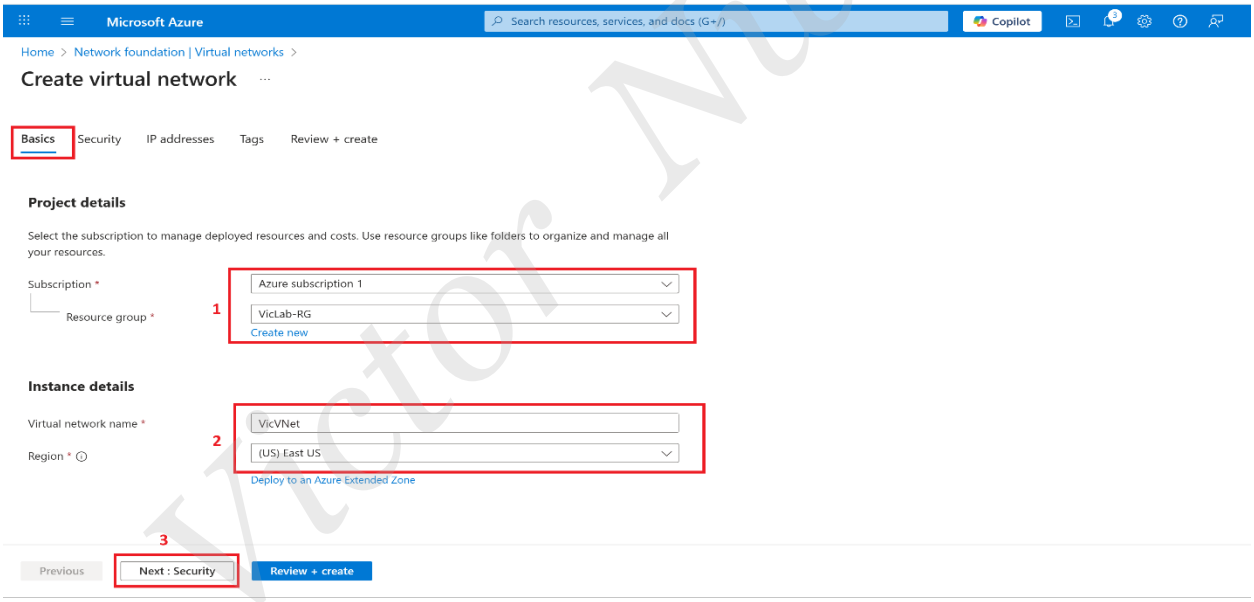
Azure Networking Skills in Action: Step-by-Step Point-to-Site VPN Configuration

Step 1: Creating a Virtual Network

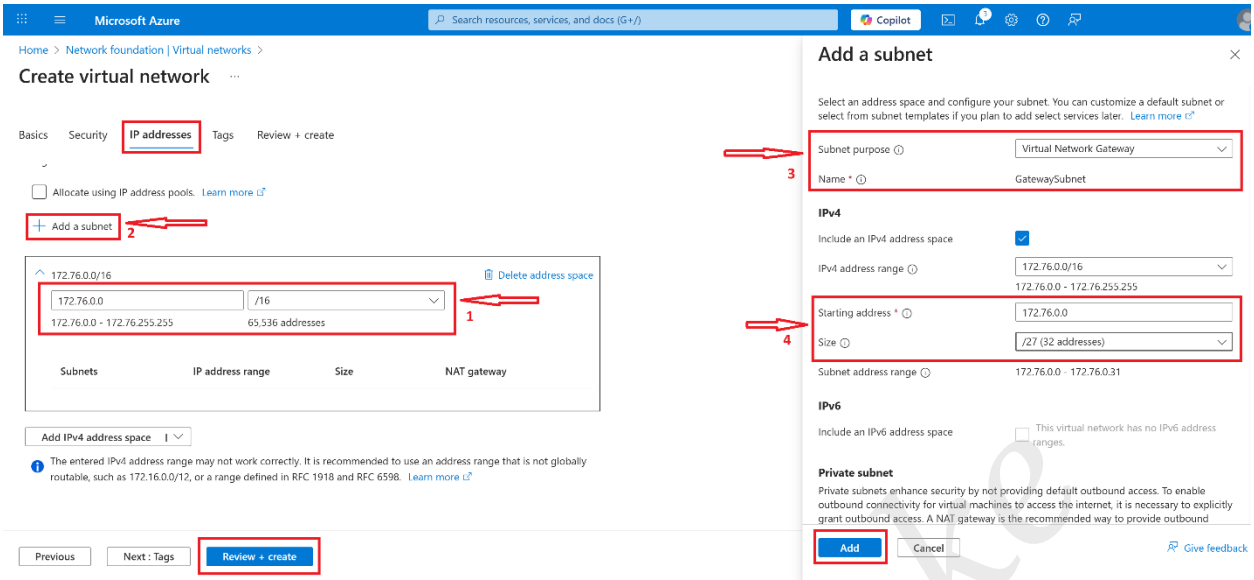
- Sign in to the **Azure Portal** → Search for **Virtual Network** in the Azure portal. → In the search results, select the virtual network right under **Marketplace**
- On the Virtual network page, click on **Create**
- Under the Basics tab, choose the Subscription you want to use for this project → 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.

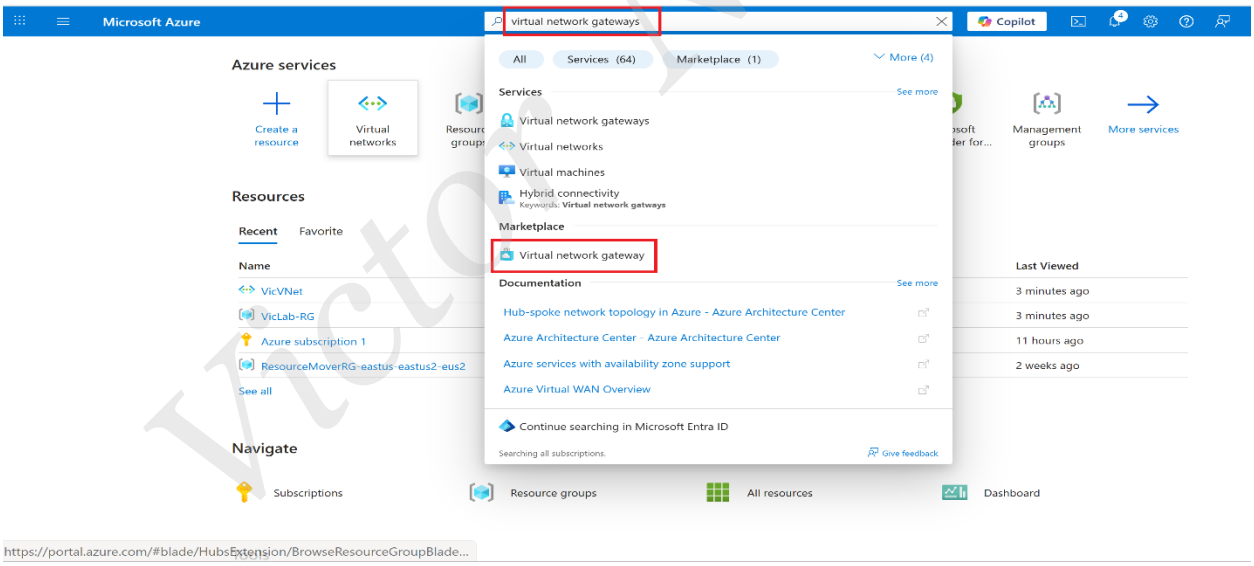
- On the IP addresses tab, click on the Add an IP address space and select the Address space type radio button for IPv4.

