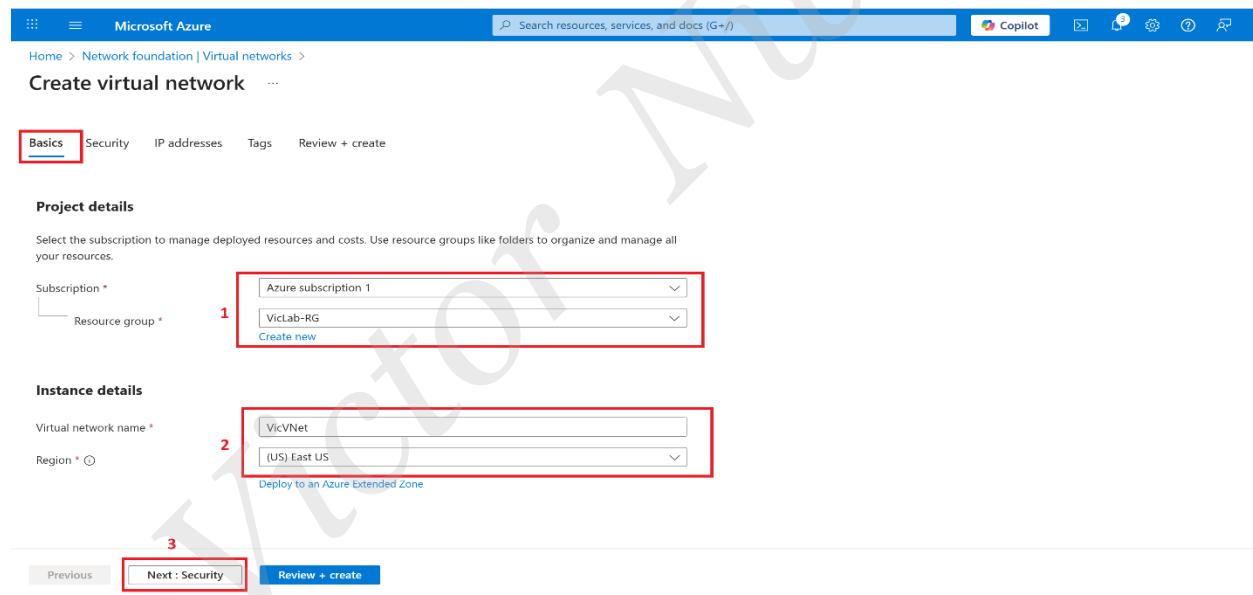


Azure Networking Skills in Action: Step-by-Step Point-to-Site VPN Configuration

Step 1: Creating a Virtual Network

- ◆ Sign in to the **Azure Portal** → Search for **Virtual Network** in the Azure portal. → In the search results, select the virtual network right under **Marketplace**
- ◆ On the Virtual network page, click on **Create**
- ◆ Under the Basics tab, choose the Subscription you want to use for this project → Resource group can be selected from an existing resource group, or a new resource group can be created by clicking on Create new.
- ◆ Enter a preferred name for the virtual network
- ◆ Choose a preferred location for the Virtual Network



For this tutorial, we will leave the Security tab with the default settings and then click on Next.

- ◆ On the IP addresses tab, click on the Add an IP address space and select the Address space type radio button for IPv4.

We will be using the 172.16.0.0/16 address space for our network. For the subnet, we will create two subnets: GatewaySubnet with a subnet address space of 172.16.0.0/27, and another subnet address space for our Virtual Machine.

Step-by-Step Point-to-Site VPN Configuration

The screenshot shows the Azure portal interface for creating a virtual network. The 'IP addresses' tab is active. A new subnet is being added, indicated by the 'Add a subnet' button and the 'Add' button in the subnet configuration dialog. The subnet configuration dialog includes fields for Subnet purpose (set to Virtual Network Gateway), Name (GatewaySubnet), IPv4 address range (172.76.0.0/16), Starting address (172.76.0.0), and Size (27 (32 addresses)).

Step 2: Let's deploy the VPN Gateway

- ◆ Search for Virtual Network Gateway in the Azure portal.
- ◆ In the Search results, under Marketplace, select Virtual network gateway.

The screenshot shows the Azure search results page with the query 'virtual network gateway'. The results list includes 'Virtual network gateway' under the Marketplace section. Other results like 'Virtual networks' and 'Virtual machines' are also shown.

- ◆ Select the subscription from the drop-down.
- ◆ The Resource Group will be auto-filled when we select the virtual network.
- ◆ Enter any preferred name for your gateway.
- ◆ We will be using VPN as the Gateway type in this project.

- ◆ Route-based is the VPN type we will be using for this project.
- ◆ SKU will be VpnGw2AZ for this project.
- ◆ Generation2 will be the choice for this tutorial.
- ◆ Our Virtual network will be the one we created previously.
- ◆ Select the subnet in your virtual subnet with the name Gatewaysubnet
- ◆ Public IP address is set to Create new → Enter a preferred name for the Public IP.
- ◆ Enable active-active mode & Configure BGP will be disabled in this project.
- ◆ Click on Review + Create and then Create. Tags can be added based on your requirements.

