IEO REPORT

Week 7 – Week 10

Student Name : Tobias Halomoan

Student Number : **4252861**

Class : **P-CB-06**

Group: 10

Fontys University of Applied Sciences

Eindhoven – Netherland

2020-2021

Contents of The Report

Revision Table 3
Introduction 4
WEEK 7 – Network Basics 5
WEEK 8 - IP 11
WEEK 9 – IP Routing 15
Week 10 – TCP/UDP 19
Conclusion 24
Personal Refraction 25

Revision Table

WEEK	Name of The File	Date	Time
Week 7	Report Week 7 – Week 10	Saturday,24/10/2020	15 : 20
Week 8	Report Week 7 – Week 10	Monday, 01/11/2020	16 : 37
Week 9	Added Week 9 items	Sunday, 08/11/2020	23 : 00
Week 10	Added Week 10 items	Sunday, 15/11/2020	21:15
Week 11	Edited reflections, finishing as final report	Sunday, 22//11/2929	20:19

Introduction

Networking is an essential function for computers. Every day, most computer systems interact with each other and transfer information through networks. Behind these processes lies a comprehensive networking system that deserves to be studied.

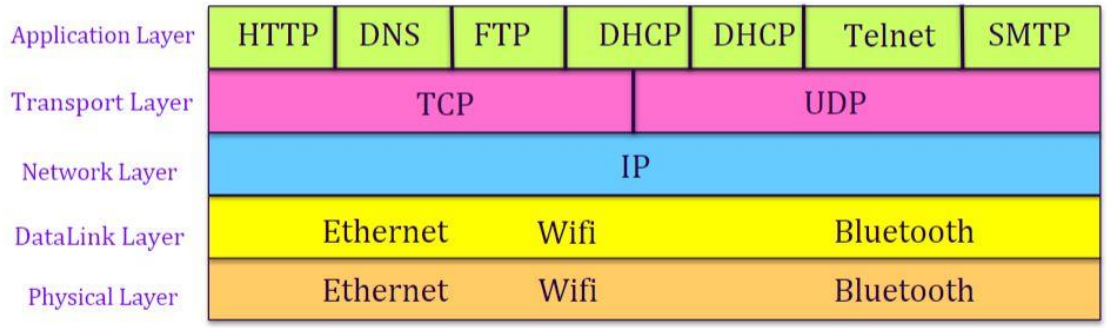


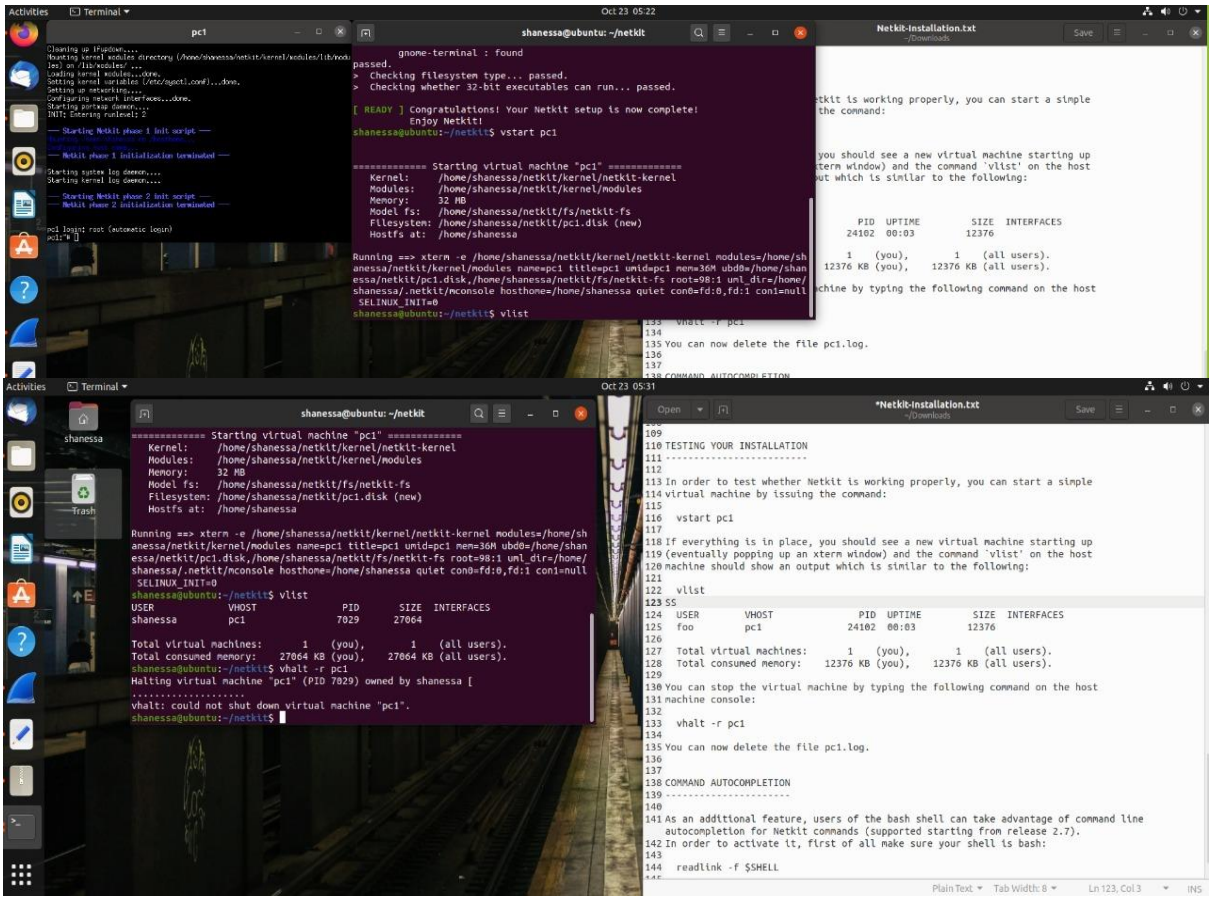
Figure 1. Layers of a networking system

In **Figure 1**, we can see the overview of the many layers of protocol and hardware used to execute networking operations. Throughout these 4 weeks, each layer (with the exception of the DataLink layer) will be dissected and analysed with the use of simulated learning environments (through Netkit and a Linux-based virtual machine), and network packet sniffing (Wireshark). Through this study, a thorough understanding of networking systems can hopefully be achieved, and would provide a valuable resource working in IT at a professional setting.

WEEK 7 – Network Basics

This week, we learned about the network and physical layer of networking, using Netkit and VMWare virtual machine running Ubuntu 20.04 LTS.

The Linux environment was chosen since Netkit, the program that allows for experiments in computer networking, functions solely on this OS. Also, the extensive command line functionality in Linux allows for a more flexible working environment.



The first step was to install and configure Netkit in our working environment. Installing in an unconfigured Linux was a challenge, but we managed to get it functional. By using the command **Vstart pc1** in Netkit, we created a network called PC1. Using the **Vlist** command, we can see the networks in the system, as shown in **Figure 2**. We stopped the program using the command **Vhalt -r pc1**.

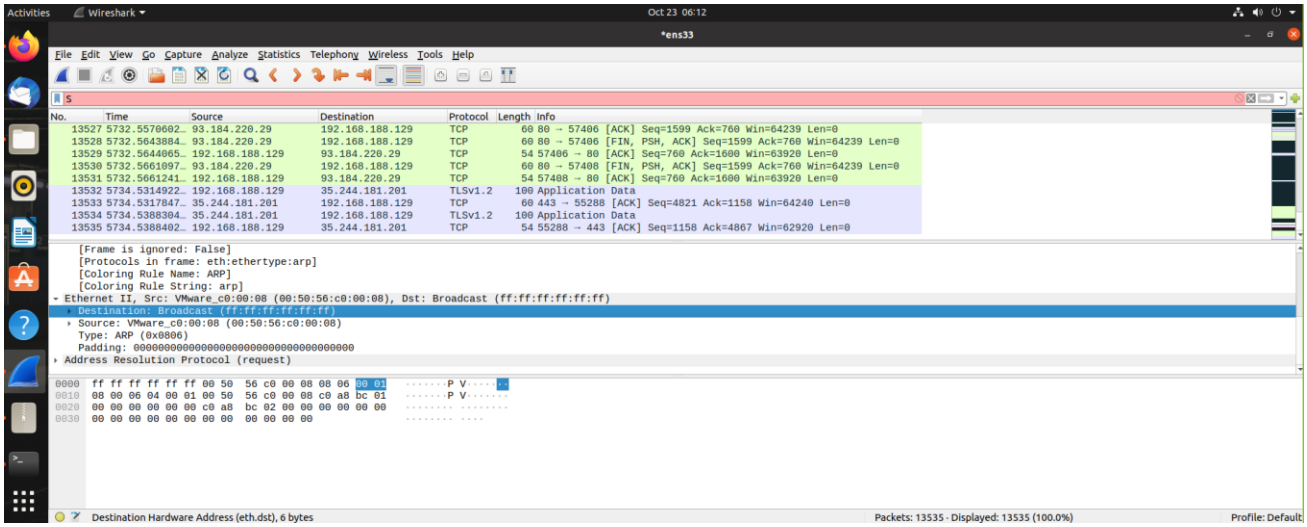
The second step was to install and configure Wireshark, a handy packet sniffing tool that shows what happens behind a network operation. Running the application in sync with a network operation allows us to see useful properties of the network in general, such as the MAC address, IP address, ports, host, and the HTTP GET to HTTP RESPONSE response time.

