LAB 1

CONSTRUCT A SIMPLE NETWORK

Name: Nguyễn Võ Thuận Thiên

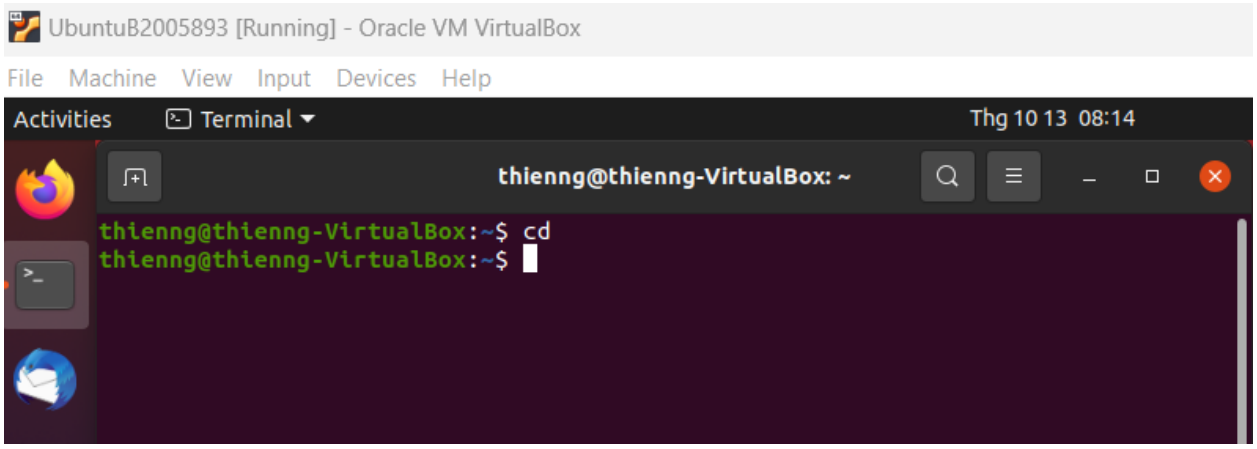
ID: B2005893

Group: M01

Submission: an ID_NAME_Lab01.pdf file describes clearly how did you solve the problem

Exercise 0: My home directory

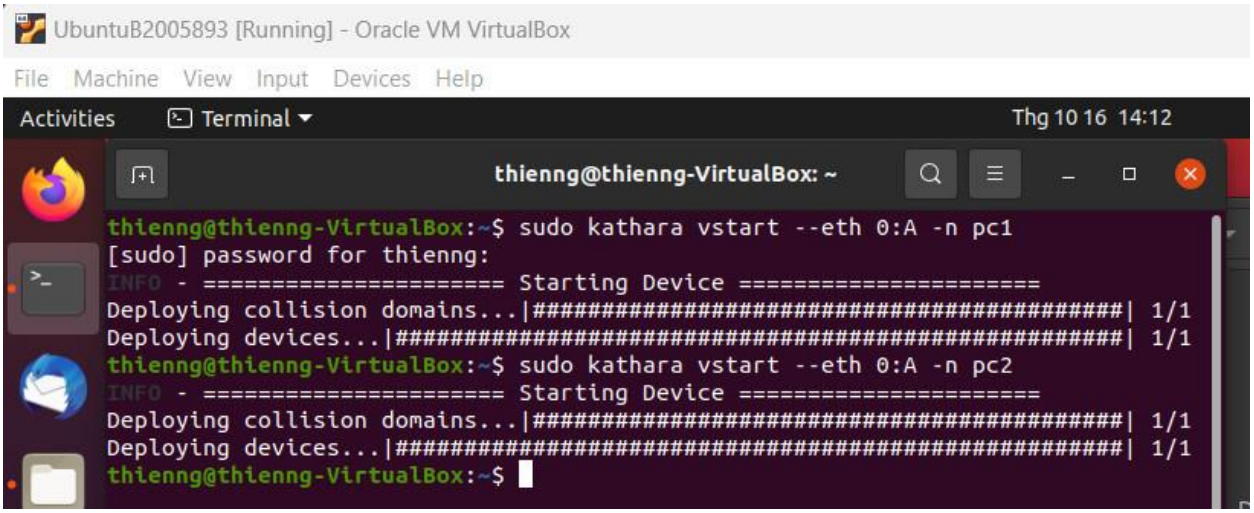
Answer: $cd



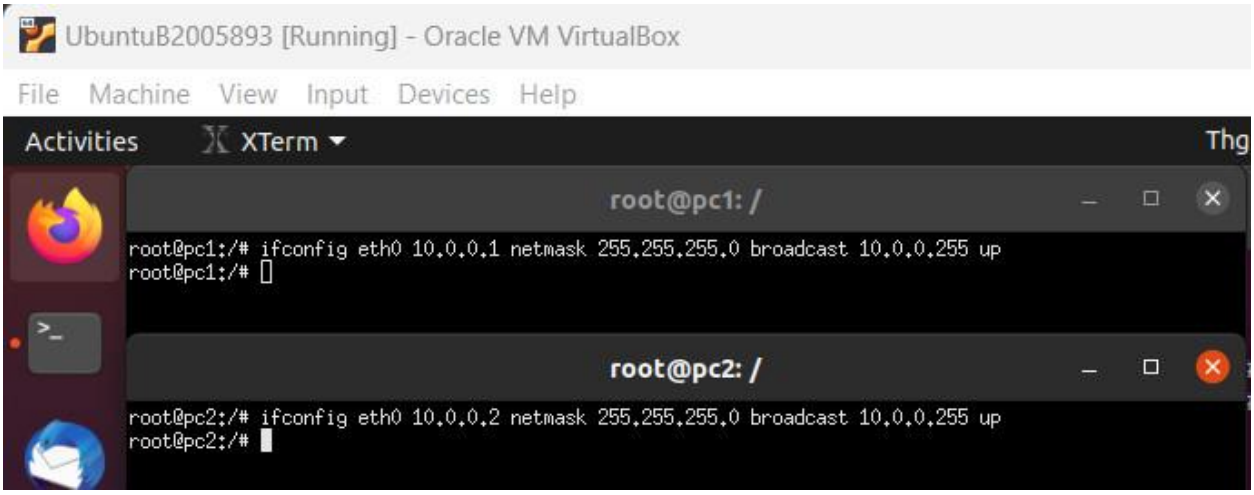
Exercise 1: Construct a simple network with two hosts connected to the same collision domain

Answer:

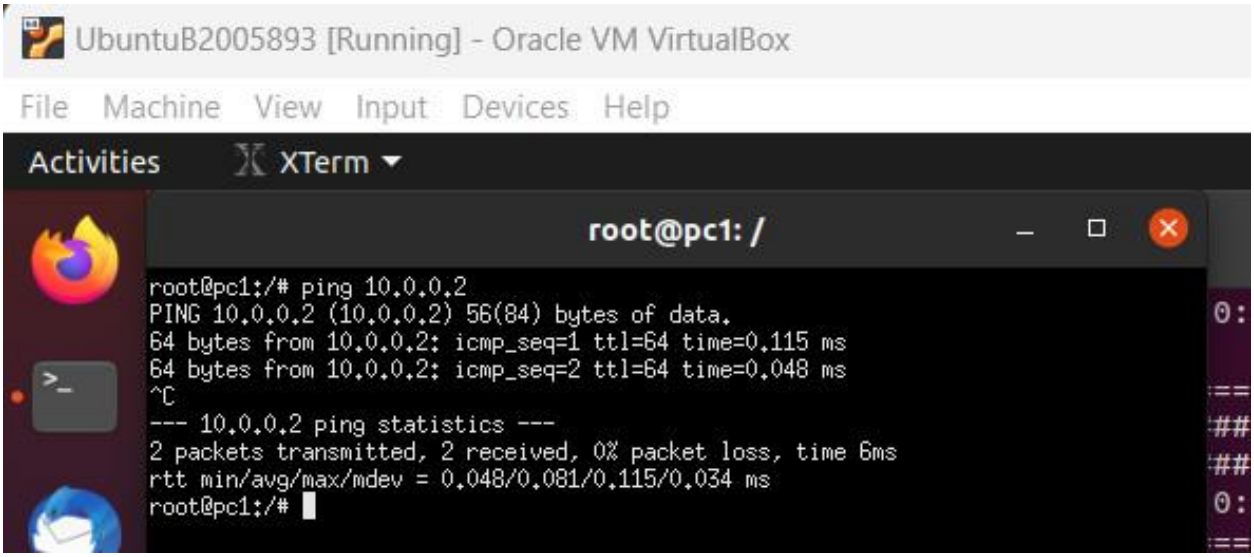
1. *Creating the VMs*
   $ sudo kathara vstart –eth 0:A –n pc1
   $ sudo kathara vstart –eth 0:A –n pc2



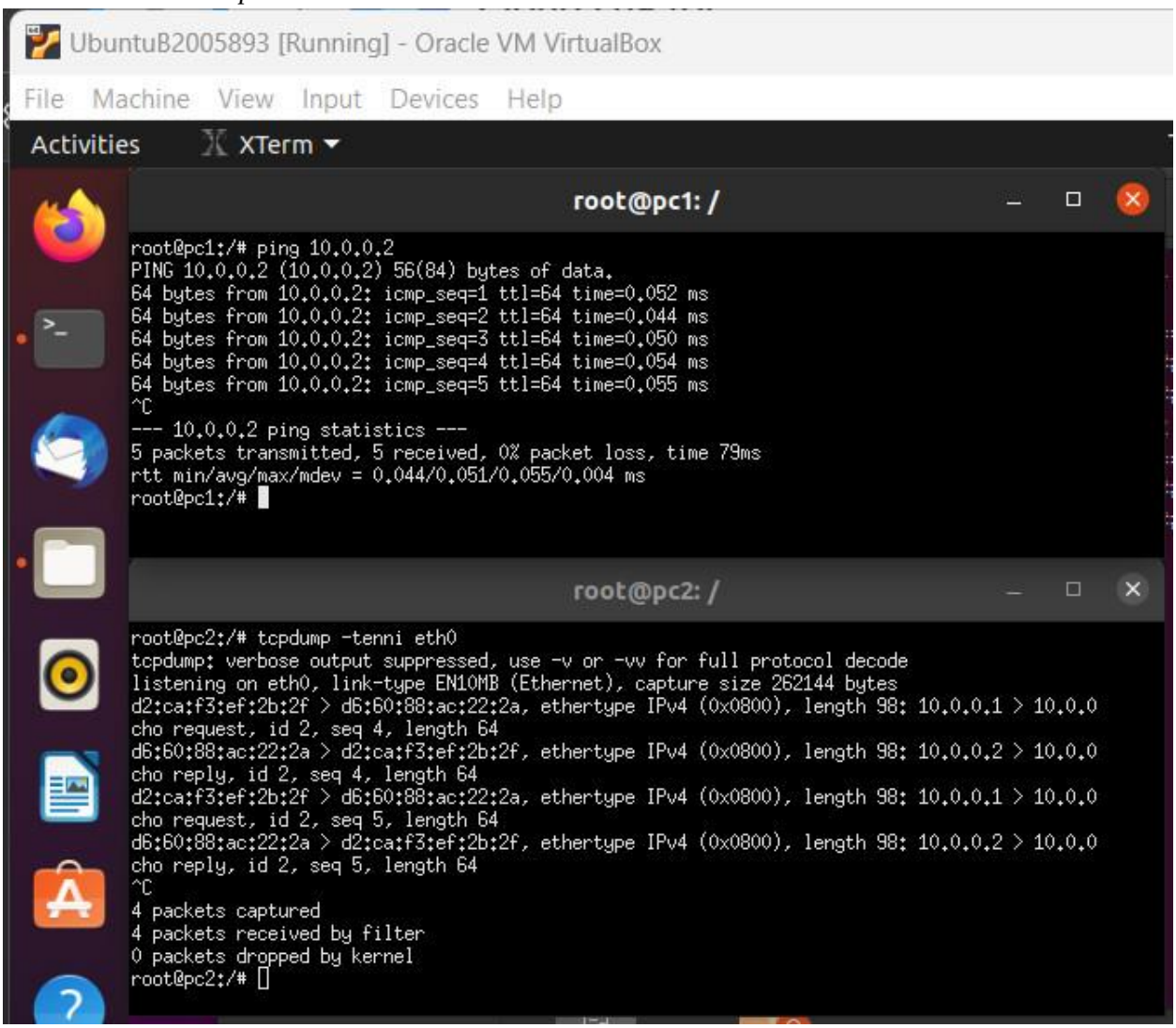
2. *Configuring network interfaces*
   From pc1: #ifconfig eth0 10.0.0.1 netmask 255.255.255.0 broadcast 10.0.0.255 up
   From pc2: #ifconfig eth0 10.0.0.2 netmask 255.255.255.0 broadcast 10.0.0.255 up