Basics

Azure has provided a planning and design guide to help you configure the various VPN gateway options. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources. [View](#)

Subscription * Azure subscription 1 1

Resource group VicLab-RG (derived from virtual network's resource group)

Name * VicLabVNetGW 2

Region * East US 2

Gateway type * VPN ExpressRoute 3

SKU * VpnGw2AZ 3

Generation Generation2 3

Enable Advanced Connectivity Enabled

Review + create Previous Next : Tags > Download a template for automation

Virtual network * VicVNet 4

Subnet GatewaySubnet (172.76.0.0/27) 4

Only virtual networks in the currently selected subscription and region are listed.

Public IP address

Public IP address * Create new Use existing 5

Public IP address name * VicLabVNetGWPip 6

Public IP address SKU Standard

Assignment Dynamic Static 7

Enable active-active mode * Enabled Disabled

Configure BGP * Enabled Disabled

Authentication Information (Preview)

Enable Key Vault Access Enabled Disabled

Azure recommends using a validated VPN device with your virtual network gateway. To view a list of validated devices and

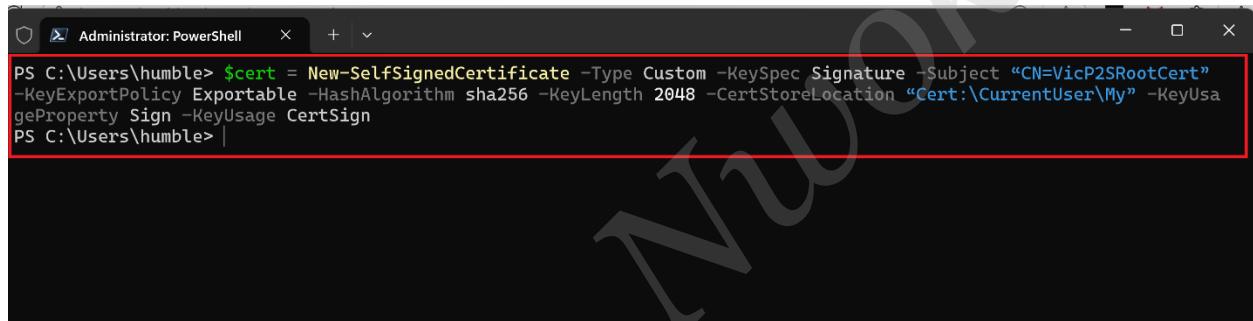
Review + create Previous Next : Tags > Download a template for automation

Step 3: Creating a Self-Signed Certificate

- Generated both **Root** and **Client** certificates using PowerShell

Copy and paste the cmdlet below into PowerShell. This cmdlet will create a self-signed root certificate named ‘VicP2SRootCert’ that is automatically installed in ‘Certificates-Current User\Personal\Certificates’. If you want to use your preferred name, modify the CN value.

```
$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature -Subject "CN=VicP2SRootCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign.
```

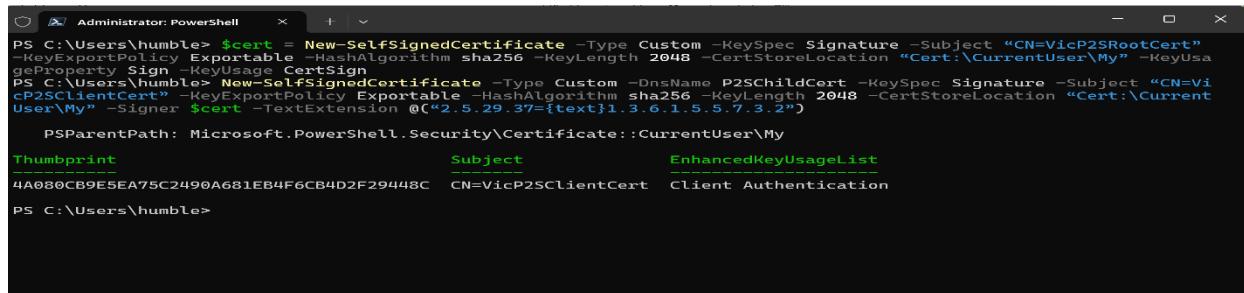


```
Administrator: PowerShell
PS C:\Users\humble> $cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature -Subject "CN=VicP2SRootCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign
PS C:\Users\humble> |
```

Note: Don't close the PowerShell console; copy the second command to generate a client certificate.

- If you haven't closed your PowerShell console after creating the self-signed root certificate. This cmdlet continues and uses the declared ‘\$cert’ variable. modify the CN value named ‘VicP2SClientCert’.

```
New-SelfSignedCertificate -Type Custom -DnsName P2SChildCert -KeySpec Signature -Subject "CN=VicP2SClientCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")
```



```
Administrator: PowerShell
PS C:\Users\humble> $cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature -Subject "CN=VicP2SRootCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign
PS C:\Users\humble> New-SelfSignedCertificate -Type Custom -DnsName P2SChildCert -KeySpec Signature -Subject "CN=VicP2SClientCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My
Thumbprint          Subject                                EnhancedKeyUsageList
4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C  CN=VicP2SClientCert  Client Authentication
```

If you mistakenly closed the PowerShell console after running the first? There is a way to handle that. **Please skip Step A, Step B, and Step C if you didn't close your console before running the second cmdlet.**

Step A: Run this cmdlet to get a list of certificates that are installed on your computer.

