# Step-By-Step: Creating an Azure Point-to-Site VPN

Site-to-Site VPN is the most common method organizations use to connect on-premises network to Azure vNet. This VPN connection is initiated in your edge firewall or router level. But what if you connecting from remote location such as home? We can use point-to-site method to do that. In this method it will use certificates to do the authentication between end point and azure virtual network.

So,
let's go ahead and see how we can do that,

Create Resource Group
In this exercise, I like to use separate resource group for virtual network and other components.

1. Log in to **Azure portal** as global administrator
2. Launch Cloud Shell
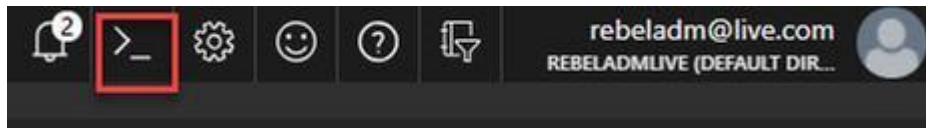
1. Then run **New-AzureRmResourceGroup -Name REBELVPNRG -Location "East US".** In here **REBELVPNRG** is resource group name and **East US** is the location.

Create Virtual Network

Now we need to create new virtual network. We can create virtual network using,

New-AzureRmVirtualNetwork -ResourceGroupName REBELVPNRG -Name REBEL-VNET -AddressPrefix 192.168.0.0/16 -Location "East US"

In above, **REBEL-VNET** is the virtual network name. it uses **192.168.0.0/16** IP address range.

```
PS Azure:\> New-AzureRmVirtualNetwork -ResourceGroupName REBELVPNRG -Name REBEL-VNET -AddressPrefix 192.168.0.0/16 -Location "East US"
WARNING: The output object type of this cmdlet will be modified in a future release.


Name                   : REBEL-VNET
ResourceGroupName      : REBELVPNRG
Location               : eastus
Id                     : /subscriptions/d                           resourceGroups/REBELVPNRG/providers/Microsoft.Network/virtualNetworks/REBEL-VNET
Etag                   : W/"2861e6b2-92df-4f00-9c04-13f5ea9dba13"
ResourceGuid           :
ProvisioningState      : Succeeded
Tags                   :
AddressSpace           : {
                           "AddressPrefixes": [
                             "192.168.0.0/16"
                           ]
                         }
DhcpOptions            : {}
Subnets                : []
VirtualNetworkPeerings : []
EnableDdosProtection   : false
DdosProtectionPlan     : null
EnableVmProtection     : false
```

Create Subnets

Under the virtual network I am going to create a subnet for my servers. To create subnet use,

$vn = Get-AzureRmVirtualNetwork -ResourceGroupName REBELVPNRG -Name REBEL-VNET

Add-AzureRmVirtualNetworkSubnetConfig -Name REBEL-SVR-SUB -VirtualNetwork $vn -AddressPrefix 192.168.100.0/24