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.



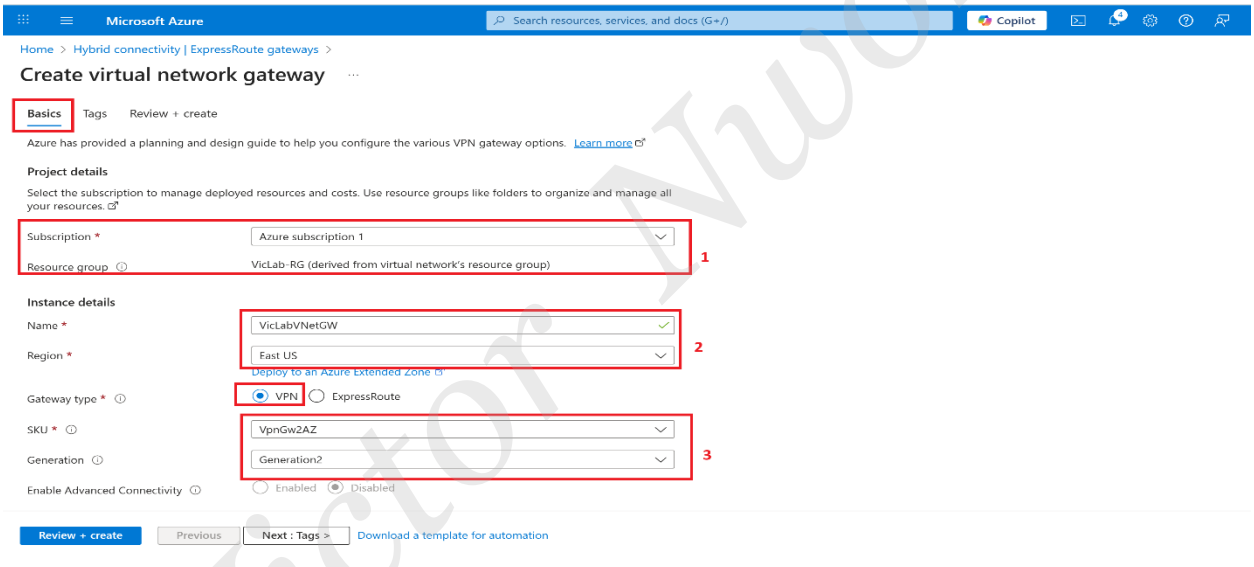
Step 2: Let's deploy the VPN Gateway

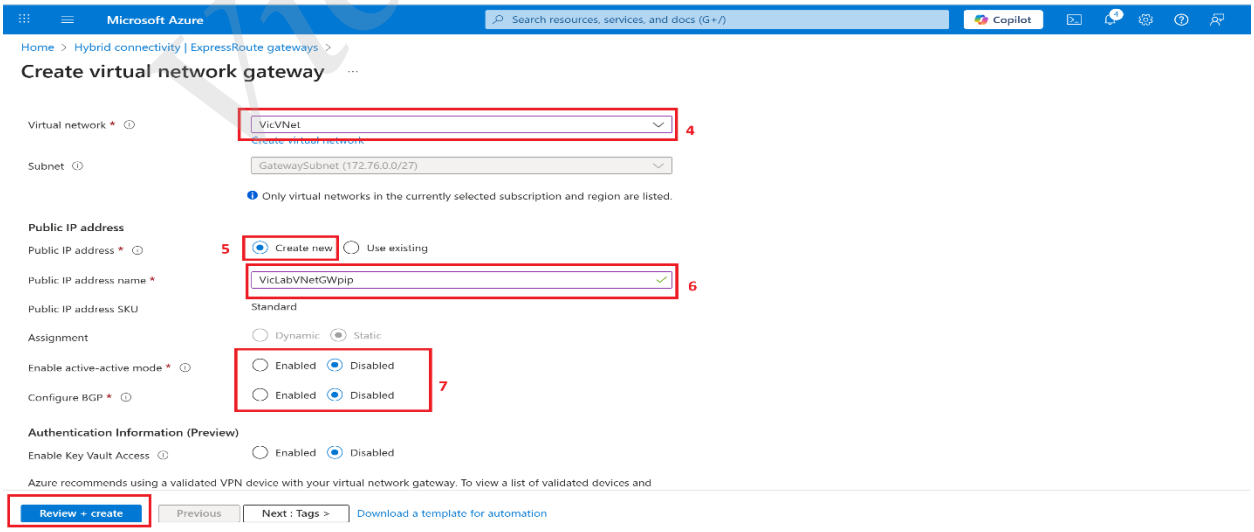
- Search for Virtual Network Gateway in the Azure portal.
- In the Search results, under Marketplace, select Virtual network gateway.



- 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.



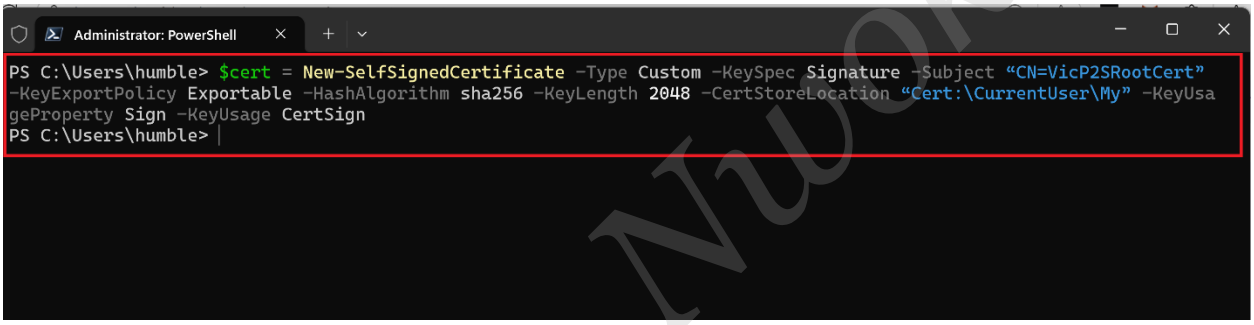


Step 3: Creating a Self-Signed Certificate

- 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.

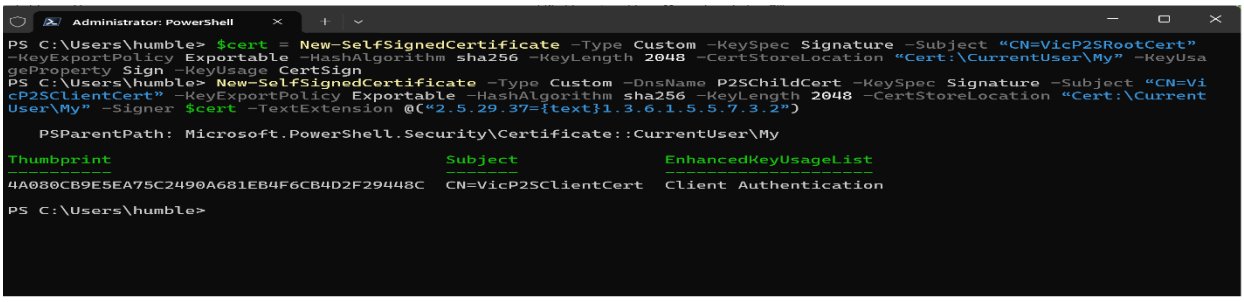
```
$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature -Subject "CN=VicP2SRootCert" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign.
