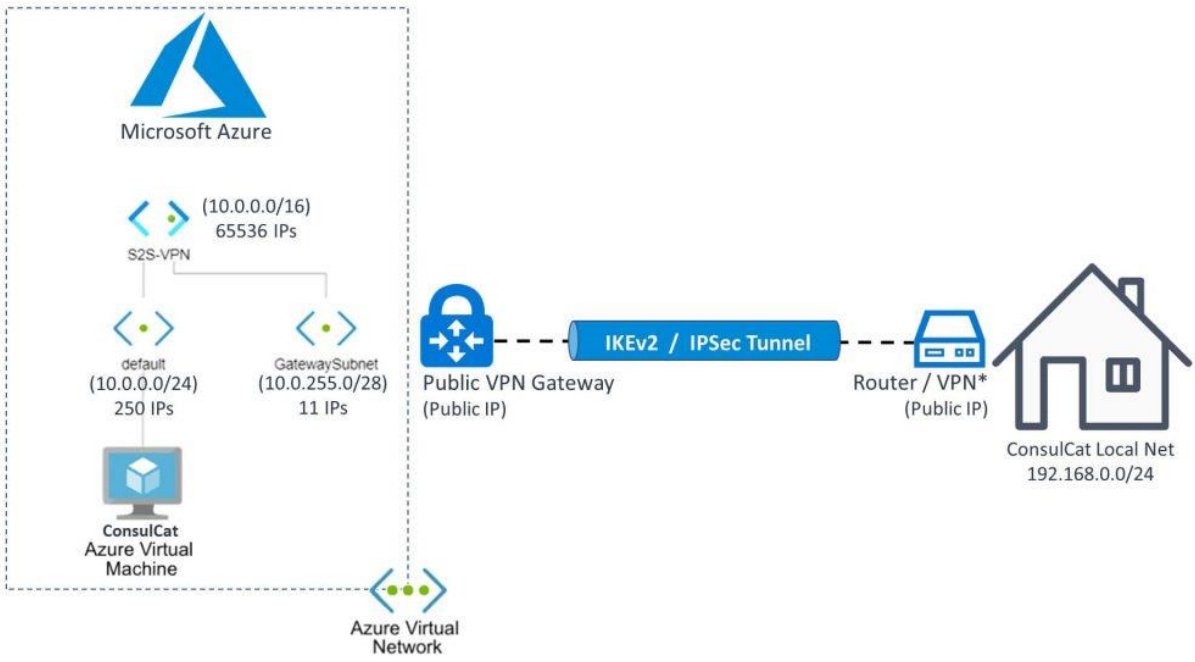
SITE TO SITE VPN CONNECTION

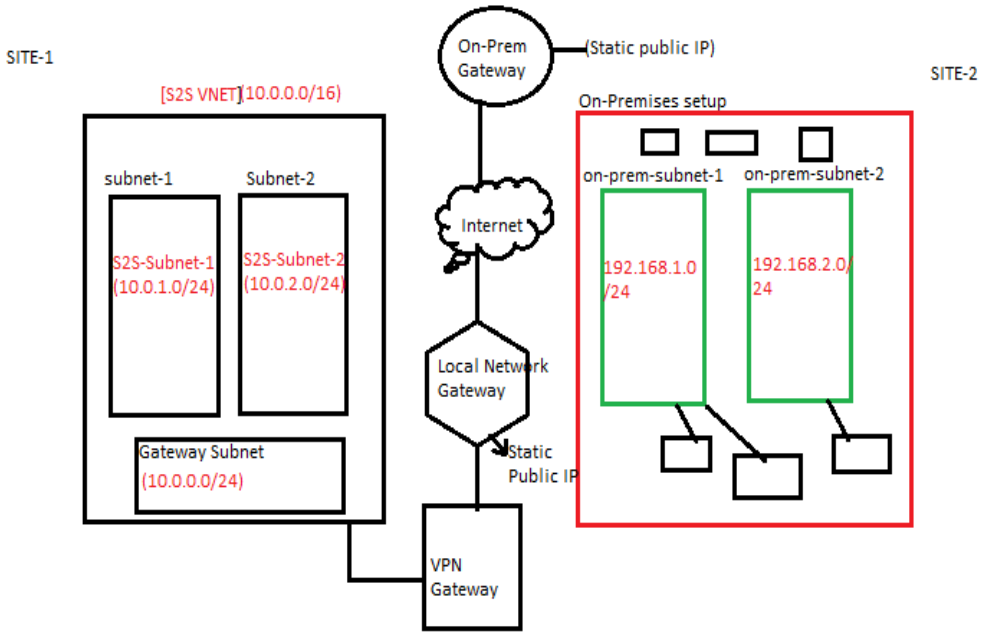


A Site-to-Site VPN gateway connection is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it.

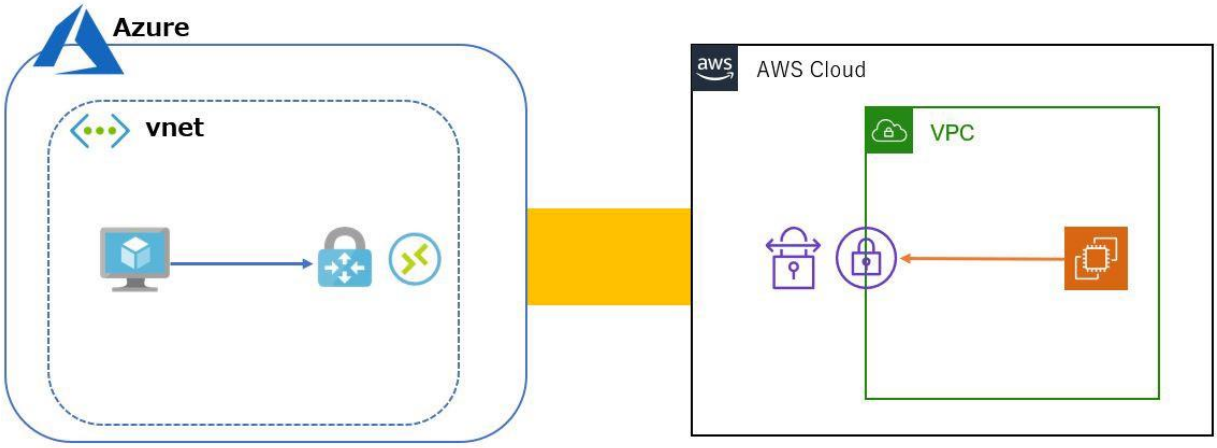
About the gateway subnet

The virtual network gateway uses specific subnet called the gateway subnet. The gateway subnet is part of the virtual network IP address range that you specify when configuring your virtual network. It contains the IP addresses that the virtual network gateway resources and services use. The subnet must be named 'GatewaySubnet' in order for Azure to deploy the gateway resources. You can't specify a different subnet to deploy the gateway resources to. If you don't have a subnet named 'GatewaySubnet', when you create your VPN gateway, it will fail.

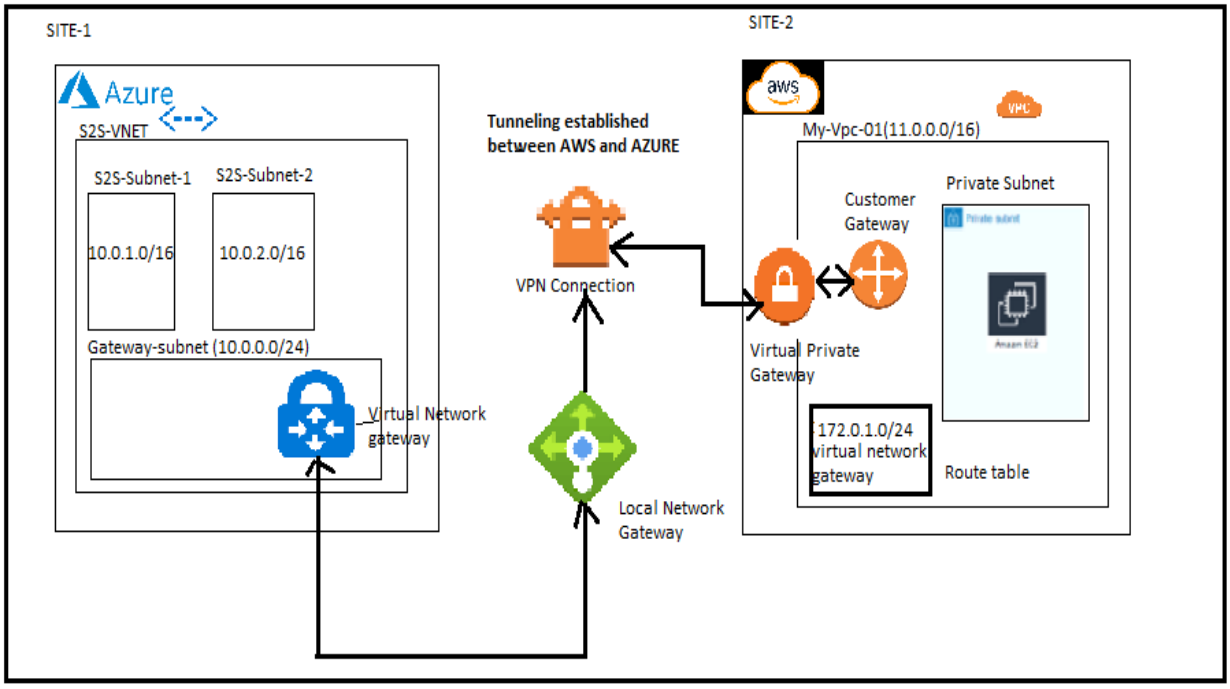
Project Deployment Architecture:



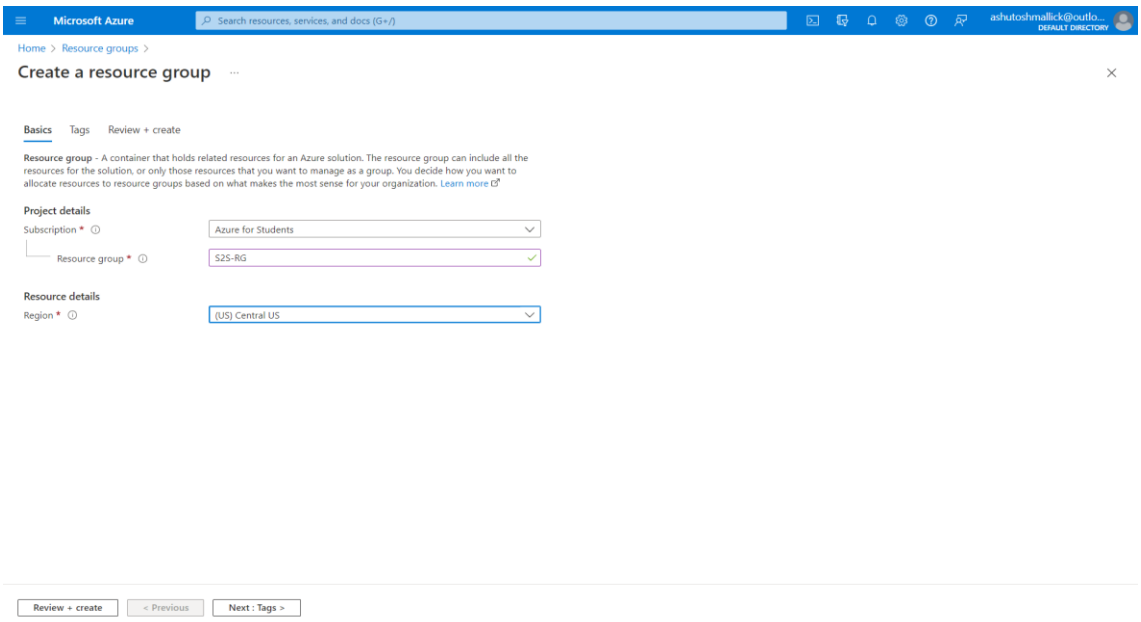
SITE TO SITE VPN CONNECTION BETWEEN AZURE AND AWS:



PROJECT DEPLOYMENT ARCHITECTURE(S2S CONNECTIVITY BETWEEN AWS AND AZURE):

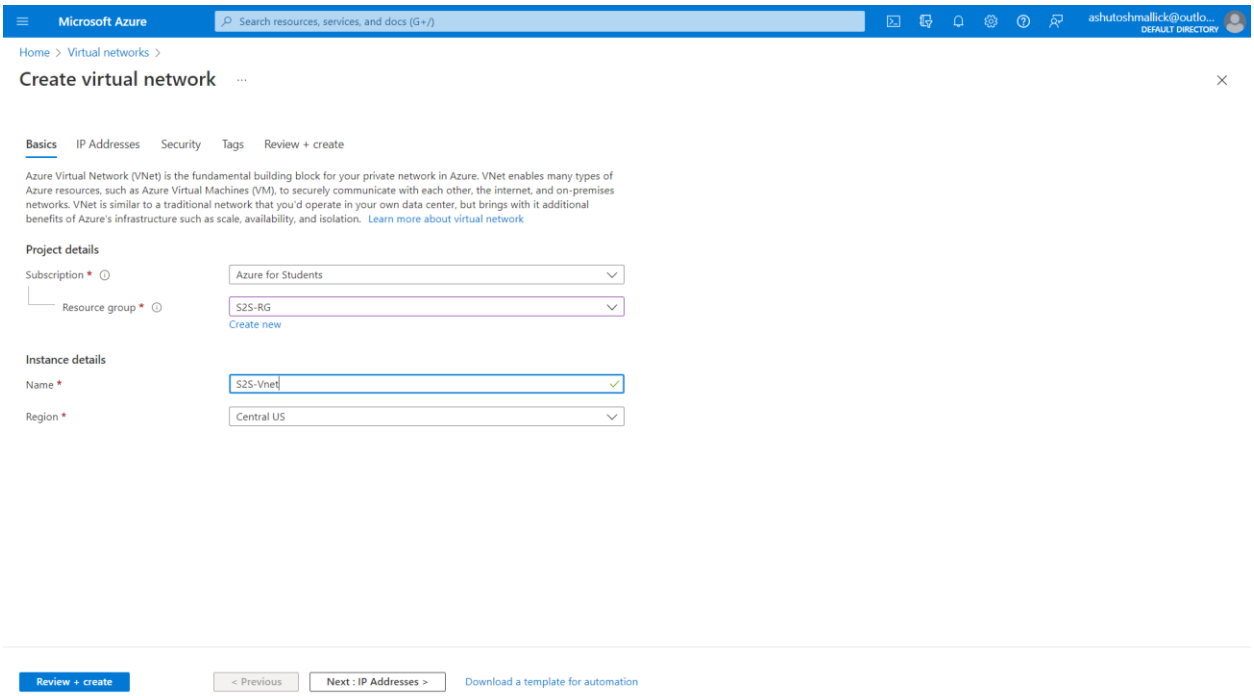


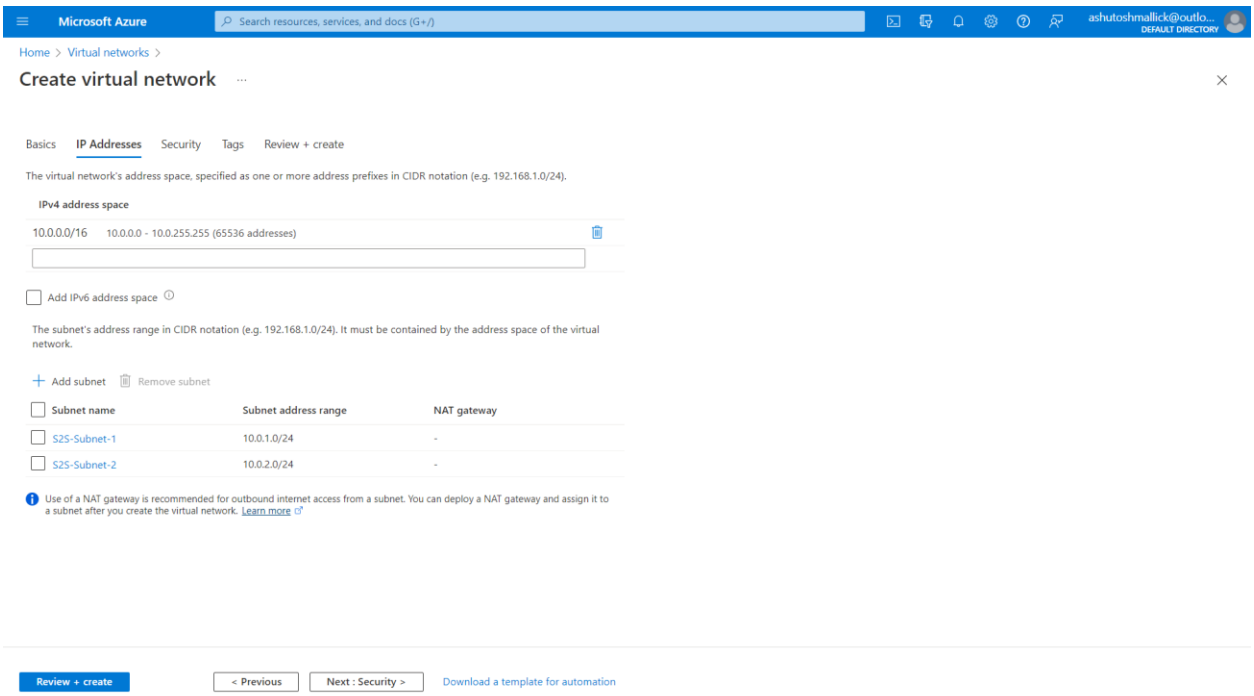
AZURE SITE CONFIGURATION:



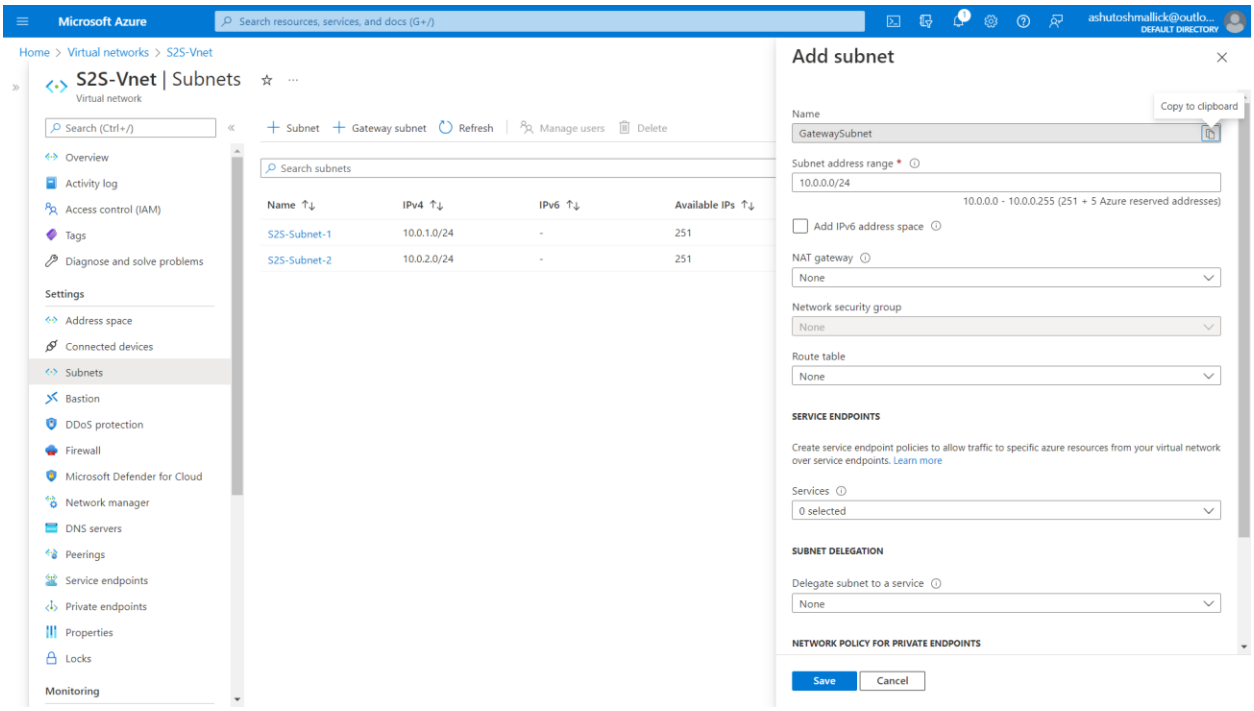
Created a Resource group (S2S-RG).

Create VNET (S2S-Vnet) with S2S-Subnet-1(10.0.1.0/24) and S2S-Subnet-2(10.0.2.0/24).



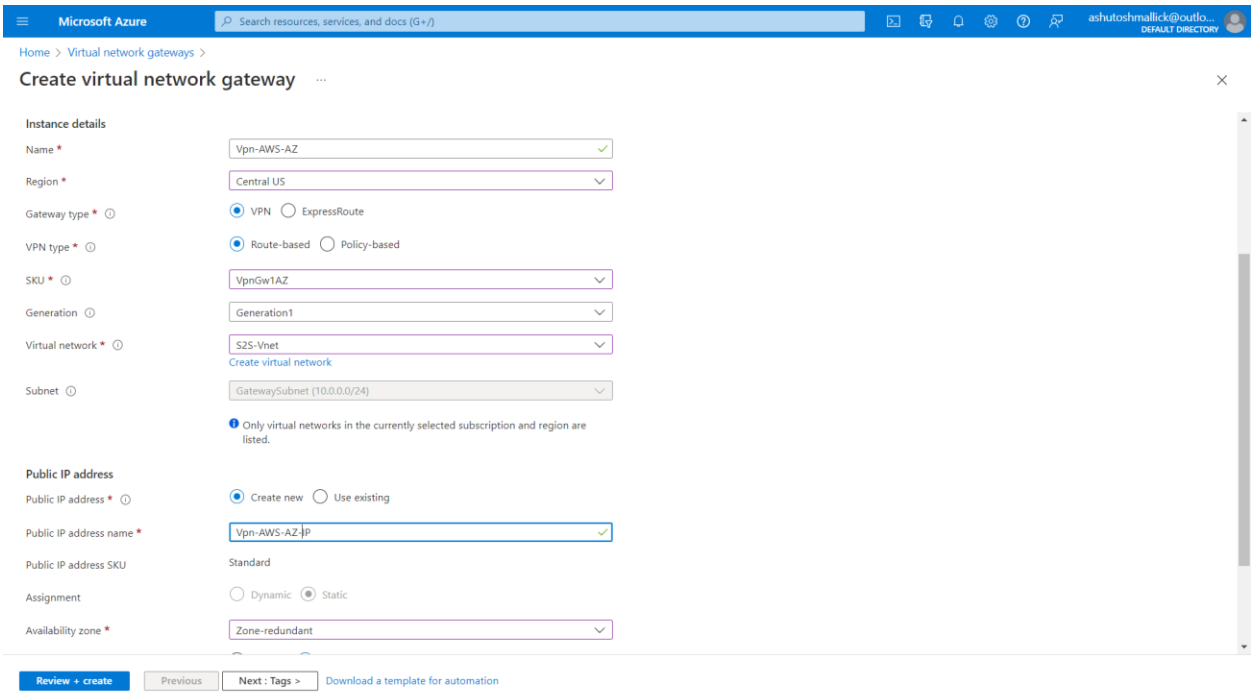


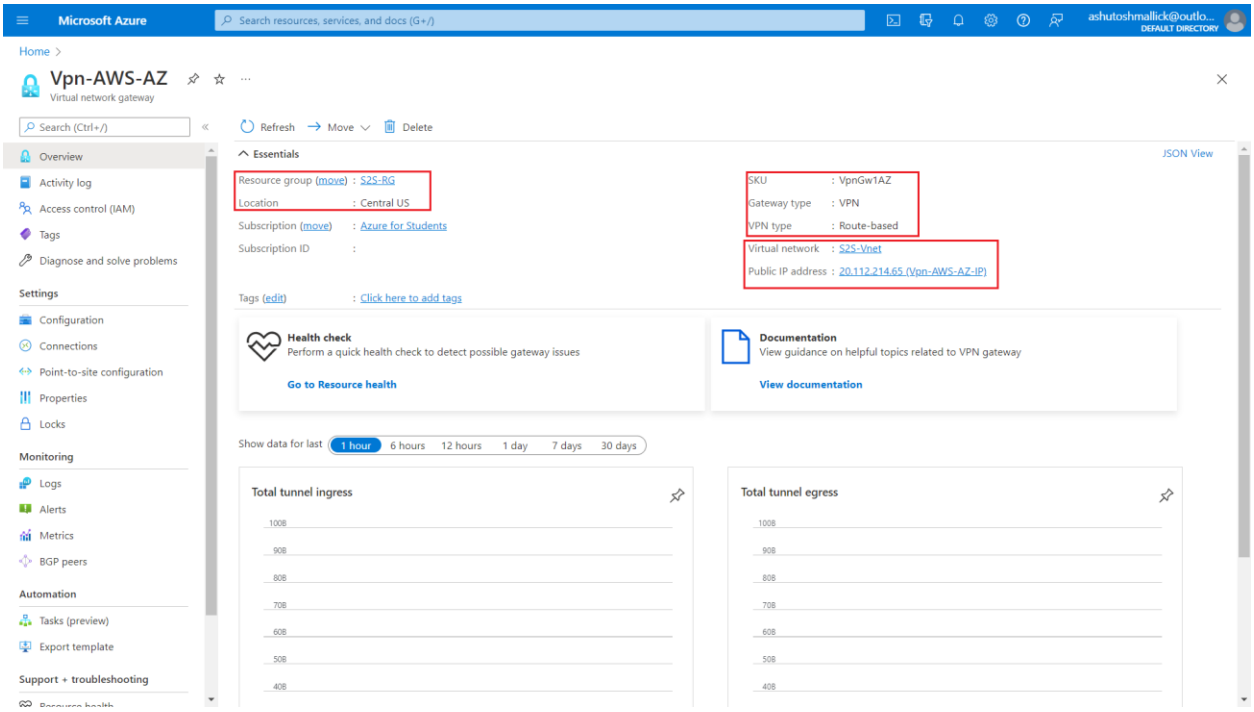
Create Gateway-Subnet (10.0.0.0/24)



Now let's configure Virtual network gateway.

Name	Vpn-AWS-AZ
Gateway Type	VPN
VPN Type	Route Based
SKU	VpnGw1AZ
Generation	Generation1
Virtual Network	S2S-Vnet
Public IP (create new)	Vpn-AWS-AZ-IP



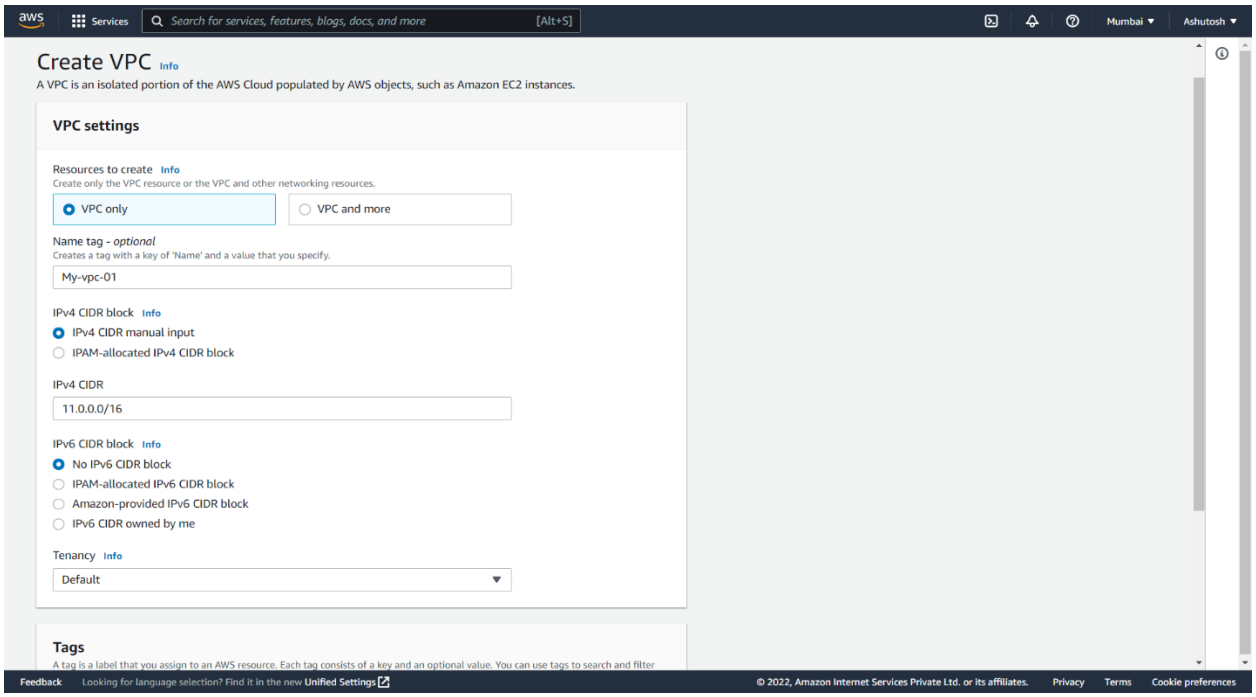


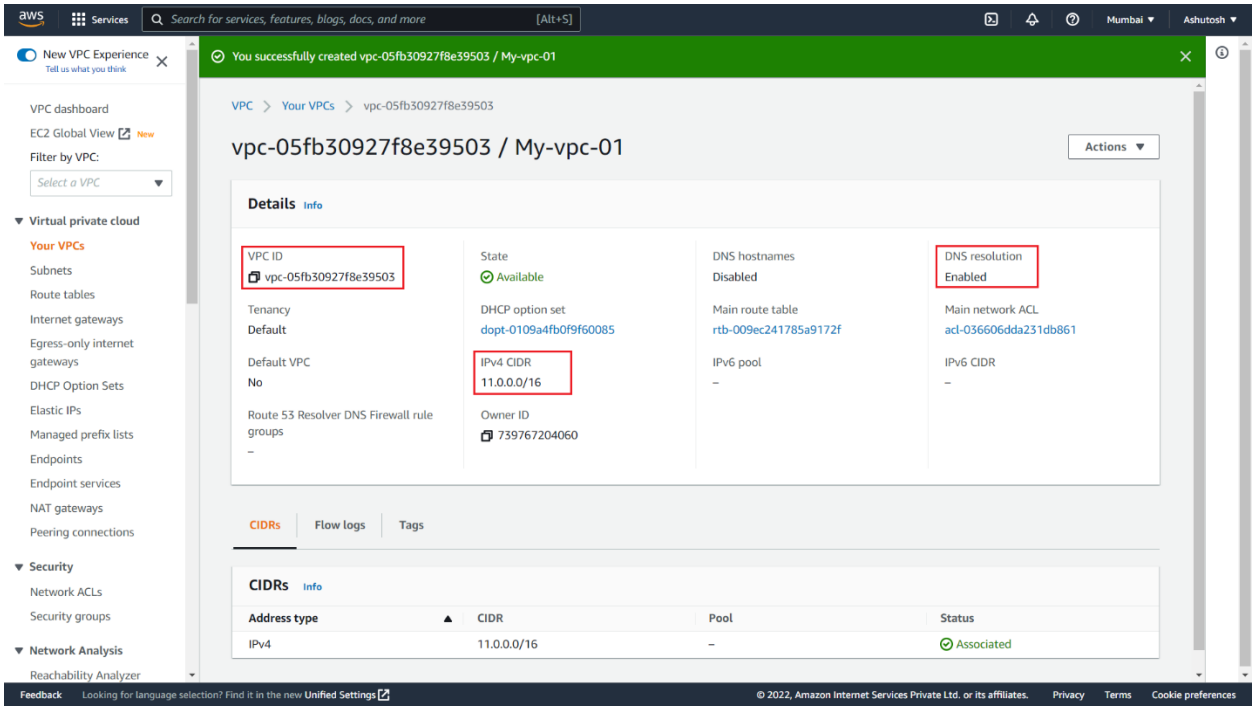
1. AWS SITE CONFIGURATION:

Create a VPC “My-vpc-01” with CIDR 11.0.0.0/16.

