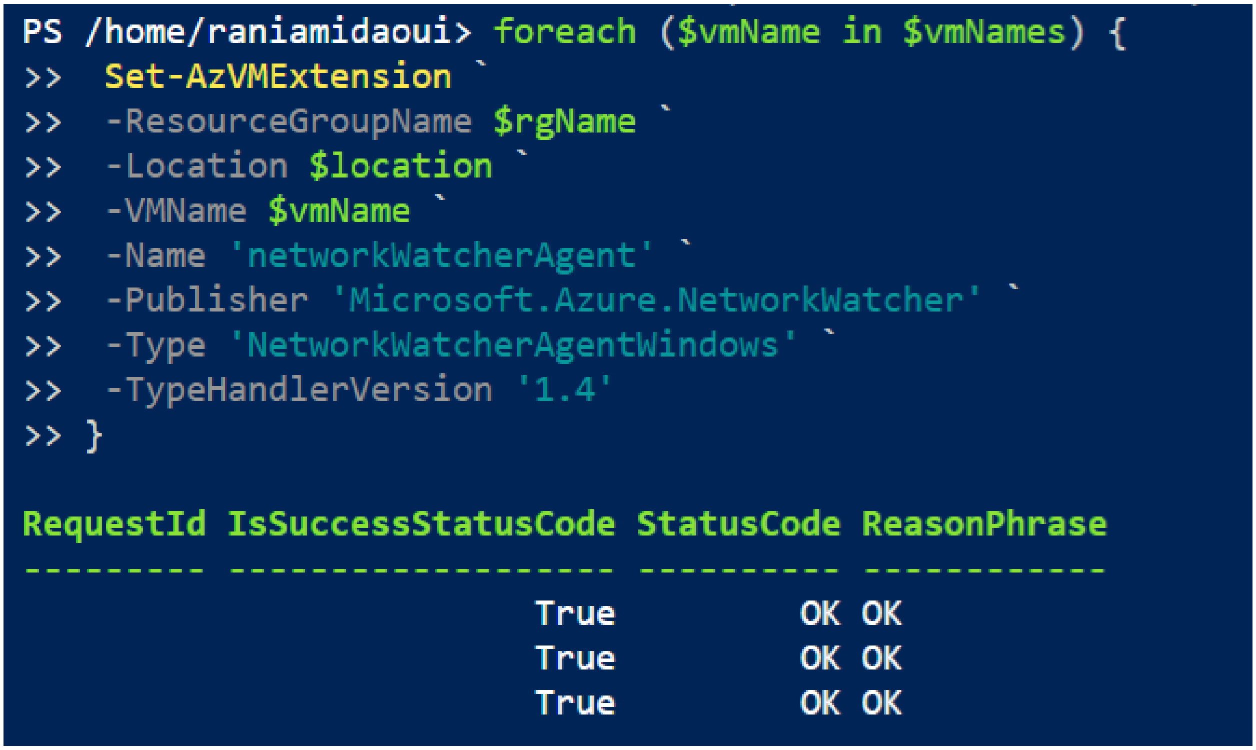


```
PS /home/raniamidaoui> New-AzResourceGroupDeployment `
>>  -ResourceGroupName $rgName `
>>  -TemplateFile $HOME/vnetvm-template_2.json `
>>  -TemplateParameterFile $HOME/vnetvm-parameters_2.json

DeploymentName          : vnetvm-template_2
ResourceGroupName       : tp2-rg2
ProvisioningState       : Succeeded
Timestamp               : 2/28/2023 7:00:19 PM
Mode                    : Incremental
TemplateLink            :
Parameters              :
                          Name             Type                       Value
                          ===============  =========================  ==========
                          vmSize           Array                      ["Standard_D2s_v3","Standard_D2s_v3","Standard_DS1_v2"]
                          vmName           String                     "vm0"
                          vmCount          Int                        3
                          adminUsername    String                     "userstudent"
                          adminPassword    SecureString               null

Outputs                 :
DeploymentDebugLogLevel :
```

4 - We install the Network Watcher extension on the virtual machines deployed in 3-:

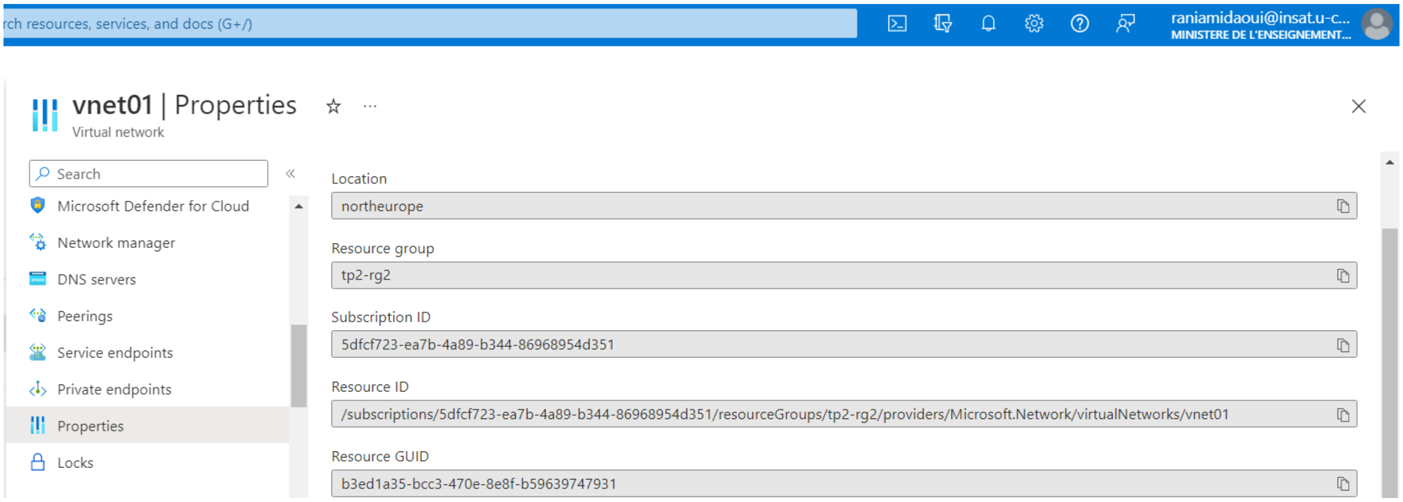


```
PS /home/raniamidaoui> $location = (Get-AzResourceGroup -ResourceGroupName $rgName).location
PS /home/raniamidaoui> $vmNames = (Get-AzVM -ResourceGroupName $rgName).Name
```



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

RequestId IsSuccessStatusCode StatusCode ReasonPhrase
--------- ------------------- ---------- ------------
                         True         OK OK
                         True         OK OK