Get-ChildItem -Path "Cert:\CurrentUser\My"

Step B: Using the thumbprint of the root certificate that was returned after running the previous cmdlet, run this next cmdlet.

\$cert = Get-ChildItem -Path

"Cert:\CurrentUser\My\4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C"

Note: 4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C is the thumbprint from my own root certificate.

```

PowerShell 7.5.2
PS C:\Users\humble> Get-ChildItem -Path "Cert:\CurrentUser\My"
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My

Thumbprint          Subject                           EnhancedKeyUsageList
-----          ...
EDC8D807E0A80743AF09645F282A0A948173D8FF CN=f0a7f53f-66a5-4b... Client Authentication
E9CF7F0C7C364E626C79F561B18D637EAA461AA9 CN=d35e451-3bb2-46...
D977B508D71A314A19919AD43D1DF01F5ABDD1EB CN=VicP2SClientCert Client Authentication
4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C CN=VicP2SClientCert Client Authentication
2E859F06BE4B2939B121ECB53B0600BCD61F3343 CN=VicP2SRootCert Client Authentication
1BDB124D2EF340605577A724CCCF2F659D6A6583 CN=VicP2SRootCert Client Authentication
1657F1C5D57288FC09E2A1135FD5ACBE596FF3D2 CN=fe6ee67e-825c-4c... Client Authentication

PS C:\Users\humble>

```

Step C: You can now run this cmdlet to generate a client certificate. Modify the CN to give it your preferred name.

```

New-SelfSignedCertificate -Type Custom -DnsName P2SChildCert -KeySpec Signature -
Subject "CN=VicP2SClientCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -
KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -Signer $cert -TextExtension
@("2.5.29.37={text}1.3.6.1.5.5.7.3.2")