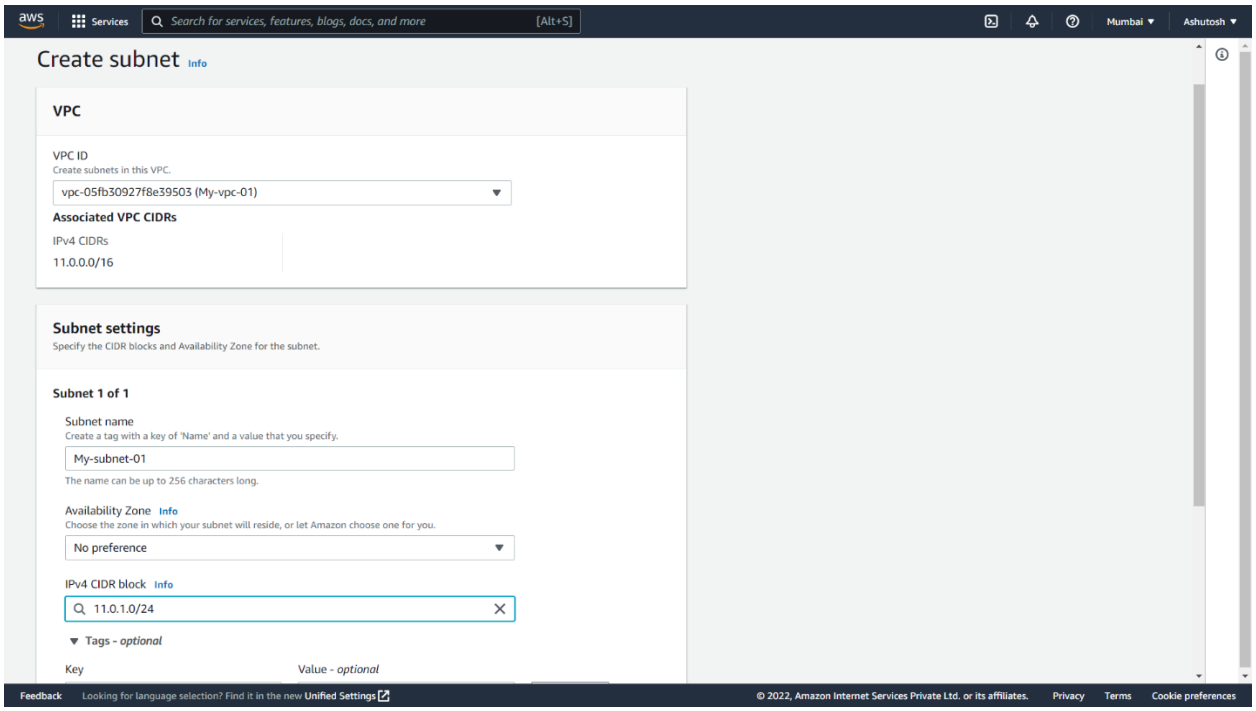
[A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range when you create a VPC. Specify the IPv4 address range as a Classless Inter-Domain

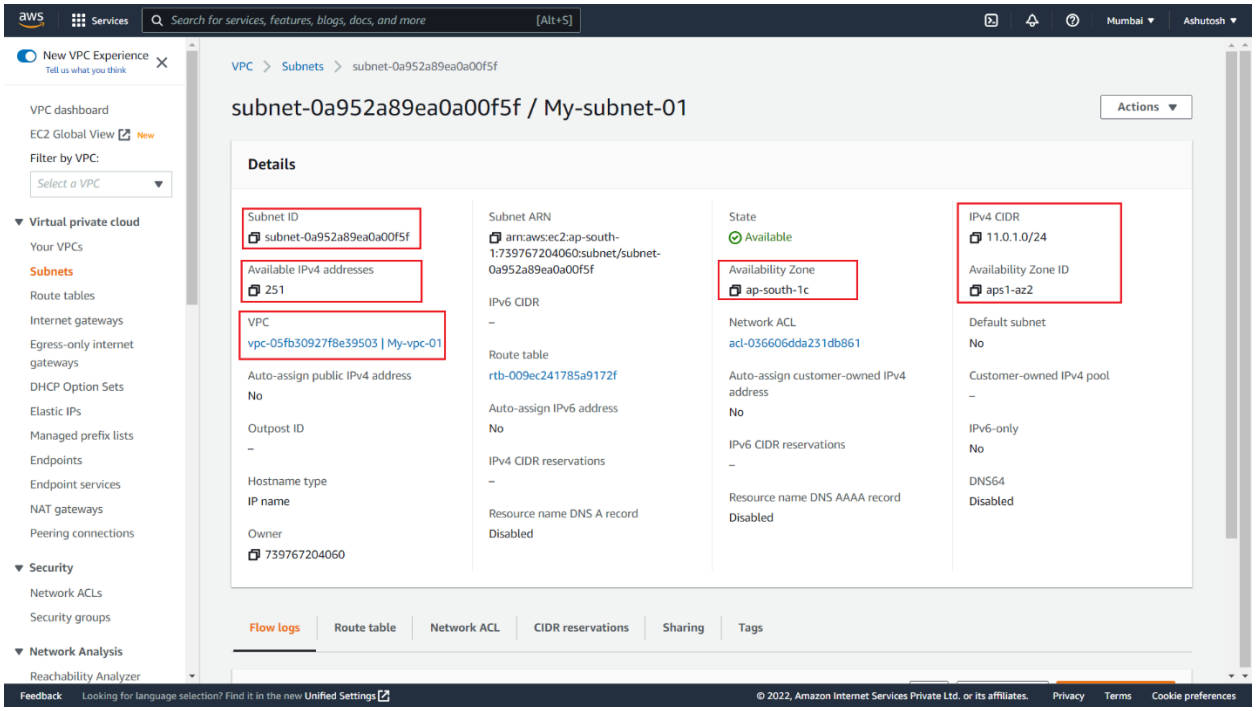
Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16.]





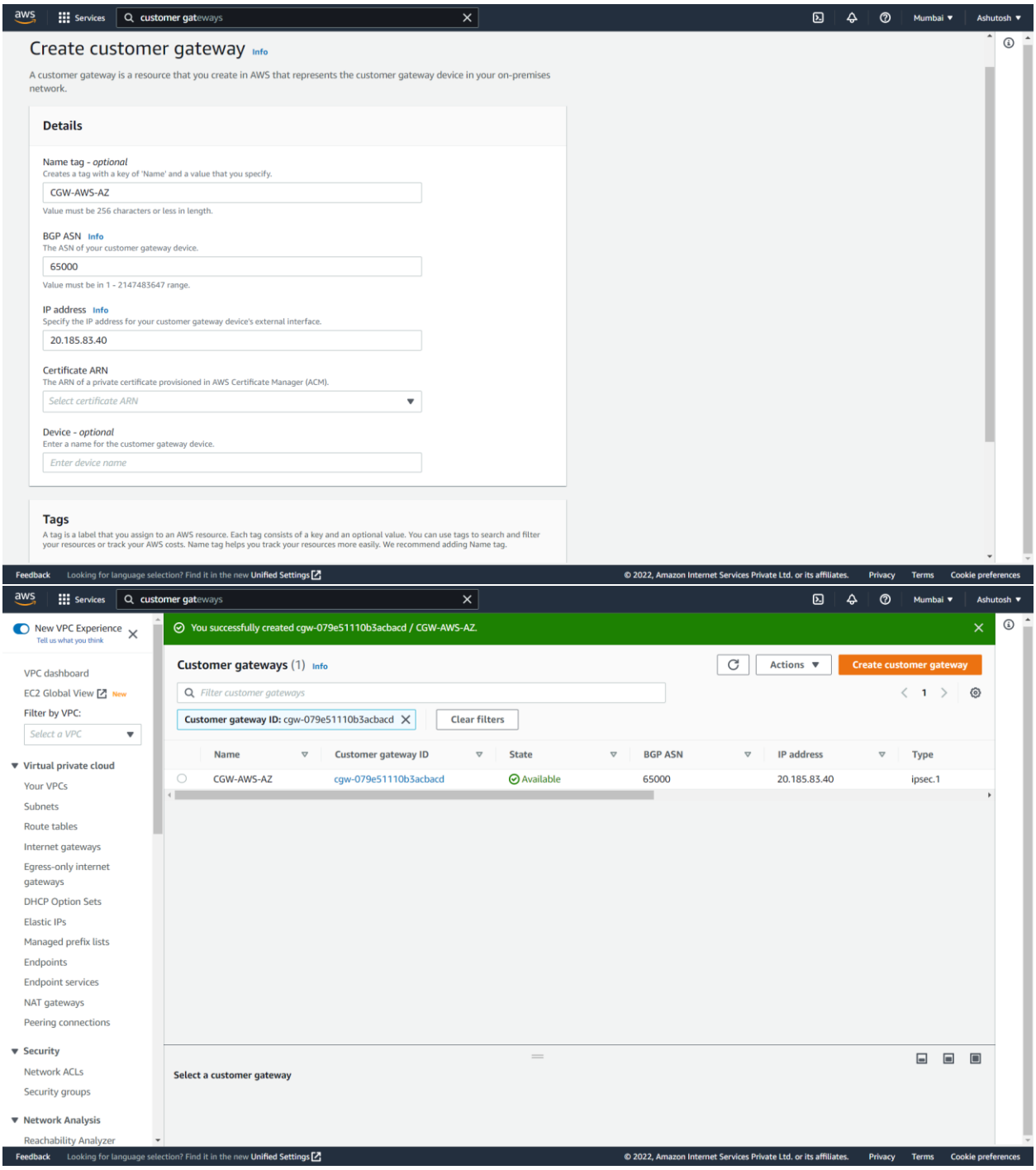
Create a subnet “My-subnet-01” with CIDR (11.0.1.0/24) inside the VPC created.





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

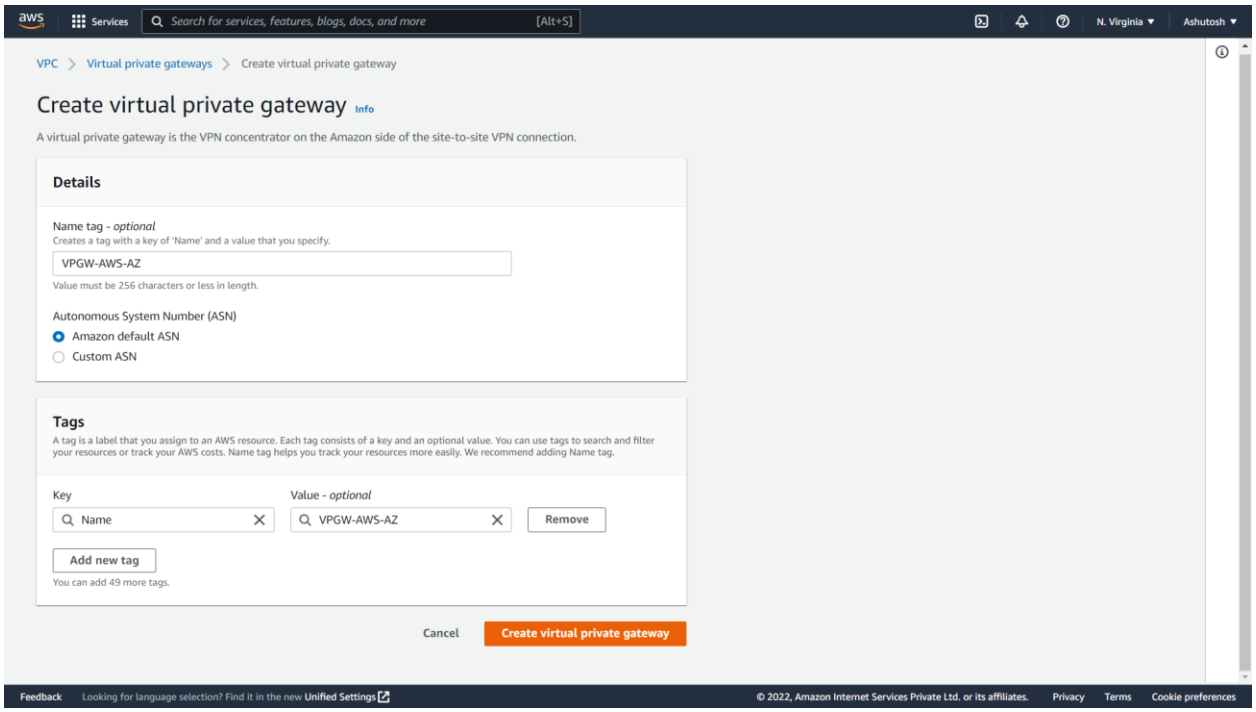
The Customer Gateway is an AWS resource with information to AWS about the customer gateway device, which in this case is the Azure VPN Gateway.