3. *Ping*
   From pc1: # ping 10.0.0.2

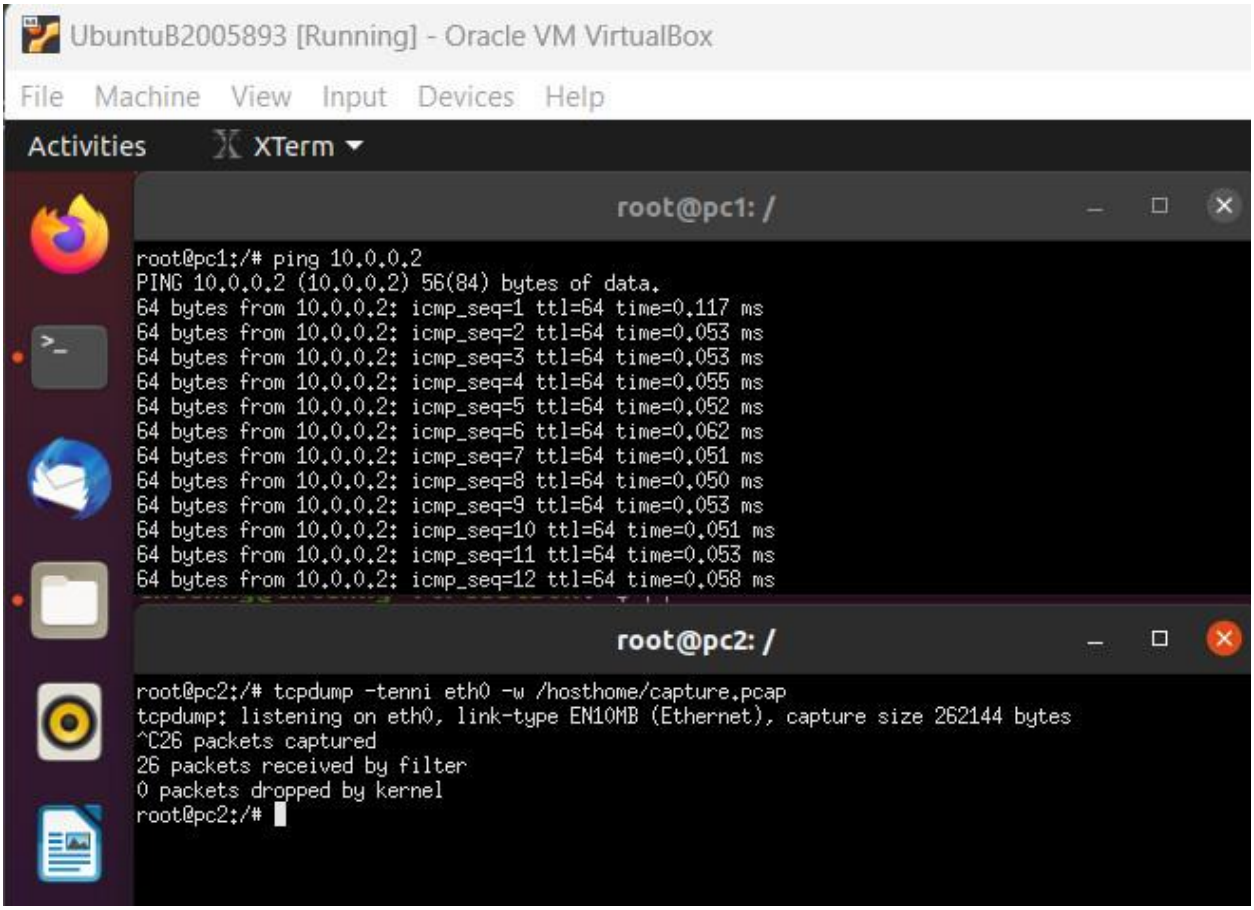


4. *A look at the packets*

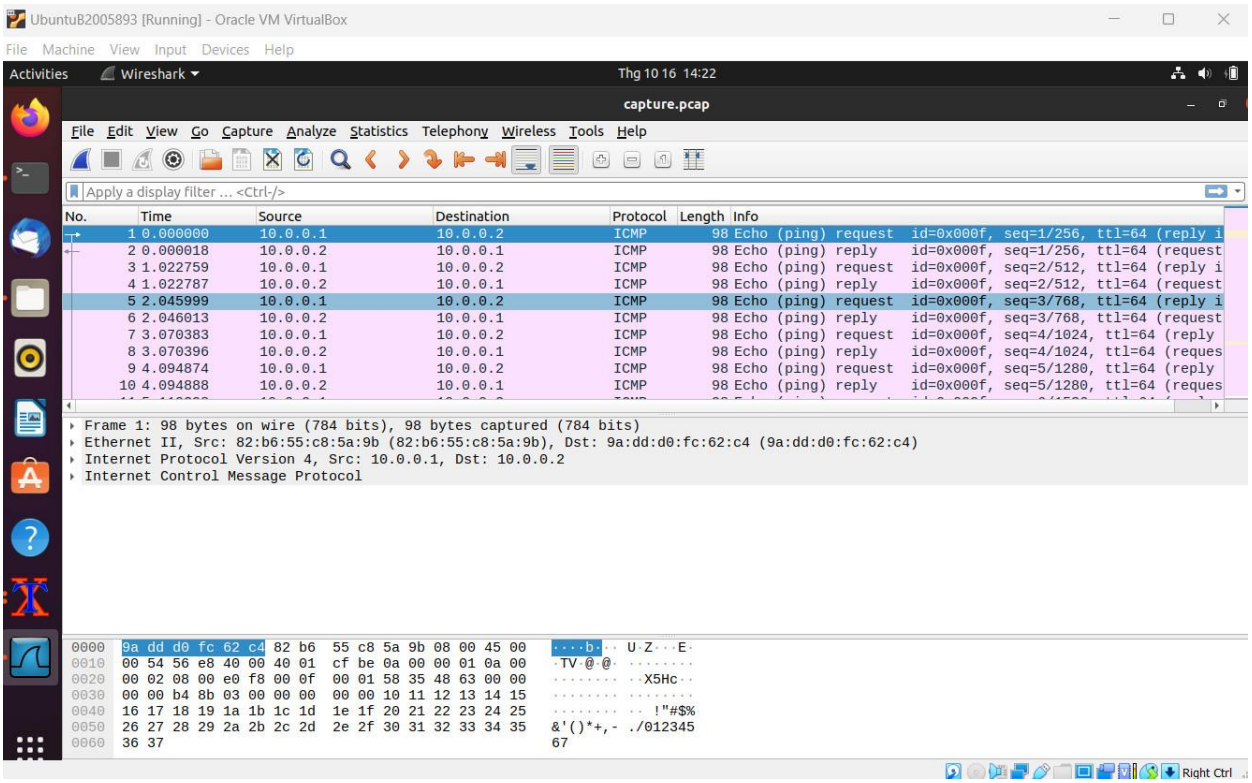


Ping from pc1. at the same time, sniff from pc2 (ctrl+C to interrupt)
On pc1: # ping 10.0.0.2
On pc2: # tcpdump –tenni eth0

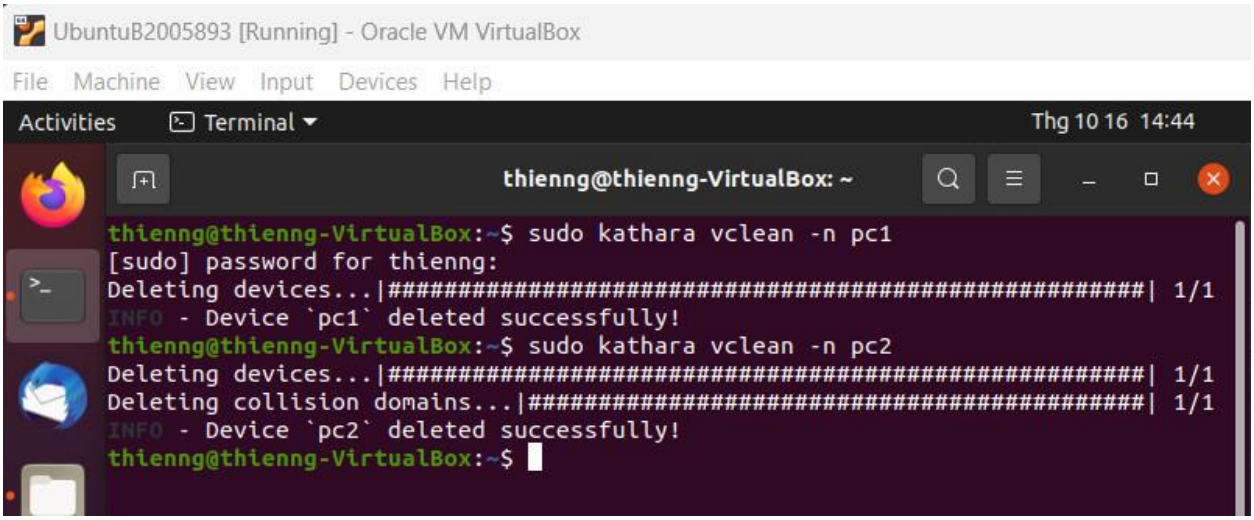
5. *Looking at the packets with a graphical interface*
   Same as (4), but store sniffed packets into file capture.pcap (on the host machine)
   On pc1: # ping 10.0.0.2
   On pc2: # tcpdump –tenni eth0 –w /hosthome/capture.pcap



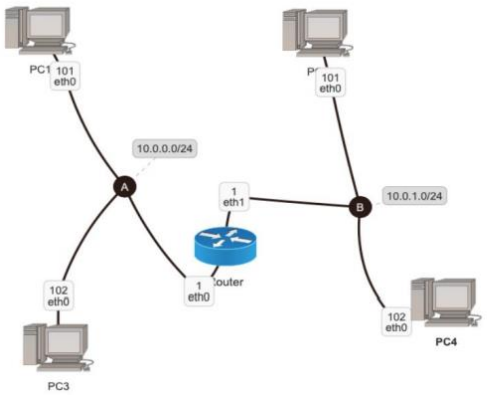
6. *Open capture.pcap on real host machine using wireshark*



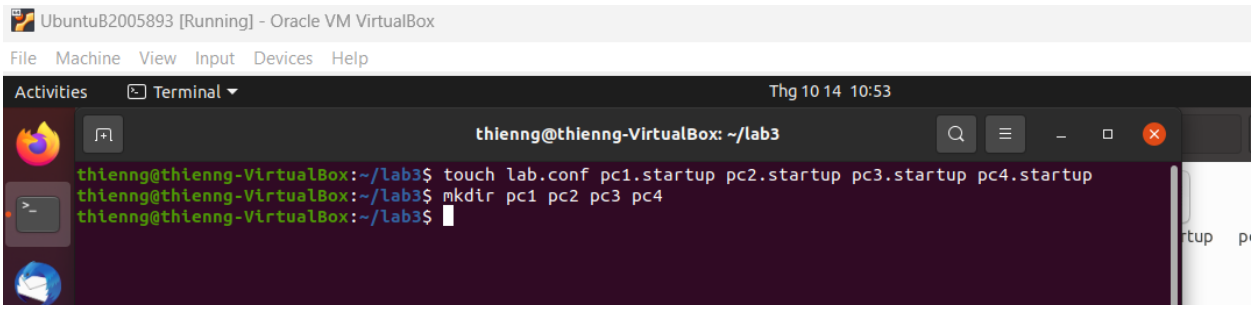
7. *Delete the VMs*



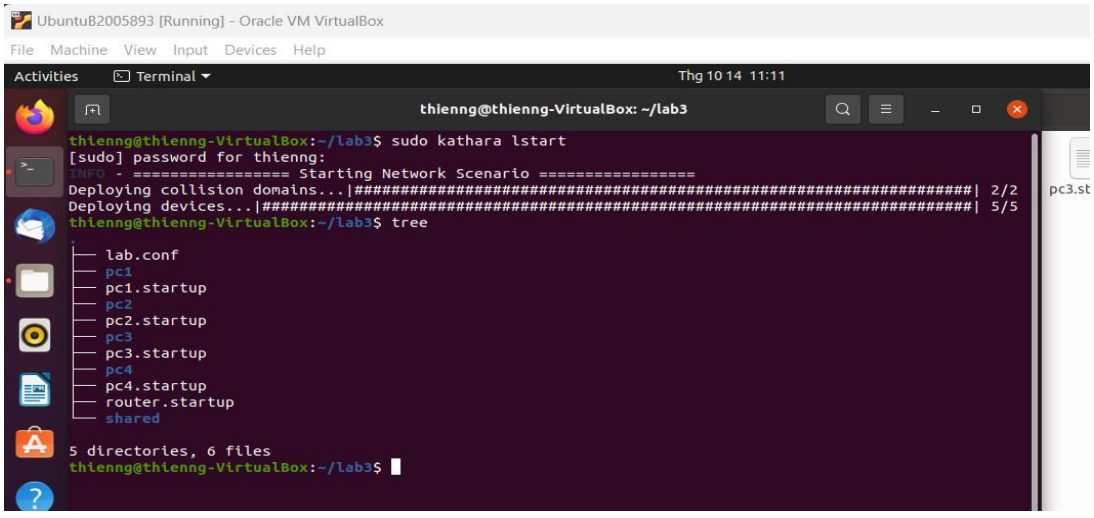
Exercise 2: Construct the following network



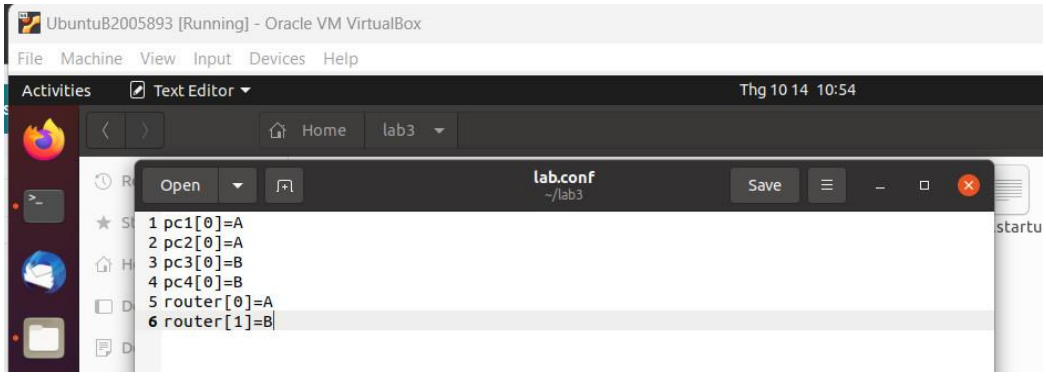
1. *Create specific files and folders*
   $ touch lab.conf pc1.startup pc2.startup pc3.startup pc4.startup
   $ mkdir pc1 pc2 pc3 pc4

