How to create a VPN between Azure and AWS using only managed solutions

By Ricardo Macedo Martins
Published Jun 17 2021 12:48 PM

What if you can establish a connection between Azure and AWS using only managed solutions instead to have to use virtual machines? This is exactly what we'll be covering on this article connecting AWS Virtual Private Gateway with the Azure VPN Gateway directly without worry to manage IaaS resources like virtual machines.

Below the draw of our lab:

Co-Authors

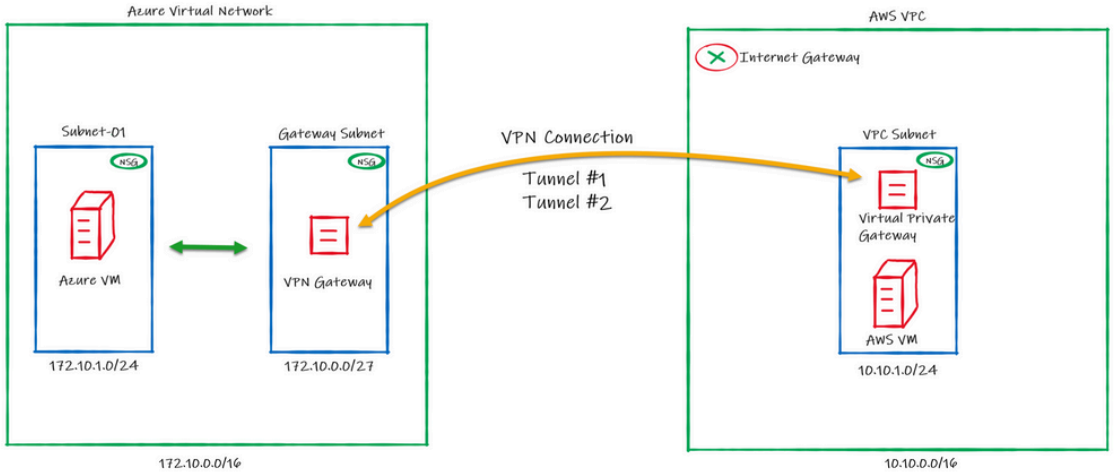
rmmartins

Version history

Last update: Aug 22 2022 04:50 PM
Updated by: rmmartins

Labels

Infra 122



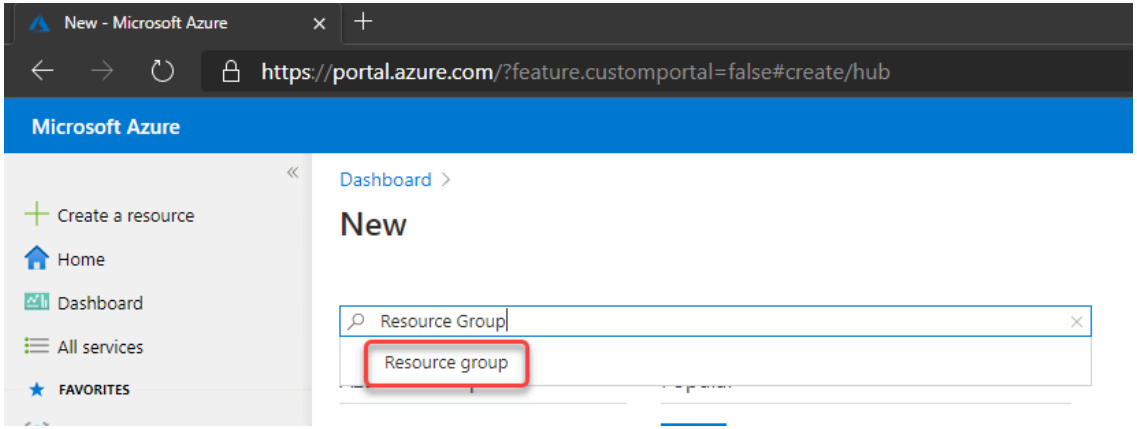
Share

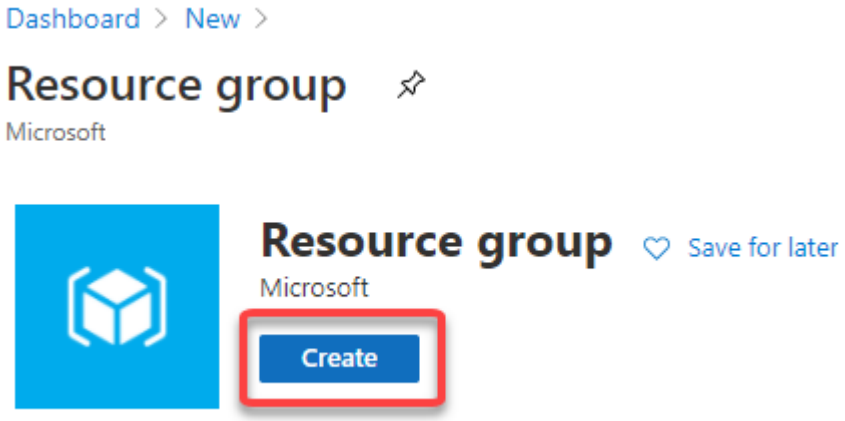
Regarding the high availability, please note that on AWS, by default a VPN connection always will have 2 Public IPs, one per tunnel. On Azure it doesn't happens by default and in this case you will be using Active/Passive from Azure side.

This means that we will be setting only one "node" from Azure VPN Gateway to establish two VPN connections with AWS. In case of a failure, the second node from Azure VPN Gateway will connect to AWS in a Active/Passive mode.

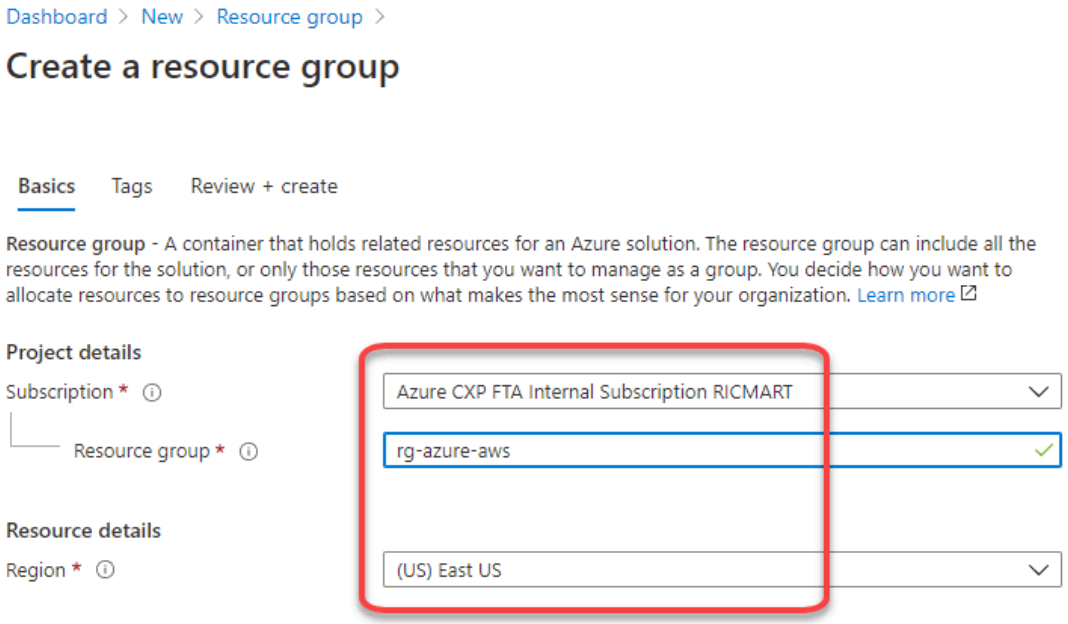
Configuring Azure

1. Crate a resource group on Azure to deploy the resources on that

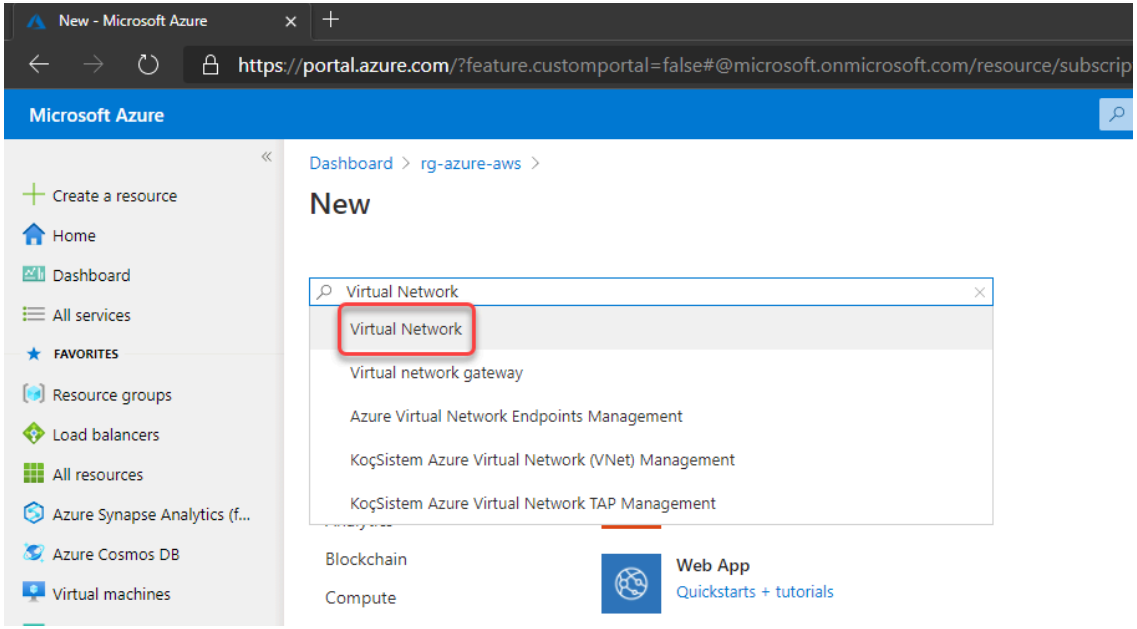


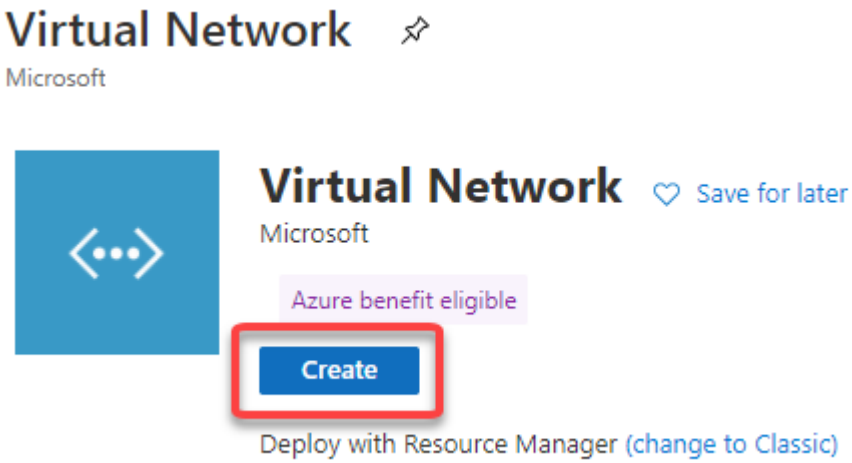


Choose the subscription, the name and the region to be deployed:



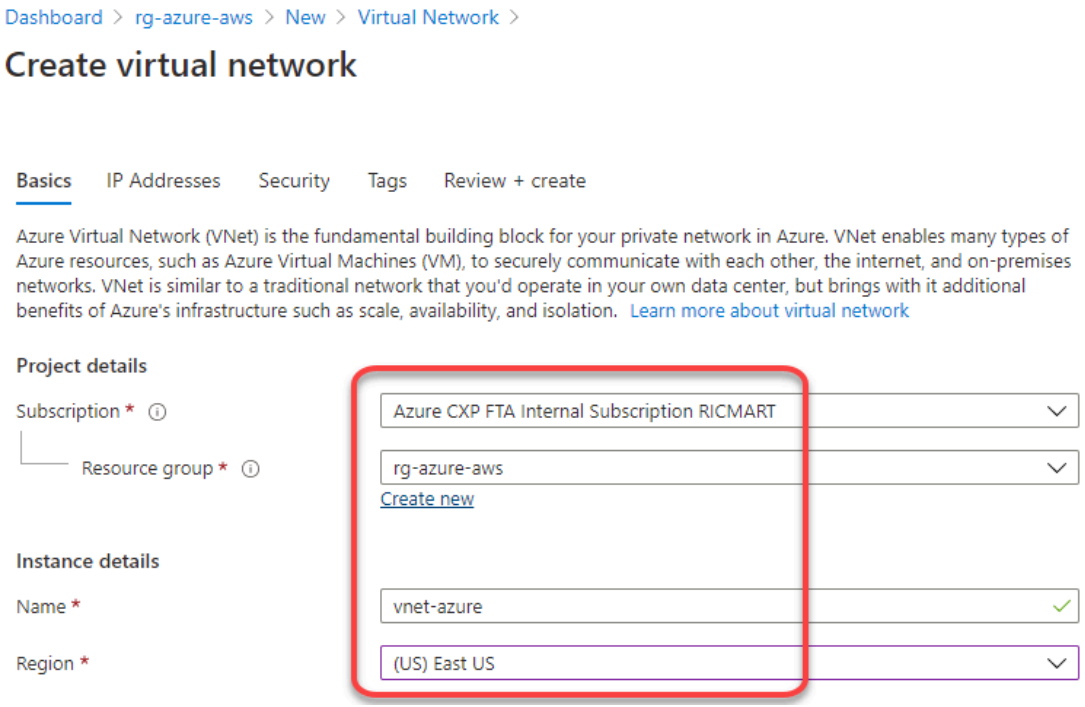
2. Create a Virtual Network and a subnet





Define the subscription, resource group, name and region to be deployed:

Skip to Main Content



Set the address space for the virtual network and for the subnet. Here I'm defining the virtual network address space to **172.10.0.0/16**, changing the "default" subnet name to **"subnet-01"** and defining the subnet address range to **172.10.1.0/24**:

Create virtual network

Basics | IP Addresses | Security | Tags | Review + create

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

IPv4 address space

172.10.0.0/16

Add IPv6 address space

The subnet's address range in CIDR notation (e.g. 192.168.1.0/24). It must be contained by the address space of the virtual network.

Add subnet | Remove subnet

Subnet name	Subnet address range
subnet-01	172.10.1.0/24

Create virtual network

Validation passed

Basics IP Addresses Security Tags **Review + create**

Basics

Subscription	Azure CXP FTA Internal Subscription RICMART
Resource group	rg-azure-aws
Name	vnet-azure
Region	East US

IP addresses

Address space	172.10.0.0/16
Subnet	subnet-01 (172.10.1.0/24)

Tags

None

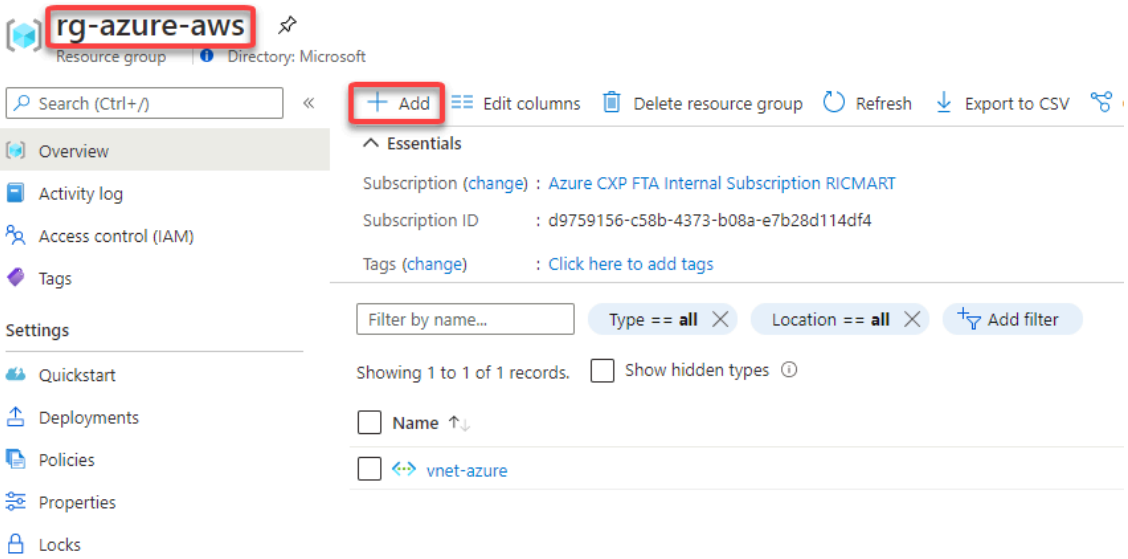
Security

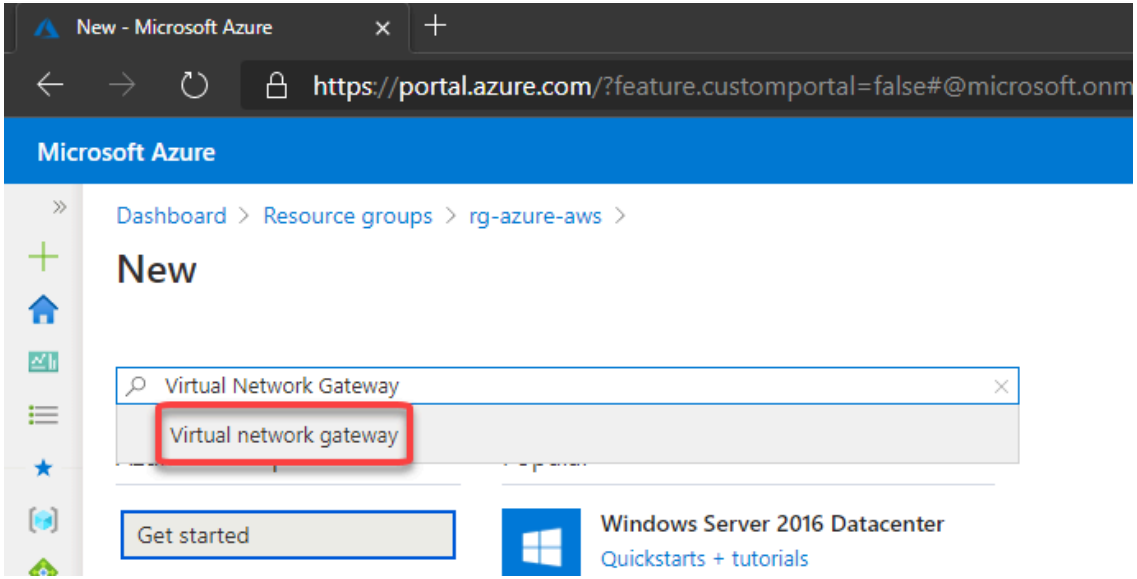
BastionHost	Disabled
DDoS protection plan	Basic
Firewall	Disabled