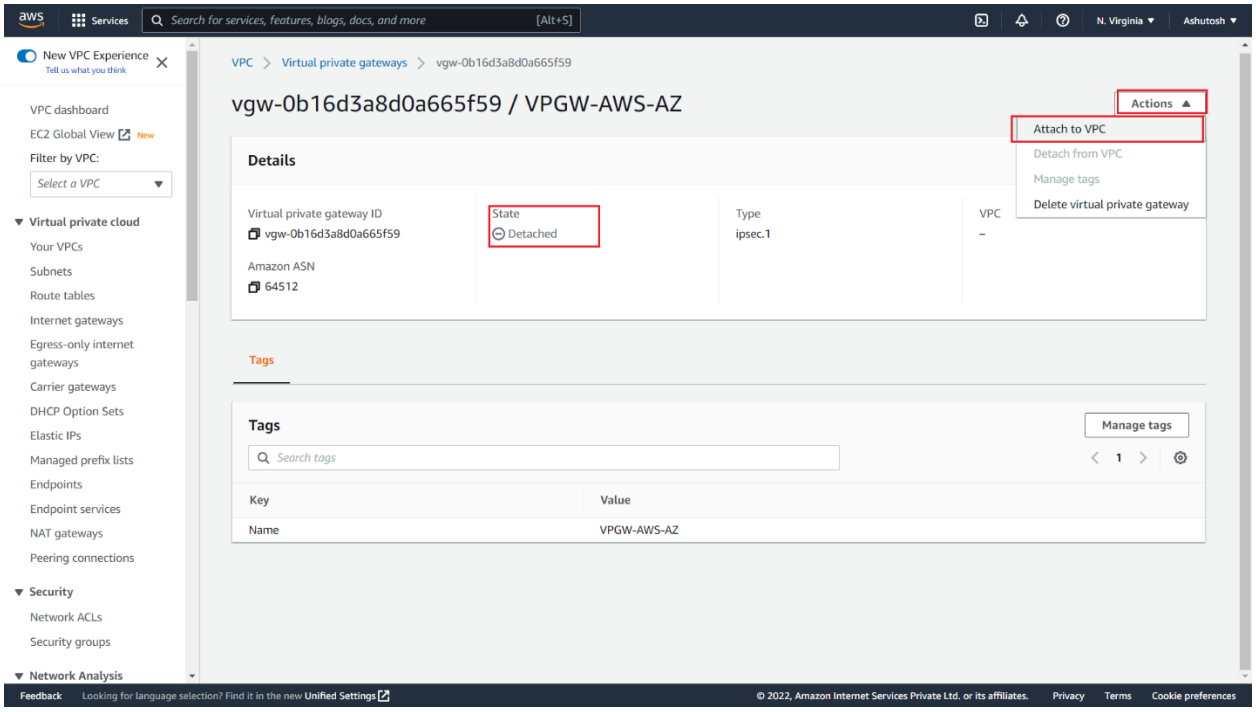


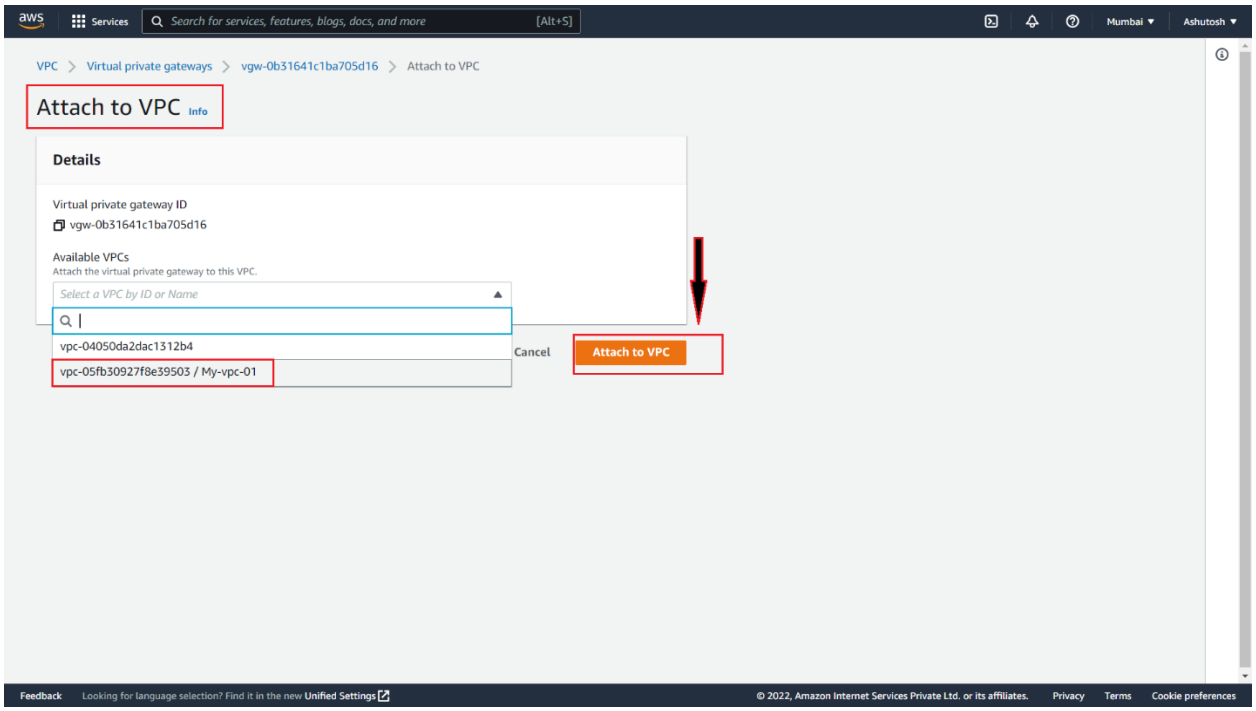
When you create a customer gateway, you provide information about your device to AWS. You or your network administrator must configure the device to work with the site-to-site VPN connection.

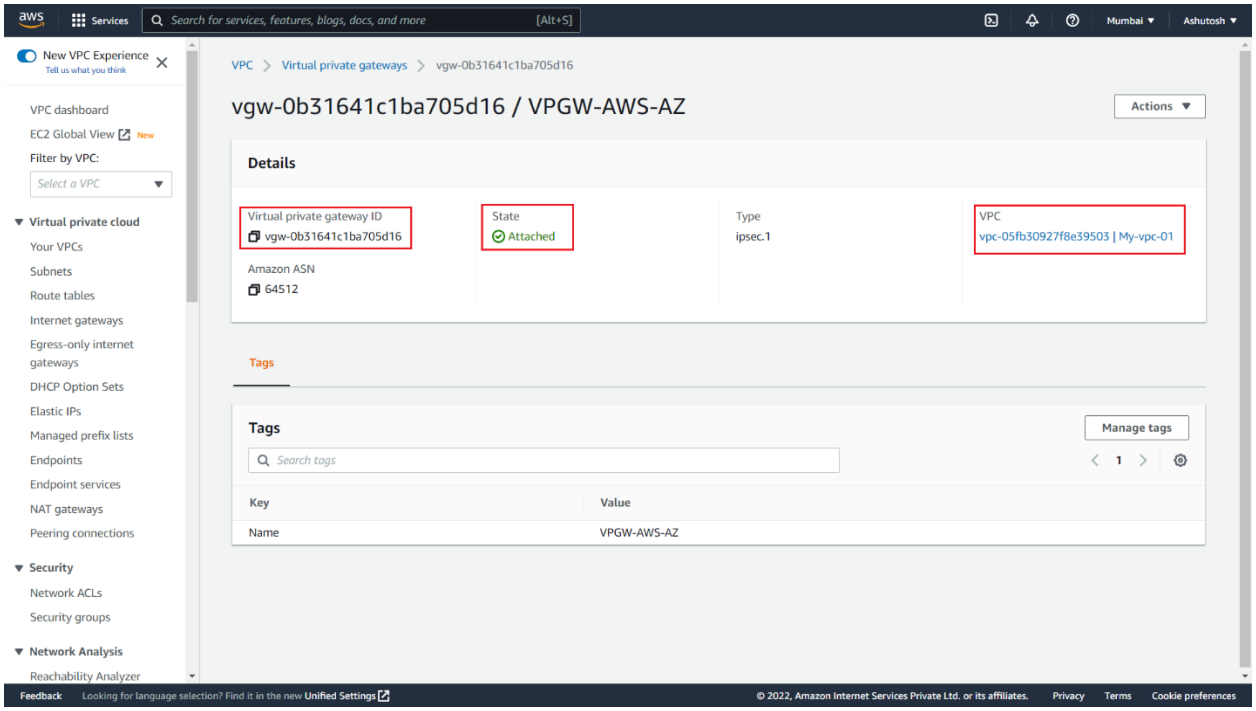
Create the Virtual Private Gateway then attach to the VPC:

A virtual private gateway is the VPN concentrator on the Amazon side of the site-to-site VPN connection. You create a virtual private gateway and attach it to the VPC you want to use for the site-to-site VPN connection.



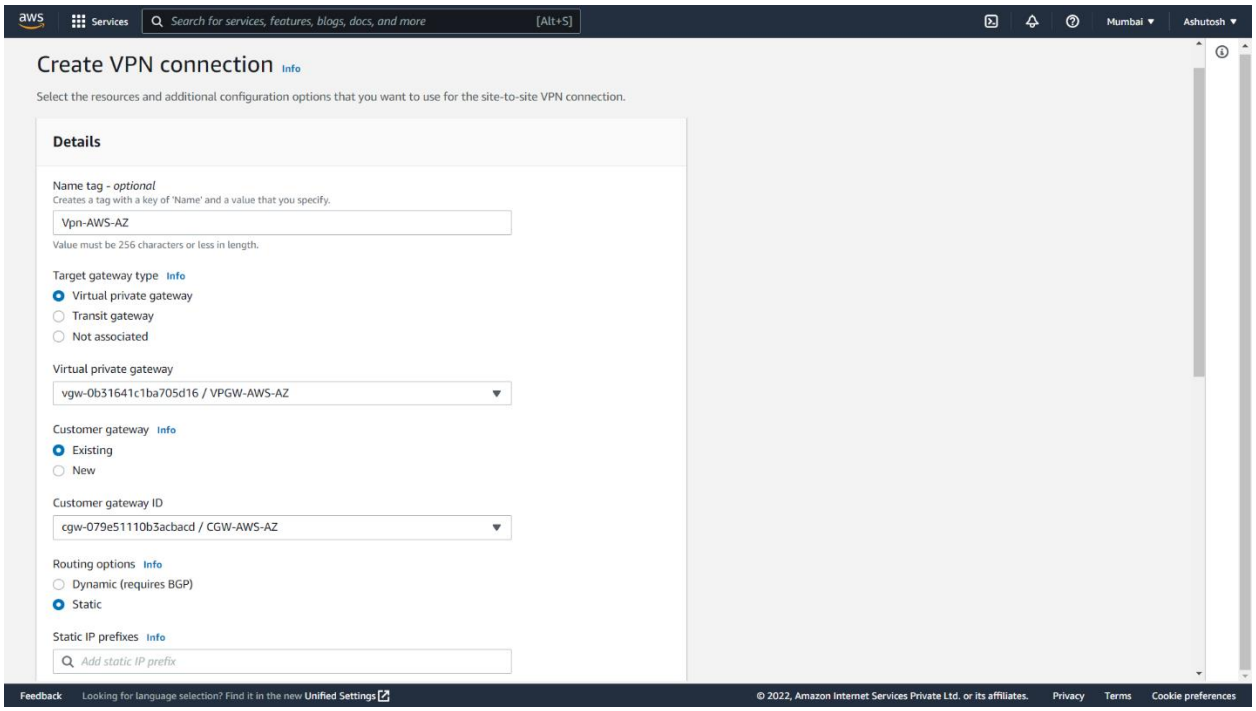


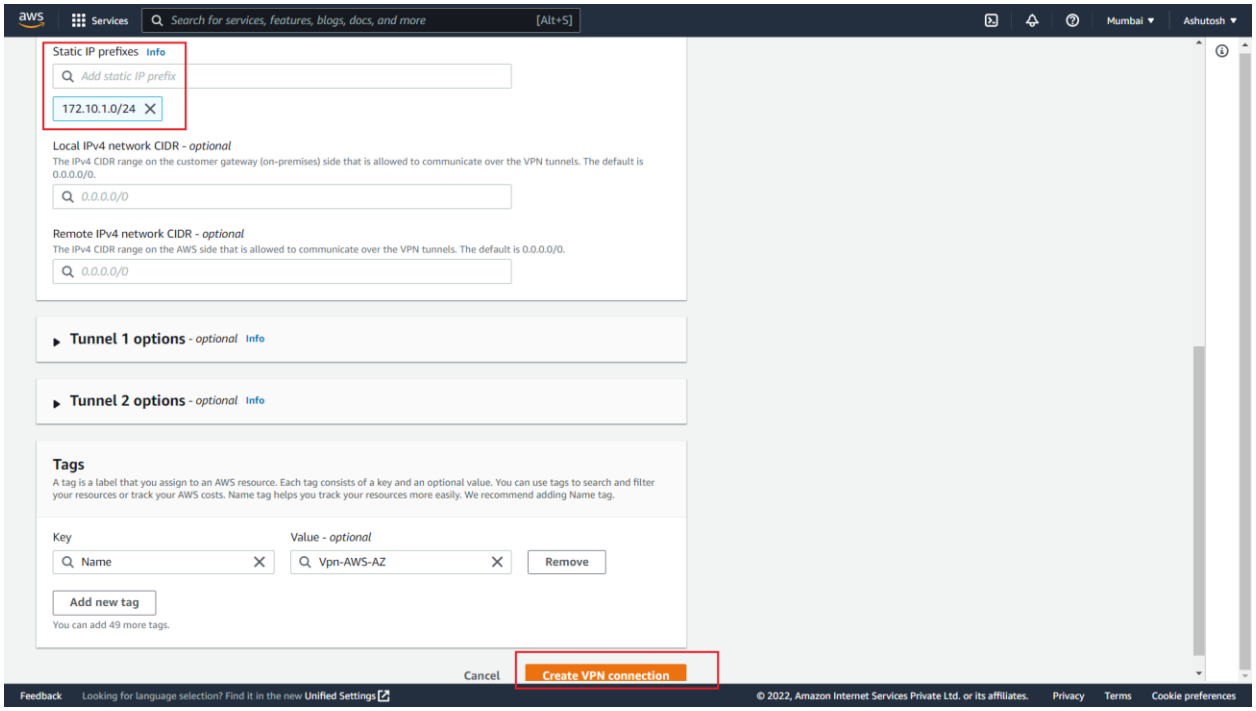




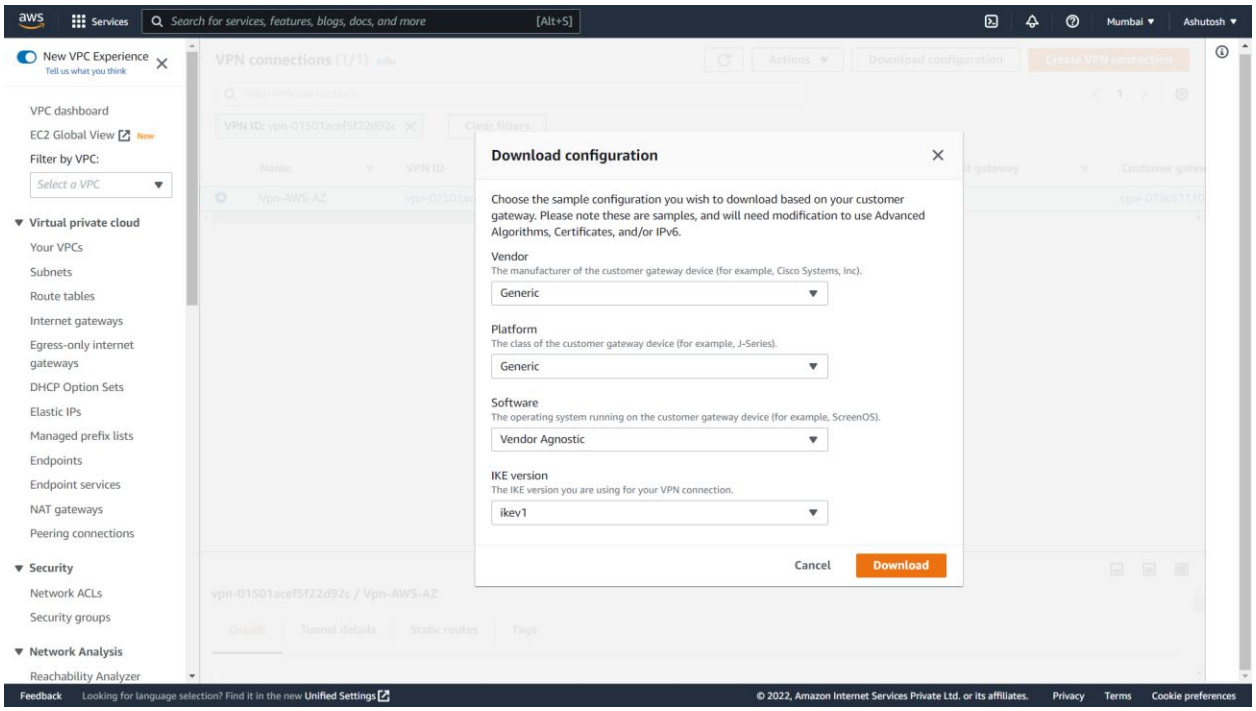
Create a site-to-site VPN Connection:

A Site-to-Site VPN connection offers two VPN tunnels between a virtual private gateway or a transit gateway on the AWS side, and a customer gateway (which represents a VPN device) on the remote (on-premises) side.