                         True         OK OK
```

5- On the vnet01 blade, we record the value of the Resource ID property



Location: northeurope

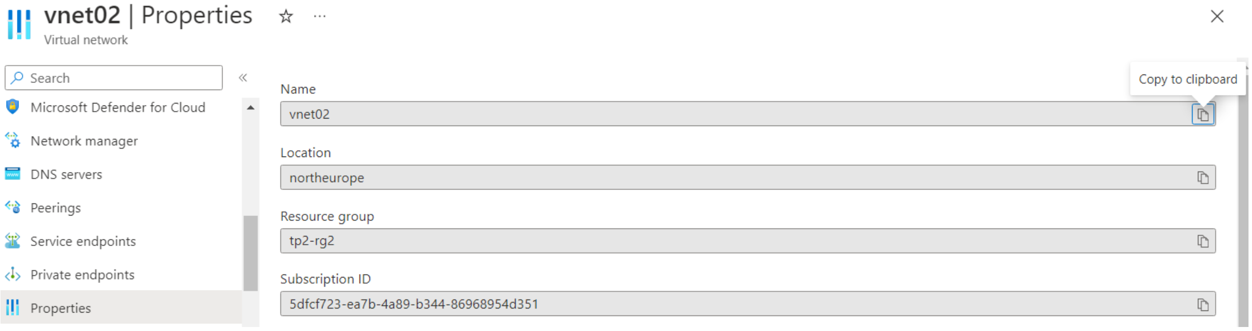
Resource group: tp2-rg2

Subscription ID: 5dfcf723-ea7b-4a89-b344-86968954d351

Resource ID: /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg2/providers/Microsoft.Network/virtualNetworks/vnet01

Resource GUID: b3ed1a35-bcc3-470e-8e8f-b59639747931

6- On the vnet02 blade, we record the value of the Resource ID property



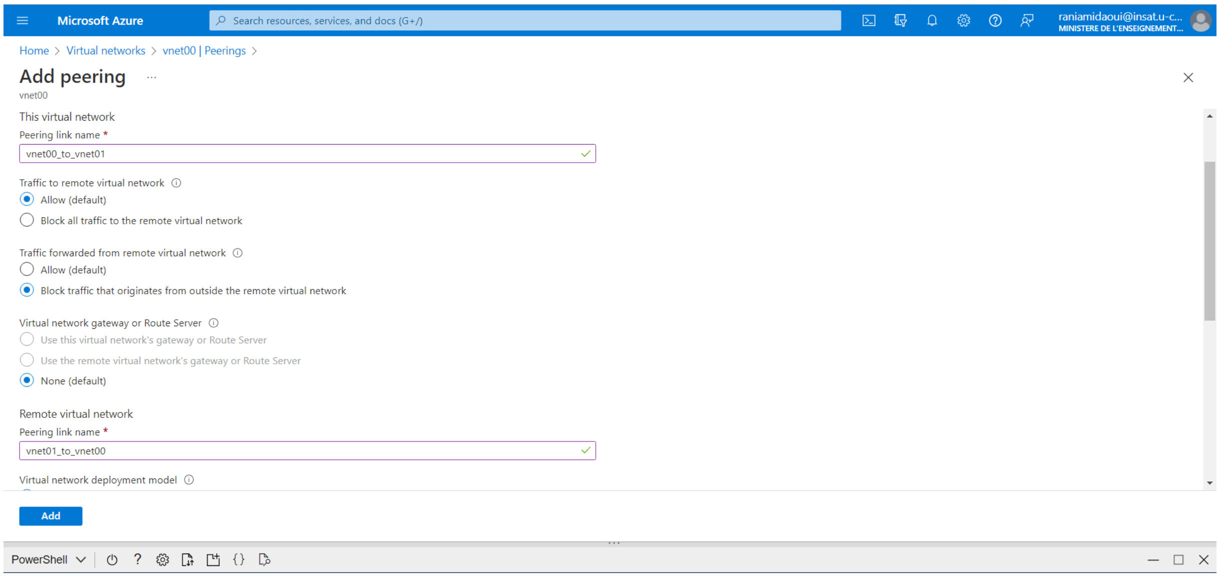
Name: vnet02

Location: northeurope

Resource group: tp2-rg2

Subscription ID: 5dfcf723-ea7b-4a89-b344-86968954d351

7- On the vnet00 blade, we add a peering:


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home > Virtual networks > vnet00 | Peerings >
Add peering
vnet00
Add
PowerShell

This virtual network

Peering link name *: vnet00_to_vnet01

Traffic to remote virtual network
● Allow (default)
○ Block all traffic to the remote virtual network

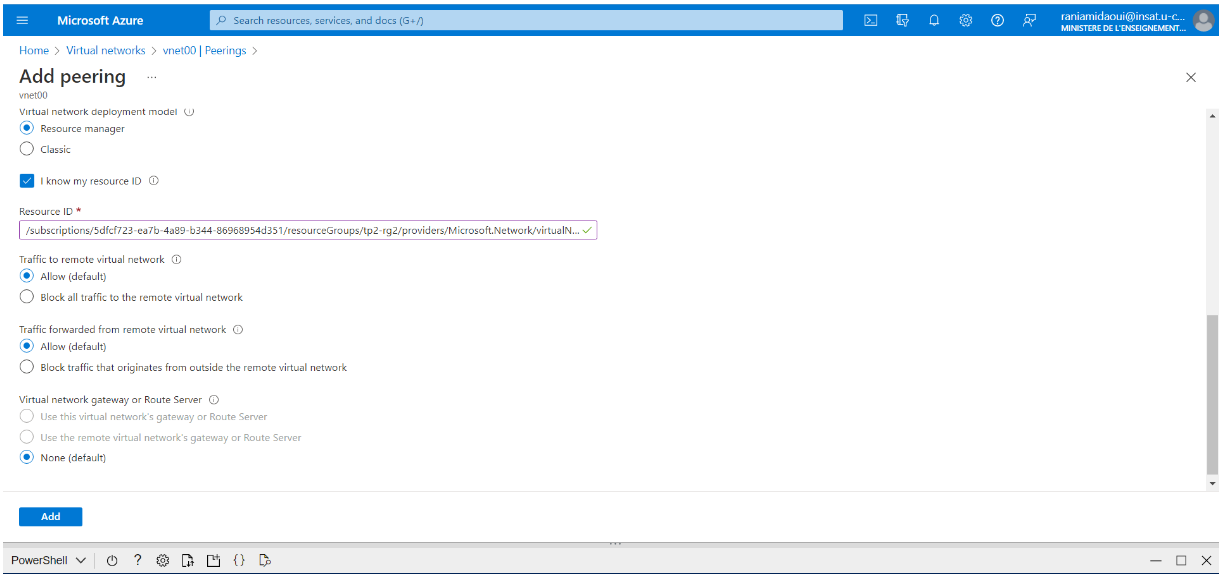
Traffic forwarded from remote virtual network
○ Allow (default)
● Block traffic that originates from outside the remote virtual network

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

Peering link name *: vnet01_to_vnet00

Virtual network deployment model


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home > Virtual networks > vnet00 | Peerings >
Add peering
vnet00
Add
PowerShell

Virtual network deployment model
● Resource manager
○ Classic

☑ I know my resource ID

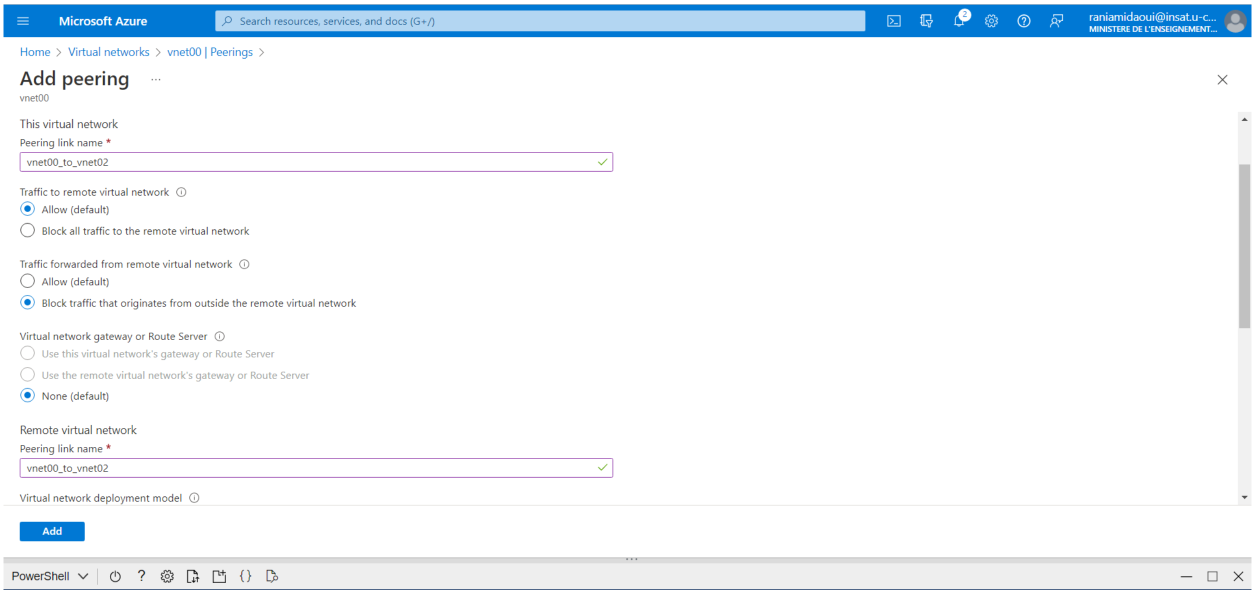
Resource ID *: /subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg2/providers/Microsoft.Network/virtualN...