Figure 1. Layers of a networking system

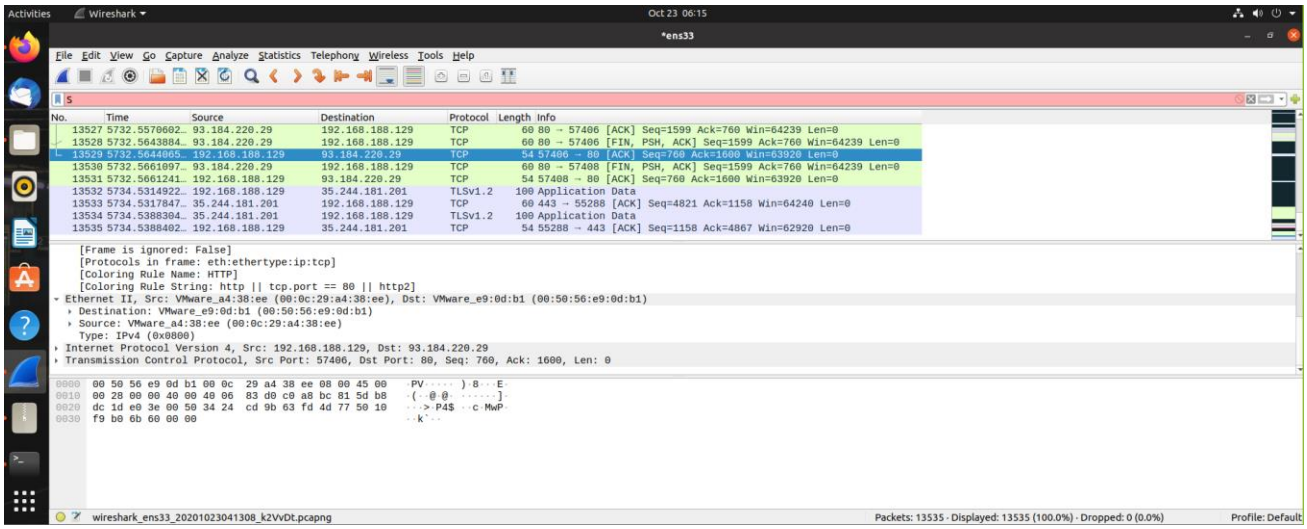
In order to generate HTTP packets for the Wireshark tool, we browsed the http://courses.codemax.net/w2.html website while Wireshark is on.

- *What is the source and destination MAC address of this HTTP packet?*
  Source MAC address: ***00:50:56:c0:00:08***
  Destination MAC address: ***ff:ff:ff:ff:ff:ff***
  Provide a screenshot below with the Wireshark snapshot and highlight these addresses:

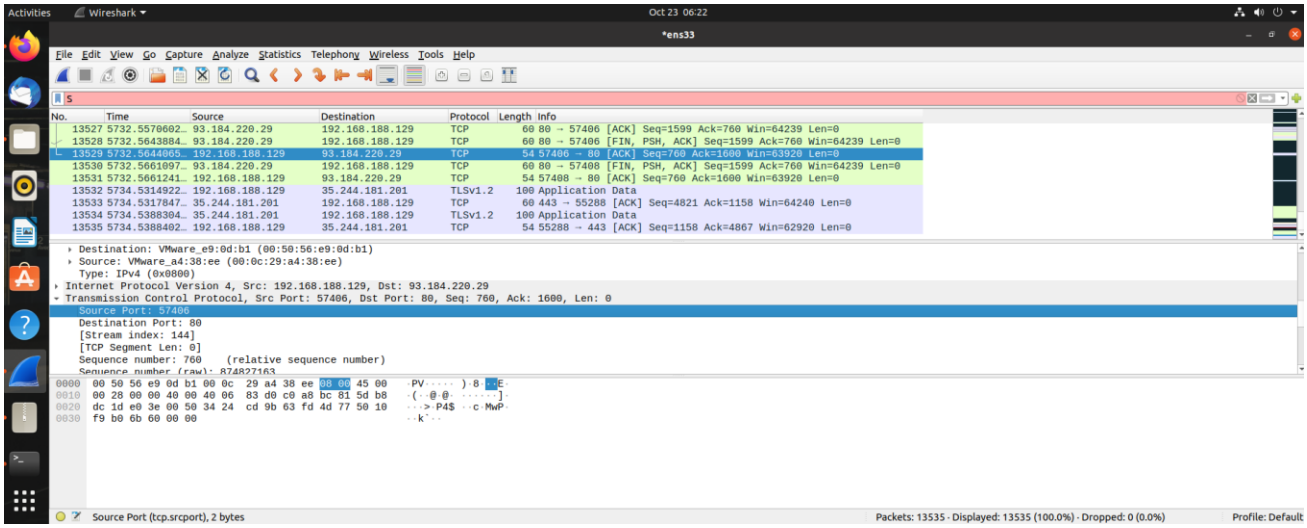
Figure 2.1 Testing Netkit



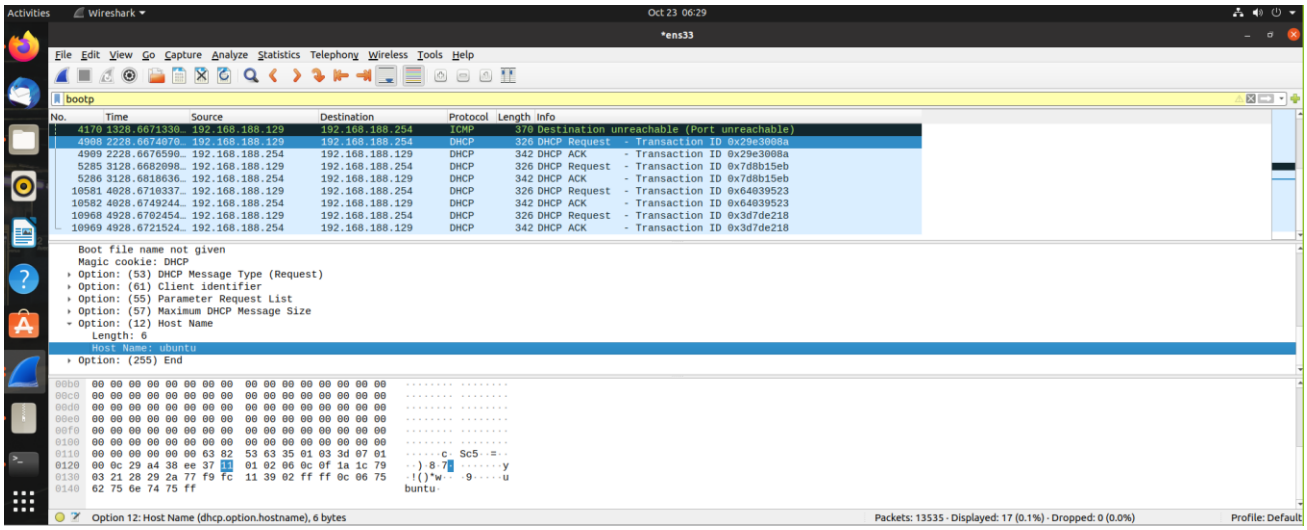
- *What is the source and destination IP address of this HTTP packet?*
  *Source IP address : **192.168.188.129***
  *Destination IP address : **93.184.220.29***
  Provide a screenshot below with the Wireshark snapshot and highlight these addresses:



- *What is the source and destination port of this HTTP packet? Provide a screenshot to prove it*
  *Source port : **63140***
  *Destination port: **80***
  Provide a screenshot below with the Wireshark snapshot and highlight these addresses:



- *What is the host name of this HTTP Get packet?*
  Host name: ***course.codemax.net\r\n***
  Provide a screenshot below with the Wireshark snapshot and highlight the host name:



- *Find the HTTP Response belonging to the HTTP Get packet. How much time elapsed between the HTTP Get and HTTP response?*
  Time elapsed: ***1.125780448 - 0.000103813 = 1.125676635***
  Provide a screenshot below with the Wireshark snapshot and highlight the elapsed time:

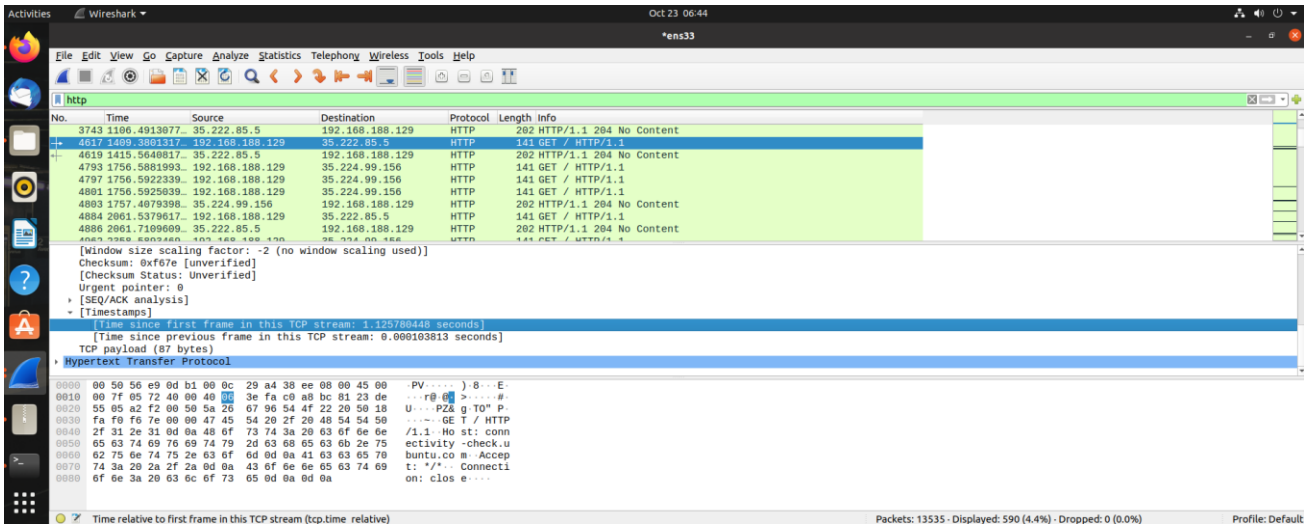


Figure 2.2 – 2.6 Testing Wireshark

In Wireshark, we discovered that a simple connection with the internet involves several addresses. In order to identify and communicate with each other, every computer network is assigned an exclusive number called the IP (Internet Protocol) Address. In the hardware layer, every computer is also assigned a MAC (Media Access Control) address.

To specify its destinations, every computer network is assigned ports, where the user would connect to the specified server's port and commit the data transfer. In this case, the server's name, or host name, is **course.codemax.net\r\n.**

In order to familiarize ourselves in the Linux environment, we also did exercises to learn the essential operations of the system.

Provide screenshots of all exercises in section 3.4

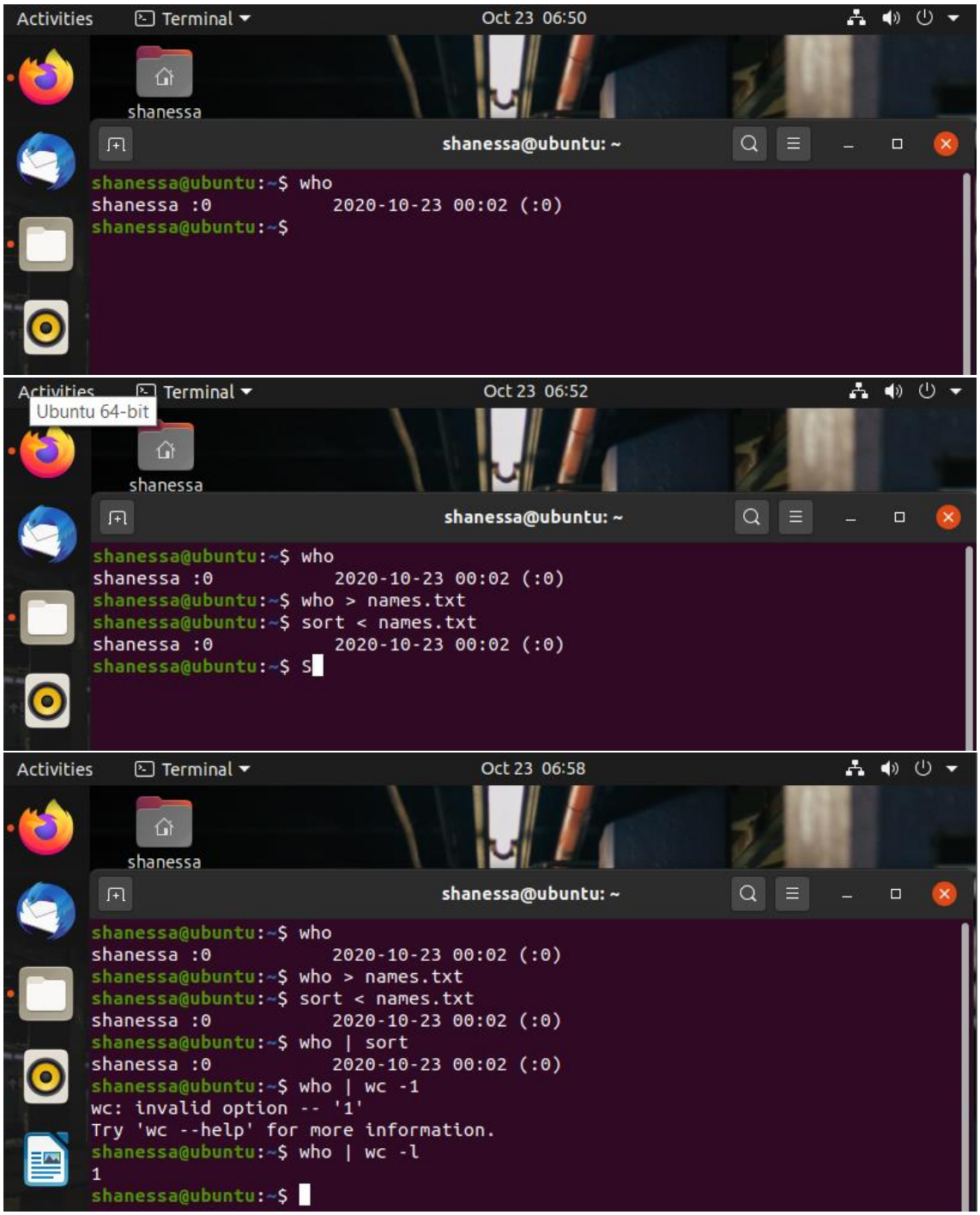
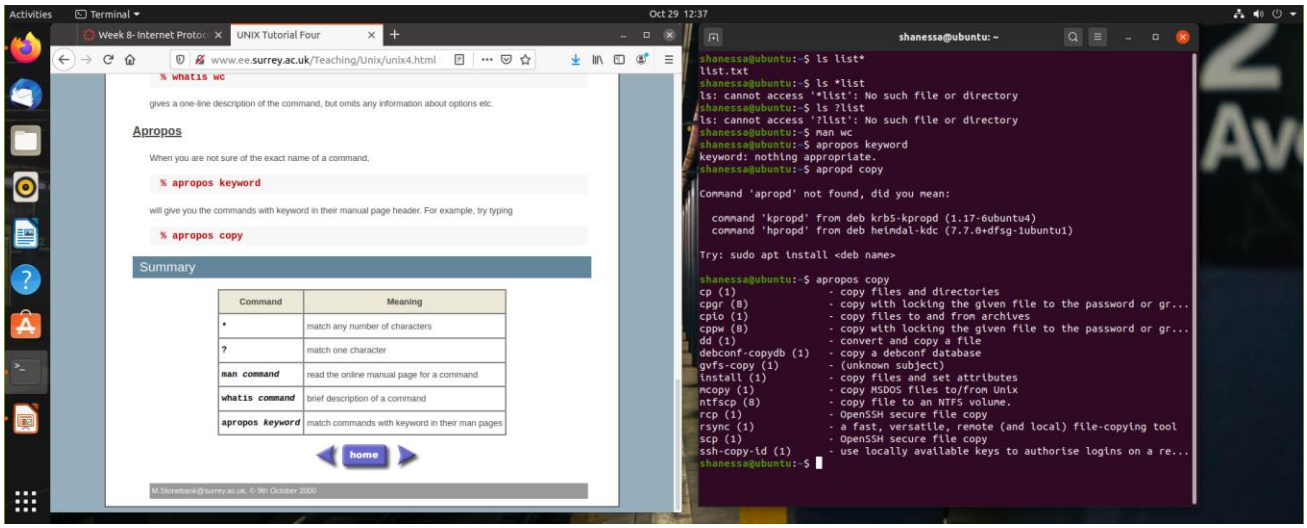


Figure 2.7 Testing Linux

WEEK 8 – IP

LINUX, STATIC IP ADDRESS/SUBNETS CONFIGURATION

***Task 1a**: Do Linux Tutorial*



***Task 1b**: Networking exercise*

Provide screenshots of all exercises.

Consider a Network Device with the following address:*255.120.213.32/20*

Which of the following IP addresses belongs to the same network?