Download the configuration file.




```
Amazon Web Services
Virtual Private Cloud

VPN Connection Configuration
================================================================================
AWS utilizes unique identifiers to manipulate the configuration of
a VPN Connection. Each VPN Connection is assigned a VPN Connection Identifier
and is associated with two other identifiers, namely the
Customer Gateway Identifier and the Virtual Private Gateway Identifier.

Your VPN Connection ID                 : vpn-01501acef5f22d92c
Your Virtual Private Gateway ID        : vgw-0b31641c1ba705d16
Your Customer Gateway ID               : cgw-079e51110b3acbacd

A VPN Connection consists of a pair of IPSec tunnel security associations (SAs).
It is important that both tunnel security associations be configured.
```

```
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 AES128, SHA1, and DH Group 2.
Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.
You will need to modify these sample configuration files to take advantage of AES256, SHA256, or other DH groups like 2, 14-18, 22, 23, and 24.
NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these sample configurations to match the custom settings for your tunnels.

Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".
The address of the external interface for your customer gateway must be a static address.
Your customer gateway may reside behind a device performing network address translation (NAT).
To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules to unblock UDP port 4500.
If not behind NAT, and you are not using an Accelerated VPN, we recommend disabling NAT-T. If you are using an Accelerated VPN, make sure that NAT-T is enabled.
  - 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
```

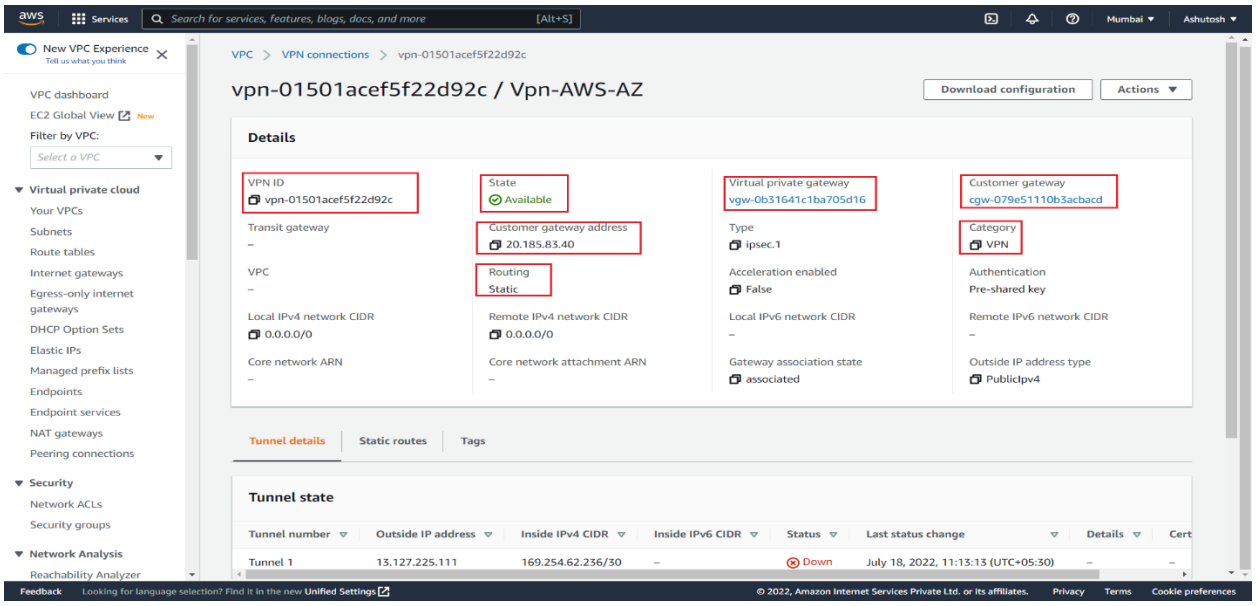
```
#2: IPSec Configuration

Configure the IPSec SA as follows:
Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.
Please note, you may use these additionally supported IPSec parameters for encryption like AES256 and other DH groups like 2, 5, 14-18, 22, 23, and 24.
NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these sample configurations to match the custom settings for your tunnels.

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

```
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 AES128, SHA1, and DH Group 2.
Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.
You will need to modify these sample configuration files to take advantage of AES256, SHA256, or other DH groups like 2, 14-18, 22, 23, and 24.
NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these sample configurations to match the custom settings for your tunnels.

Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".
The address of the external interface for your customer gateway must be a static address.
Your customer gateway may reside behind a device performing network address translation (NAT).
To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules to unblock UDP port 4500.
If not behind NAT, and you are not using an Accelerated VPN, we recommend disabling NAT-T. If you are using an Accelerated VPN, make sure that NAT-T is enabled.
  - 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 AES128, SHA2, and DH Group 14.
Please note, you may use these additionally supported IPSec parameters for encryption like AES256 and other DH groups like 2, 5, 14-18, 22, 23, and 24.
NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these sample configurations to match the custom settings for your tunnels.

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

```
#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 relate
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         : 13.127.225.111

Inside IP Addresses
  - Customer Gateway                : 169.254.62.238/30
  - Virtual Private Gateway         : 169.254.62.237/30

Configure your tunnel to fragment at the optimal size:
  - Tunnel interface MTU     : 1436 bytes
```

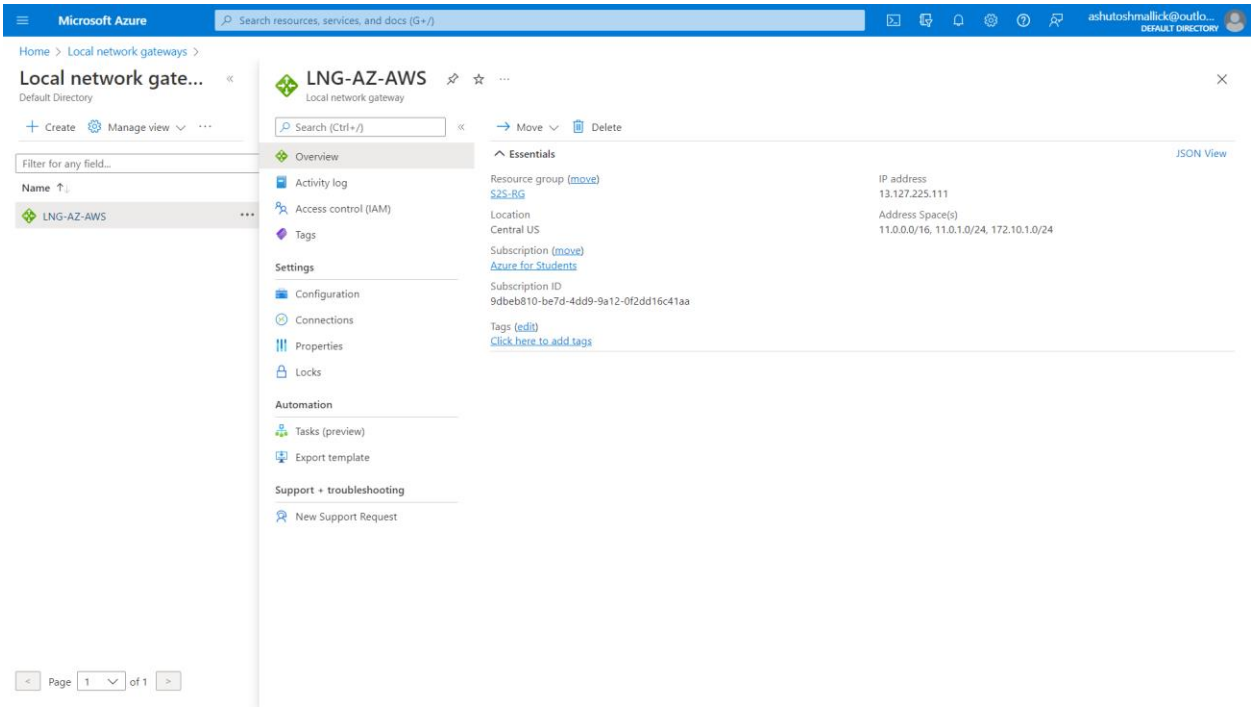
```
#3: Tunnel Interface Configuration

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 relate
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          : 65.1.105.70

Inside IP Addresses
  - Customer Gateway                : 169.254.161.38/30
  - Virtual Private Gateway          : 169.254.161.37/30

Configure your tunnel to fragment at the optimal size:
  - Tunnel interface MTU     : 1436 bytes
```

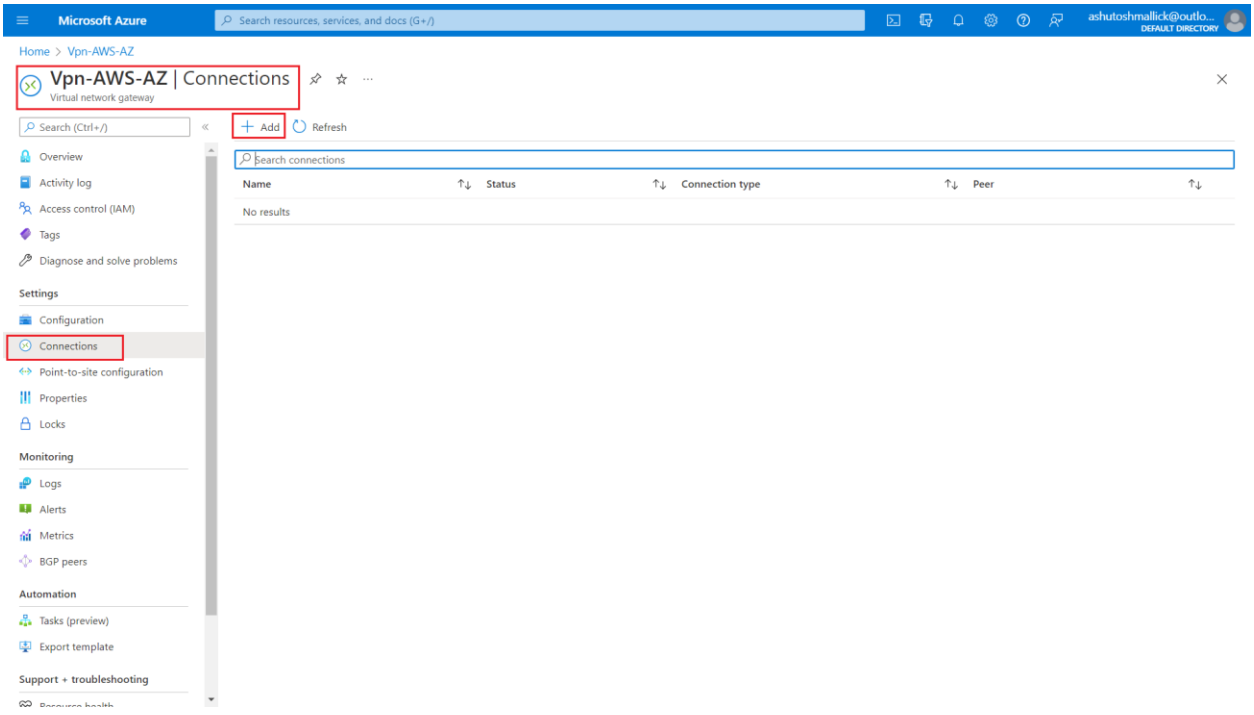
After the creation, you should have something like this:



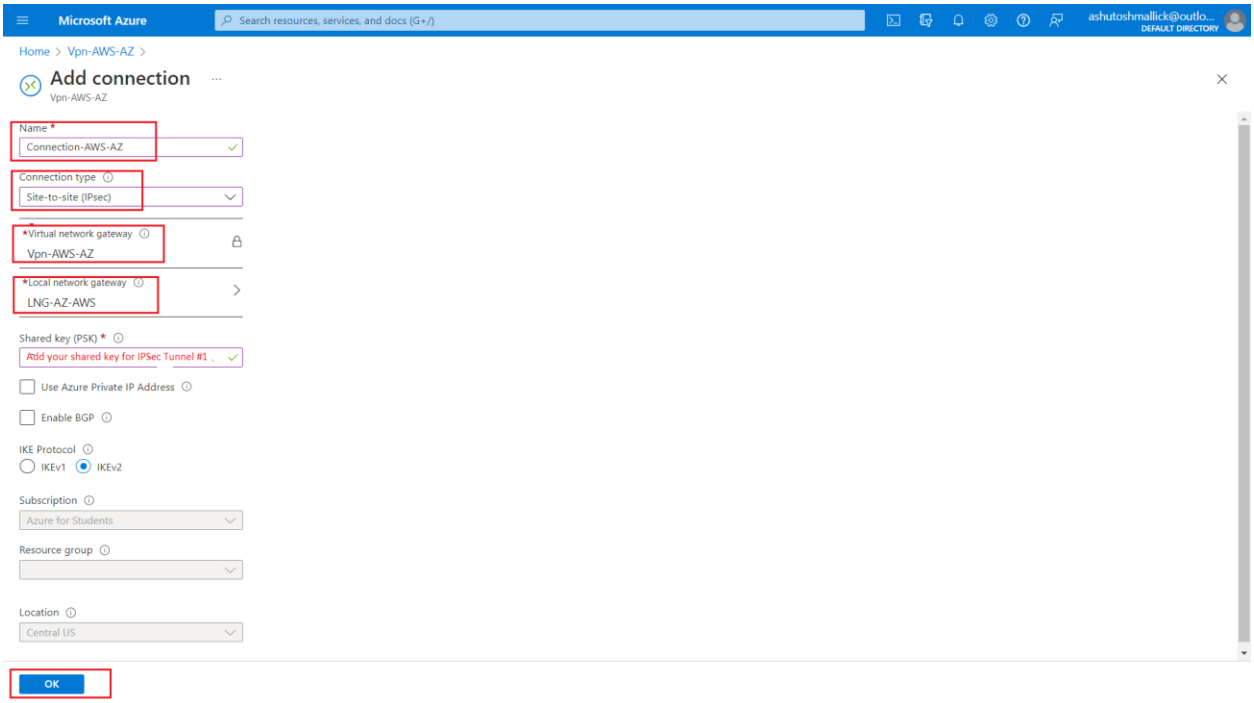
Adding the AWS information on Azure Configuration:

Now let’s create the Local Network Gateway.