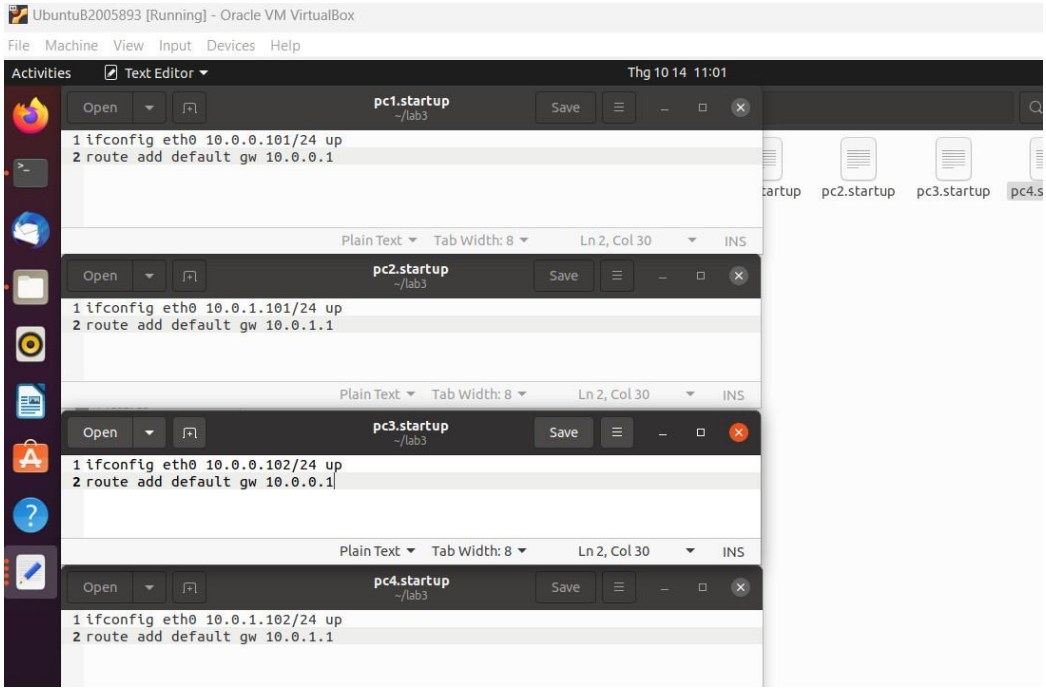


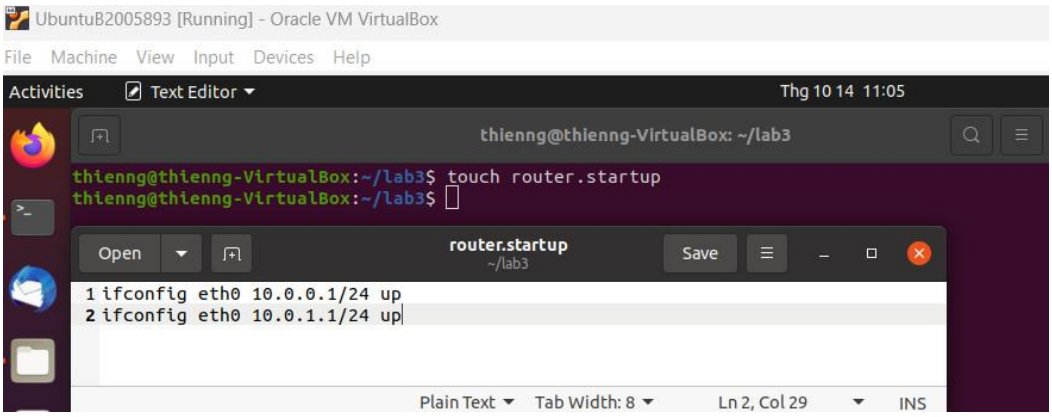
2. *Here is the tree*
   $ ls



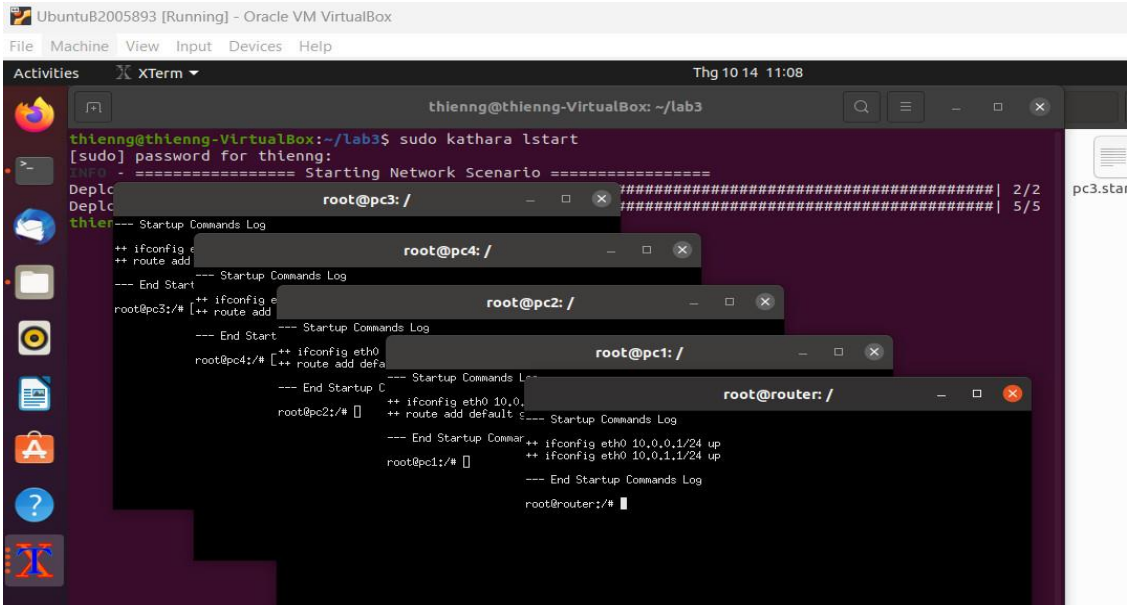
3. *Enter network configurations into files*



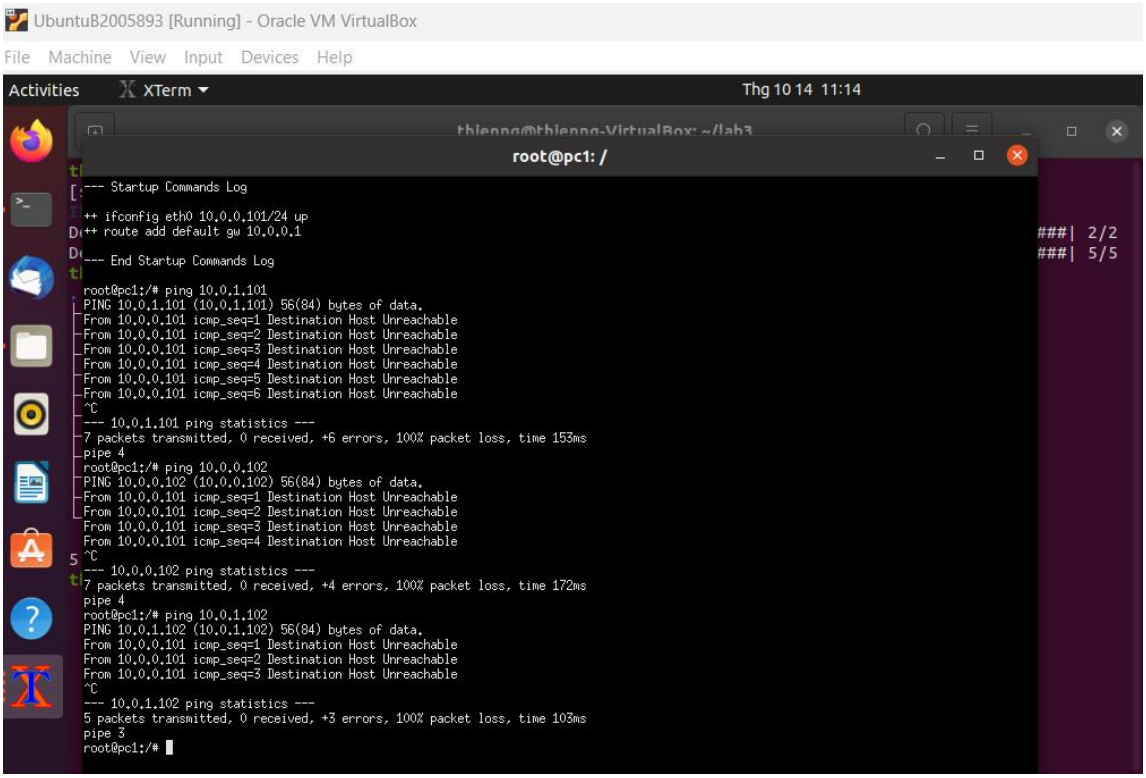




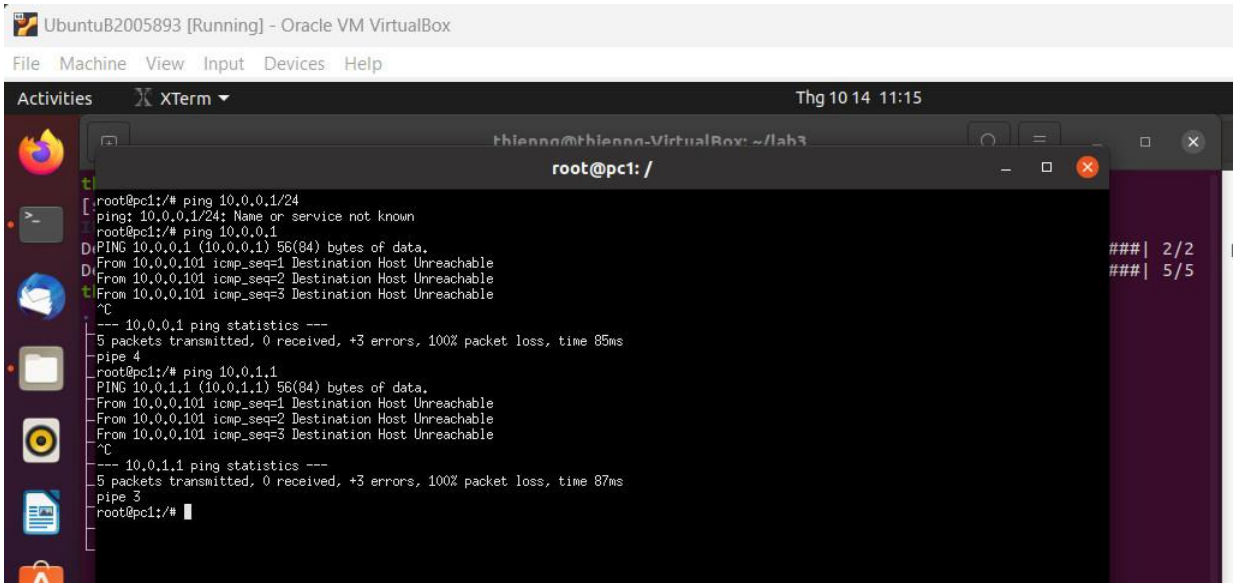
4. *Start kathara*
   $sudo kathara lstart



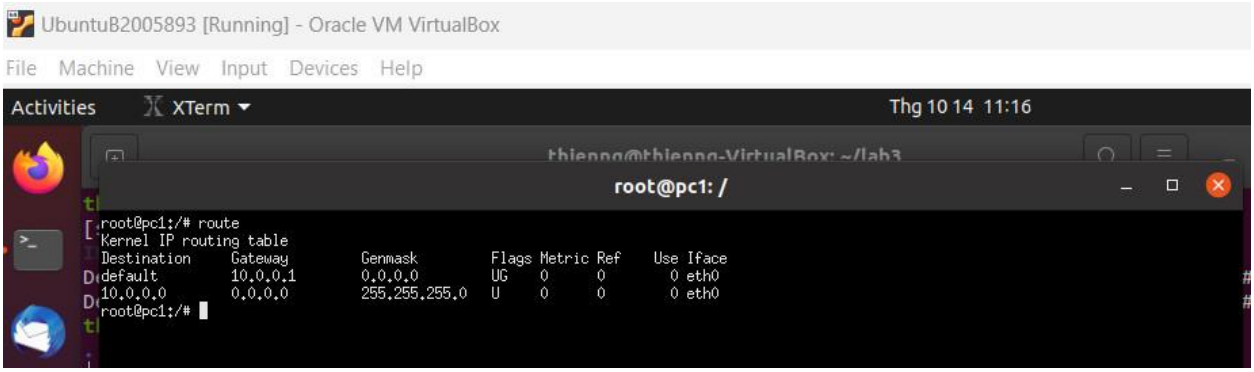
5. *Ping to other vms from pc1*
   # ping 10.0.1.101
   # ping 10.0.0.102
   # ping 10.0.1.102



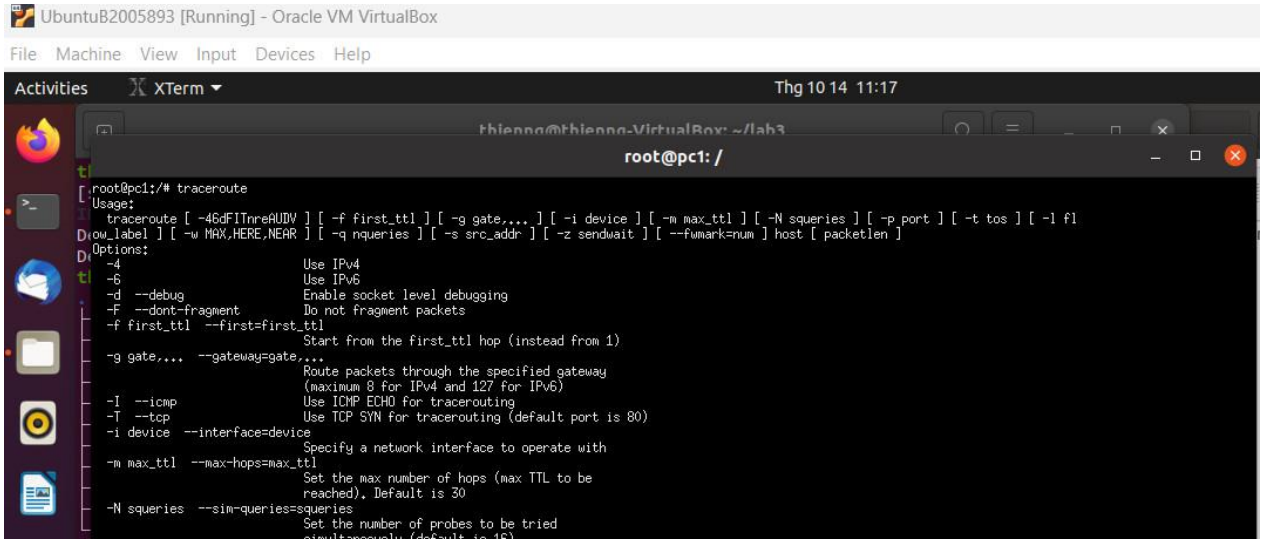
6. *Ping to 2 interfaces of the router*
   # ping 10.0.0.1
   # ping 10.0.1.1



