Startups at Microsoft

STARTUPS AT MICROSOFT 6 MIN READ

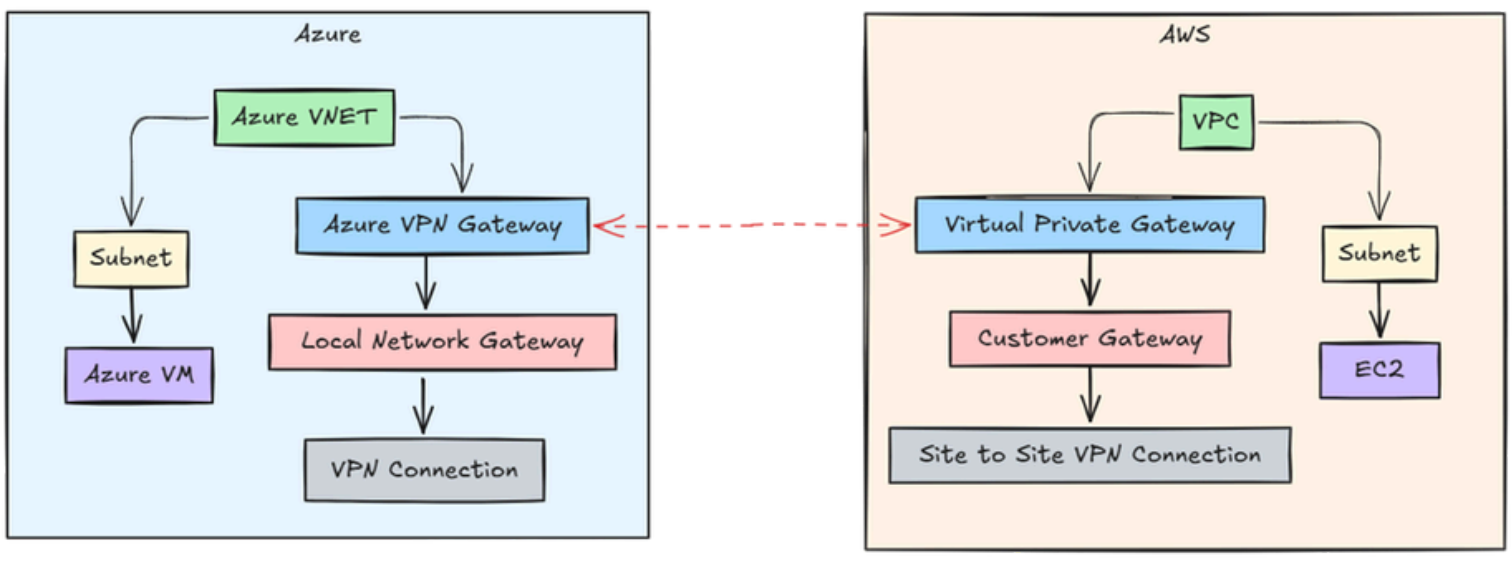
How to easily set up a VPN between Azure and AWS using managed services (Updated 2024)

rmmartins MICROSOFT
Oct 25, 2024

Introduction

Setting up a secure VPN connection between **Azure** and **AWS** doesn't have to be complicated. In this guide, we'll demonstrate an easy and straightforward method to establish a multicloud static VPN using only **managed services**—no need to manage virtual machines or third-party appliances. This method provides a quick and reliable way to connect **Azure VPN Gateway** and **AWS Virtual Private Gateway** over IPsec tunnels (without BGP config), ensuring secure communication between the two environments.

This post is an updated version of a similar guide I published three years ago, reflecting changes in services and adding valuable troubleshooting tips to streamline the process.



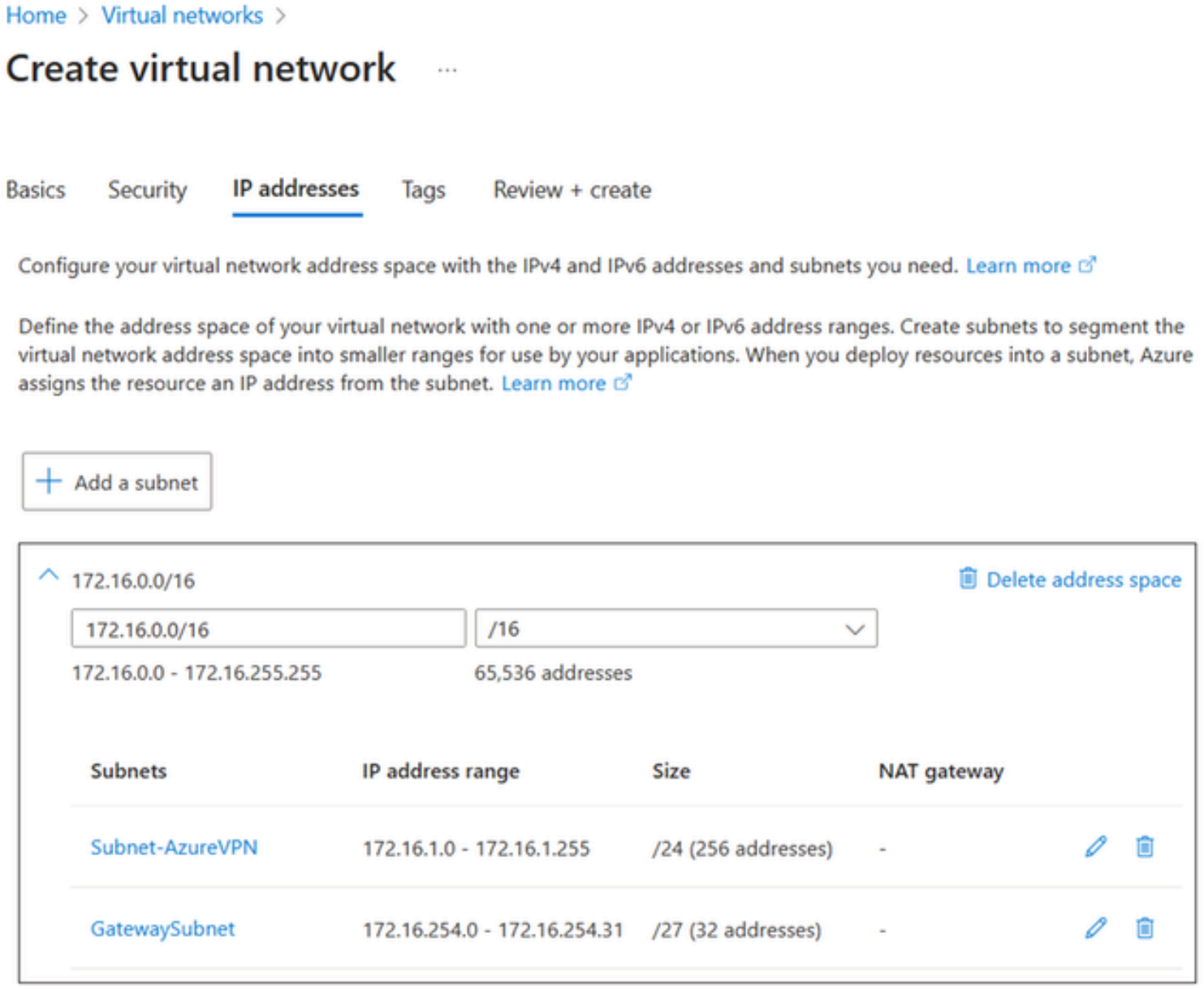
For more advanced scenarios, such as integrating dynamic routing with **BGP (Border Gateway Protocol)** to support automatic route exchanges, we recommend referring to the official Azure VPN Gateway Documentation for in-depth guidance.

1.1. Create a Resource Group

1. Go to **Azure Portal** > **Resource groups** > **Create**.
2. Select your subscription and region, and give the resource group a name like **RG-AzureAWSVPN**.

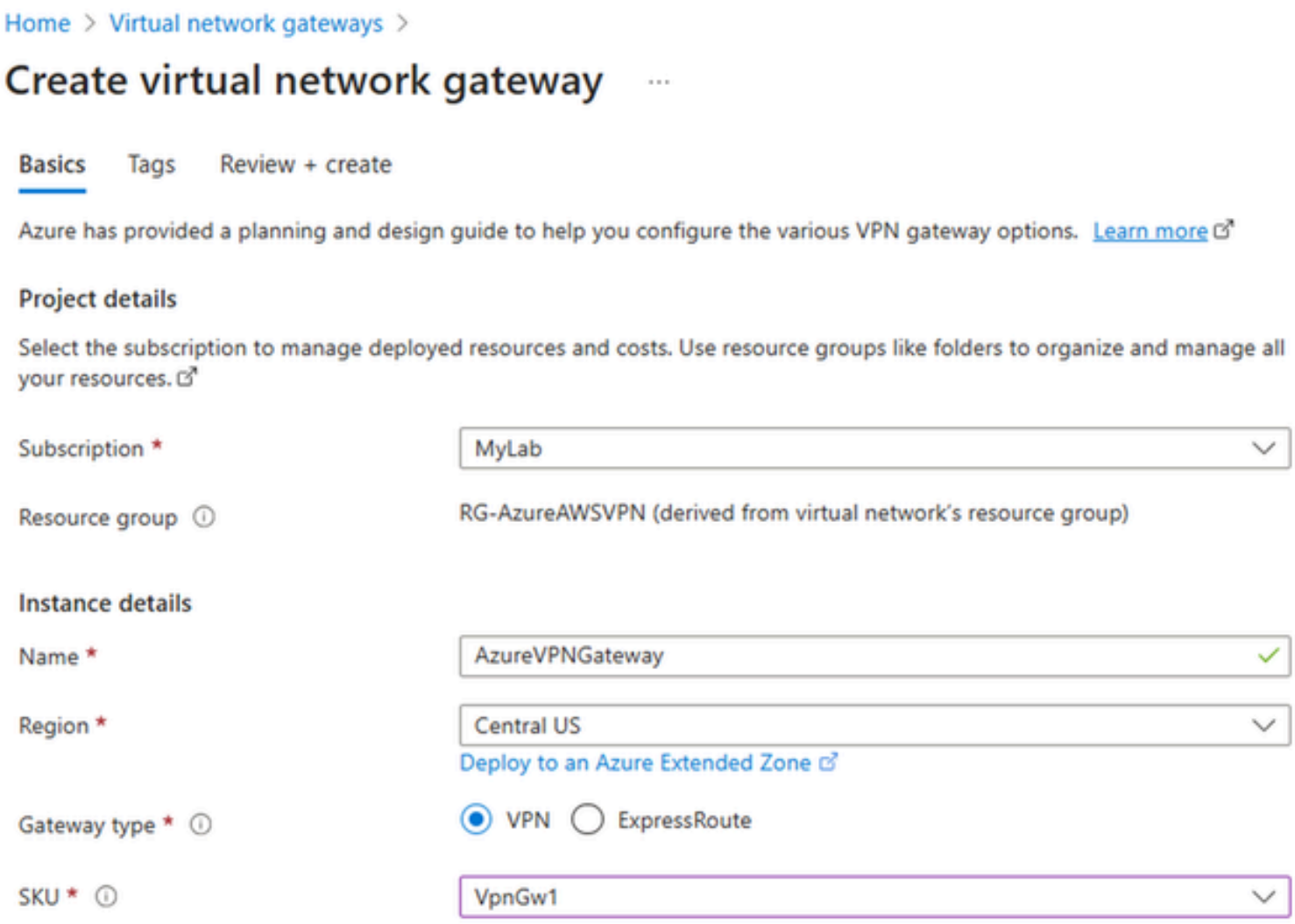
1.2. Create a Virtual Network (VNet) and Subnet

