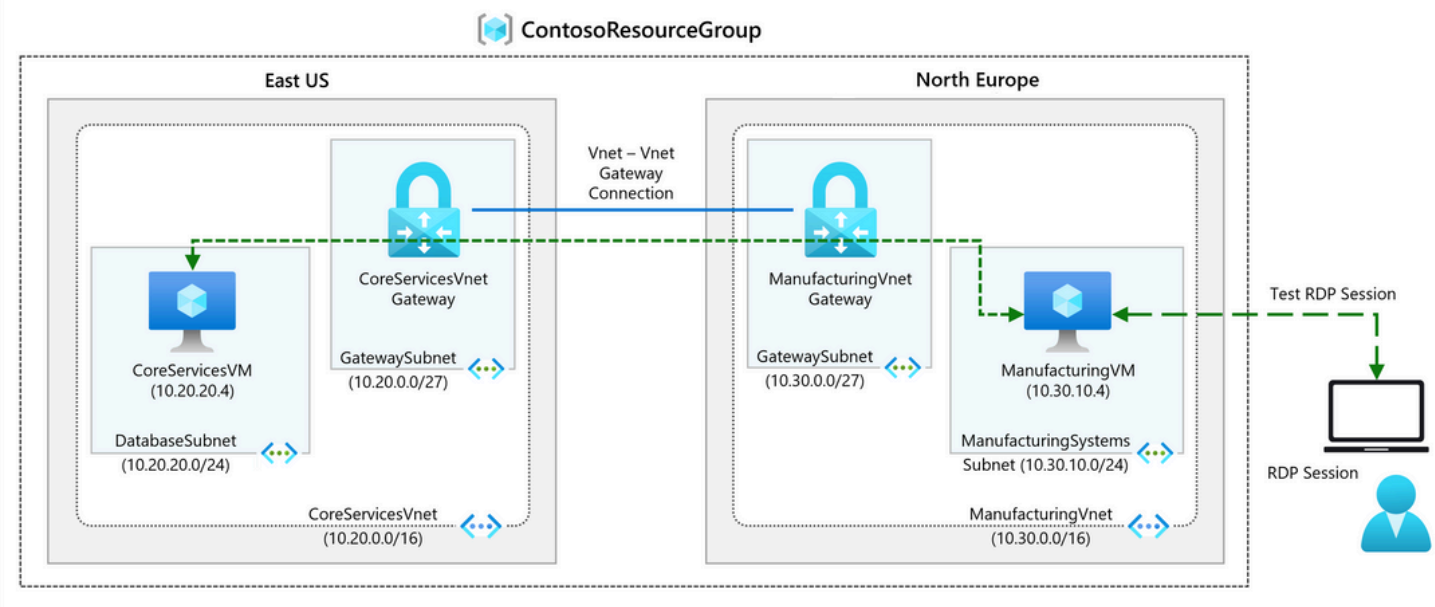
Azure Hybrid Networking Lab: Create and configure a virtual network gateway



In this exercise, you will:

- *+ Task 1: Create CoreServicesVnet and ManufacturingVnet*
- *+ Task 2: Create CoreServicesVM*
- *+ Task 3: Create ManufacturingVM*
- *+ Task 4: Connect to the VMs using RDP*
- *+ Task 5: Test the connection between the VMs*
- *+ Task 6: Create CoreServicesVnet Gateway*
- *+ Task 7: Create ManufacturingVnet Gateway*
- *+ Task 8: Connect CoreServicesVnet to ManufacturingVnet*
- *+ Task 9: Connect ManufacturingVnet to CoreServicesVnet*
- *+ Task 10: Verify that the connections connect*
- *+ Task 11: Test the connection between the VMs*

Task 1: Create CoreServicesVnet and ManufacturingVnet

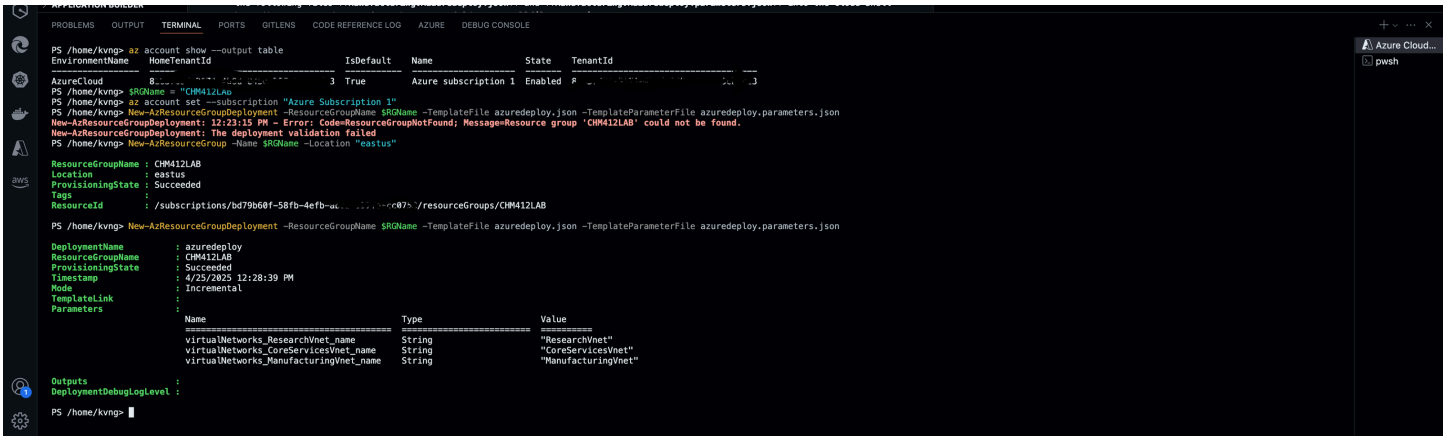
Deploy the following ARM templates to create the virtual network and subnets needed for this exercise via the ARM templates. What is an Azure Resource Manager (ARM) Template ??