```



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")
```



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.

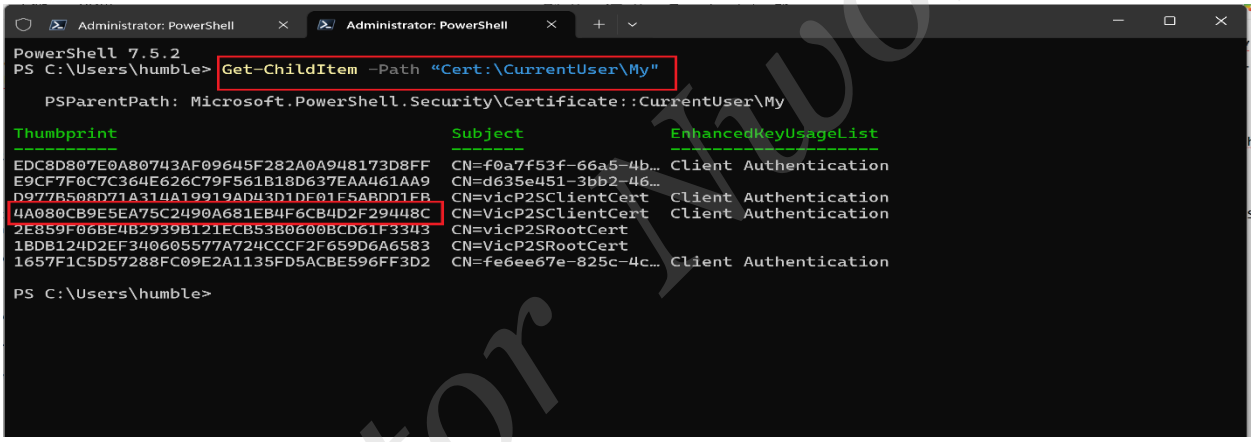
Step A: Run this cmdlet to get a list of certificates that are installed on your computer.

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.

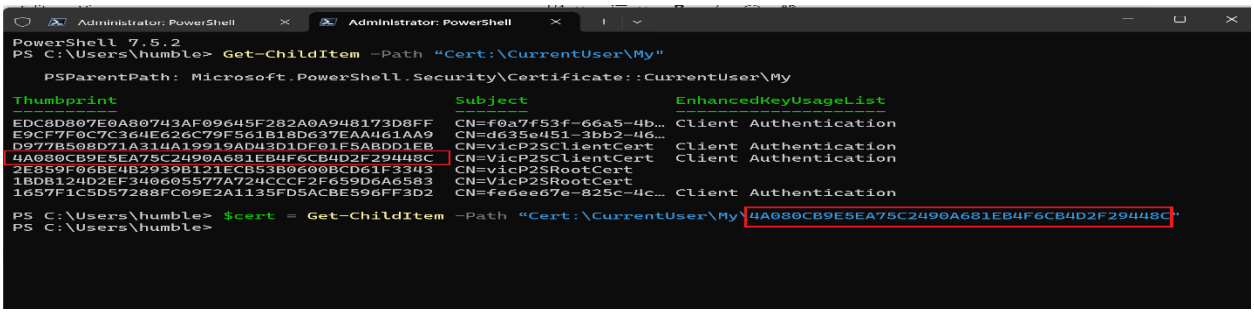
$cert = Get-ChildItem -Path "Cert:\CurrentUser\My\4A080CB9E5EA75C2490A681EB4F6CB4D2F29448C"

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



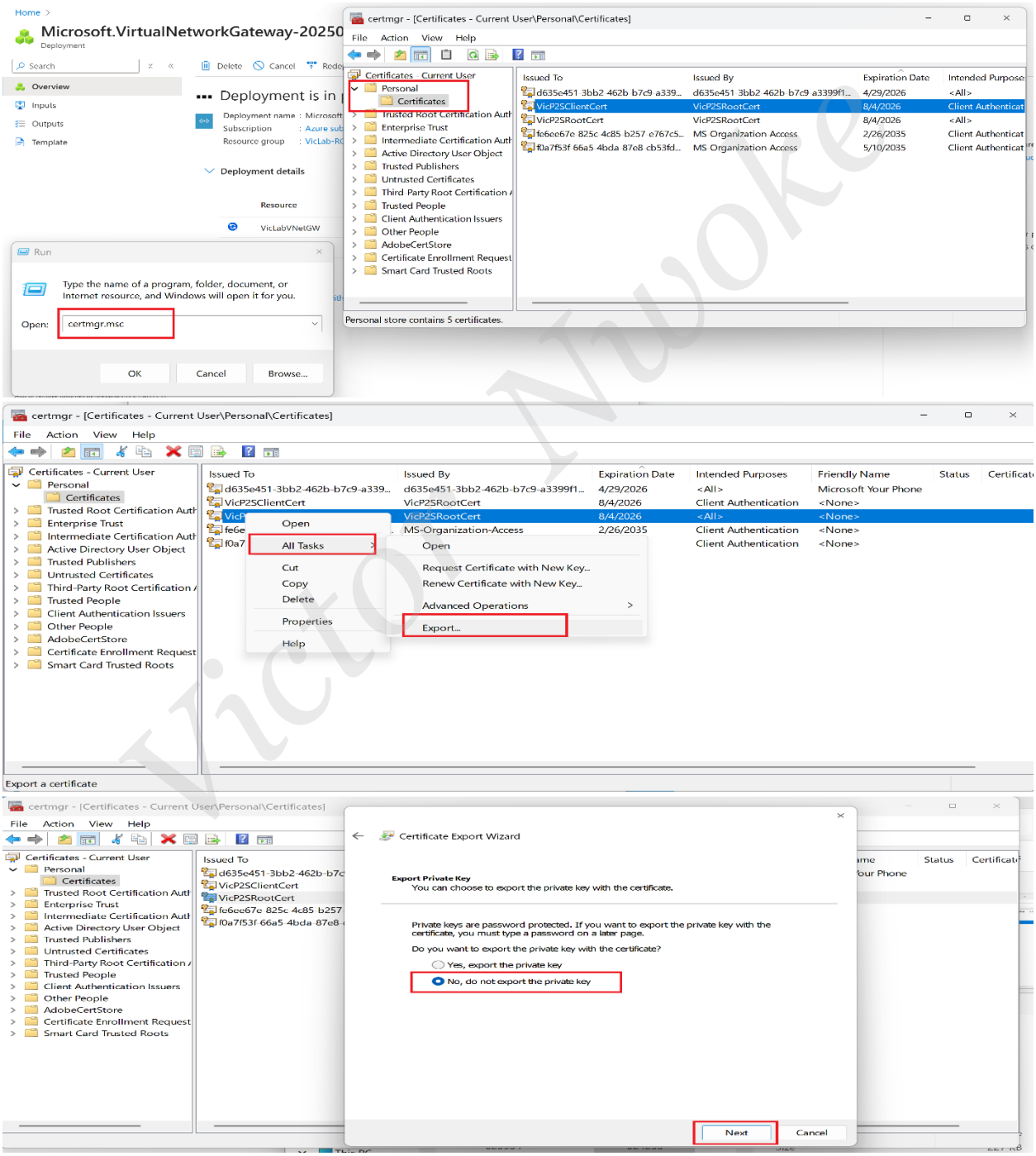
Step C: You can now run this cmdlet to generate a client certificate. Modify the CN to give it your preferred name.