- 255.120.207.46
- 255.120.206.27
- 255.120.224.123
- 255.120.207.16
- 255.120.213.86
- 255.120.225.128
- 255.120.224.237

Completed exercises

Classful IP addresses

address class	:	15 / 5
classful subnets	:	5 / 5

Classless IP addresses

8-bit aligned subnets	:	5 / 5
arbitrary subnets	:	4 / 4
network address ranges	:	2 / 2

Completed exercises

Classful IP addresses

address class	:	15 / 5
classful subnets	:	5 / 5

Classless IP addresses

8-bit aligned subnets	:	5 / 5
arbitrary subnets	:	4 / 4
network address ranges	:	2 / 2

***Task 2:** Build A Simple Netkit Network*

This week, we configured our Netkit to run multiple networks by using the pre-determined netkit lab. First, we learned about using the ping function, which is used to determine whether a network is connected to the computer. The ping command would allow the computer to

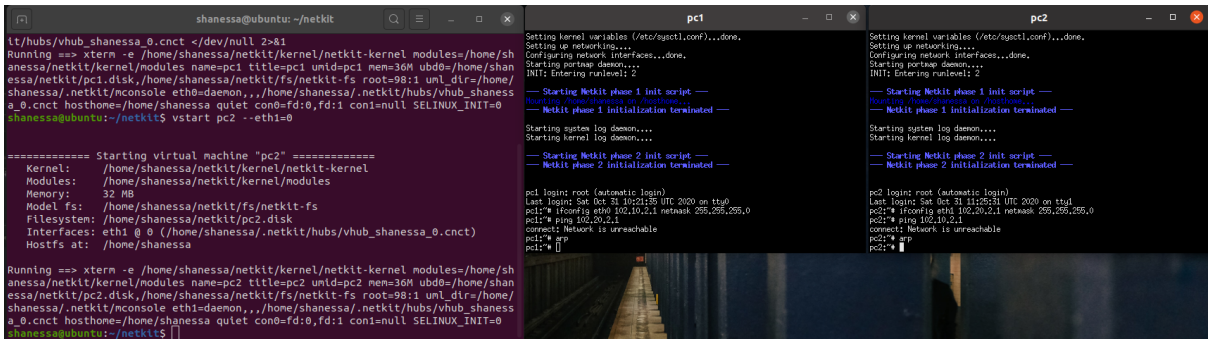
emit packets to a specified IP address, and if the packets are reflected back, it would indicate a connection was established.
We also learned on the use of ARP caches, which are useful as a log of the many networks that may be connected to a computer in a time.

1. *What is the result of the ping? Can you explain it? Provide a screenshot.*

- *The result of the ping is "Network is unreachable" because the subnet / the network is different, so the PC can not connect to each other.*

2. *Look at the ARP entries of your Node1 and Node2. Which command do you use? Which ARP entries are there?*

- *The command that I used is ARP*
- *The list on the ARP is empty because the PC are not connected to each other*



B) Configure the IP addresses of the 2 nodes by using the "ip" command explained in the theory lesson.

1. *Node1 has an IP address 102.10.2.1/10*
2. *Node2 has an IP address 102.20.2.1/10*

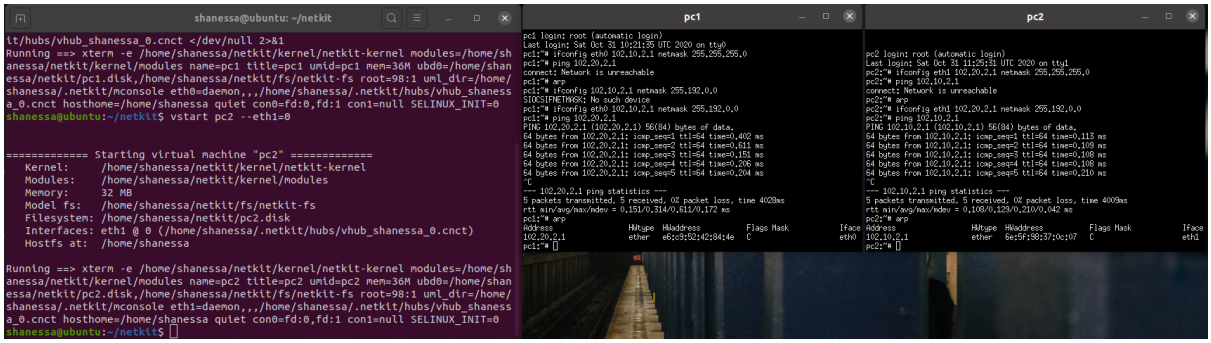
Check whether your configuration was successful by using ping command between these two nodes.

1. *What is the result of the ping? Can you explain it? Provide a screenshot of your configured interfaces.*

- *The result of the ping is succeeded, it's working. The PC are connected to each other.*

2. *Look at the ARP entries of your Node1 and Node2. Which ARP entries are there?*

- *The ARP of PC 2 already on the PC 1 and vice versa.*



C) Configure both nodes to have a subnet mask 255.255.255.0, and change the IP address of Node2 in such a way that the ping between them is successful.

1. *Provide a screenshot of your configuration and successful ping.*
2. *After successful ping ARP entries of both nodes should be changed. Provide a screenshot of the new ARP situation and explain it. What is the command to clear the ARP cache again*

- *I changed the IP address of PC 2 and the ARP list showed us the new IP address of PC 2 on the PC 1. And when we want to clear the ARP cache we can use command "arp -d and put the IP address", e.g. arp -d 102.10.2.2*

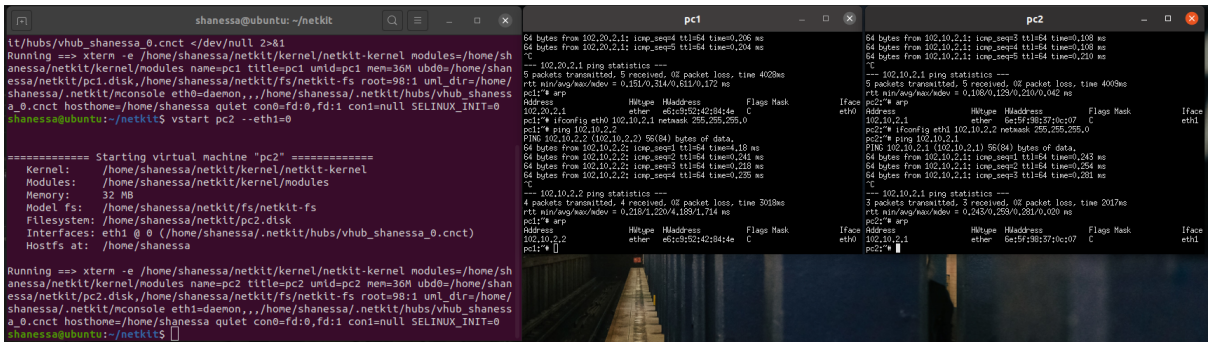
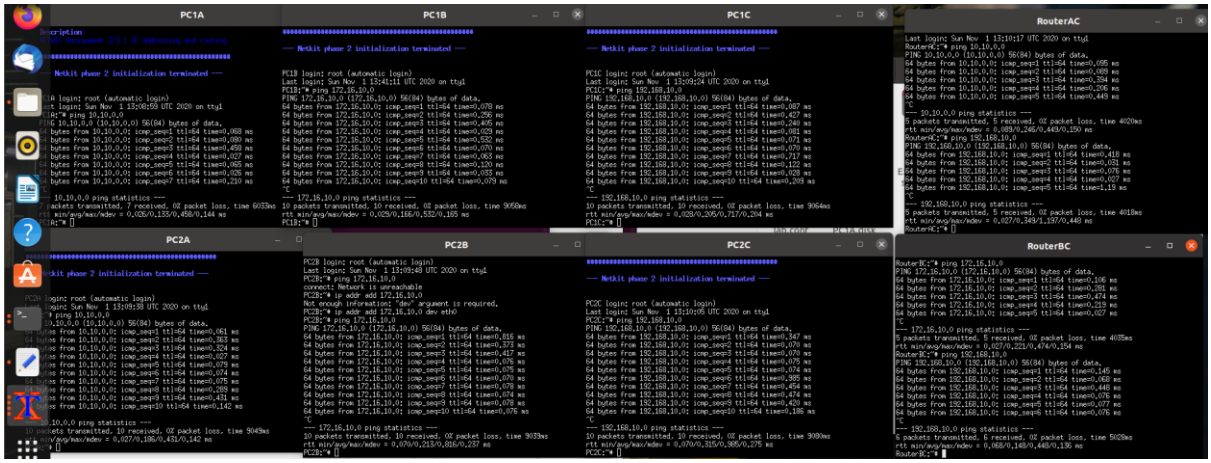


Table 1 : IPv4 address ranges per student group

Group	LANA	LANB	LANC
1	10.1.0.0/16	172.16.1.0/24	192.168.1.0/24
2	10.2.0.0/16	172.16.2.0/24	192.168.2.0/24
…			
n	10.n.0.0/16	172.16.n.0/24	192.168.n.0/24

This week, we configured multiple networks in a single computer using the ifconfig command on our netkit lab startup files. We assigned it as according to **Table 1**, in this case n= 10 as it is our group number.