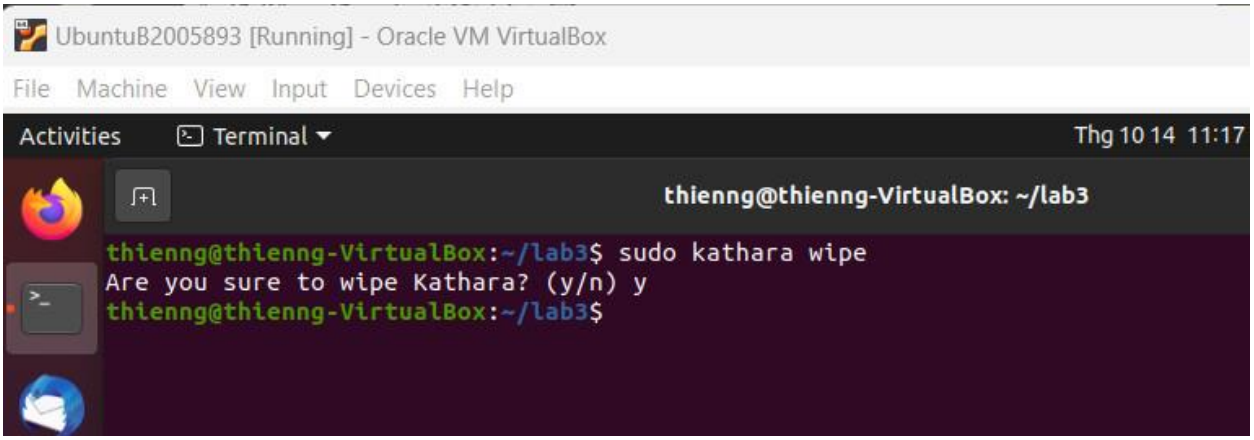
7. *Route pc1*
   # route



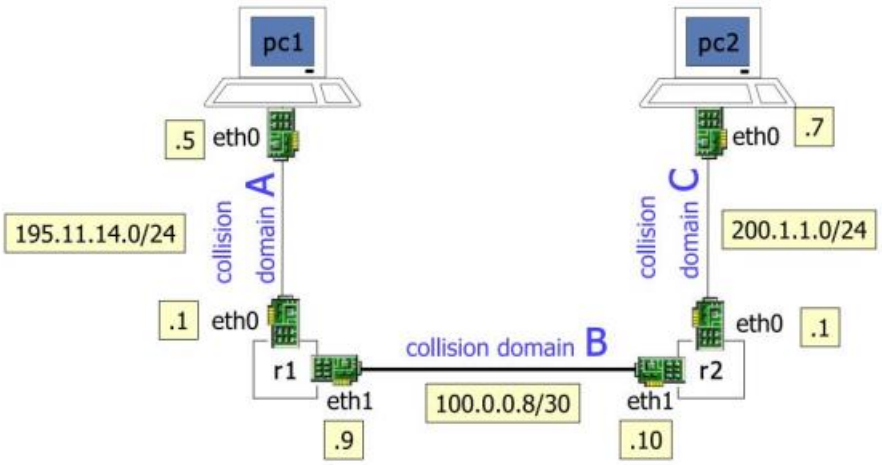
8. *Traceroute pc1*
   #traceroute



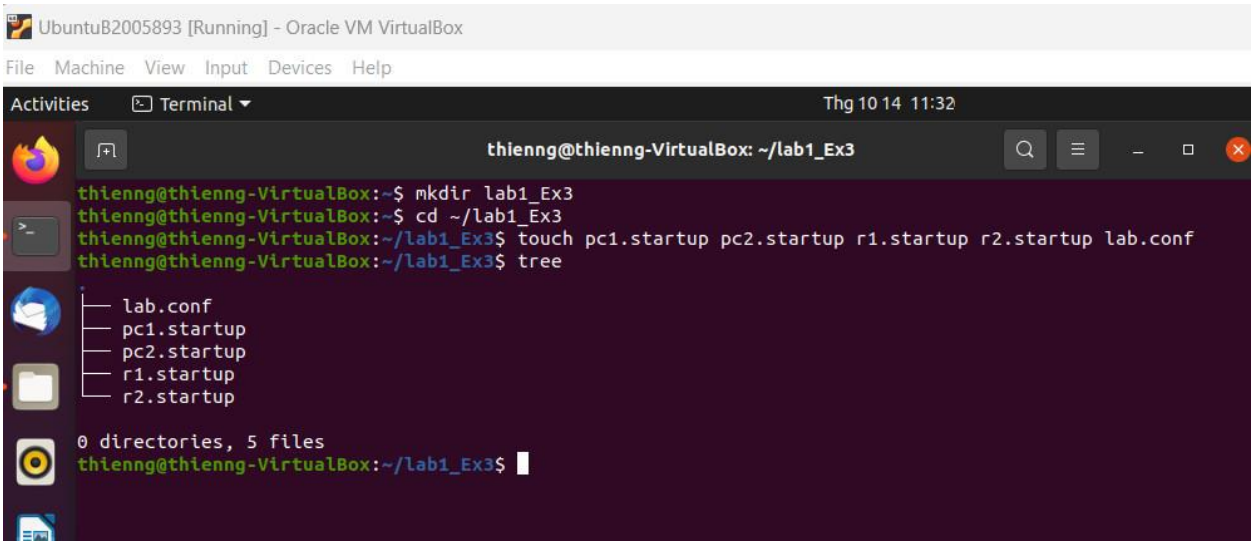
9. *Delete all vms*
   $ sudo kathara wipe



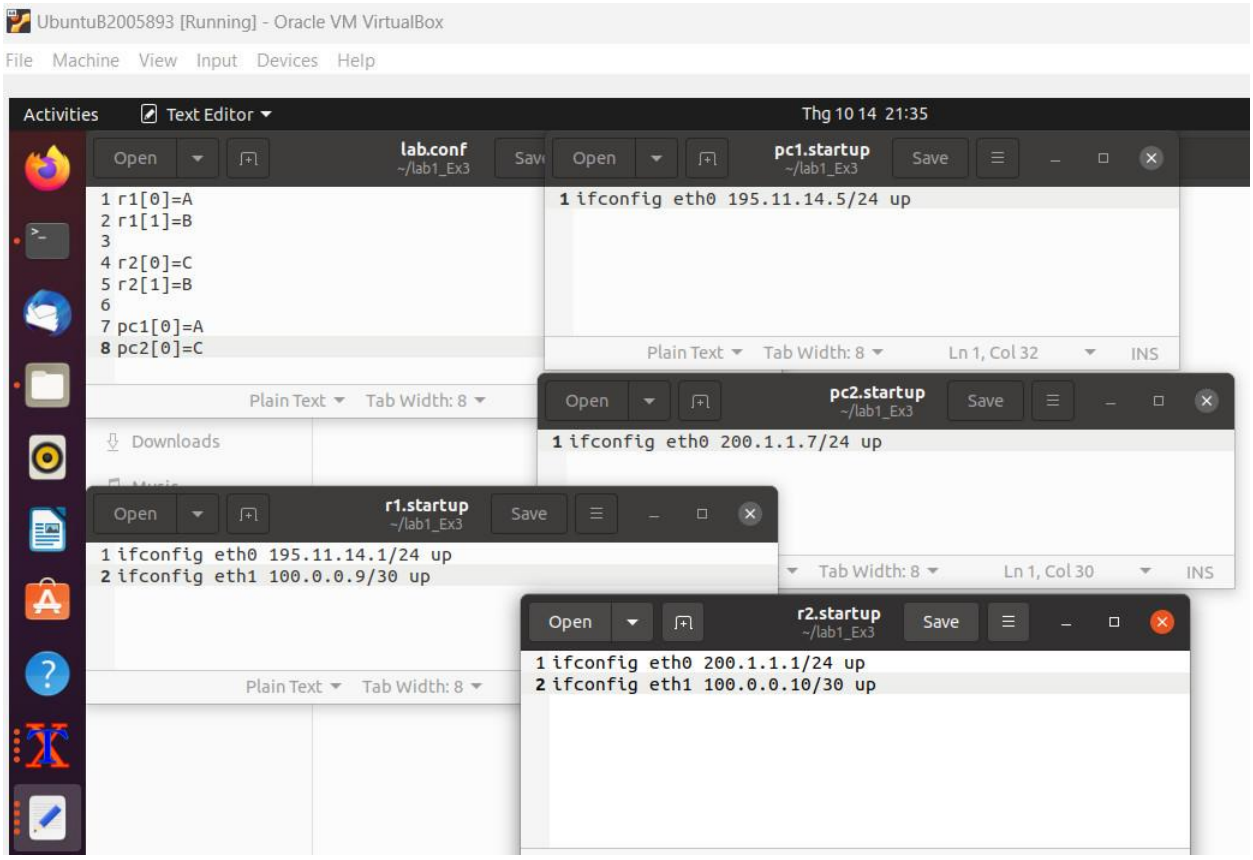
Exercise 3: Construct the following network



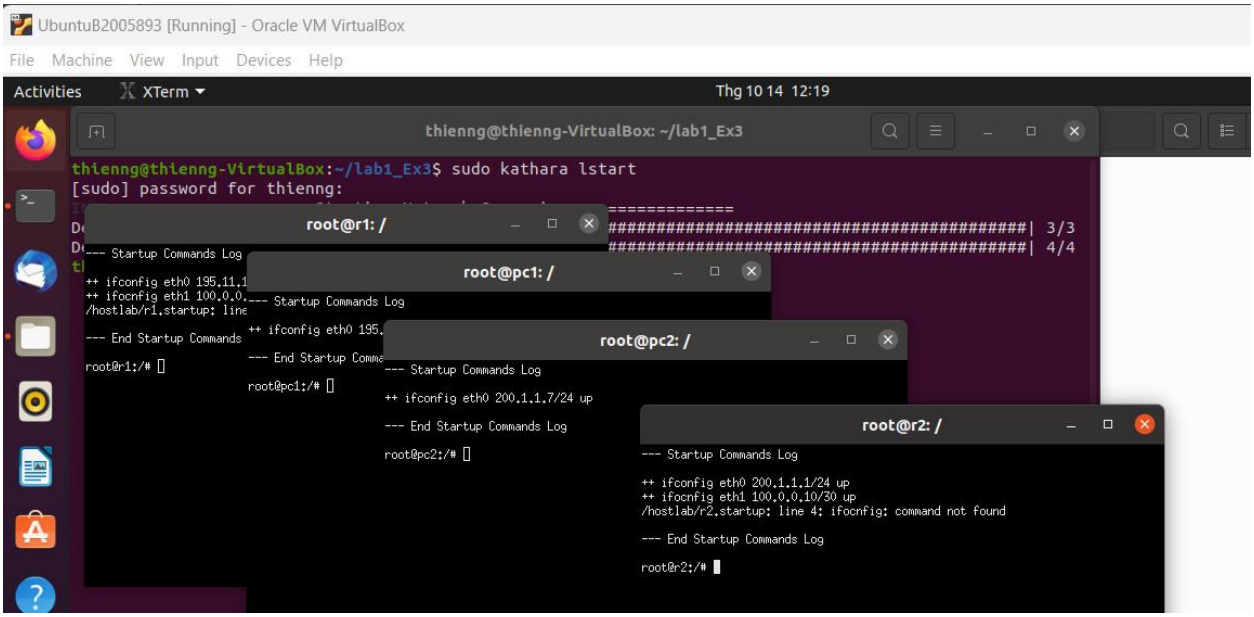
1. *Create lab directory hierarchy*
   $ mkdir lab1_Ex3
   $ cd ~/lab1_Ex3
   $ touch pc1.startup pc2.startup r1.startup r2.startup lab.conf
   $ tree



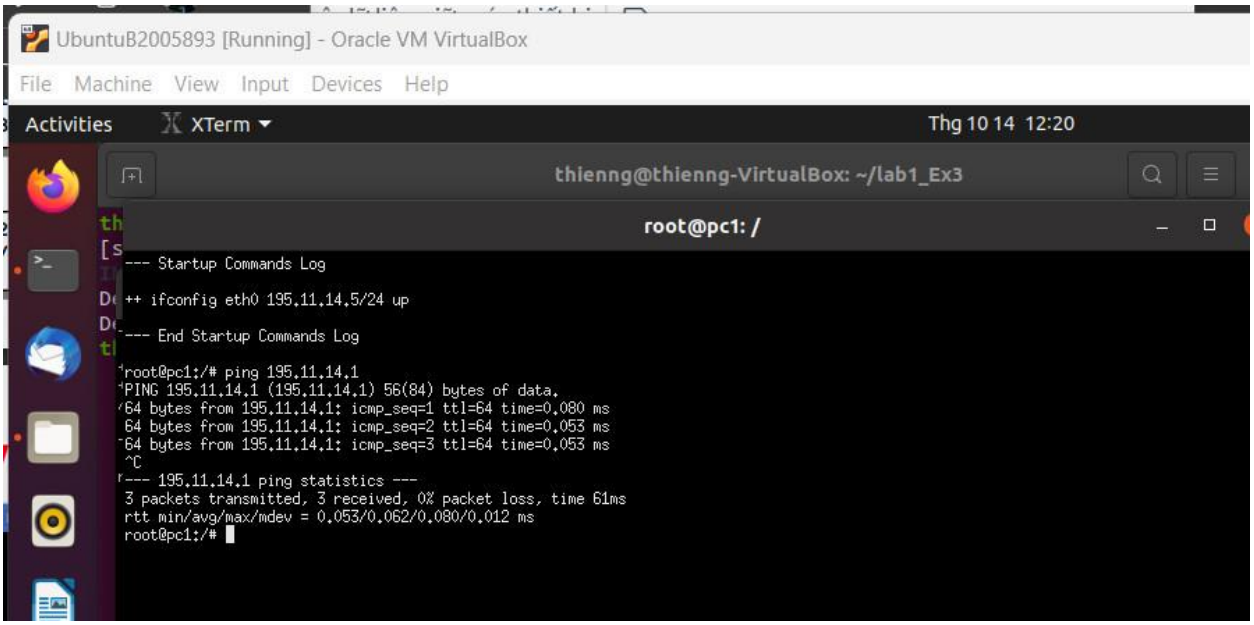
2. *Enter network configuration into files*



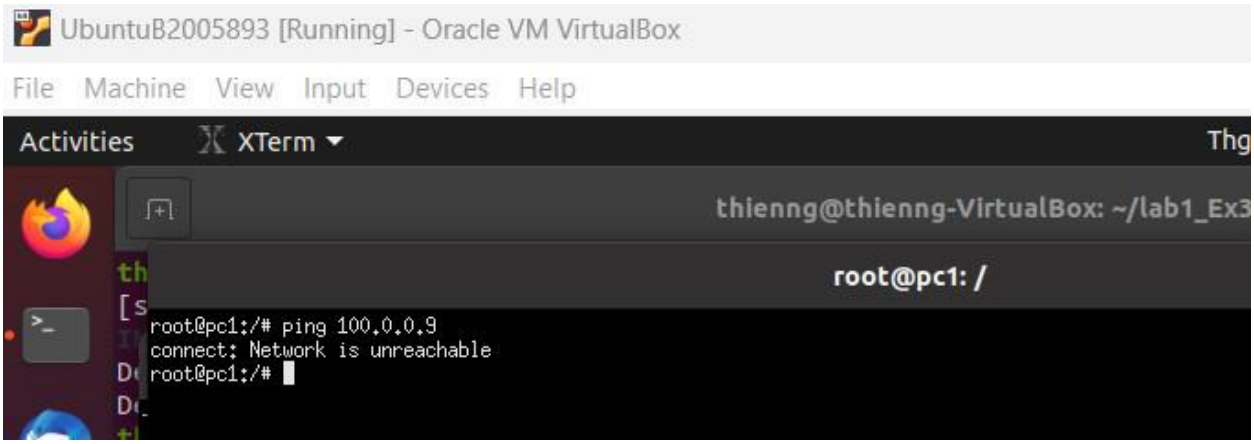
3. *Start kathara*
   $ sudo kathara lstart



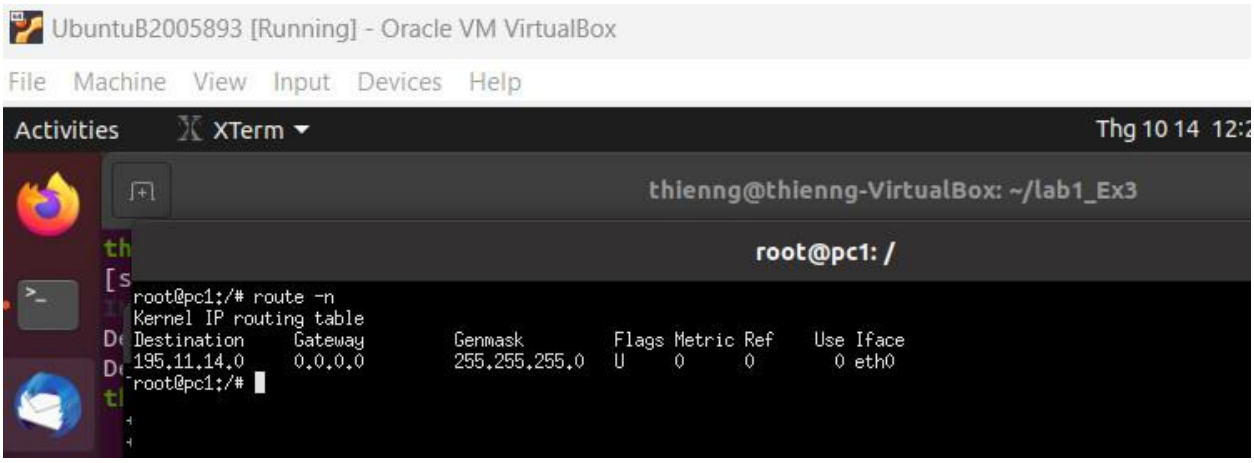
4. *Test connectivity (1) – from pc1*
   # ping 195.11.14.1



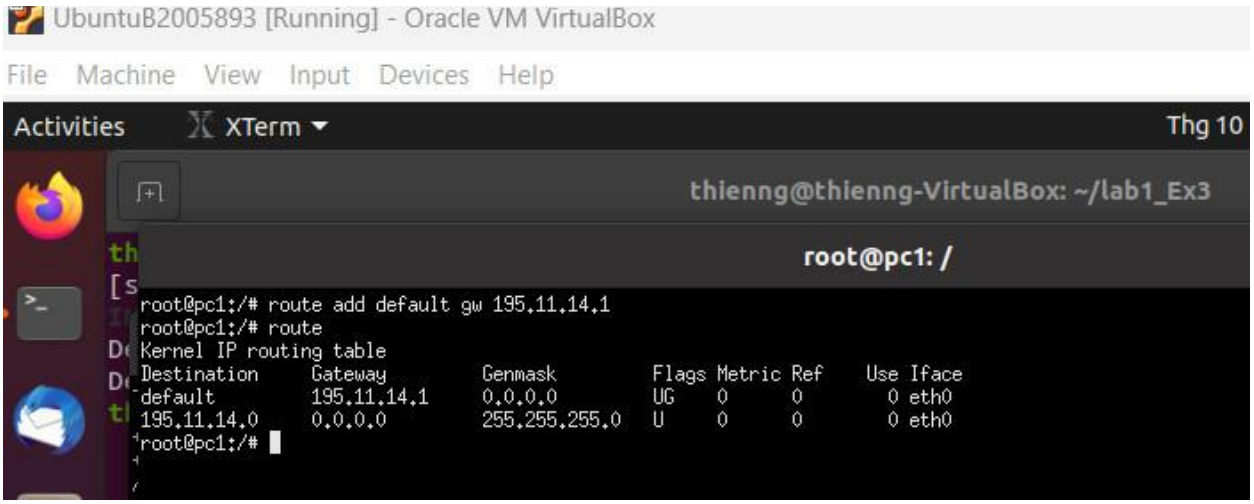
5. *Test connectivity (2) – from pc1 to r1*
   $ ping 100.0.0.9
   It is unreachable because the specified domain either does not exist or could not be contacted.