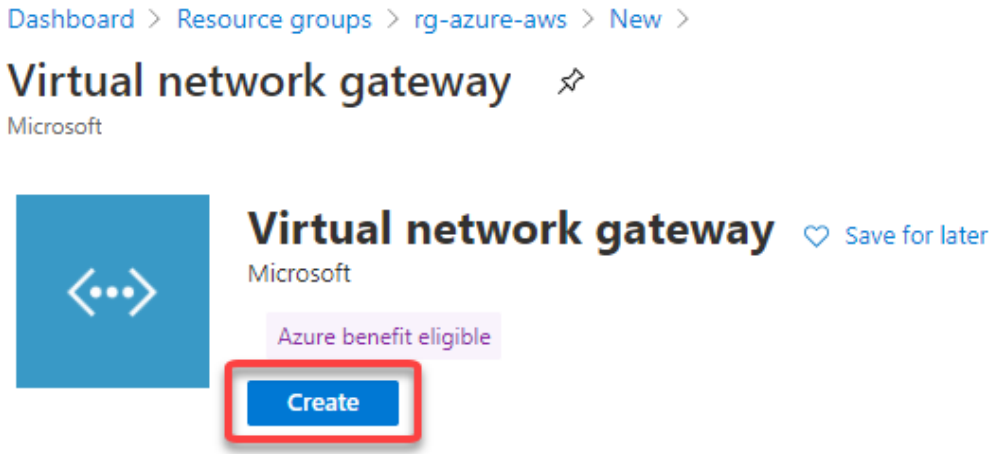
3. Create the VPN Gateway

The Azure VPN Gateway is a resource composed of 2 or more VM's that are deployed to a specific subnet called Gateway Subnet where the recommendation is to use a /27. He contain routing tables and run specific gateway services. Note that you can't access those VM's.

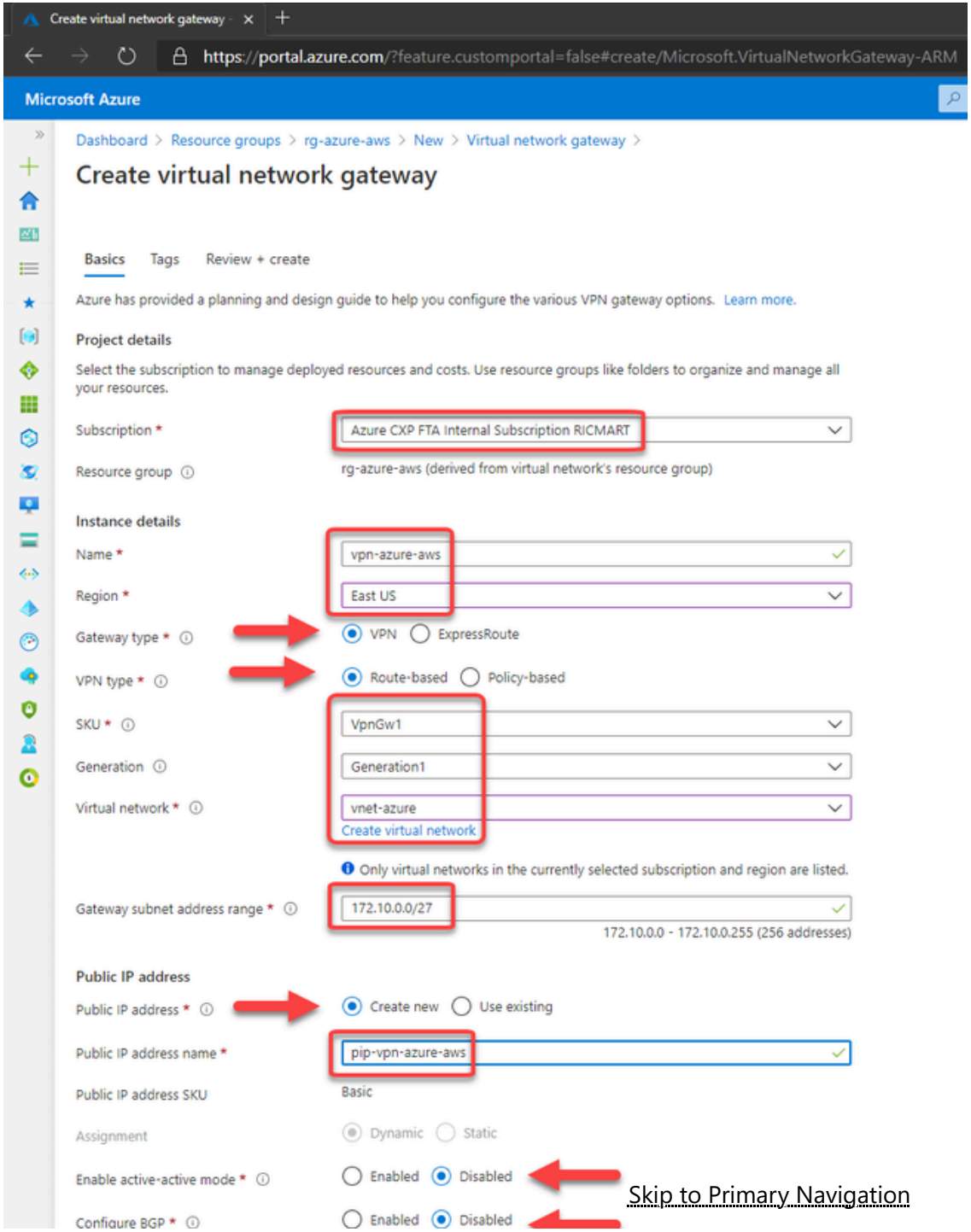
To create, go to your Resource Group, then click to **+ Add**



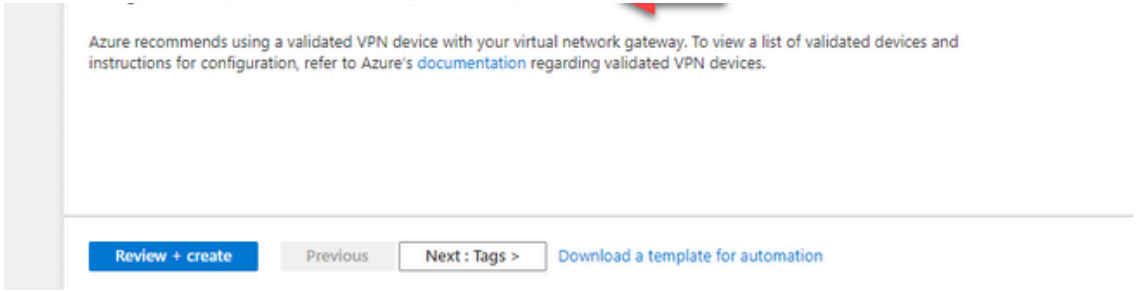




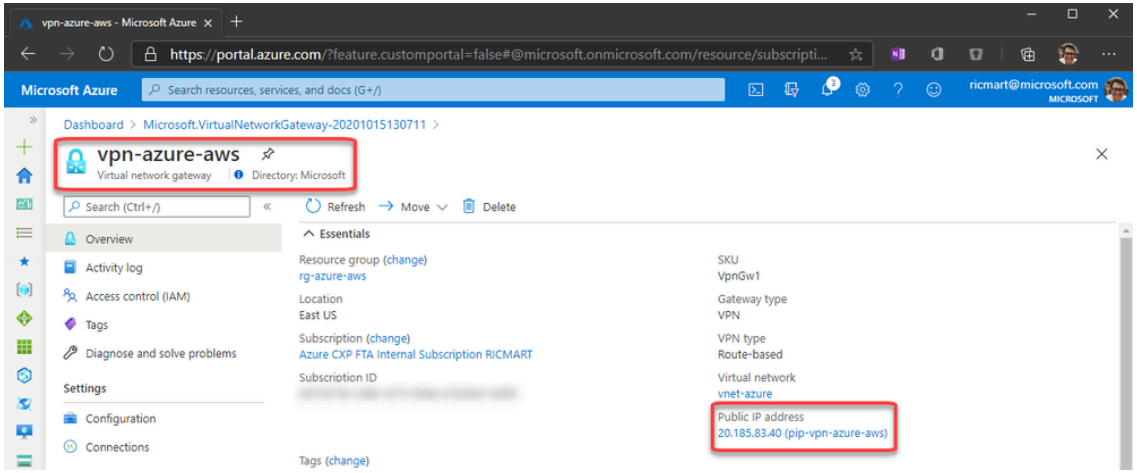
Then fill the fields like below:



Skip to Primary Navigation

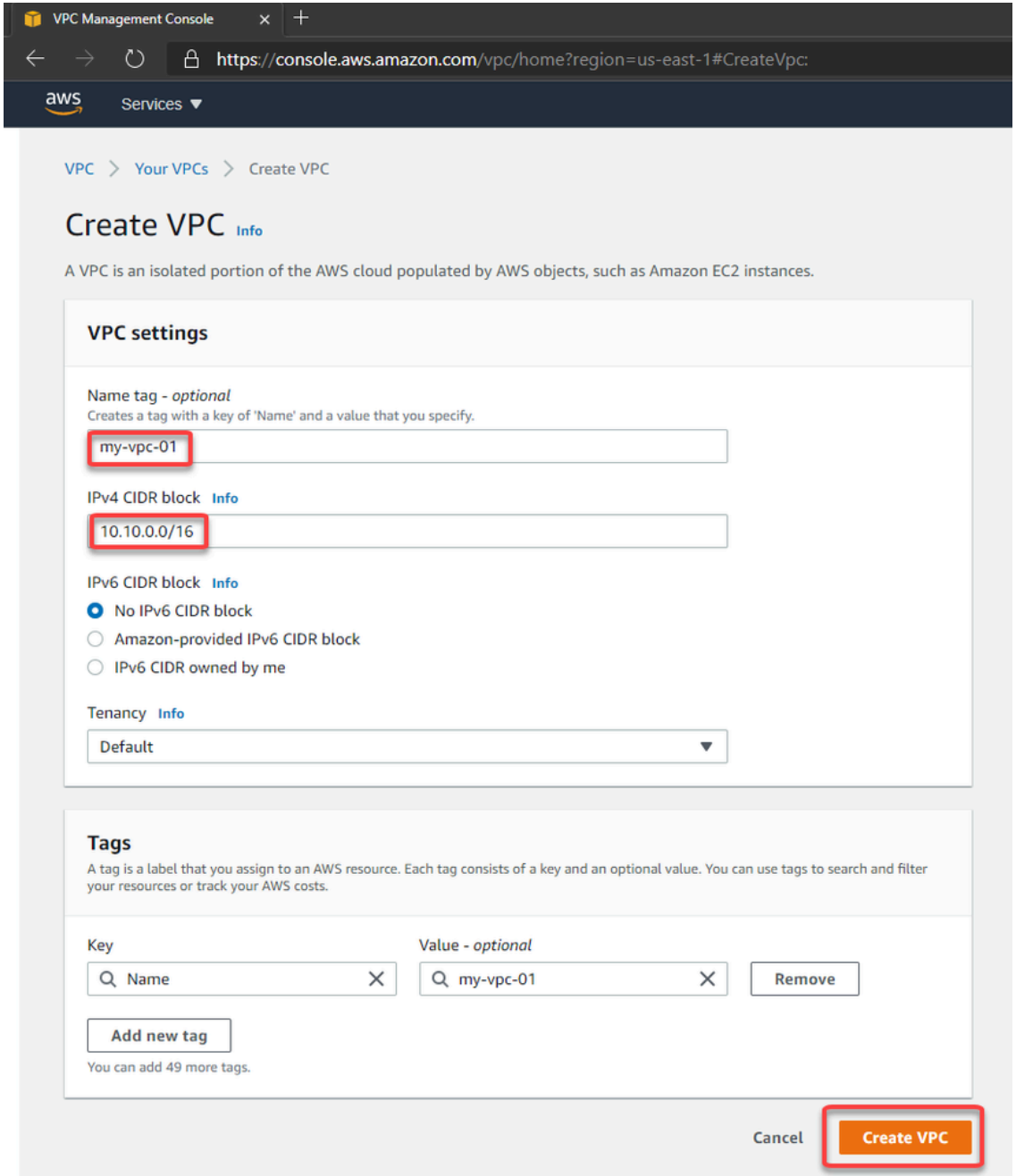


After click to Review + create, in a few minutes the Virtual Network Gateway will be ready:



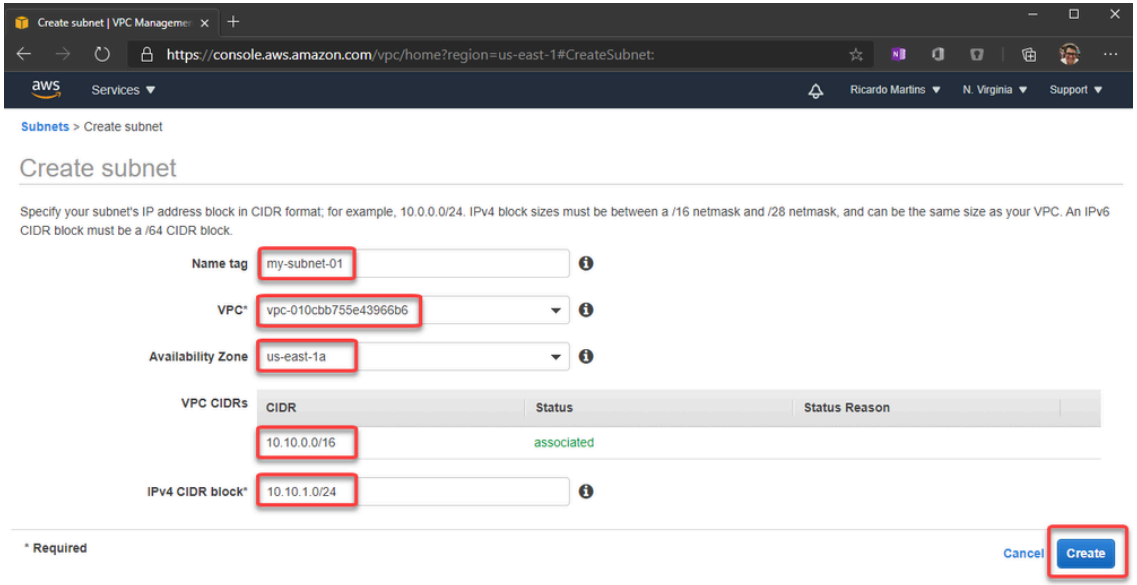
Configuring AWS

4. Create the Virtual Private Cloud (VPC)



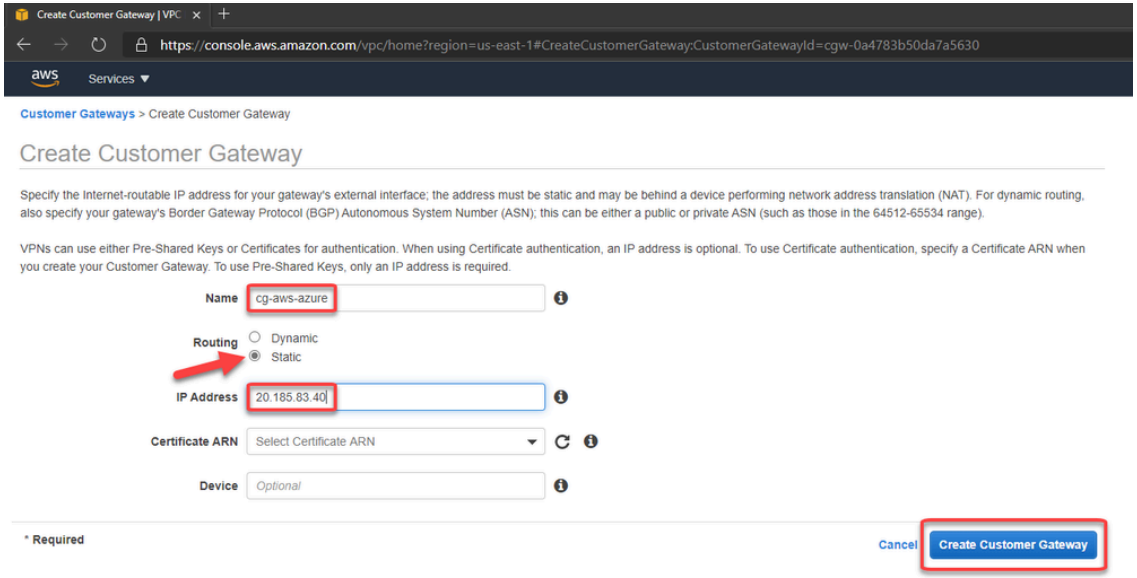
5. Create a subnet inside the VPC (Virtual Network)