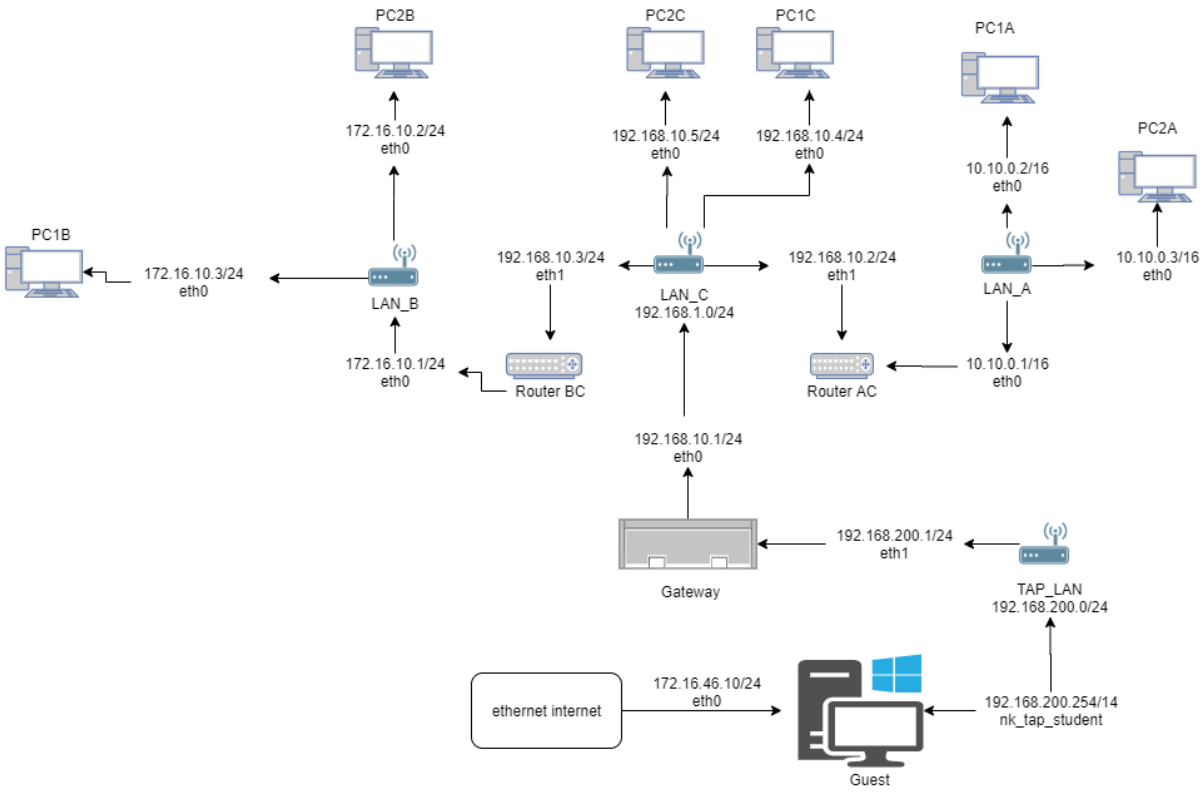


Figure 3.1 Network Diagram

Task 4: CIDR IP Addressing Exercises

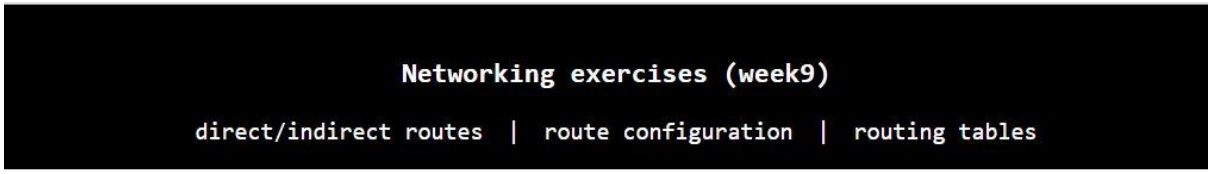
1. Suppose we have IP address 122.33.196.145/24
Fill in the following items for this address:
   1. Network Address : **122.33.196.0**
   2. Broadcast Address : **122.33.196.255**
   3. Subnet Mask : **255.255.255.0**

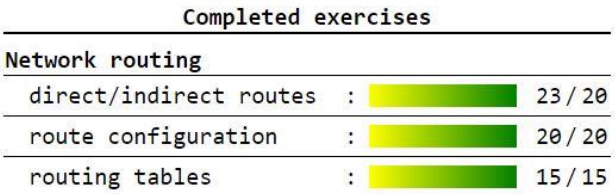
2. Suppose we have IP address 163.249.223.229/25
Fill in the following items for this address:
   1. Network Address : **163.249.223.128**
   2. First Host : **163.249.233.129**
   3. Last Host : **163.249.223.254**
   4. Broadcast Address : **163.249.233.255**

WEEK 9 – IP Routing

IP Routing

Task 1a: Online exercises



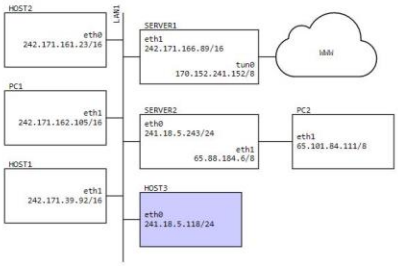


Consider the routing table of *HOST3*

Destination	Gateway	Genmask	Flags	Iface	
242.171.0.0	0.0.0.0	255.255.0.0	U	eth0	[?]
241.18.5.243	0.0.0.0	255.255.255.255	UH	eth0	[?]
65.101.84.111	241.18.5.243	255.255.255.255	UGH	eth0	[?]

Type the missing default route command so that all network nodes are reachable from ***HOST3***

route add default gw 242.171.166.89 done



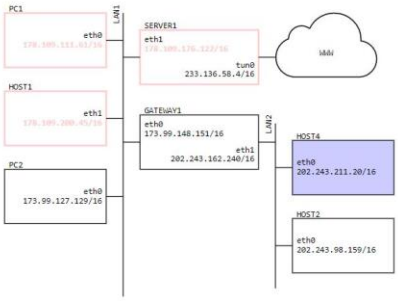
Completed exercises

Network routing		
direct/indirect routes	:	23 / 20
route configuration	:	20 / 20
routing tables	:	14 / 15

Consider the following network diagram:

5/5 Fill in the command below so that ***HOST4*** can send packets to ***178.109.0.0/16***

route add -net 178 . 109 . 0 . 0 netmask 255 . 255 . 0 . 0 gw 202 . 243 . 162 . 240 next



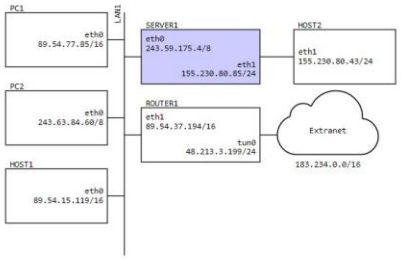
Completed exercises

Network routing		
direct/indirect routes	:	23 / 20
route configuration	:	15 / 20
routing tables	:	0 / 15

Consider the following network diagram:

1/3 ***SERVER1*** can best access ***Extranet*** using…

(select)



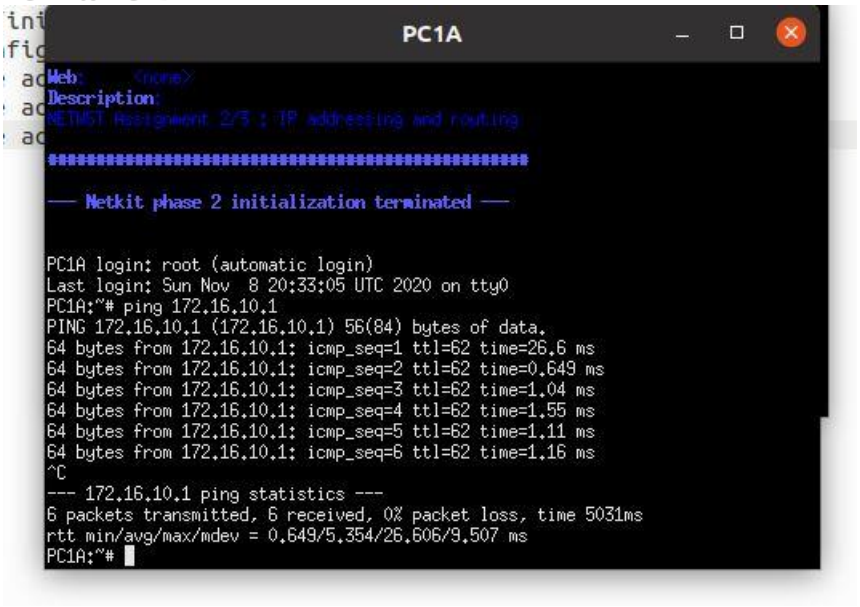
Completed exercises

Network routing		
direct/indirect routes	:	19 / 20
route configuration	:	0 / 20
routing tables	:	0 / 15