Set-AzureRmVirtualNetwork -VirtualNetwork $vn

```
Azure:/
PS Azure:\> Set-AzureRmVirtualNetwork -VirtualNetwork $vn


Name                   : REBEL-VNET
ResourceGroupName      : REBELVPNRG
Location               : eastus
Id                     : /subscriptions/d                          /resourceGroups/REBELVPNRG/providers/Microsoft.Network/virtualNetworks/REBEL-VNET
Etag                   : W/"bfd3fbb6-4506-4b61-a3a6-72c64be69735"
ResourceGuid           :
ProvisioningState      : Succeeded
Tags                   :
AddressSpace           : {
                           "AddressPrefixes": [
                             "192.168.0.0/16"
                           ]
                         }
DhcpOptions            : {
                           "DnsServers": []
                         }
Subnets                : [
                           {
                             "Name": "REBEL-SVR-SUB",
                             "Etag": "W/\"I                         \"",
                             "Id": "/subscriptions/d                          /resourceGroups/REBELVPNRG/providers/Microsoft.Network/virtualNetworks/REBEL-VNET/subnets/REBEL-SVR-SUB",
                             "AddressPrefix": "192.168.100.0/24",
                             "IpConfigurations": [],
                             "ResourceNavigationLinks": [],
                             "ServiceEndpoints": [],
                             "ProvisioningState": "Succeeded"
```
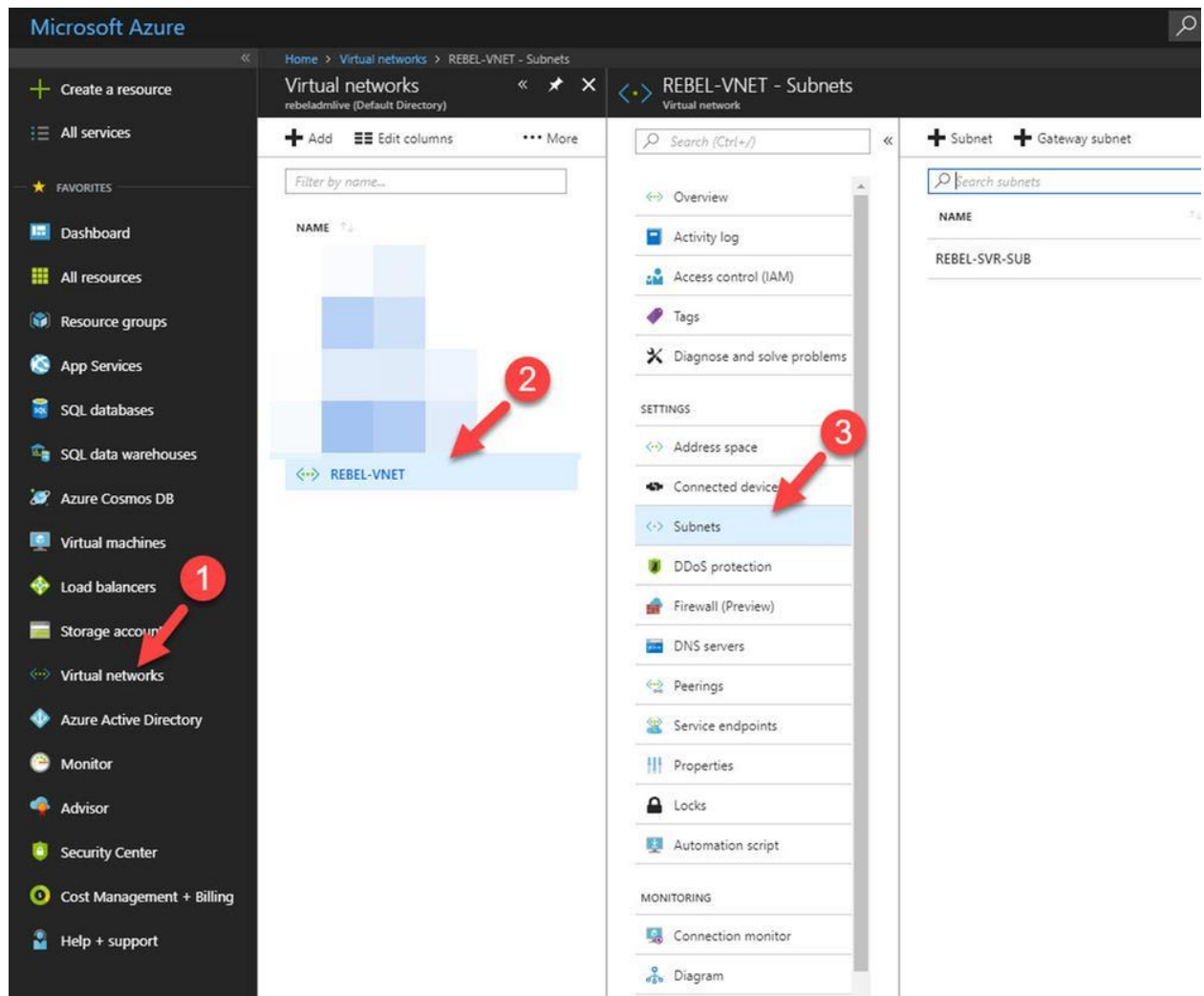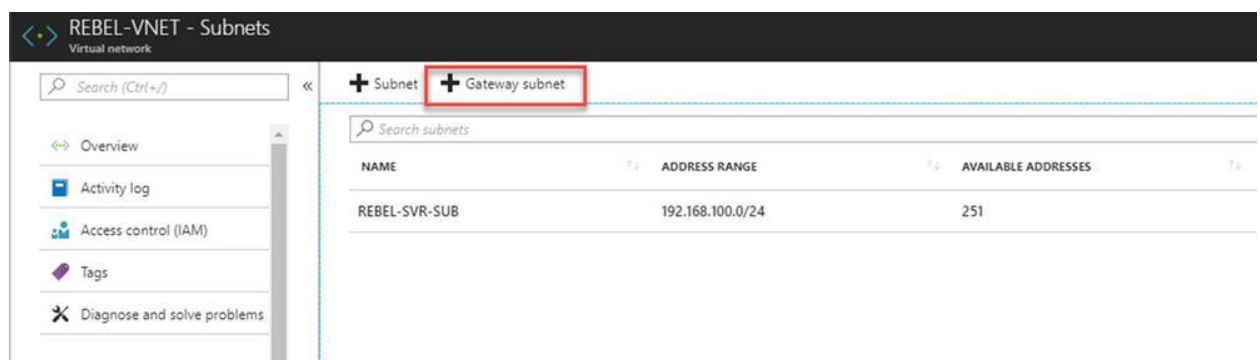
Create Gateway Subnet
Before we create VN gateway, we need to create gateway subnet for it. so gateway will use ip addresses assigned in this subnet.
To do that,

- Log in to Azure portal as global administrator
- Go to **Virtual Networks | REBEL-VNET** (VNet created on previous steps) **| Subnets**