1. In the **Azure Portal**, go to **Virtual Networks** > **Create**.
2. Name the VNet **AzureVNet** and specify an address space of **172.16.0.0/16**.
3. Under **Subnets**, create a subnet named **Subnet-AzureVPN** with the address range **172.16.1.0/24**.
4. Add a **GatewaySubnet** with a /27 address block (e.g.,) for the VPN gateway.



1.3. Set Up the Azure VPN Gateway

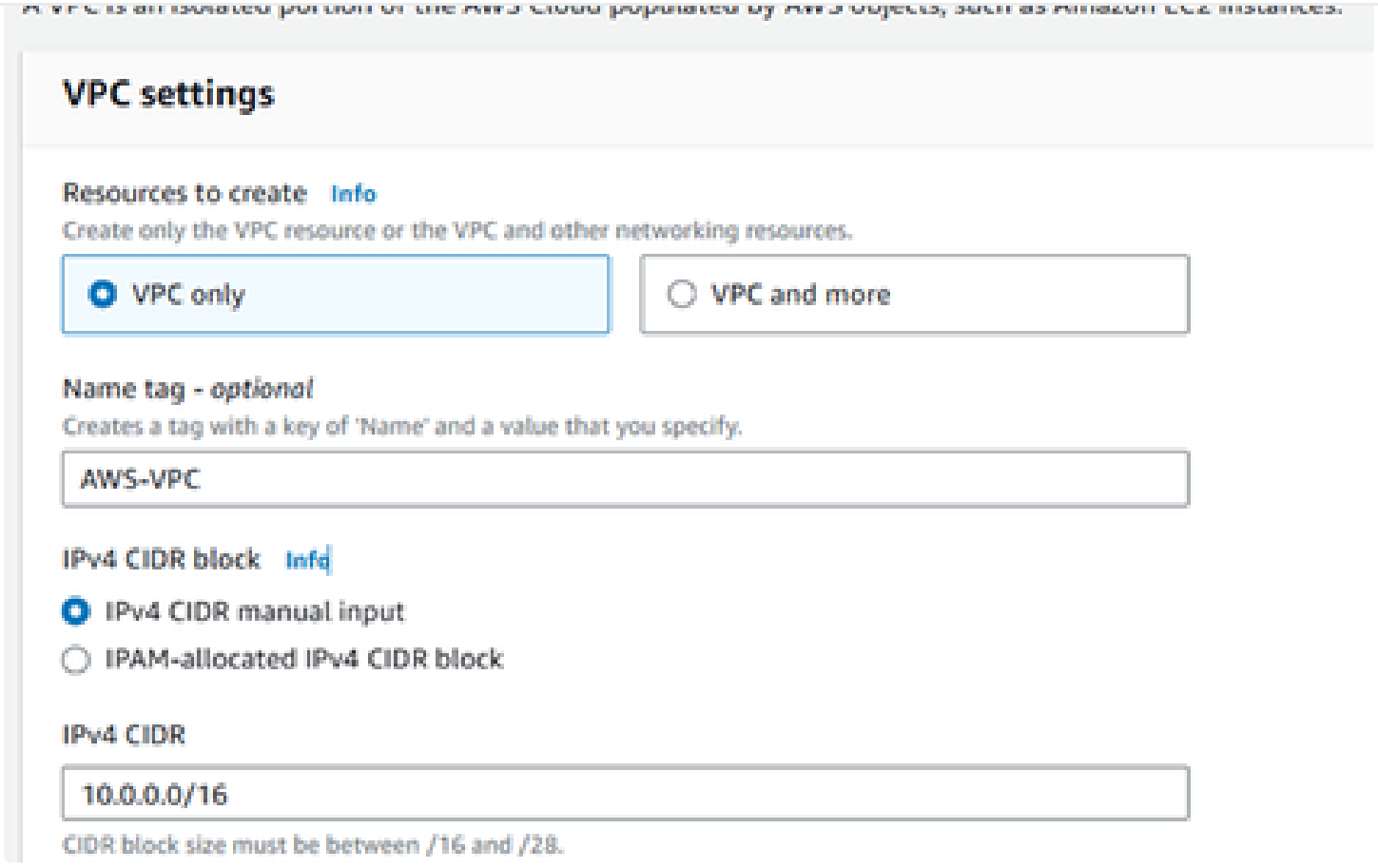
1. Go to **+Create a resource**, search for **Virtual Network Gateway**, and select **Create**.
2. Fill in the details:
    - **Name**: *AzureVPNGateway*
    - **Gateway Type**: **VPN**
    - **SKU**: **VpnGw1** (or higher if needed)
    - **Public IP Address**: Create a new one and name it *AzureVPNGatewayPublicIP*.
    - **Active-Active Mode**: Leave **disabled** unless high availability is required.
        - If you need to ensure High Availability, enabling the Active-Active mode will made needed the following additional configurations on the Azure side:
            - Create a second Public IP Address for the Virtual Network Gateway
            - Create a second Local Network Gateway pointing to the public IP address of the Tunnel 2 on AWS side
            - Create a second VPN connection pointing to the Tunnel 2 on AWS side
    - **Configure BGP:** Leave disabled for this lab



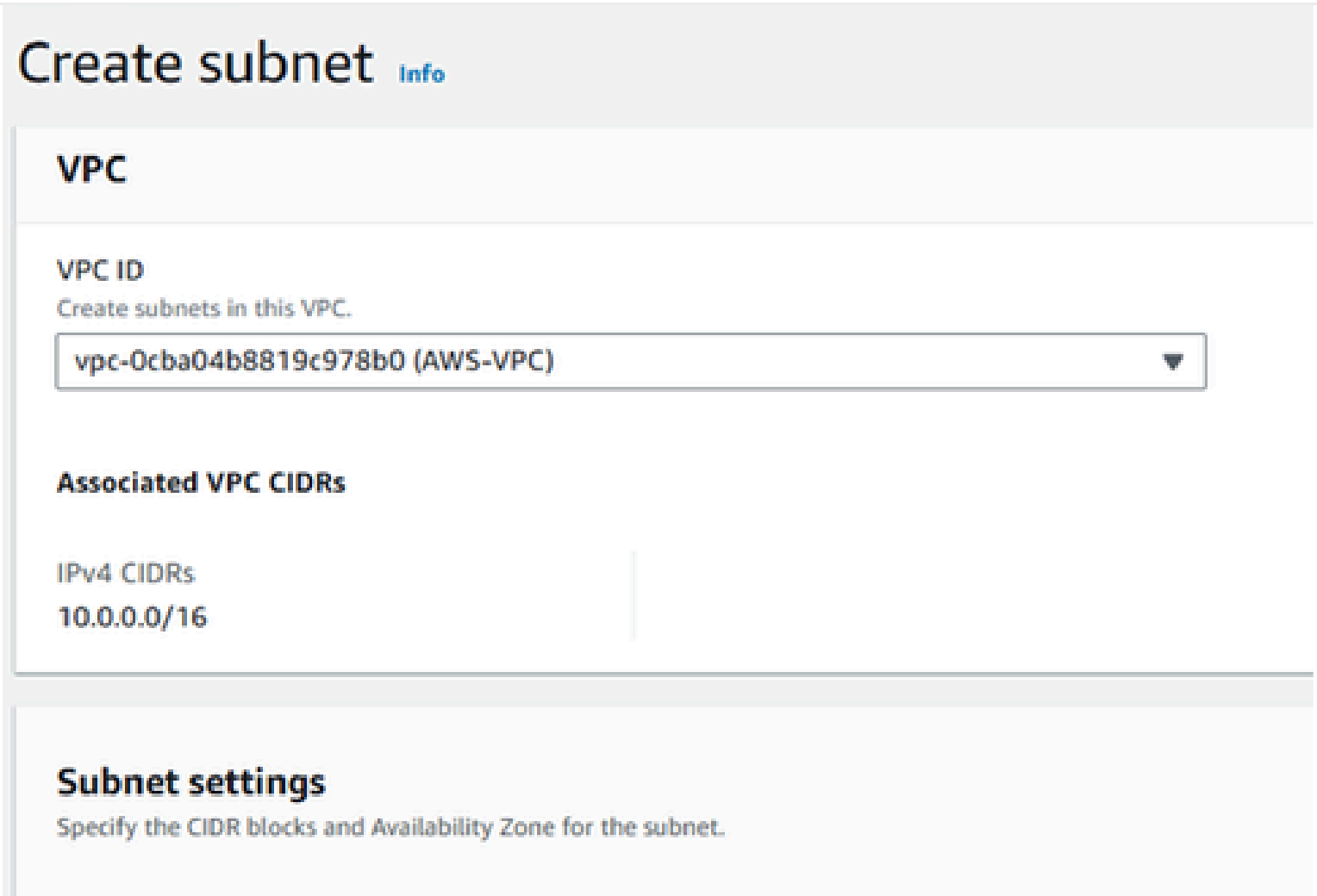
Step 2: Set Up Your AWS Environment

2.1. Create a VPC and Subnet in AWS

1. In the **AWS Console**, go to **VPC** > **Create VPC**.
2. Use an address space (e.g., *10.0.0.0/16*) for the **AWS-VPC**.