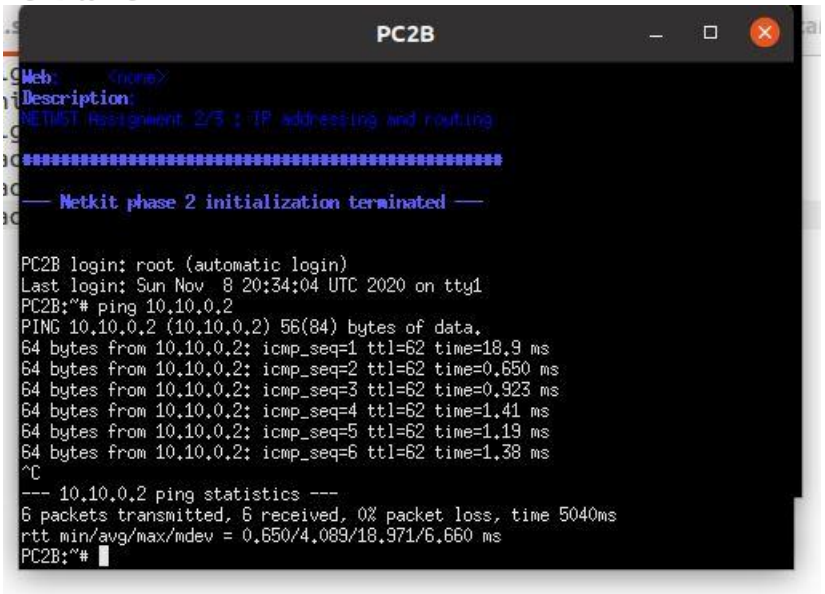
Task 1b: A bit more complex network: Part 2

This week, we used our configured IP networks from the previous week and used route commands to connect each network with each other and connect each router with their specific set of networks, as shown in **Figure 3.1**. The results were given below.

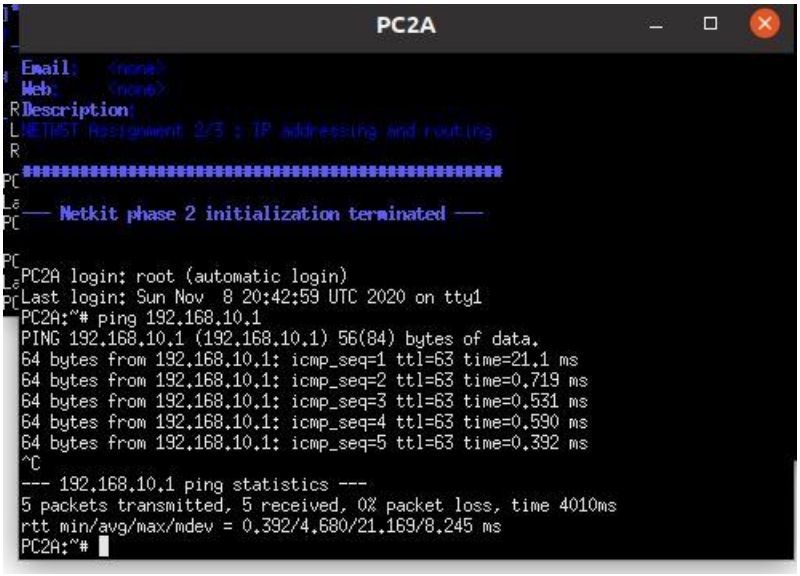
1. **PC1A to PC1B**



2. **PC2B to PC2A**



3. PC2A to PC1C



Give a list of all nodes where you had to adjust the routing tables and the screenshots of their configured routing tables.

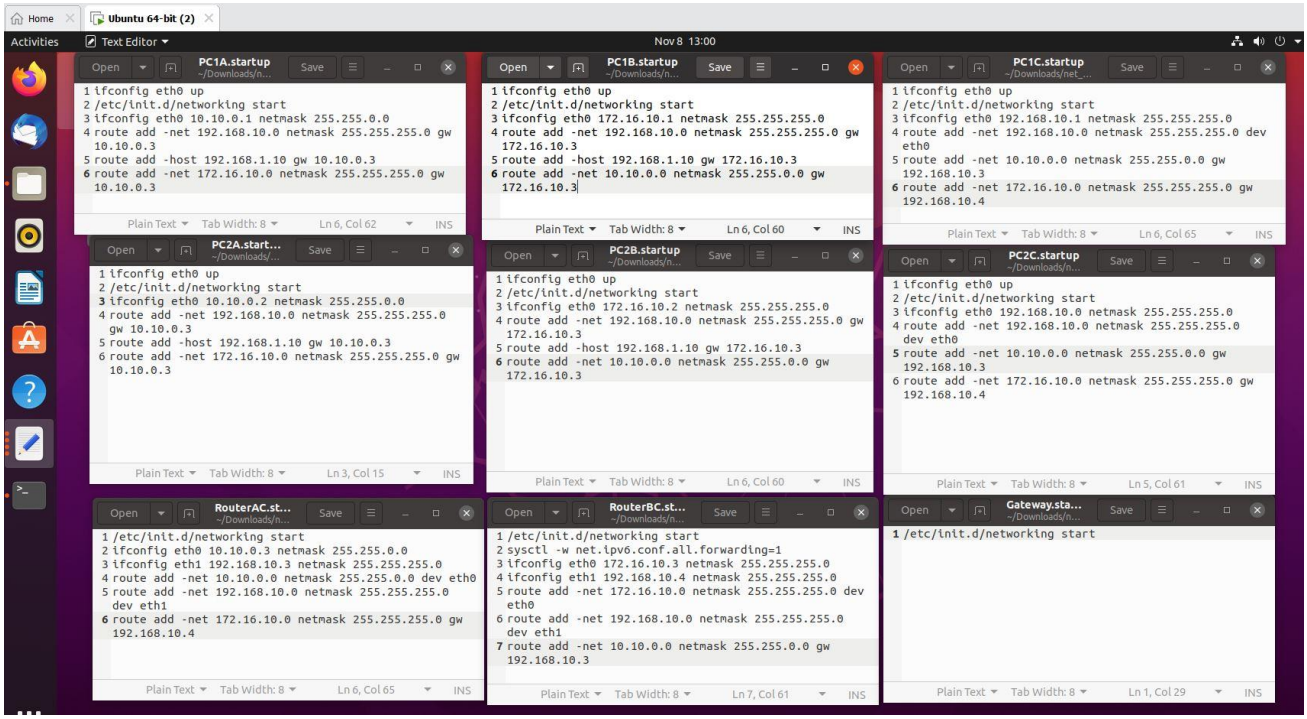


Figure 4.1 Building a routed connection using route commands in our netkit lab startup files.

WEEK 10 TCP/UDP

Task 1: TCP in Netcat

To do this assignment we will use the Netcat tool which is provided in the Netkit. Netcat makes it possible to create and use TCP/UDP connections. If you want more info about Netcat you can consult Internet. To make this assignment we will reuse the net_routing lab from the previous assignments. Let's start a chat session by connecting 2 netcat instances via a TCP connection.

To listen to the TCP connections, go to one of your simulated nodes (e.g. PC1A) and issue the following command:

nc –l –p <port_nr>

This will make netcat listen to port number that you have specified in port_nr and accept connections.

Note: Any port number would be ok, as long as it is not used by another application.

To establish a TCP connection you can issue the following command from another simulated node (e.g. PC1C)

- **nc <IP address of the "listening" node> <port_nr of the "listening node">**

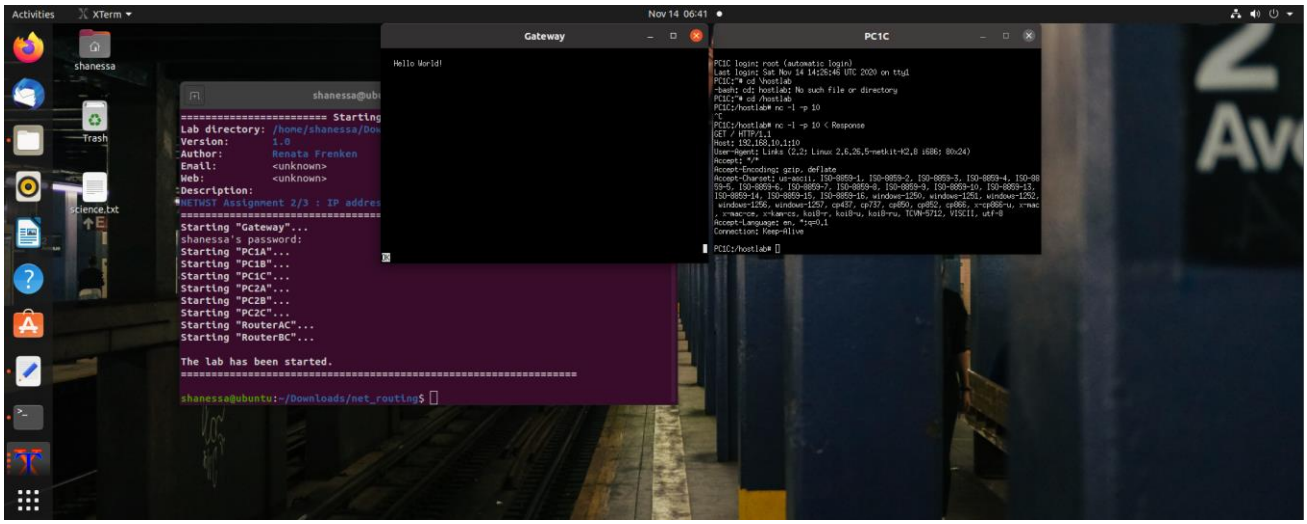
This will make a TCP connection with the listening netcat instance.Now you can chat from one netcat instance to the another. Try it out!