```
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")
```

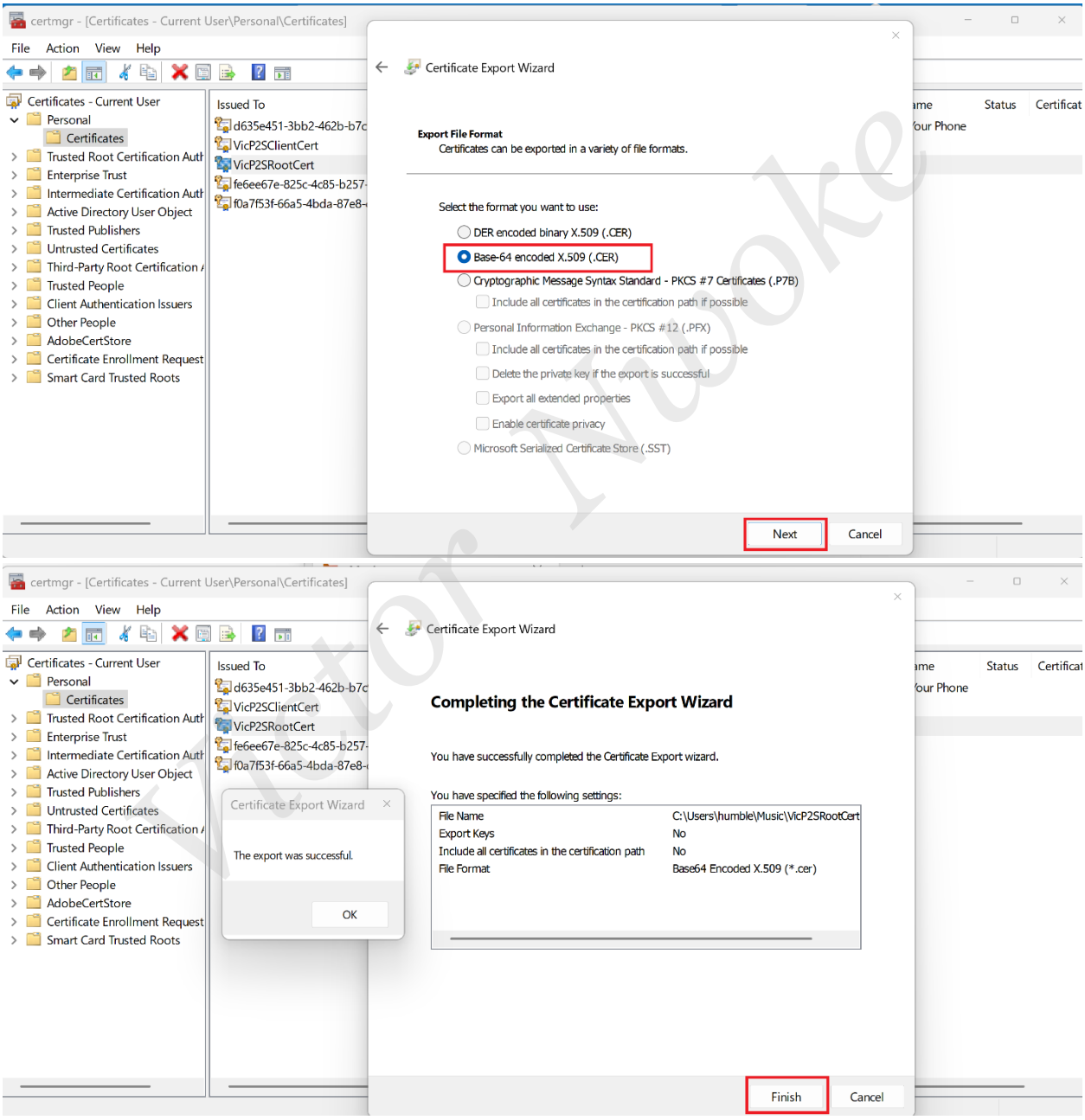


Step 4: View and Export Root Certificate

- 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.**

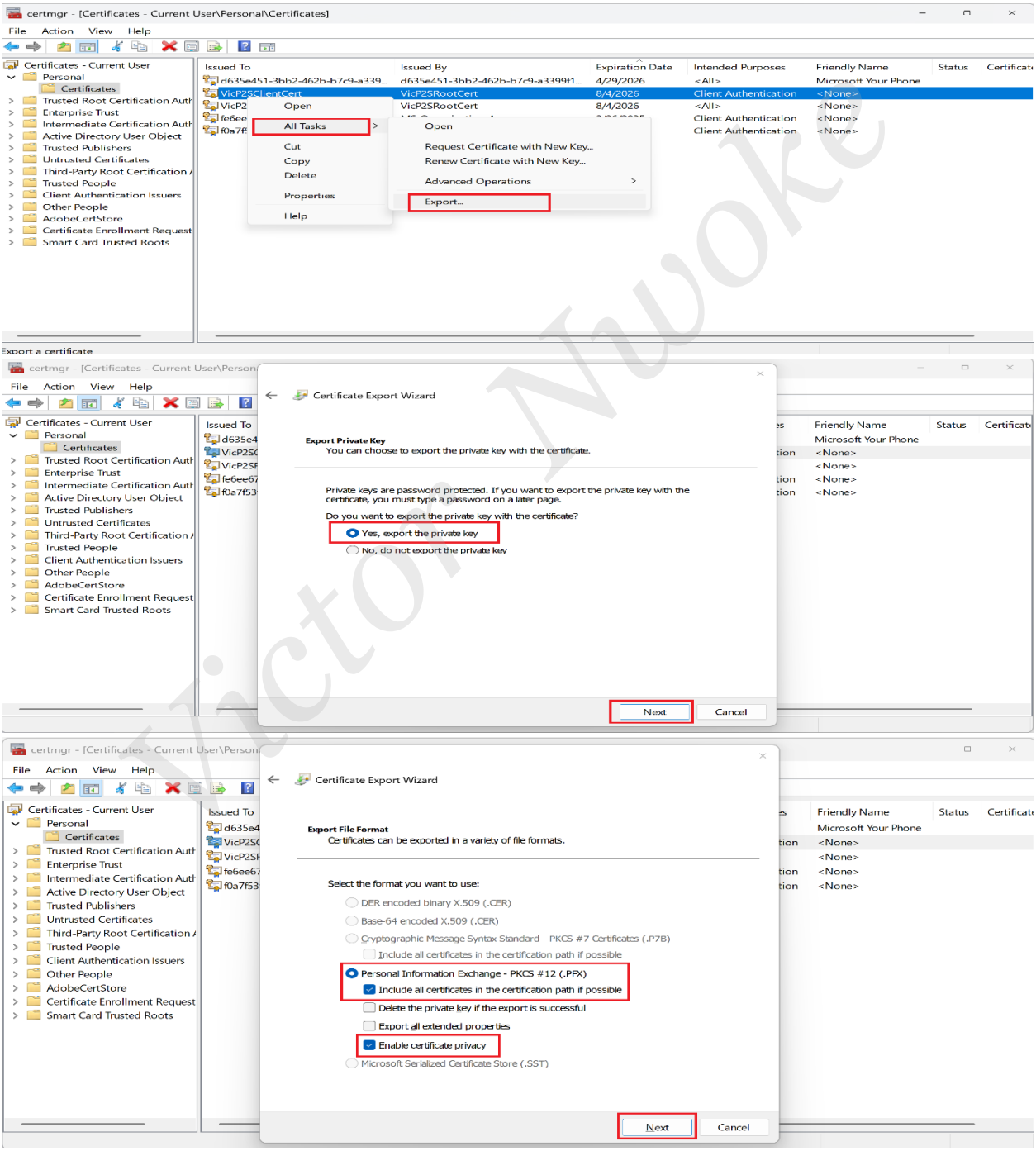


- On the Export File Format page, select the **Base-64 encoded X.509 (.CER)**, and then click **Next.**
- Browse the location you want to save it to and then give it a preferred name.