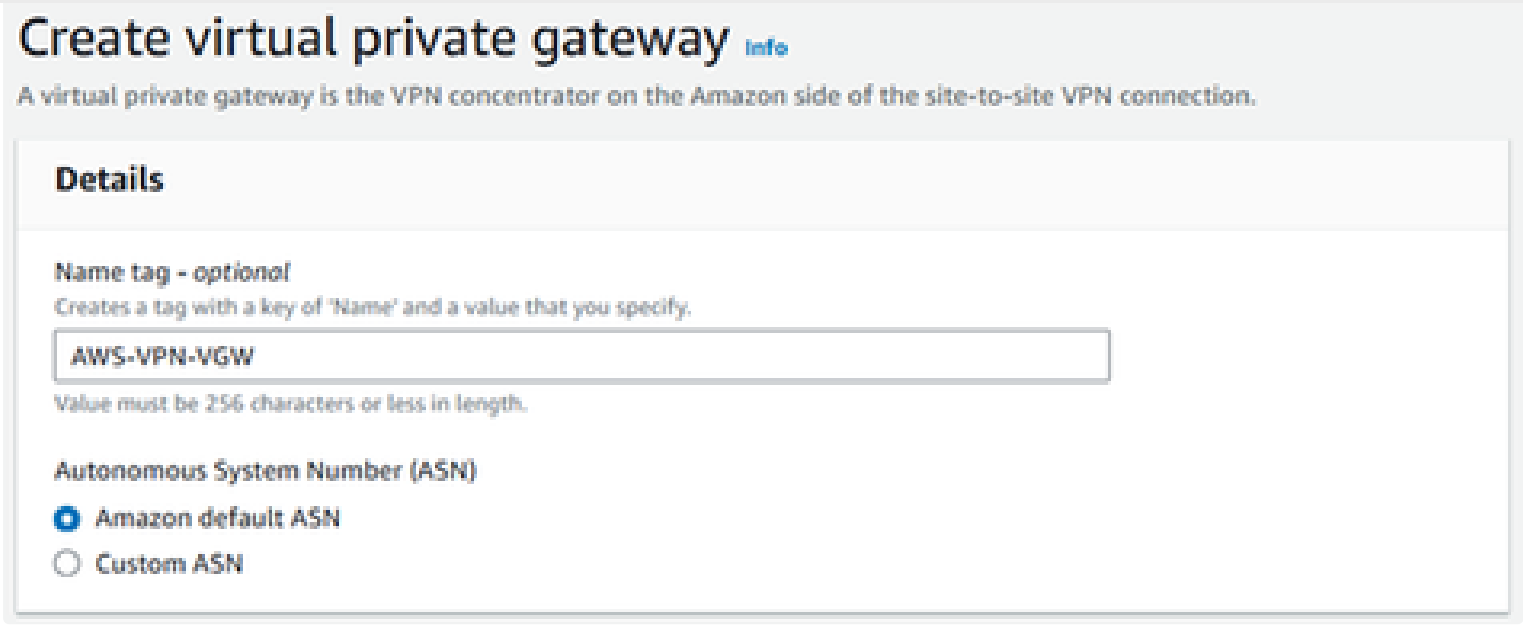


3. Under **Subnets**, create a subnet with a name like **Subnet-AWSVPN** and the address space **10.0.1.0/24** for your subnet.

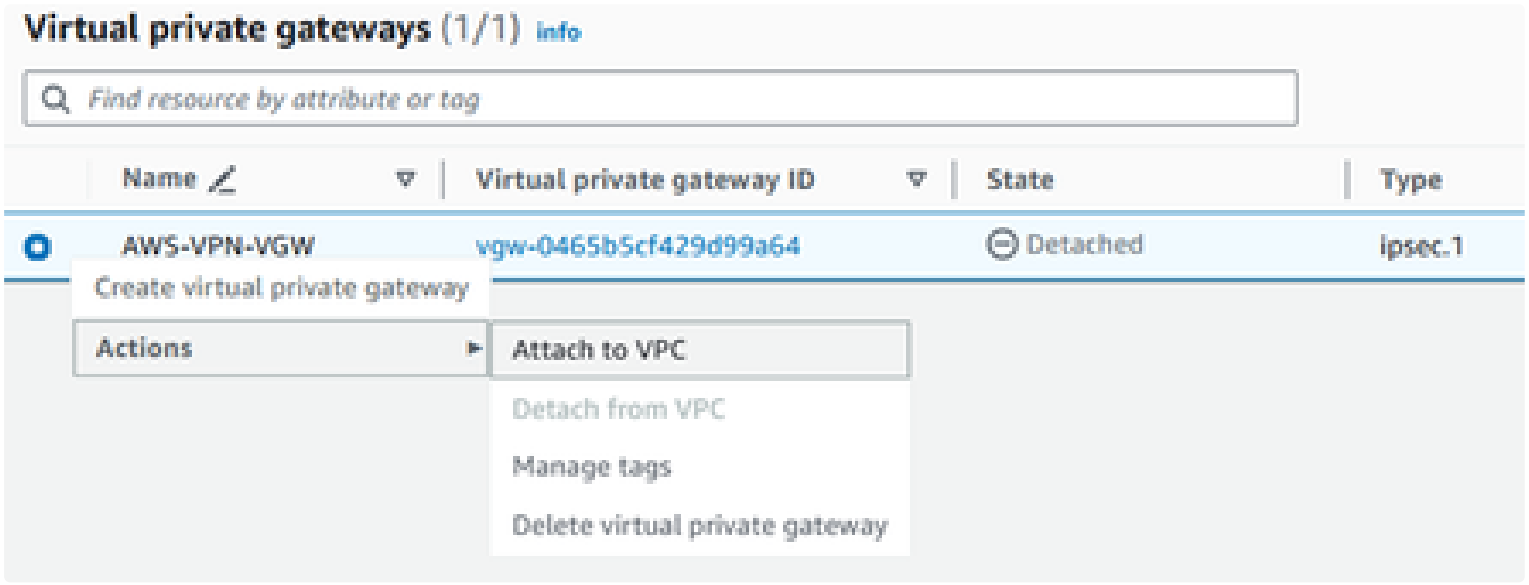


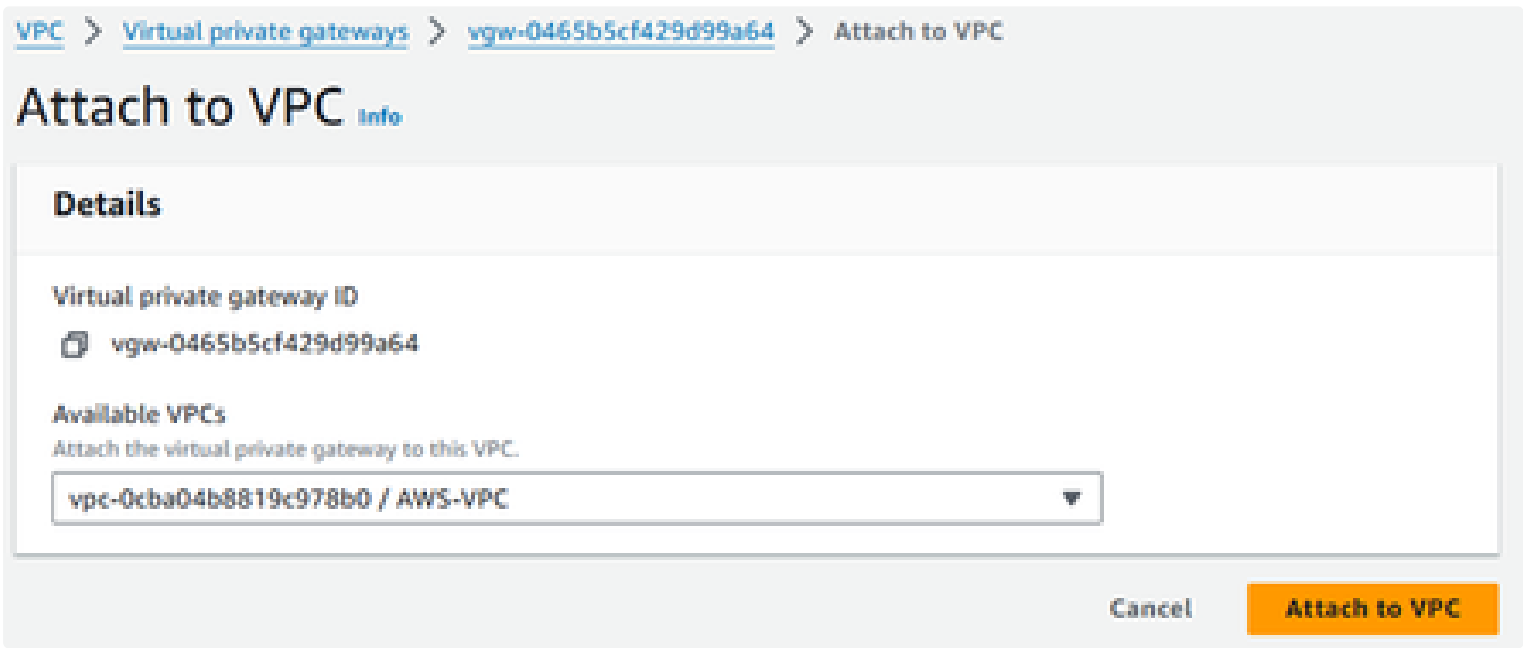
2.2. Create an AWS Virtual Private Gateway (VGW)