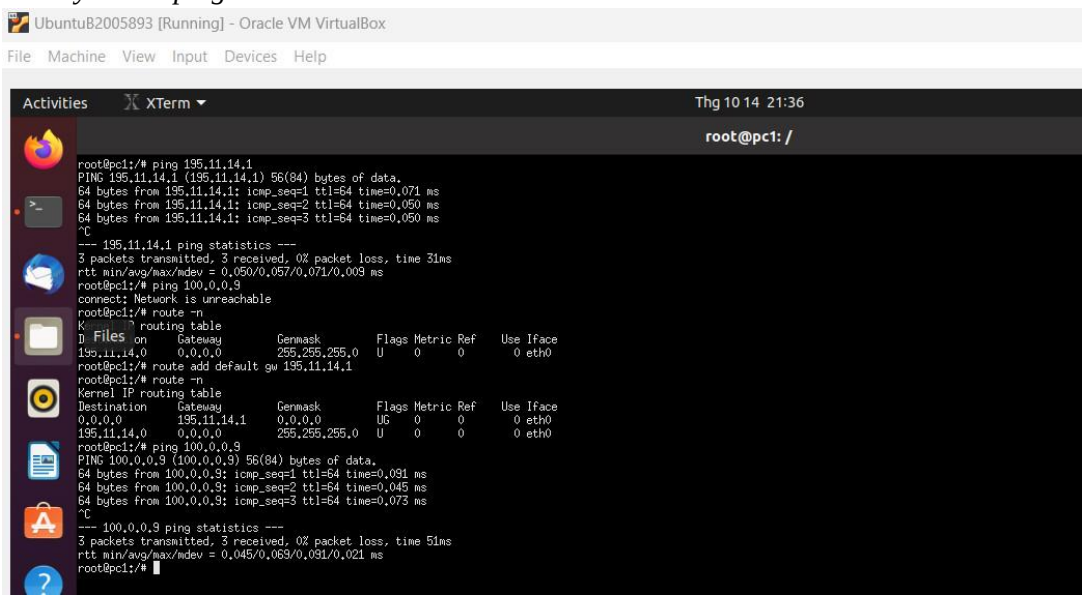
6. *Solution to make pc1 can reach to r1*
   a. *Inspecting routing table*
      From pc1:
      # route –n



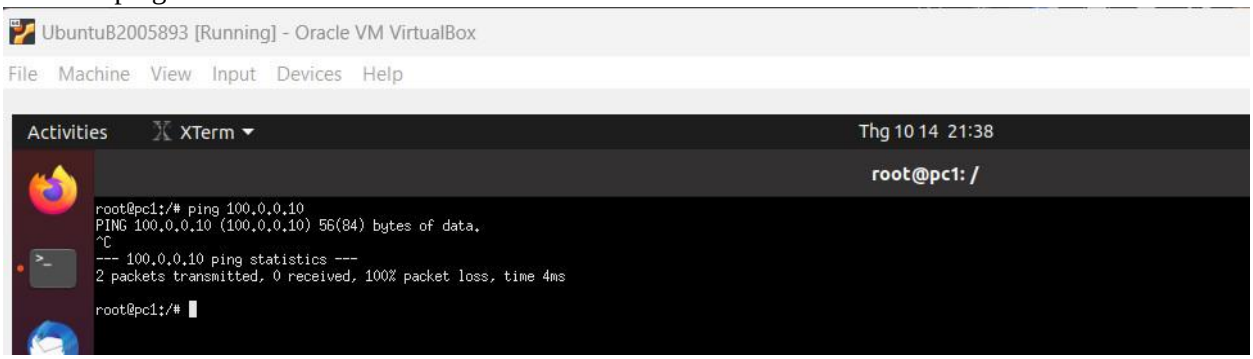
   b. *Defaute routes on PC1*
      # route add default gw 195.11.14.1
      # route



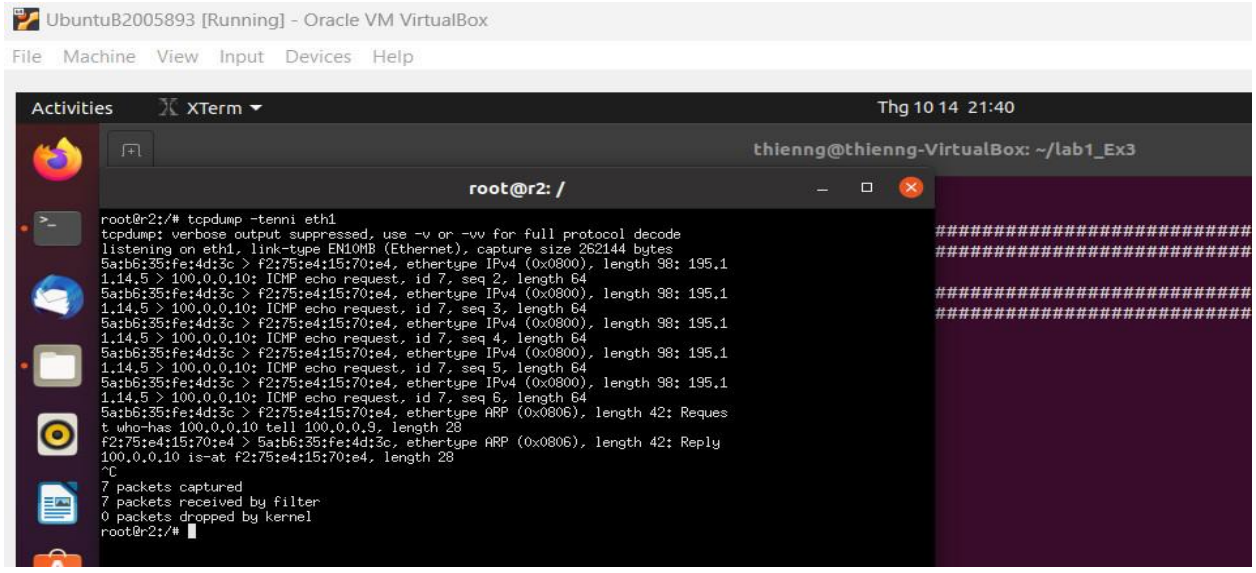
7. *Now you can ping with 100.0.0.9*



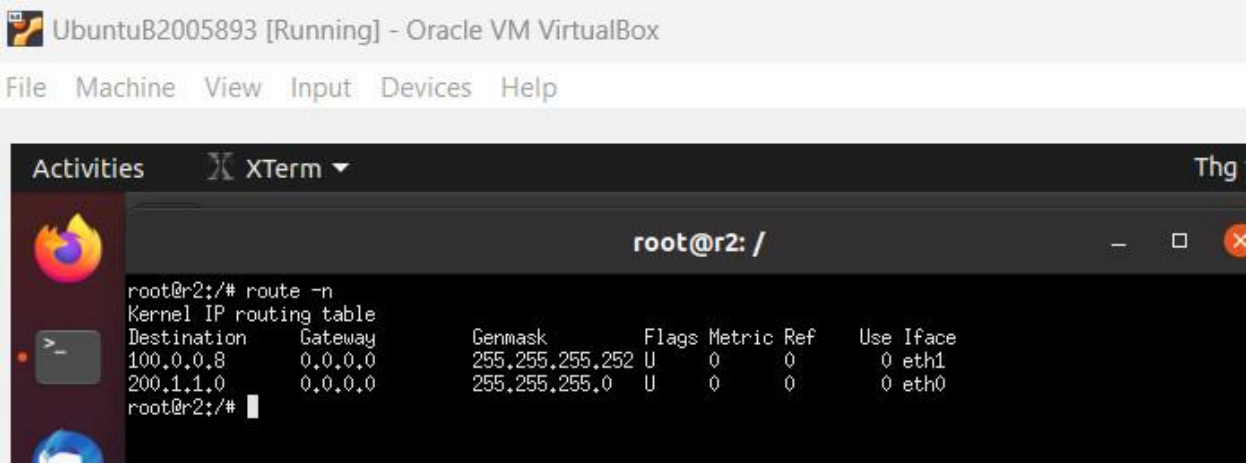
8. *Ping 100.0.0.10 is not reach*
#ping 100.0.0.10



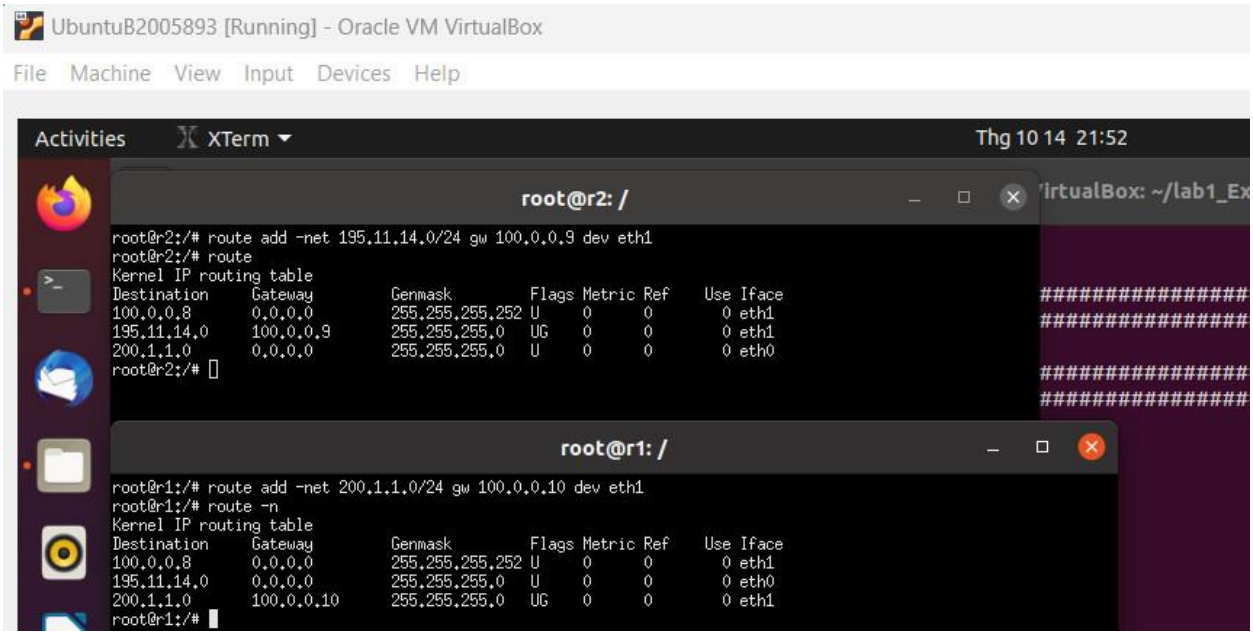
9. *While pinging from pc1 100.0.0.10 sniff on interface eth1 of r2*
From r2: # tcpdump –tenni eth1



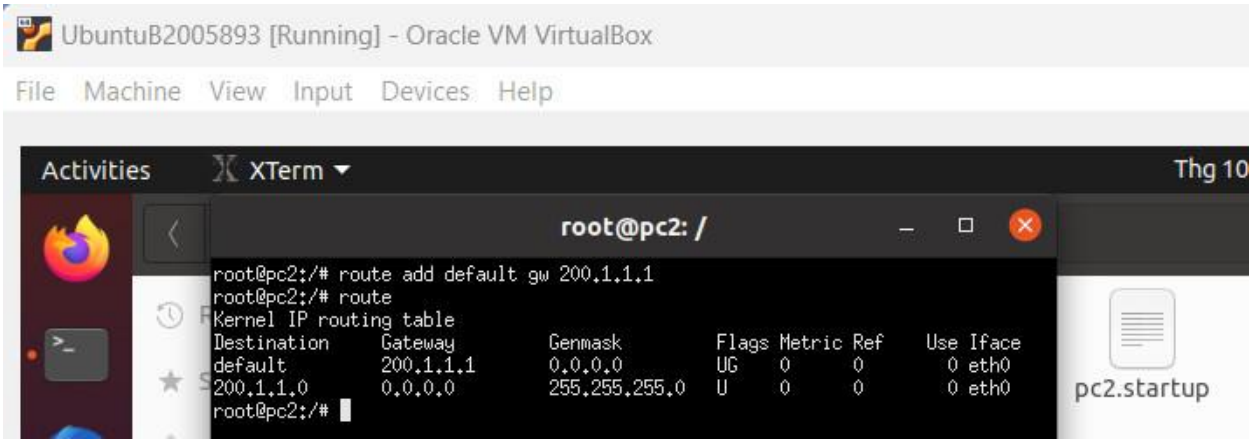
10. *r2's routing table*
    # route -n



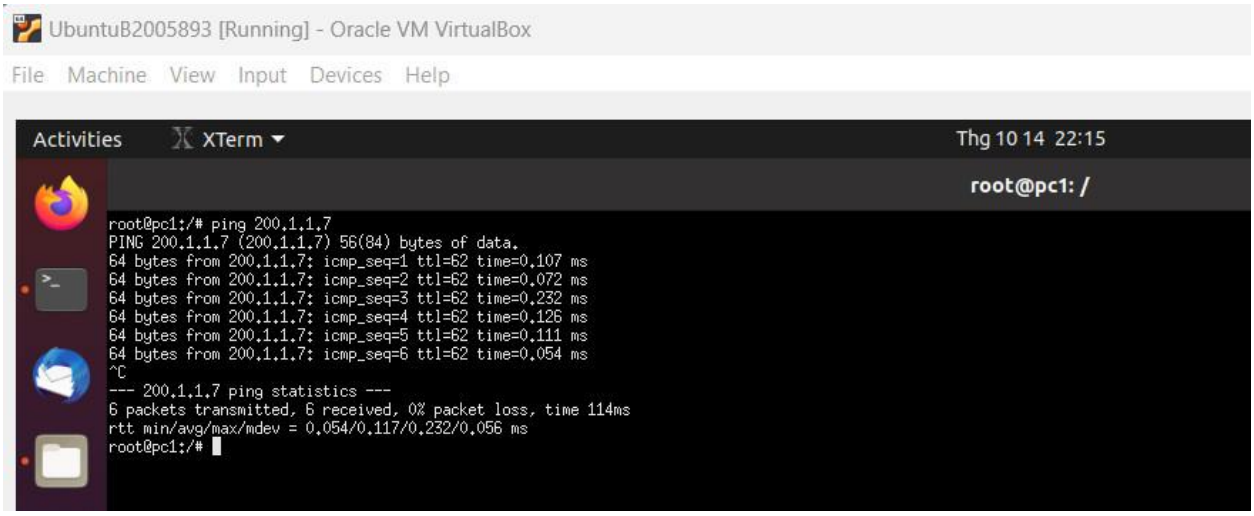
11. *Configuring the stactic route of r1 and r2*
    a. *From r1:*
       # route add –net 200.1.1.0/24 gw 100.0.0.10 dev eth1
       # route –n
    b. *From r2:*
       #route add –net 195.11.14.0/24 gw 100.0.0.9 dev eth1
       #route -n