An ARM Template is a JSON file designed to deploy multiple virtual networks (VNets) and their associated subnets across different Azure regions. It follows a modular and parameterized approach, allowing flexibility in naming and configuration.

$RGName = "CHM412LAB"

New-AzResourceGroup -Name $RGName -Location "eastus"

New-AzResourceGroupDeployment -ResourceGroupName $RGName -TemplateFile azuredeploy.json -TemplateParameterFile azuredeploy.parameters.json



Task 2&3: Create CoreServicesVM & ManufacturingVM

Deploy the following ARM templates to create the VMs needed for this exercise:

>**Note**: You will be prompted to provide an Admin password.

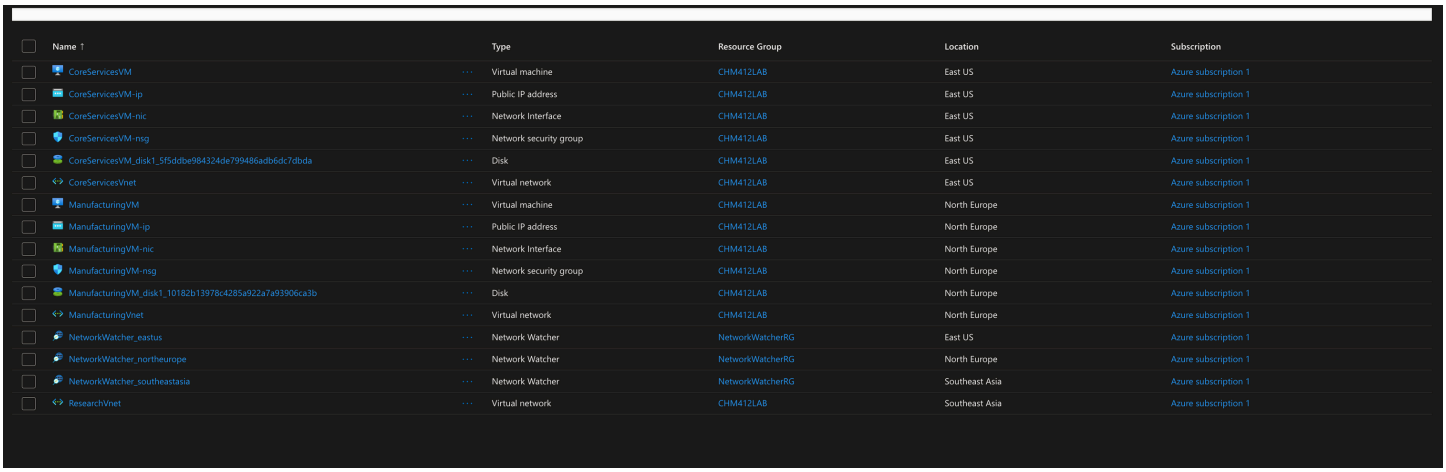
$RGName = "CHM412LAB"

New-AzResourceGroupDeployment -ResourceGroupName $RGName -TemplateFile CoreServicesVMazuredeploy.json -TemplateParameterFile CoreServicesVMazuredeploy.parameters.json

New-AzResourceGroupDeployment -ResourceGroupName $RGName -TemplateFile ManufacturingVMazuredeploy.json -TemplateParameterFile ManufacturingVMazuredeploy.parameters.json

When the deployment is complete, go to the Azure portal home page, and then select **Virtual Machines**. Verify that the virtual machine has been created.

To effectively query me for creating ARM templates like the ones you've shared, you can use a structured and concise prompt that includes the following key elements:



Suggested Query Format for ARM Template Creation using Copilot

1. Purpose of the Template

Clearly state the purpose of the template, such as creating VNets and their subnets with virtual machines, virtual networks, or other Azure resources.

2. Resource Details

Provide a list of resources to include, such as VMs, NSGs, subnets, public IPs, etc., along with any specific configurations (e.g., VM size, region, number of instances).

3. Parameters and Variables

Specify any parameters (e.g., `vmName`, `adminUsername`) and variables (e.g., `subnetRef`, `nsgName`) you want to use for flexibility and reusability.

4. Best Practices

Mention that Azure best practices should be followed, such as using secure strings for sensitive data, defining dependencies, and ensuring modularity.

5. Output Requirements

Indicate if you need outputs, such as resource IDs or connection strings.

Example Query for Future Use: Create an Azure Resource Manager (ARM) template to deploy the following resources:

1. A virtual network named `CoreServicesVnet` with a subnet named `DatabaseSubnet`.

2. A virtual machine named `CoreServicesVM` with the following:

   - Size: `Standard_D2s_v3`

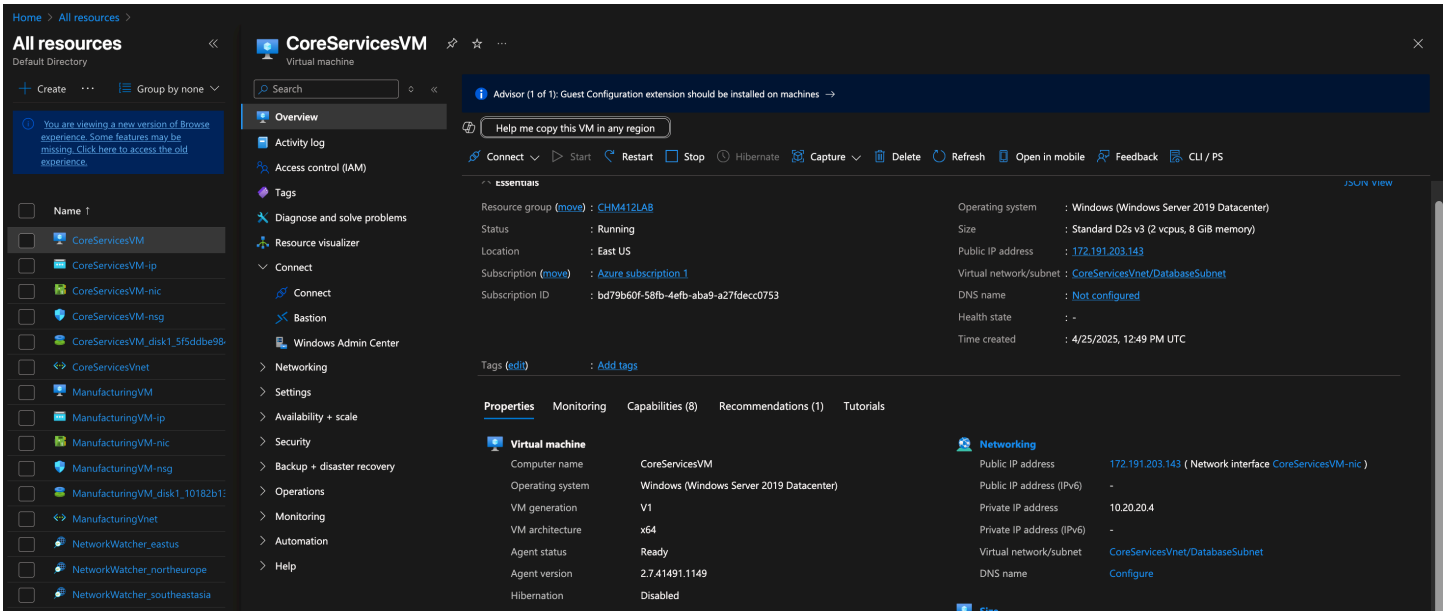
   - Admin username and password as parameters

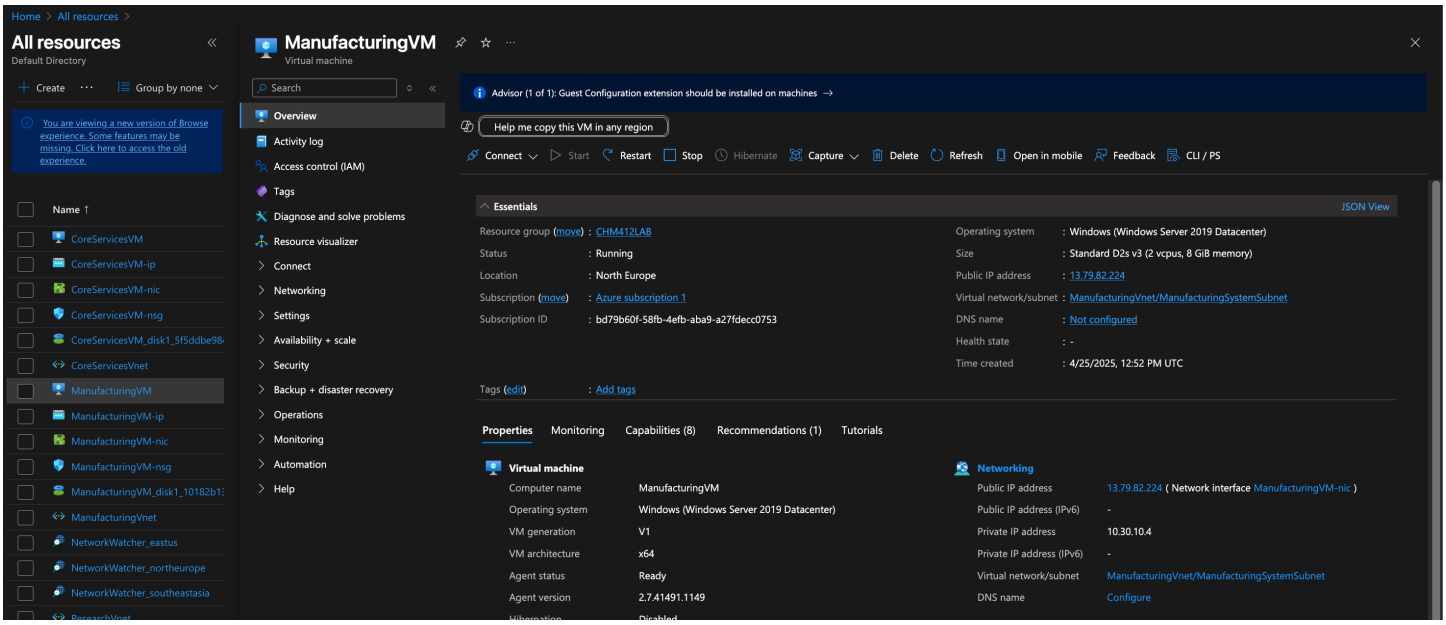
   - A public IP address and a network interface

   - A Network Security Group (NSG) allowing RDP (port 3389)

