**Design Template**

|  |
| --- |
| Sparkit Company |
| Desktop Virtualization Design and Build |
| Desktop Virtualization – D086 |

|  |
| --- |
| [Author name]  [Date]  [Version 1.1] |

Contents

[A. Problem statement 3](#_Toc40894166)

[B. Hardware Specifications 4](#_Toc40894167)

[C. Hypervisor 5](#_Toc40894168)

[D. Allocations 5](#_Toc40894169)

[E. Implementation 7](#_Toc40894170)

[F. Testing/Evaluation 9](#_Toc40894171)

[F.1. Network Connectivity 9](#_Toc40894172)

[F.2. Memory Utilization 10](#_Toc40894173)

[F.3. CPU Performance 10](#_Toc40894174)

[G. Virtualization Build 11](#_Toc40894175)

[G.1. Step one – Hyper-V Setup and Settins 11](#_Toc40894176)

[G.2. Step Two – Client A 14](#_Toc40894177)

[G.3. Step Three – Client B 15](#_Toc40894178)

[G.4. Step Four – CO1 16](#_Toc40894179)

[G.5. STEP Five – UBU1 16](#_Toc40894180)

[G.6. Step 6 – pfSense Router 16](#_Toc40894181)

# Problem statement

Sparkit Company has requested a design plan for cutting costs while still building the brand and grow in areas that have proven profitable. They are looking identify was to innovate and improve productivity through virtualization. Sparkit is expanding their gaming products into the Windows and Linux environments and their IT and QA staff will need access to these platforms in an isolated environment for development and quality control. A virtualized environment on their desktop with networking capabilities allows for multiple operating systems to run on the single host for under 800$.

# Hardware Specifications

**Hard Drives**

500GB Internal SSD - SATA III 6Gb/s - $69.99

* This drive is selected as the host machines boot operating system drive. The SSD drive is cost effective and fast to allow for great performance of the host system when needing to access certain system files while the virtual environment is running.

2TB 7200 RPM 256MB Cache SATA 6.0Gb/s 3.5" Hard Drive – $82.95

* This drive was selected to house all source files for the virtual environment as well as the virtual hard drives of the VM’s. With 2TB of storage it can not only house all the virtual systems hard drive files, it has plenty of room for growth for additional virtual system to be stored without impacting the boot drive of the host system.

**Processor**

Intel Core i9-9900K 8-Core, 16-Thread, 3.6 GHz LGA 1151 - $518.99

* With the requirement that all VM workloads not exceed 50% of the hosts processor, we chose to go with the i9-990K 8-core to ensure great performance and allow for plenty of headroom incase the VM’s were hitting CPU spikes concurrently.

**Memory**

16GB (2 x 8GB) 288-Pin DDR4 SDRAM DDR4 3000 (PC4 24000) - $125.99

* In choosing to go with 16Gb of RAM we opted for the higher clock speed for increased performance the RAM will grant. In the current configuration of the VM’s and their workload the 16GB will suffice and not take more than 60% of the total capacity of the RAM.

**Network Card**

Since all VM’s will be on an internal network without access to the internet, we opted to not install a network interface card and instead chose to utilize the on-board network card that comes with the motherboard. It will allow the host system to access the internet and download anything needed from the network or externally from the internet if needed.

**Total Build Cost: $797.92**

# Hypervisor

The hypervisor chosen for this solution is Hyper-V. This was chosen for several reason. The first reason it was chosen was to be cost effective. Hyper-V is free of cost with any version of Windows 10 Pro or Enterprise versions. With this being free of cost no additional software purchase or licensing is required saving the company money. The second reason this was chosen with its flexibility in supporting multiple OS’s from all versions of Windows to a wide variety of Linux based operating systems. The final reason why Hyper-V was chosen was for its ease of migration and its compatibility to be expanded into a full client/server environment on the network.

# Allocations

The resource allocations for the virtual systems are shown below in the table (Figure D.1) and explained in detail below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Client** | **OS** | **CPU Cores** | **RAM MB** | **Storage 1 GB** | **Storage 2 GB** | **Network 1** | **Network 2** |
| **Client A** | Windows 10 Pro | 2 | 2048 | 100 | 5 | NetWest |  |
| **CO1** | CentOS Workstation | 2 | 2048 | 100 | 5 | NetWest |  |
| **Router** | FreeBSD/pfSense | 1 | 512 | 50 |  | NetEast | NetEast |
| **Client B** | Windows 10 Pro | 2 | 2048 | 100 | 5 | NetEast |  |
| **UBU1** | Ubuntu Desktop | 1 | 2048 | 100 | 5 | NetEast |  |
|  | **Totals** | 8 | 8704 | 450 | 10 |  |  |

Figure D.1

Since game development and QA was the main criteria that guided the virtual systems build. I gave each VM 2GB of RAM to benefit the development and QA of the games. When developing or performing intensive QA on games more Ram in a system is beneficial. I gave the pfSense Router only minimal hardware, I gave it 512MB RAM and a single core processor as it will only be performing routing and a 50GB HDD to be able to give room for growth for future updates to the OS.

I allocated 8 cores in total to all systems; 2 cores to each of the Windows systems, 2 cores to CentOS since it is normally run as a server even though I installed workstation, 1 core for Ubuntu since Linux typically performs well with low CPU resources, and 1 core for pfSense since it is only used for networking.

I allocated each system a 100GB primary HDD to allow for ample storage and growth of files, except for the pfSense router VM since its only function is routing. The storage at 100GB gives plenty of room for storage and the 2TB storage drive housing the virtual hard drives also has plenty of room incase a VM needs its HDD expanded.

