Lab 4 – AWS – Transit Gateway with BGP

**Goal** – Utilize dynamic routing with Transit Gateway and FortiGates.

**Task** – Create attachment associations + propagations then configure FortiGate routes and firewall policies to allow secured traffic to pass.

**Validation** – Confirm outbound and east/west connectivity from EC2 Instance-A.

**Introduction**

In this task, there are multiple VPCs in the same region that have one instance each. Transit Gateway is configured with multiple Transit Gateway Route Tables. You will need to create the appropriate VPC attachment associations and propagations to the correct TGW Route Tables, FW policy and update BPG configuration on the independent FortiGates.

In this scenario, the FortiGates are completely independent of each other (not clustered, nor sharing config/sessions, etc.) and are showing different connectivity options to attach remote locations to Transit Gateway. VPN attachments can be used to connect to any IPsec capable device located anywhere. TGW Connect attachments require a private path to reach a VM deployed in a VPC or HW/VM deployed on premise and must be reachable over Direct Connect (a dedicated, private circuit).

**Topology**

Graphical user interface, application

AI-generated content may be incorrect.

**Pre-Work:**

* Delete the Lab3 stacks by navigating to the CloudFormation console and the Lab3 stack:

A screenshot of a computer

AI-generated content may be incorrect.

**Deleting the Lab3 stack will also delete all the nested stacks it created**

Graphical user interface, text, application

AI-generated content may be incorrect.

**All resources from Lab 3 must be deleted prior to beginning the next lab**

1 - Create a new stack by navigating to the CloudFormation console and importing the TGWwithBGP.json file:

* Click on **Create stack**
* Select **With new resources**

A screenshot of a computer

AI-generated content may be incorrect.

* Click **Choose an existing template**
* Choose **Upload a template file**
* Choose **TGWwithBGP.json** as the template file
* Click **Next**

A screenshot of a computer

AI-generated content may be incorrect.

* Stack name: **Lab-4**
* Click **Next**

A screenshot of a computer

AI-generated content may be incorrect.

* IAM Role: **ffirole**
* Behavior on provisioning failure: **Roll back all stack resources**
* Delete newly created resources during a rollback: **Delete all newly created resources**

A screenshot of a computer

AI-generated content may be incorrect.

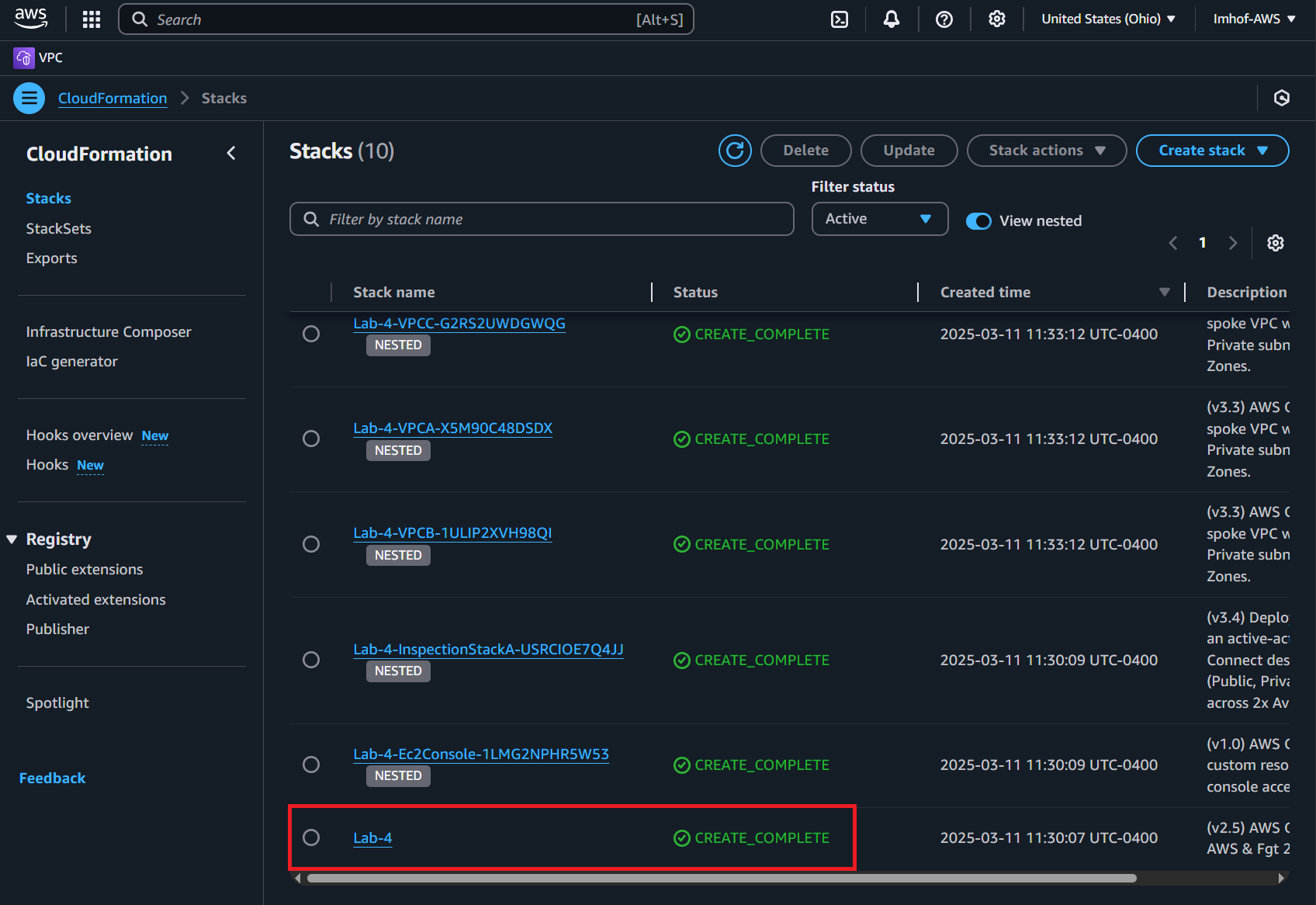
**Acknowledge the capabilities** and then click **Next**