```

```

PowerShell 7.5.2
PS C:\Users\humble> Get-ChildItem -Path "Cert:\CurrentUser\My"
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My

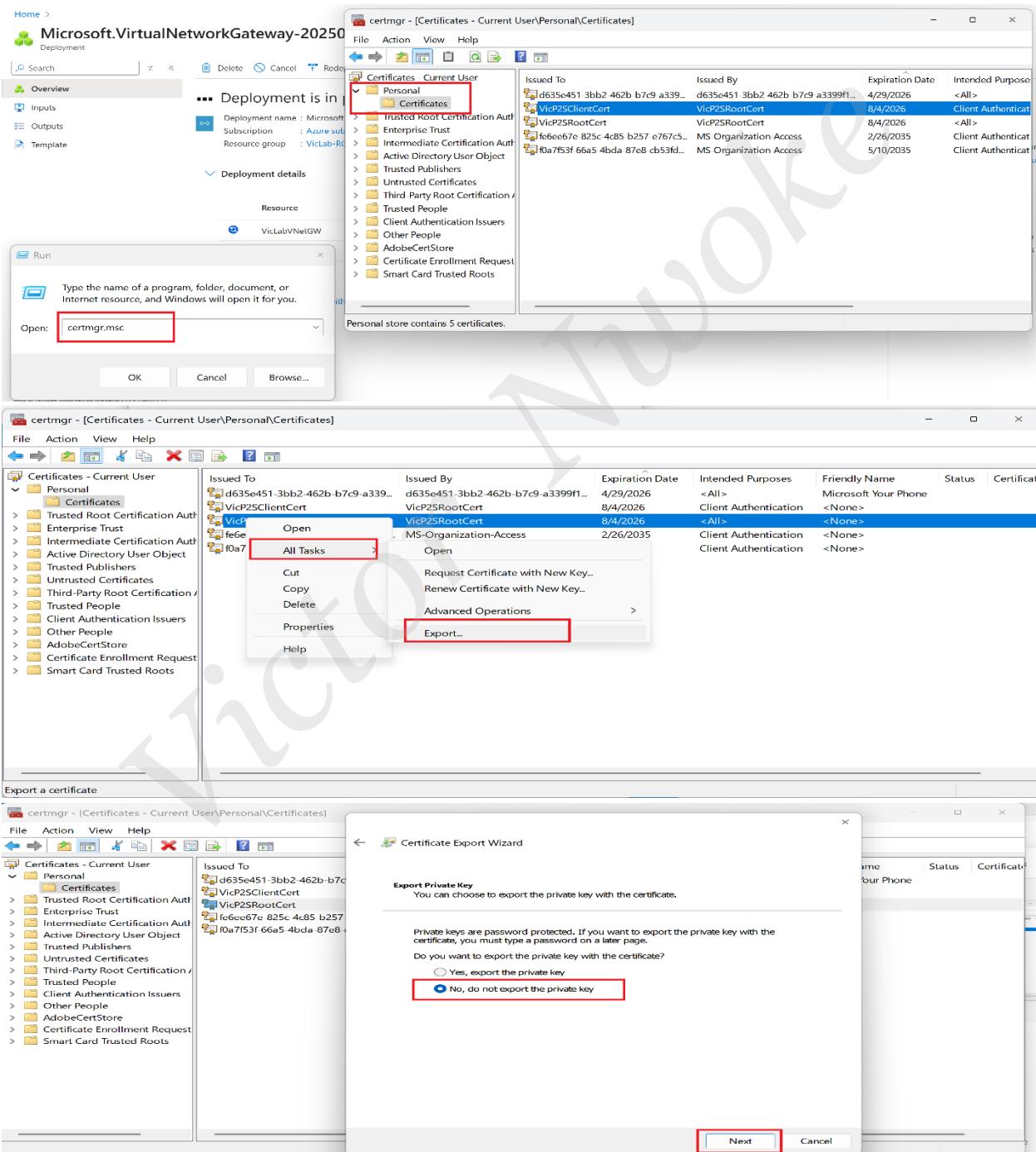
Thumbprint          Subject                           EnhancedKeyUsageList
-----          ...
EDC8D807E0A80743AF09645F282A0A948173D8FF CN=f0a7f53f-66a5-4b... Client Authentication
E9CF7F0C7C364E626C79F561B18D637EAA461AA9 CN=d35e451-3bb2-46...
D977B508D71A314A19919AD43D1DF01F5ABDD1EB CN=VicP2SClientCert Client Authentication
4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C CN=VicP2SClientCert Client Authentication
2E859F06BE4B2939B121ECB53B0600BCD61F3343 CN=VicP2SRootCert Client Authentication
1BDB124D2EF340605577A724CCCF2F659D6A6583 CN=VicP2SRootCert Client Authentication
1657F1C5D57288FC09E2A1135FD5ACBE596FF3D2 CN=fe6ee67e-825c-4c... Client Authentication

PS C:\Users\humble> $cert = Get-ChildItem -Path "Cert:\CurrentUser\My\4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C"
PS C:\Users\humble>

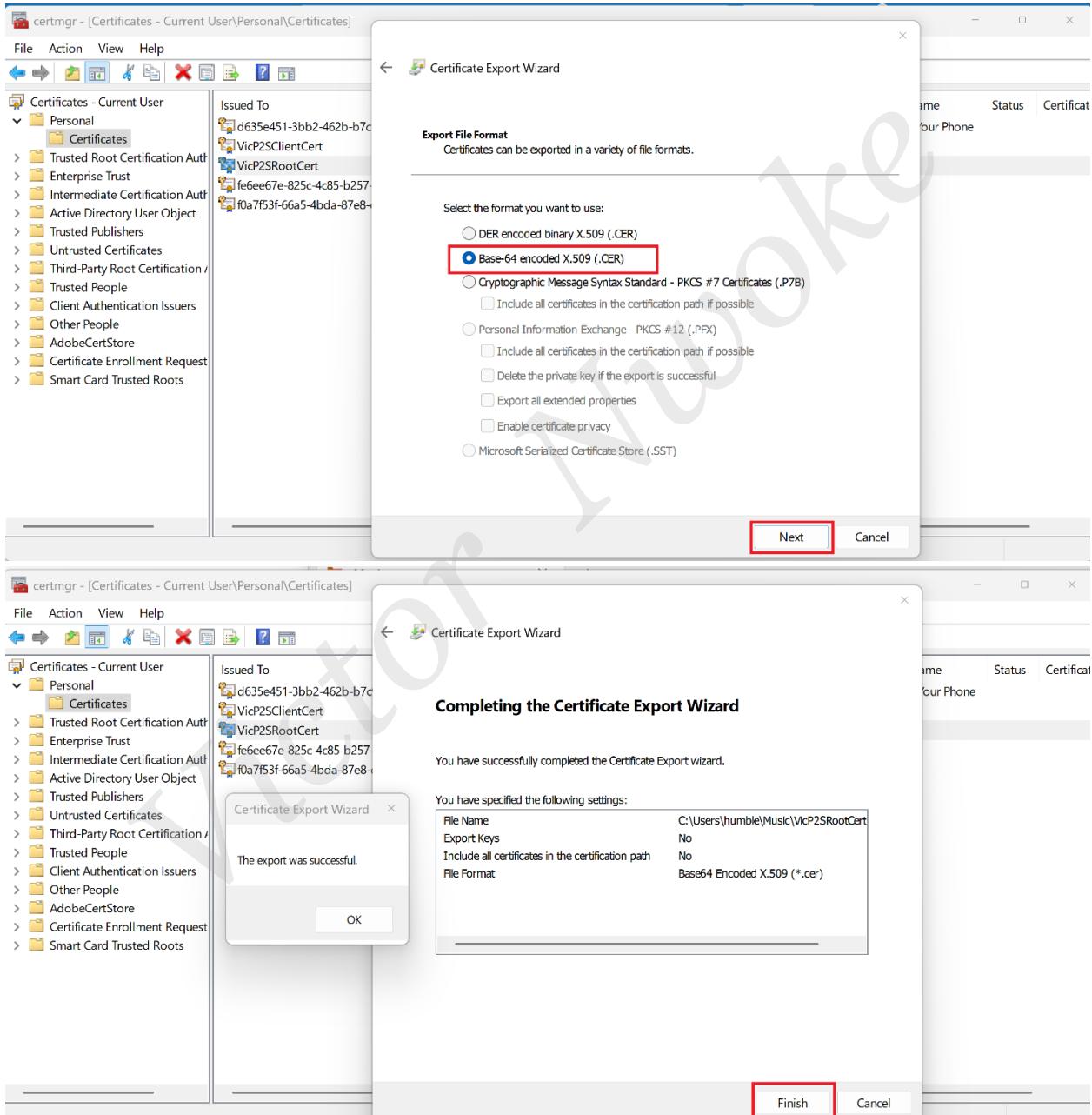
```