Your task:

- Netcat can also be used to copy the contents of a file from one place (file, folder, computer) to another. Find out how and try it out.

Provide screenshots of the sending and receiving command.

Provide a screenshot of the netcat command you used and of the links browser output.

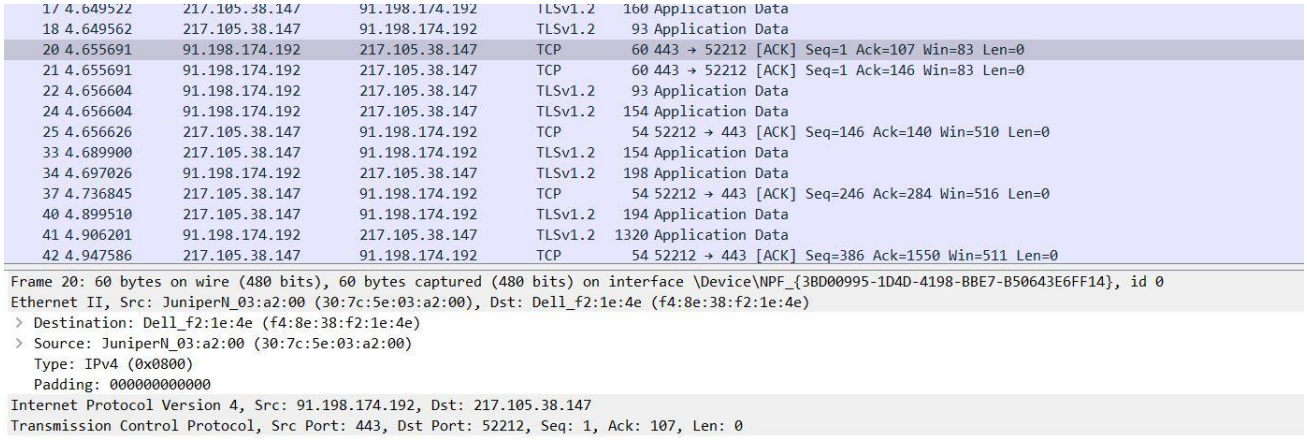


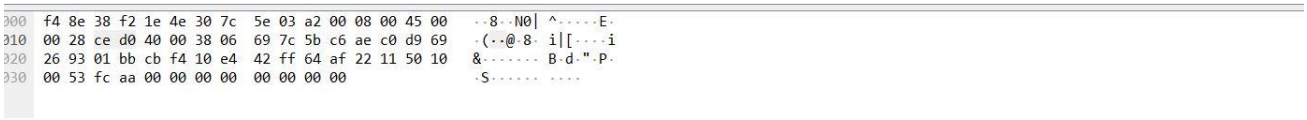
Task 2: Find 2 TCP uses

Describe the chosen scenarios and a proof of TCP use in them by attaching a Wireshark trace showing TCP packets.

Scenario 1 : HTTPS Traffic

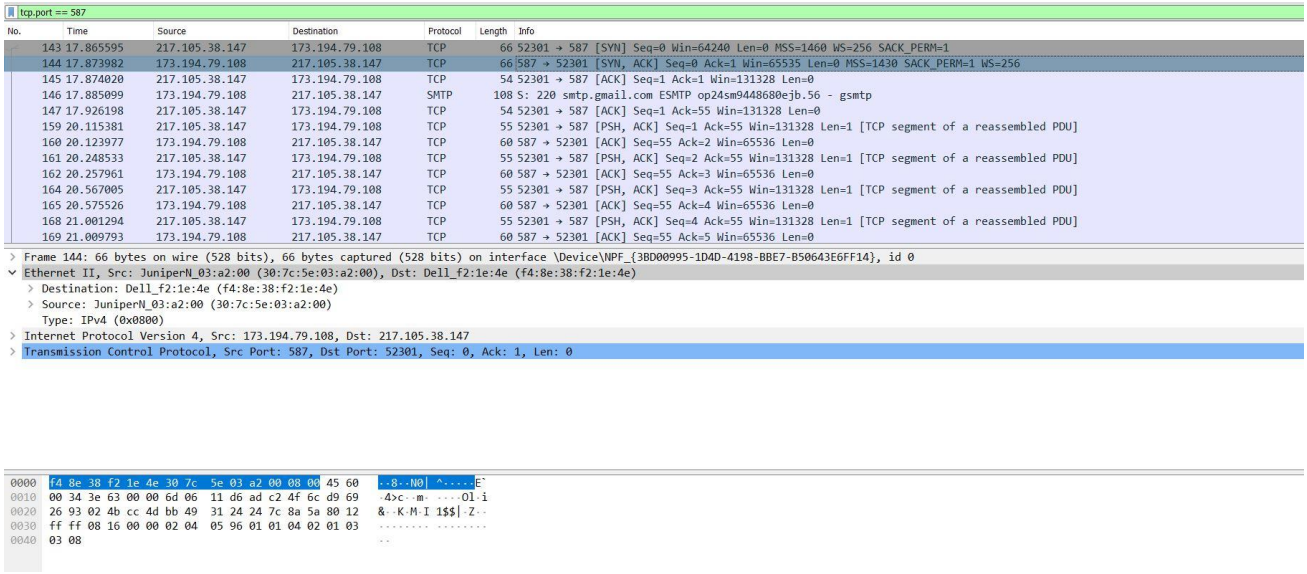
We capture HTTPS Traffic through opening multiple websites for a couple of seconds. We found traffic to the port 443. HTTPS is used for secure communication over computer networks and is widely used in the internet.





Scenario 2: SMTP Mail Server

We connect to the Google SMTP Mail Server using the Windows Telnet client to generate the traffic. The SMTP is a widely used communications protocol for email transmission. The TCP port was port 587.

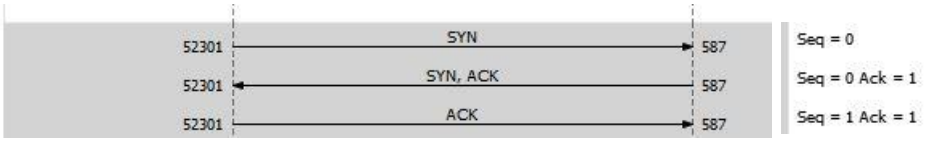


The Beginning:

The client (my computer) establishes a connection to the server, using the SYN (Synchronize sequence number) which informs server that client is ready to communicate.

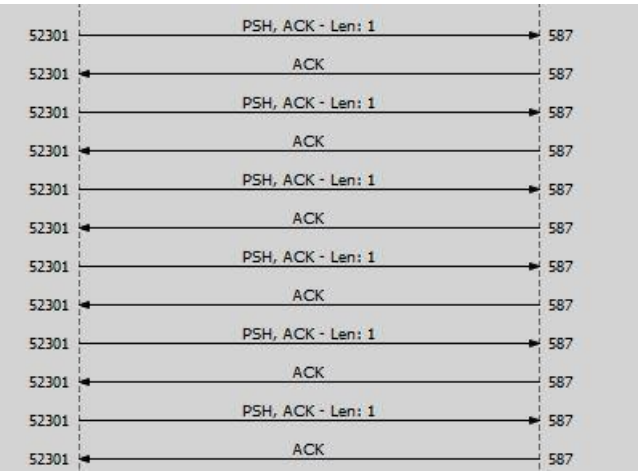
Then, the server responds with SYN + ACK signal, ACK (Acknowledgement) signifies the OK signal from the server side, and SYN signifies which sequence number it will try to start the transfer with.

The final ACK from the client side signifies the OK signal from the client side, finishing the “Three Way Handshake” and establishing a reliable connection between the two.



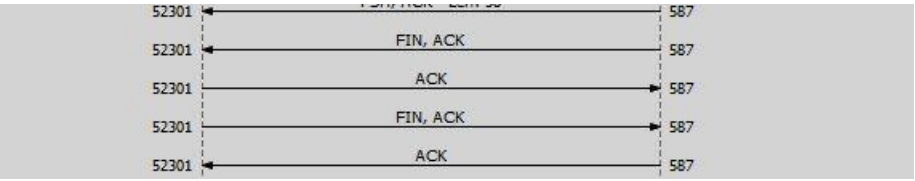
The LEN values signifies the length of the TCP payload in bytes. The SEQ value signifies the sequence number for communication, and the ACK value indicates that the server has acknowledged the client’s SYN flag and will proceed with the transfer.