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)

8- We Add another peering:


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home > Virtual networks > vnet00 | Peerings >
Add
PowerShell

Add peering

vnet00

This virtual network

Peering link name *

vnet00_to_vnet02

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

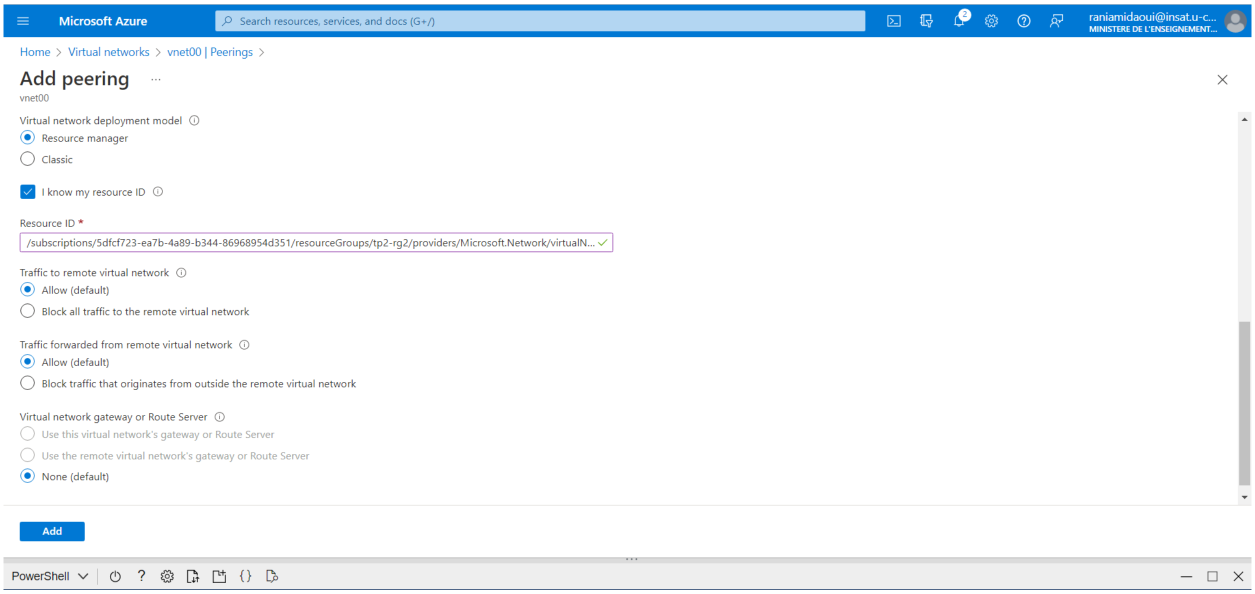
- ○ Use this virtual network's gateway or Route Server
- ○ Use the remote virtual network's gateway or Route Server
- ● None (default)

Remote virtual network

Peering link name *

vnet00_to_vnet02

Virtual network deployment model


Microsoft Azure
Search resources, services, and docs (G+/)
raniamidaoui@insat.u-c...
MINISTERE DE L'ENSEIGNEMENT...
Home > Virtual networks > vnet00 | Peerings >
Add
PowerShell

Add peering

vnet00

Virtual network deployment model

- ● Resource manager
- ○ Classic

☑ I know my resource ID

Resource ID *

/subscriptions/5dfcf723-ea7b-4a89-b344-86968954d351/resourceGroups/tp2-rg2/providers/Microsoft.Network/virtualN...

Traffic to remote virtual network

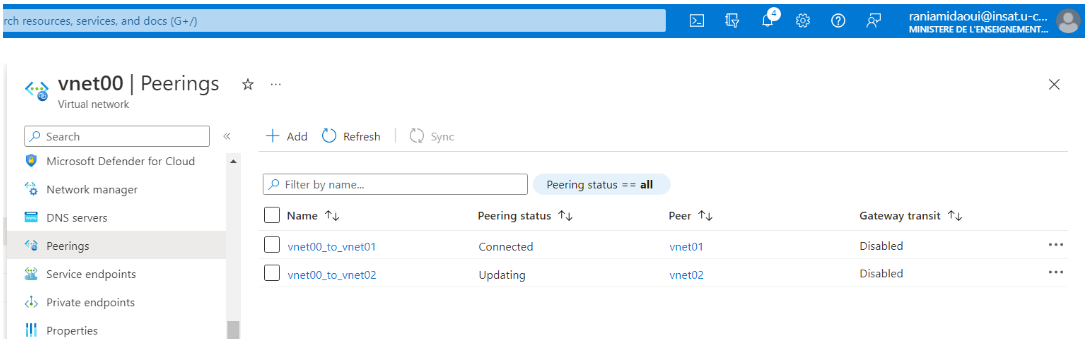
- ● Allow (default)
- ○ Block all traffic to the remote virtual network

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)

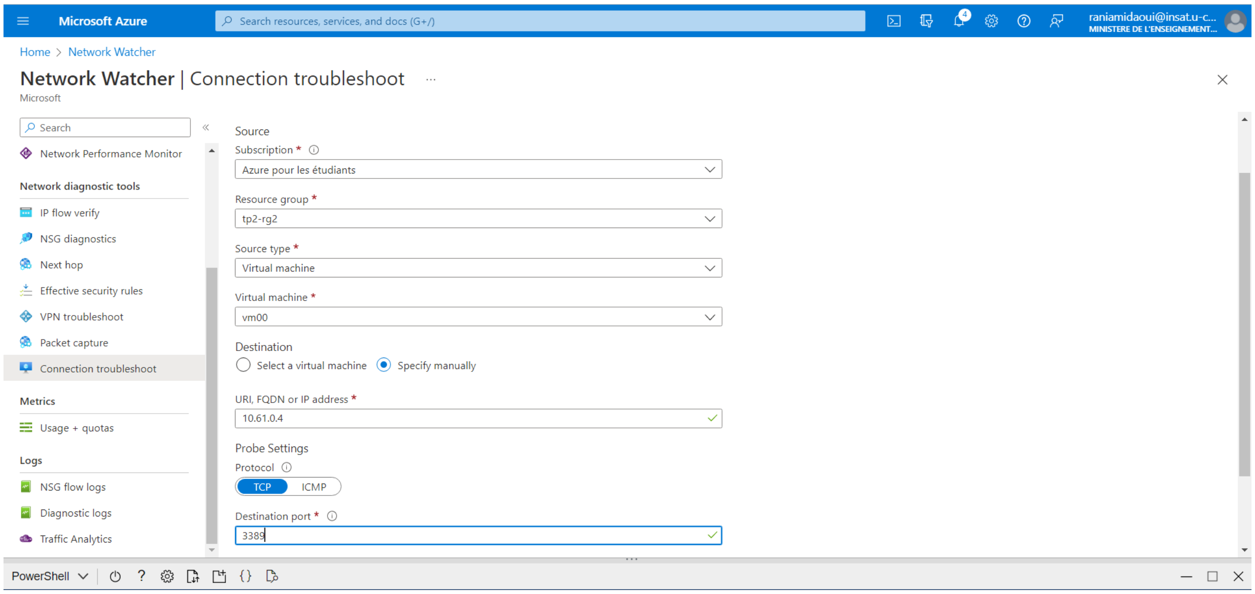


vnet00 | Peerings