Step 4: View and Export Root Certificate

- ◆ Press **Win + R** key and type certmgr.msc
- ◆ Right-click on the **root certificate**, click on **All Tasks**, then click on **Export**
- ◆ Click on **Next** on the Certificate Export Wizard, and select **No**, do not export the private key, then click on **Next**.

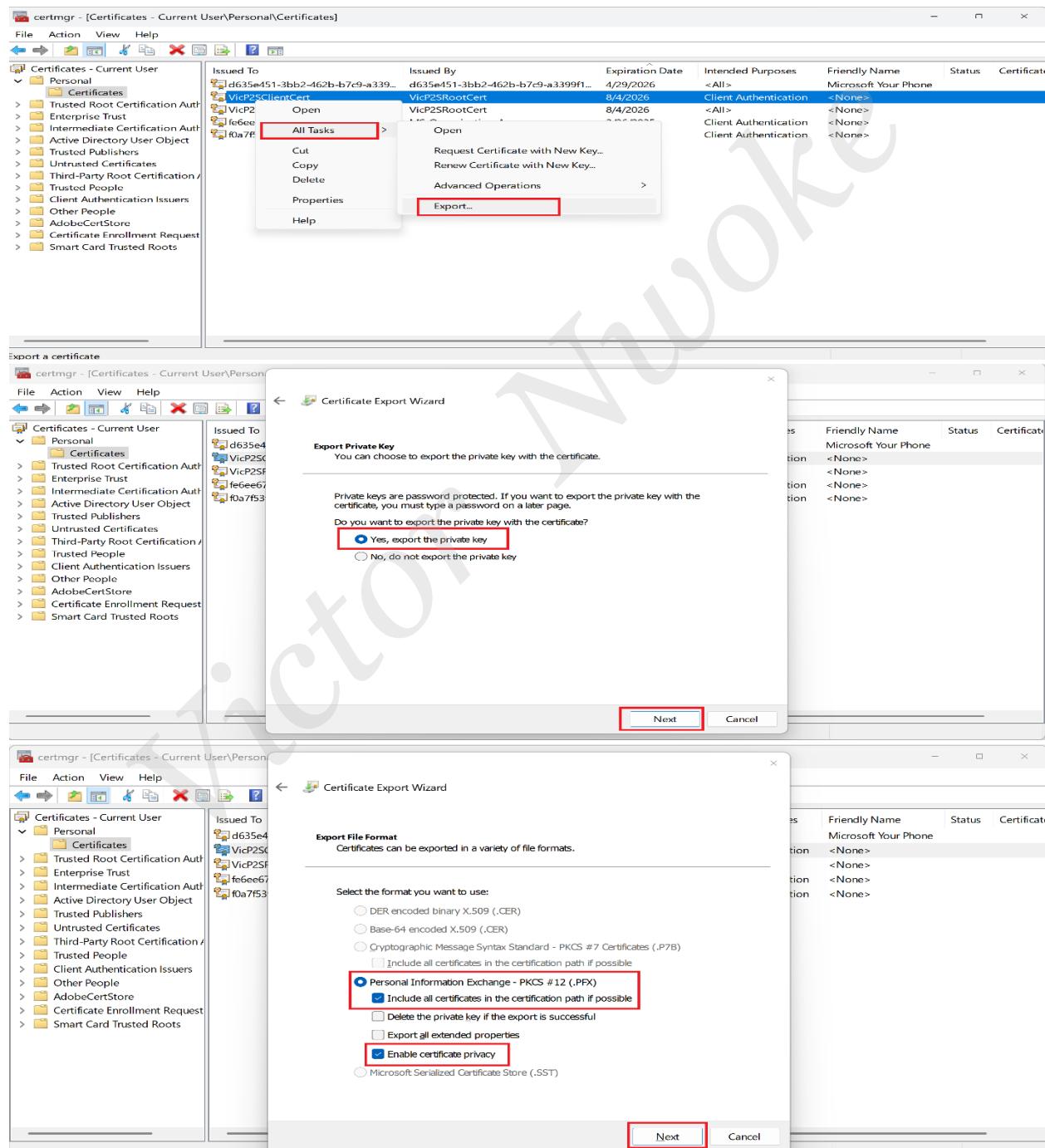


- On the Export File Format page, select the **Base-64 encoded X.509 (.CER)**, and then click **Next**.
- Browse the location you want to save it to and then give it a preferred name.
- Then, click **Next**, and click **Finish** to export the certificate to your preferred location.
- You'll see a confirmation pop-up saying, "**The export was successful.**"



