

Network Principles in Practice: Cloud Networking

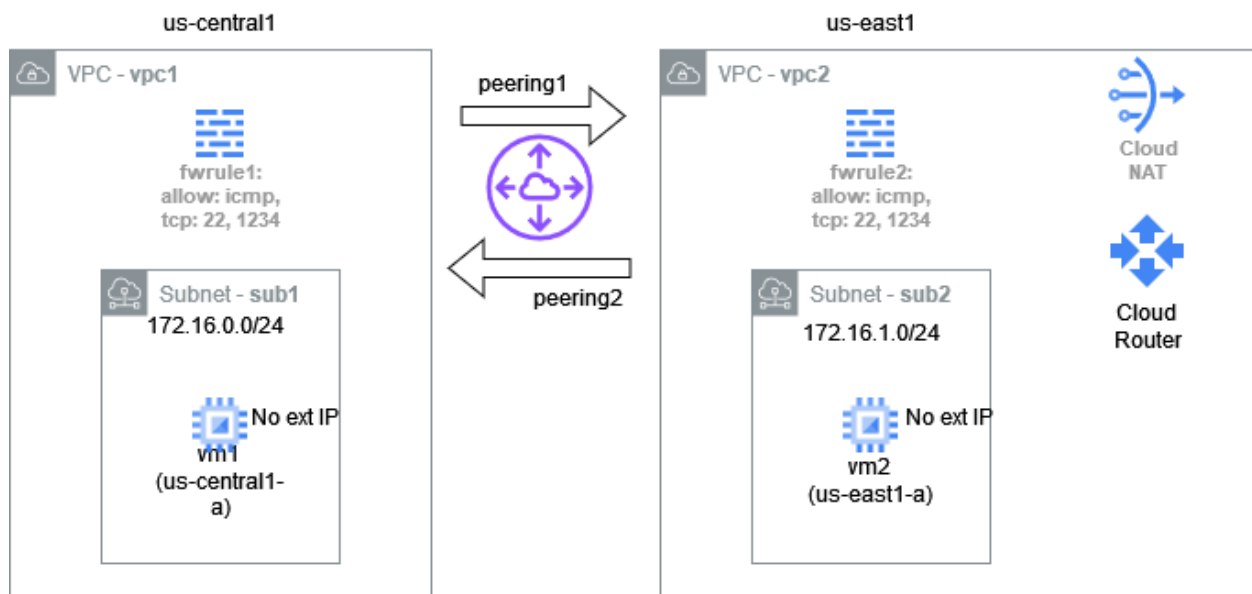
Module: WAN

Assignment

Overview

In this lab, you will build the following infrastructure in terraform.

Note: In the diagram (and in the text description below), we use shortened resource names (vpc1, sub1, vm1, etc.). You should prefix all resource names with “tf-mod3-lab1-”. For example, vpc1 would be tf-mod3-lab1-vpc1.



See video in coursera for more information.

Set up a VXLAN tunnel (note, this will use UDP port 50000):

On vm1

- > sudo ip link add vxlan0 type vxlan id 5001 local 172.16.0.2 remote 172.16.1.2 dev ens4 dstport 50000
- > sudo ip addr add 192.168.100.2/24 dev vxlan0
- > sudo ip link set up dev vxlan0

(will add a routing table entry for 192.168.100.0/24)

On vm2

```
> sudo ip link add vxlan0 type vxlan id 5001 remote 172.16.0.2 local 172.16.1.2 dev ens4 dstport 50000
```

```
> sudo ip addr add 192.168.100.3/24 dev vxlan0
```

```
> sudo ip link set up dev vxlan0
```

(will add a routing table entry for 192.168.100.0/24)

Setup NAT on vm2:

Note: this is slightly different than what was shown in the video.

```
> sudo /sbin/sysctl -w net.ipv4.ip_forward=1
```

```
> sudo /sbin/iptables -t nat -A POSTROUTING -s 192.168.0.0/16 -o ens4 -j MASQUERADE
```

Setup a route for traffic to <IP address of neverssl.com> to go through vm2 over the vxlan tunnel:

On vm1:

```
sudo ip route add 34.223.124.45/32 via 192.168.100.3
```

Try connectivity:

On vm1:

```
wget 34.223.124.45
```

You should see it download something. Note the file name.

Other actions (for the questions):

1. On VM1: `grep "Follow" <file-that-was-downloaded>`
2. On VM1: `ping -c 1 8.8.8.8`
3. `terraform show --json > terraform_show_out.json`

4. Perform the following: `jq '[query]' terraform_show_out.json`. That is, put each of the following queries inside of single quotes, and note the output. This assumes you named resources correctly (with the prefix of `tf-mod3-lab1-`)

Query1:

```
.values.root_module.resources[] | select(.type == "google_compute_network_peering" and .name == "tf-mod3-lab1-peering1").values.state
```

Query2:

```
.values.root_module.resources[] | select(.type == "google_compute_network_peering" and .name == "tf-mod3-lab1-peering2").address
```

Query3:

```
.values.root_module.resources[] | select(.type == "google_compute_network_peering" and .name == "tf-mod3-lab1-peering2").values.network
```

Query4:

```
.values.root_module.resources[] | select(.type == "google_compute_network_peering" and .name == "tf-mod3-lab1-peering2").values.peer
```