A screenshot of a computer

AI-generated content may be incorrect.

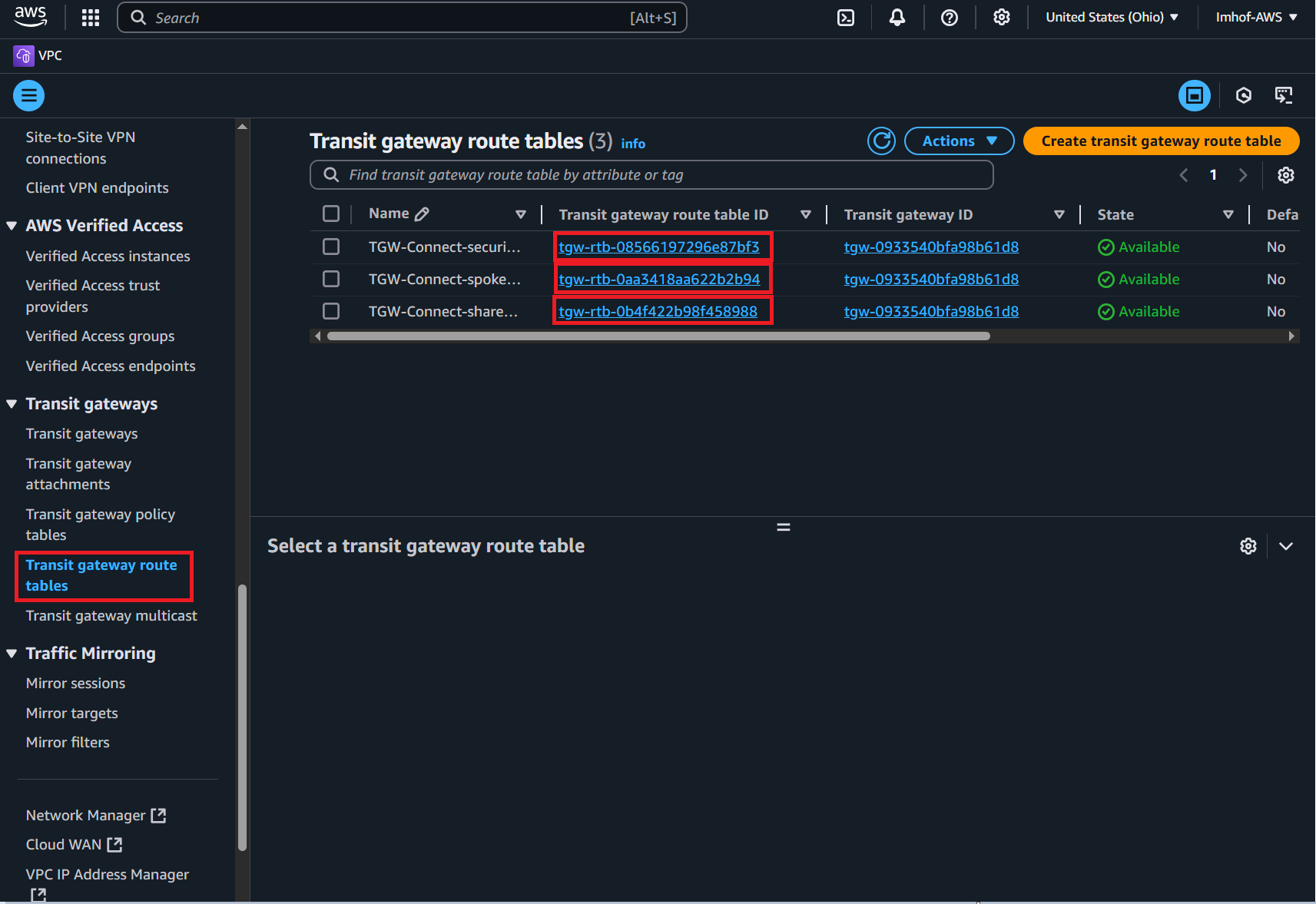
**Review the template parameters then click on Submit.**

