Implement the Azure IaaS Project

Project for CalTech Az104 Course

David Hill, Jr.

Description

OSS Corporation is a globally distributed firm. They have their headquarters in the East US with another branch office in the WEST US. Currently, they are working on a project and decided that the application tier of this project will reside in one of its branch regions. For security reasons, OSS Corporation management is adamant on keeping their data tier in the headquarter region.

Background of the problem statement:

As an organization, they are open to suggestions and are currently evaluating Azure as a deployment platform. To prepare for the deployment of IaaS Standard_B1ms, OSS Corporation must deploy an IaaS v2 virtual network in the headquarters region for its database. But for the application, it should create another IaaS v2 virtual network in the branch region. In addition, because the communication between App and data should happen over a private channel, one needs to prepare their branch office virtual network for establishing connectivity to the headquarter's IaaS v2 virtual network by creating a virtual network gateway and deploy a test IaaS Standard_B1ms VM to the virtual networks for verifying the connection.

After the deployment team ensures the connectivity between both the networks, you can validate the same using Ping.

Following requirements should be met:

- ☒ Create virtual networks in the aforementioned region
- ☒ Create test virtual machines in both the virtual networks
- ☒ Establish the connectivity between both the networks via VNet peering
- ☒ Ensure connectivity is established properly

Creating Virtual Networks

The first step in this project was to create a resource group for the OSS Corporation to make all of the created resources easier to manage. I then set up two virtual networks, one in East US for the OSS headquarters and one in West US for the remote location. I accomplished this using the azure portal. I set the East US IP range to be 10.0.0.0/24

for simplicity purposes. Moreover, I set the IP range of the West US location to be 10.1.0.0/16. The Vnets were configured identically for everything else.