Virtual network

Name	Peering status	Peer	Gateway transit
vnet00_to_vnet01	Connected	vnet01	Disabled
vnet00_to_vnet02	Updating	vnet02	Disabled

9- We test theTest now transitivity of virtual network peering by using Network Watcher and on Network Watcher - Connection troubleshoot blade we initiate a check:



Network Watcher | Connection troubleshoot

Microsoft

Source

Subscription * : Azure pour les étudiants

Resource group * : tp2-rg2

Source type * : Virtual machine

Virtual machine * : vm00

Destination

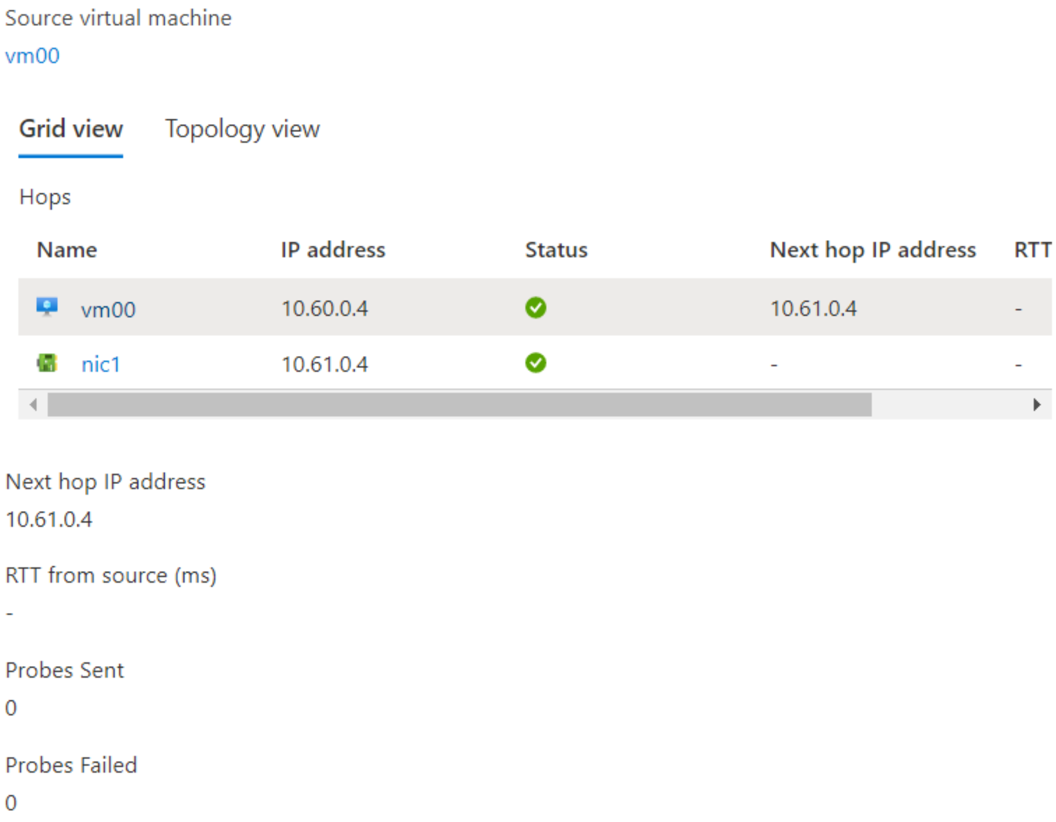
○ Select a virtual machine ● Specify manually

URI, FQDN or IP address * : 10.61.0.4

Probe Settings

Protocol: ● TCP ○ ICMP

Destination port * : 3389



Source virtual machine

vm00

Grid view | Topology view

Hops

Name	IP address	Status	Next hop IP address	RTT
vm00	10.60.0.4	✓	10.61.0.4	-
nic1	10.61.0.4	✓	-	-

Next hop IP address

10.61.0.4

RTT from source (ms)

-

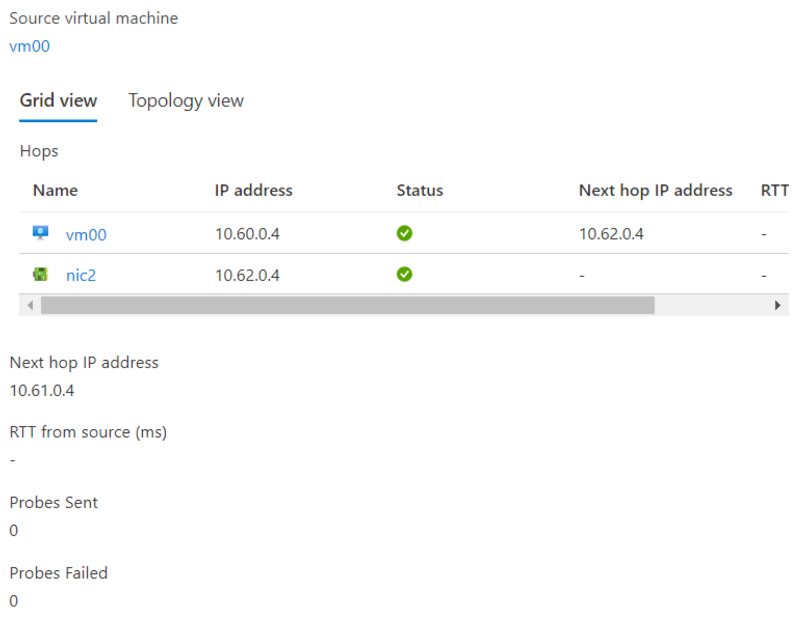
Probes Sent

0

Probes Failed

0

10- On Network Watcher - Connection troubleshoot blade we initiate a check:



Source virtual machine

vm00

Hops

Name	IP address	Status	Next hop IP address	RTT
vm00	10.60.0.4	✓	10.62.0.4	-
nic2	10.62.0.4	✓	-	-

Next hop IP address

10.61.0.4

RTT from source (ms)

-