1. In the **AWS VPC Console**, go to **Virtual Private Gateway** and create a new VGW named *AWS-VPN-VGW*.



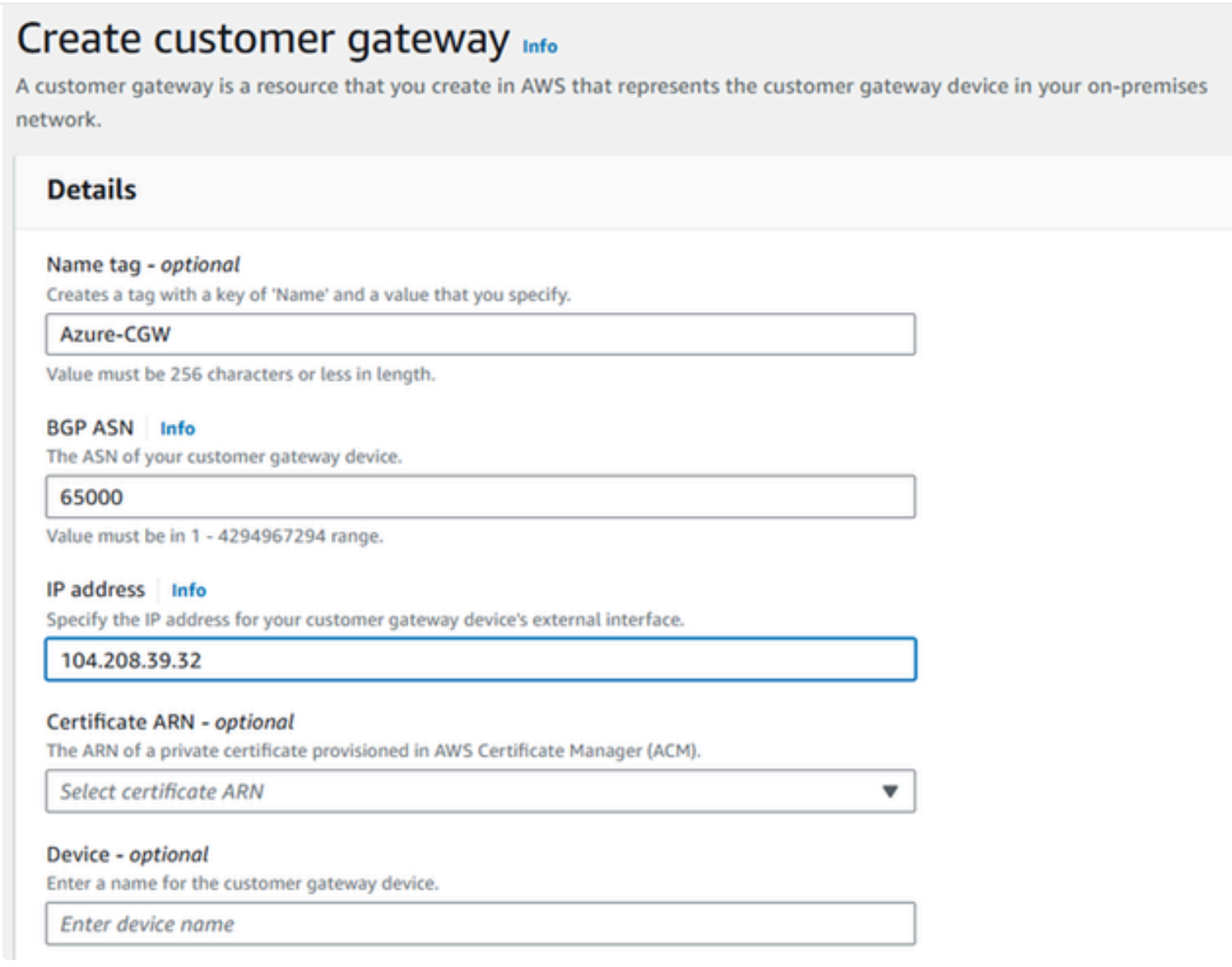
2. Attach the VGW to the **VPC**.





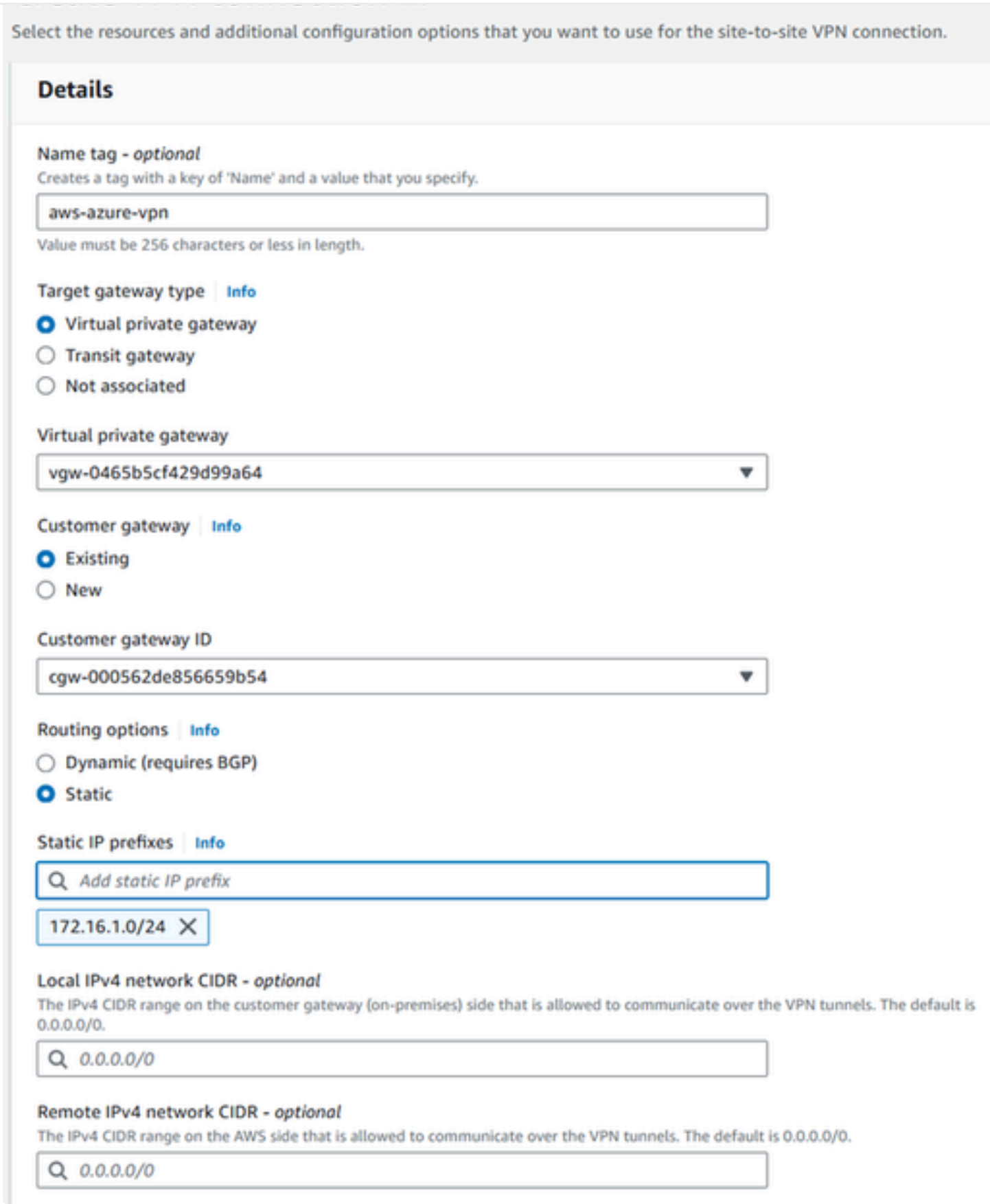
2.3. Set Up a Customer Gateway (CGW)

1. In the **AWS Console**, go to **Customer Gateway**, and create a CGW using the **public IP** of the Azure VPN Gateway. Name it **Azure-CGW**.

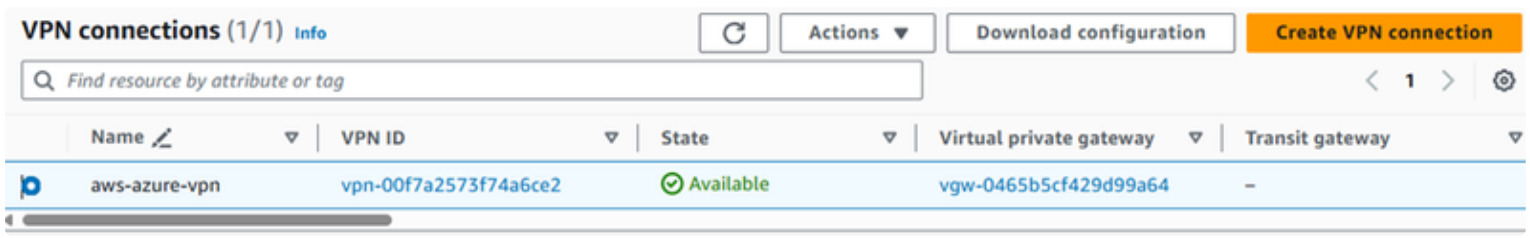


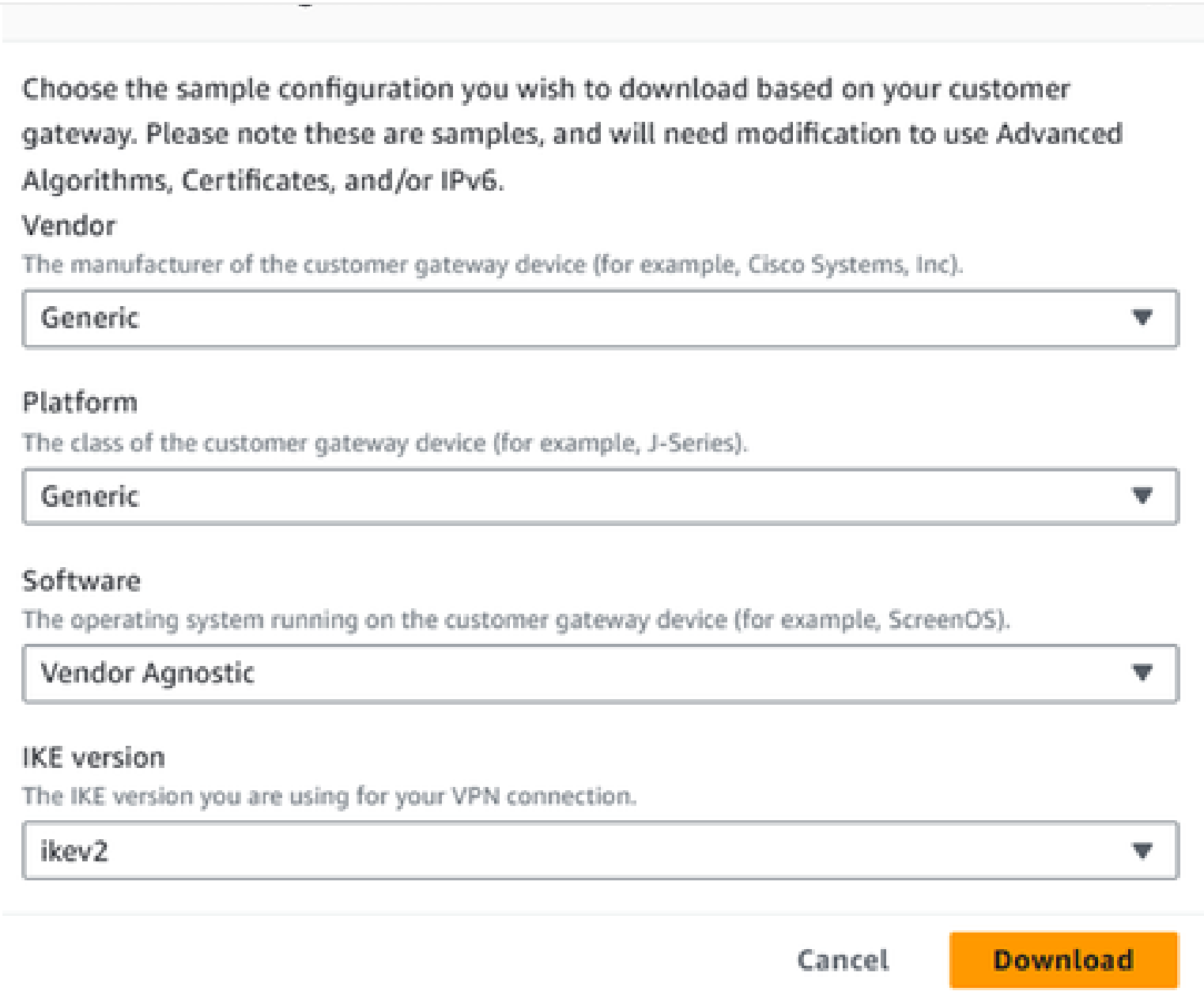
2.4. Create the Site-to-Site VPN Connection