- Click on **Gateway Subnet**

- In new window, define the ip range for gateway subnet and click **Ok**

Add subnet
REBEL-VNET

* Name

GatewaySubnet

* Address range (CIDR block) ⓘ

192.168.5.0/24 ✓

192.168.5.0 - 192.168.5.255 (251 + 5 Azure reserved addresses)

Route table
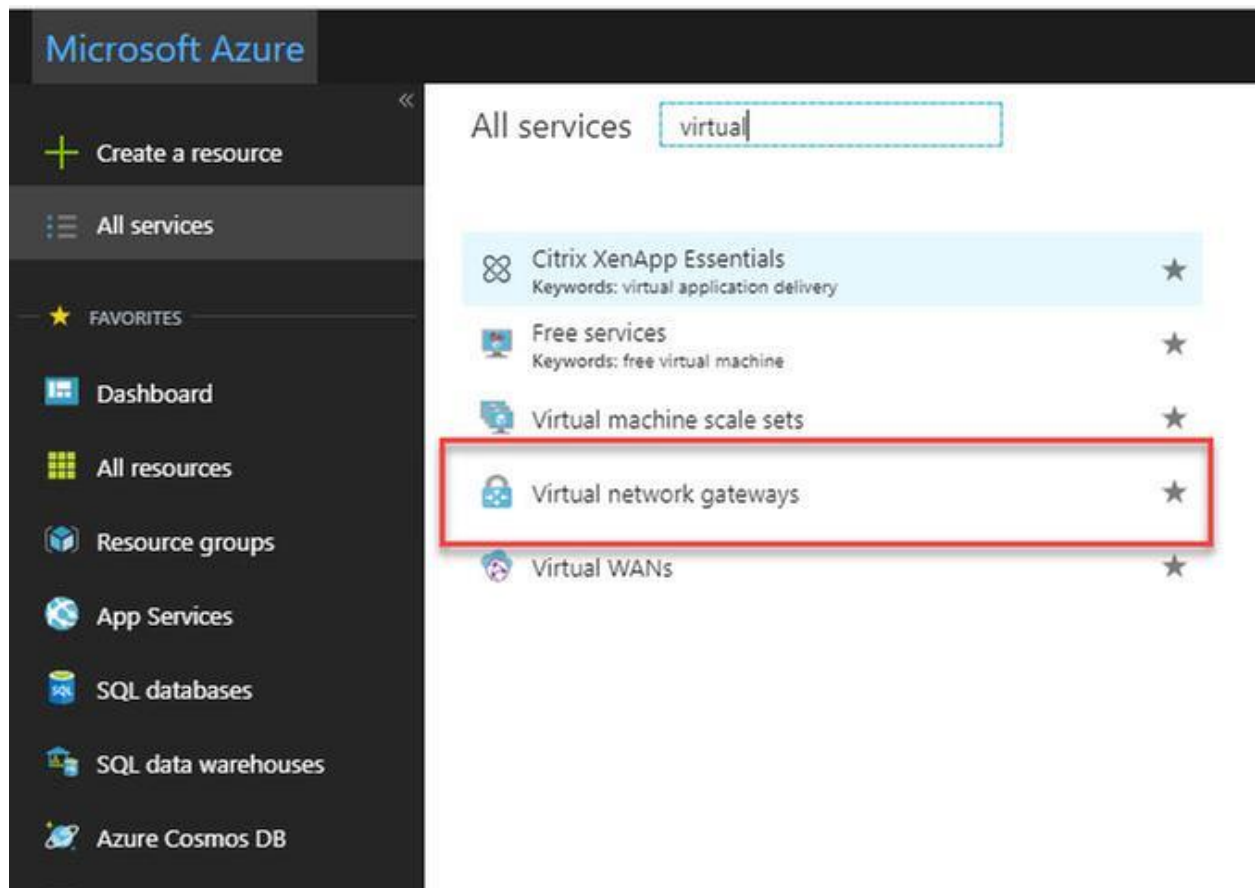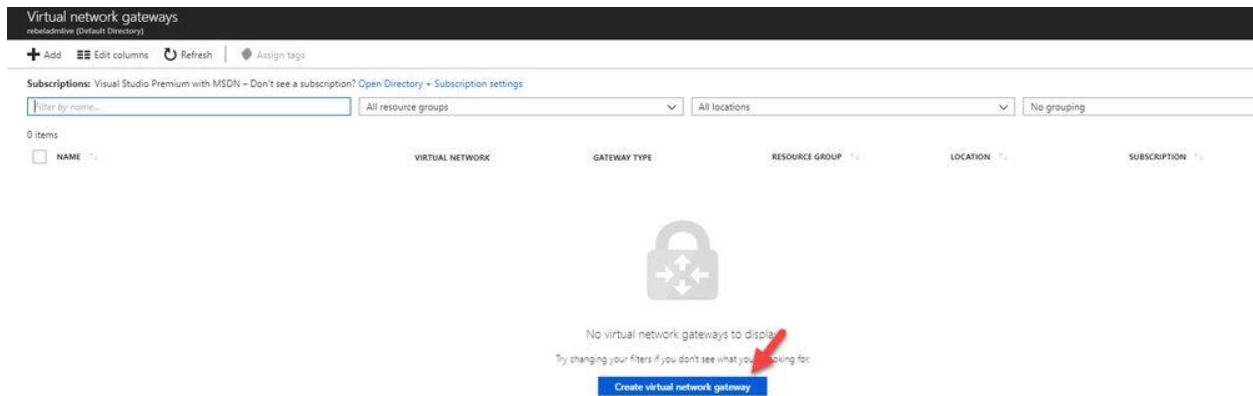None >