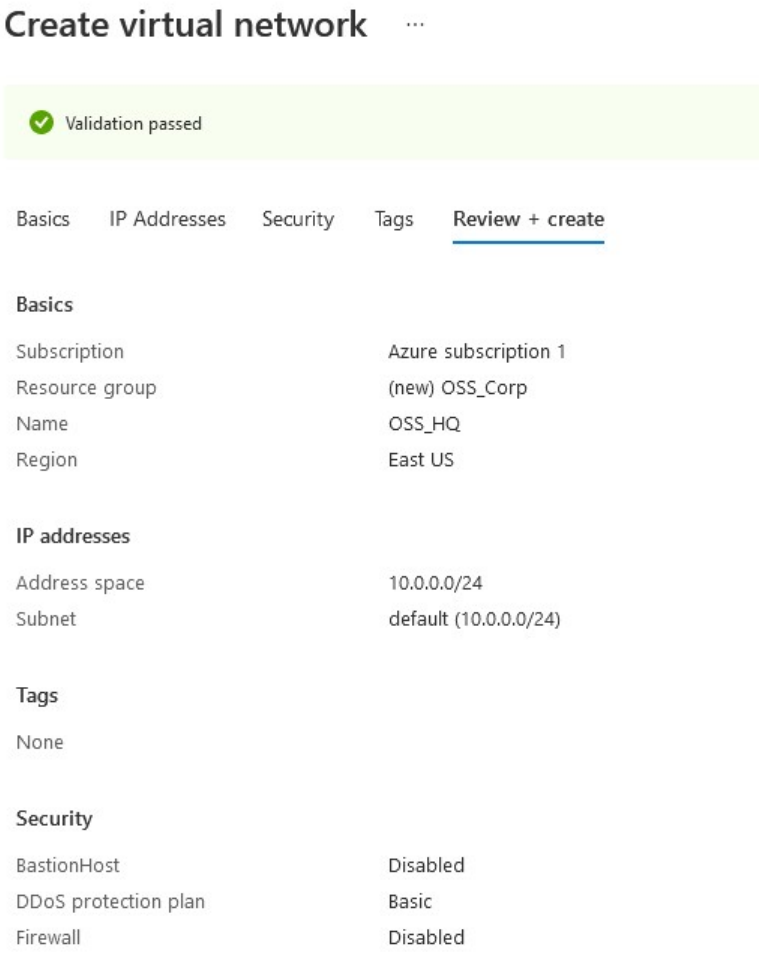

Figure 1: HQ Vnet before creation

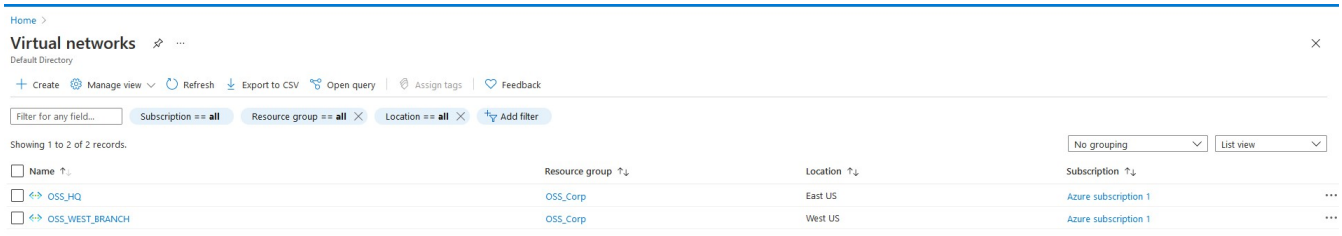

Figure 2: Both VNets after creation

Virtual Machines

Next, I set up the virtual machines (VMs) per the specifications of the OSS corporation. Both were Standard_B1ms virtual machines running the Windows Server 2019 operating

system. One VM called “Database” I set to reside in the default subnet of the headquarters in East US. The other VM called “Application” I set to reside in the default subnet of the branch location in West US. In figure 3 you will see both virtual machines after creation.

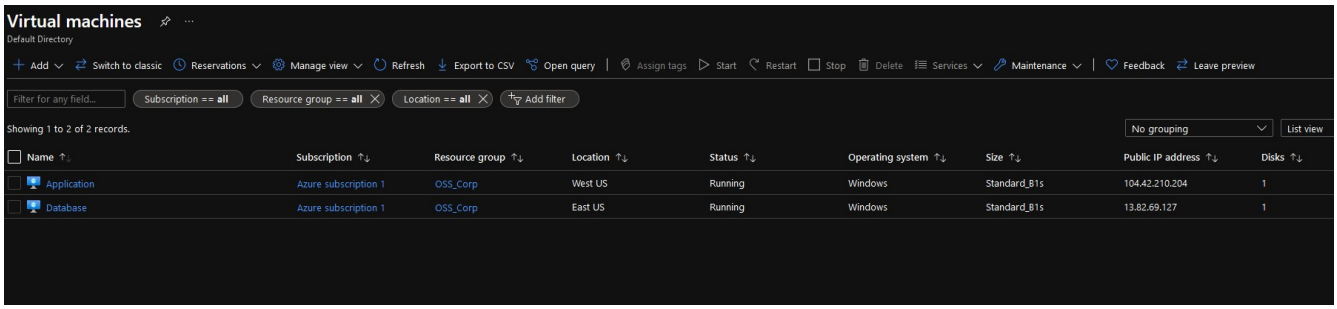

Figure 3: VMs after creation

Vnet Peering

For the VMs in the two regions to interact, the OSS networks need to have peering established so that they can communicate privately. To accomplish this, I set up Vnet Peering between the two networks. One peering is from the Headquarters to the Branch location and one is from the Branch location to the Headquarters. The peering was set up to enable bidirectional communication.