3. Follow Azure best practices, including secure strings for sensitive data, modular design, and explicit dependencies.

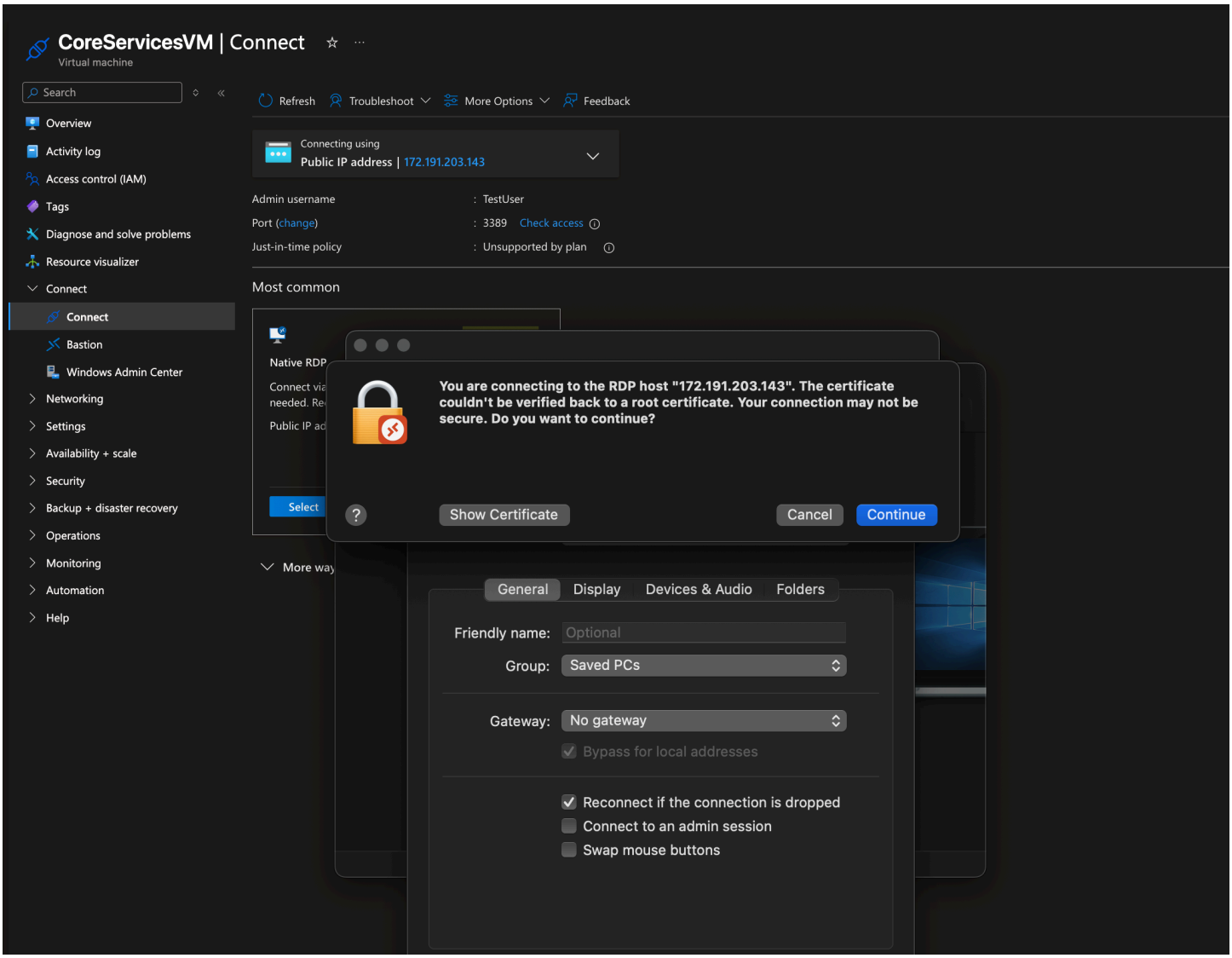
4. Include outputs for the VM's public IP address and NIC ID.





Task 4: Connect to the VMs using RDP

On the Azure Portal home page, select Virtual Machines >> Select ManufacturingVM / CoreServicesVM >> On ManufacturingVM/CoreServicesVM, select Connect and then RDP. >> Select Download RDP file >> Save the RDP file to your desktop >> Connect to ManufacturingVM/CoreServicesVM using the RDP file, and the username **TestUser** and the password you provided during deployment. After connecting, minimize the RDP session >> On both VMs, in Choose privacy settings for your device, select **Accept**.



Note the IPv4 address.

Task 5: Test the connection between the VMs

On the **ManufacturingVM**, open PowerShell. Use the following command to verify that there is no connection to CoreServicesVM on CoreServicesVnet. Be sure to use the IPv4 address for CoreServicesVM. The test connection should fail, and you will see a result similar to the following. However you will notice that, we get a connection via their public IP, But no connection via their private-IP address.