Step 5: View and Export Client Certificate

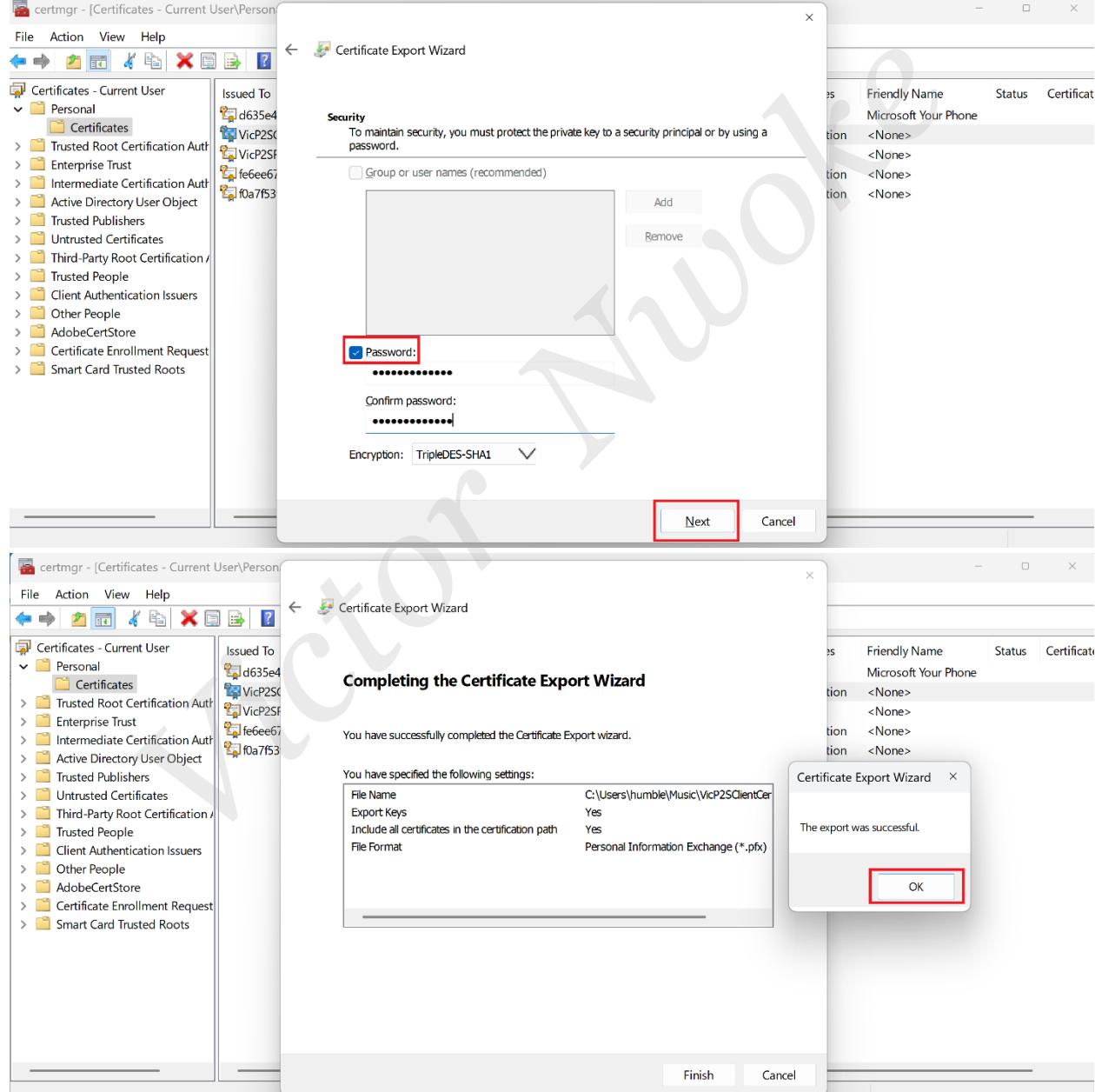
- Right-click on the **client certificate** and click on All Tasks, then click on Export
- On the Certificate Export Wizard, click **Next** to continue.
- Select Yes, export the private key, and then click Next.
- On the Export File Format page, leave the defaults selected.



- ◆ Tick the Password check box on the Security page and type in your password.

Remember, you will need this password if you are trying to connect with this certificate from another client.