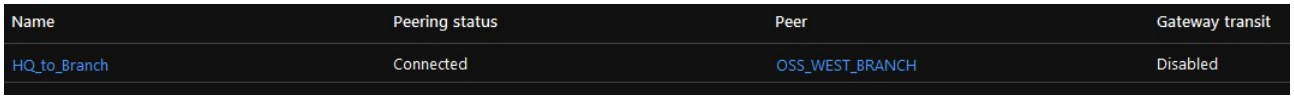

Figure 4: HQ to Branch Peering

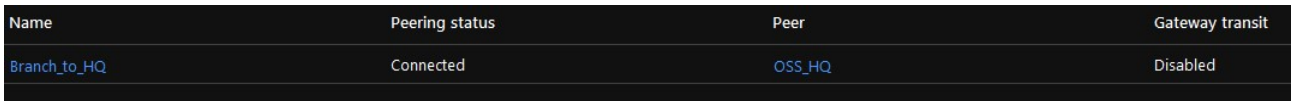

Figure 5: Branch to HQ Peering

Establishing Connectivity

The last step is to verify the connectivity of the deployed IaaS. To do this I connected to both VMs using the Remote Desktop Connection (RDC) utility on Windows. I signed in using the credentials I set up when making the VMs. Putting both instances of RDC side by side so that I could work with both VMs at the same time. By running a simple ping command I was able to successfully send bytes from one VM to the other and vice-versa, thus confirming that we set up the Azure IaaS correctly for the OSS Corporations deployment.

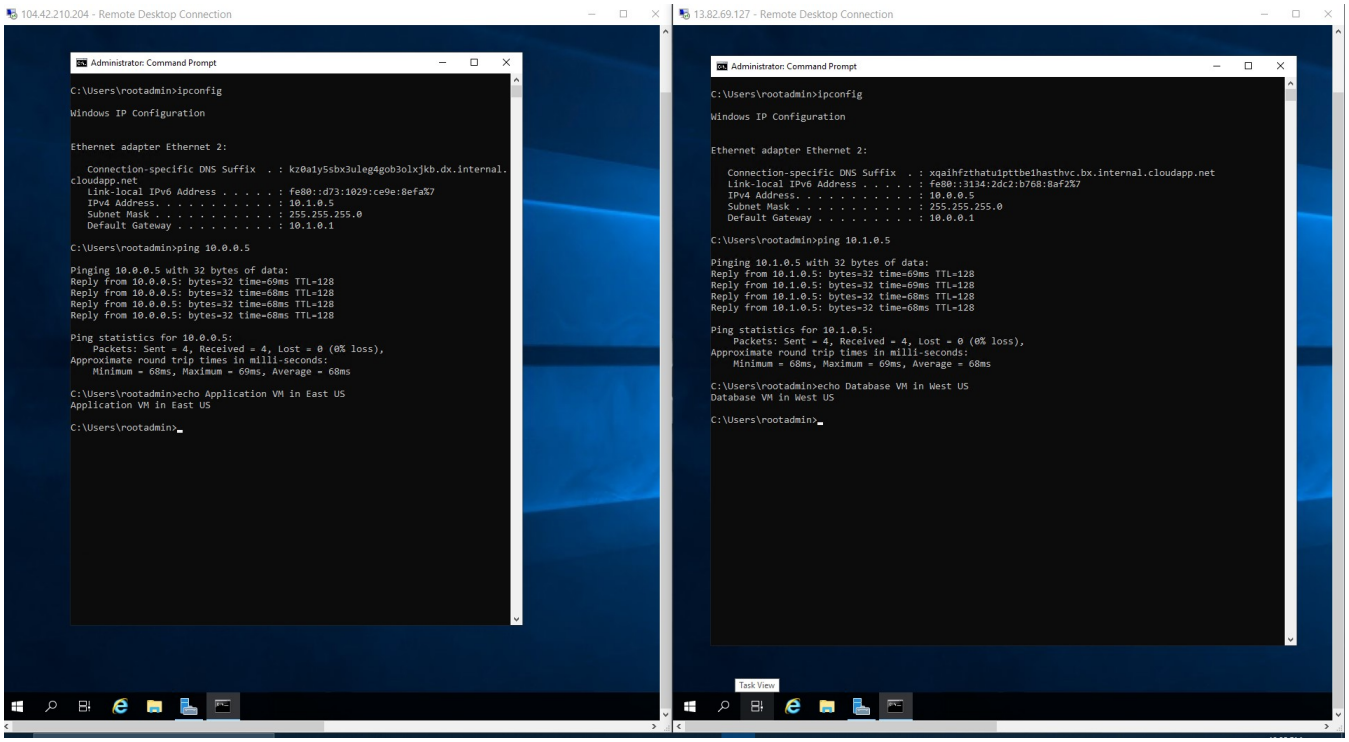

Figure 6: Verifying the Connection of the VMs