- Then, click **Next**, and click **Finish** to export the certificate to your preferred location.
- You'll see a confirmation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

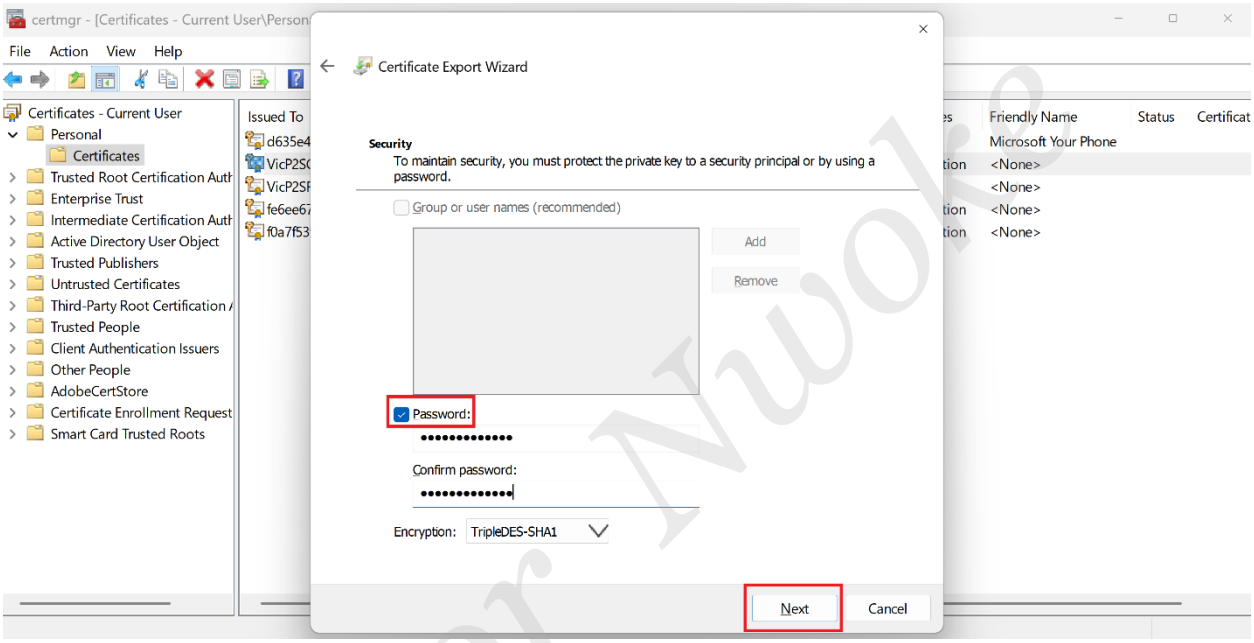


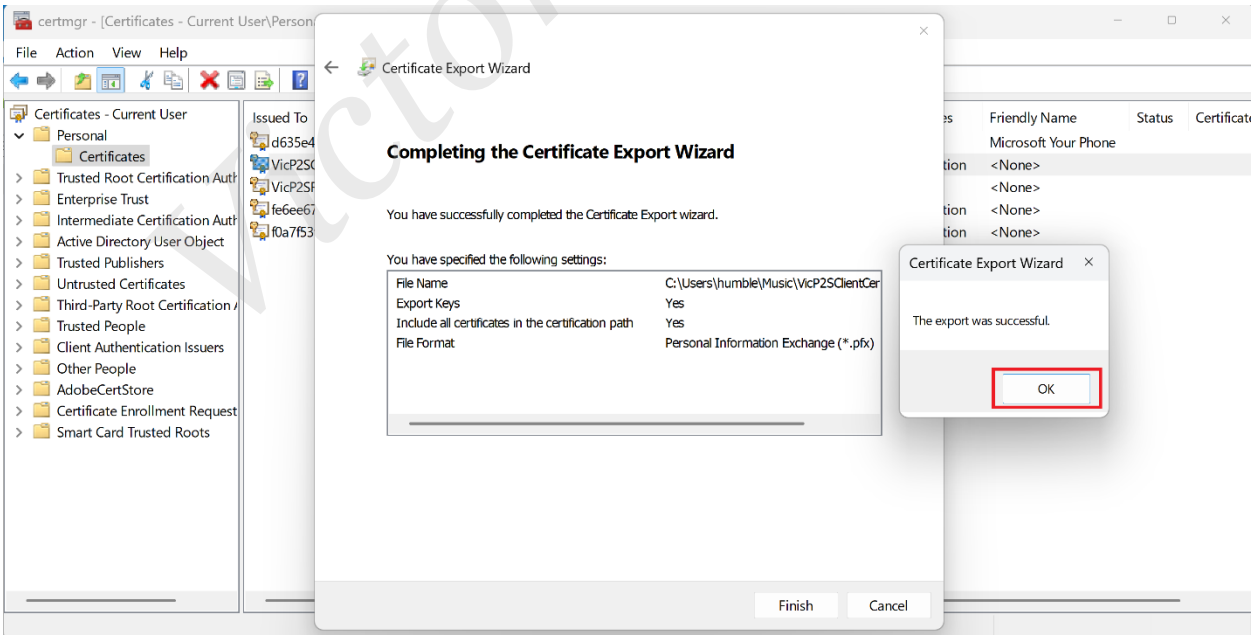
- Tick the Password check box on the Security page and type in your password.