- ◆ Browse to the location you want to save it and give it your preferred name. Once that is done, click Next and then Finish.
- ◆ You'll see a confirmation pop-up saying, "**The export was successful.**"

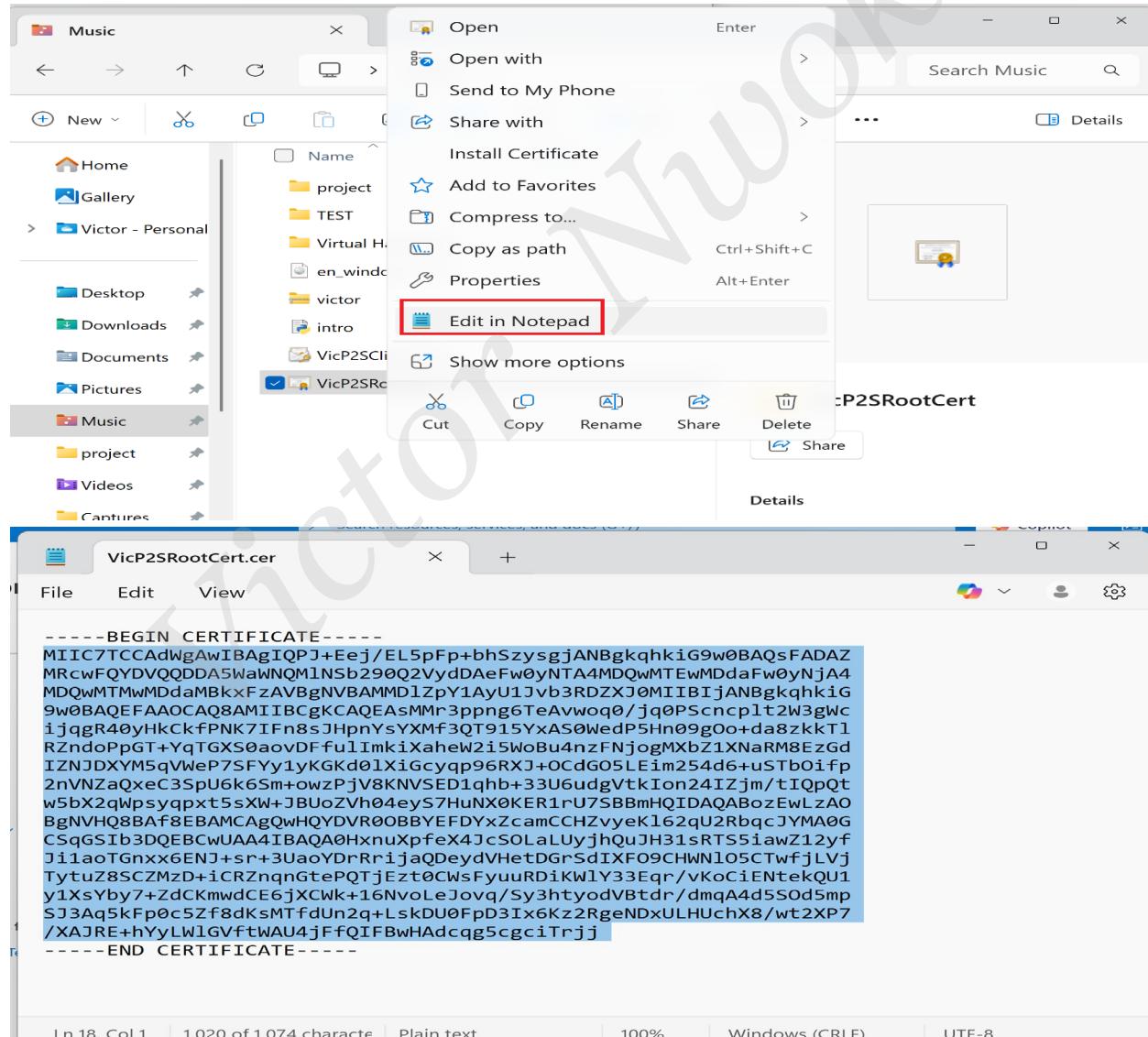


We have successfully exported our root and client certificates.

To configure Point-to-site on the virtual network gateway we created, we will need our root certificate.

Configure Point-to-Site VPN Settings in Azure

- ◆ Browse to the location where the root certificate was exported to and open it with a text editor such as Notepad
- ◆ Log in to the Azure portal and open the Virtual network gateway we created earlier.
- ◆ On the Virtual network gateway page, click on the **Point-to-site configuration** under Settings on the left side menu and then click on Configure now on the right.



Step-by-Step Point-to-Site VPN Configuration

The screenshot shows the Microsoft Azure portal interface for a Virtual Network Gateway named 'VicLabVNetGW'. In the left sidebar, under the 'Connections' section, the 'Point-to-site configuration' link is highlighted with a red box and labeled '1'. At the top right of the main content area, there is a button labeled 'Configure now' which is also highlighted with a red box and labeled '2'.

- ◆ Add the Address pool you want the client computers to get their IP address from when they connect. For this project, the address pool I will be using is **192.168.20.0/24**.
- ◆ The tunnel type will be **IKEv2 and SSTP (SSL)**, and the Authentication type should be **Azure certificate**.
- ◆ Under the **Root certificates**, select a preferred name for this project. I will be using **VicP2SRootCert**, and in the public certificate data, paste the text that you copied from the root certificate on our client computer, and click on Save at the top.