It will take 5-10 minutes for the template to be fully deployed. Once it is finished the main/root stack will show as complete:



2 – Check the TGW Route Tables and confirm East/West is not working

* Navigate to the VPC console and click on the Transit gateway route tables page



* After clicking on each respective Transit gateway route table ID you can review their respective **Associations**, **Propagations**, and **Routes**.

A screenshot of a computer

AI-generated content may be incorrect.

* Verify each route table’s Associations using the following table:

Graphical user interface, application

AI-generated content may be incorrect.

* Verify each route table’s Propagations using the following table:

Graphical user interface, table

AI-generated content may be incorrect.

* Verify each route table’s Routes using the following table:

A picture containing graphical user interface

AI-generated content may be incorrect.

3 – Find EC2 Instance-A and verify that it cannot access Instance-B or C

* Navigate to the EC2 console
* Find Instance-A and connect to it using the following serial console directions:
  + Select the instance.
  + Click **Connect > EC2 serial console**.
  + **Copy the instance ID** as this will be the username and click connect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* Login to the EC2 instance:
  + You may need to hit enter to get a login prompt
  + Username: **«copied Instance ID from above»**
  + Password: **FORTInet123!**

Graphical user interface, application

AI-generated content may be incorrect.

* Run the following ping tests, they should all fail (Ctrl+C terminates the command):
  + **ping 10.2.2.10**
  + **curl ipinfo.io**

A screenshot of a computer

AI-generated content may be incorrect.

4 – Review FortiGate1 GRE + BGP config and advertise a summary route to TGW

* Navigate to the CloudFormation Console and **toggle View Nested to off**

A screenshot of a computer

AI-generated content may be incorrect.

* Select the Lab-4 stack then click on **Outputs**

A screenshot of a computer

AI-generated content may be incorrect.

* Access FGT1 CLI using the **FGT1LoginURL**, **Username**, and **Password** outputs presented
* Run the command **show system gre** and notice that the IP addresses are private.

A picture containing shape

AI-generated content may be incorrect.

* Run the command **show system interface tgw-conn-peer** and notice the IPs assigned to the inside interfaces of the GRE tunnel.

Text

AI-generated content may be incorrect.

* Run the command **show router bgp** and notice the BGP peers fall into the remote-ip CIDR in the output of the command above.

Text

AI-generated content may be incorrect.

* Run the command **get router info route-table all** and notice the routes received match the Routes tab for the route-table TGW-Connect-security-tgw-rtb.

Graphical user interface, text

AI-generated content may be incorrect.

* Copy and paste the following commands into FGT1 to advertise a 10.0.0.0/8 summary route back to Transit Gateway:

**config router bgp**

**config aggregate-address**

**edit 1**

**set prefix 10.0.0.0 255.0.0.0**

**set summary-only enable**

**next**

**end**

**end**

* Verify that 10.0.0.0/8 is now being advertised to our BGP neighbors:
  + **get router info bgp summary**

A screenshot of a computer screen

AI-generated content may be incorrect.

* + **get router info bgp neighbors 169.254.6.2 advertised-routes**

A screenshot of a computer

AI-generated content may be incorrect.