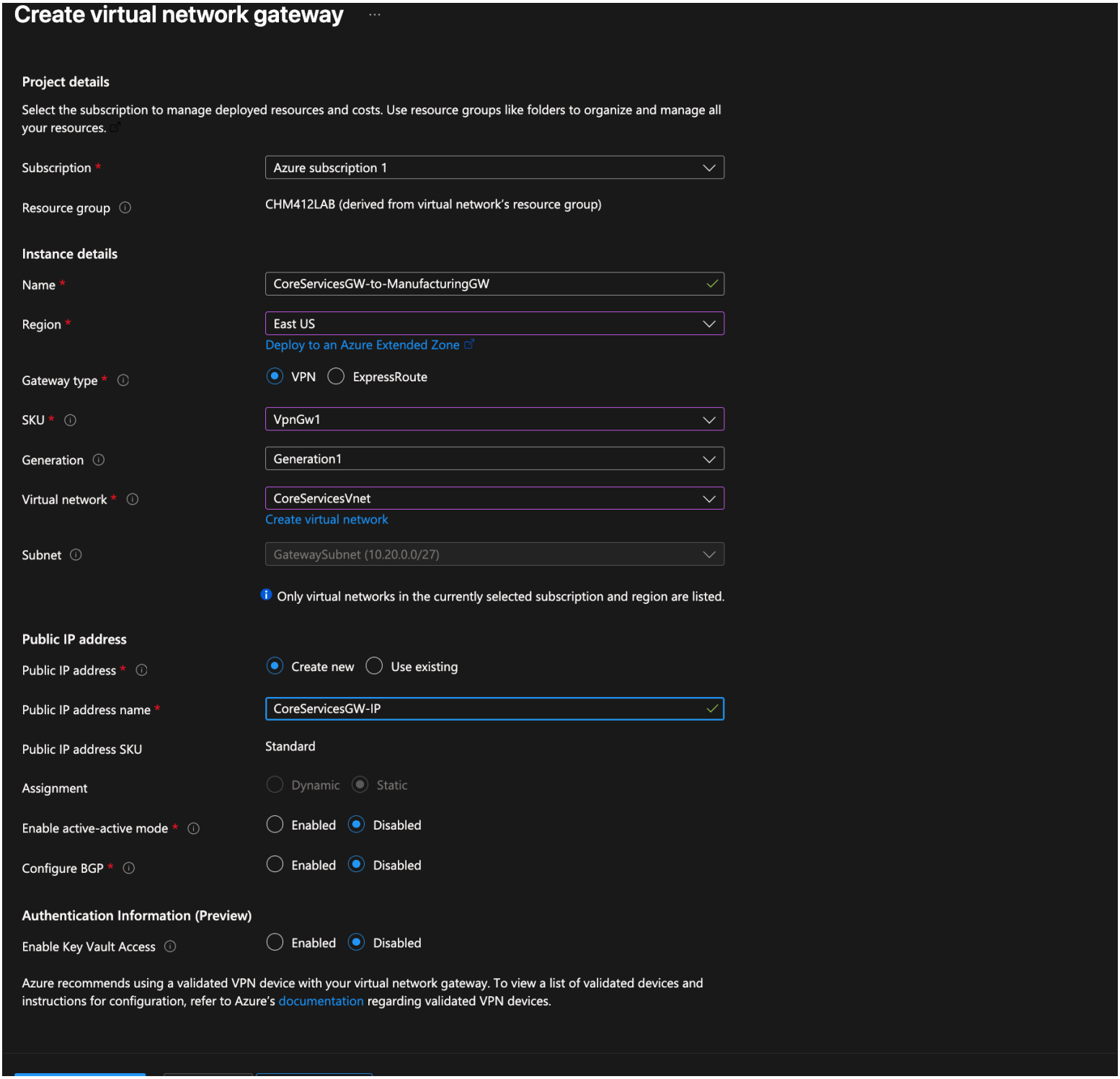
Test-NetConnection “Private-IP” -port 3389

```
PS C:\Users\TestUser> Test-NetConnection 172.191.203.143 -port 3389


ComputerName     : 172.191.203.143
RemoteAddress    : 172.191.203.143
RemotePort       : 3389
InterfaceAlias   : Ethernet
SourceAddress    : 10.30.10.4
TcpTestSucceeded : True



PS C:\Users\TestUser> Test-NetConnection 10.20.20.4 -port 3389
WARNING: TCP connect to (10.20.20.4 : 3389) failed
WARNING: Ping to 10.20.20.4 failed with status: TimedOut


ComputerName           : 10.20.20.4
RemoteAddress          : 10.20.20.4
RemotePort             : 3389
InterfaceAlias         : Ethernet
SourceAddress          : 10.30.10.4
PingSucceeded          : False
PingReplyDetails (RTT) : 0 ms
TcpTestSucceeded       : False



PS C:\Users\TestUser>
```

```
PS C:\Users\TestUser> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . : fol25ml2ofcuje3w0w44e3lmif.bx.internal.cloudapp.net
   Link-local IPv6 Address . . . . . : fe80::465:21e3:3e2f:c559%6
   IPv4 Address. . . . . . . . . . . : 10.20.20.4
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.20.20.1
PS C:\Users\TestUser> Test-NetConnection 13.79.82.224 -port 3389


ComputerName     : 13.79.82.224
RemoteAddress    : 13.79.82.224
RemotePort       : 3389
InterfaceAlias   : Ethernet
SourceAddress    : 10.20.20.4
TcpTestSucceeded : True



PS C:\Users\TestUser> Test-NetConnection 10.30.10.4 -port 3389
WARNING: TCP connect to (10.30.10.4 : 3389) failed
WARNING: Ping to 10.30.10.4 failed with status: TimedOut


ComputerName           : 10.30.10.4
RemoteAddress          : 10.30.10.4
RemotePort             : 3389
InterfaceAlias         : Ethernet
SourceAddress          : 10.20.20.4
PingSucceeded          : False
PingReplyDetails (RTT) : 0 ms
TcpTestSucceeded       : False



PS C:\Users\TestUser>
```

Task 6&7: Create CoreServicesVnet Gateway & Create ManufacturingVnet Gateway

In Search resources, services, and docs (G+/), enter Virtual network gateway, and then select Virtual network gateways from the results. In Virtual network gateways, select + Create. Use the information in the following table to create the virtual network gateway:

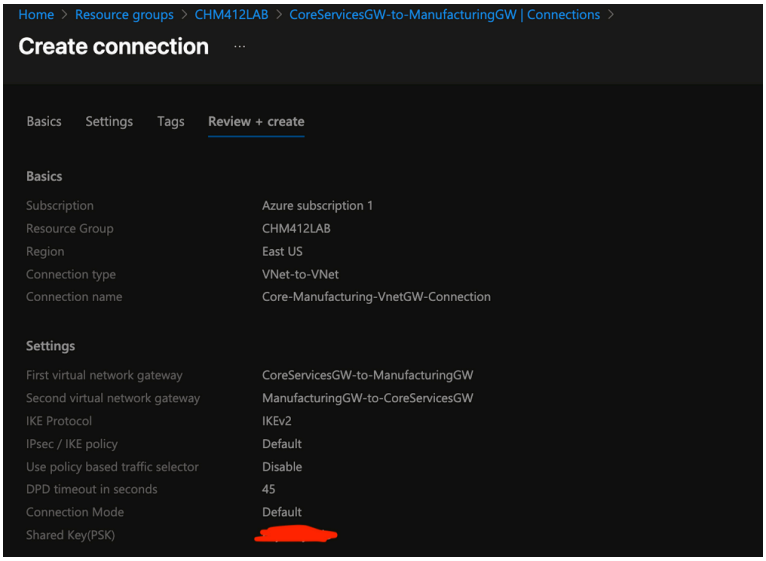


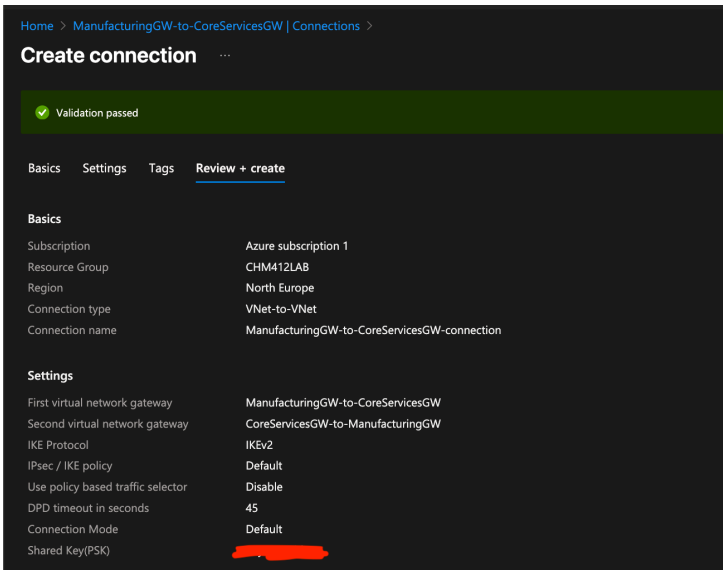
Note: *It can take up to 15 - 30 minutes to create a virtual network gateway. You don't need to wait for the deployment to complete. Proceed to creating the next gateway.*