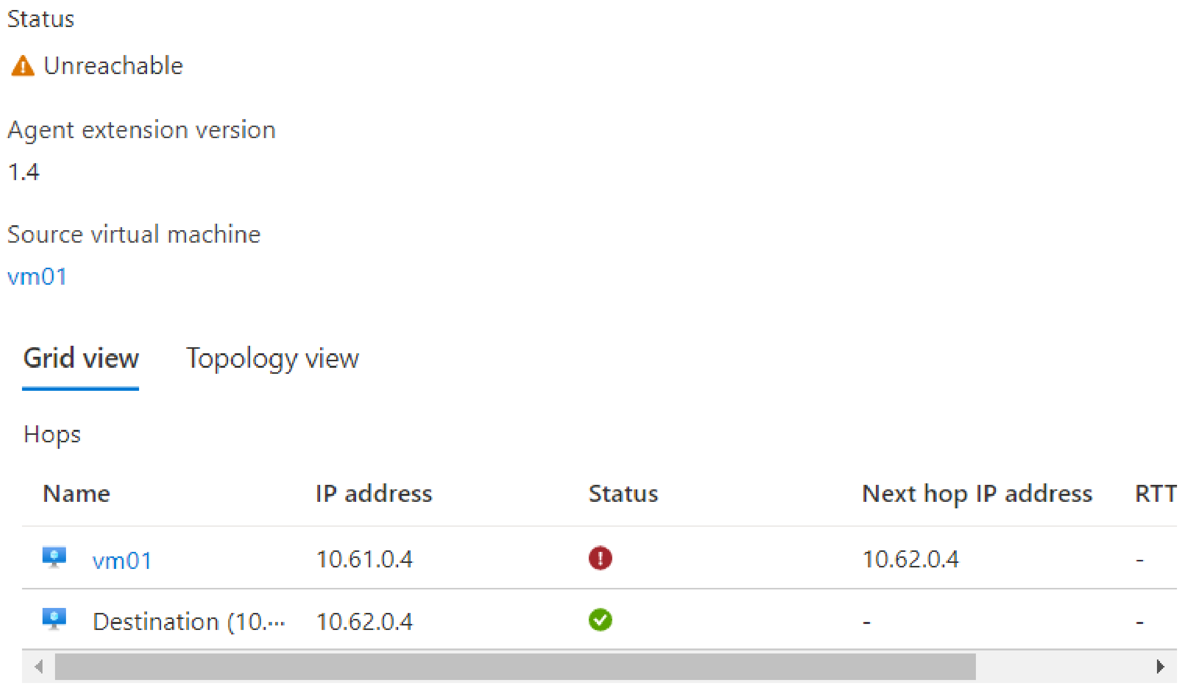
Probes Sent

0

Probes Failed

0

11- We initiate another check with different parameters:



Status

⚠ Unreachable

Agent extension version

1.4

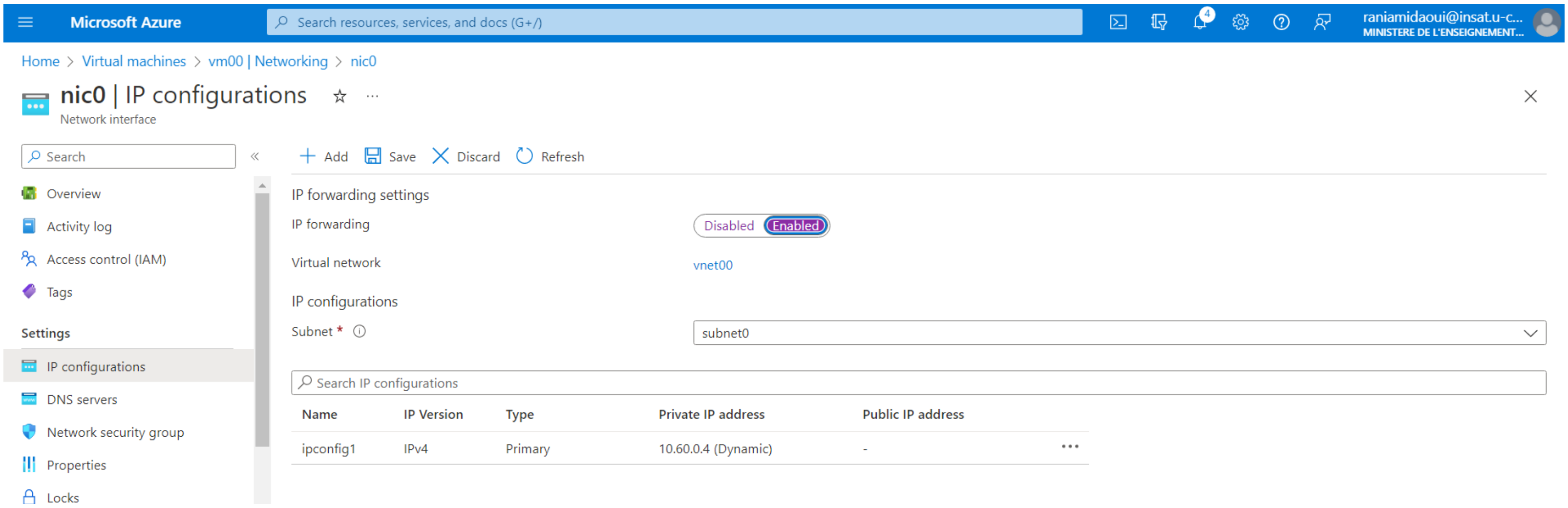
Source virtual machine

vm01

Hops

Name	IP address	Status	Next hop IP address	RTT
vm01	10.61.0.4	!	10.62.0.4	-
Destination (10....	10.62.0.4	✓	-	-

12- We set IP forwarding to Enabled for the nic0 network interface of vm00 :



IP forwarding settings

IP forwarding: Disabled [✓ Enabled]

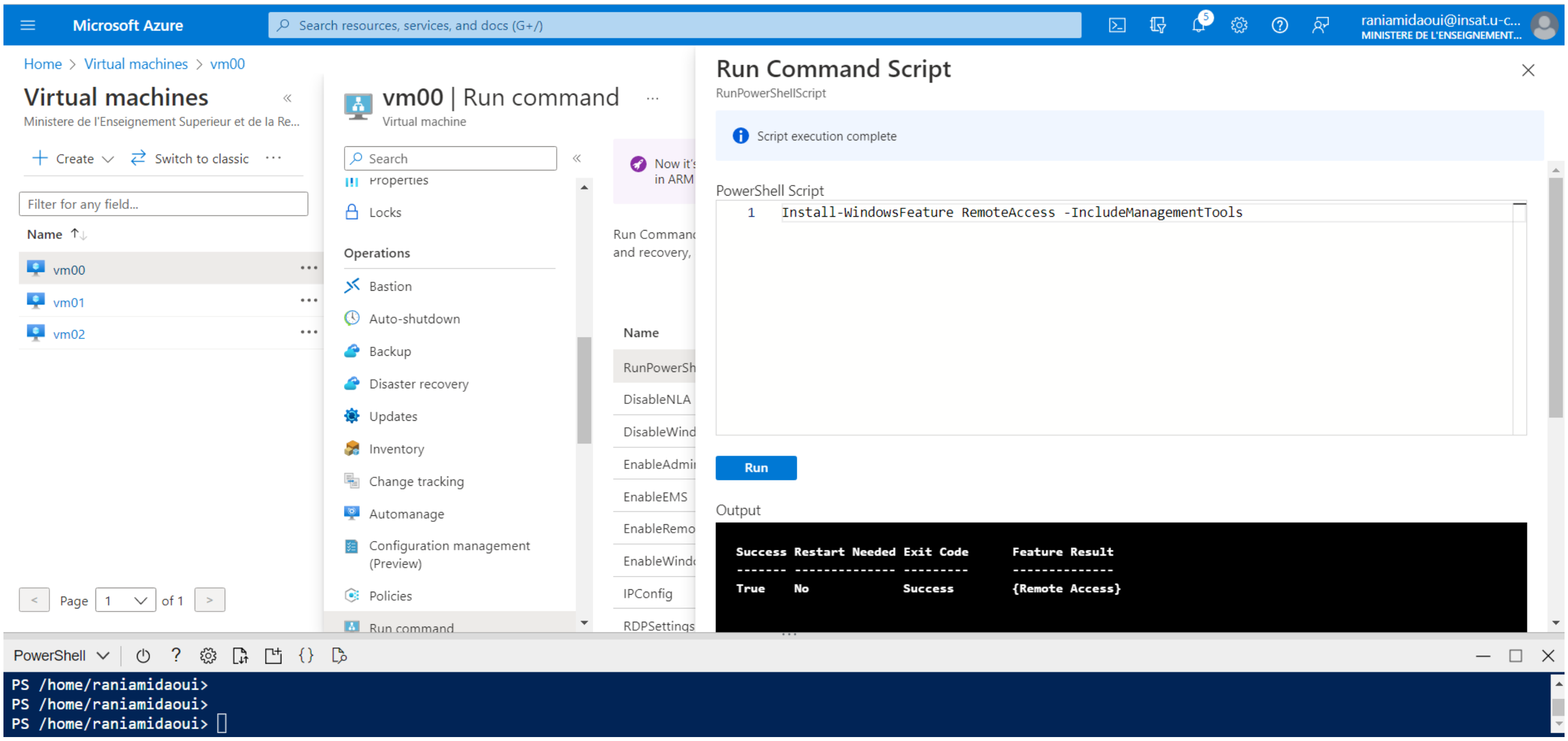
Virtual network: vnet00

IP configurations

Subnet *: subnet0

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

13- We install the Remote Access Windows Server role:



Run Command Script

RunPowerShellScript

Script execution complete

PowerShell Script

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

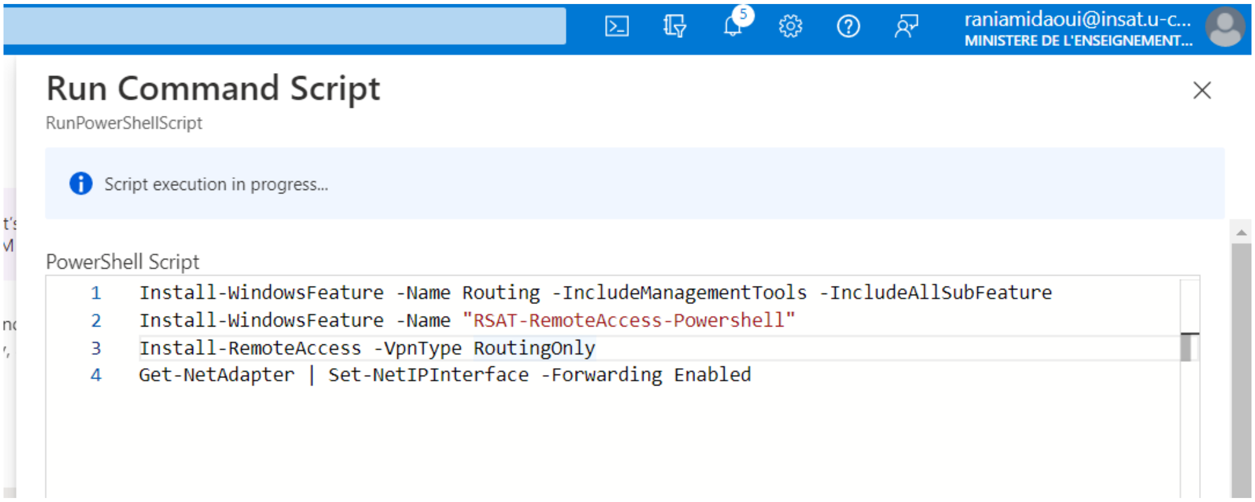
Run

Output

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

```
PS /home/raniamidaoui>
PS /home/raniamidaoui>
PS /home/raniamidaoui> 
```

14- We install the Routing role service:



Run Command Script

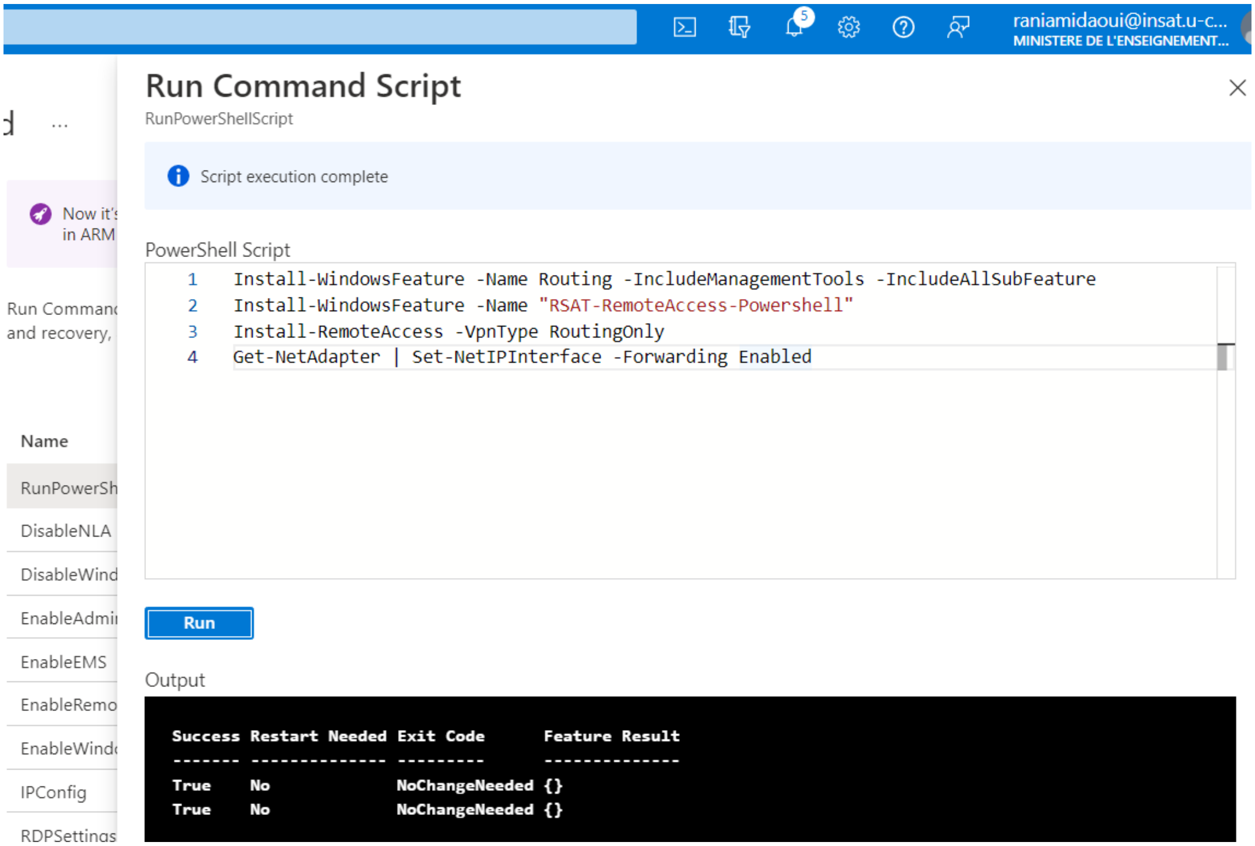
RunPowerShellScript

Script execution in progress...

PowerShell Script

```
Install-WindowsFeature -Name Routing -IncludeManagementTools -IncludeAllSubFeature