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."**



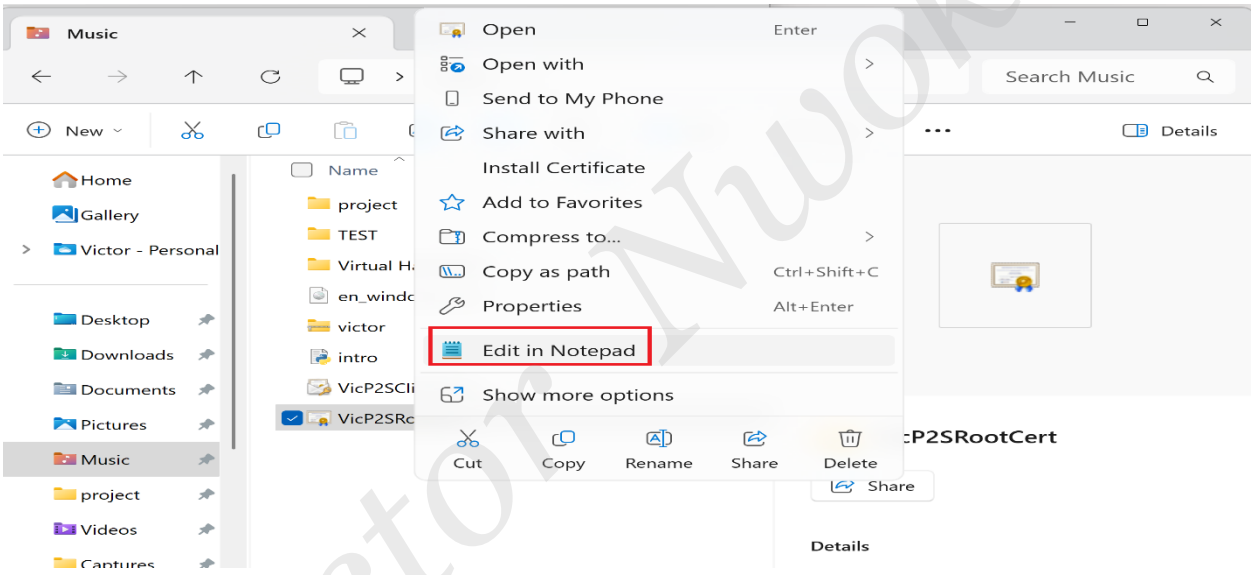


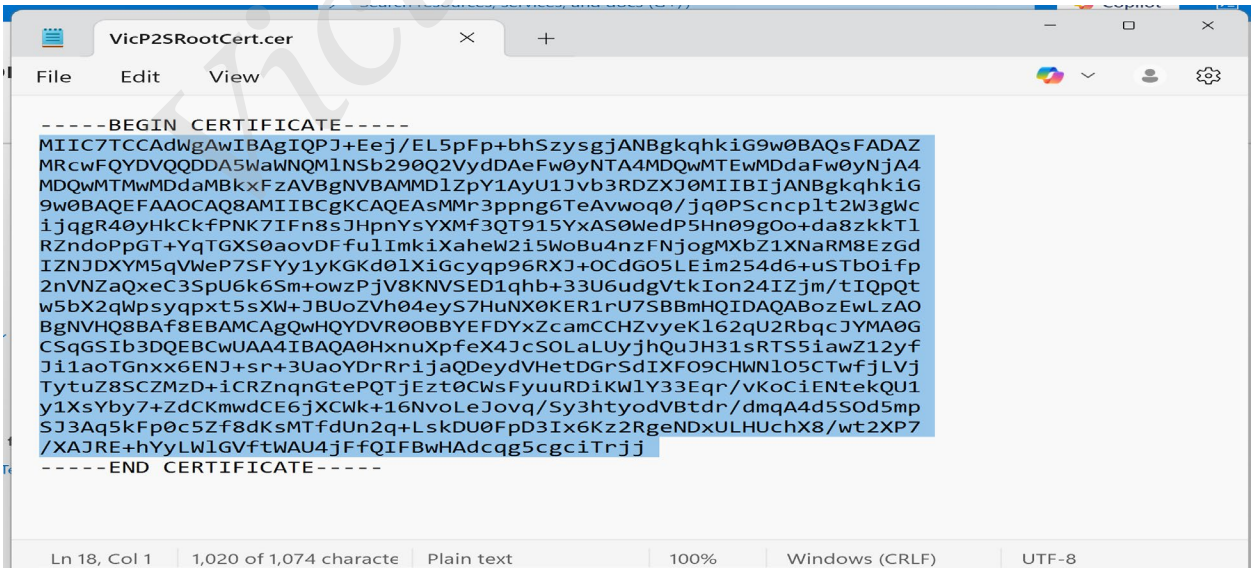
We have successfully exported our root and client certificates.

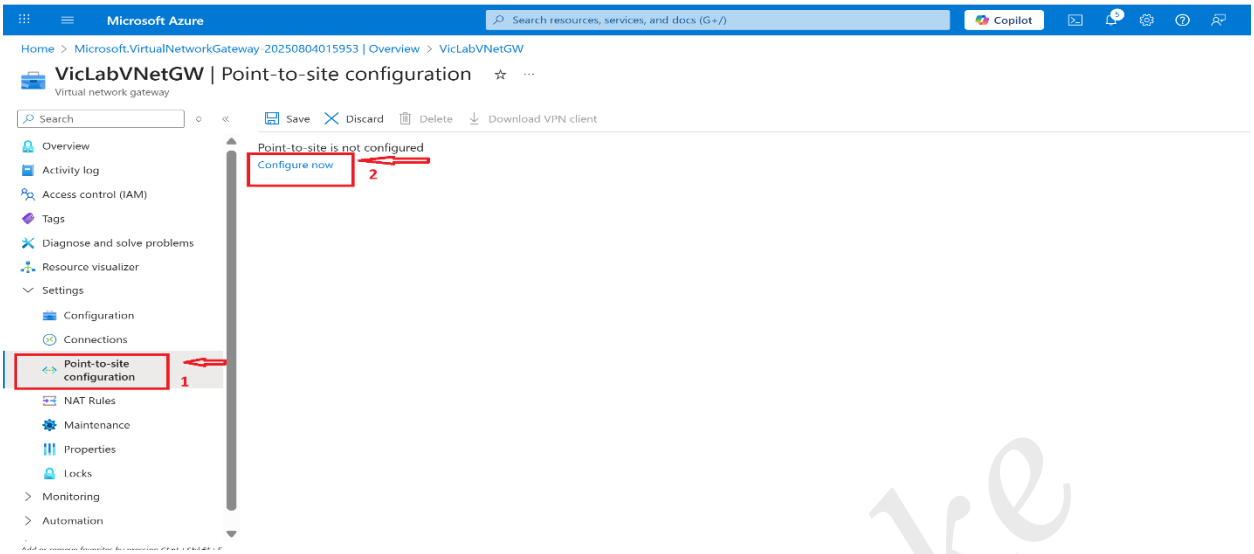
To configure Point-to-site on the virtual network gateway we created, we will need our root certificate.

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.