Service endpoints

Services ⓘ

0 selected

OK

Create Virtual Network Gateway

Now we have all the things needed to create new VN gateway. To do that,

- Log in to Azure portal as global administrator
- Go to **All Services** and search for virtual network gateway. Once it is in list, click on it.

- Then click on **Create virtual network gateway**

- In new window fill relevant info and click on **Create**

In here, **REBEL-VPN-GW** is the gateway name. I have selected **REBEL-VNET** as the virtual network. I am also creating public ip called **REBEL-PUB1**. This is only supported with dynamic mode. This doesn't mean it is going to change randomly. It will only happen when gateway is deleted or read.

Create virtual network gateway

* Name

REBEL-VPN-GW ✓

Gateway type ⓘ

⦿ VPN   ◯ ExpressRoute

VPN type ⓘ

⦿ Route-based   ◯ Policy-based

* SKU ⓘ

VpnGw1 ⌄

☐ Enable active-active mode ⓘ

* Virtual network ⓘ
REBEL-VNET   ❯

* Public IP address ⓘ

⦿ Create new   ◯ Use existing

REBEL-PUB1 ✓

⌃ Configure public IP address

SKU
Basic

* Assignment
⦿ Dynamic   ◯ Static

☐ Configure BGP ASN ⓘ

* Subscription

Visual Studio Premium with MSDN ⌄

Resource group ⓘ
REBELVPNRG

* Location ⓘ

East US ⌄

Create   Automation options

Create Self-sign root & client certificate

If your organization using internal CA, you always can use it to generate relevant certificates for this exercise. If you do not have internal CA, we still can use self-sign certs to do the job.

As first step I am going to create root certificate. In Windows 10 machine I can run this to create root cert first.

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

This will create root cert and install it under current user cert store.
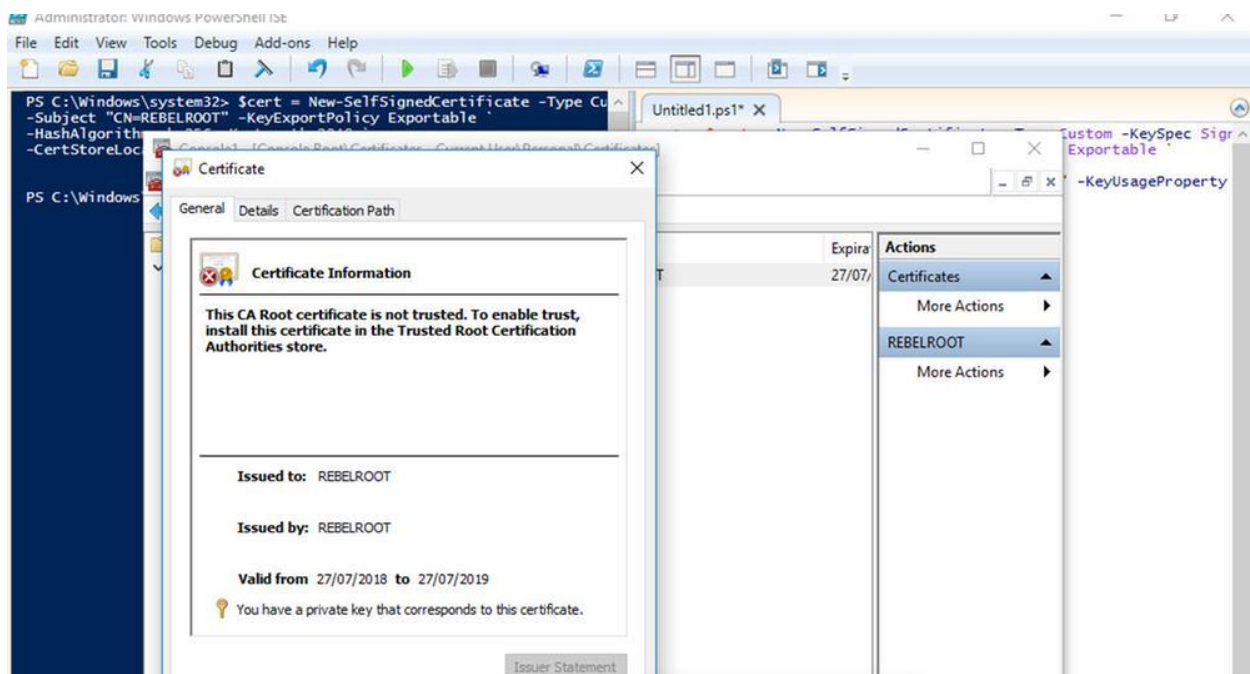
Then we need to create client certificate. We can do this using