The remainder of the configurations and allocations were required by the project requirements.

# Implementation

1. Hyper-V Installation
   1. Access Windows Features and place a check in the Hyper-V box, click OK, allow the install to finish and reboot host system
   2. Launch Hyper-V
   3. Access the settings from Actions>Hyper-V Settings…
      1. Set Virtual Hard Disks to the second HDD on the host <drive>:\HyperV\
      2. Set Virtual Machines to the second HDD on the host <drive>:\HyperV\
      3. Click OK
   4. Access Virtual Switch Manager from Actions>Virtual Switch Manager…
      1. Create 2 Private Virtual Switches named NetEast and NetWest
2. Client A
   1. Create new virtual machine with the given specification
      1. Name: Client A
      2. Generation: 1
      3. Memory: 2048
      4. Networking: NetWest
      5. HDD: 100GB
      6. Add Second 5GB HDD from VM Settings after creation
      7. Add Second Processor from VM Settings after Creation
   2. Install Windows 10 Pro
   3. Create OS install Checkpoint
   4. Add rule to Windows Firewall to Allow ICMPv4 in (needed for testing)
   5. Configure IPv4 to static IP for Class B network 172.16.0.3 and subnet 255.255.255.0
   6. Create Configuration Checkpoint
3. Client B
   1. Create new virtual machine with the given specification
      1. Name: Client B
      2. Generation: 1
      3. Memory: 2048
      4. Networking: NetEast
      5. HDD: 100GB
      6. Add Second 5GB HDD from VM Settings after creation
      7. Add Second Processor from VM Settings after Creation
   2. Install Windows 10 Pro
   3. Create OS install Checkpoint
   4. Add rule to Windows Firewall to Allow ICMPv4 in (needed for testing)
   5. Configure IPv4 to static IP for Class B network 172.16.1.3 and subnet 255.255.255.0
   6. Create Configuration Checkpoint
4. CO1
   1. Create new virtual machine with the given specification
      1. Name: CO1
      2. Generation: 1
      3. Memory: 2048
      4. Networking: NetWest
      5. HDD: 100GB
      6. Add Second 5GB HDD from VM Settings after creation
      7. Add Second Processor from VM Settings after Creation
   2. Create configuration Checkpoint, this differs from the Windows systems as final OS configuration occurs during install
   3. Install CentOS Workstation
   4. Configure IPv4 to static IP for Class B network 172.16.0.2 and subnet 255.255.255.0 during OS install
   5. Create OS install checkpoint
5. UBU1
   1. Create new virtual machine with the given specifications above
   2. Create new virtual machine with the given specification
      1. Name: UBU1
      2. Generation: 1
      3. Memory: 2048
      4. Networking: NetEast
      5. HDD: 100GB
      6. Add Second 5GB HDD from VM Settings after creation
   3. Install Ubuntu Workstation
   4. Create OS install Checkpoint
   5. Configure IPv4 to static IP for Class B network 172.16.1.2 and subnet 255.255.255.0 during OS install
   6. Create OS install checkpoint

# Testing/Evaluation

Explain how network connectivity, memory utilization, and CPU performance will be measured as successful, including how each would be tested and what the acceptance criteria would be.

# F.1. Network Connectivity

|  |
| --- |
| Describe the testing scenario.   * Router will be configured with NetEast and NetWest each attached to their own NIC’s each with static IP’s 172.16.1.1 and 172.16.0.1 respectively * Client A and CO1 will be configured with static IP’s for NetWest with 172.16.0.3 and 172.16.0.2 respectively and Client A will have ICMPv4 ping in rule added to firewall * Client B and UBU1 will be configured with static IP’s for NetEast with 172.16.0.3 and 172.16.0.2 respectively and Client B will have ICMPv4 ping in rule added to firewall   This allows for testing network connectivity to each client on the same switch as well between the two different switches. |
| Describe the acceptance criteria for the scenario.  All clients will be able to successfully ping all other clients including each IP assigned to the switches.   * Client A will be able to ping IP for NetEast, NetWest, CO1, UBU1 and Client B * Client B will be able to ping IP for NetEast, NetWest, CO1, UBU1 and Client A * Client CO1 will be able to ping IP for NetEast, NetWest, UBU1, Client A and Client B * Client UBU1 will be able to ping IP for NetEast, NetWest, CO1, Client A and Client B |

# F.2. Memory Utilization

|  |
| --- |
| Describe the testing scenario.  Host machine will record the amount of RAM utilized by only the Host machines operating system meaning all applications are closed and all VM’s or Hyper-V are not running.  Hyper-V will be launched, and Memory will be recorded again  All VM’s will be started and memory will be recorded once again. All VM’s are set to static RAM and will not be using Dynamic Memory Allocation |
| Describe the acceptance criteria for the scenario.  With the task manager open on the Host machine and view the Performance Tab select Memory, record the total amount of RAM used. Acceptance is if the total RAM used after all VM’s and Hyper-V are running, minus the starting value, is less than 9.8GB then the test is successful |

# F.3. CPU Performance

|  |
| --- |
| Describe the testing scenario.  Host machine will record the amount of CPU usage by only the Host machines operating system meaning all applications are closed and all VM’s or Hyper-V are not running.  Hyper-V will be launched, and CPU usage will be recorded again  All VM’s will be started and CPU usage will be recorded once again. |
| Describe the acceptance criteria for the scenario.  With the task manager open on the Host machine and view the Performance Tab select CPU, record the total amount of CPU usage. Acceptance is if the total CPU usage after all VM’s and Hyper-V are running after initial boot up of each VM, minus the starting value, is less than 50% then the test is successful |

# Virtualization Build

# Step one – Hyper-V Setup and Settings

These screenshots show how to install Hyper-V on windows Pro.