Skip to Main Content

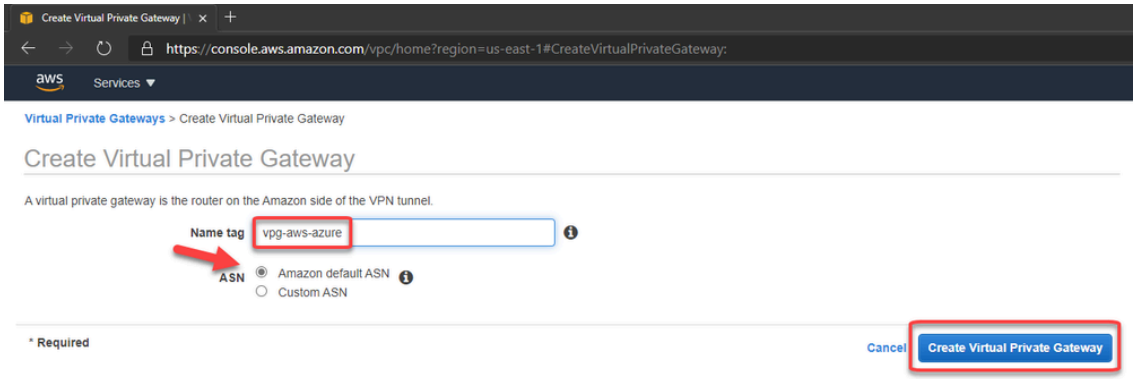


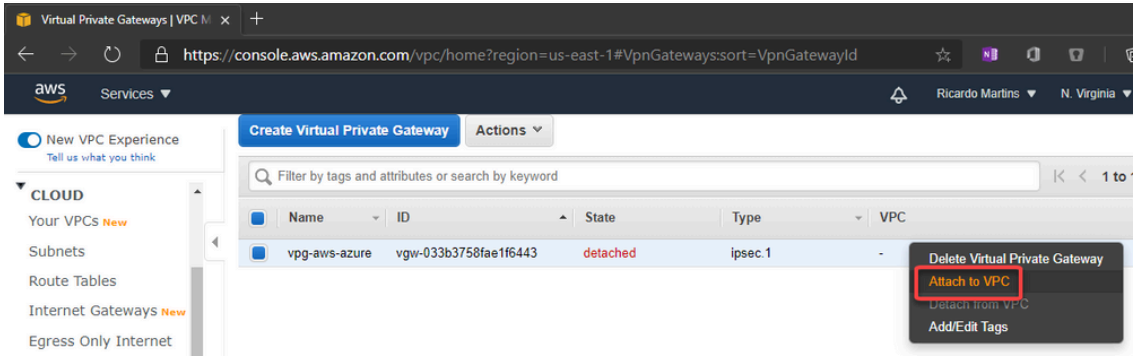
6. Create a customer gateway pointing to the public ip address of Azure VPN Gateway

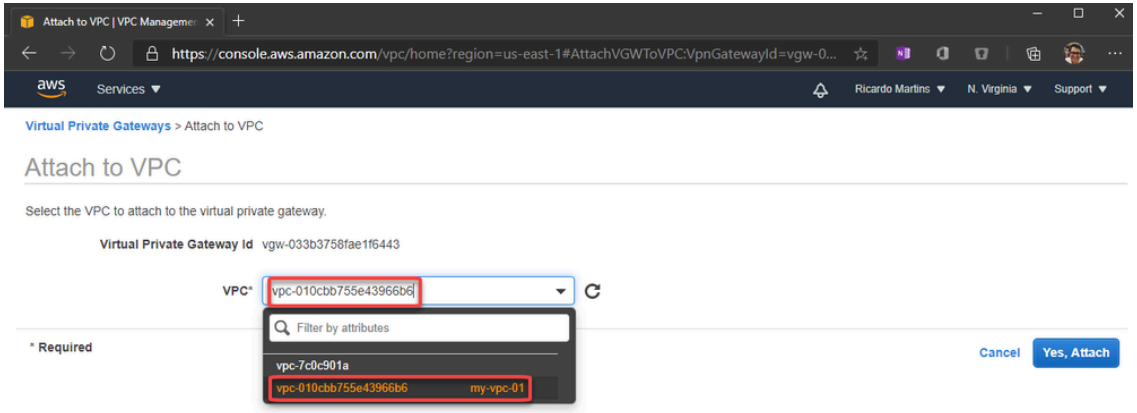
The Customer Gateway is an AWS resource with information to AWS about the customer gateway device, which in this case is the Azure VPN Gateway.



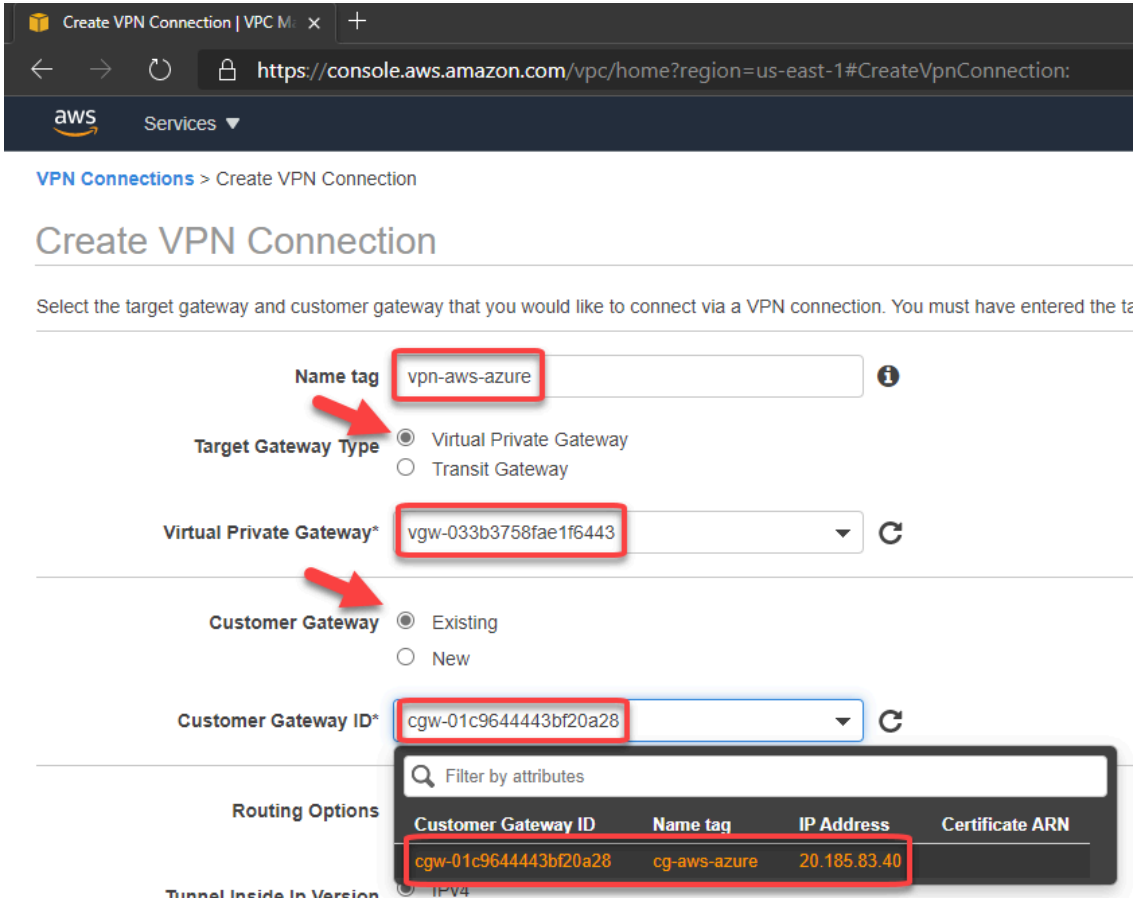
7. Create the Virtual Private Gateway then attach to the VPC







8. Create a site-to-site VPN Connection

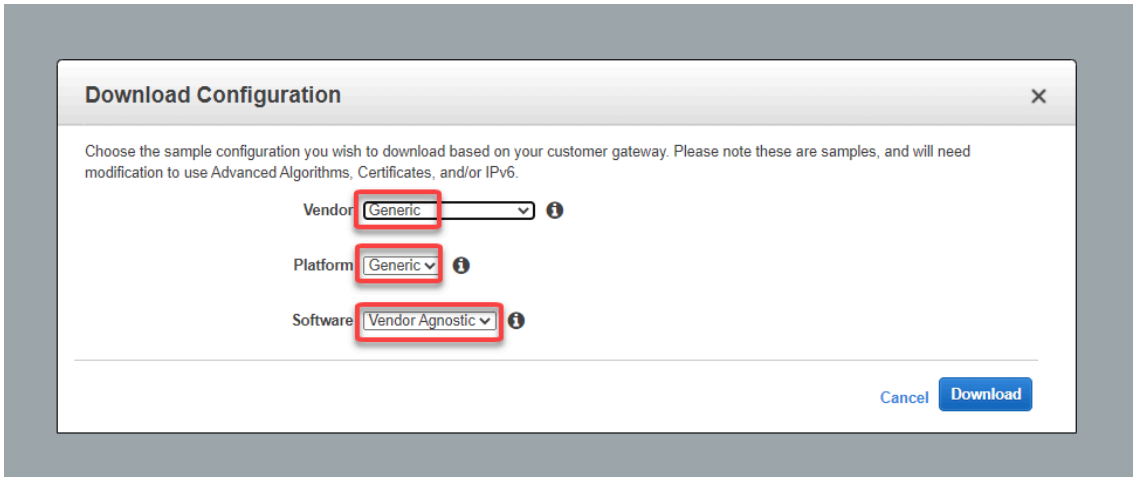


Set the routing as static pointing to the azure subnet-01 prefix (172.10.1.0/24)

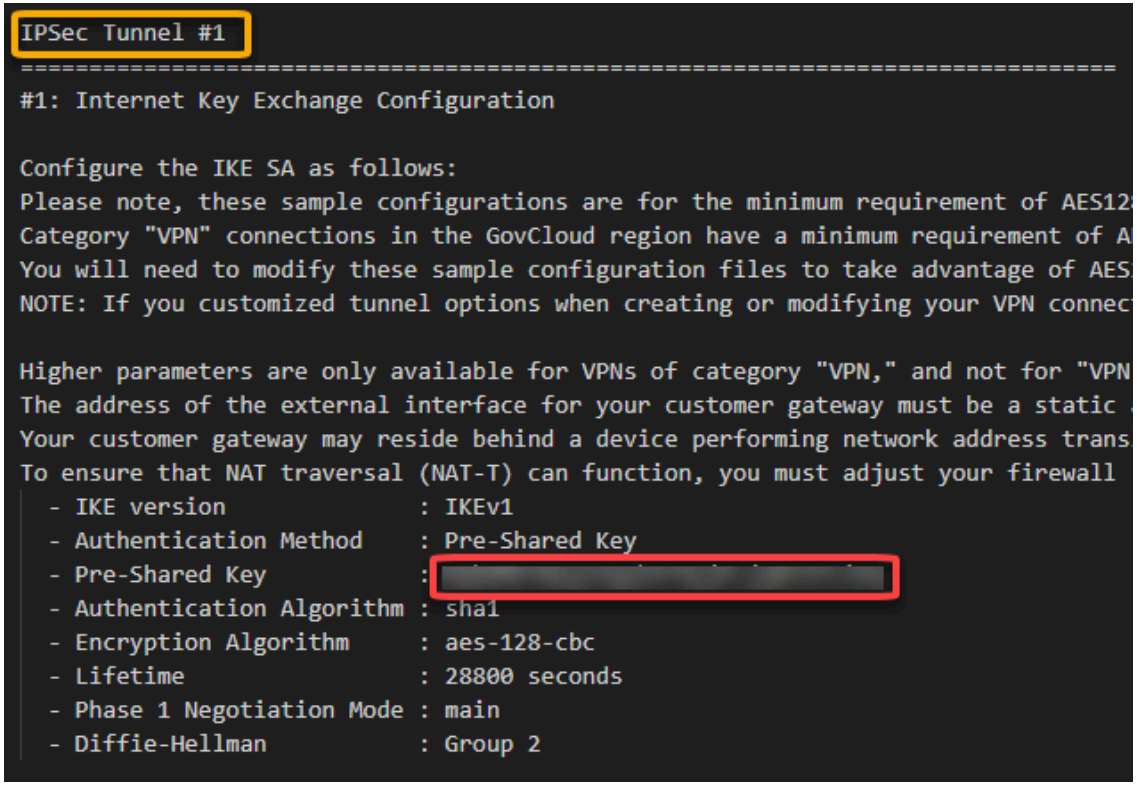
After fill the options, click to create.

9. Download the configuration file

Please note that you need to change the Vendor, Platform and Software to Generic since Azure isn't a valid option:



In this configuration file you will note that there are the Shared Keys and the Public Ip Address for each of one of the two IPSec tunnels created by AWS:


IPSec Tunnel #1
#1: Internet Key Exchange Configuration
Configure the IKE SA as follows:
Please note, these sample configurations are for the minimum requirement of AES12
Category "VPN" connections in the GovCloud region have a minimum requirement of A
You will need to modify these sample configuration files to take advantage of AES
NOTE: If you customized tunnel options when creating or modifying your VPN connec
Higher parameters are only available for VPNs of category "VPN," and not for "VPN
The address of the external interface for your customer gateway must be a static
Your customer gateway may reside behind a device performing network address trans
To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall
- IKE version : IKEv1
- Authentication Method : Pre-Shared Key
- Pre-Shared Key :
- Authentication Algorithm : sha1
- Encryption Algorithm : aes-128-cbc
- Lifetime : 28800 seconds
- Phase 1 Negotiation Mode : main
- Diffie-Hellman : Group 2

```
#2: IPSec Configuration

Configure the IPSec SA as follows:
Category "VPN" connections in the GovCloud region have a minimum requirement of AE
Please note, you may use these additionally supported IPSec parameters for encrypt
NOTE: If you customized tunnel options when creating or modifying your VPN connect

Higher parameters are only available for VPNs of category "VPN," and not for "VPN-
  - Protocol                : esp
  - Authentication Algorithm : hmac-sha1-96
  - Encryption Algorithm     : aes-128-cbc¤æ
  - Lifetime                 : 3600 seconds
  - Mode                     : tunnel
  - Perfect Forward Secrecy  : Diffie-Hellman Group 2

IPSec Dead Peer Detection (DPD) will be enabled on the AWS Endpoint. We
recommend configuring DPD on your endpoint as follows:
  - DPD Interval             : 10
  - DPD Retries              : 3

IPSec ESP (Encapsulating Security Payload) inserts additional
headers to transmit packets. These headers require additional space,
which reduces the amount of space available to transmit application data.
To limit the impact of this behavior, we recommend the following
configuration on your Customer Gateway:
  - TCP MSS Adjustment       : 1379 bytes
  - Clear Don't Fragment Bit : enabled
  - Fragmentation            : Before encryption

#3: Tunnel Interface Configuration

Your Customer Gateway must be configured with a tunnel interface that is
associated with the IPSec tunnel. All traffic transmitted to the tunnel
interface is encrypted and transmitted to the Virtual Private Gateway.



The Customer Gateway and Virtual Private Gateway each have two addresses that rela
to this IPSec tunnel. Each contains an outside address, upon which encrypted
traffic is exchanged. Each also contain an inside address associated with
the tunnel interface.
```



```
The Customer Gateway outside IP address was provided when the Customer Gateway
```

```
was created. Changing the IP address requires the creation of a new
Customer Gateway.

The Customer Gateway inside IP address should be configured on your tunnel
interface.

Outside IP Addresses:
  - Customer Gateway            : 20.185.83.40
  - Virtual Private Gateway      : 3.209.186.24
```

```
IPSec Tunnel #2
================================================================================
#1: Internet Key Exchange Configuration

Configure the IKE SA as follows:
Please note, these sample configurations are for the minimum requirement of AES12
Category "VPN" connections in the GovCloud region have a minimum requirement of A
You will need to modify these sample configuration files to take advantage of AES
NOTE: If you customized tunnel options when creating or modifying your VPN connec

Higher parameters are only available for VPNs of category "VPN," and not for "VPN
The address of the external interface for your customer gateway must be a static
Your customer gateway may reside behind a device performing network address trans
To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall
  - IKE version               : IKEv1
  - Authentication Method     : Pre-Shared Key
  - Pre-Shared Key            : [redacted]
  - Authentication Algorithm  : sha1
  - Encryption Algorithm      : aes-128-cbc
  - Lifetime                  : 28800 seconds
  - Phase 1 Negotiation Mode  : main
  - Diffie-Hellman            : Group 2
```