New-SelfSignedCertificate -Type Custom -DnsName REBELCLIENT -KeySpec Signature `
-Subject "CN=REBELCLIENT" -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")

This will create cert called **REBELCLIENT** and install in same store location.
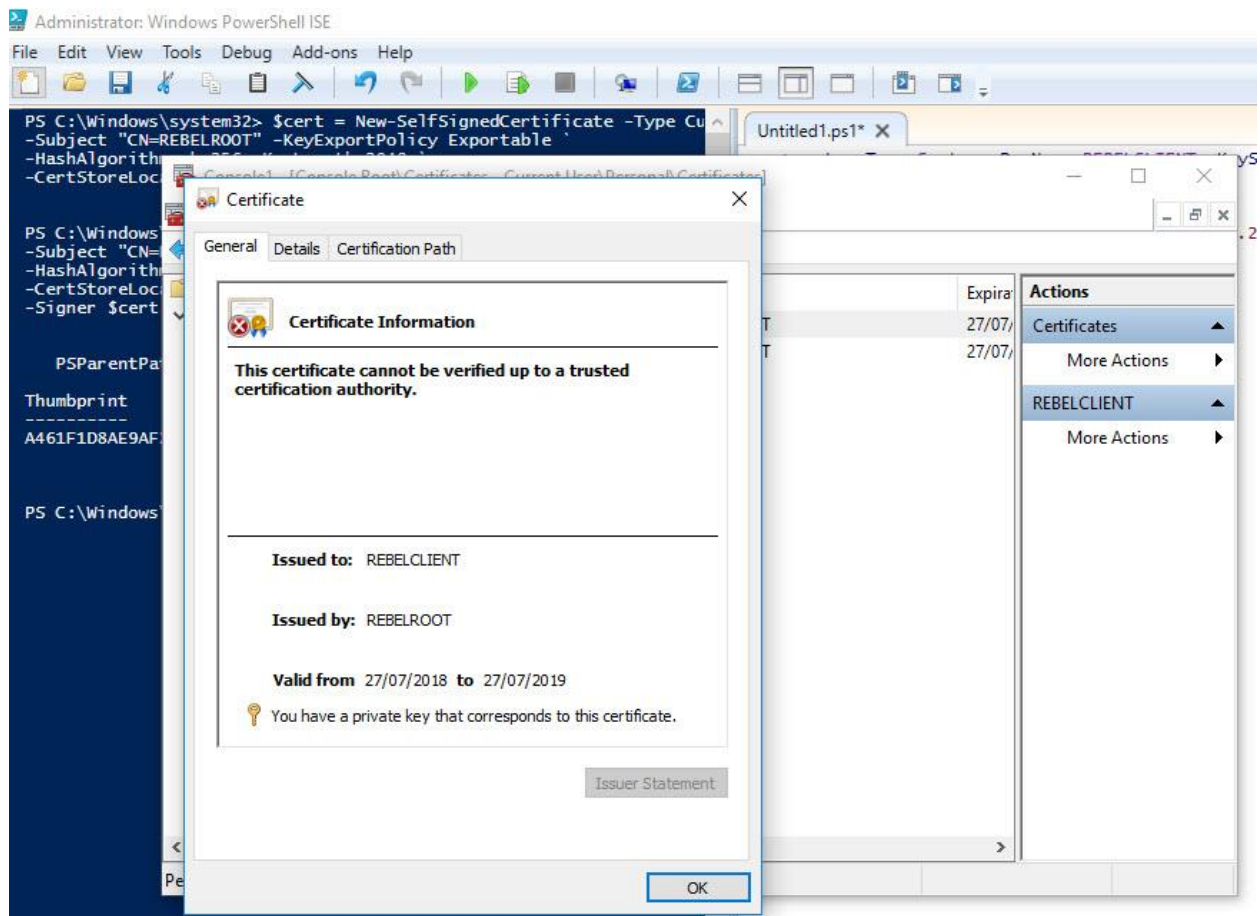
Now we have certs in place. But we need to export these so we can upload it to Azure.

To export root certificate,

- Right click on root cert inside certificate mmc.
- Click on **Export**
- In private key page, select not to export private key

← 🔖 Certificate Export Wizard   ✕

Export Private Key
    You can choose to export the private key with the certificate.

_____

    Private keys are password protected. If you want to export the private key with the
    certificate, you must type a password on a later page.

    Do you want to export the private key with the certificate?