1. In **AWS Console**, go to **Site-to-Site VPN Connections** > **Create VPN Connection**.
2. Select the **Virtual Private Gateway** created earlier.
3. Select the **Customer Gateway** created earlier.
4. Set **Routing** as **Static**, and define the **Azure VNet subnet (172.16.1.0/24)** as the static route.



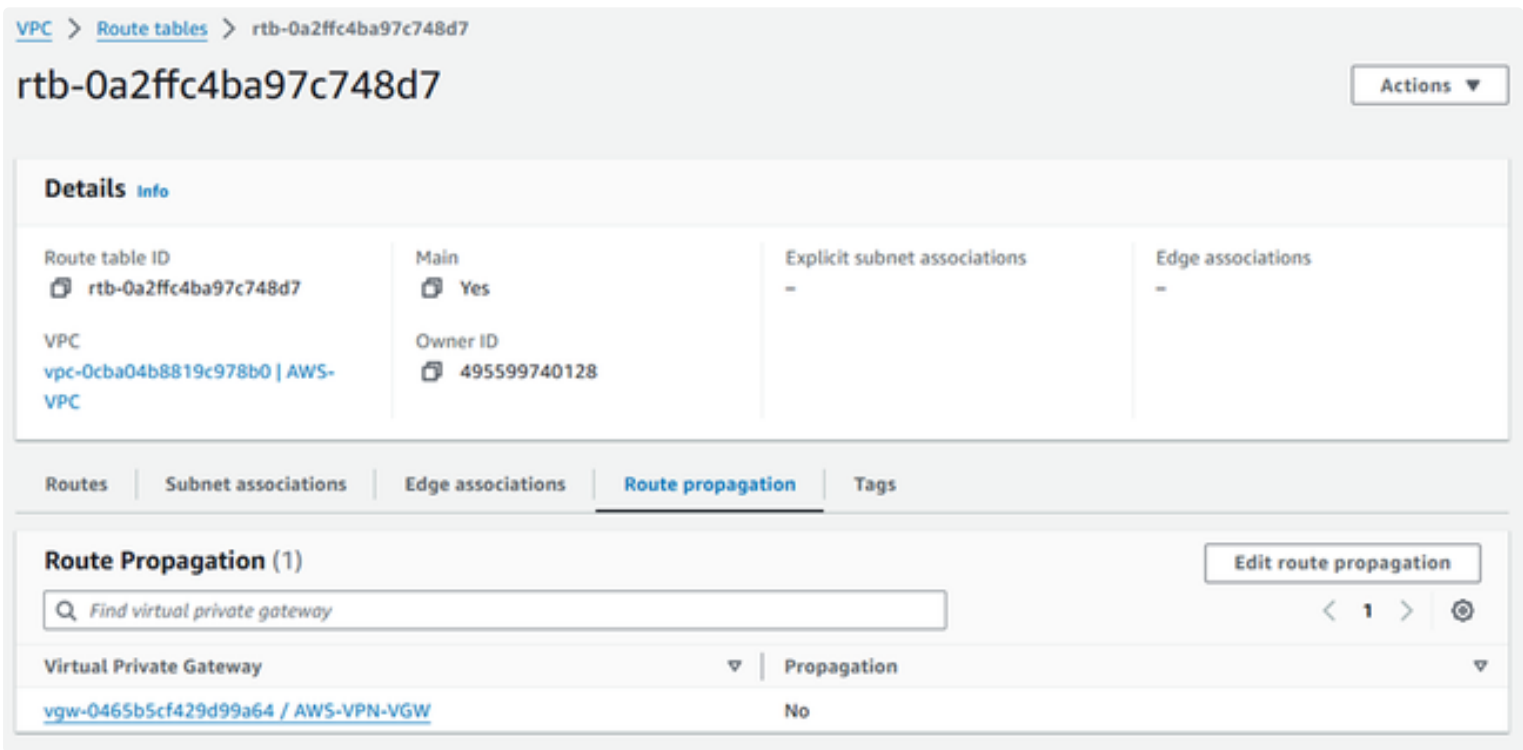
5. Download the VPN Configuration File
    1. After the VPN is set up, download the configuration file.
    2. Select **Generic** for the platform and **Vendor agnostic** for the software.
    3. Select **IKEv2** for the IKE version.

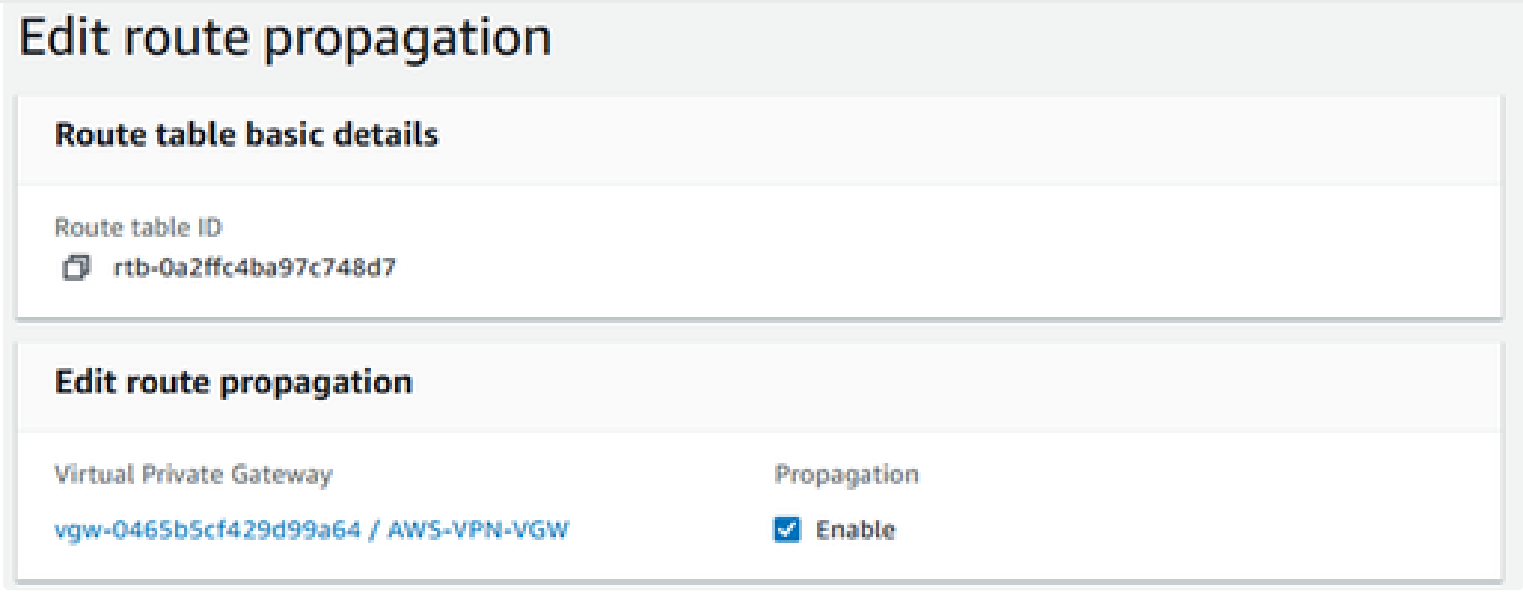




2.5. Enable Route Propagation

After creating the VPN connection, go to Route Tables > Select the existing route table > Route Propagation > Edit Route Propagation, and enable propagation for the VGW.