These show how to access the Hyper-V settings and what the server settings are for the VM storage when configured correctly.

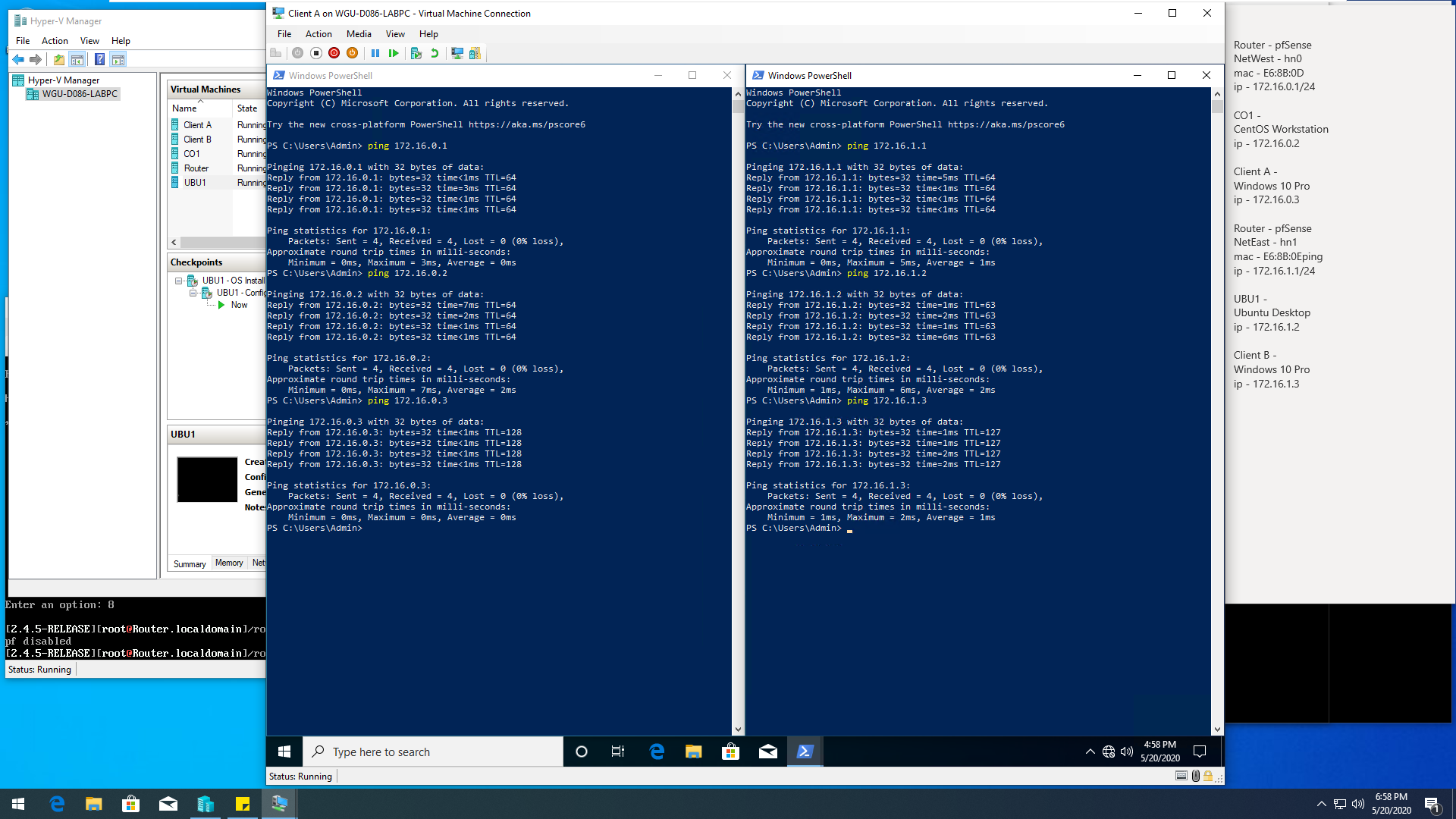


These show how to access the Virtual Switch Manager and what is needed to setup the two private switches for NetEast and NetWest. It also shows what the screen will look like once correctly configured.