#2: IPSec Configuration

Configure the IPSec SA as follows:
Category "VPN" connections in the GovCloud region have a minimum requirement of AE
Please note, you may use these additionally supported IPSec parameters for encrypt
NOTE: If you customized tunnel options when creating or modifying your VPN connect

Higher parameters are only available for VPNs of category "VPN," and not for "VPN-

- Protocol : esp
- Authentication Algorithm : hmac-sha1-96
- Encryption Algorithm : aes-128-cbc
- Lifetime : 3600 seconds
- Mode : tunnel
- Perfect Forward Secrecy : Diffie-Hellman Group 2

IPSec Dead Peer Detection (DPD) will be enabled on the AWS Endpoint. We
recommend configuring DPD on your endpoint as follows:

- DPD Interval : 10
- DPD Retries : 3

IPSec ESP (Encapsulating Security Payload) inserts additional
headers to transmit packets. These headers require additional space,
which reduces the amount of space available to transmit application data.
To limit the impact of this behavior, we recommend the following
configuration on your Customer Gateway:

- TCP MSS Adjustment : 1379 bytes
- Clear Don't Fragment Bit : enabled
- Fragmentation : Before encryption

#3: Tunnel Interface Configuration

Your Customer Gateway must be configured with a tunnel interface that is
associated with the IPSec tunnel. All traffic transmitted to the tunnel
interface is encrypted and transmitted to the Virtual Private Gateway.

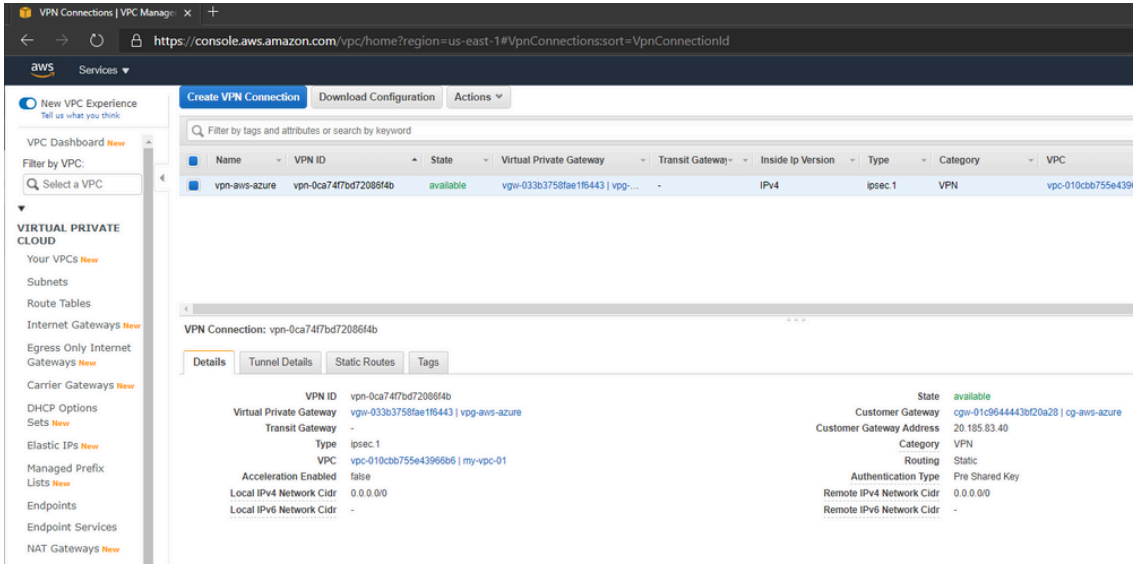
The Customer Gateway and Virtual Private Gateway each have two addresses that rela
to this IPSec tunnel. Each contains an outside address, upon which encrypted
traffic is exchanged. Each also contain an inside address associated with
the tunnel interface.

```
was created. Changing the IP address requires the creation of a new
Customer Gateway.

The Customer Gateway inside IP address should be configured on your tunnel
interface.

Outside IP Addresses:
  - Customer Gateway            : 20.185.83.40
  - Virtual Private Gateway      : 52.70.171.101
```

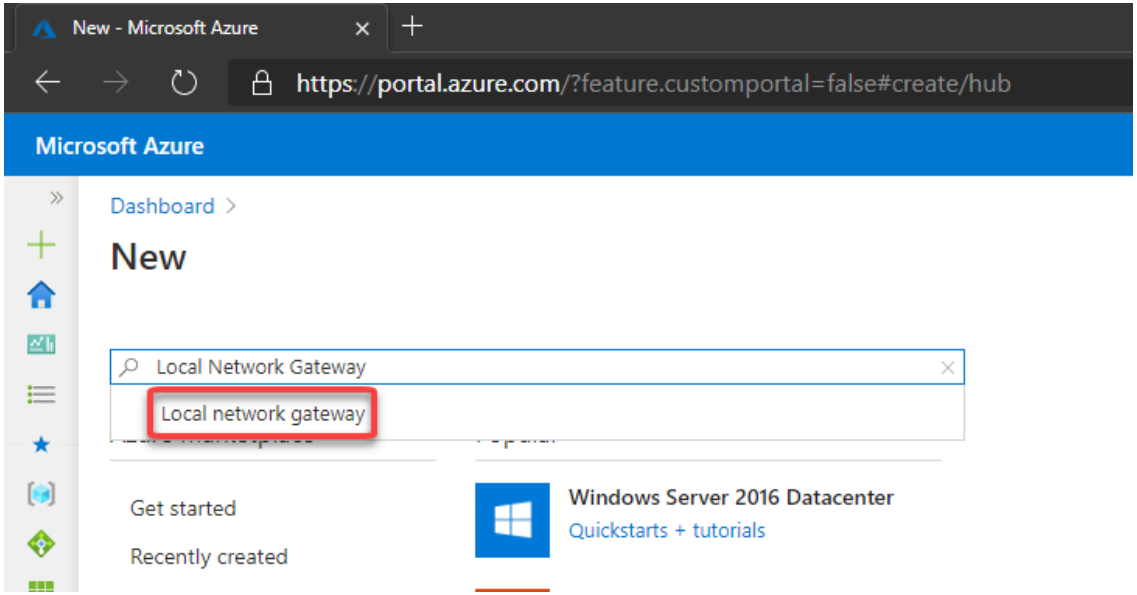
After the creation, you should have something like this:

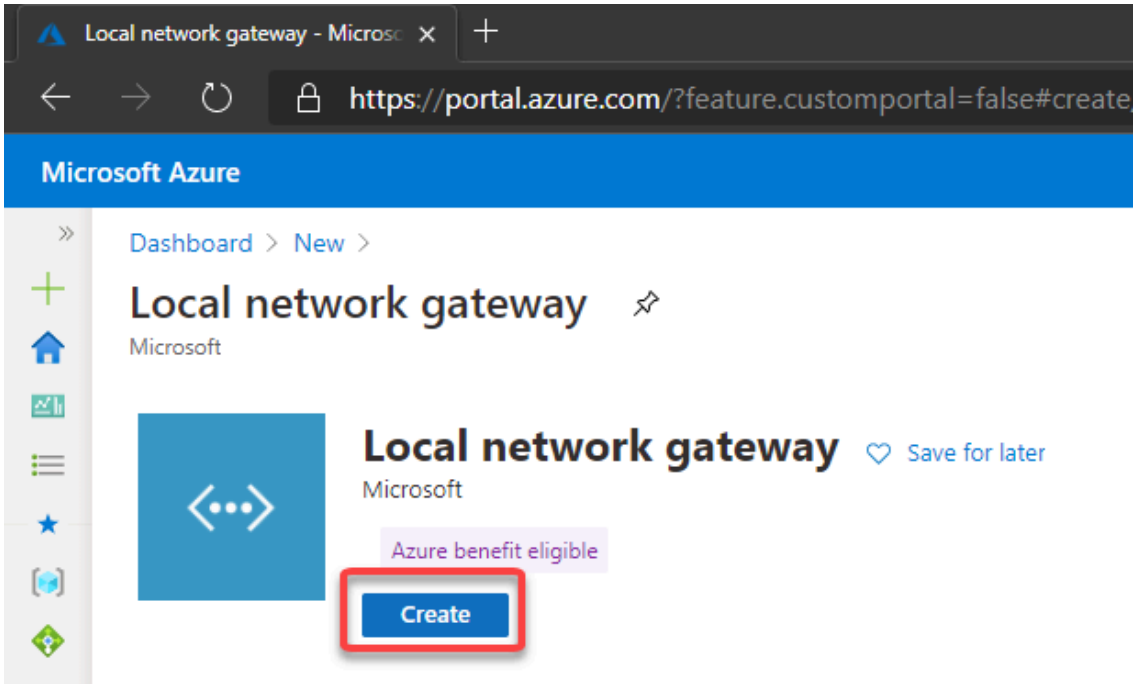


Adding the AWS information on Azure Configuration

10. Now let's create the Local Network Gateway

The Local Network Gateway is an Azure resource with information to Azure about the customer gateway device, in this case the AWS Virtual Private Gateway

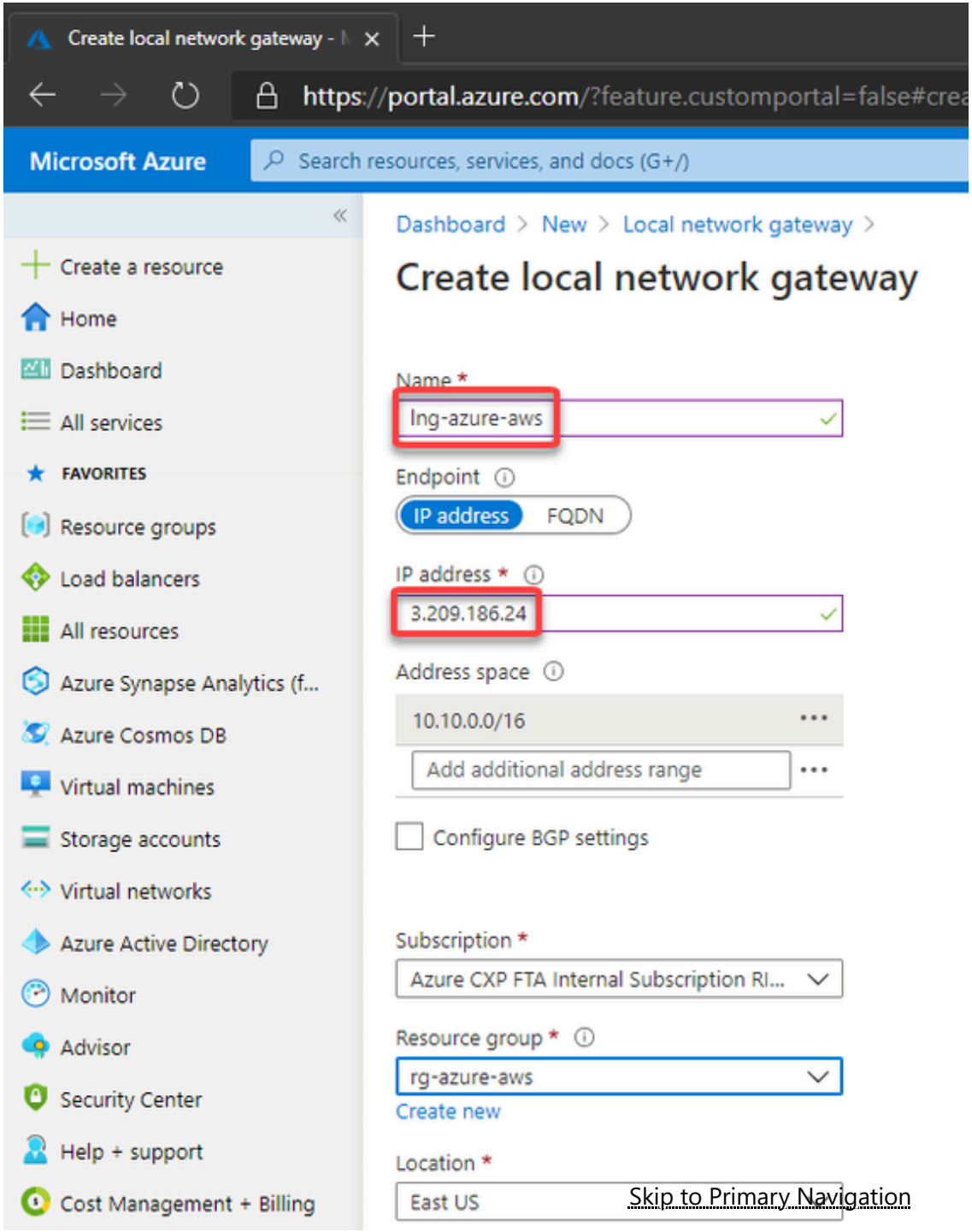

New - Microsoft Azure
https://portal.azure.com/?feature.customportal=false#create/hub
Microsoft Azure
Dashboard
New
Local Network Gateway
Local network gateway
Get started
Recently created
Windows Server 2016 Datacenter
Quickstarts + tutorials

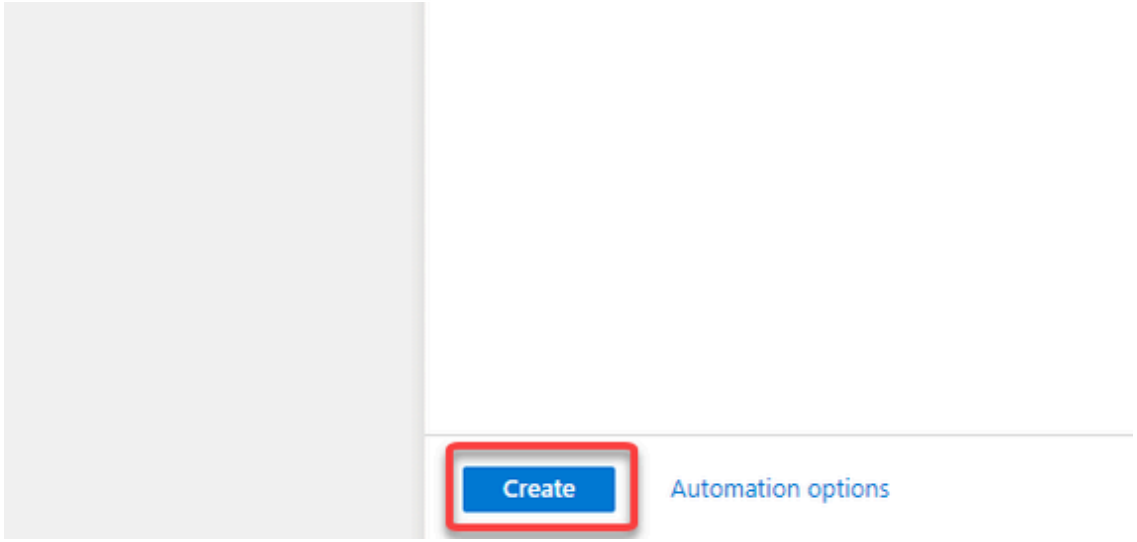

Local network gateway - Microsc
https://portal.azure.com/?feature.customportal=false#create
Microsoft Azure
Dashboard > New >
Local network gateway
Microsoft
Local network gateway
Save for later
Microsoft
Azure benefit eligible
Create

Now you need to specify the public ip address from the AWS Virtual Private Gateway and the VPC CIDR prefix.