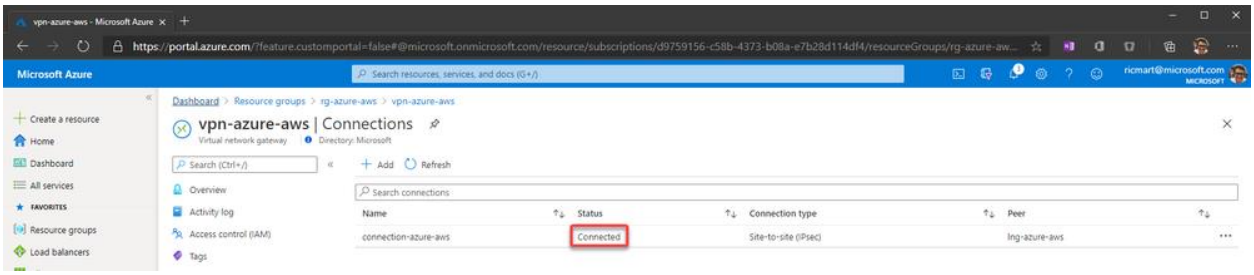




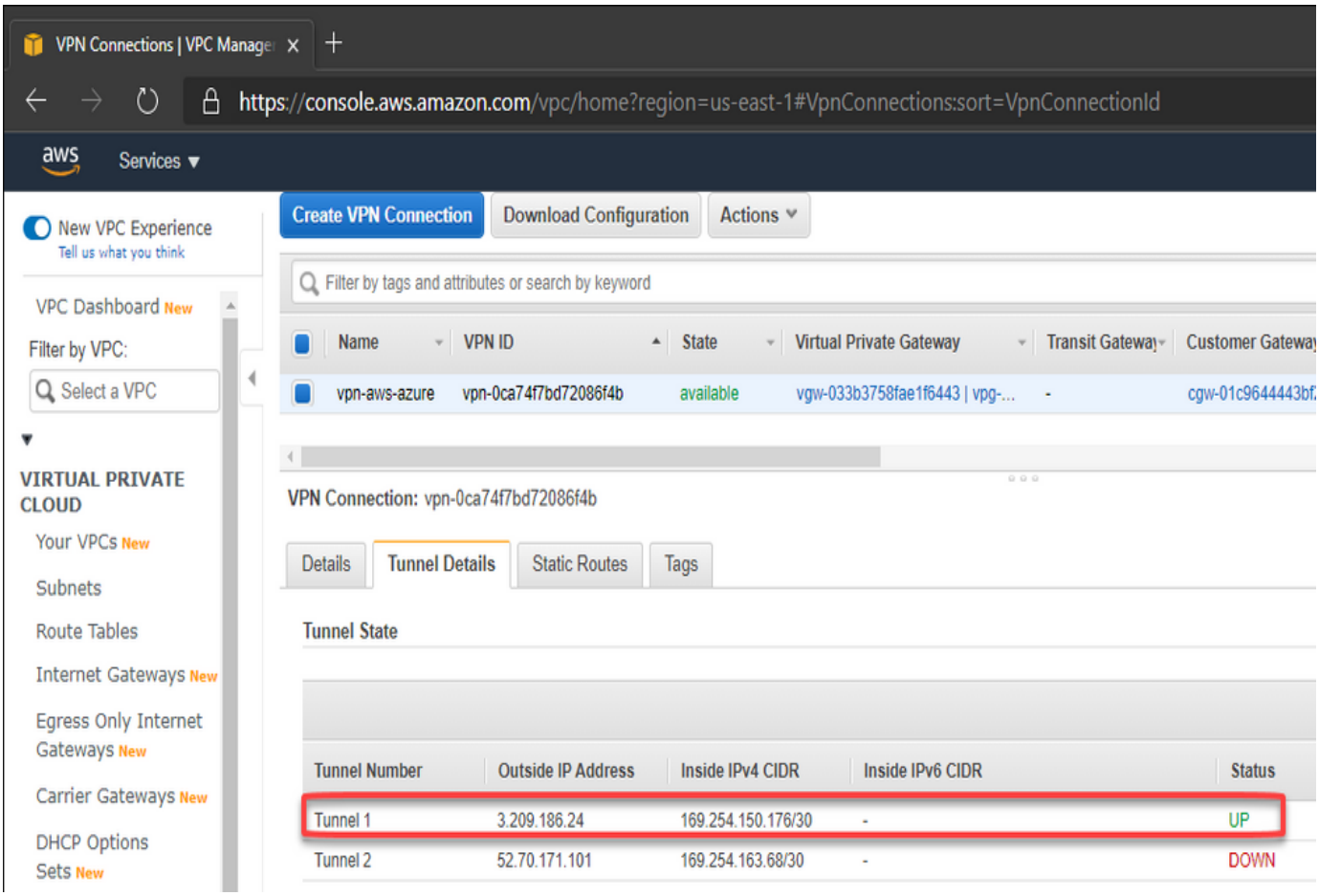
You should fill the fields according below. Please note that the Shared key was obtained at the configuration file downloaded earlier and in this case, I'm using the Shared Key for the Ipsec tunnel #1 created by AWS and described at the configuration file.



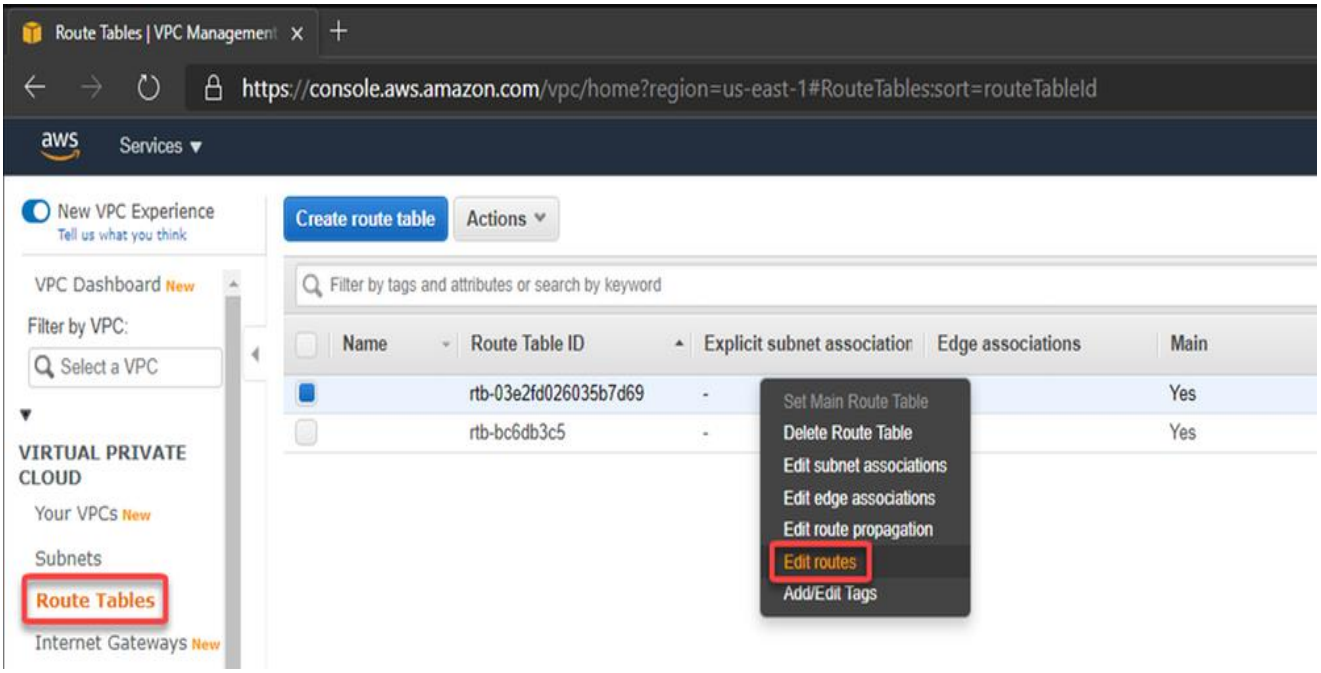
After a few minutes, you can see the connection established:



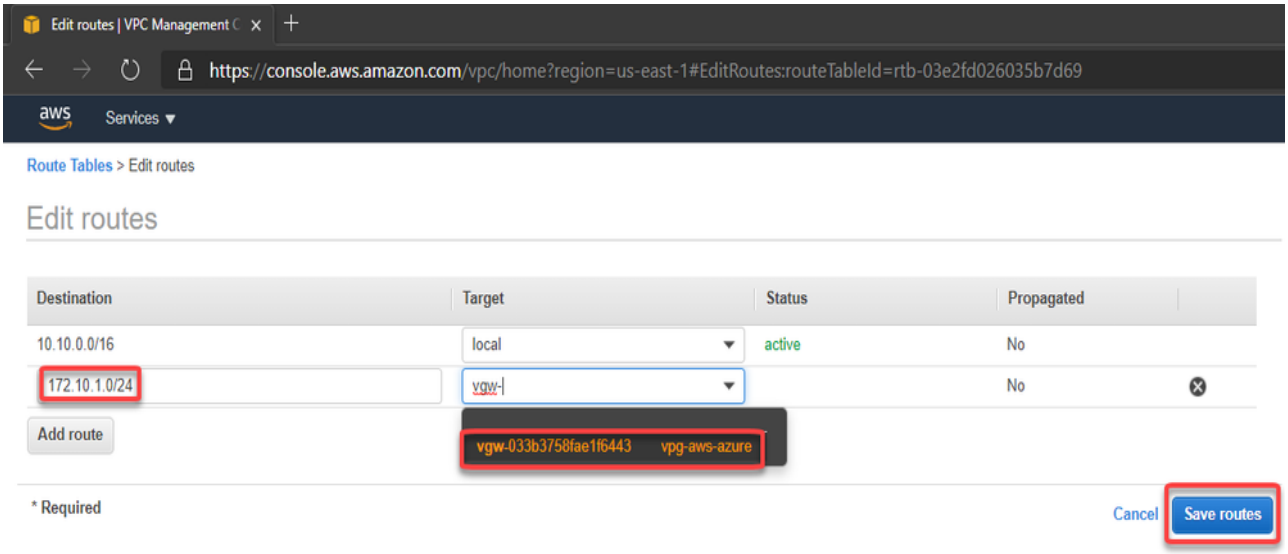
In the same way, we can check on AWS that the 1st tunnel is up:



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

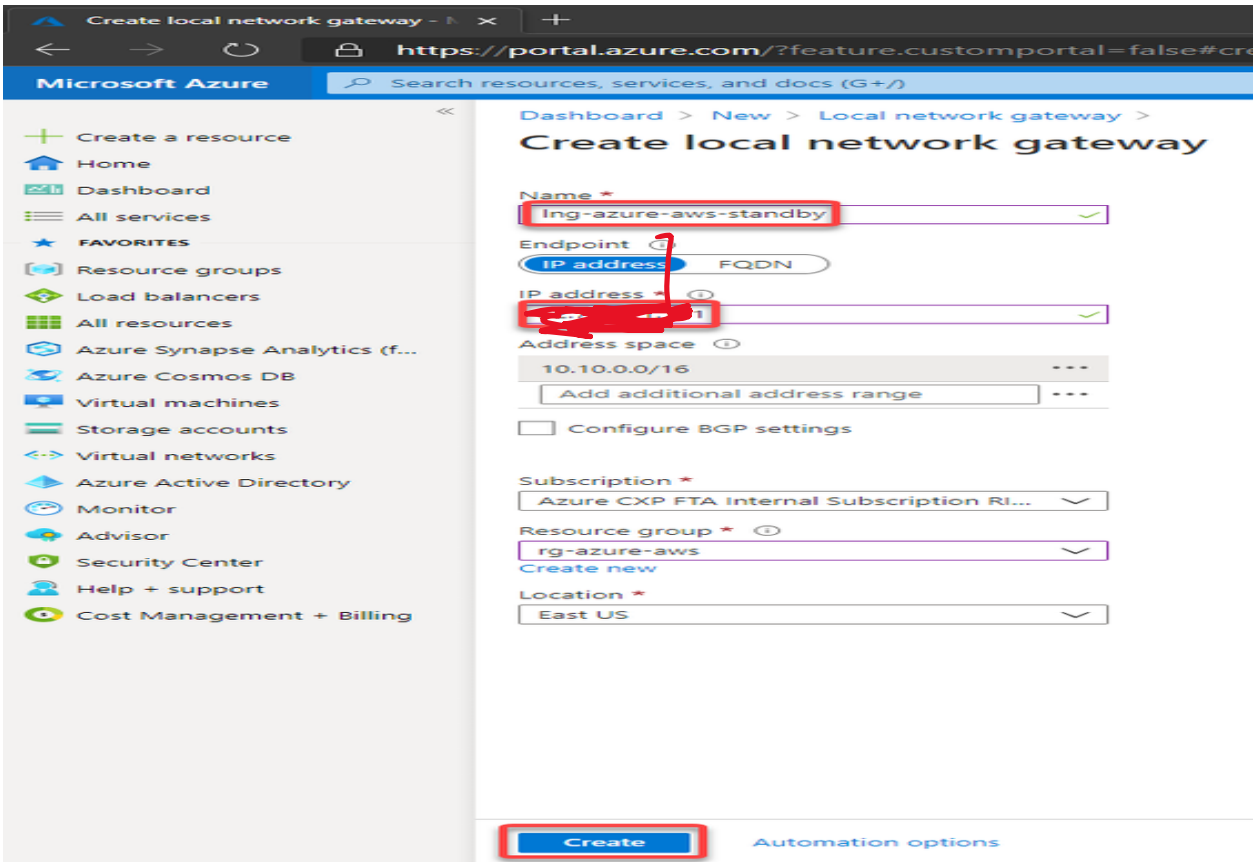


And add the route to Azure subnet through the Virtual Private Gateway:

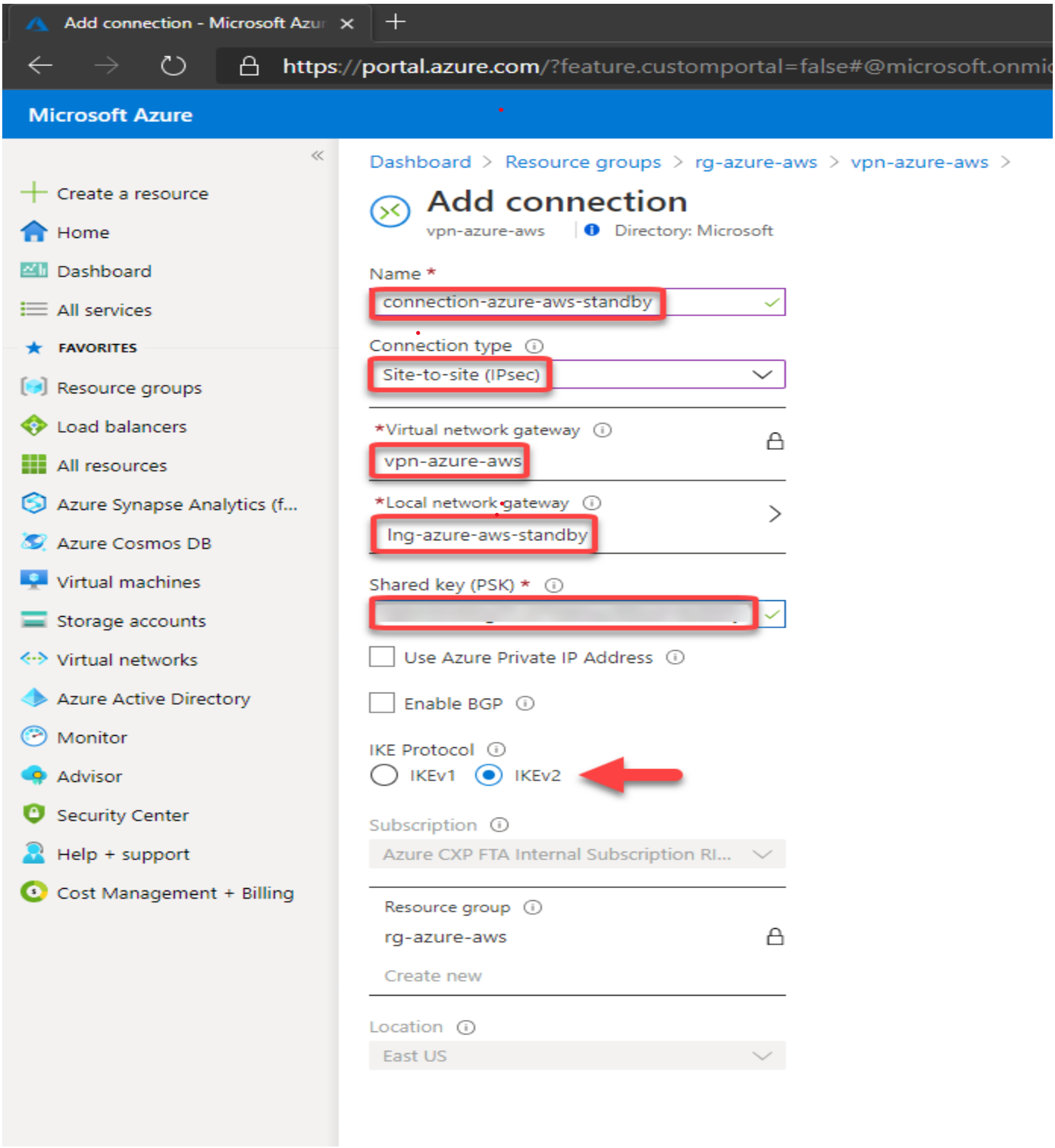


12. Adding high availability

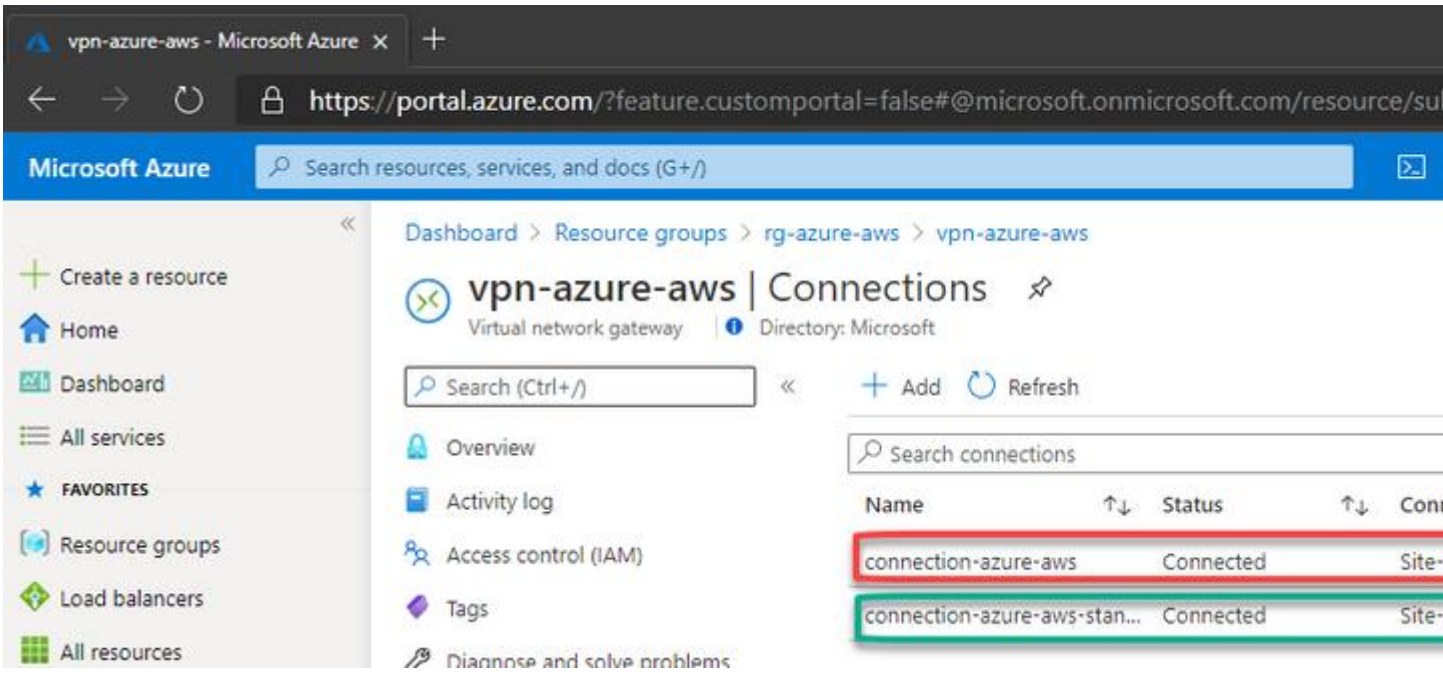
Now we can create a 2nd connection to ensure high availability. To do this let's create another Local Network Gateway which we will point to the public ip address of the IPSec tunnel #2 on the AWS

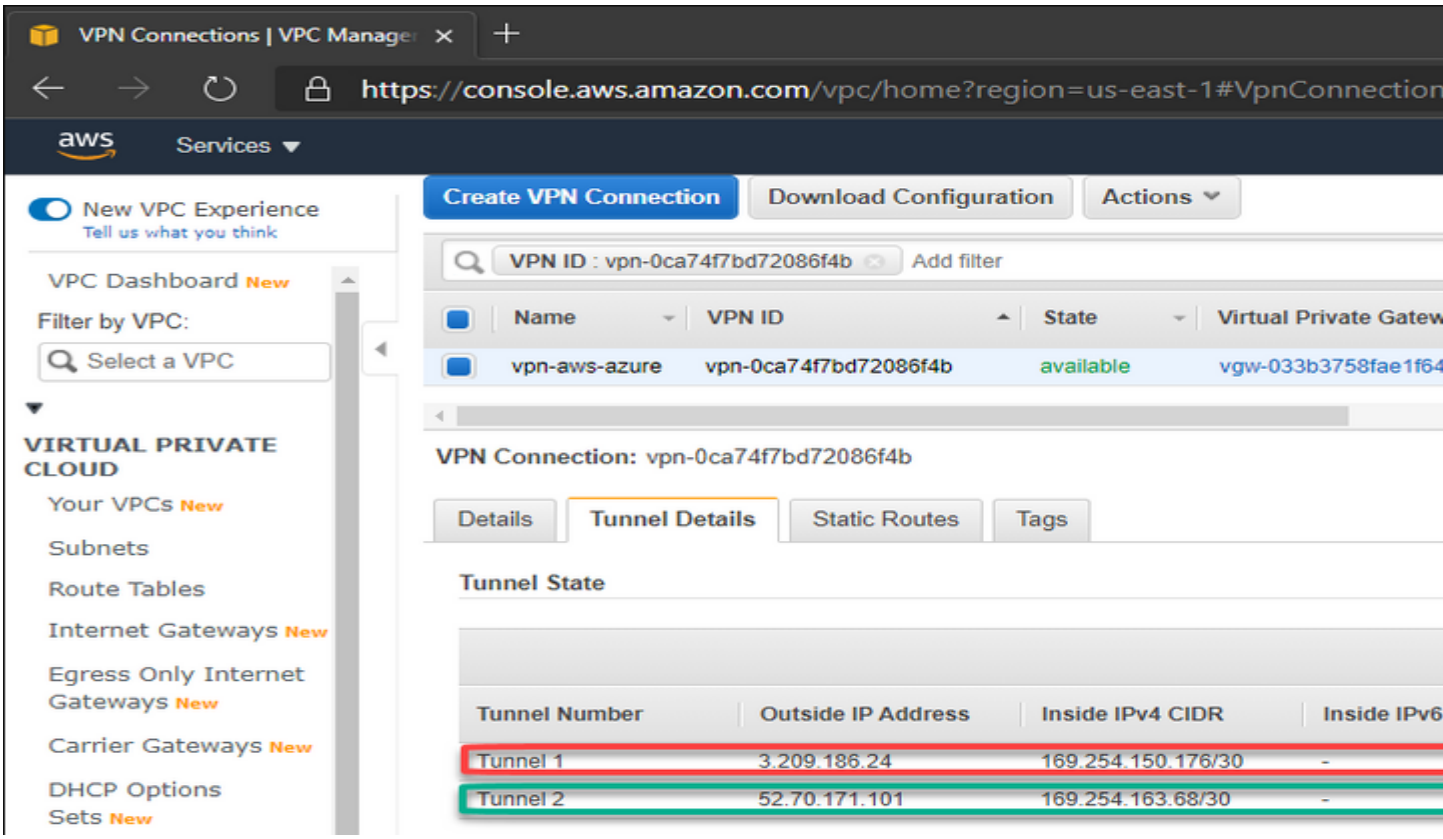


Then we can create the 2nd connection on the Virtual Network Gateway:



And in a few moments we'll have:

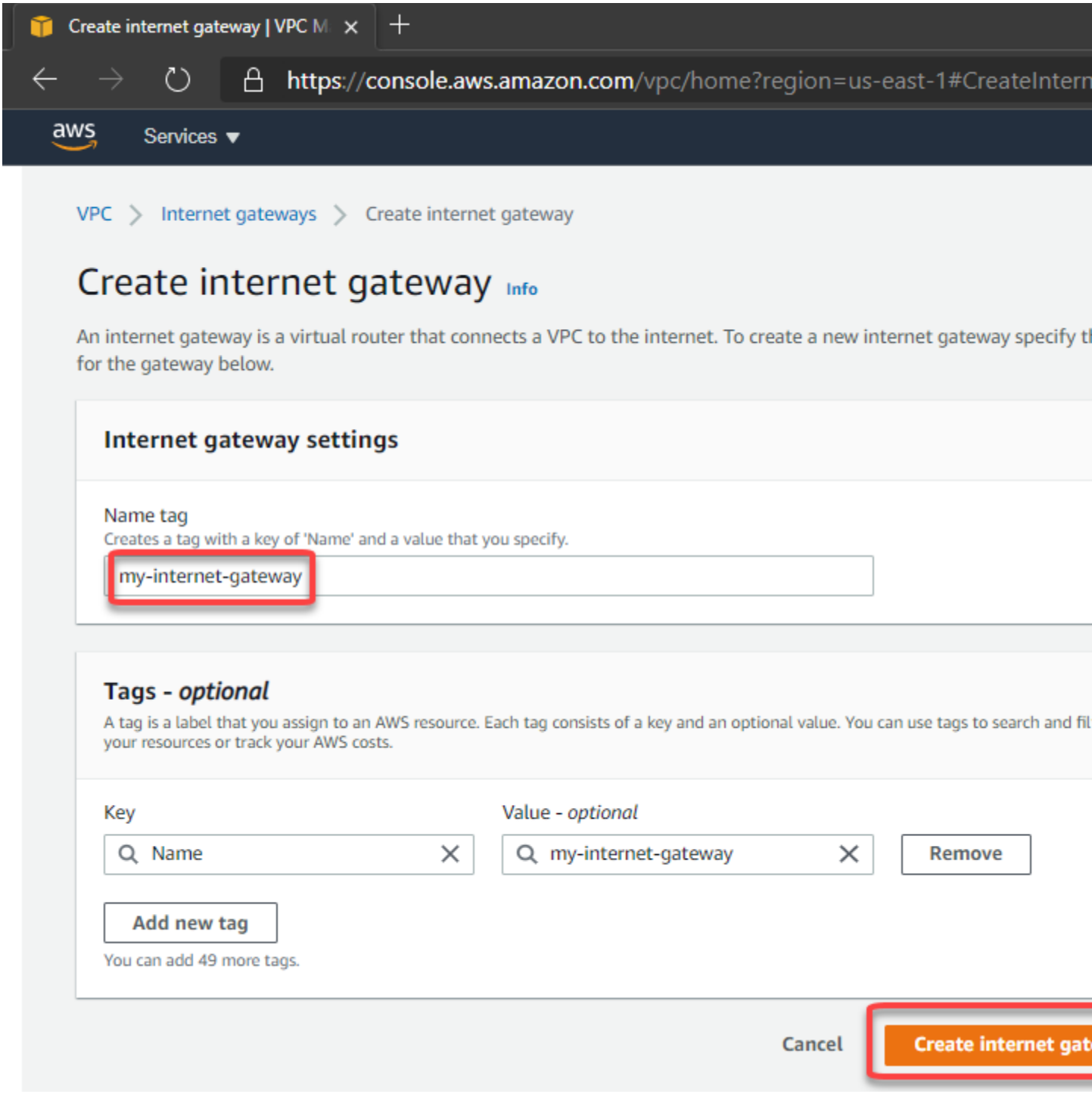




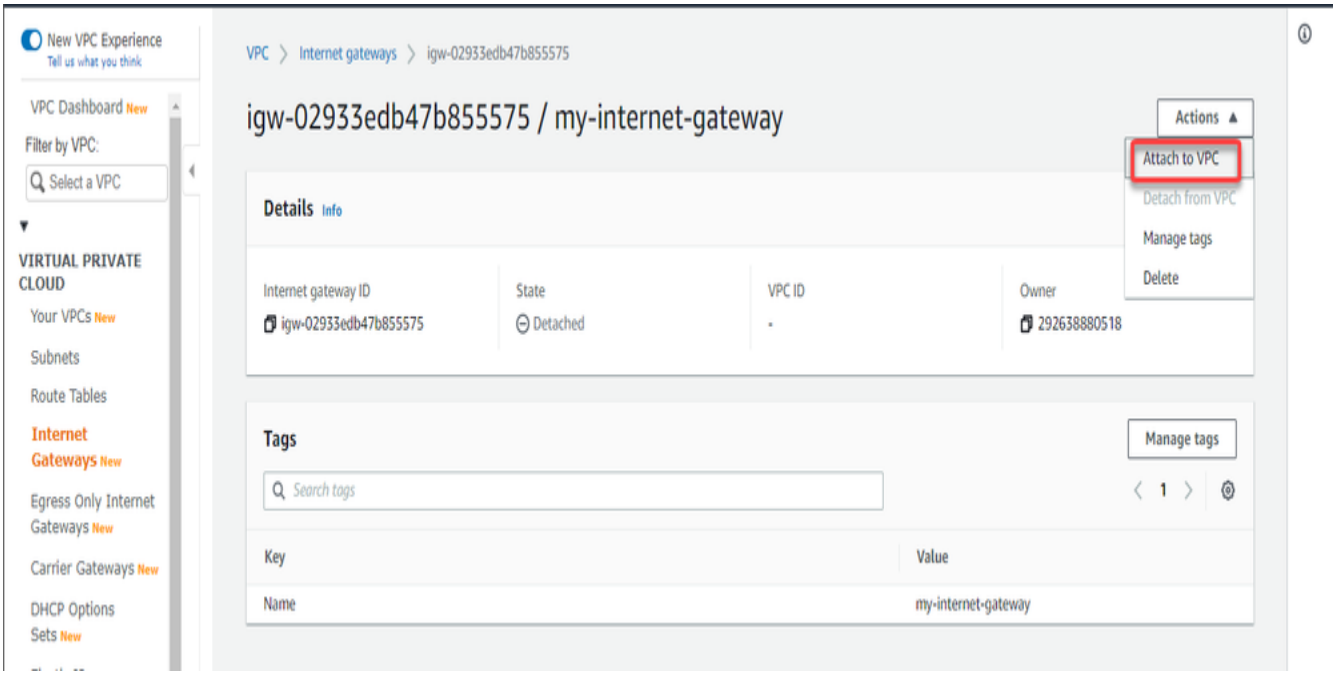
With this, our VPN connection is established on both sides and the work is done.