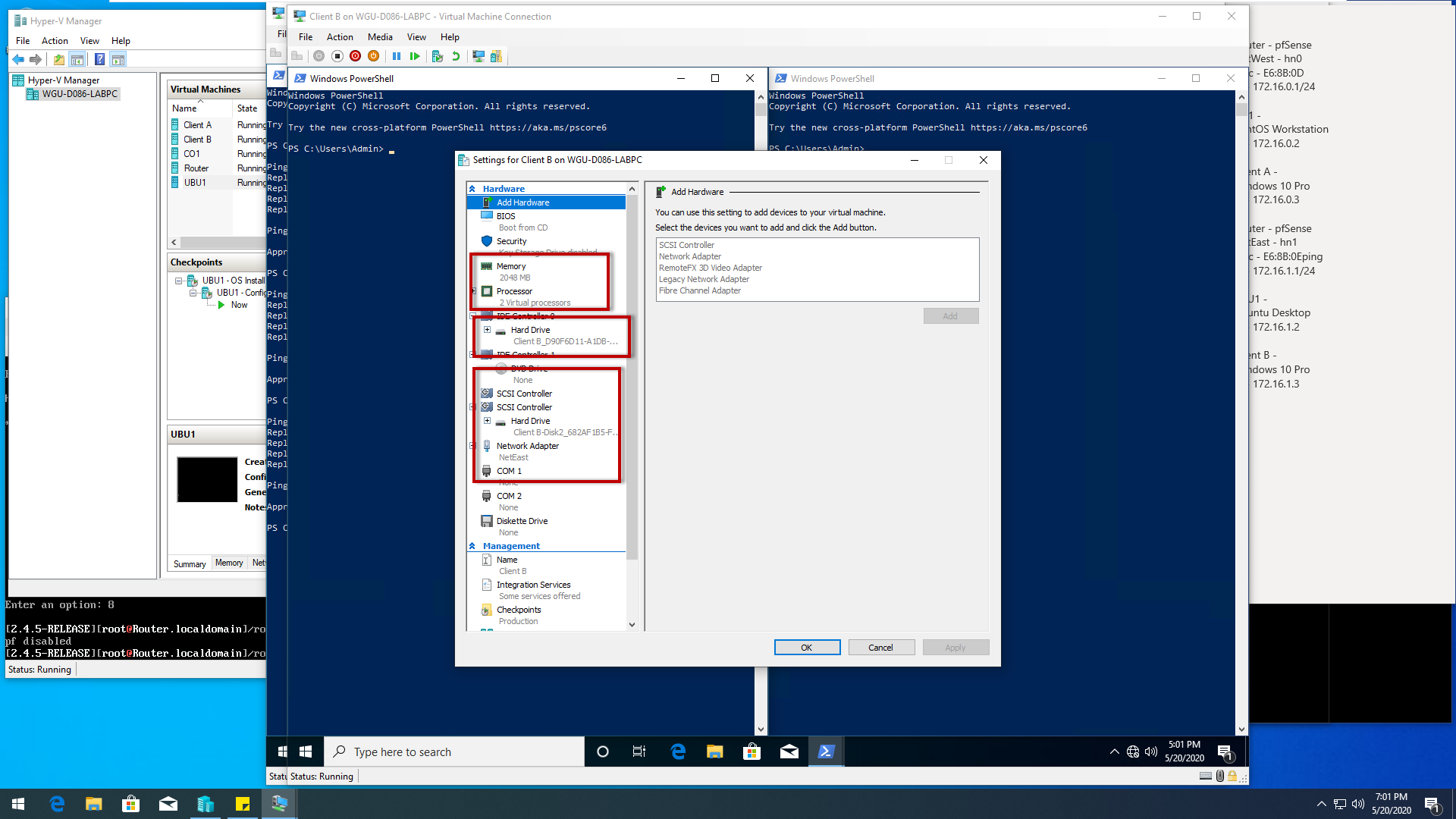


# Step Two – Client A

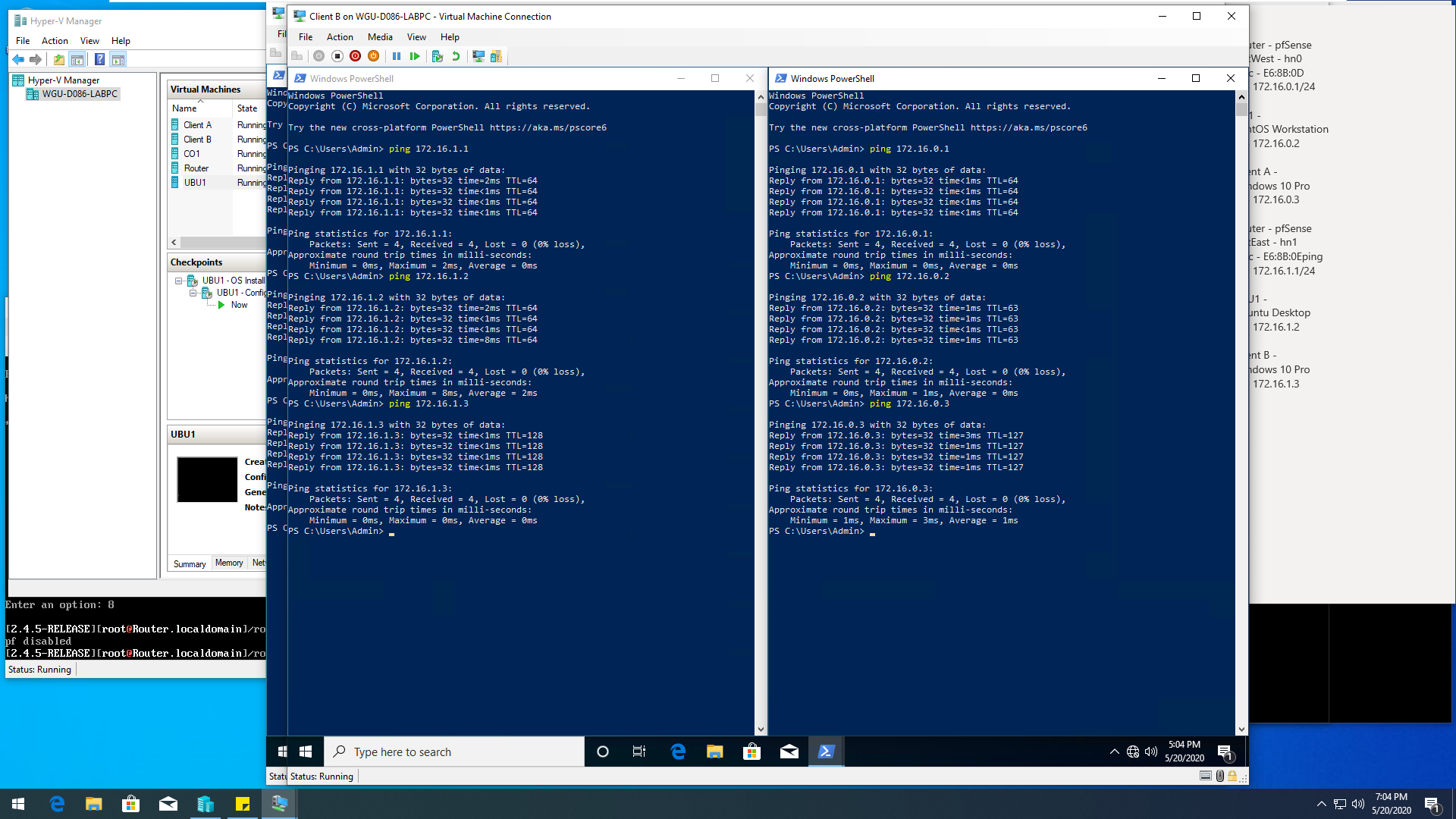
This shows Client A. The below screenshot shows the configuration of the client showing 2GB RAM, 2 processors, both HDDs, connected to NetWest while all VM’s are running.