Install-WindowsFeature -Name "RSAT-RemoteAccess-Powershell"
Install-RemoteAccess -VpnType RoutingOnly
Get-NetAdapter | Set-NetIPInterface -Forwarding Enabled
```

15- We check that the command completed successfully:



Run Command Script

RunPowerShellScript

Script execution complete

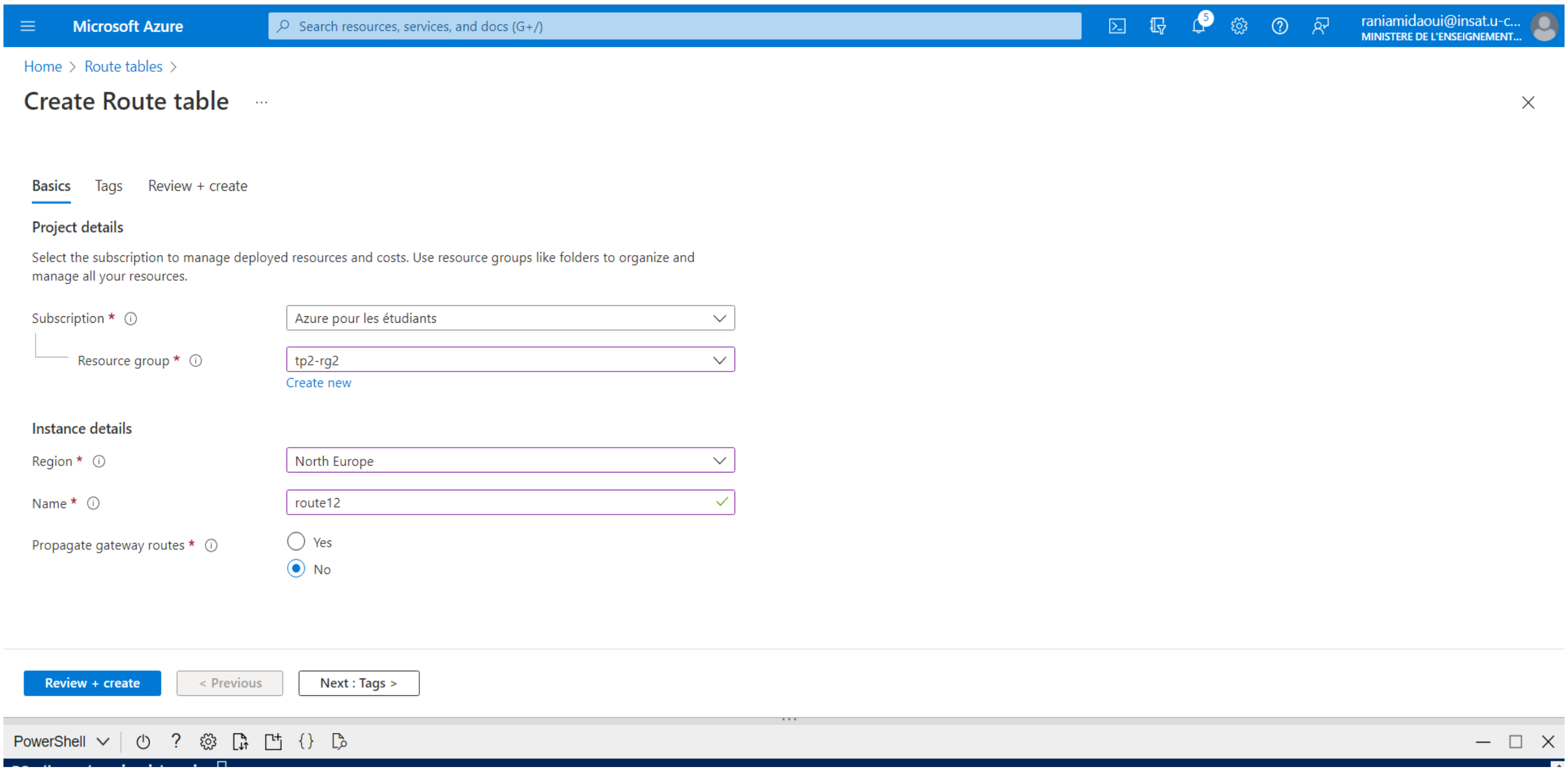
PowerShell Script

```
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             NoChangeNeeded {}
True    No             NoChangeNeeded {}
```

16- We create a route table:



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 pour les étudiants

Resource group * : tp2-rg2

Create new

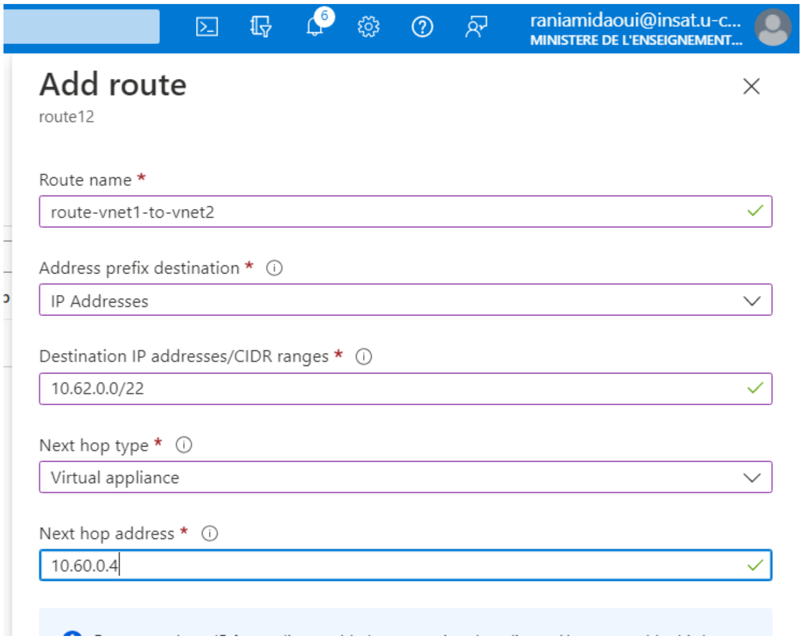
Instance details

Region * : North Europe

Name * : route12

Propagate gateway routes * : ○ Yes ● No

17- On the route12 route table blade, in the Settings section, we click Routes, and add a new route:



Add route

route12

Route name * : route-vnet1-to-vnet2

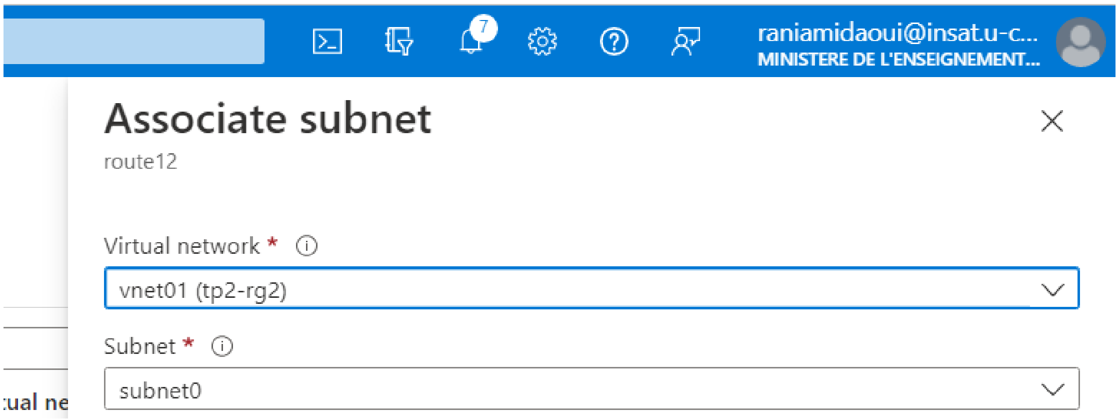
Address prefix destination * : IP Addresses

Destination IP addresses/CIDR ranges * : 10.62.0.0/22

Next hop type * : Virtual appliance

Next hop address * : 10.60.0.4

18- On the route12 route table blade, in the Settings section, we click Subnets, and then associate the route table route12 with a subnet:



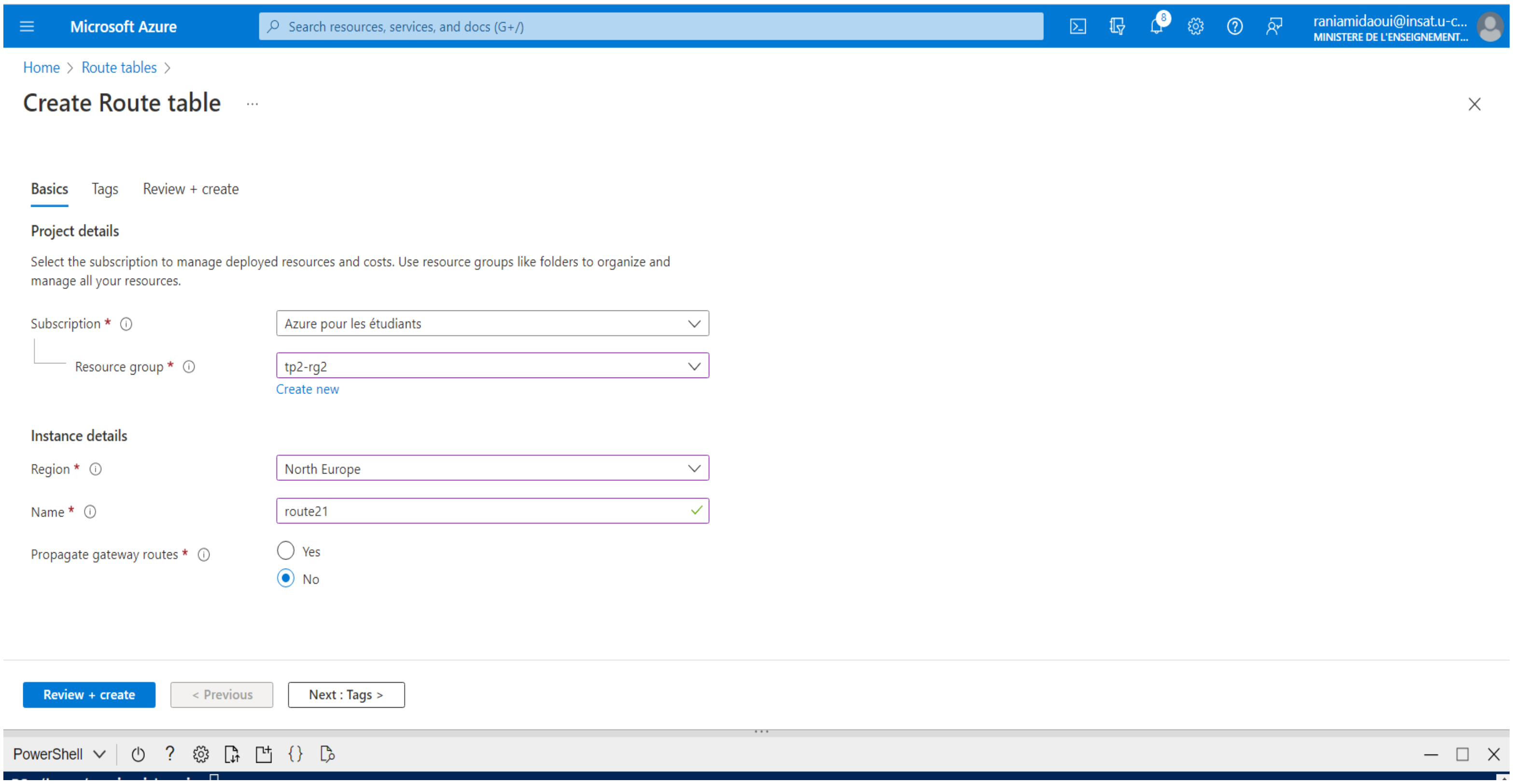
Associate subnet

route12

Virtual network * : vnet01 (tp2-rg2)

Subnet * : subnet0

19- We create another route table:



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 pour les étudiants

Resource group * : tp2-rg2

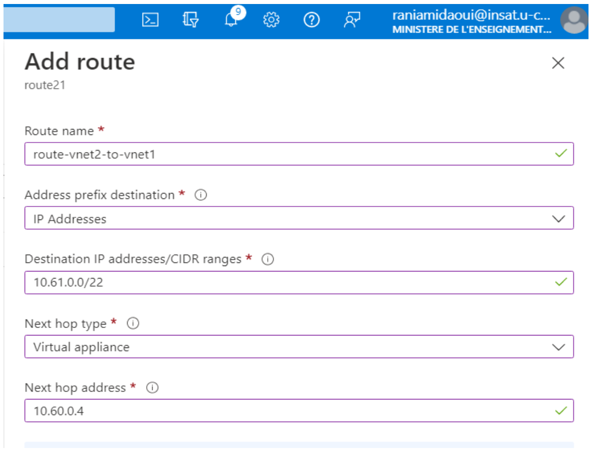
Instance details

Region * : North Europe

Name * : route21

Propagate gateway routes * : ○ Yes ● No

20- On the route21 route table blade, in the Settings section, we click Routes, and add a new route:



Add route

route21

Route name *
route-vnet2-to-vnet1

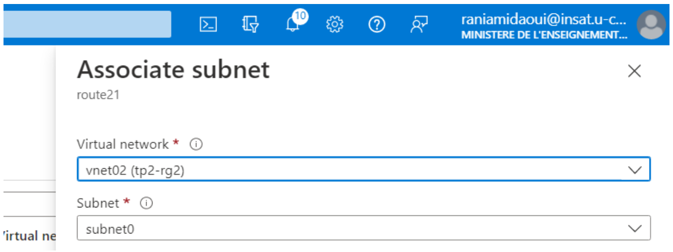
Address prefix destination *
IP Addresses

Destination IP addresses/CIDR ranges *
10.61.0.0/22

Next hop type *
Virtual appliance

Next hop address *
10.60.0.4

21- On the route21 route table blade, in the Settings section, we click Subnets, and then associate the route table route21 with a subnet:



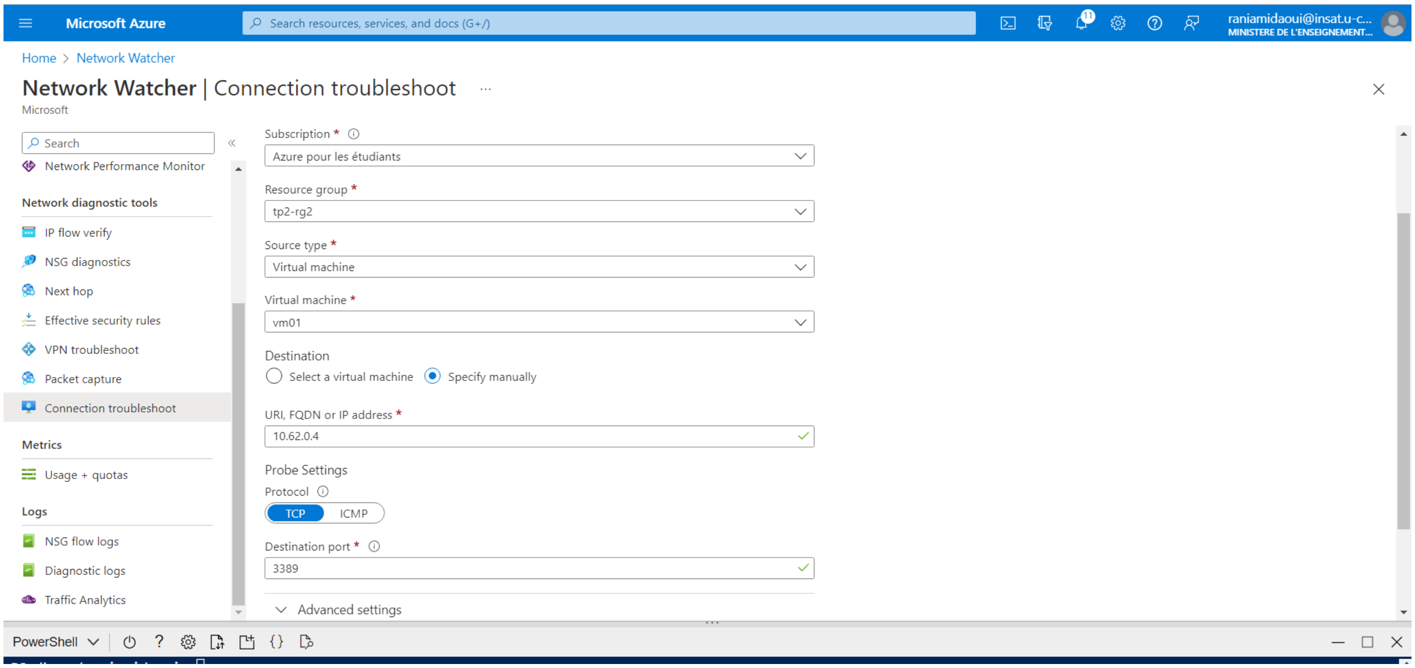
Associate subnet

route21

Virtual network *
vnet02 (tp2-rg2)

Subnet *
subnet0

22- On the Network Watcher - Connection troubleshoot blade we initiate a check



Network Watcher | Connection troubleshoot

Microsoft

Subscription *
Azure pour les étudiants

Resource group *
tp2-rg2

Source type *
Virtual machine

Virtual machine *