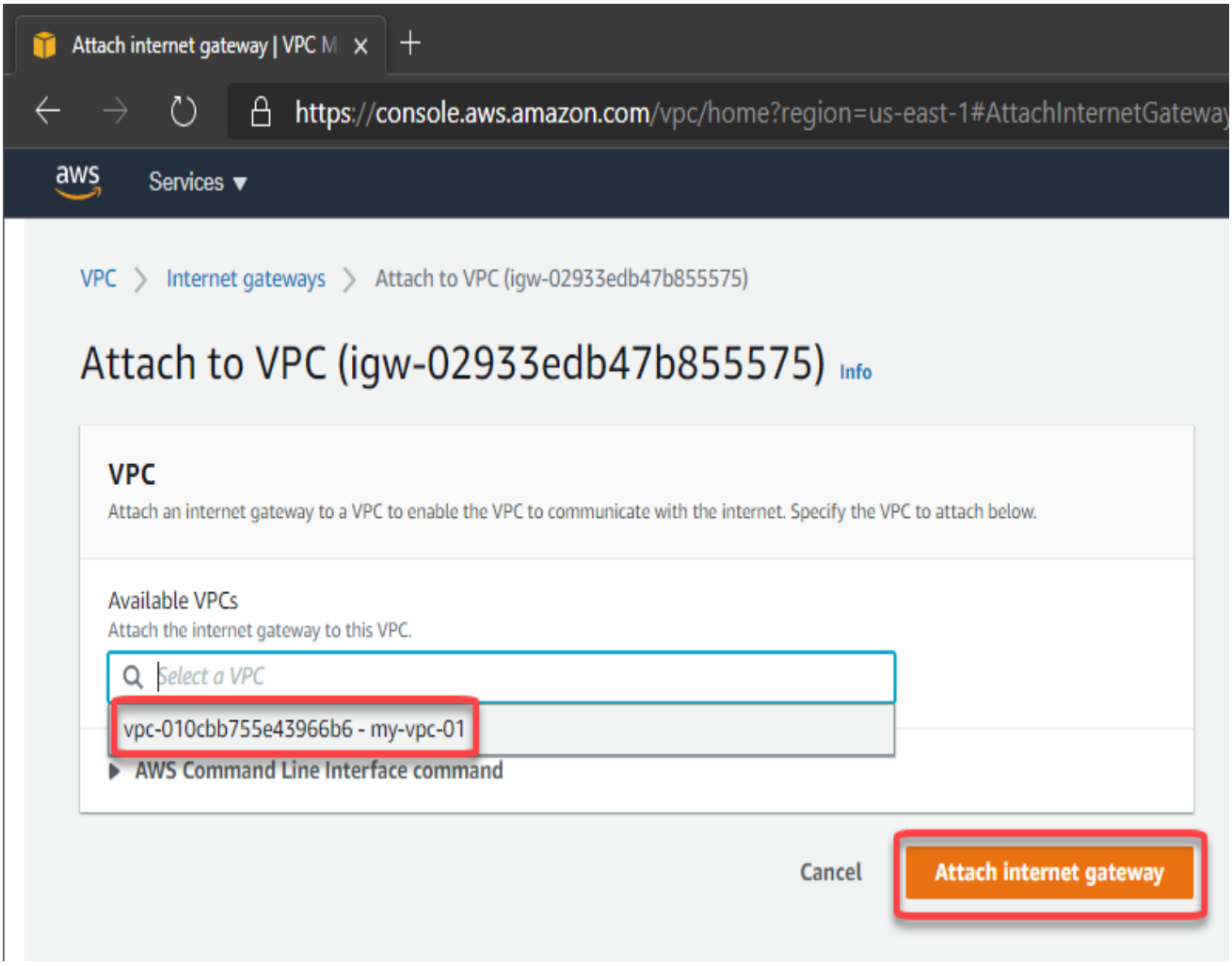
13. Let's test!

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:

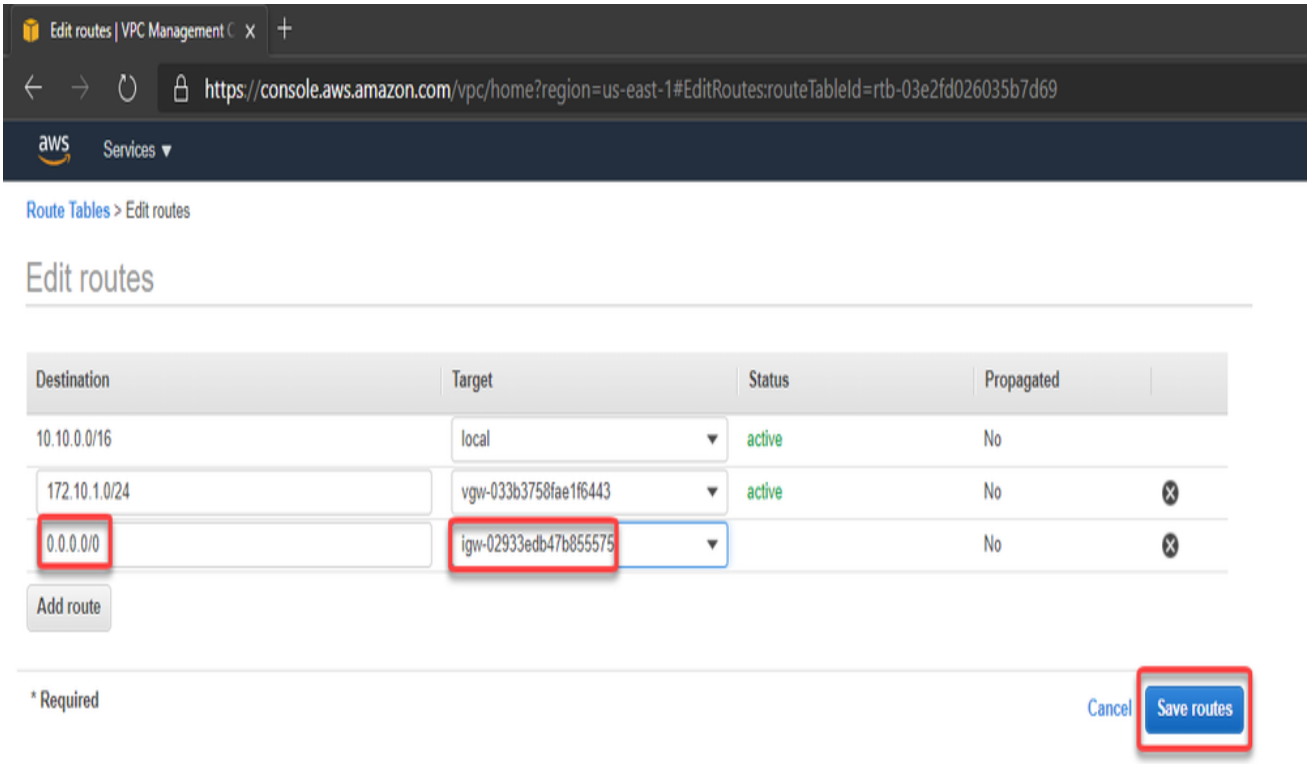


After create, let's attach to the VPC:

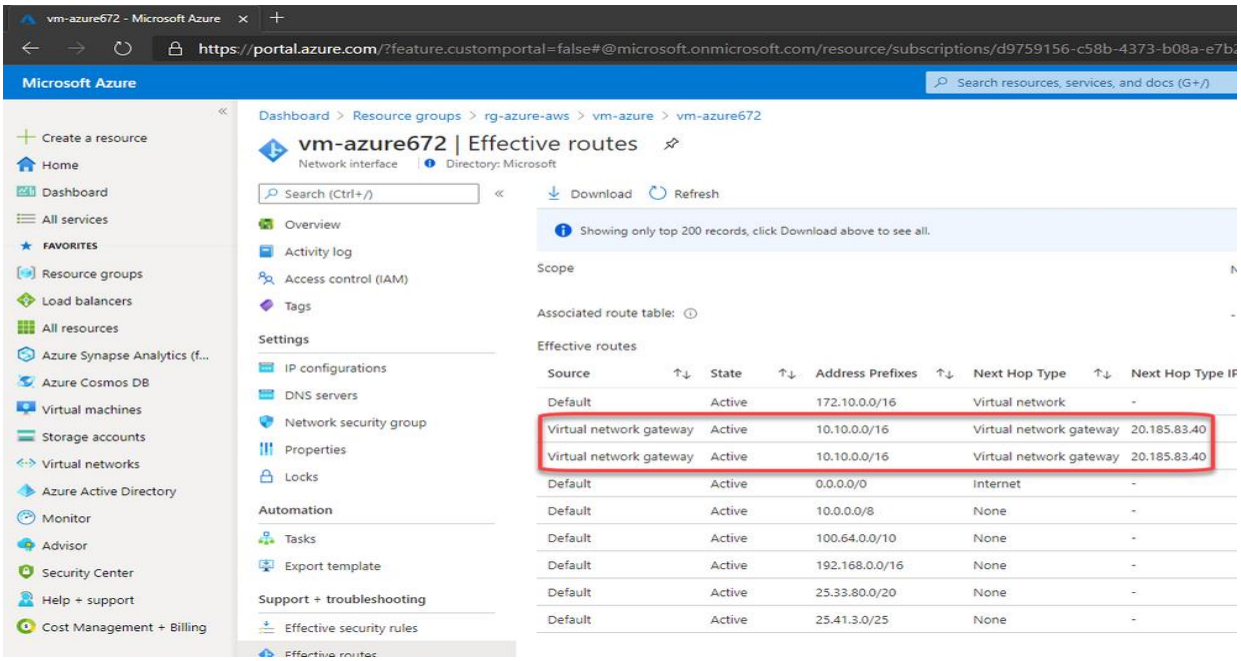

New VPC Experience
Tell us what you think
VPC Dashboard New
Filter by VPC:
Select a VPC
VIRTUAL PRIVATE CLOUD
Your VPCs New
Subnets
Route Tables
Internet Gateways New
Egress Only Internet Gateways New
Carrier Gateways New
DHCP Options Sets New
VPC > Internet gateways > igw-02933edb47b855575
igw-02933edb47b855575 / my-internet-gateway
Actions
Attach to VPC
Detach from VPC
Manage tags
Delete
Details Info
Internet gateway ID
igw-02933edb47b855575
State
Detached
VPC ID
-
Owner
292638880518
Tags
Manage tags
Search tags
1
Key
Value
Name
my-internet-gateway


Attach internet gateway | VPC M
https://console.aws.amazon.com/vpc/home?region=us-east-1#AttachInternetGateway
aws
Services
VPC > Internet gateways > Attach to VPC (igw-02933edb47b855575)
Attach to VPC (igw-02933edb47b855575) Info
VPC
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.
Available VPCs
Attach the internet gateway to this VPC.
Select a VPC
vpc-010cbb755e43966b6 - my-vpc-01
AWS Command Line Interface command
Cancel
Attach internet gateway

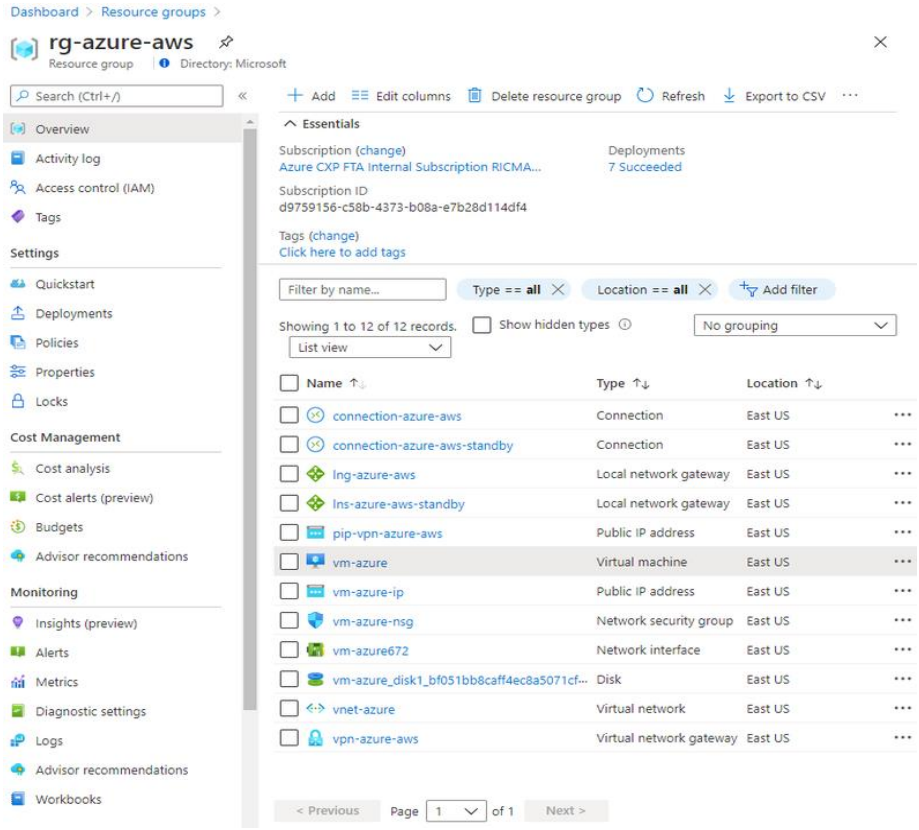
Now we can create a route to allow connections to **0.0.0.0/0** (Internet) through the Internet Gateway:



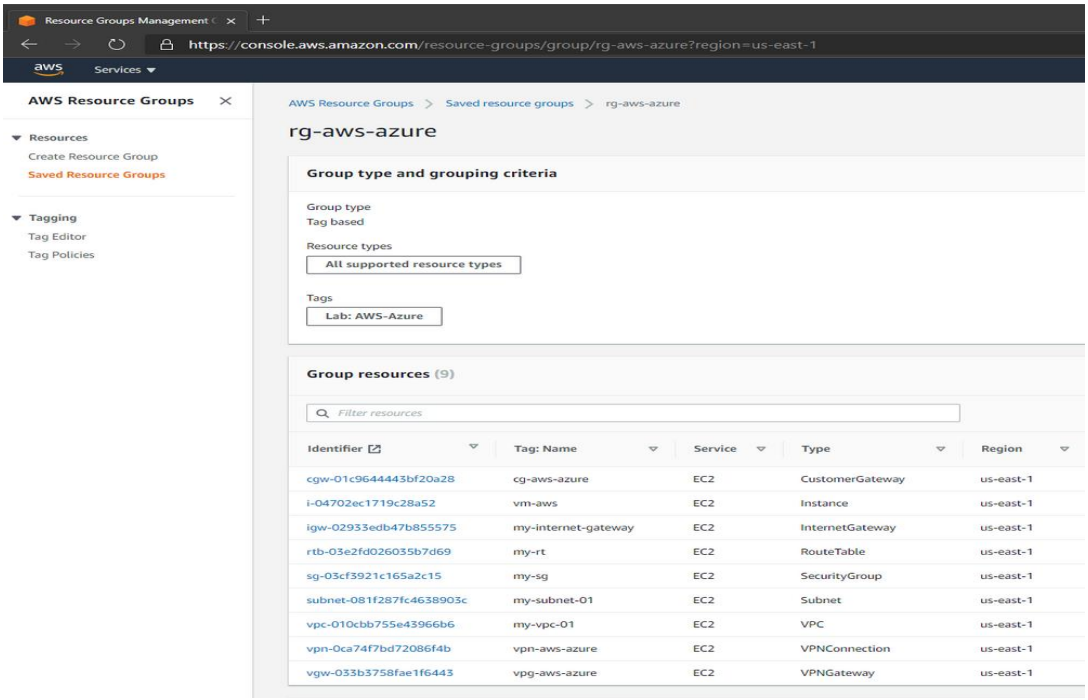
On Azure the route was automatically created. You can check selecting the Azure VM > Networking > Network Interface > Effective routes. Note that we have 2 (1 per connection):



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 between Azure and AWS through our VPN connection:

11.0.0.0/16

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.10.1.4  netmask 255.255.255.0  broadcast 172.10.1.255
        inet6 fe80::20d:3aff:fe8d:1be9  prefixlen 64  scopeid 0x20<link>
        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

mmartins@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
```

AWS Linux VM

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.10.1.49  netmask 255.255.255.0  broadcast 10.10.1.255
        inet6 fe80::cd3:92ff:fe9b:a2bd  prefixlen 64  scopeid 0x20<link>
        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

~$ 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
```