Intersite Connectivity

Eli Chang

MSSA Cohort #2

Lab Summary 5

4/5/2020

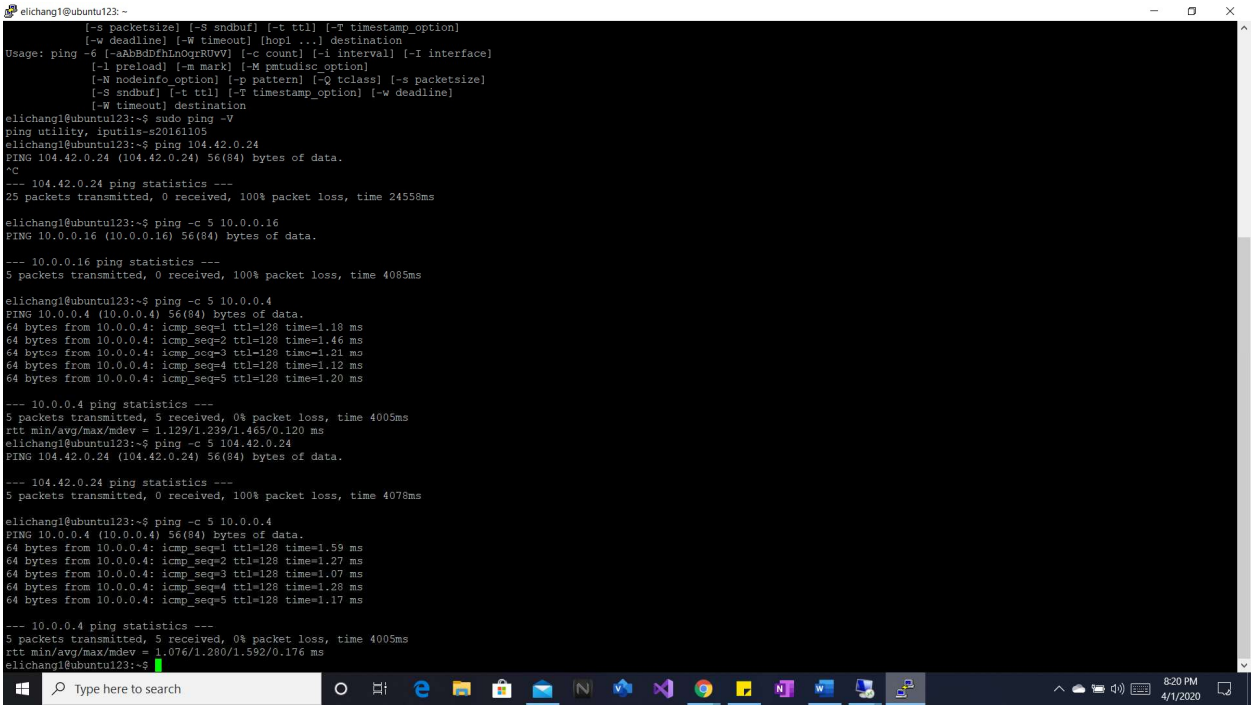
Intersite Connectivity

The environment of the lab is to prepare one Windows Server 2016 virtual machine, one Windows 10 virtual machine, and one Linux virtual machine. The virtual machines are running under the Azure environment. Once creating and running these machines, the communication between the Linux machine and other machines should be established. From the Linux machine, the command line ping could retrieve the status of communication between the machines.

The Linux machine is in one resource group while the other machines are in another resource group. To enable the connectivity between these machines, the VNet peering should be configured. There are three options to choose: allow forwarded traffic, allow gateway transit, and use remote gateways. Both sides would be able to communicate with each other, but others would not communicate with them.

When starting pinging from the Linux machine, ping "IP address" would not stop pinging until the proper command line to stop types. In the Linux machine, the number of pinging should be determined by a user with command line "ping -c # "IP address." The meaning of -c is to count, and # means the number of pinging. For example, the command line, "ping -c 5 "102.15.0.100," would ping the IP address for five times, and then it will show the pinging attempts were successful or not.

Windows 10 and Windows Server 2016 machines did not let the Linux machine to communicate with them even though the ports were allowed. In PowerShell in each Windows machine, the command line with netsh advfirewall firewall add rule name="ICMP Allow incoming V4 echo request" protocol=icmpv4:8, any dir=in action=allow was used to resolve the issue.


elichang1@ubuntu123: ~
[-s packetsize] [-S sndbuf] [-t ttl] [-T timestamp_option]
[-w deadline] [-W timeout] [hop1 ...] destination
Usage: ping -6 [-aAbBdDfhLnOqrRUvV] [-c count] [-i interval] [-I interface]
[-l preload] [-m mark] [-M pmtudisc_option]
[-N nodeinfo_option] [-p pattern] [-Q tclass] [-s packetsize]
[-S sndbuf] [-t ttl] [-T timestamp_option] [-w deadline]
[-W timeout] destination
elichang1@ubuntu123:~$ sudo ping -V
ping utility, iputils-s20161105
elichang1@ubuntu123:~$ ping 104.42.0.24
PING 104.42.0.24 (104.42.0.24) 56(84) bytes of data.
^C
--- 104.42.0.24 ping statistics ---
25 packets transmitted, 0 received, 100% packet loss, time 24558ms
elichang1@ubuntu123:~$ ping -c 5 10.0.0.16
PING 10.0.0.16 (10.0.0.16) 56(84) bytes of data.
--- 10.0.0.16 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4085ms
elichang1@ubuntu123:~$ ping -c 5 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=128 time=1.18 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=128 time=1.46 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=128 time=1.21 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=128 time=1.12 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=128 time=1.20 ms
--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 1.129/1.239/1.465/0.120 ms
elichang1@ubuntu123:~$ ping -c 5 104.42.0.24
PING 104.42.0.24 (104.42.0.24) 56(84) bytes of data.
--- 104.42.0.24 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4078ms
elichang1@ubuntu123:~$ ping -c 5 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=128 time=1.59 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=128 time=1.27 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=128 time=1.07 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=128 time=1.28 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=128 time=1.17 ms
--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 1.076/1.280/1.592/0.176 ms
elichang1@ubuntu123:~$
Type here to search
8:20 PM
4/1/2020