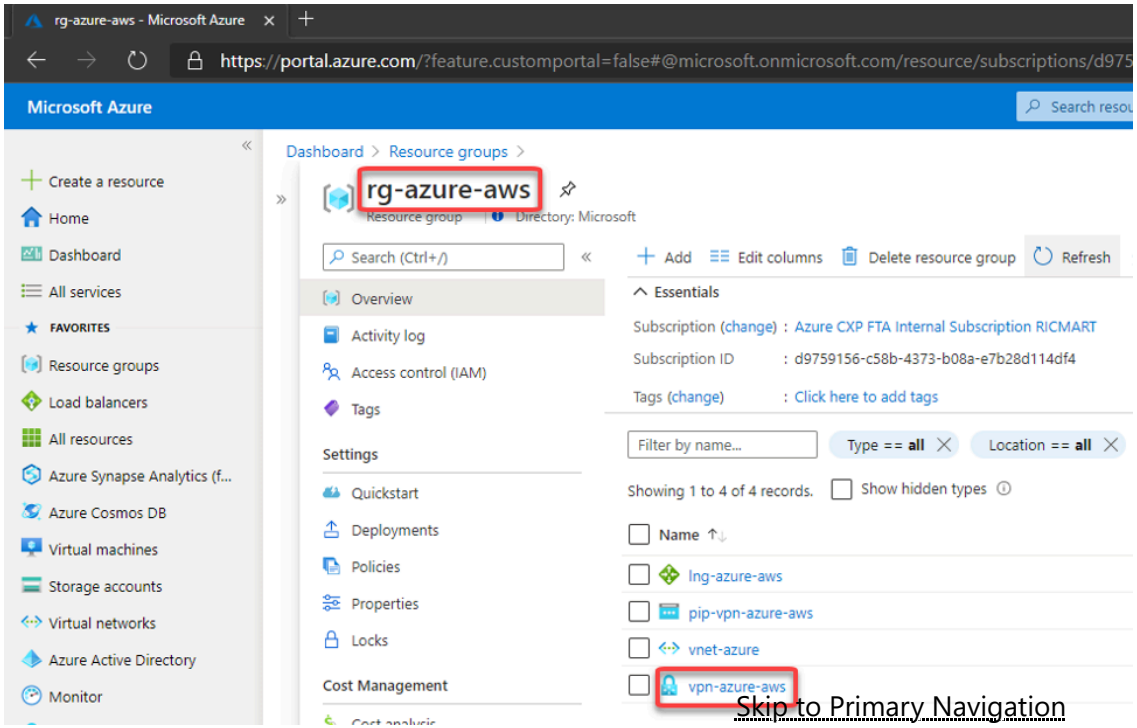
Please note that the public address from the AWS Virtual Private Gateway is described at the configuration file you have downloaded.

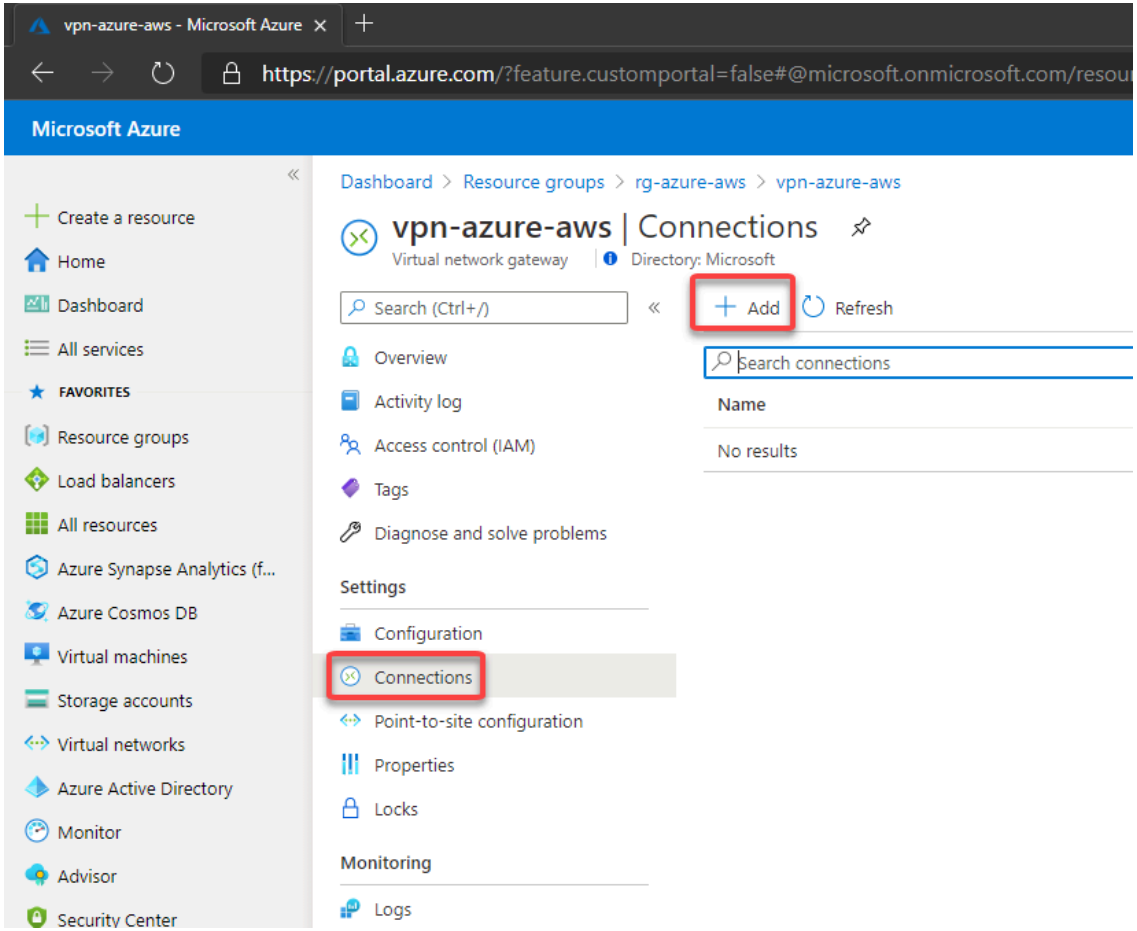
As mentioned earlier, AWS creates two IPSec tunnels to high availability purposes. I'll use the public ip address from the IPSec Tunnel #1 for now.


Create local network gateway -
https://portal.azure.com/?feature.customportal=false#crea
Microsoft Azure
Search resources, services, and docs (G+/)
Create a resource
Home
Dashboard
All services
FAVORITES
Resource groups
Load balancers
All resources
Azure Synapse Analytics (f...
Azure Cosmos DB
Virtual machines
Storage accounts
Virtual networks
Azure Active Directory
Monitor
Advisor
Security Center
Help + support
Cost Management + Billing
Dashboard > New > Local network gateway >
Create local network gateway
Name *
lng-azure-aws
Endpoint
IP address
FQDN
IP address *
3.209.186.24
Address space
10.10.0.0/16
Add additional address range
Configure BGP settings
Subscription *
Azure CXP FTA Internal Subscription RI...
Resource group *
rg-azure-aws
Create new
Location *
East US
Skip to Primary Navigation

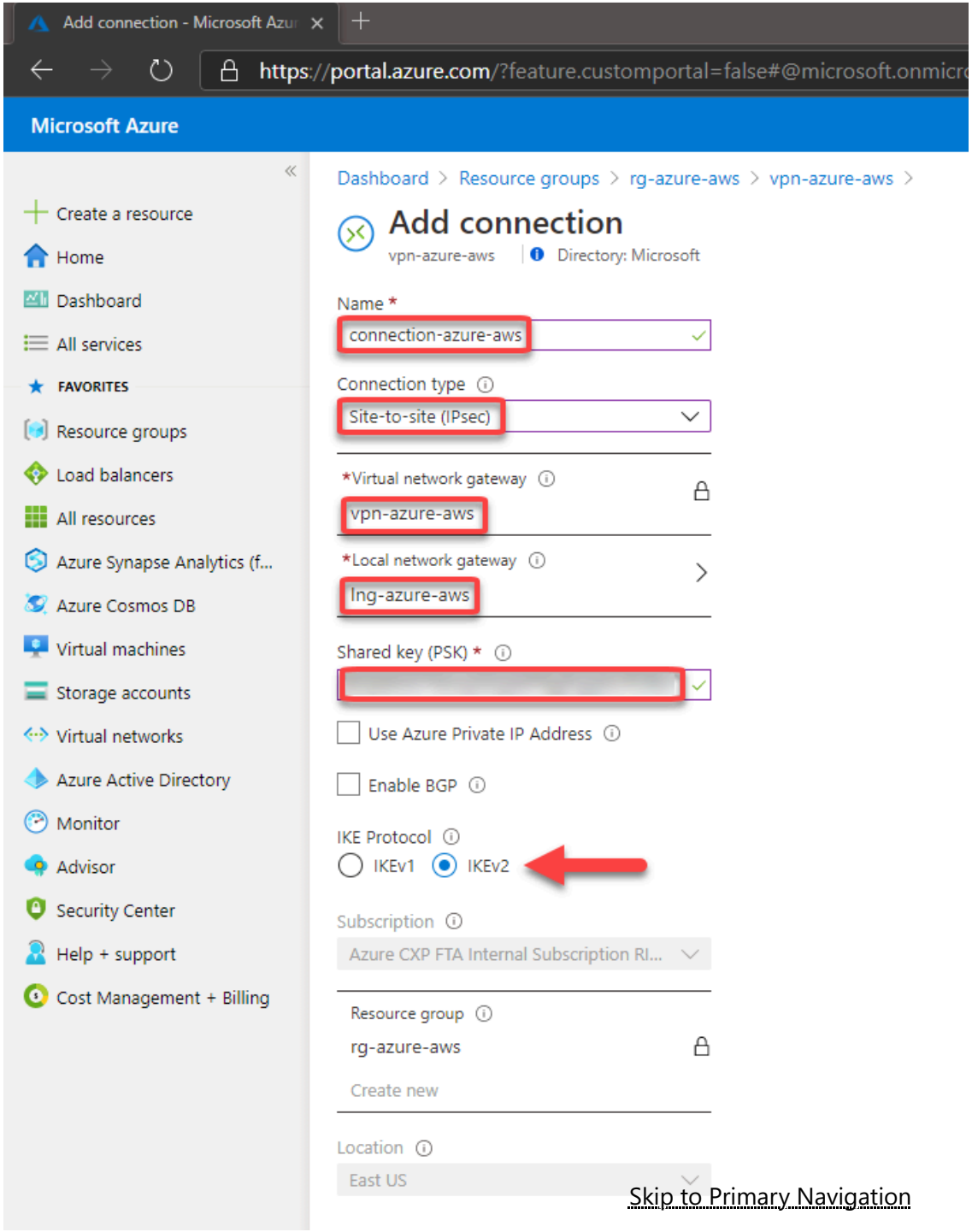


11. Then let's create the connection on the Virtual Network Gateway



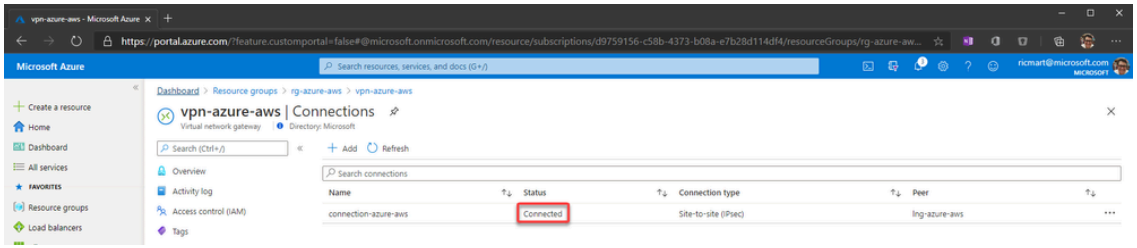


You should fill the fields according below. Please note that the Shared key was obtained at the configuration file downloaded earlier and In this case, I'm using the Shared Key for the Ipsec tunnel #1 created by AWS and described at the configuration file.

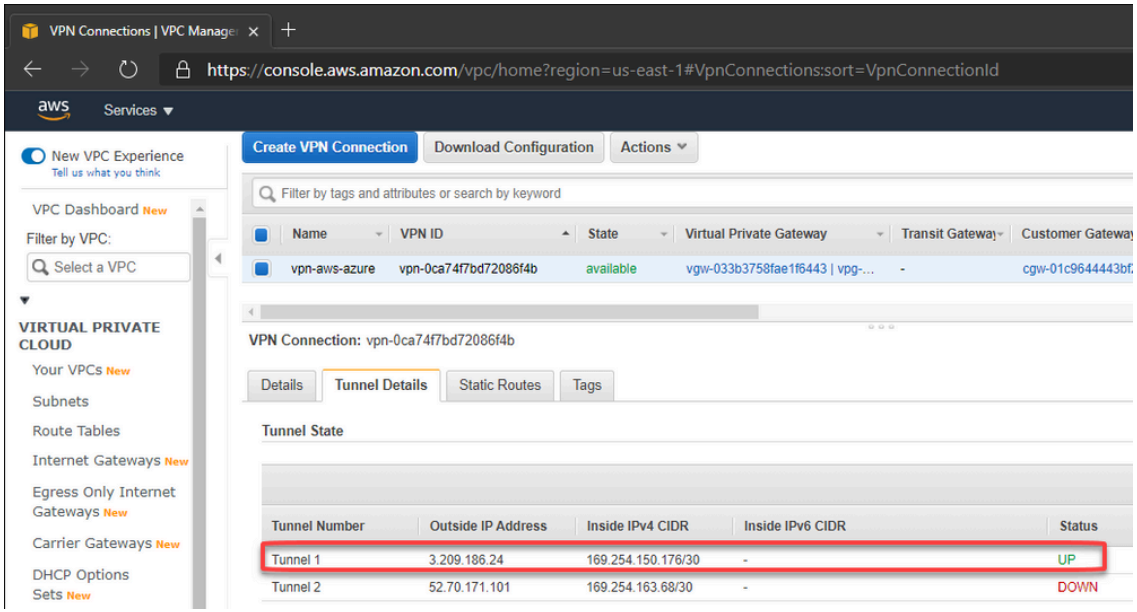


Skip to Primary Navigation

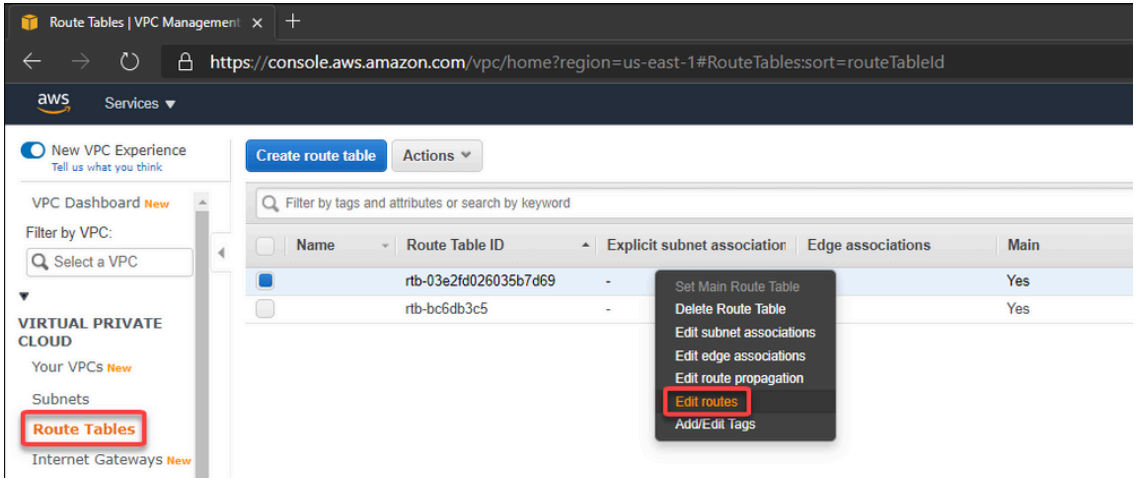
After a few minutes, you can see the connection established:



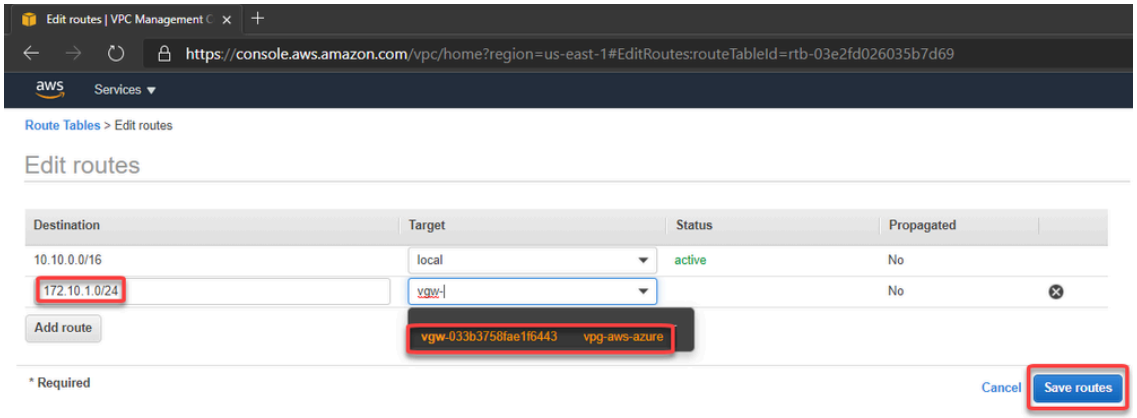
In the same way, we can check on AWS that the 1st tunnel is up:



Now let's edit the route table associated with our VPC

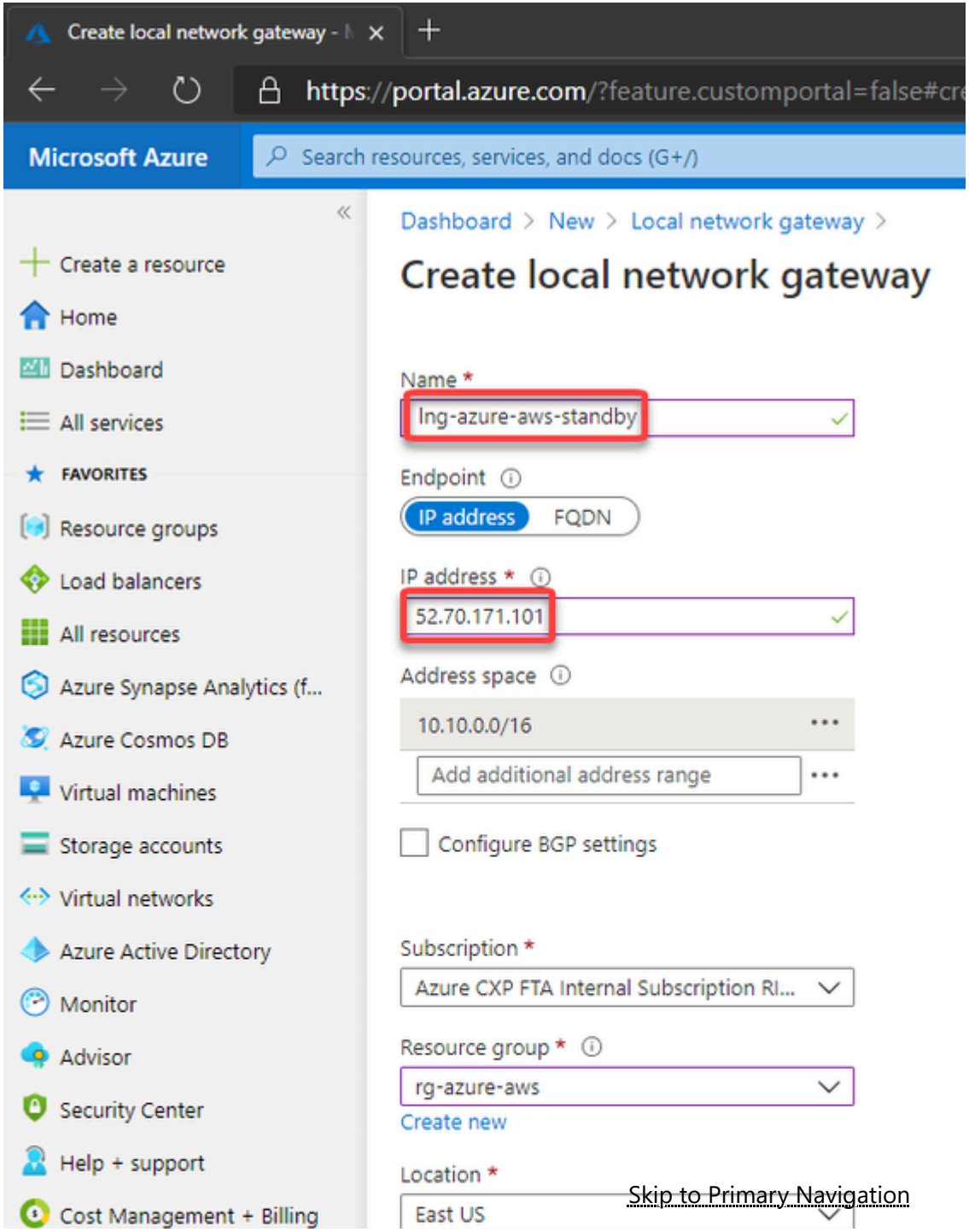


And add the route to Azure subnet through the Virtual Private Gateway:

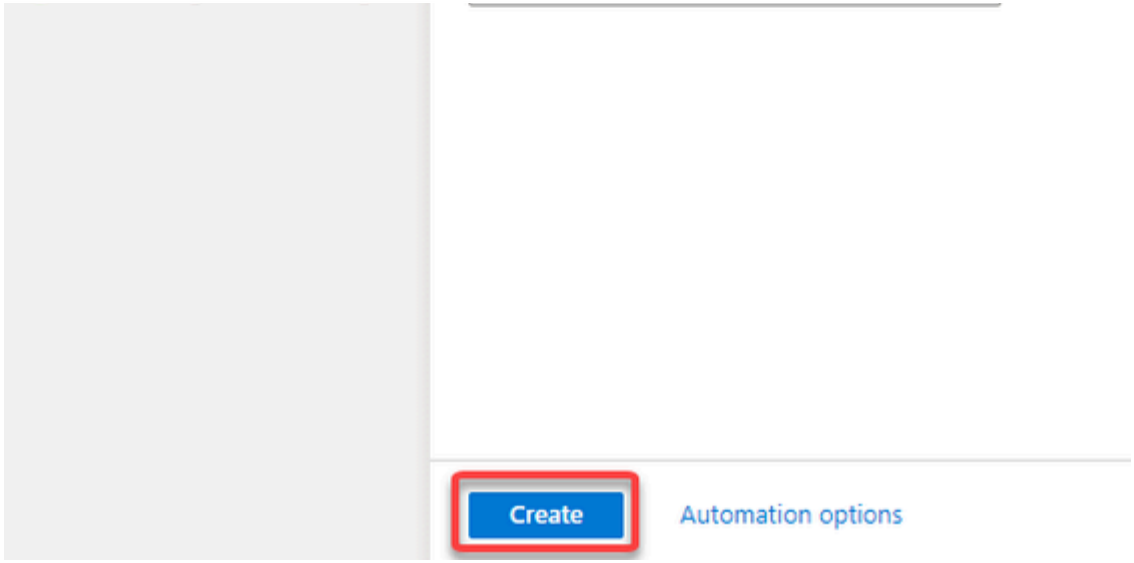


12. Adding high availability

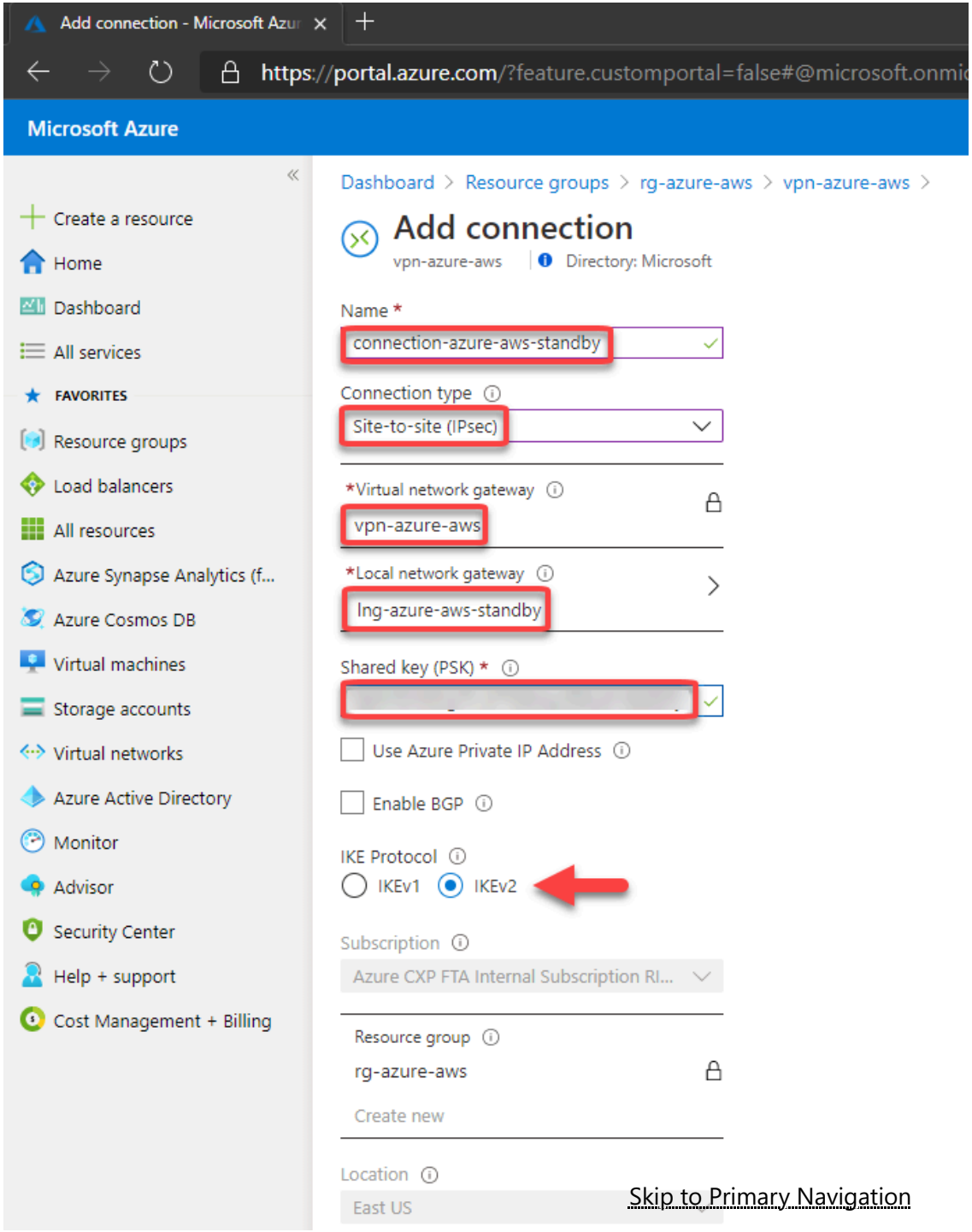
Now we can create a 2nd connection to ensure high availability. To do this let's create another Local Network Gateway which we will point to the public ip address of the IPSec tunnel #2 on the AWS



Skip to Primary Navigation

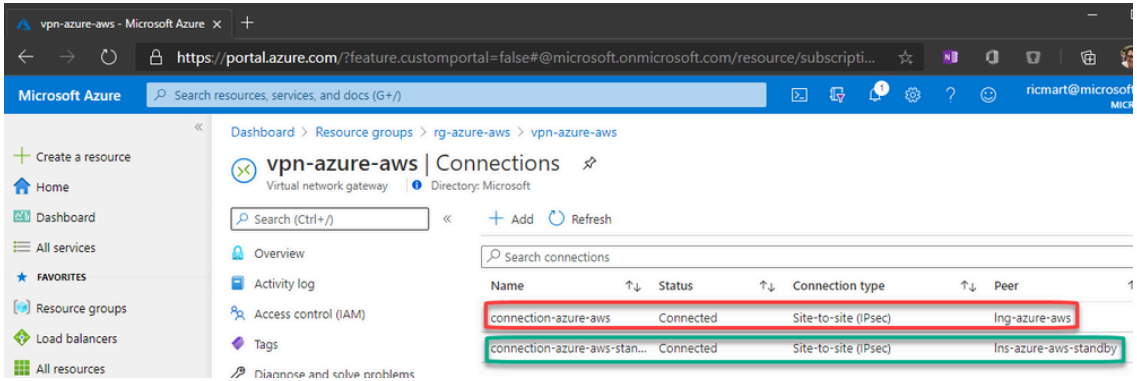


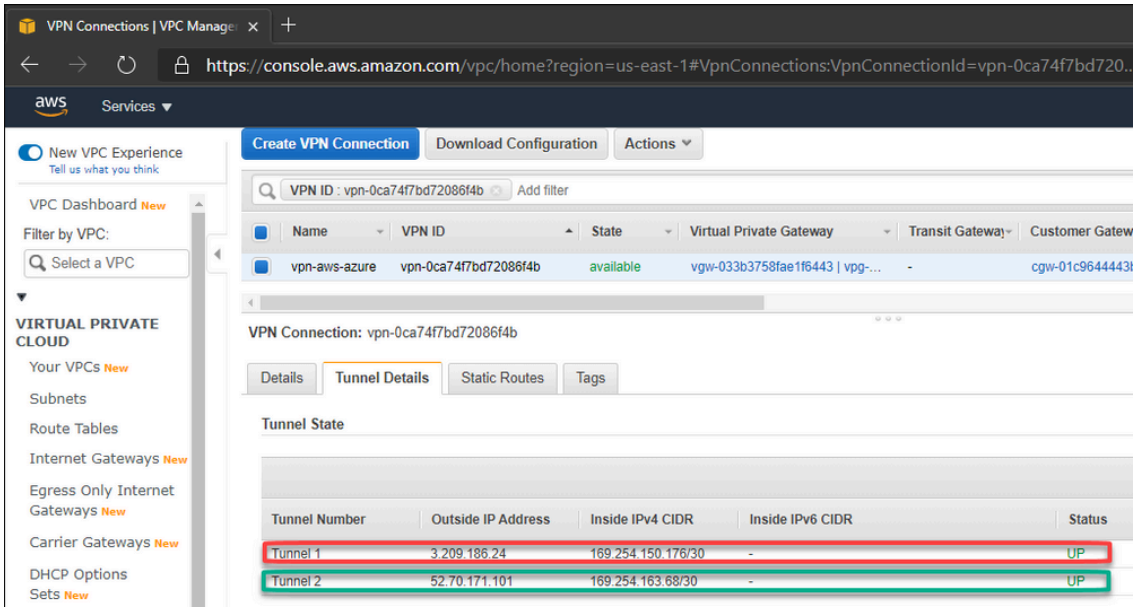
Then we can create the 2nd connection on the Virtual Network Gateway:



Skip to Primary Navigation

And in a few moments we'll have:



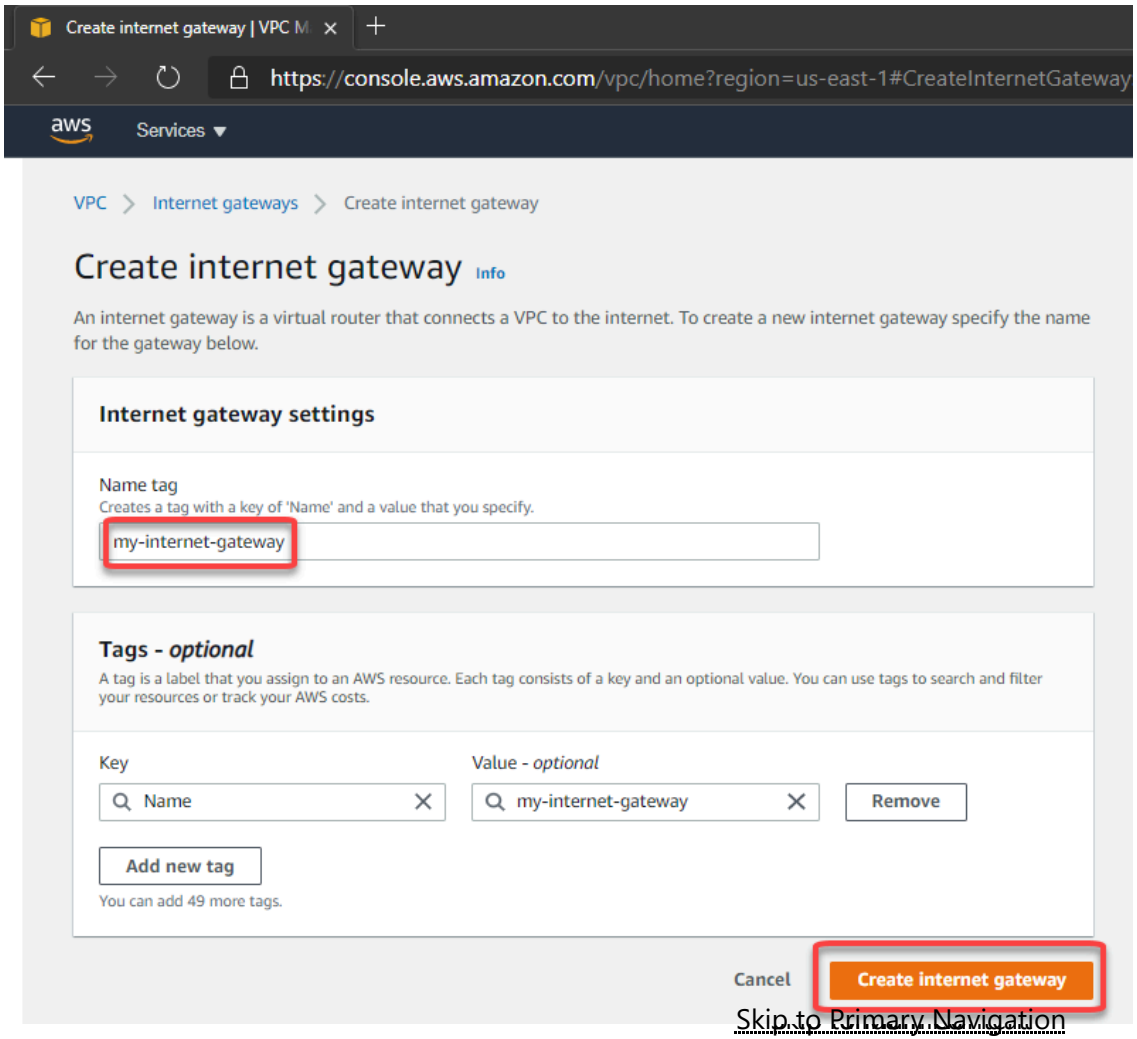


With this, our VPN connection is established on both sides and the work is done.