vm01

Destination
○ Select a virtual machine ● Specify manually

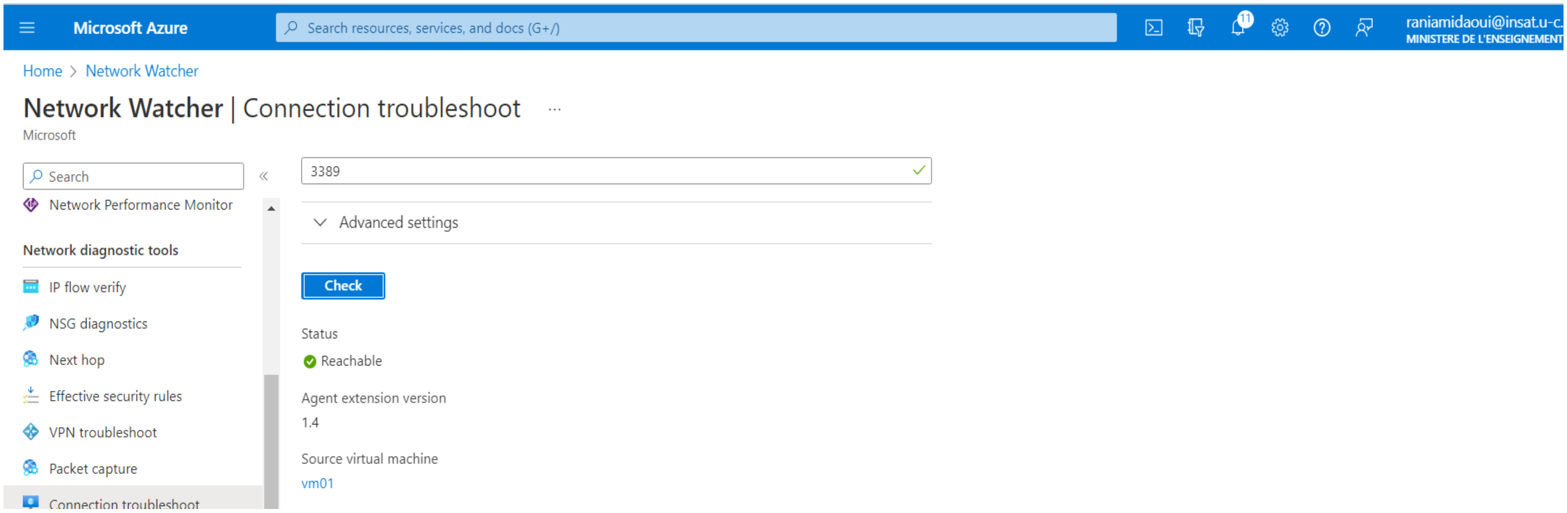
URI, FQDN or IP address *
10.62.0.4

Probe Settings

Protocol
TCP ICMP

Destination port *
3389

Advanced settings



Network Watcher | Connection troubleshoot

Microsoft

3389

Status

Reachable

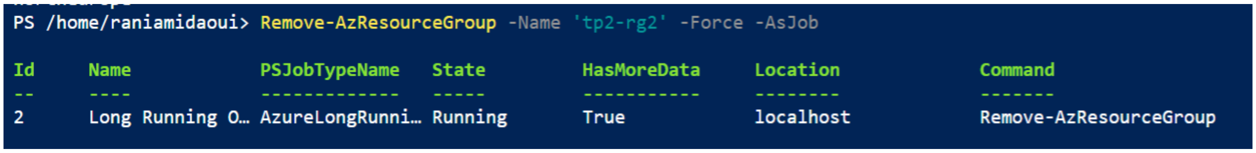
Agent extension version

1.4

Source virtual machine

vm01

23- We delete the resource group:



```
PS /home/raniamidaoui> Remove-AzResourceGroup -Name 'tp2-rg2' -Force -AsJob

Id     Name            PSJobTypeName   State         HasMoreData     Location             Command
--     ----            -------------   -----         -----------     --------             -------
2      Long Running O… AzureLongRunni… Running       True            localhost            Remove-AzResourceGroup
```

Conclusion:

This TP shows some great aspects of Azure virtual networks. It helped us get some hand-on experience by manipulating Azure Virtual Network and some networking components(Routes, route tables..), along with some virtual machines.

An Azure VNet is in conclusion similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.
It links many resources together and can be tied with many other services, helping us create efficient cloud solutions.