Task 8&9: Connect CoreServicesVnet to ManufacturingVnet and ViceVersa

In Search resources, services, and docs (G+/), enter Virtual network gateway, and then select Virtual network gateways from the results. In Virtual network gateways, select CoreServicesVnetGateway. In CoreServicesGateway, select Connections, and then select + Add. *Note: You will not be able to complete this configuration until the virtual network gateways are fully deployed.* Use this information and the Settings tab to create the virtual network gateway.

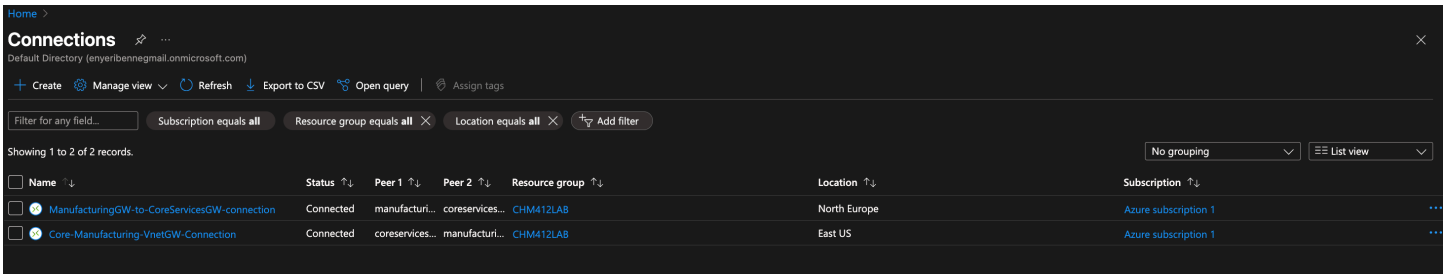
- | Option | Value
- | ---------- | ---------- |
- | Name |
- | Connection type | VNet-to-VNet
- | Location |
- | First virtual network gateway | ManufacturingVnetGateway
- | Second virtual network gateway | CoreServicesVnetGateway
- | Shared key (PSK) | abc123
- | Use Azure Private IP Address | Not selected
- | Enable BGP | Not selected
- | IKE Protocol | IKEv2
- | Subscription | No changes required
- | Resource group | No changes required

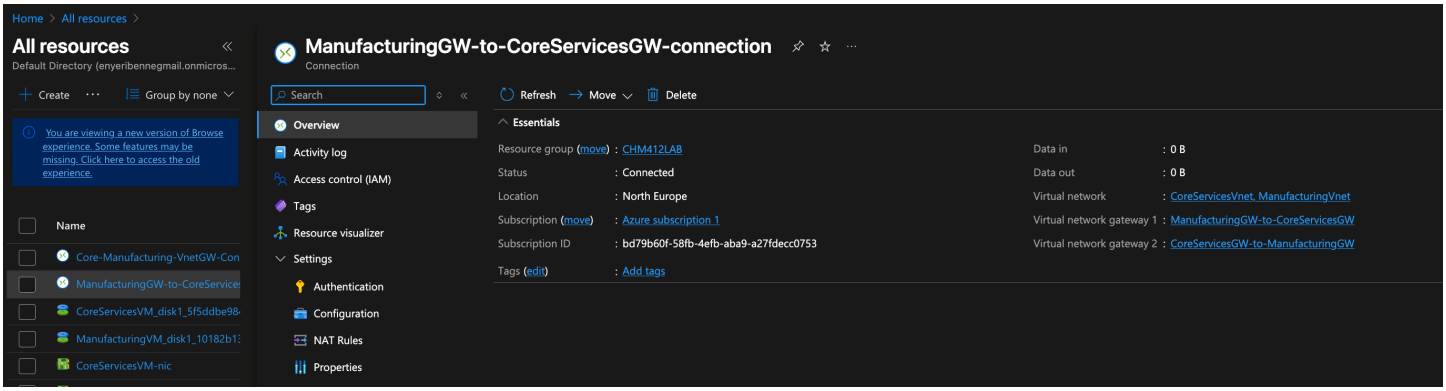


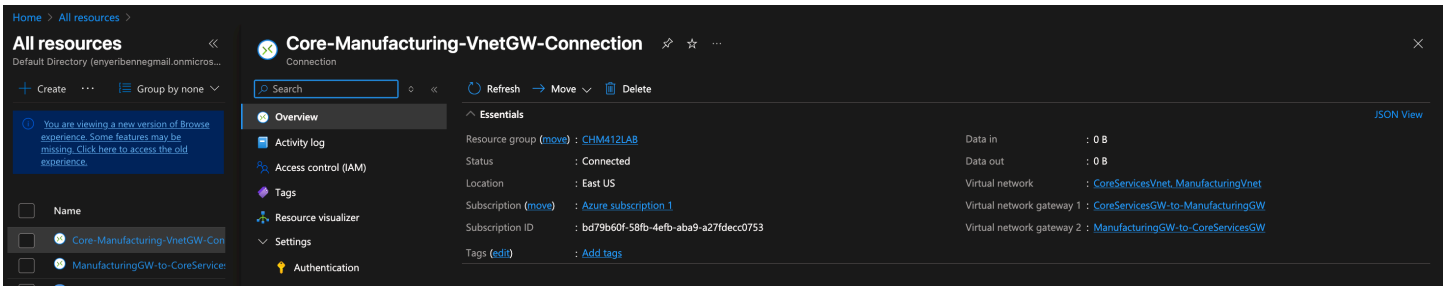


Task 10: Verify that the connections connect

In Search resources, services, and docs (G+/), enter vpn, and then select connections from the results. Wait until the status of both connections is Connected. You may need to refresh your screen.