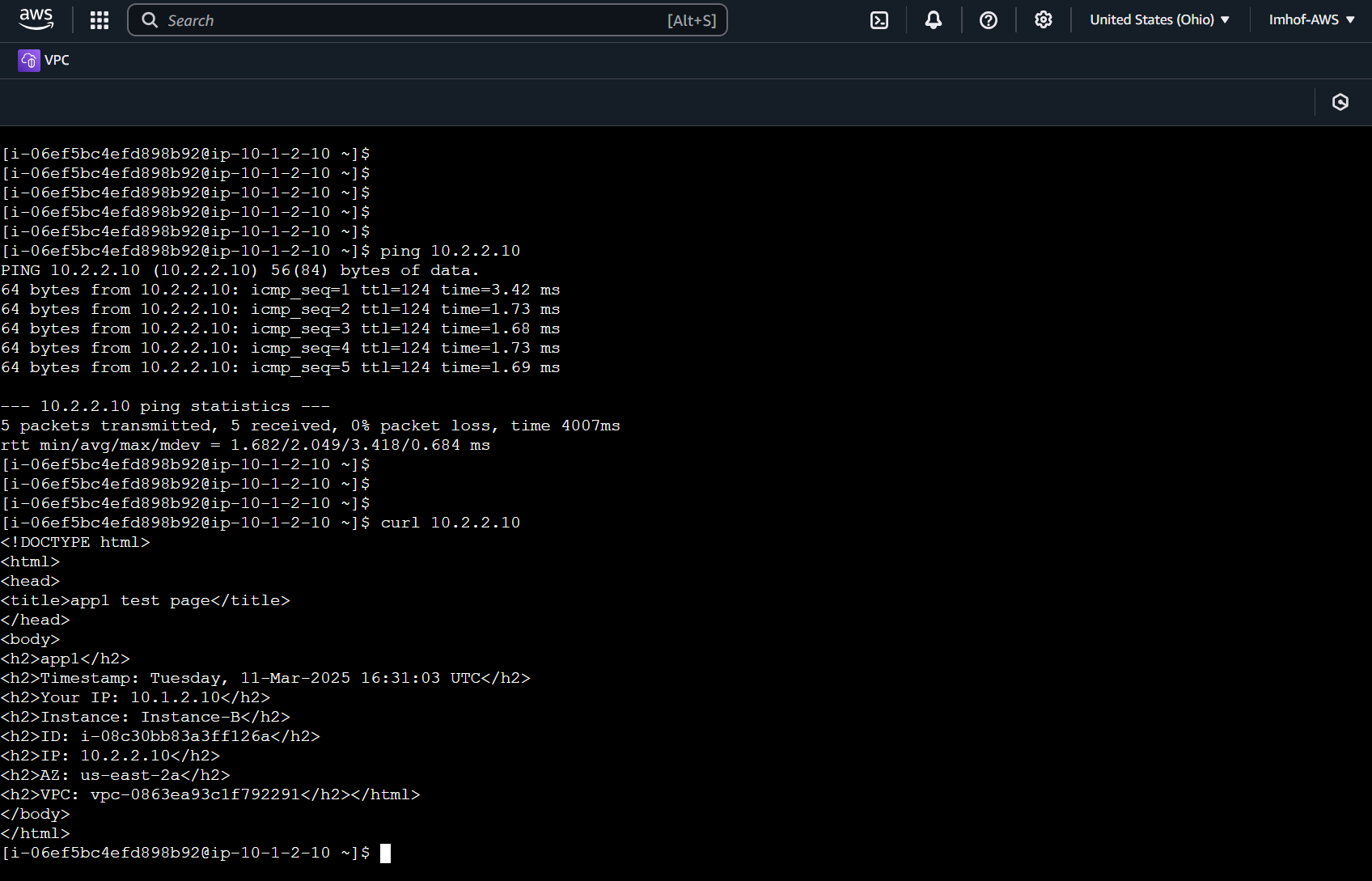
* + **get router info bgp neighbors 169.254.6.3 advertised-routes**

A screenshot of a computer

AI-generated content may be incorrect.

5 – Test East/West connectivity from Instance-A to Instance-B and validate there is no internet connectivity

* Run the following commands on Instance-A to validate connectivity to VPC B:
  + **ping 10.2.2.10**
  + **curl 10.2.2.10**



**How is all this working?**

This is a high level diagram showing how the Connect attachment goes over a VPC attachment which allows FortiGate1 to have a GRE tunnel over a private path which has BGP peering configured as well. This provides an overlay tunnel where dynamic routes and data-plane traffic can be routed without adding additional routes to the VPC router.