        ○ Yes, export the private key

        ◉ No, do not export the private key

                                                    [Next]   [Cancel]

- Select Base-64 encoded X.509 as export file format.

- Complete the wizard and save the cert in pc.

To export client certificate,

- Use same method to export as root cert, but this time under private key page, select option to export private key.

← 🖼 Certificate Export Wizard

✕

Export Private Key
You can choose to export the private key with the certificate.

Private keys are password protected. If you want to export the private key with the certificate, you must type a password on a later page.

Do you want to export the private key with the certificate?

● Yes, export the private key

○ No, do not export the private key

[Next]   [Cancel]

- In file format page, leave the default as following and click **Next**

- Define password for the pfx file and complete the wizard.

Note – Only root cert will use in Azure VPN, client certificate can install on other computers which need P2S connections.

Configure Point-to-Site Connection

Next step of this configuration is to configure the point-to-site connection. In here we will define client ip address pool as well. It is for VPN clients.

- Click on newly created VPN gateway connection.
- Then in new window click on **Point-to-site configuration**

REBEL-VPN-GW - Point-to-site
Virtual network gateway

🔍 Search (Ctrl+/)   «

🔒 Overview

📄 Activity log

👥 Access control (IAM)

🏷️ Tags

🔧 Diagnose and solve problems

SETTINGS

💼 Configuration

🔗 Connections

⟨⟩ Point-to-site configuration

🎚️ Properties

🔒 Locks

🖥️ Automation script

MONITORING

📊 Metrics (Preview)

SUPPORT + TROUBLESHOOTING

💗 Resource health

🔧 Reset

👤 New support request

- After that, click on **Configure Now**
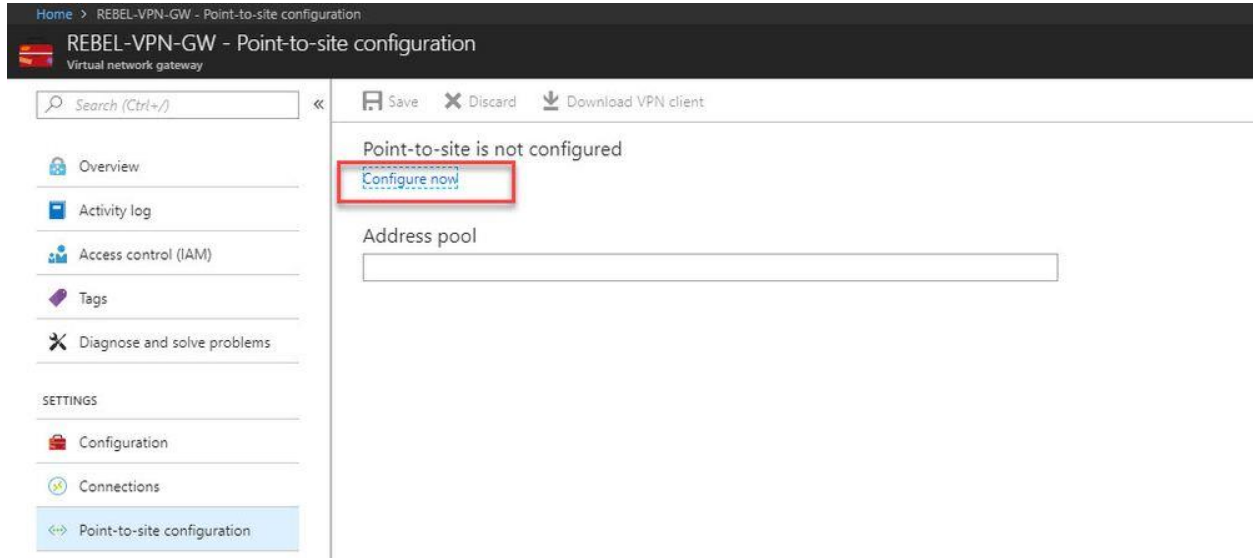
- In new window type IP address range for VPN address pool. In this demo I will be using **172.16.25.0/24**. For tunnel type use both **SSTP & IKEv2**. Linux and other mobile clients by default use **IKEv2** to connect. Windows also use **IKEv2** first and then try **SSTP**. For authentication type use **Azure Certificates**.

- In same window there is place to define root certificate. Under root certificate name type the cert name and under public certificate data, paste the root certificate data (you can open cert in notepad to get data).

Root certificates

NAME	PUBLIC CERTIFICATE DATA	
REBELROOT	-----BEGIN CERTIFICATE----- MIIC4zCCAcugAwIBAgIQGe3a6h9gnqJHOgz5PX7t1TANBgkqhkiG9w0BAQsFADAU MRIwEAYDVQQDDA1SRUJFTFJPT1	...
		...