Task 11: Test the connection between the VMs

On the ManufacturingVM, open PowerShell. Use the following command to verify that there is now a connection to CoreServicesVM on CoreServicesVnet. Be sure to use the IPv4 address for CoreServicesVM. The test connection should succeed, and you will see a result similar to the following: Close the Remote Desktop connection windows.

Test-NetConnection 10.20.20.4 -port 3389

```
PS C:\Users\TestUser> Test-NetConnection 10.30.10.4 -port 3389


ComputerName     : 10.30.10.4
RemoteAddress    : 10.30.10.4
RemotePort       : 3389
InterfaceAlias   : Ethernet
SourceAddress    : 10.20.20.4
TcpTestSucceeded : True
```

```
PS C:\Users\TestUser> Test-NetConnection 10.20.20.4 -port 3389


ComputerName     : 10.20.20.4
RemoteAddress    : 10.20.20.4
RemotePort       : 3389
InterfaceAlias   : Ethernet
SourceAddress    : 10.30.10.4
TcpTestSucceeded : True
```

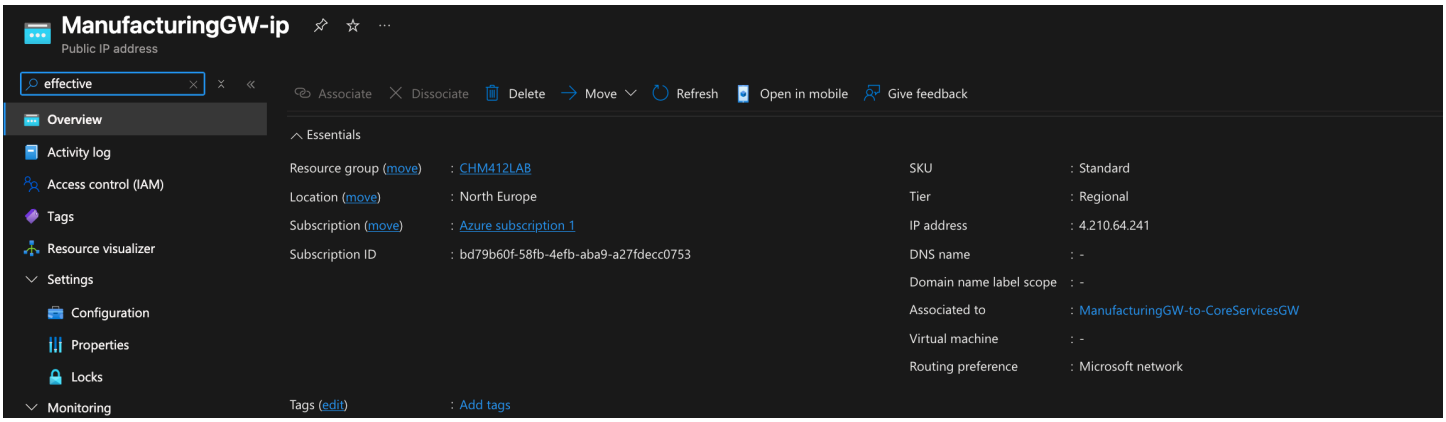
Learn more with self-paced training and Extend your learning with Copilot

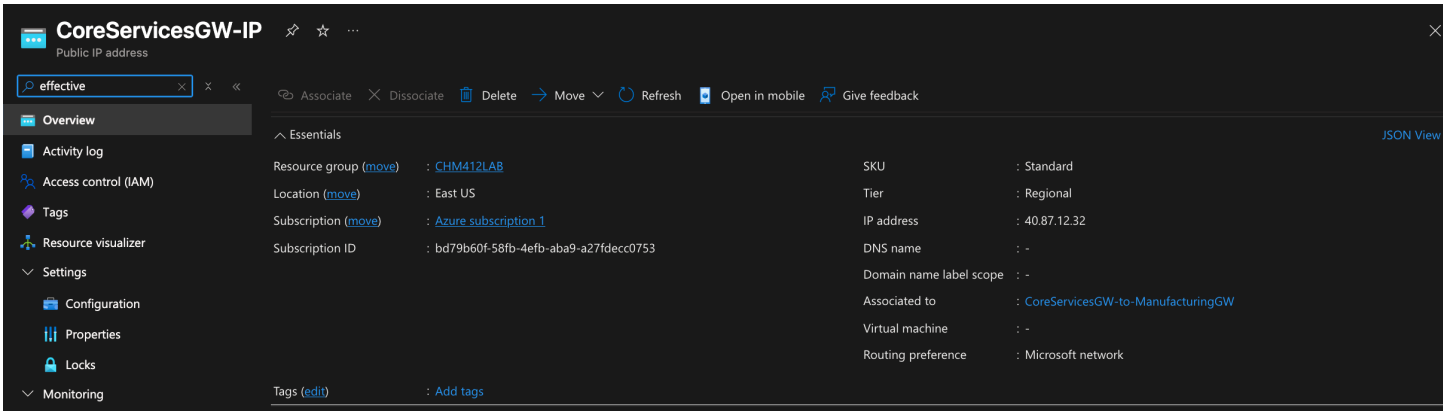
+ [Connect your on-premises network to Azure with VPN Gateway](https://learn.microsoft.com/training/modules/connect-on-premises-network-with-vpn-gateway/). In this module, you will how to use CLI to provision VPN gateways.