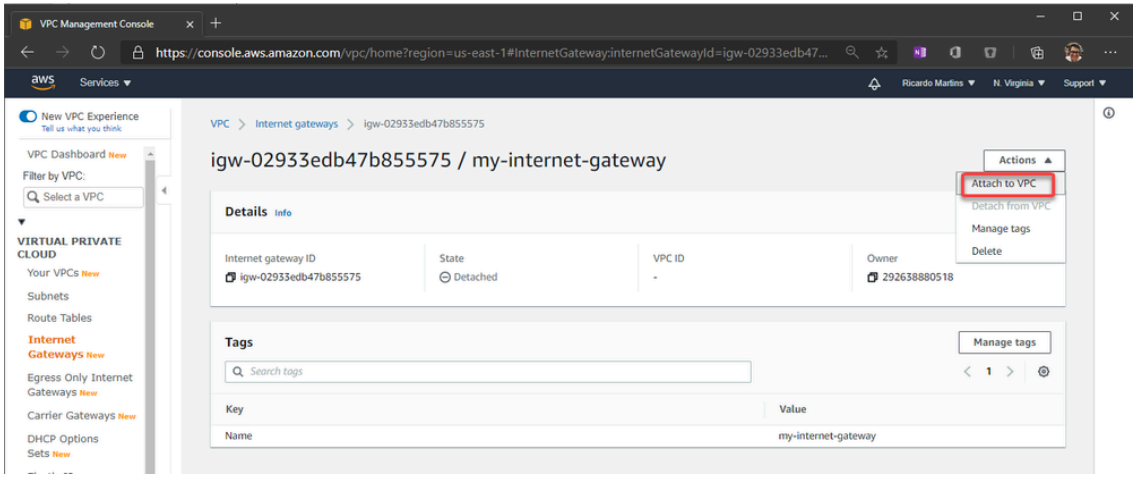
Skip to Main Navigation

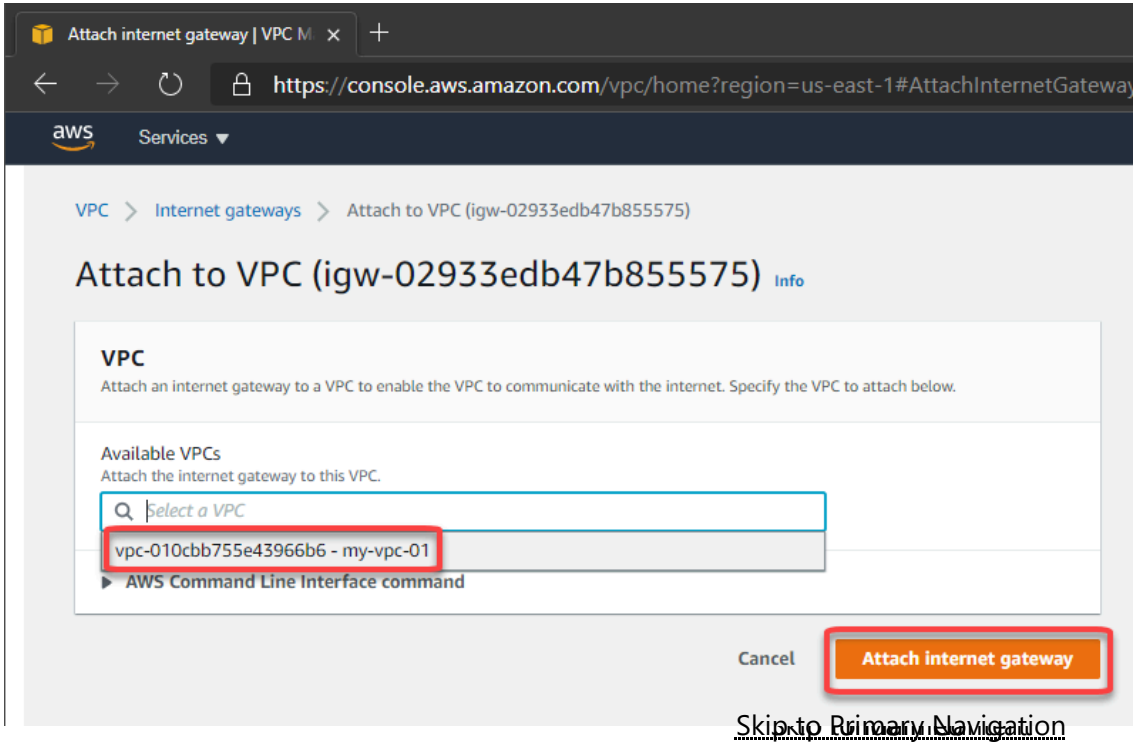
13. Let's test!

First, let's add an Internet Gateway to our VPC at AWS. The Internet Gateway is a logical connection between an Amazon VPN and the Internet. This resource will allow us to connect through the test VM from their public ip through internet. This is not required for the VPN connection, is just for our test:

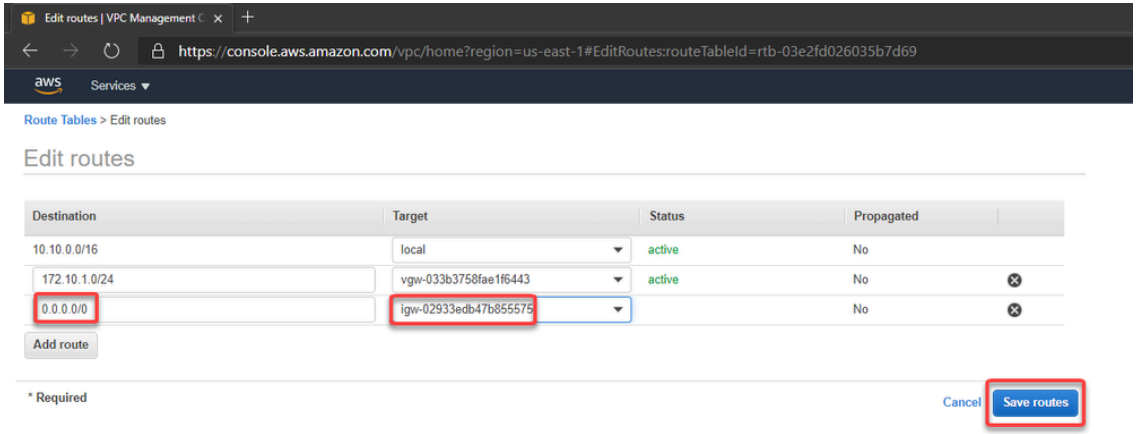


After create, let's attach to the VPC:

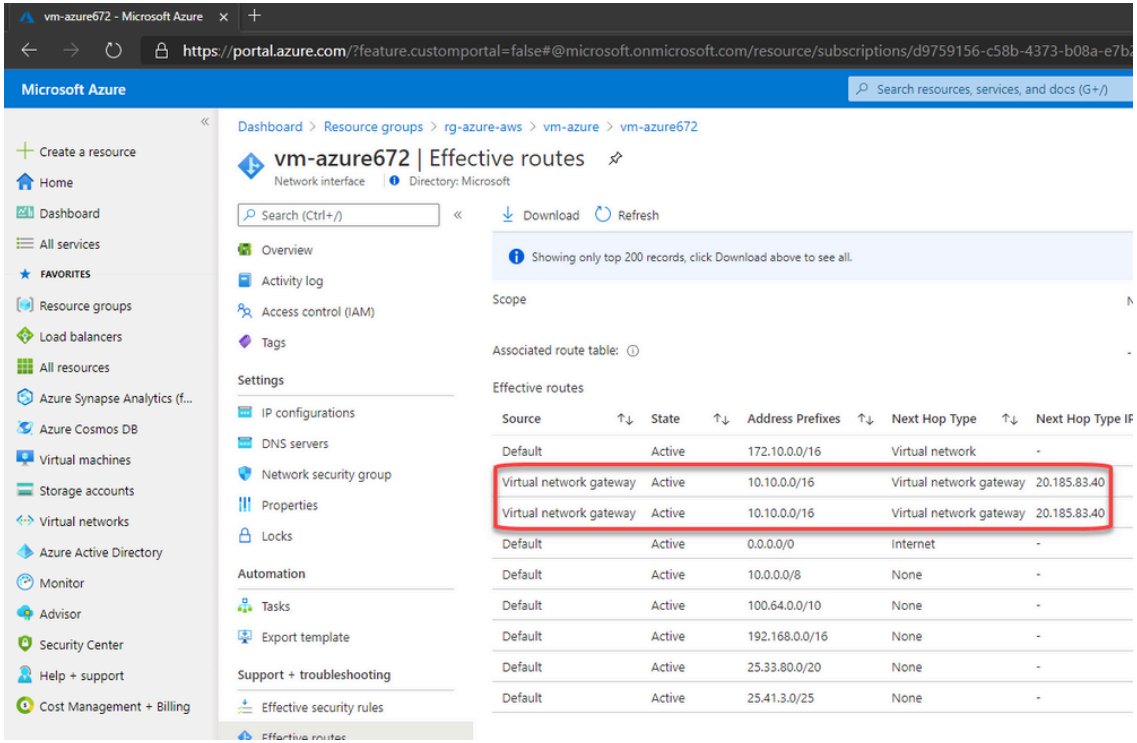




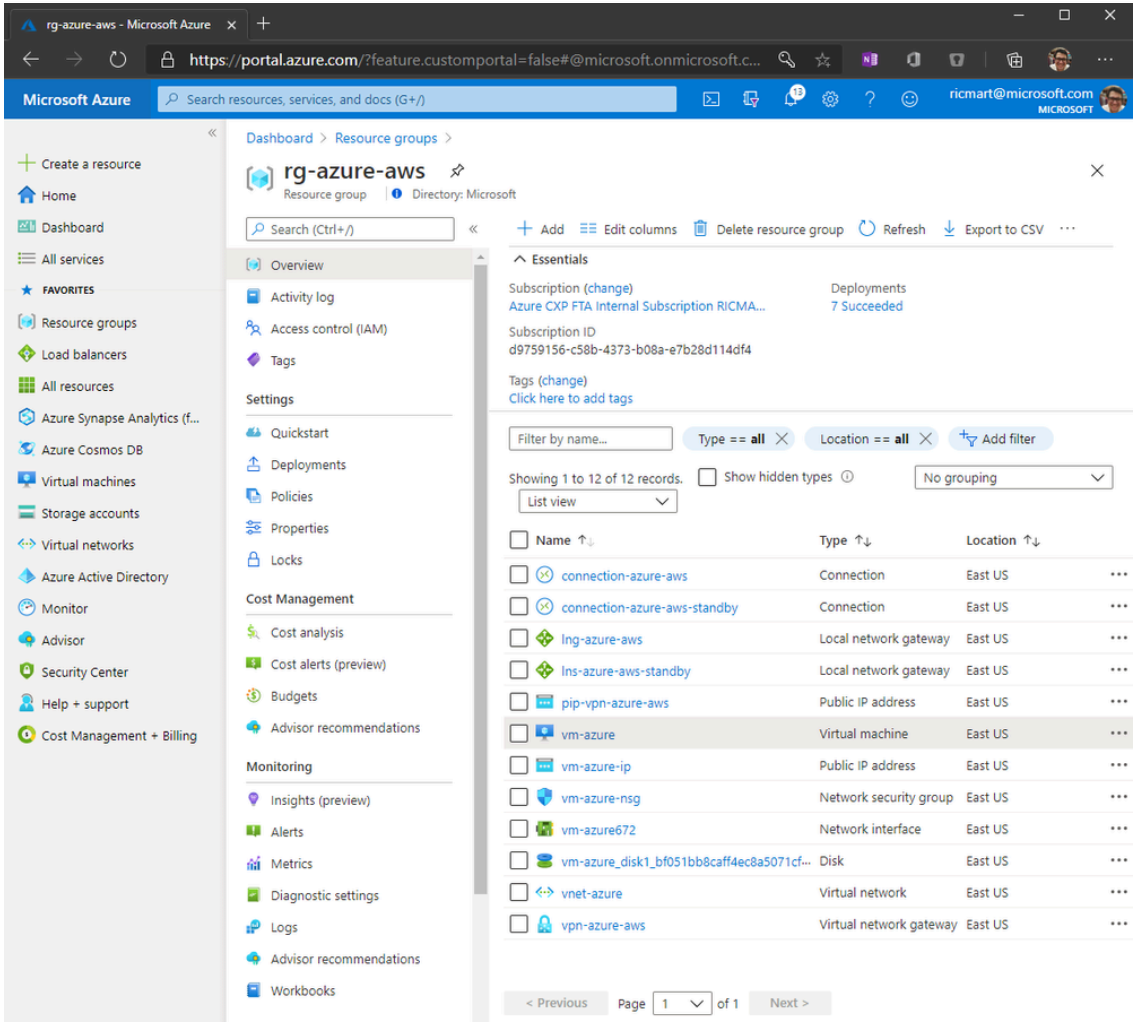
Now we can create a route to allow connections to **0.0.0.0/0** (Internet) through the Internet Gateway:



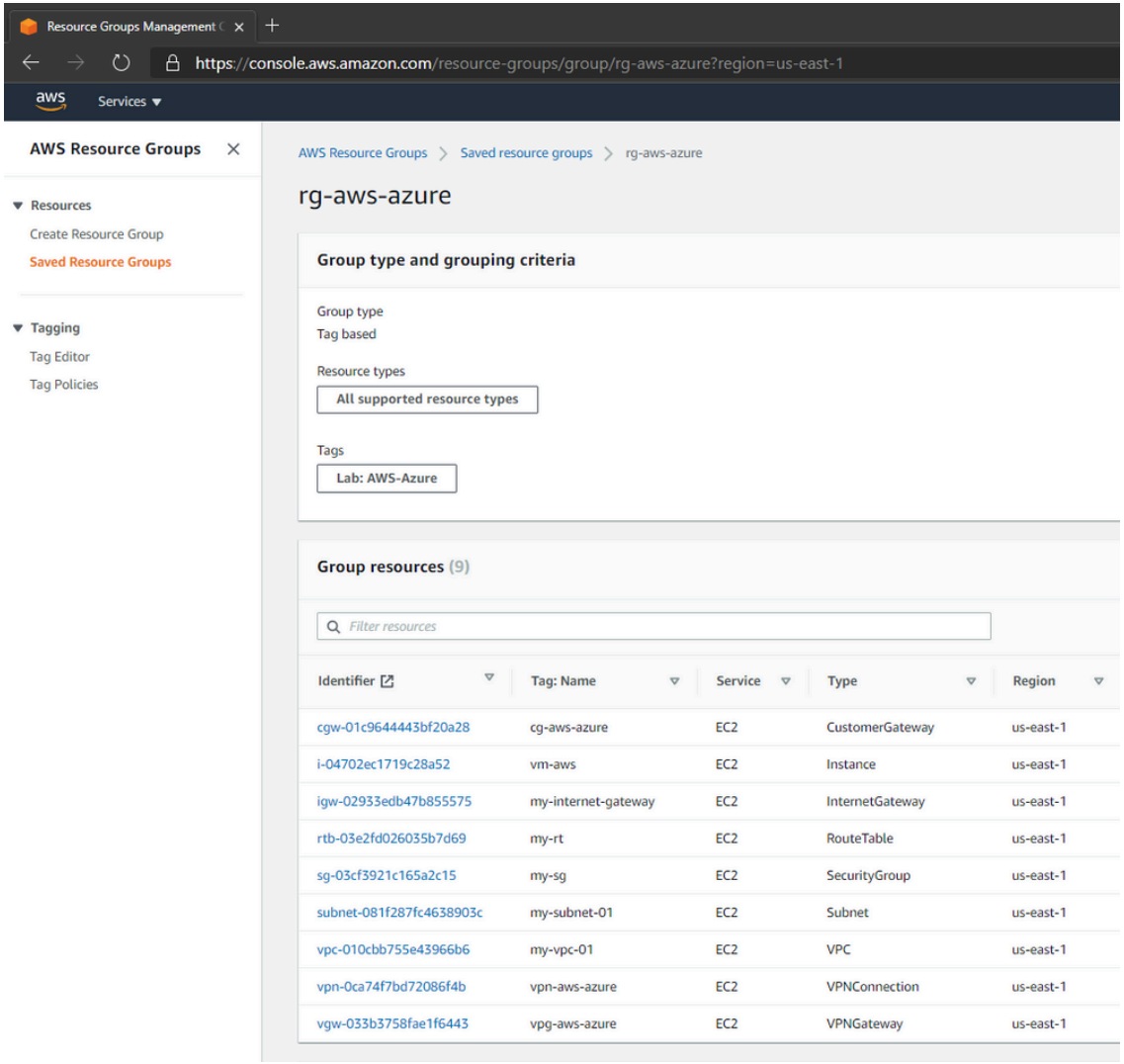
On Azure the route was automatically created. You can check selecting the Azure VM > Networking > Network Interface > Effective routes. Note that we have 2 (1 per connection):