Revoked certificates

NAME	THUMBPRINT	
		...


```
root - Notepad
File  Edit  Format  View  Help
-----BEGIN CERTIFICATE-----
MIIC4zCCAcugAwIBAgIQGe3a6h9gnqJHOgz5PX7t1TANBgkqhkiG9w0BAQsFADAU
MRIwEAYDVQQDDA1SRUJFTFJPT1QwHhcNMTgwNzI2MjMzMjIwWhcNMTkwNzI2MjM1
MjIwWjAUMRIwEAYDVQQDDA1SRUJFTFJPT1QwggEiMA0GCSqGSIb3DQEBAQUAA4IB
DwAwggEKAoIBAQCbni0FKkhsmoonl6NyUlpEUup4lJysgObS4c0ZqX2nGLuGa2L2
C7RECRoMRPQ4jw0+EwWyqFaA3ytedV14eBwKvroyXSLFJ0WIb0iGUdqM3uJP+7zn
MIC+qMBBScITz06/KzARBzdBuw9OzNkm6jAJS/3Lsf3rWrD2150nLEGmcgMgt19G
hvqoUK9ATabRUgmjKgRk4fxPLDH2D5MbX6xE+p74W9VF/NHDNPV88R8TIINtUCvo
cn+aTLLgk42yGv24PYHKE14iOaTf9A6DH+xzB4TtzufCHMxpnm5xSjudY7ZwJOn/
Gksw5FB46XEtOVgM0GDWzh8JkNVcI58wqgQFAgMBAAGjMTAvMA4GA1UdDwEB/wQE
AwICBDAdBgNVHQ4EFgQUUWua33B2rwAmfEfa8U6CZ4/dmrgwDQYJKoZIhvcNAQEL
BQADggEBACQB51Y/nWX822FnL8Ra7hHQv46xJXcjLk2HdoMrNHI0PGQm4yXxmFm9
lYgwrWw+A/1H47TOT4SWj8TjMEPE0WLipxiE2si8cEbHblMqhl+aET3yhblhiWLB
6ReNEDMneO6qEMbVOatpfaf8d4xehy3/KzYFSvHq7oWU3oVtyb7ACD8/q+9jtIX0
rNP8/CCRIpSJ0NzPJoZXb7jYXF/Sejh2FeriBO7n+Z/ZCVt69BWbdwbzwsbQVMD9
LKyXdSjl8hv0XI7L7s6LGWChYb0YhXhDPtC6jvqTPYvN3jfnQJSZ5H7v/rHdYLE9
1oi8wDKpI3uVeFTdA/kCJ/kLxjdySQU=
-----END CERTIFICATE-----
```

- Then click on **Save** to complete the process.

Note : when you paste certificate data, do not copy -----BEGIN CERTIFICATE----- & -----END CERTIFICATE----- text.

Testing VPN connection

Now we have finished with configuration. As next step, we need to test the connection. To do that log in to the same pc where we generate certificates. If you going to use different PC, first you need to import root cert & client certificate we exported.

- Log in to Azure portal from machine and go to VPN gateway config page.
- In that page, click on **Point-to-site configuration**
- After that, click on **Download VPN client**