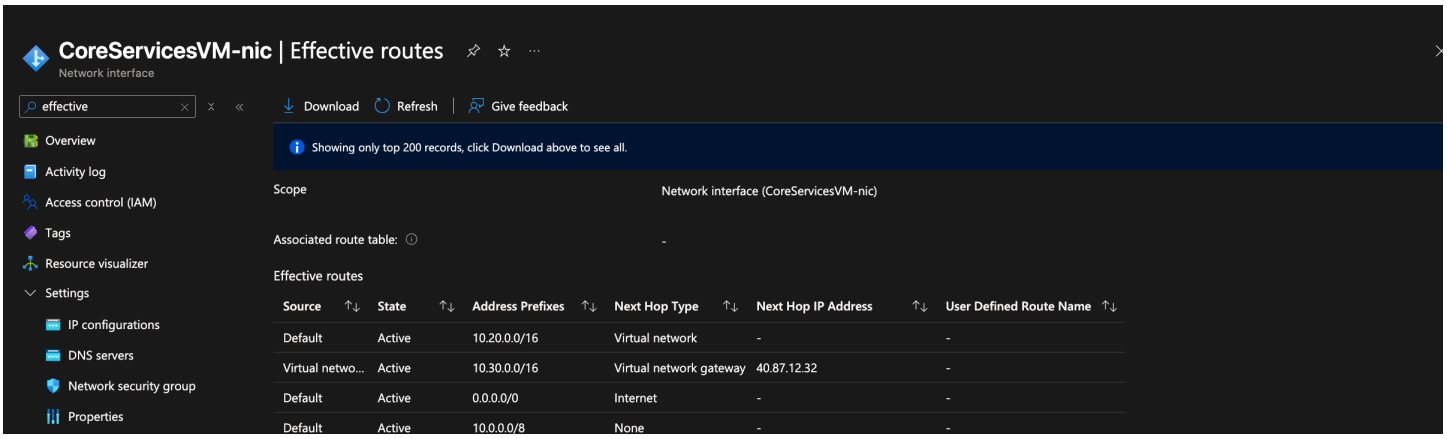
+ [Troubleshoot VPN gateways in Microsoft Azure] (https://learn.microsoft.com/training/modules/troubleshoot-vpn-gateways/). In this module, you learn how to monitor and troubleshoot site-to-site and point-to-site VPNs.

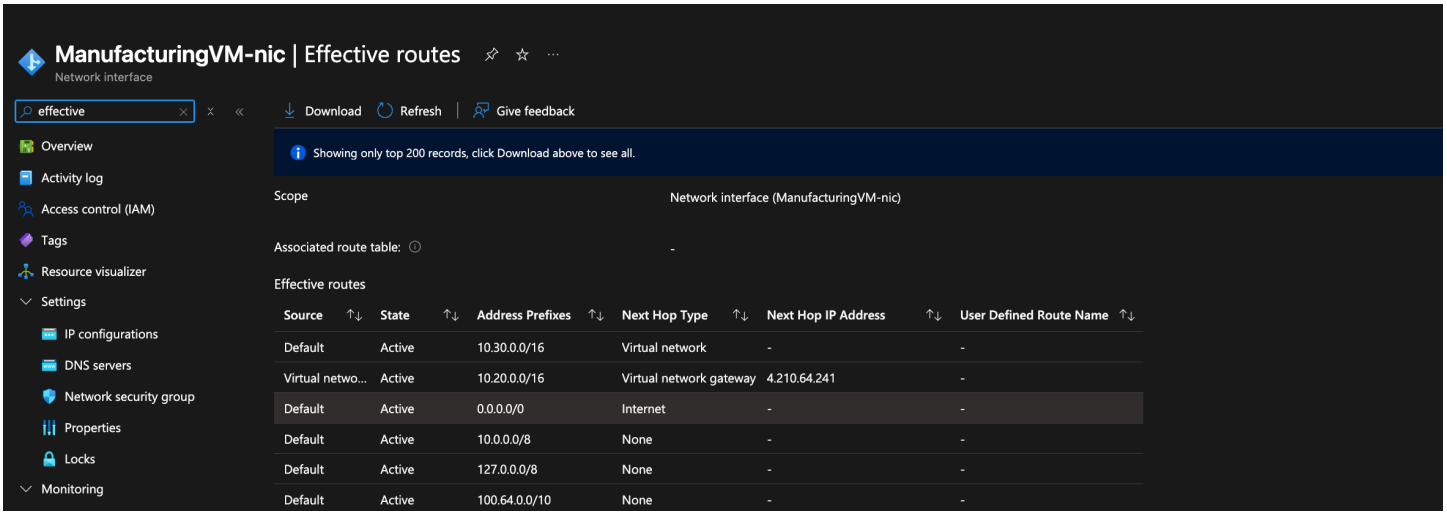
Free Pointer. During Setup, i thought to myself, how possible is it for all this to work, so i dived in deeper. *When setting up a Virtual Network Gateway, it is essential to create a dedicated subnet, known as the GatewaySubnet, within the virtual network where the gateway will be deployed. This subnet is specifically reserved for the gateway's resources and must be configured before proceeding with the gateway setup. Additionally, ensure that the location of the subnet matches the region of the virtual network, as this alignment is required during the initial configuration of the Virtual Network Gateway. Proper planning of the subnet's size and location is critical to ensure a smooth deployment process.*

Finally the Screenshots below confirms my suspicion that Azure adds Optional default routes to either specific subnets or all subnets within a VNet. When you add a VNet Gateway to a VNet, Azure adds one or more routes with Virtual Network Gateway as the next hop type. The Gateway adds the Routes to the subnet. Viewing the effective Routes for each network Interface created and you will find a route for the VNet Gateway: via the Azure portal.









Key takeaways

Finally, the key takeaways summarize the main concepts, such as the types of Azure VPN connections (Site-to-Site, Point-to-Site, and VNet-to-VNet), their use cases, and the importance of selecting the appropriate VPN Gateway SKU based on performance and requirements. This structured guide is ideal for learners aiming to understand and implement hybrid networking solutions in Azure.