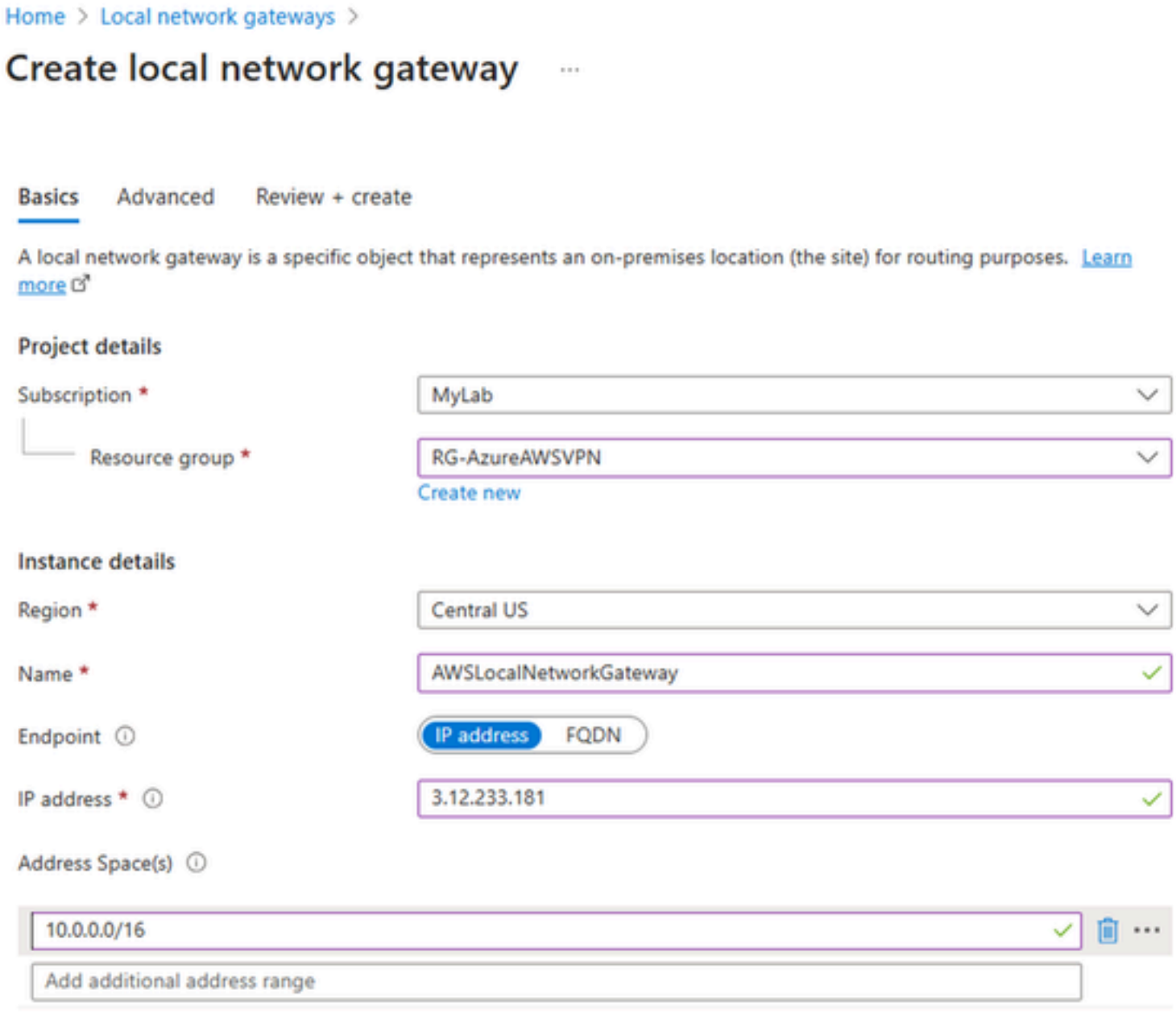




Step 3: Finish the Azure Side Configuration

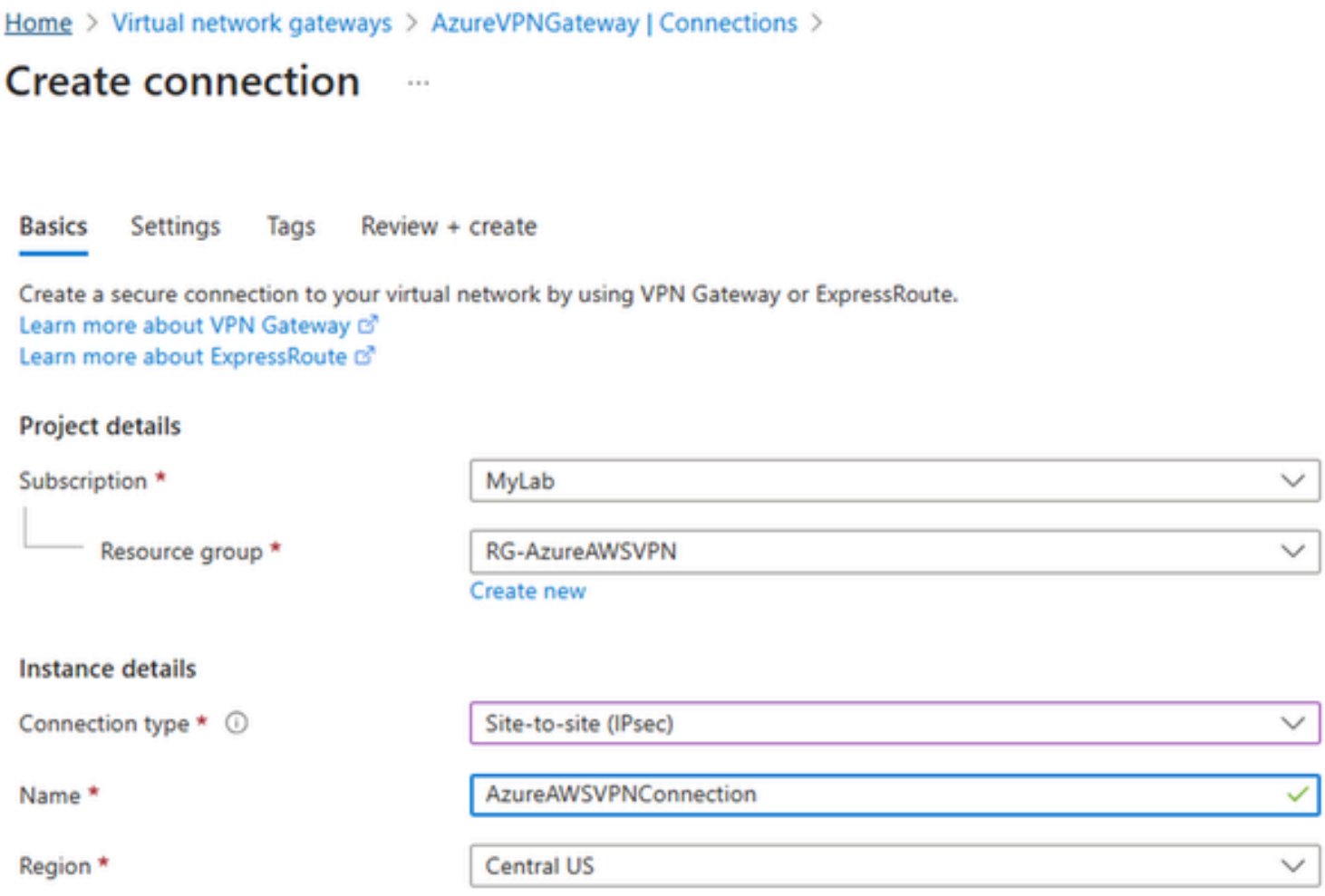
3.1. Create the Local Network Gateway

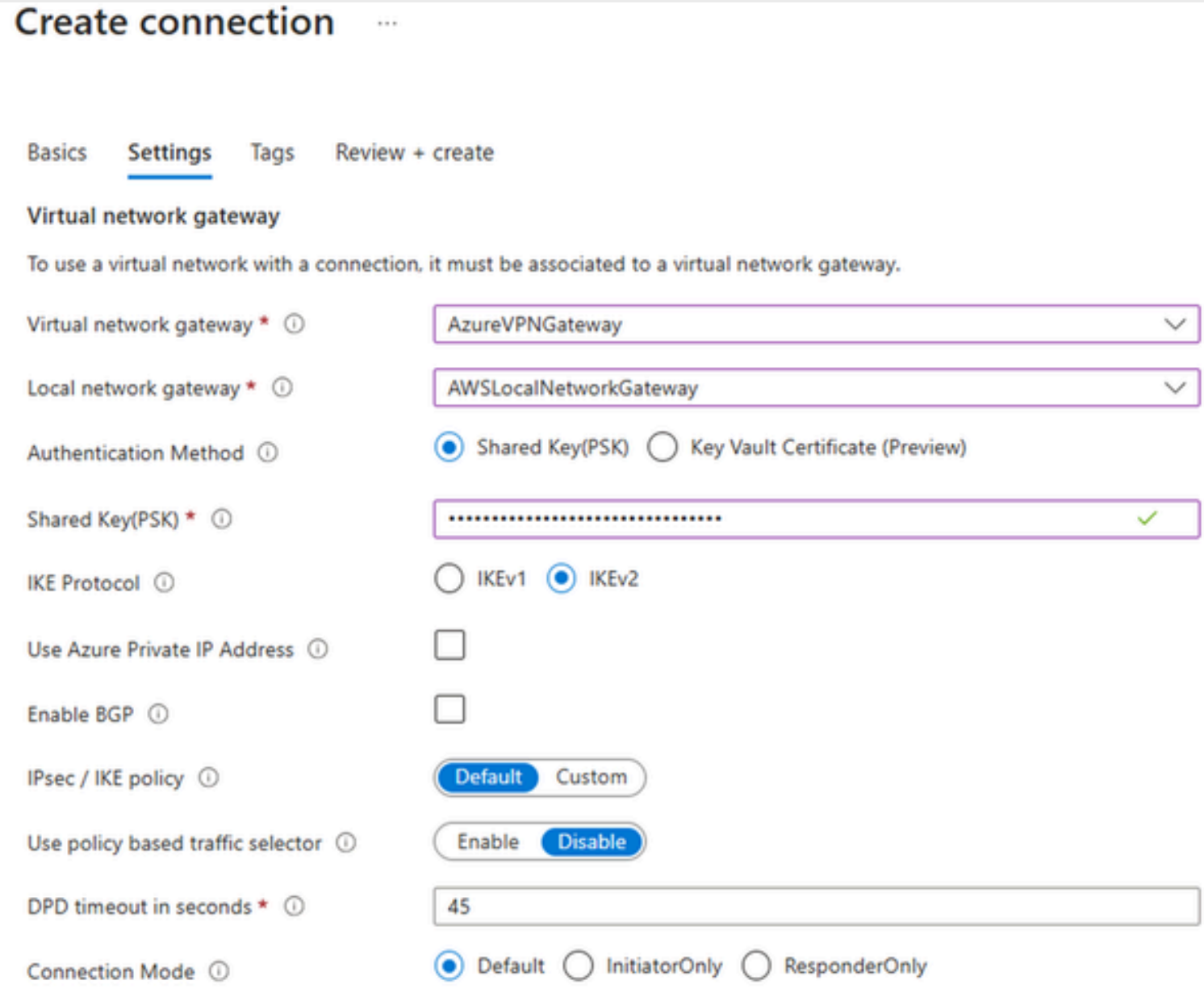
1. In the **Azure Portal**, go to **Local Network Gateway** > **Create**.
2. Name the gateway **AWSLocalNetworkGateway**, and enter the **public IP** of the AWS VPN tunnel (from the configuration file).
3. Set the **AWS VPC CIDR block** (e.g., *10.0.0.0/16*) as the address space.
4. In the next tab (Advanced), leave the option **Configure BGP Settings** defined to **No**



3.2. Create the VPN Connection

1. Go to **Azure Portal** > **Virtual Network Gateway** > **Connections** > **+ Add**.
2. Configure the connection:
    - **Name**: *AzureAWSVPNConnection*
    - **Connection Type**: **Site-to-site (IPsec)**.
    - **Virtual Network Gateway**: Select **AzureVPNGateway**.
    - **Local Network Gateway**: Select **AWSLocalNetworkGateway**.
    - **Shared Key (PSK)**: Use the shared key from the AWS VPN configuration file.
    - **IKE Protocol**: Set to **IKEv2**.
    - **IPsec/IKE Policy**: Use **Default**, or configure **custom policies** per AWS (AES128, SHA1, DH Group 2).
    - **DPD Timeout**: Set to **45 seconds**.
    - **Connection Mode**: Leave as **Default** unless specific behavior is required.