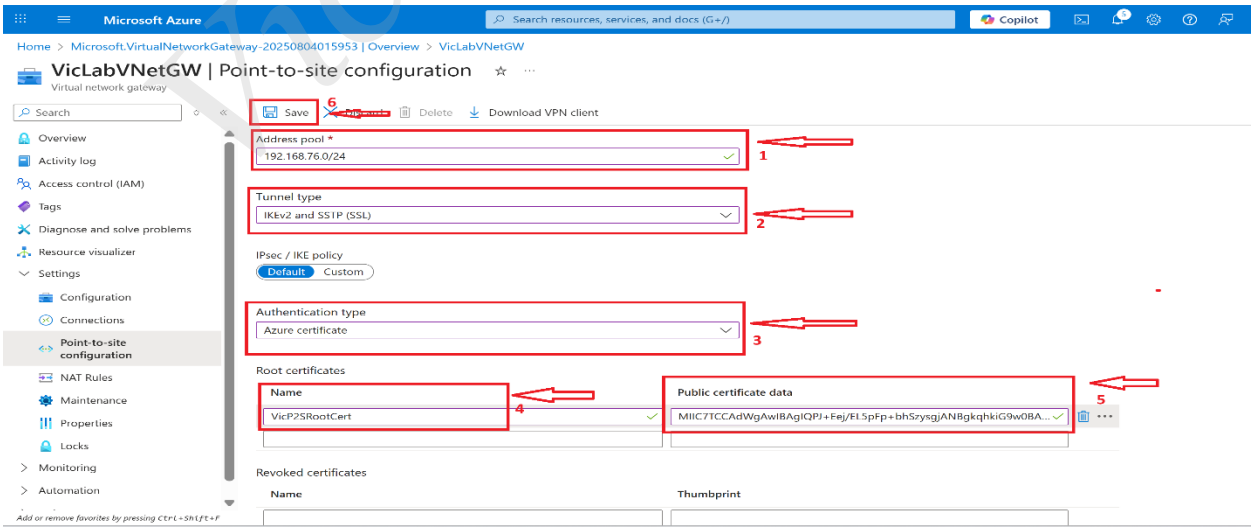




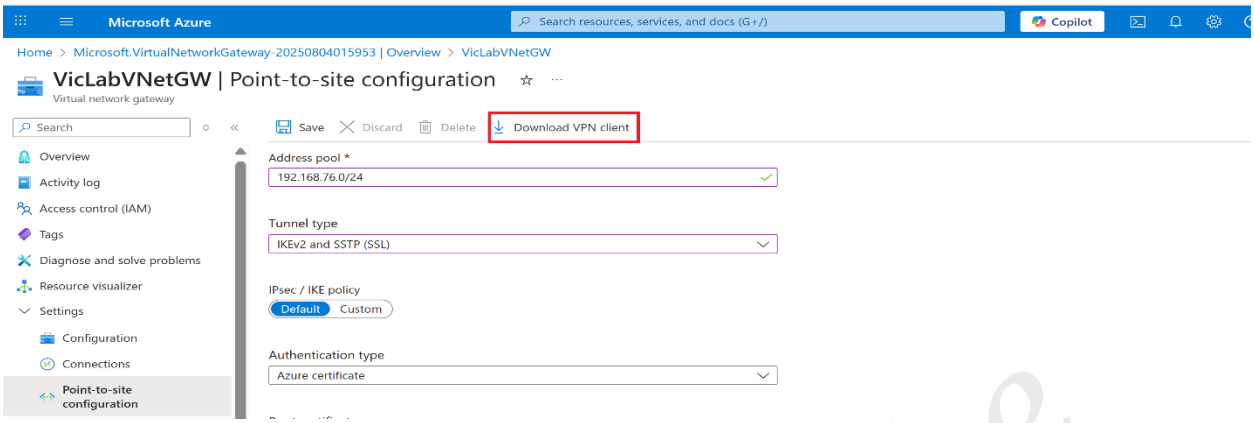
◆ Add the Address pool you want the client computers to get their IP address from when they connect. For this project, the address pool I will be using is **192.168.20.0/24**.

◆ The tunnel type will be I**KEv2 and SSTP (SSL)**, and the Authentication type should be **Azure certificate.**

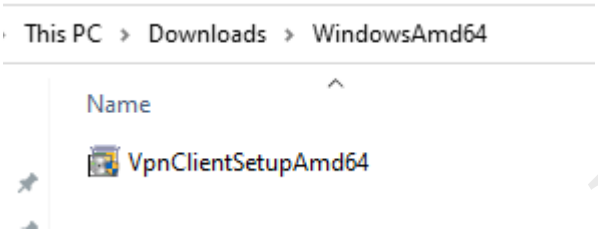
◆ Under the **Root certificates**, select a preferred name for this project. I will be using VicP2SRootCert, and in the public certificate data, paste the text that you copied from the root certificate on our client computer, and click on Save at the top.



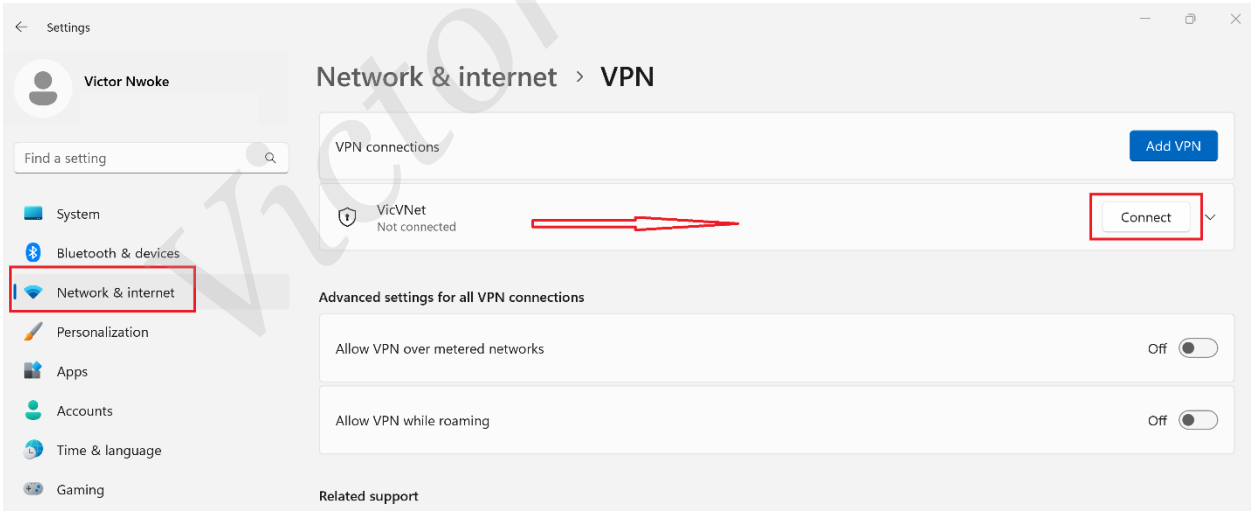
Download the VPN client by clicking on Download VPN client at the top.



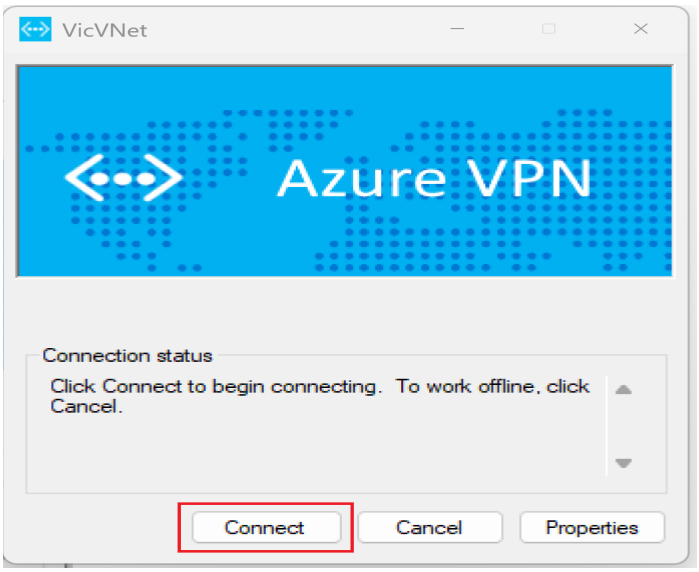
Copy the downloaded VPN client to the client computer and install it. For this tutorial, I will be installing WindowsAmd64.