This shows Client A able to ping all other VM’s, including itself, and both IP’s assigned to Router for NetEast and NetWest. For this step a firewall rule was added to allow ICMPv4 ping in for testing. On the firewall packet filtering was also disabled for testing.

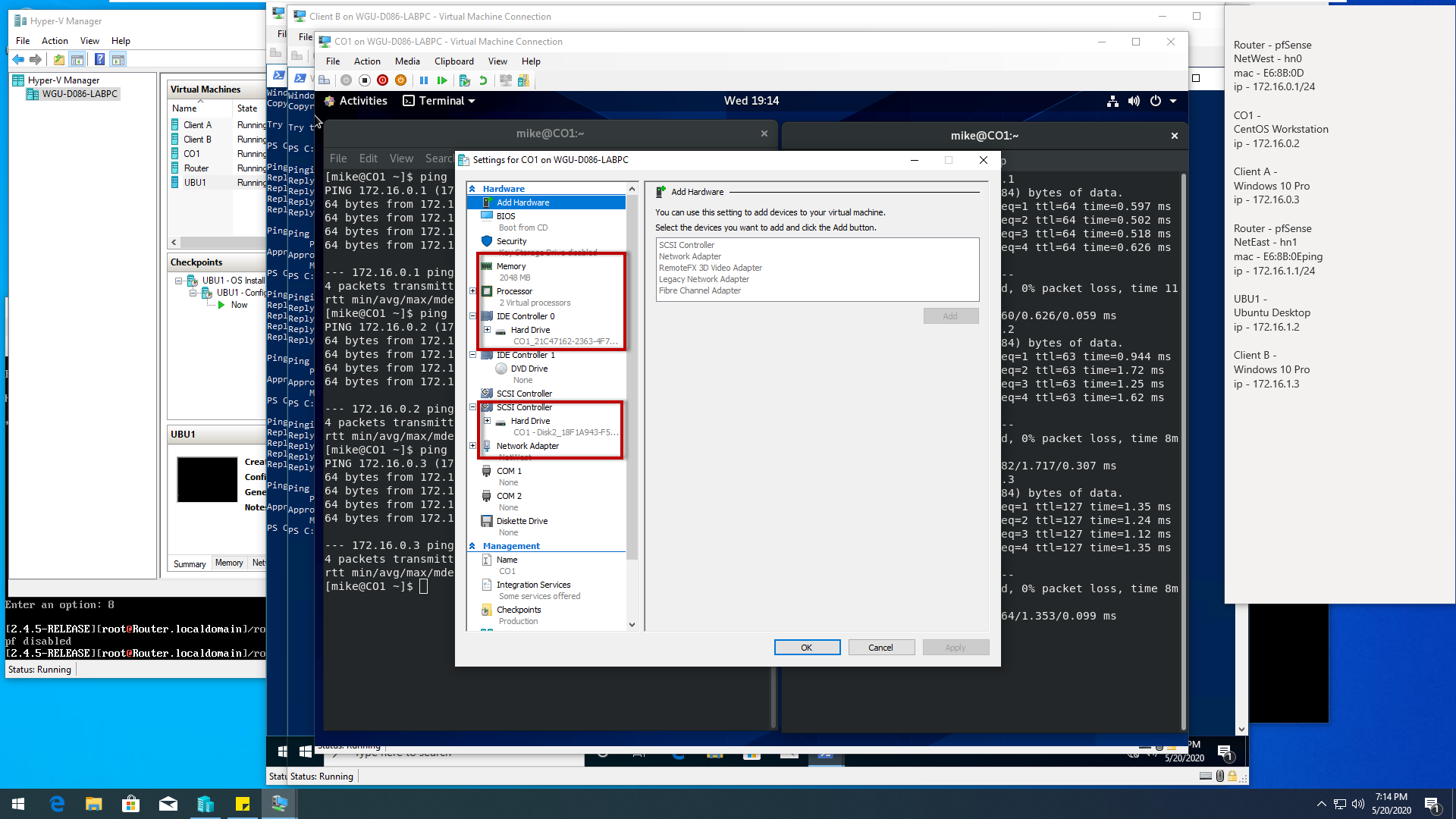
# Step Three – Client B

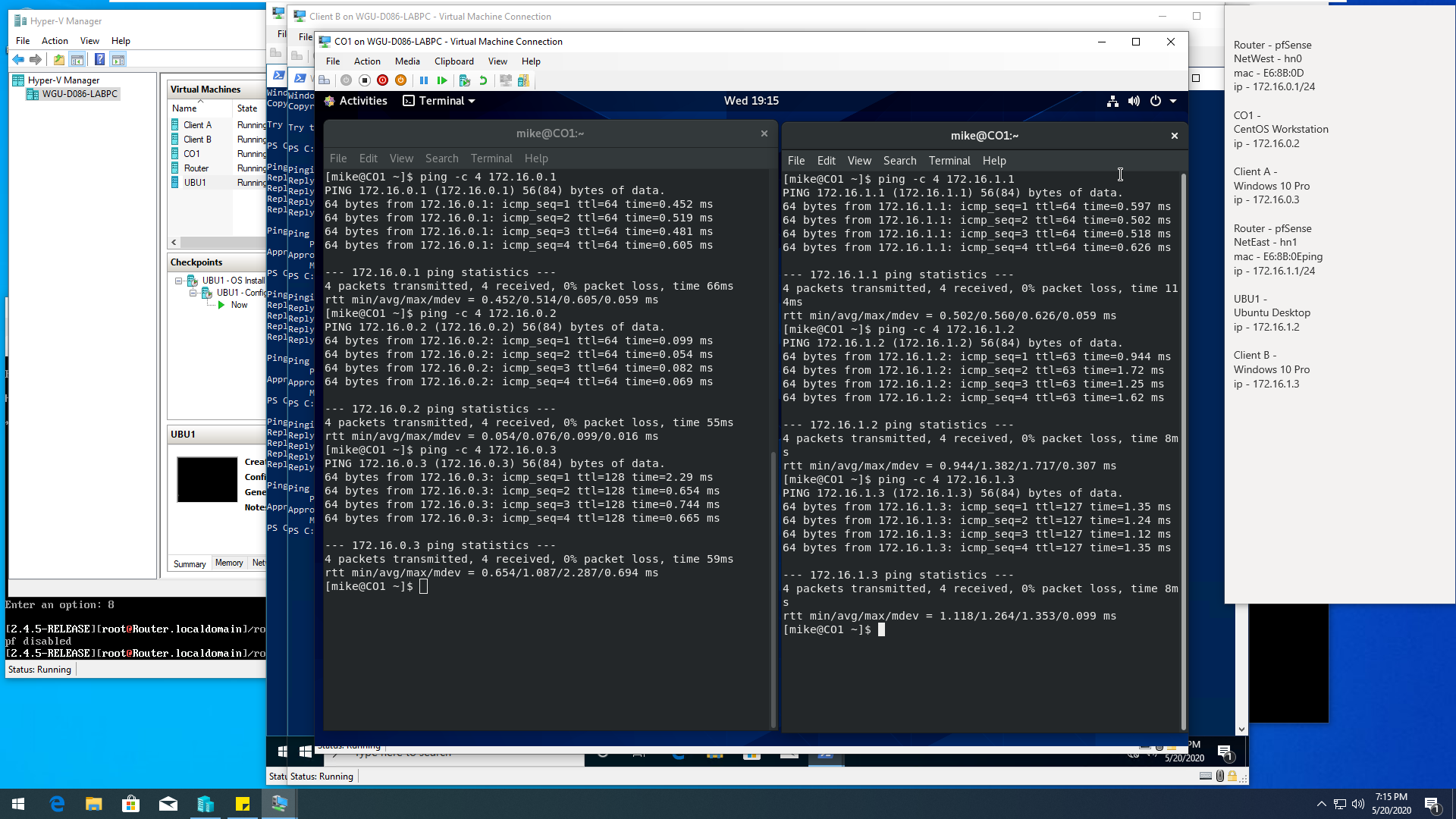
This shows Client B. The below screenshot shows the configuration of the client showing 2GB RAM, 2 processors, both HDDs, connected to NetEast while all VM’s are running.

This shows Client B able to ping all other VM’s, including itself, and both IP’s assigned to Router for NetEast and NetWest. For this step a firewall rule was added to allow ICMPv4 ping in for testing. On the firewall packet filtering was also disabled for testing.