Diagram

AI-generated content may be incorrect.

Graphical user interface, text, application

AI-generated content may be incorrect.

* TGW supports ECMP routing with routes from the same attachment type.
  + This allows scalable Active-Active centralized ingress/egress inspection.
  + Active-Active East/West inspection requires SNAT to keep flows sticky to the same FortiGate.
* TGW has a route evaluation priority to select the best path when multiple routes have the same CIDR.
* Each TGW VPN connection (2x IPsec tunnels per connection) supports up to 1.5 Gbps.
* Each TGW Connect peer supports up to 5 Gbps.
* TGW supports multiple peers per TGW Connect attachment and multiple attachments to a single VPC.
* TGW supports multiple VPN attachments to the same or different customer gateway (remote IPsec device).

6 – Review FortiGate2’s VPN and BGP configurations

* Access the FGT2 CLI using the **FGT1LoginURL**, **Username**, and **Password** outputs presented in the CloudFormation outputs (reference step 3).
* Run the command **show vpn ipsec phase1-interface** and notice there are two tunnels where the remote-gw values are different public Ips and the interfaces are port1:

Text

AI-generated content may be incorrect.

* Run the command **show router route-map rmap-aspath1** and notice that the as-path is set to 6500:

Text

AI-generated content may be incorrect.

* Copy and paste these commands top configure default-route-originate with the route-map to advertise the default route with an as-path of 65000:

**config router bgp**

**config neighbor**

**edit 169.254.10.1**

**set capability-default-originate enable**

**set default-originate-routemap rmap-aspath1**

**next**

**edit 169.254.11.1**

**set capability-default-originate enable**

**set default-originate-routemap rmap-aspath1**

**next**

**end**

**end**

* Verify that the default 0.0.0.0/0 is now being advertised to our BGP neighbors with an as-path of 65000:
  + **get router info bgp summary**

A screenshot of a computer screen

AI-generated content may be incorrect.

* + **get router info bgp neighbors 169.254.10.1 advertised-routes**

A screenshot of a computer

AI-generated content may be incorrect.

* + **get router info bgp neighbors 169.254.11.1 advertised-routes**

Graphical user interface

AI-generated content may be incorrect.

7 – Test secured egress connectivity from Instance-A through FGT2

* Run the following commands on Instance-A to validate connectivity to VPC B:
  + **ping 10.2.2.10**
  + **ping 8.8.8.8**

Text

AI-generated content may be incorrect.

* + **curl 10.2.2.10**
  + **curl ipinfo.io**

Text

AI-generated content may be incorrect.

* Navigate to the VPC console, click Transit gateway route tables, and click on the **TGW-Connect-spoke-tgw-rtb** to view the route table’s routes:

A screenshot of a computer

AI-generated content may be incorrect.

Notice there are ECMP routes for 0.0.0.0/0 since there are two VPN tunnels configured

Graphical user interface, text

AI-generated content may be incorrect.

8 – Configure FGT1 with default-route-originate and a route-map

* Log into FGT1 using the credentials from the CloudFormation output.
* Copy and paste the following commands to configure default-route-originate with a route map to advertise 0.0.0.0/0 with an as-path of 65000:

**config router bgp**

**config neighbor**

**edit 169.254.6.2**

**set capability-default-originate enable**

**set default-originate-routemap rmap-aspath1**

**next**

**edit 169.254.6.3**

**set capability-default-originate enable**

**set default-originate-routemap rmap-aspath1**

**next**

**end**

**end**

* Verify that the default 0.0.0.0/0 is now being advertised to our BGP neighbors with an as-path of 65000:
  + **get router info bgp summary**

A screenshot of a computer

AI-generated content may be incorrect.

* + **get router info bgp neighbors 169.254.6.2 advertised-routes**

Graphical user interface, text

AI-generated content may be incorrect.

* + **get router info bgp neighbors 169.254.6.3 advertised-routes**

Graphical user interface, text

AI-generated content may be incorrect.

9 – Test secured egress connectivity from Instance-A through FGT2

* Run the following commands on Instance-A to validate connectivity:
  + **ping 10.2.2.10**
  + **ping 8.8.8.8**

A screenshot of a computer

AI-generated content may be incorrect.

* + **curl 10.2.2.10**
  + **curl ipinfo.io**

A screenshot of a computer

AI-generated content may be incorrect.