12. *Set default of pc2*
    # route add default gw 200.1.1.1

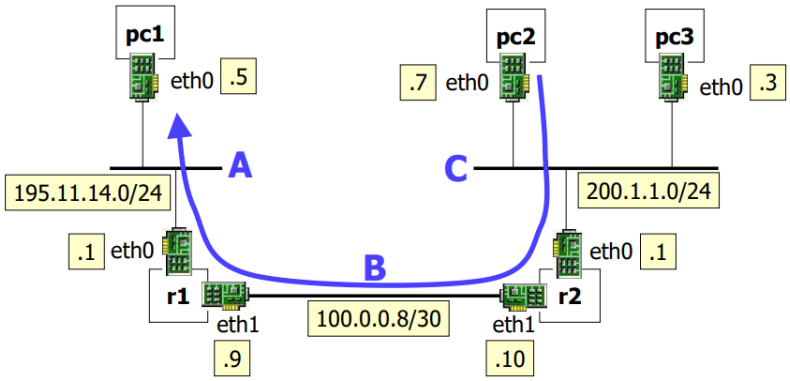


13. *Test static route*
    The Pcs can now peach each others
    From pc1:
    # ping 200.1.17



14. *Delete all vms with $ sudo kathara wipe*

Exercise 4: Study arp protocol



1. *Lab confiruation*

   $ mkdir pc1 pc2 pc3 r1 r2

   $ touch lab.conf pc1.startup pc2.startup r1.startup r2.startup

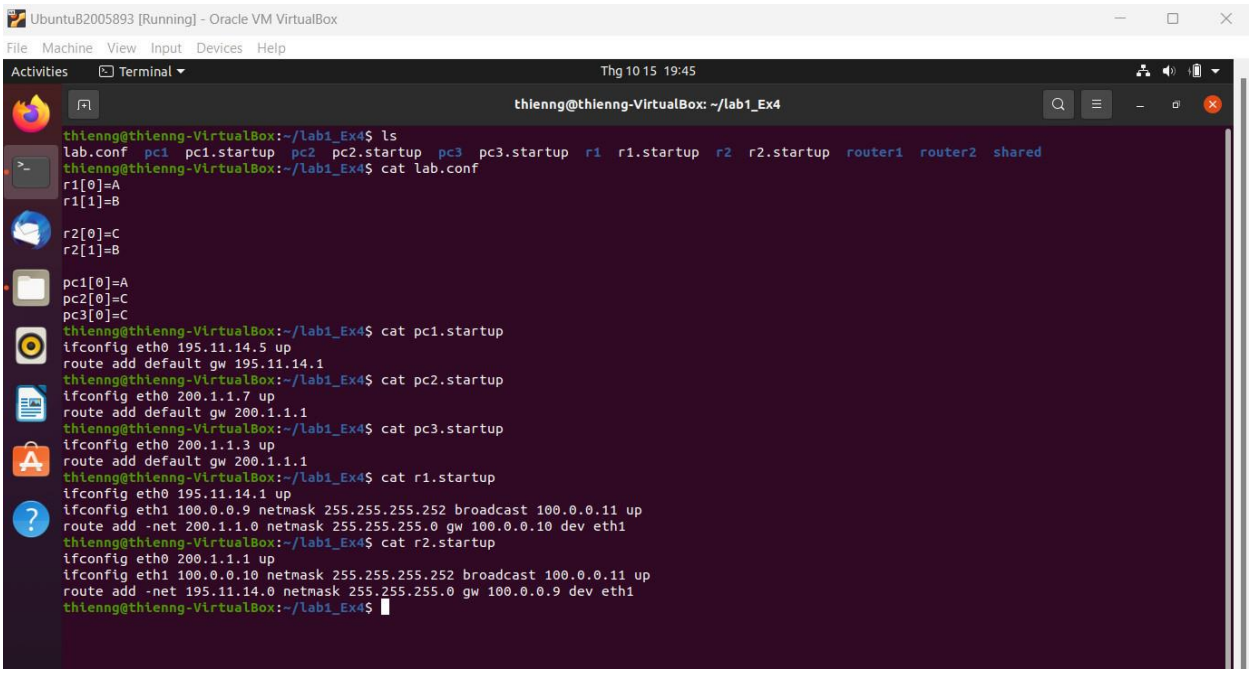
   $ cat pc1.startup

   $ cat pc2.startup

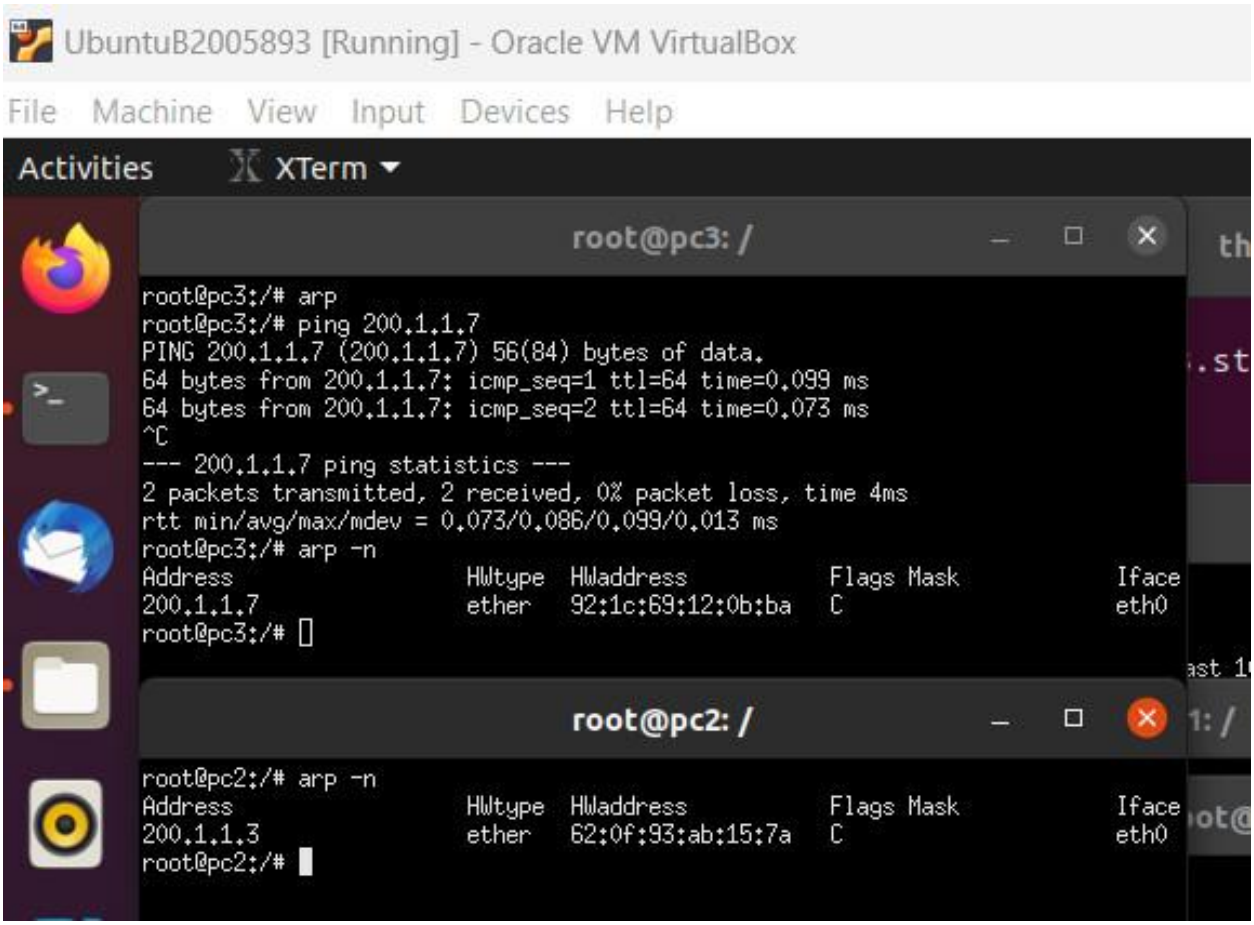
   $ cat pc3.startup

   $ cat r1.startup

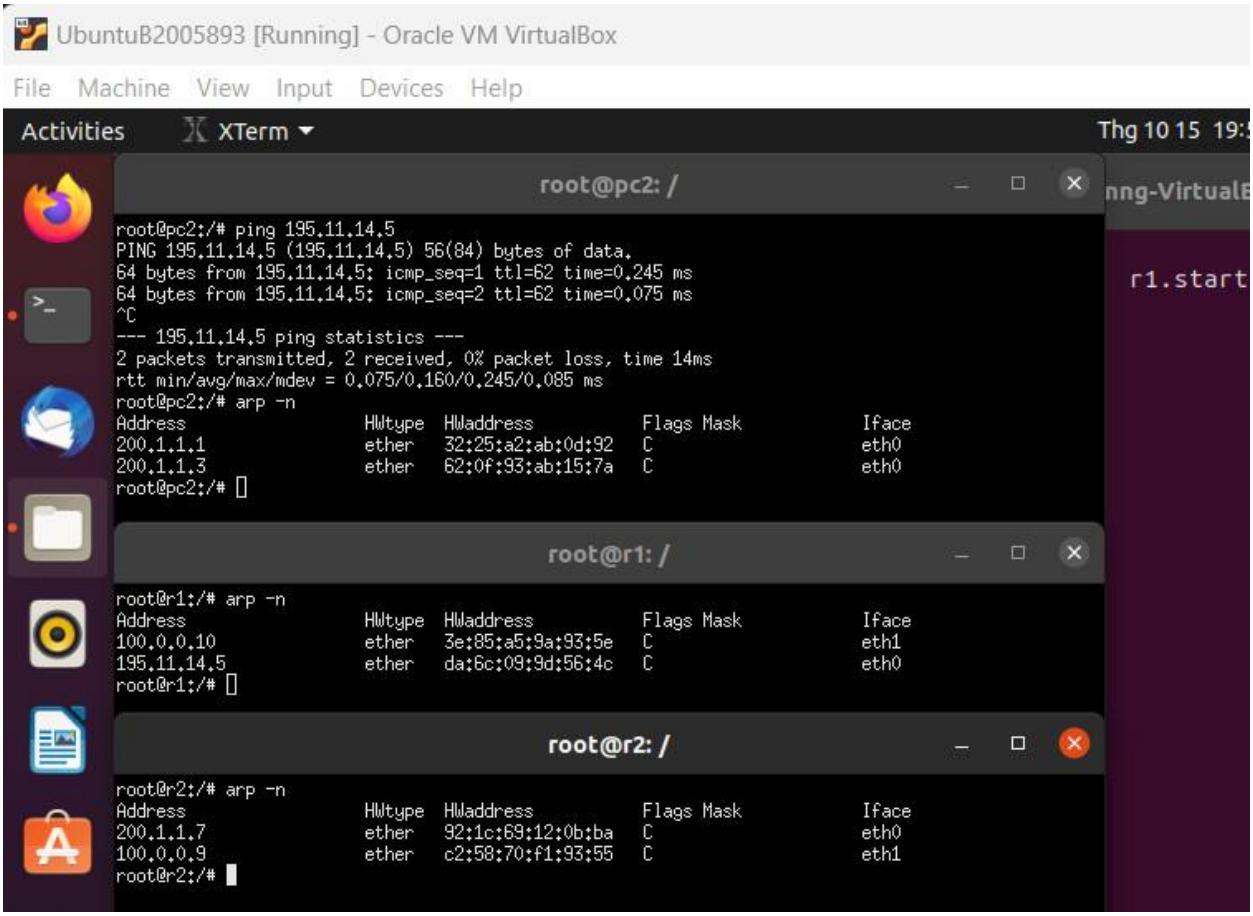
   $ cat r2.startup



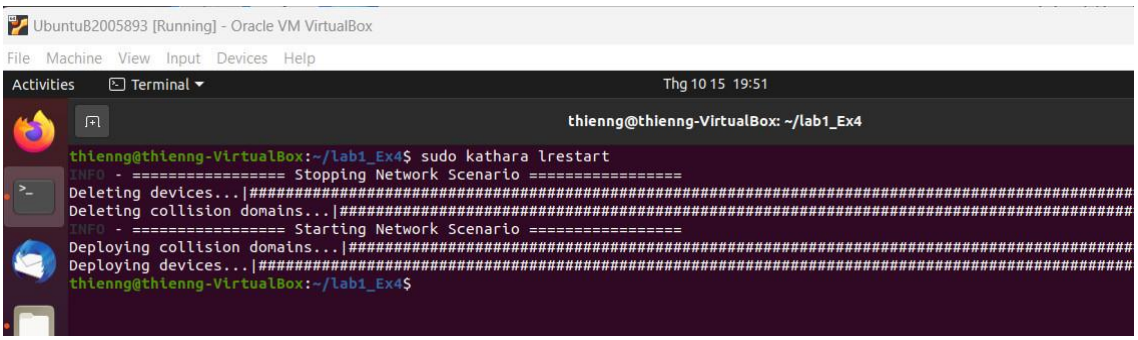
2. *Start kathara and inspecting the arp cache (LOCAL TRAFFIC)*
   a. *<u>From pc3:</u>*
      # arp
      # ping 200.1.1.7
      # arp –n
   b. *<u>From pc2:</u>*
      # arp –n