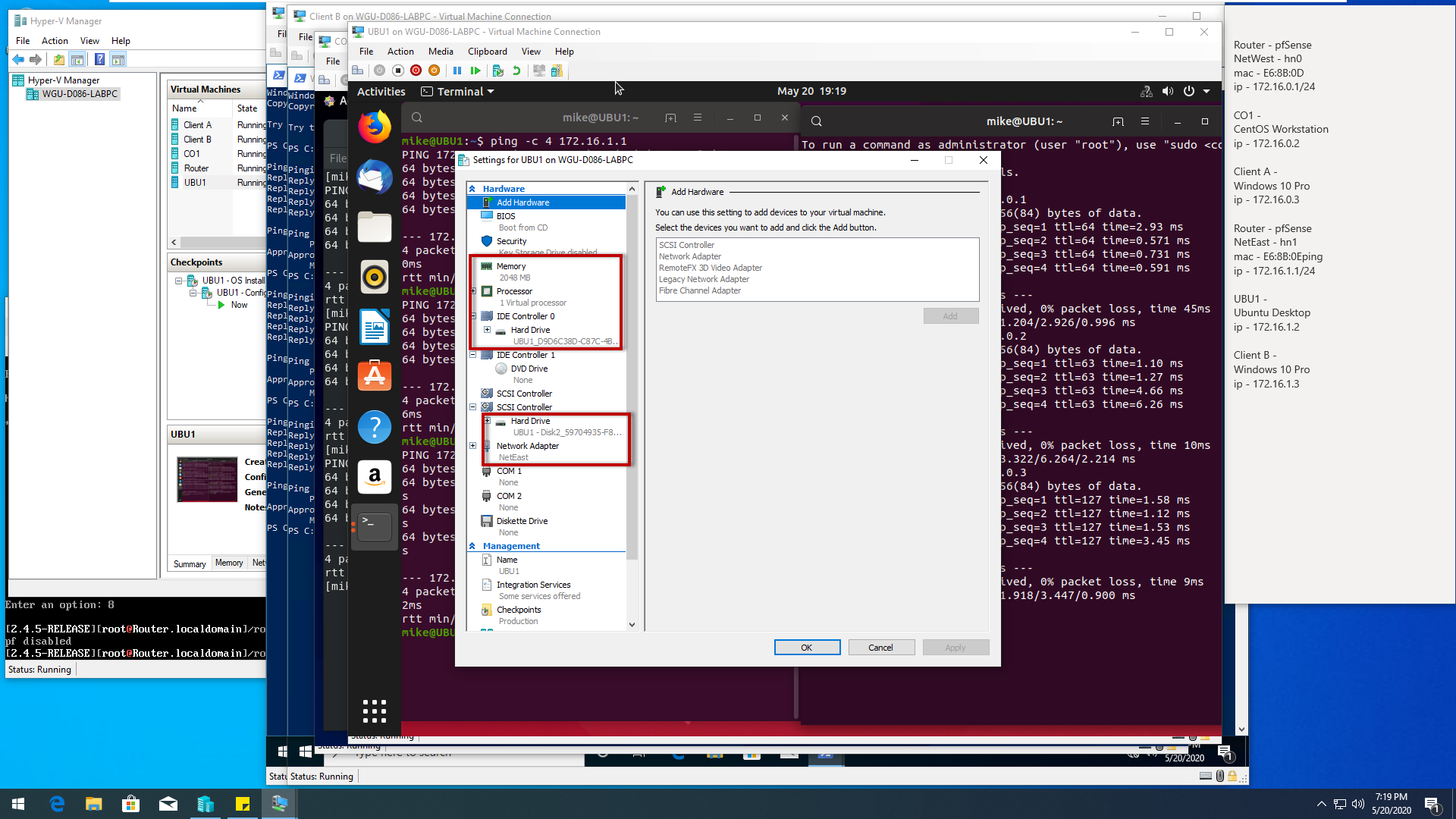


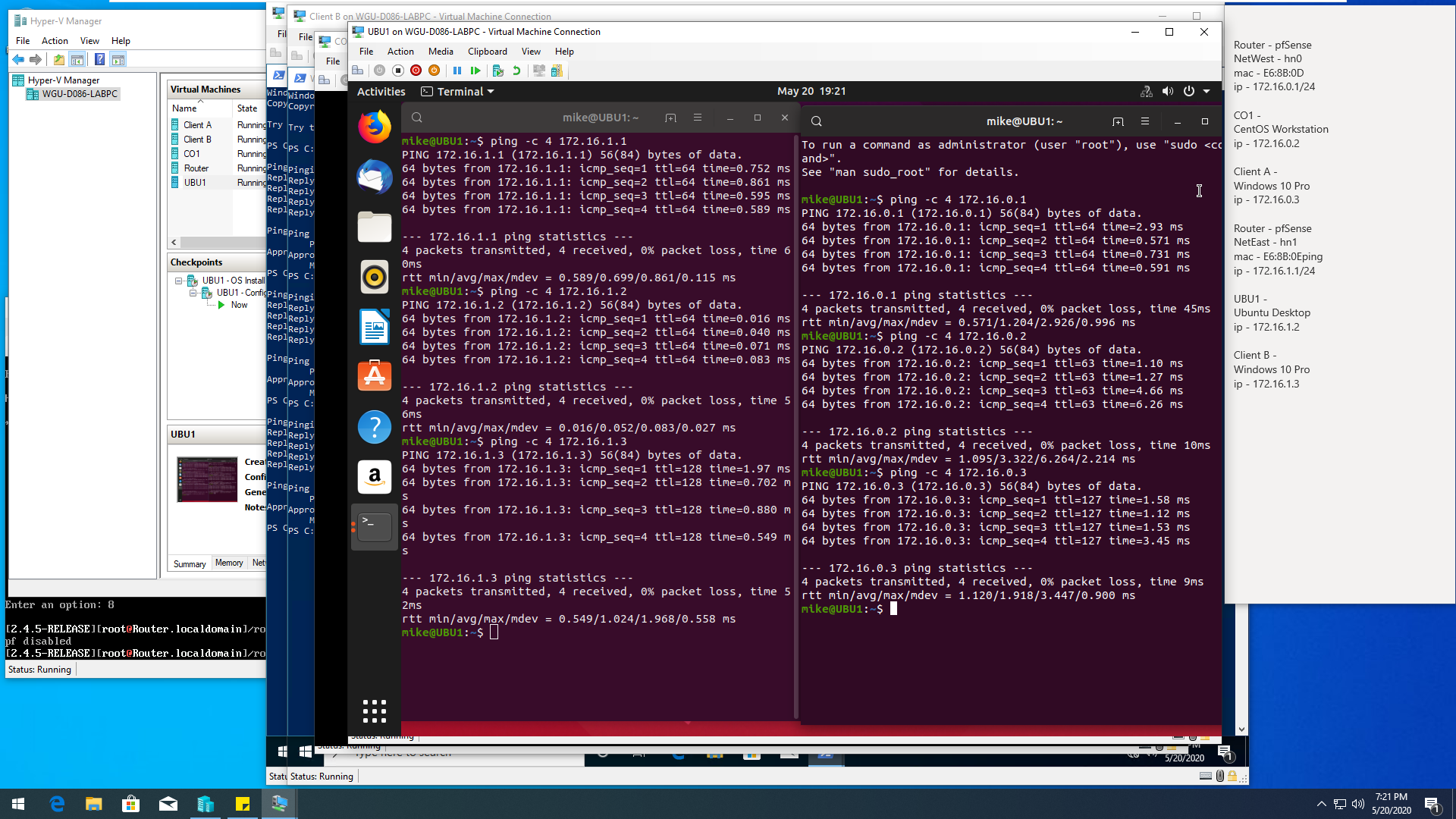
# Step Four – CO1

This shows CO1. The below screenshot shows the configuration of the client showing 2GB RAM, 2 processors, both HDDs, connected to NetWest while all VM’s are running.