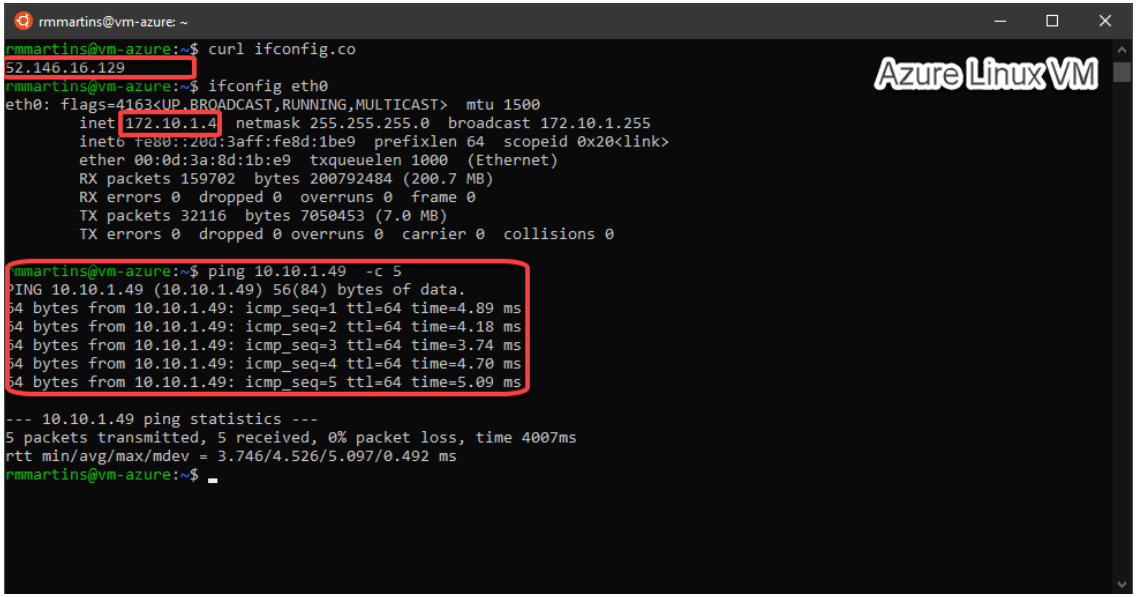
Now I've created a Linux VM on Azure and our environment looks like this:

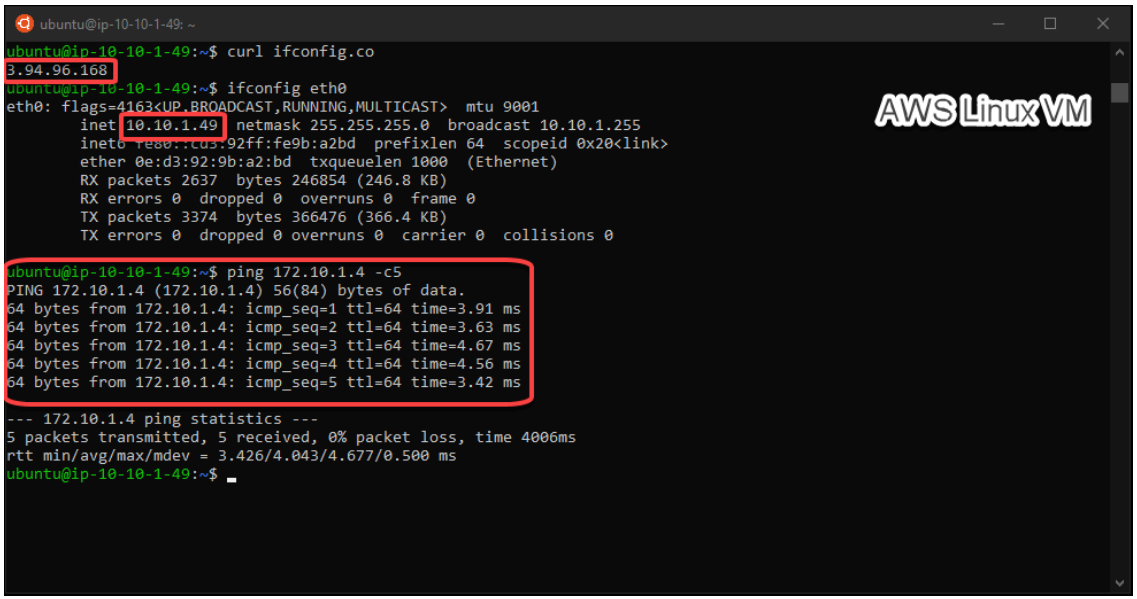


And I did the same VM creation on AWS that looks like this:



Then we can test the connectivity betweeen Azure and AWS through our VPN connection:


rmmartins@vm-azure: ~
rmmartins@vm-azure:~$ curl ifconfig.co
52.146.16.129
rmmartins@vm-azure:~$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 172.10.1.4 netmask 255.255.255.0 broadcast 172.10.1.255
ether 00:0d:3a:8d:1b:e9 txqueuelen 1000 (Ethernet)
RX packets 159702 bytes 200792484 (200.7 MB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 32116 bytes 7050453 (7.0 MB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
rmmartins@vm-azure:~$ ping 10.10.1.49 -c 5
PING 10.10.1.49 (10.10.1.49) 56(84) bytes of data.
64 bytes from 10.10.1.49: icmp_seq=1 ttl=64 time=4.89 ms
64 bytes from 10.10.1.49: icmp_seq=2 ttl=64 time=4.18 ms
64 bytes from 10.10.1.49: icmp_seq=3 ttl=64 time=3.74 ms
64 bytes from 10.10.1.49: icmp_seq=4 ttl=64 time=4.70 ms
64 bytes from 10.10.1.49: icmp_seq=5 ttl=64 time=5.09 ms
--- 10.10.1.49 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 3.746/4.526/5.097/0.492 ms
rmmartins@vm-azure:~$
Azure Linux VM


ubuntu@ip-10-10-1-49: ~
ubuntu@ip-10-10-1-49:~$ curl ifconfig.co
3.94.96.168
ubuntu@ip-10-10-1-49:~$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet 10.10.1.49 netmask 255.255.255.0 broadcast 10.10.1.255
ether 0e:d3:92:9b:a2:bd txqueuelen 1000 (Ethernet)
RX packets 2637 bytes 246854 (246.8 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 3374 bytes 366476 (366.4 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
ubuntu@ip-10-10-1-49:~$ ping 172.10.1.4 -c5
PING 172.10.1.4 (172.10.1.4) 56(84) bytes of data.
64 bytes from 172.10.1.4: icmp_seq=1 ttl=64 time=3.91 ms
64 bytes from 172.10.1.4: icmp_seq=2 ttl=64 time=3.63 ms
64 bytes from 172.10.1.4: icmp_seq=3 ttl=64 time=4.67 ms
64 bytes from 172.10.1.4: icmp_seq=4 ttl=64 time=4.56 ms
64 bytes from 172.10.1.4: icmp_seq=5 ttl=64 time=3.42 ms
--- 172.10.1.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 3.426/4.043/4.677/0.500 ms
ubuntu@ip-10-10-1-49:~$
AWS Linux VM

Update: If you want to deploy a solution more resilient and using BGP, follow this documentation: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-aws-bgp

4 Likes

20 Comments

Néstor Reverón Iron Contributor

Jul 12 2021 05:21 AM

Thanks @rmmartins

0 Likes

Chandu_P Copper Contributor

Aug 01 2021 03:04 AM

@rmmartins
Thank you very much for the steps.
I need some high level guidance for our use-case, if you don't mind please.
Ours is a early stage startup company. We are trying to setup secure employees login/connections to our AWS environment.
We have all our 10 employees using office365. We do have Azure subscription too.
How could I setup this use-case:-

Create Azure VNet Gateway and Azure VPN (I think we know steps for this).
Create a Virtual Desktop (either Ubuntu or Windows, in AWS or within Azure) for Multi-User session mode.
All our employees should login to Azure VPN client on their own personal laptops using Azure AD(O365) login; after that employees should login to the Virtual Desktop using SSO via Azure AD. After logging in to Virtual Desktop only our engineers should be able to connect to our AWS resources like AWS EKS or AWS RDS or anything which is in our AWS private subnet using AWS SSO via Azure AD.

Please provide some high level steps or point me to some resources which could help. Please & Thank you.

1 Like

Fabio_Jourdan Copper Contributor
Aug 19 2021 06:58 AM

@Chandu_P , use Azure AD Aplication Proxy to publish Private Web Applications, without using VPN or connections between Azure and AWS, you only need to deploy a proxy Machine in you AWS VPC with Internet Connection.

For AWS administrative tasks, in management console, use AWS SSO + Azure AD

1 Like

microworker Copper Contributor

Dec 15 2021 09:09 AM

During step 11 while creating the connection in Azure, I'm getting the "Unknown" status on the connection. I'm getting something similar in AWS. Configuration seems like it is correct, although clearly I'm doing something wrong. Does anyone know what causes the "Unknown" status? I'm not seeing any clear documentation on it anywhere.

0 Likes

stefanalex2360 Copper Contributor

Dec 21 2021 09:27 AM

hi microworker

check public ip addresses in both ends of the tunnel and secret key again.

And wait. sometimes i wait for like 5 to 10 minutes for tunnels to go up, sometimes it is almost instant

0 Likes

microworker Copper Contributor

Dec 21 2021 01:02 PM

@stefanalex2360

I did do that, but they were definitely correct. I did upgrade the gateway from vpngw1 to vpngw2 and tunnel 1 immediately came up. But now I can't get the second tunnel up.

0 Likes

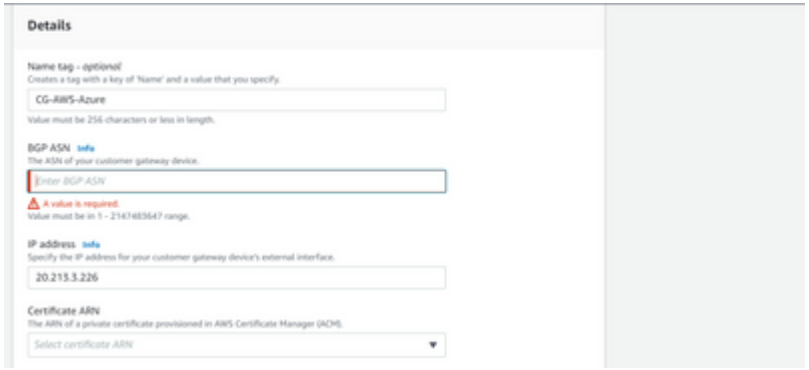
Rajarshi1590 Copper Contributor

May 09 2022 07:34 PM

Hi,

Thank you for the nice step-by-step article! I just tried the steps mentioned in the article. So, from the Azure side when we are creating a virtual network gateway, we are disabling BGP. But, from AWS side when we are creating a customer gateway with the public IP of the Azure Virtual network gateway, we cannot disable BGP. If BPP related text field is left blank, then it is asking me to enter something, as shown in the screenshot below. So, It cannot be left blank. Not sure how BGP did not appear in the screenshot that you provided for configuring CGW in AWS.

Now, if I enable BGP (by providing a BGP ASN as 65001) then will it be able to communicate with Azure once the Virtual private gateway is set up and attached to the corresponding VPC? If not, then how to make the connectivity work?



0 Likes

jdewitt66 Copper Contributor

Aug 02 2022 02:15 PM

Hello,

Bottom line up front: Need assistance moving DB from AWS to Azure

Thanks for the article above. I am working on a project in which we are trying to transfer several existing PostgreSQL DB from AWS to Azure. We have followed all directions that we can find online to use the Azure DMS but we cannot establish a successful connection between the DMS and AWS. The problem occurs on the 'Select Source' page of the Azure DMS. During the validation step I get an error stating 'Failed to connect'. I have been working on this for a few weeks trying different solutions. I get a different error if I purposely enter incorrect source info, so I believe that Azure is talking to AWS, but not creating the required connection.

I followed all of the steps in the above article and can see that both AWS tunnels are up and the connection in Azure shows connected. However, I still can't get a connection to allow transfer of the DBs from AWS to Azure. Can anyone help?

0 Likes

AlexT330 Copper Contributor

Dec 04 2022 06:58 PM

In your examples you use **172.10.0.0/16**, but those are public IP addresses assigned to AT&T. Configuring them on a private subnet is likely to cause connectivity issues with parts of the internet. Recently some members of my team configured a VPN

using those values copied from your tutorial and now I'm asking them to redo their work to use private address space instead.
May I suggest switching your tutorial to **172.16.0.0/16** which is an appropriate private space instead.

0 Likes

Anoop Singh Copper Contributor

Apr 05 2023 01:59 PM

It's very precise and clear content. Loved it.
I have successfully completed my use case for VPN connectivity with high availability between Azure and AWS cloud.
Thanks Once again for your great content.

Thanks,
Anoop Singh

1 Like

mileytores Copper Contributor

May 05 2023 03:30 PM

Muchas gracias! funcionó a la perfección, pero si tuve que cambiar a sku VpnGW2

1 Like

sonali335 Copper Contributor

Jun 08 2023 07:52 AM

I did all the setup as mentioned in above step my tunnel is also up and vpn is also connected.

but when im doing the ping from aws to azure and azure to aws its not working
showing

--- 10.34.0.4 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2052ms

0 Likes

vsk2023 Copper Contributor

Oct 21 2023 10:36 PM

Hi @sonali335 ,
Could you pls try the below

In AWS side of the config, in the Site-to-Site VPN connections, under "Static routes" add the IP prefix of the Azure VM i.e.

172.10.1.0/24
Also ensure the route propagation is enabled in the route table config.

Hope this helps.

1 Like

David2075 Copper Contributor
Dec 11 2023 03:10 PM

Saw this after many hours of pulling my hair out...."Also ensure the route propagation is enabled in the route table config."
After enabling, my issues were resolved.
FYI: Great article !

0 Likes

mshah1525 Copper Contributor
Jan 18 2024 12:13 PM

I am able to ssh from AWS to Azure, but not from Azure to AWS. On both end, the tunnel connectivity shows as connected.
Also if we plan to add another Azure region, is there a way to use existing tunnel or do I need to setup another tunnel between Azure East to AWS.

0 Likes

Dan1232275 Copper Contributor

Feb 07 2024 04:47 PM

@rmmartins @mshah1525 Were you able to resolve it? I am having same issue.
I can ssh from AWS to Azure but not the other way.
I can ping both ways.

0 Likes

Dan1232275 Copper Contributor

Feb 12 2024 10:56 PM

If your GatewaySubnet have a route table, ensure Propagate gateway routes is set to true for the gateway.

1 Like

mshah1525 Copper Contributor

Feb 20 2024 11:39 AM

Yes, we were configuring the route table and hence the "Propagate gateway routes" which the VPN was configuring were invalidated. Once we deleted our routing configuration, it started working fine.

We see the VPN routing configuration, once we create a VM and look at the Network config. Is there any other place we can see all the routing configuration?

0 Likes

jlewis Copper Contributor

Mar 28 2024 07:06 AM

@rmmartins

Thank you very much for the steps.

0 Likes

venkatesh00hotmailco Copper Contributor

Jul 02 2024 01:27 PM

Super helpful!!

0 Likes

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

Comment

What's new

Surface Pro 9

Surface Laptop 5

Surface Studio 2+

Surface Laptop Go 2

Surface Laptop Studio

Surface Duo 2

Microsoft 365

Windows 11 apps

Microsoft Store

Account profile

Download Center

Microsoft Store support

Returns

Order tracking

Virtual workshops and training

Microsoft Store Promise

Flexible Payments

Education

Skip to main navigation Skip to footer navigation

Microsoft in education

Devices for education

Microsoft Teams for Education

Microsoft 365 Education

Education consultation appointment

Educator training and development

Deals for students and parents

Azure for students

Business

Microsoft Cloud

Microsoft Security

Dynamics 365

Microsoft 365

Microsoft Power Platform

Microsoft Teams

Microsoft Industry

Small Business

Developer & IT

Azure

Developer Center

Documentation

Microsoft Learn

Microsoft Tech Community

Azure Marketplace

AppSource

Visual Studio

Company

Careers

About Microsoft

Company news

Privacy at Microsoft

Investors

Diversity and inclusion

Accessibility

Sustainability

Your Privacy Choices