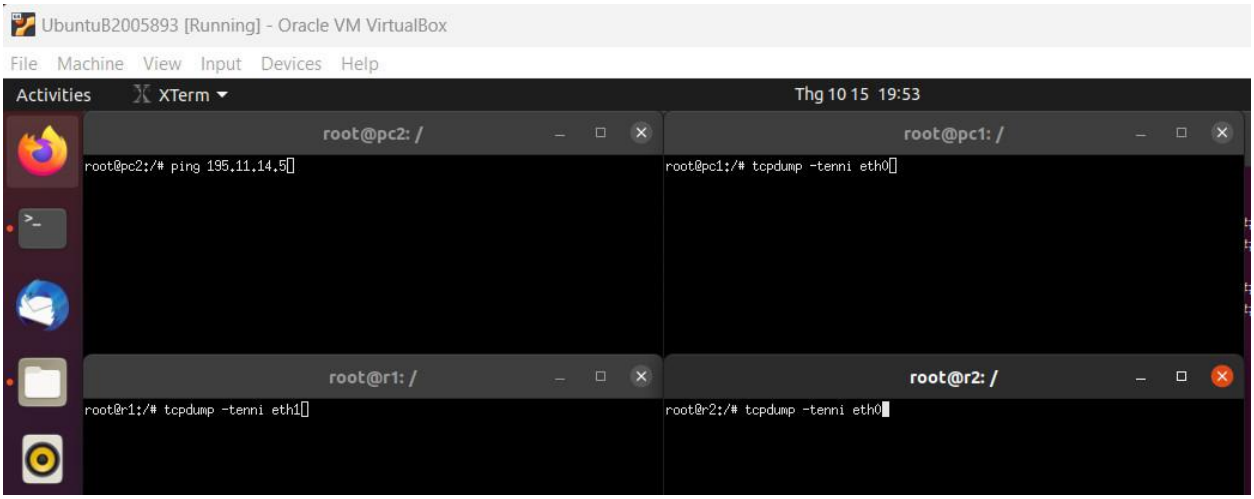


3. *Inspecting the arp cache (NON LOCAL TRAFFIC)*
   a. *From pc2:*
      # ping 195.11.14.5
      # arp –n
   b. *From r1:*
      # arp –n
   c. *From r2:*
      # arp -n

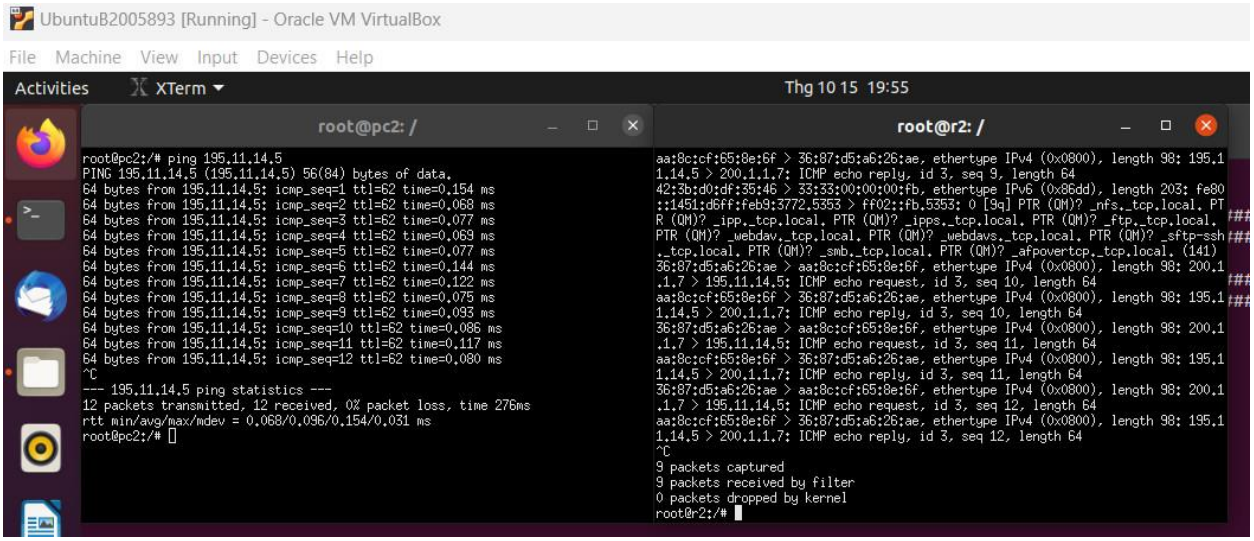


4. *Restart the lab and prepare to sniff*
   $ sudo kathara restart





5. *Sniffing arp traffic*
   a. *<u>On collsion domain C</u>*
      While pinging from pc2 195.11.14.5 sniff on interface eth0 of r2
      From pc2: # ping 195.11.14.5
      From r2: # tcpdump –tenni eth0

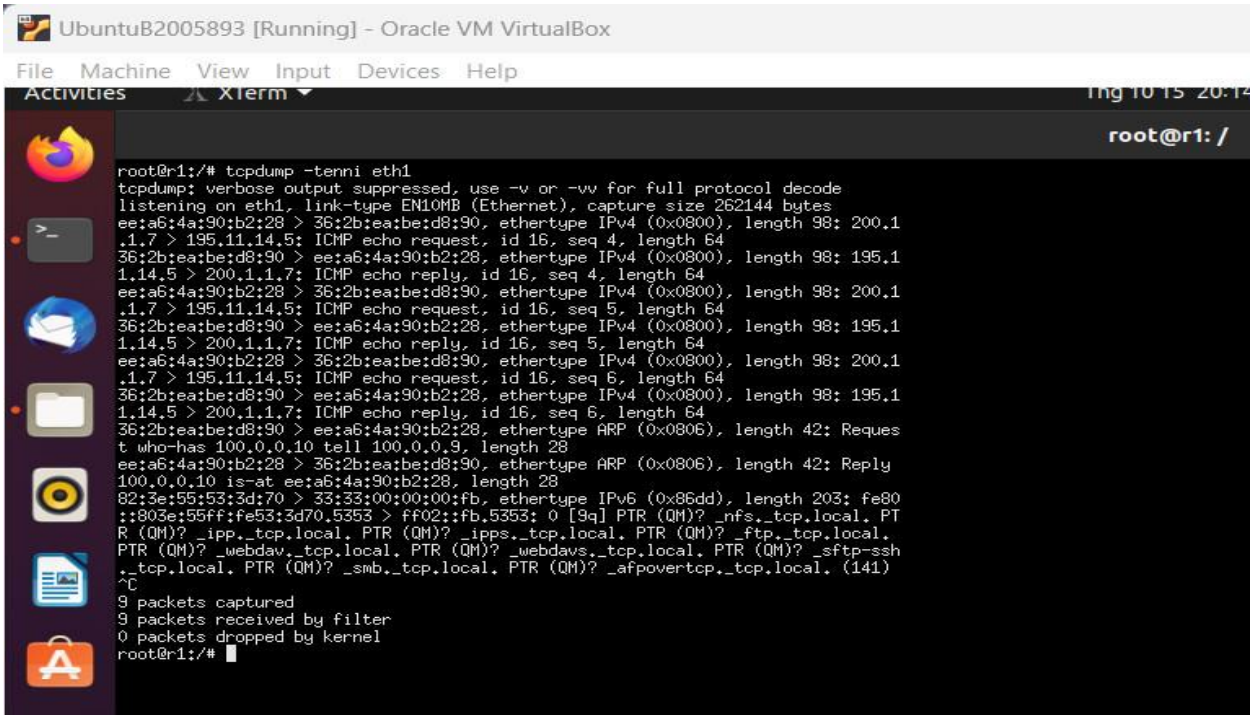


b. *On collision domain B*

While pinging from pc2 195.11.14.5 sniff on interface eth1 of r1

From pc2: # ping 195.11.14.5

From r1: # tcpdump –tenni eth1

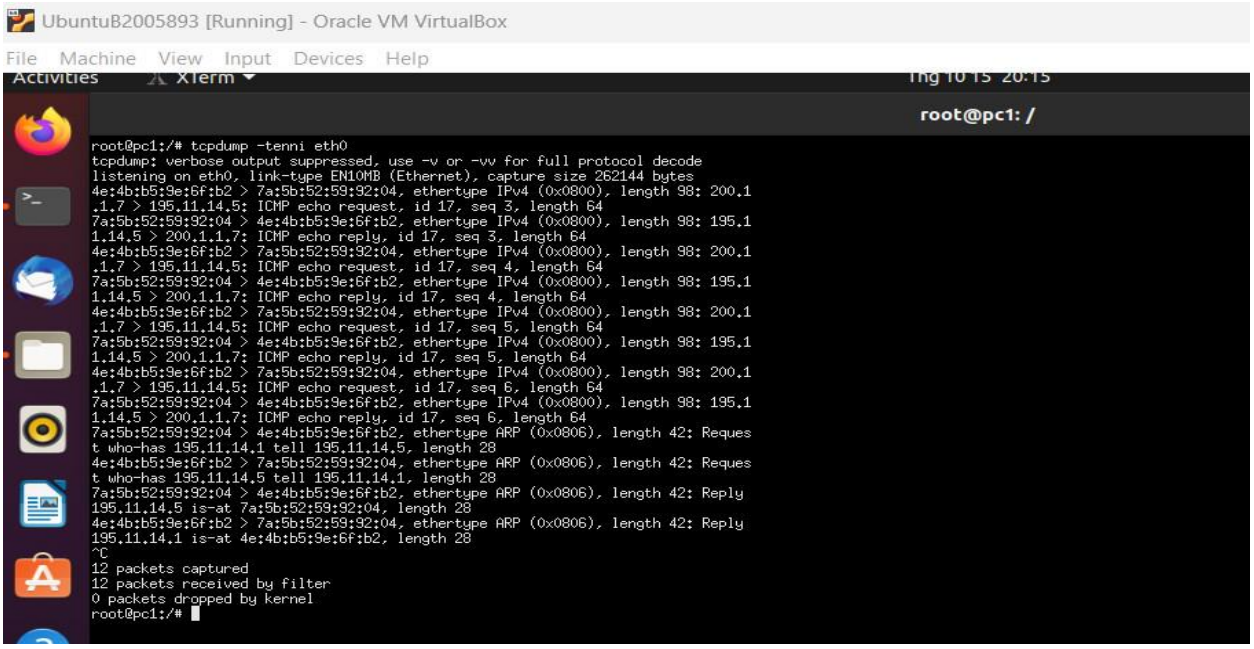


c. *On collision domain A*

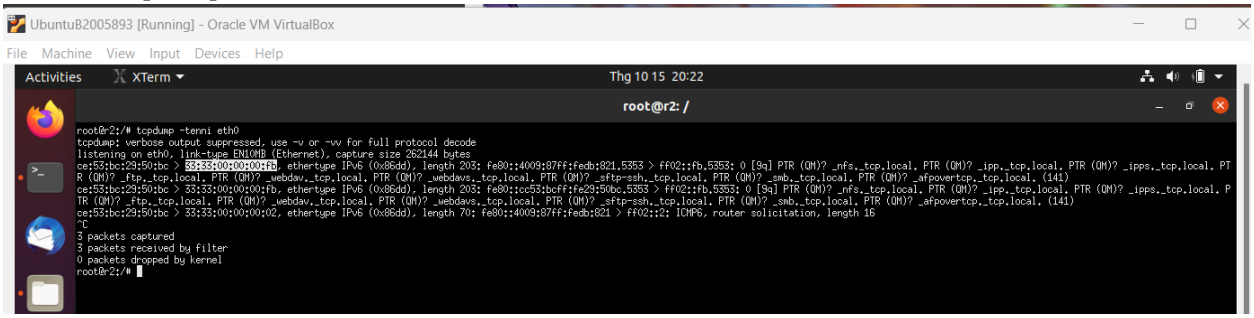
While pinging from pc2 195.11.14.5 sniff on interface eth0 of pc1

From pc2: # ping 195.11.14.5

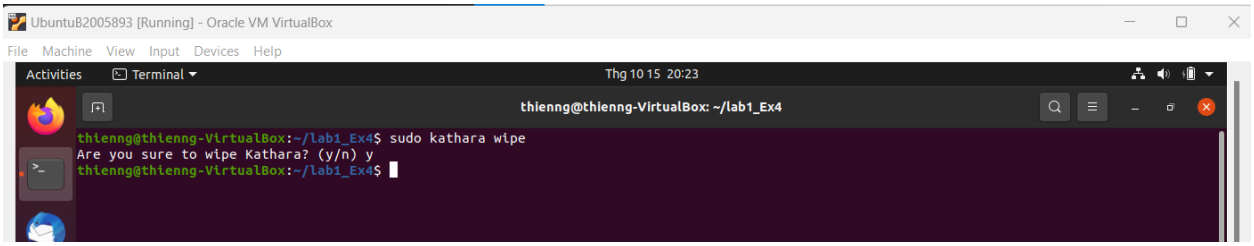
From pc1: # tcpdump –tenni eth0



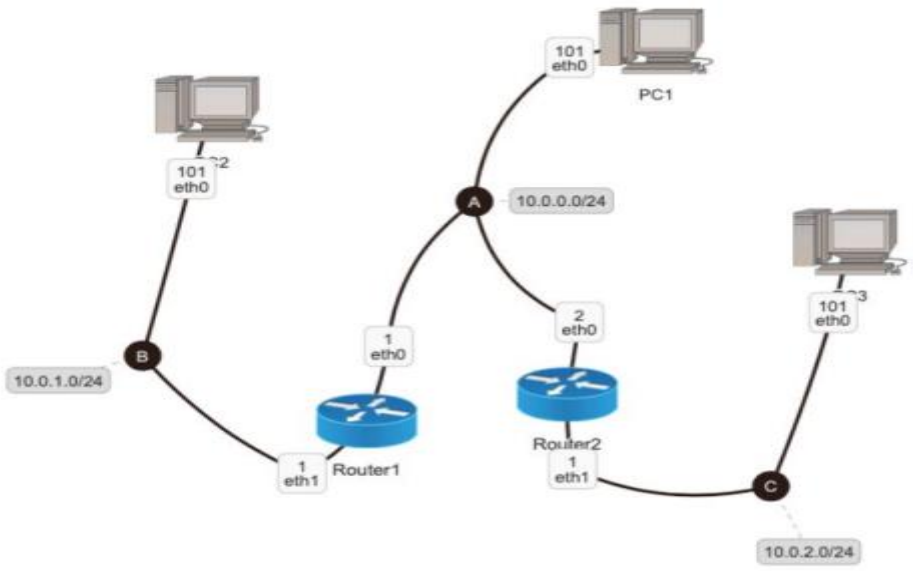
6. *Arp implementation details*
   From r2:
   # tcpdump –tenni eth0