- Then double click on the VPN client setup. In my case I am using 64bit vpn client.
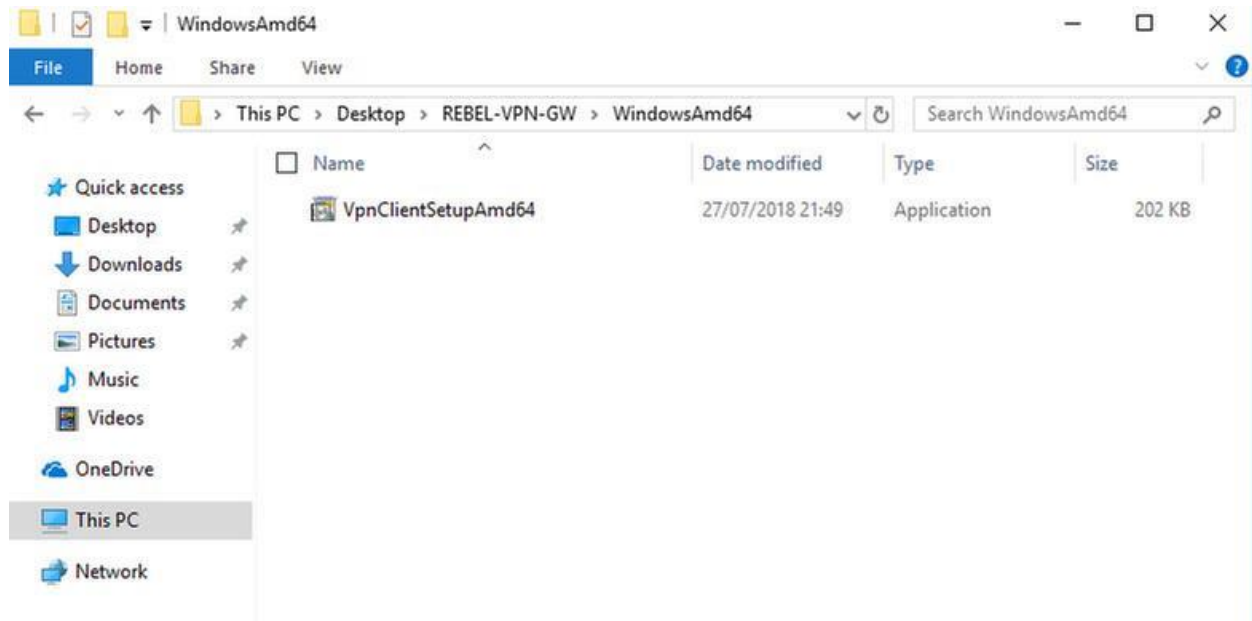
- After that, we can see new connection under windows 10 VPN page.
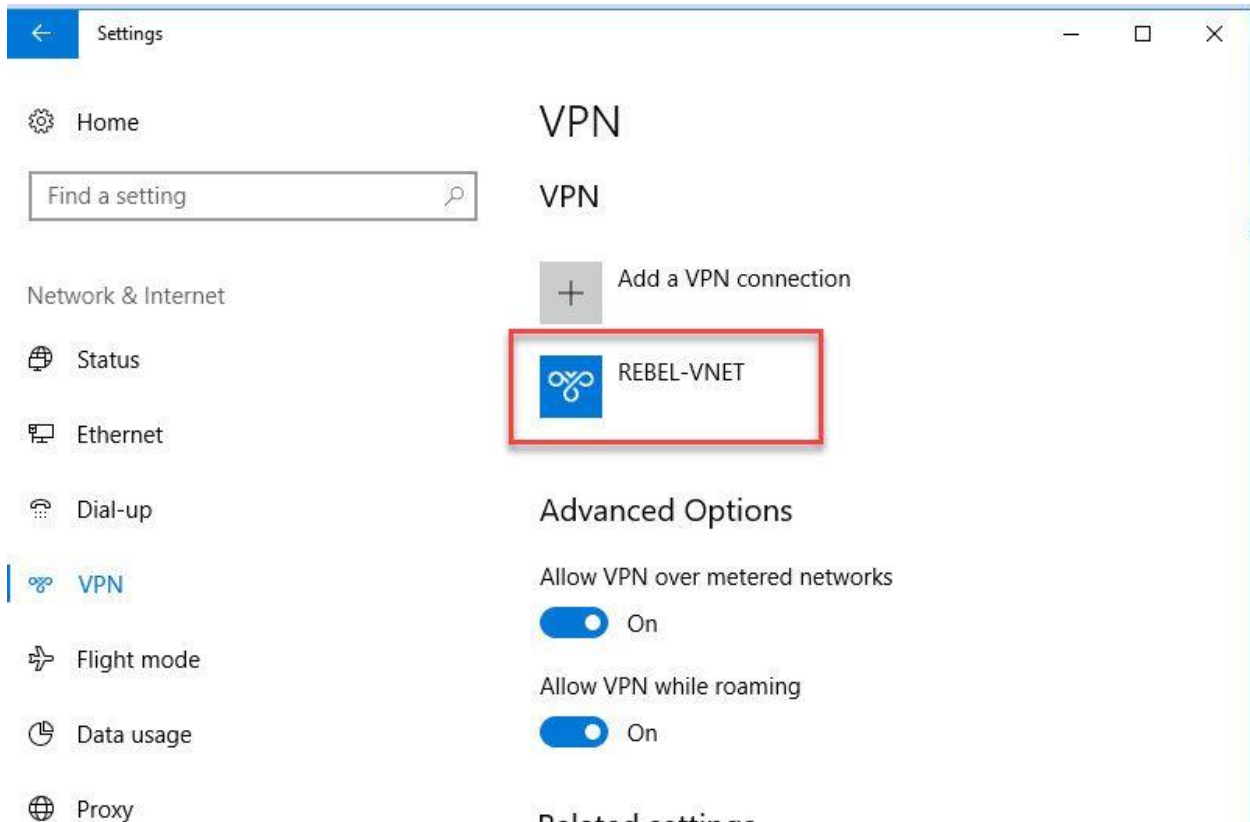
- Click on connect to VPN. Then it will open up this new window. Click on **Connect** in there.

- Then run ip config to verify ip allocation from VPN address pool.

- In VPN gateway page also, I can see one connection is made.

🖫 Save   ✕ Discard   ⬇ Download VPN client

Connection health

Connections	1
Ingress (bytes)	0
Egress (bytes)	0

Address pool

172.16.25.0/24

Tunnel type

SSL VPN (SSTP)   ☑

IKEv2 VPN   ☑

Authentication type

◉ Azure certificate   ◯ RADIUS authentication

Root certificates

NAME	PUBLIC (
REBELROOT	MIIC4z(

Revoked certificates

NAME	THUMBF

Allocated IP addresses

172.16.25.2

- I have a server setup under new virtual network we created. This server only has private ip and its 192.168.100.4

- As expected, I can RDP to this via VPN.