The Middle:



In the middle, data transfer is done by continuous transfers and acknowledgement between the two parties. The PSH is the PUSH flag, that signifies the sending of data even when the buffer is not full during transfers!

In the End:

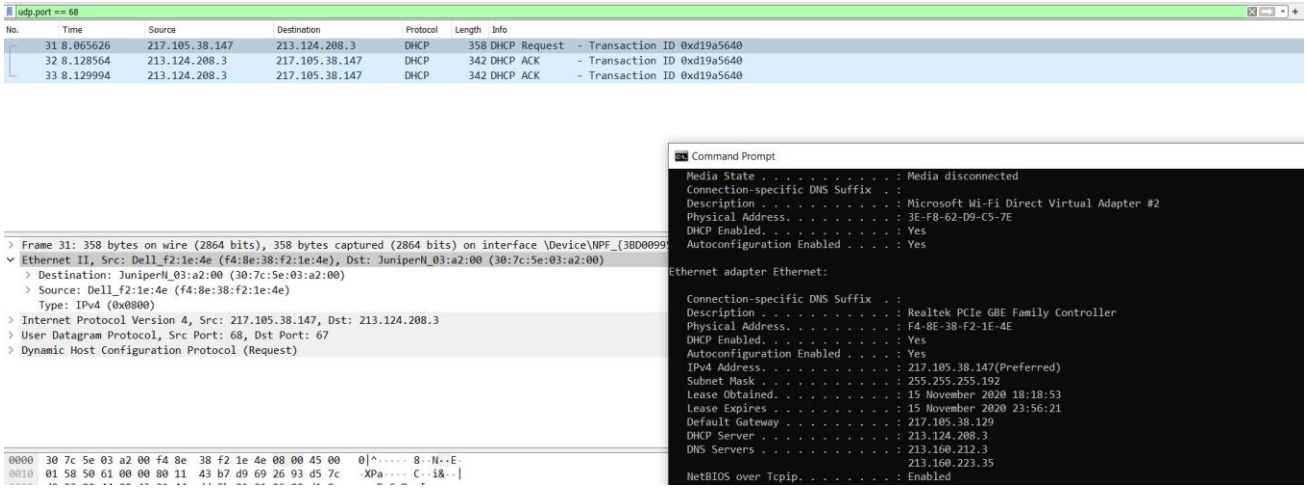


In the end, the FIN (FINISH) flag is established. This flag signifies the end of data transfer and the closing of the connection.

Task 3: Find 2 UDP uses

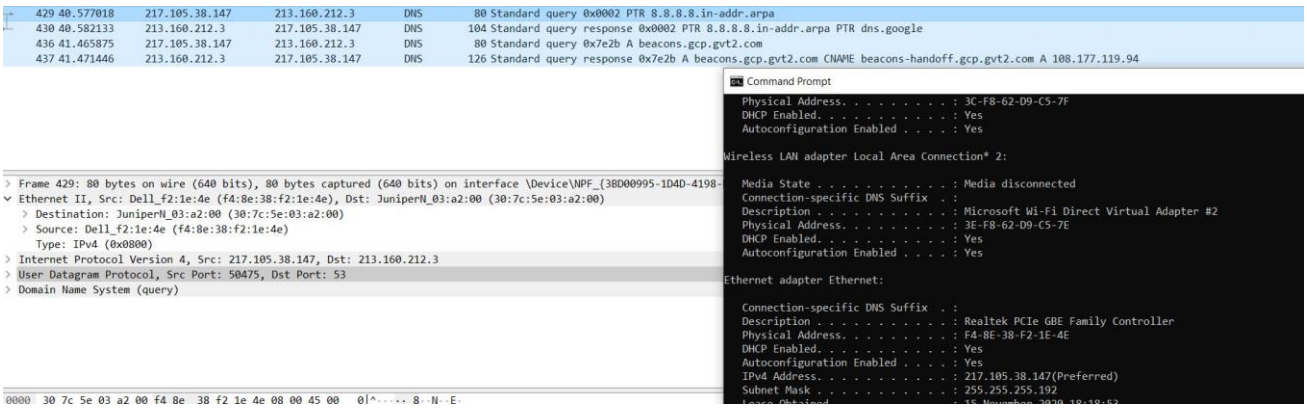
Scenario 1: Renew DHCP:

Using the ipconfig /renew command in the terminal, my DHCP assigned IP address is renewed. This will generate UDP traffic, particularly in port 68. Dynamic Host Configuration Protocol is a network management protocol that dynamically assigns an IP address to devices.



Scenario 2: Reverse DNS Lookup (rDNS)

By using the command nslookup 8.8.8.8, the Domain Name System protocol, which uses UDP, looks up the aforementioned IP address. DNS is a naming system that acts as a "finder" which translates domain names to the numerical IP addresses needed for locating networks under many different networking protocols. Reverse DNS Lookup generates traffic particularly in port 53.



Conclusion

The past 4 weeks have been an eye-opening look towards the inner mechanisms of networking. The weekly assignments provided a gradual understanding towards each layer of networking operations. Our research allowed us to learn the different terms, properties, and methods used in networking, which would be invaluable in a professional setting.

On Week 7, we learned on the networking and physical layer of networking through our experiments on Netkit and Wireshark. On Week 8, we investigated Netkit much further and managed to configure our own computer network. On Week 9, we connected our simulated computer network with each other and learned the basics of routing. On Week 10, we researched the difference between TCP and UDP transport protocols and its uses on the application layer, especially in the common networks that we used daily.

Of course, multiple challenges arose during our work. There are numerous times that we were halted due to the complicated nature of the networking systems. Incidentally, through our own personal research and discourse with our peers, we managed to find the solutions needed to complete these assignments.

The internet, the computer, and the networking systems are remarkable achievements of humanity. It ushered us unto the information age, where borders are blurred, and the world becomes much smaller than it used to be. Networking allows us to be more empathetic towards our global community; it gave us a valuable tool to access information and gain knowledge like never before.

Personal Refraction

In the final week, we created a presentation to report our findings to our peers and teacher. It is a 10-minute demonstration that were meant to show our increased understanding of computer networks and brush up on our presentation skills. The file can be found in our github and attached with the zip file alongside this report.

In this session, we were hoping to receive feedback from our peers in order to improve our presentation skills. We managed to acquire a few, as shown in **Figure 4**.

A positive feedback that we received was that our presentation was comprehensive, and the information provided were gladly received. Our peers also thought that we were engaging enough in presenting, in the sense that our explanations were clear and in sync with the written content of the presentation.

However, being comprehensive has its significant drawback. Our peers thought that the presentation was a bit wordy, and it showed as it ended up being longer than our assigned time frame. One peer recommended us to remove some sentences which might be deemed unnecessary.

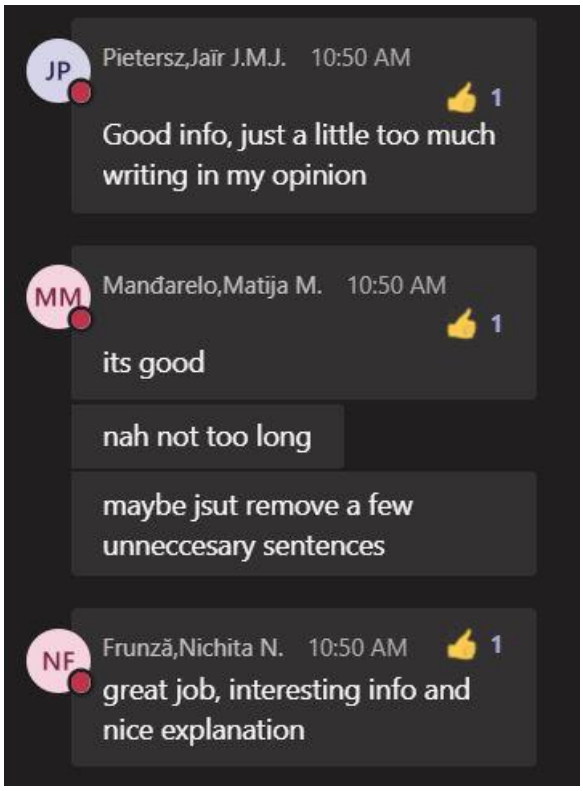


Figure 4. Feedback from our peers

The feedback showed us a particular challenge of presenting a topic: balancing between providing the right amount of content and completing the demonstration in a short timeframe. Significant reworking of the presentation in order to be more concise in our content and explanations would be necessary. Learning this fact was great for our self-improvement in presenting, which of course will be invaluable in a professional setting.

Personally, I enjoyed the week 10 assignments, especially in simulating the different communications protocol necessary to gather examples of TCP and UDP transport protocols usage in our daily lives.

One Netkit project exercise that would be interesting to research on is in using the simulated networking environment to generate cyber attacks, such as Denial of Service (DDos) attacks, spoofing attacks, and syntactic attacks using viruses, worms, etc. Through this exercise, I feel that we would gain some insight on the inner undertakings of how such attacks would be executed and how to prevent/combat it.