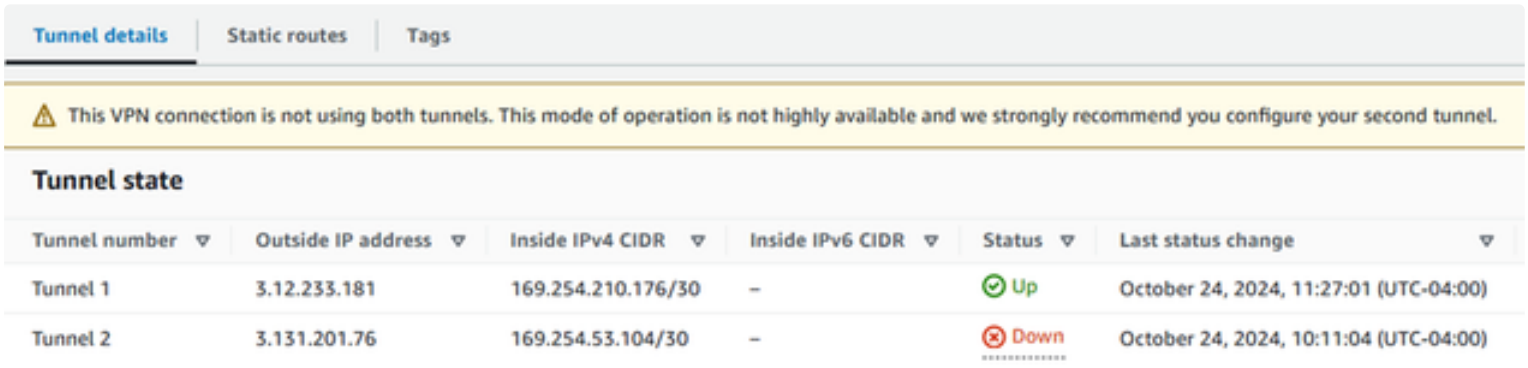




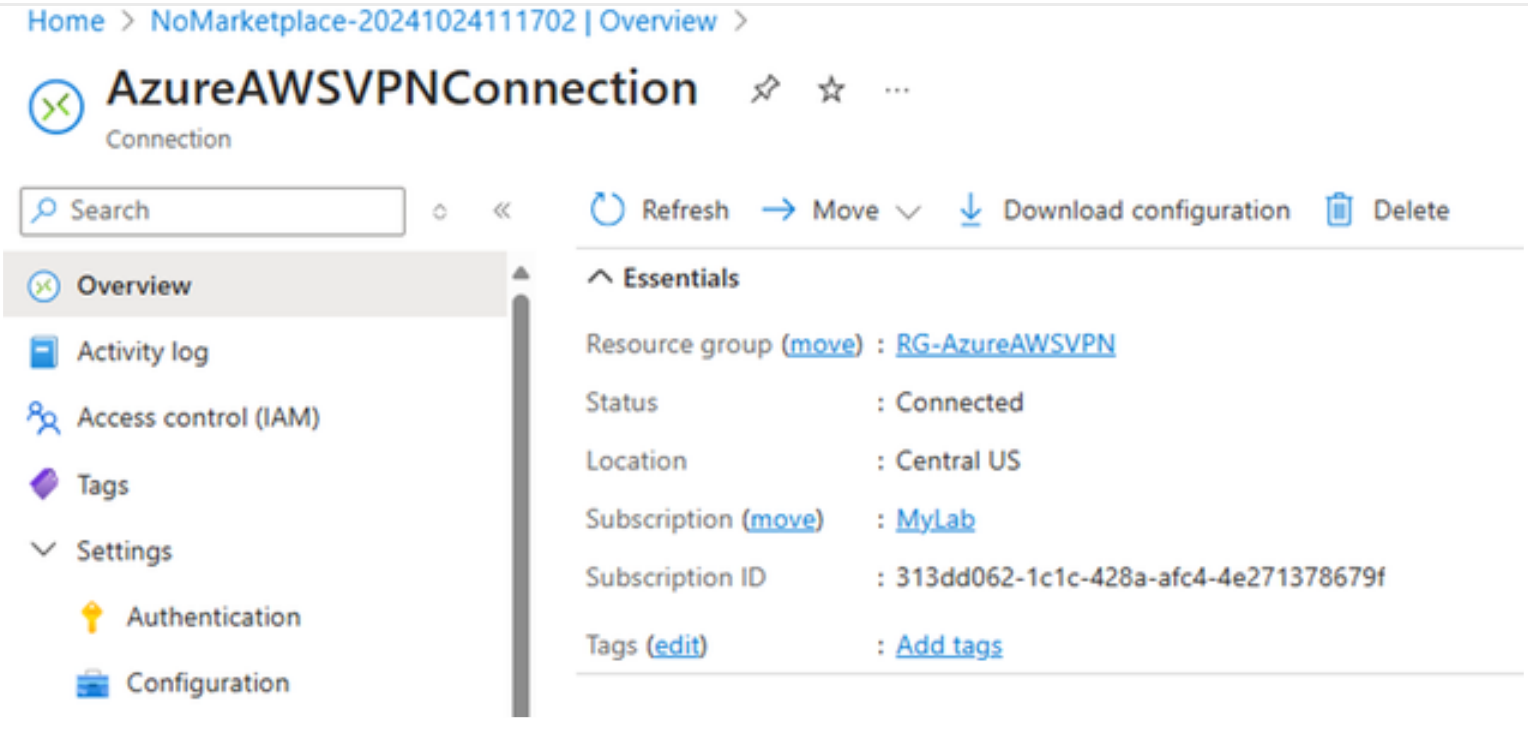
In about 5 minutes, you can check the VPN connection established.

3.3. Ensure the VPN is established

1. From Site-to-Site VPN connections on AWS, go to Tunnel details and check that the Tunnel 1 is UP:



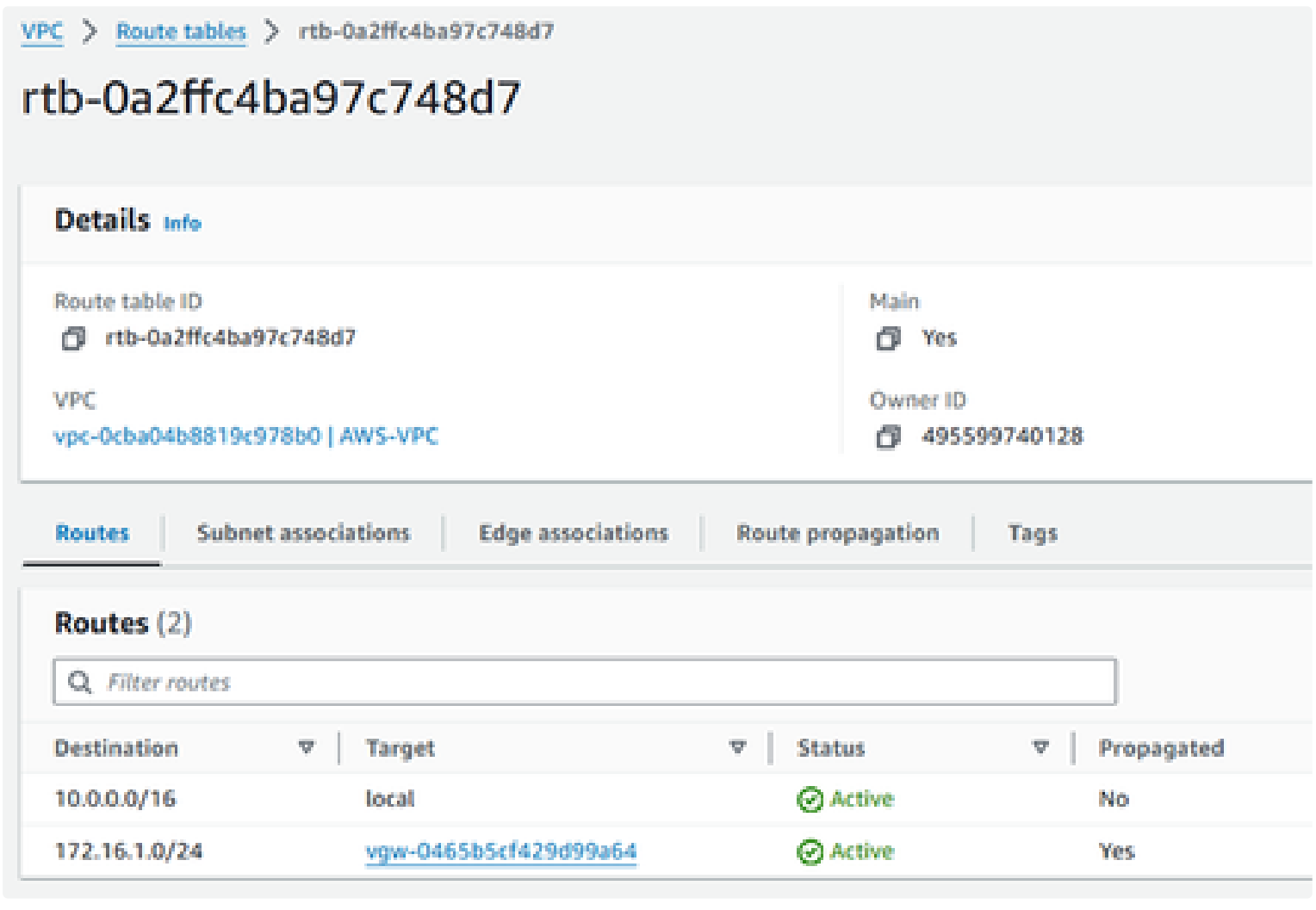
2. From Azure side, check if the status of the VPN connection is Connected:



Step 4: Add Routes and Configure Security

4.1. Check the Route for Azure Subnets in AWS Route Table

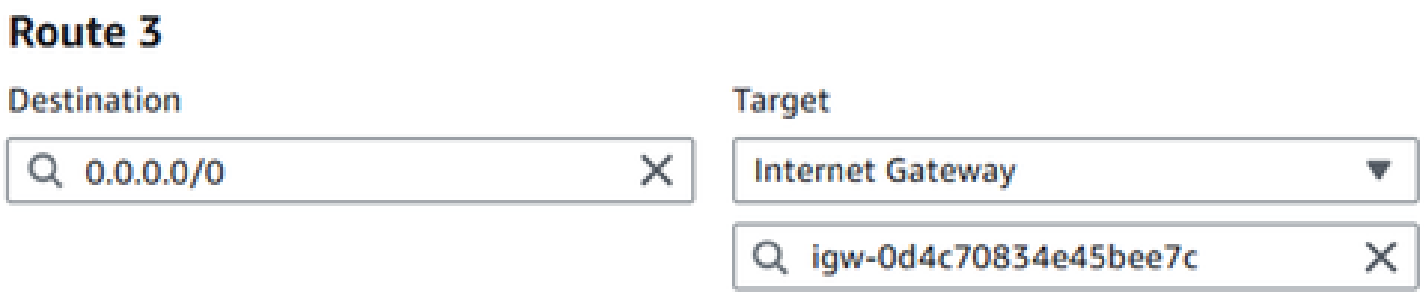
1. In the **AWS Console**, go to **VPC** > **Route Tables**.
2. Check if the AWS Route Table has a route for the **Azure VNet subnet (172.16.1.0/24)** with the **VGW** as the target. If the route propagation enabled before was done correctly, you should be able to see the routes to Azure subnet (172.16.1.0/24) automatically added:



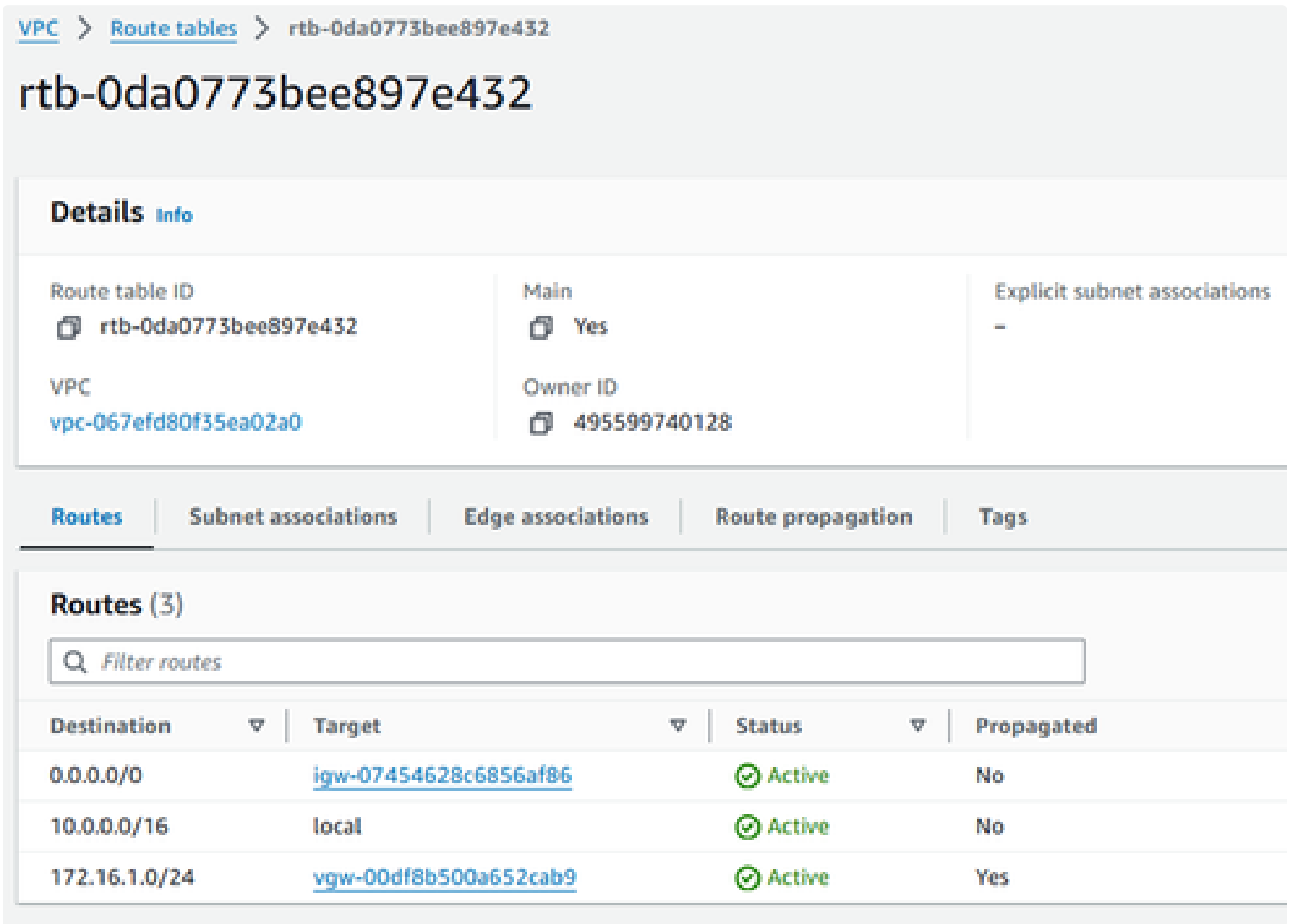
4.2. Add an Internet Gateway (IGW)

Note: *An Internet Gateway (IGW) is required for the EC2 instance to be accessible via its public IP address. Without the IGW, the EC2 instance won't be reachable over the public internet, preventing you from logging into the EC2 using their public IP address. This is the sole purpose of deploying the IGW.*

AWS-IGW, then attach it to the AWS VPC.
2. **Update Route Table**: Add a route to **0.0.0.0/0** pointing to the IGW for external connectivity.



After adding the new route, you should have 3 routes as below:



4.3. Set Security Group and NSG Rules

1. **AWS Security Group**: Ensure the **Security Group** for the AWS EC2 instance allows **ICMP (ping)** and other protocols (e.g., SSH) from Azure.
2. **Azure NSG**: Similarly, ensure the **NSG** attached to the Azure VM's NIC allows inbound traffic from AWS.

Step 5: Test Connectivity Between Azure and AWS VMs

To test connectivity between Azure and AWS, first deploy a virtual machine in the appropriate subnet on each cloud provider—an EC2 instance on AWS and a VM on Azure. Once both machines are running, connect to each VM using their respective public IP addresses. After logging in, use the private IP addresses of both instances to run a ping test and verify private network connectivity between them.

the Azure VM using their public IP address and test unilaterally running the ping command against the private IP of the EC2 VM.

5.1. Ensure ICMP Traffic Is Allowed

Both the **AWS Security Group** and **Azure NSG** should allow ICMP (ping) traffic.

5.2. Test Connectivity with ping

1. From the **Azure VM**, ping the **AWS VM** using its private IP:

```
rmmartins@vm01:~$ ping 10.0.1.177
PING 10.0.1.177 (10.0.1.177) 56(84) bytes of data.
64 bytes from 10.0.1.177: icmp_seq=1 ttl=64 time=21.9 ms
64 bytes from 10.0.1.177: icmp_seq=2 ttl=64 time=22.9 ms
64 bytes from 10.0.1.177: icmp_seq=3 ttl=64 time=21.3 ms
64 bytes from 10.0.1.177: icmp_seq=4 ttl=64 time=21.3 ms
^C
--- 10.0.1.177 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 21.306/21.875/22.941/0.663 ms
rmmartins@vm01:~$
```

2. From the **AWS VM**, ping the **Azure VM** using its private IP:

```
ubuntu@ip-10-0-1-177:~$ ping 172.16.1.4
PING 172.16.1.4 (172.16.1.4) 56(84) bytes of data.
64 bytes from 172.16.1.4: icmp_seq=1 ttl=64 time=20.8 ms
64 bytes from 172.16.1.4: icmp_seq=2 ttl=64 time=21.7 ms
64 bytes from 172.16.1.4: icmp_seq=3 ttl=64 time=21.7 ms
64 bytes from 172.16.1.4: icmp_seq=4 ttl=64 time=21.4 ms
^C
--- 172.16.1.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 20.837/21.401/21.736/0.351 ms
ubuntu@ip-10-0-1-177:~$
```

Troubleshooting Common Issues

1. Missing Static Route in AWS VPN

- Ensure that the **static route** for the **Azure VNet subnet (172.16.1.0/24)** is added in the AWS VPN configuration. Without this route, AWS will not know to send traffic to Azure through the VPN.

2. No Inbound Traffic on Azure VPN Gateway

3. Custom IPsec/IKE Policies

- If the default policies aren't working, apply **custom IPsec/IKE policies** based on AWS configuration (AES128, SHA1, DH Group 2 for Phase 1 and Phase 2).

4. Further Troubleshooting

- For additional troubleshooting guidance, refer to the official Azure VPN diagnostics documentation: Troubleshoot VPN with Azure Diagnostics.

Conclusion

By following this guide, you've successfully set up a VPN connection between **Azure** and **AWS** using managed services. Ensuring that the route for Azure's subnet is added to the AWS Route Table is crucial for proper communication between the two clouds. If you need more advanced configurations, such as **BGP** for dynamic routing, consult the Azure VPN Gateway documentation.

Updated Oct 28, 2024 VERSION 3.0

Microsoft Store

Account profile

Download Center

Microsoft Store support

Returns

Order tracking

Virtual workshops and training

Microsoft Store Promise

Flexible Payments

Education

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