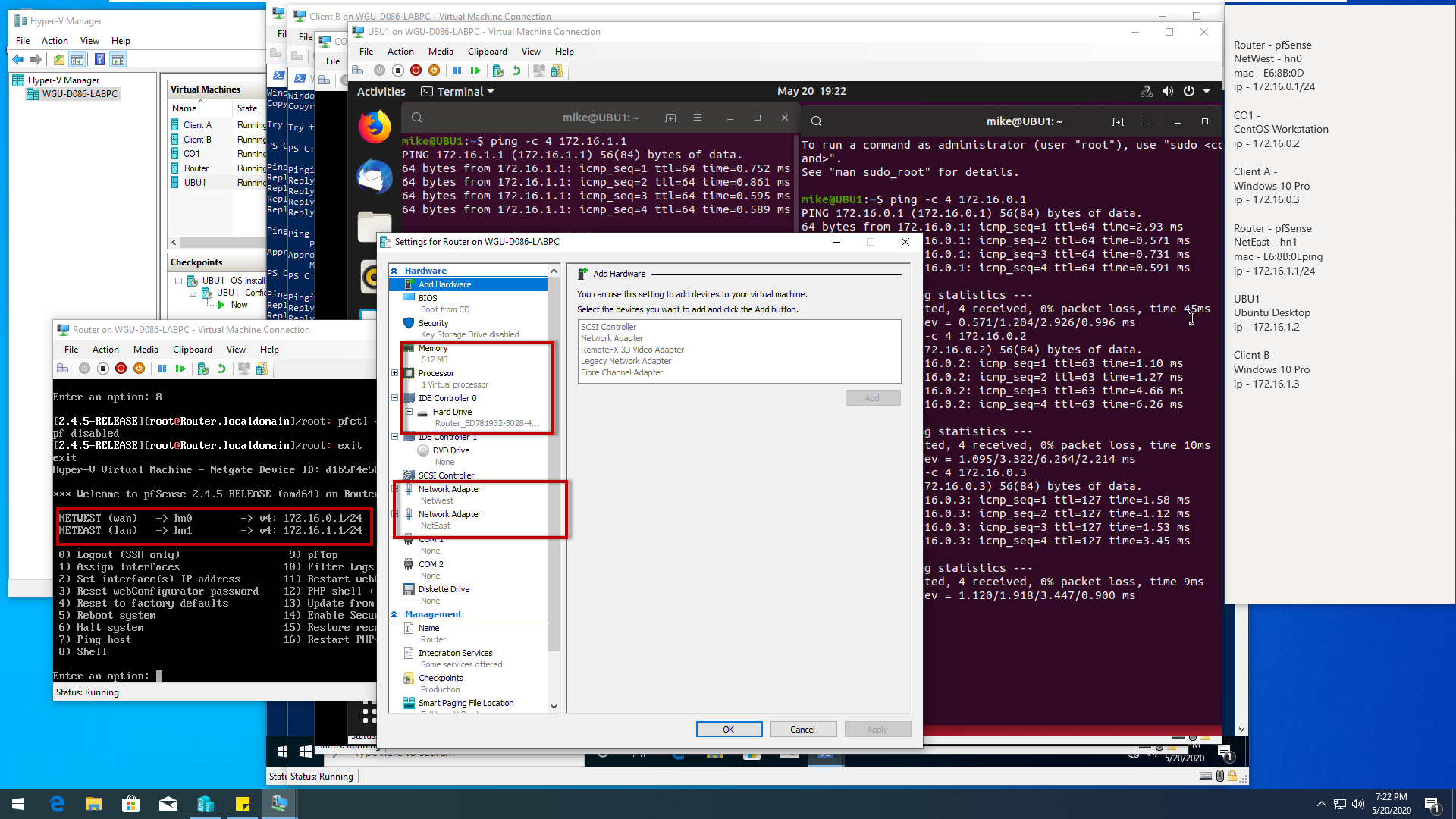
This shows CO1 able to ping all other VM’s, including itself, and both IP’s assigned to Router for NetEast and NetWest. On the firewall packet filtering was also disabled for testing.

# STEP Five – UBU1

This shows UBU1. The below screenshot shows the configuration of the client showing 2GB RAM, 1 processor, both HDDs, connected to NetEast while all VM’s are running.

This shows UBU1 able to ping all other VM’s, including itself, and both IP’s assigned to Router for NetEast and NetWest. On the firewall packet filtering was also disabled for testing. 

# Step 6 – pfSense Router

This shows pfSense Router. The below screenshot shows the configuration of the client showing 512MB RAM, 1 processor, 1 HDD, connected to NetEast and NetWest while all VM’s are running with each interface named with assigned IP’s.

The below show Router able to ping all other VM’s, and both IP’s assigned to Router for NetEast and NetWest. On the firewall packet filtering was also disabled for testing. 