The screenshot shows the 'Point-to-site configuration' blade for 'VicLabVNetGW'. Several fields are highlighted with red boxes and numbered arrows:

- 1: 'Address pool' input field containing '192.168.76.0/24'.
- 2: 'Tunnel type' dropdown menu showing 'IKEv2 and SSTP (SSL)'.
- 3: 'Authentication type' dropdown menu showing 'Azure certificate'.
- 4: 'Name' input field for 'Root certificates' containing 'VicP2SRootCert'.
- 5: 'Public certificate data' input field containing a long certificate string.

 The 'Save' button at the top is also highlighted with a red box and labeled '6'.

Download the VPN client by clicking on Download VPN client at the top.

The screenshot shows the Microsoft Azure portal interface. The left sidebar has a tree view with 'Point-to-site configuration' selected. The main area shows 'VicLabVNetGW | Point-to-site configuration'. It includes fields for 'Address pool' (192.168.76.0/24), 'Tunnel type' (IKEv2 and SSTP (SSL)), 'IPsec / IKE policy' (Default), and 'Authentication type' (Azure certificate). At the top right, there's a 'Download VPN client' button, which is highlighted with a red box.

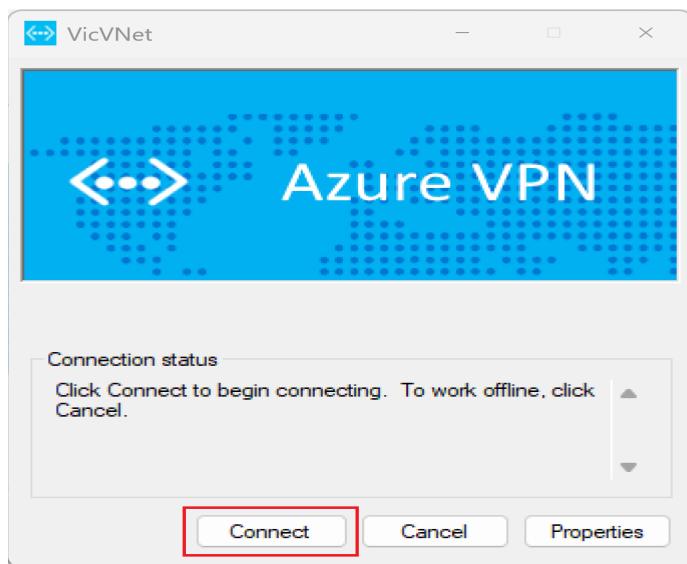
Copy the downloaded VPN client to the client computer and install it. For this tutorial, I will be installing WindowsAmd64.

This screenshot shows a Windows File Explorer window with the path 'This PC > Downloads > WindowsAmd64'. Inside the 'WindowsAmd64' folder, there is a file named 'VpnClientSetupAmd64'. A red box highlights the file name.

After installing the VPN client, go to Network and Internet in settings, then click on VPN, you should see the Virtual Network we created on Azure, click on it, then click on Connect.

This screenshot shows the Windows Settings app. The left sidebar has 'Network & internet' selected. The main area shows 'Network & internet > VPN'. It lists a 'VPN connections' section with a connection named 'VicVNet' (Status: Not connected) and a 'Connect' button. Below it is an 'Advanced settings for all VPN connections' section with two toggle switches: 'Allow VPN over metered networks' (off) and 'Allow VPN while roaming' (off). A red box highlights both the 'Network & internet' option in the sidebar and the 'Connect' button.

Once you click on the connect button the Azure VPN will pop up, then click on connect again.



At this point, you have successfully configured your Point-to-Site VPN.

A composite screenshot showing the Windows Settings app and a Command Prompt window. The Settings app shows the 'Network & internet > VPN' section with a red box around the 'VicVNet' connection entry. The Command Prompt window shows the output of 'ipconfig' with a red box around the 'IPv4 Address' for the 'PPP adapter VicVNet'.

Bonus: I create a Virtual Machine on my VNet, and I will connect to the Virtual Machine using the private IP of the Virtual Machine from my client computer.

I went ahead to use Remote Desktop Connection to connect with Virtual Machines using the Private IP: 172.76.1.4.

Azure Network Settings Screenshot:

- Network interface: vicvm501
- Virtual network / subnet: VicVNet / subnet
- Public IP address: - (Configure)
- Private IP address: 172.76.1.4 (highlighted with a red box)
- Admin security rules: 0 (Configure)

Windows Network & internet Settings Screenshot:

- VPN connections: VicVNet (Connected)
- Bytes sent: 503,427
- Bytes received: 3,503,667
- Duration: 00:27:03

Command Prompt Output:

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.20348.3932]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vicadmin>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . : dfbbbrzq3sw4uxai5f4uht0cxpb.bx.internal.cloudapp.net
Link-local IPv6 Address . . . . . : fe80::6cdd:ef87:6465:1c9d%6
IPv4 Address. . . . . : 172.76.1.4 (highlighted with a red box)
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 172.76.1.1

C:\Users\vicadmin>
```

Thank you for your time!