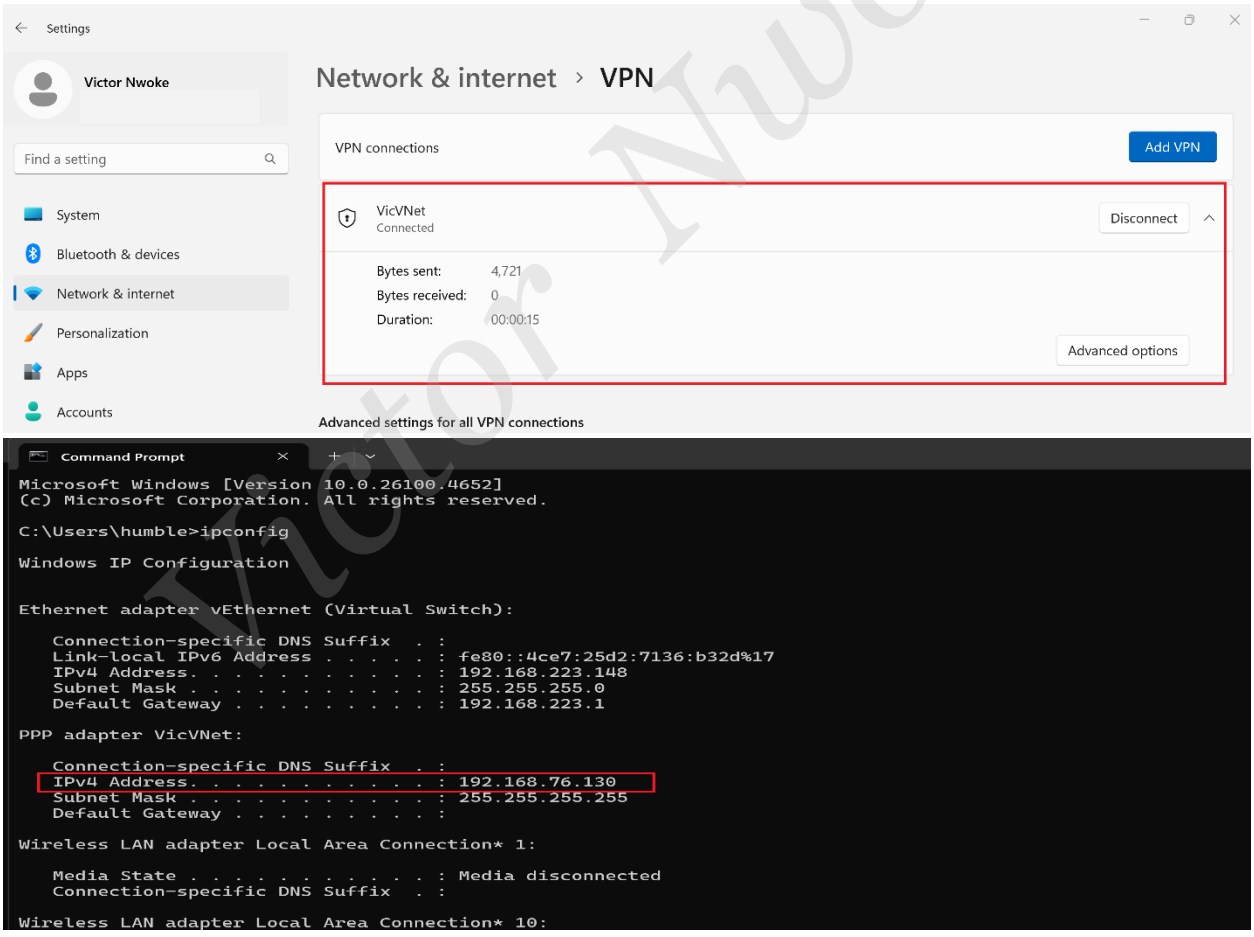
After installing the VPN client, go to Network and Internet in settings, then click on VPN, you should see the Virtual Network we created on Azure, click on it, then click on Connect.



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

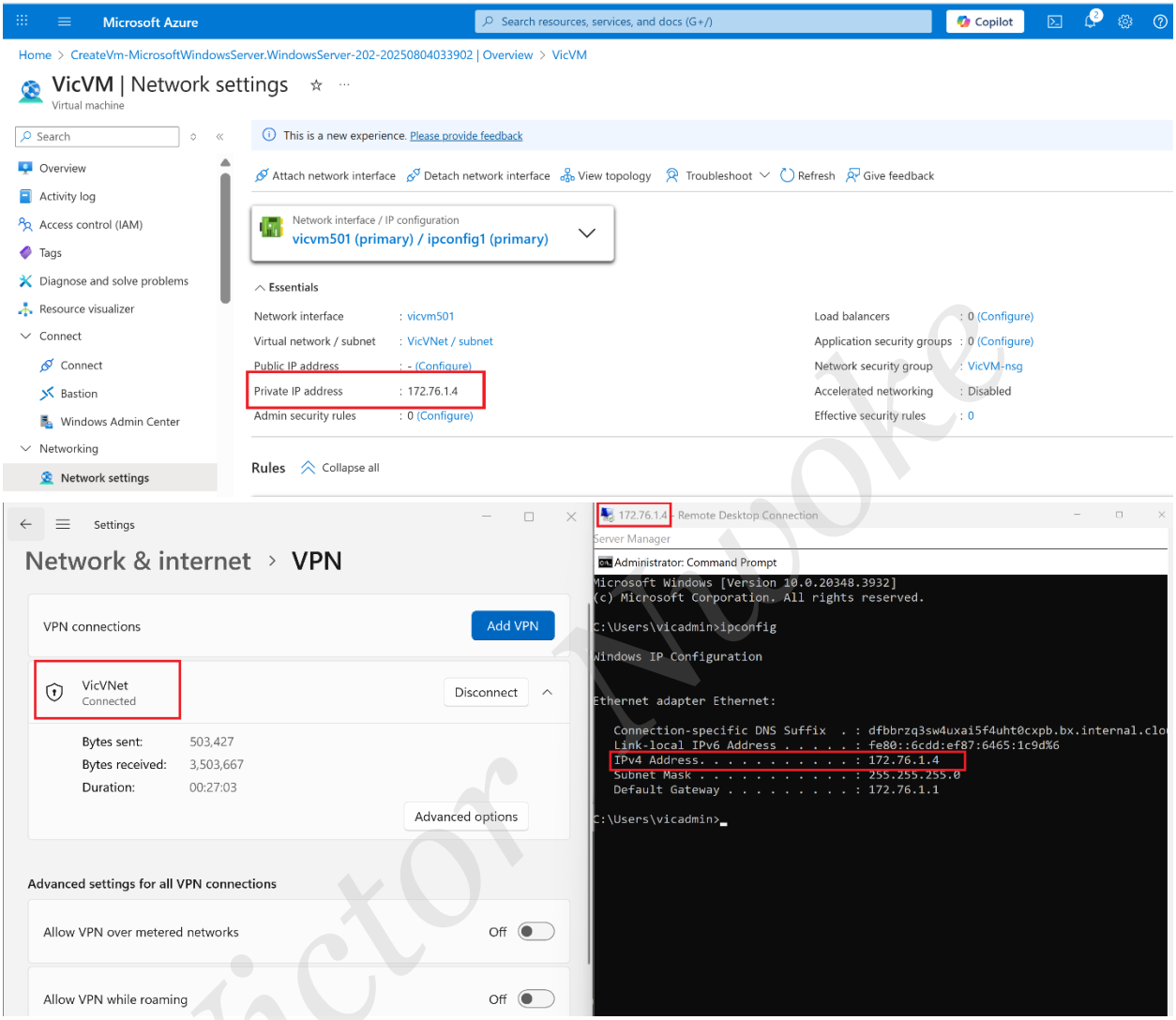


At this point, you have successfully configured your Point-to-Site VPN.



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.



Thank you for your time!