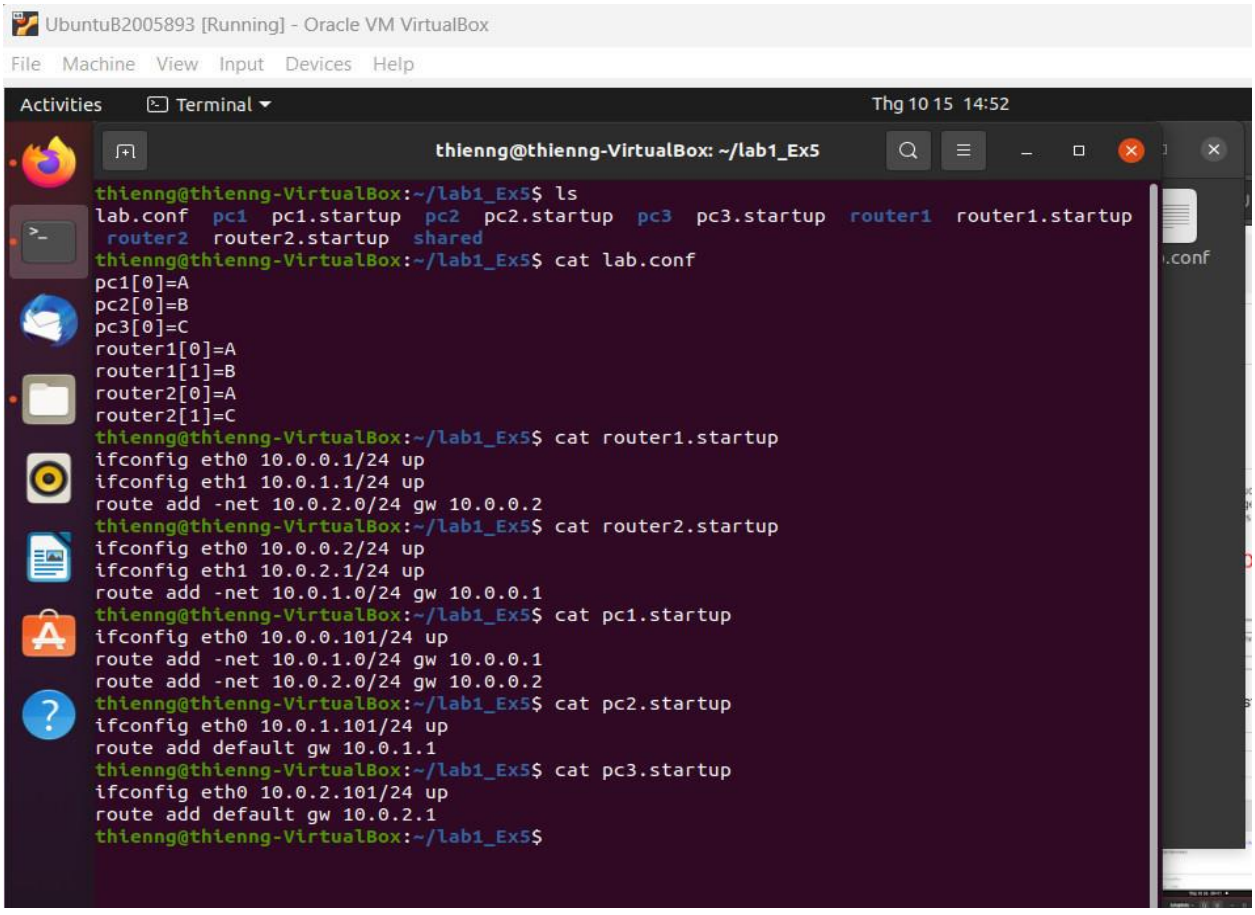
7. *Delete all vms with $ sudo kathara wipe*



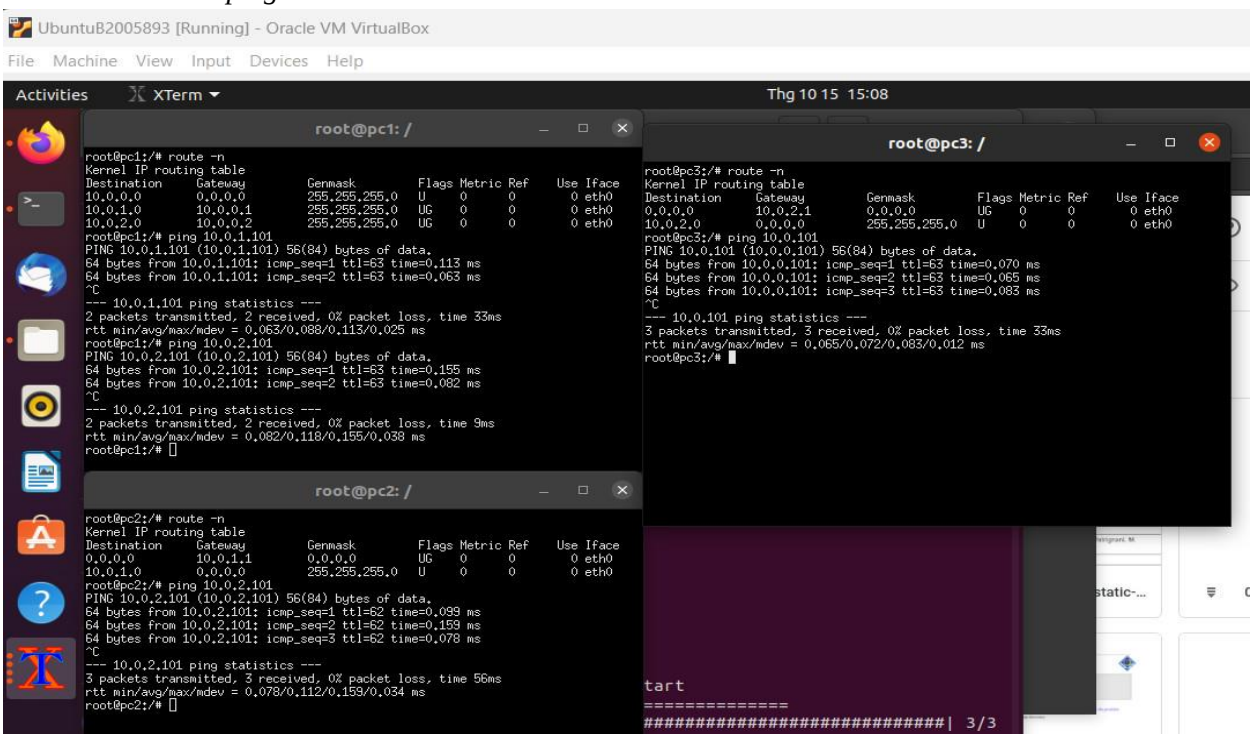
Exercise 5: Construct the following network



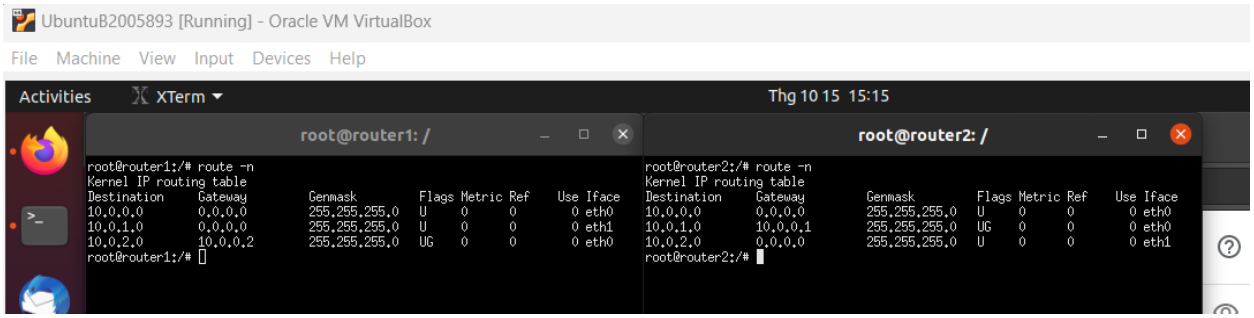
1. *Lab configuration*



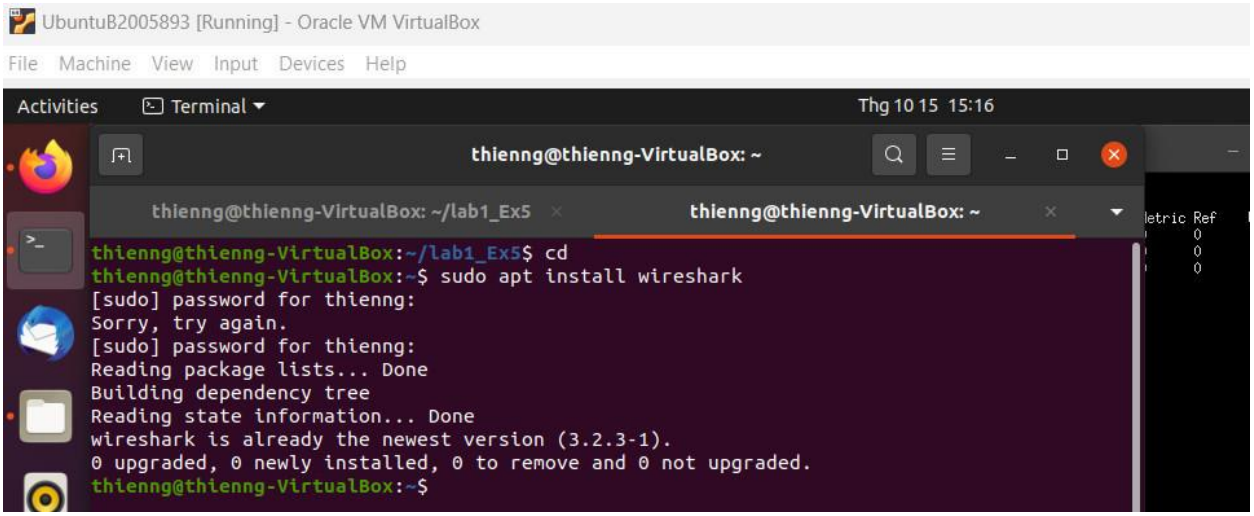
2. *Route and ping to check*



3. *Route the routers*



4. *Already install wireshark*



5. *Tcpdump*

On pc2, type:

tcpdump -s 1536 -w /hostlab/Ex5_pc2.pcap

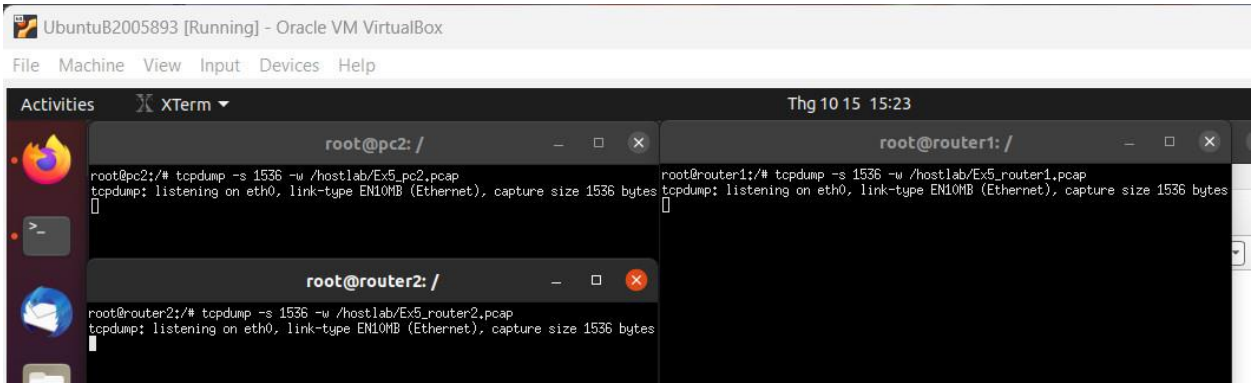
On router1, type:

tcpdump -s 1536 -w /hostlab/BT5_router1.pcap

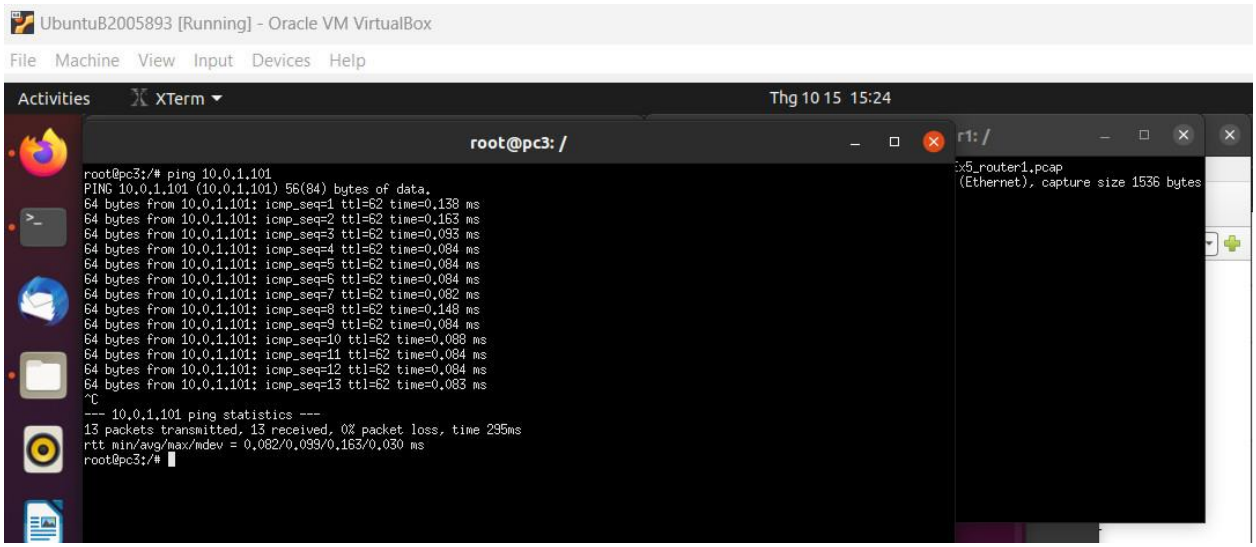
On router2, type:

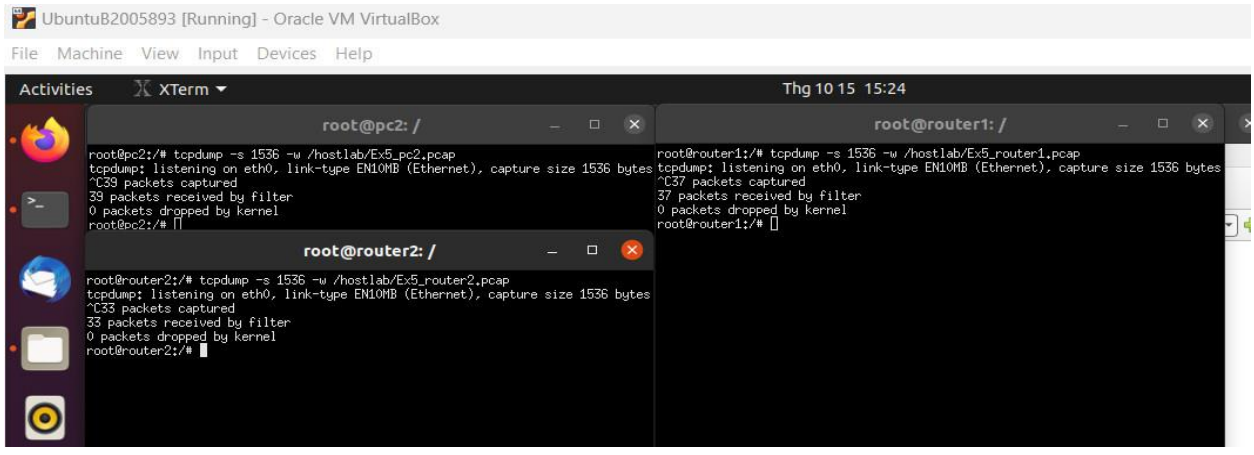
tcpdump -s 1536 -w /hostlab/BT5_router2.pcap

→ All packets are save in .pcap files which are in the /shared folder

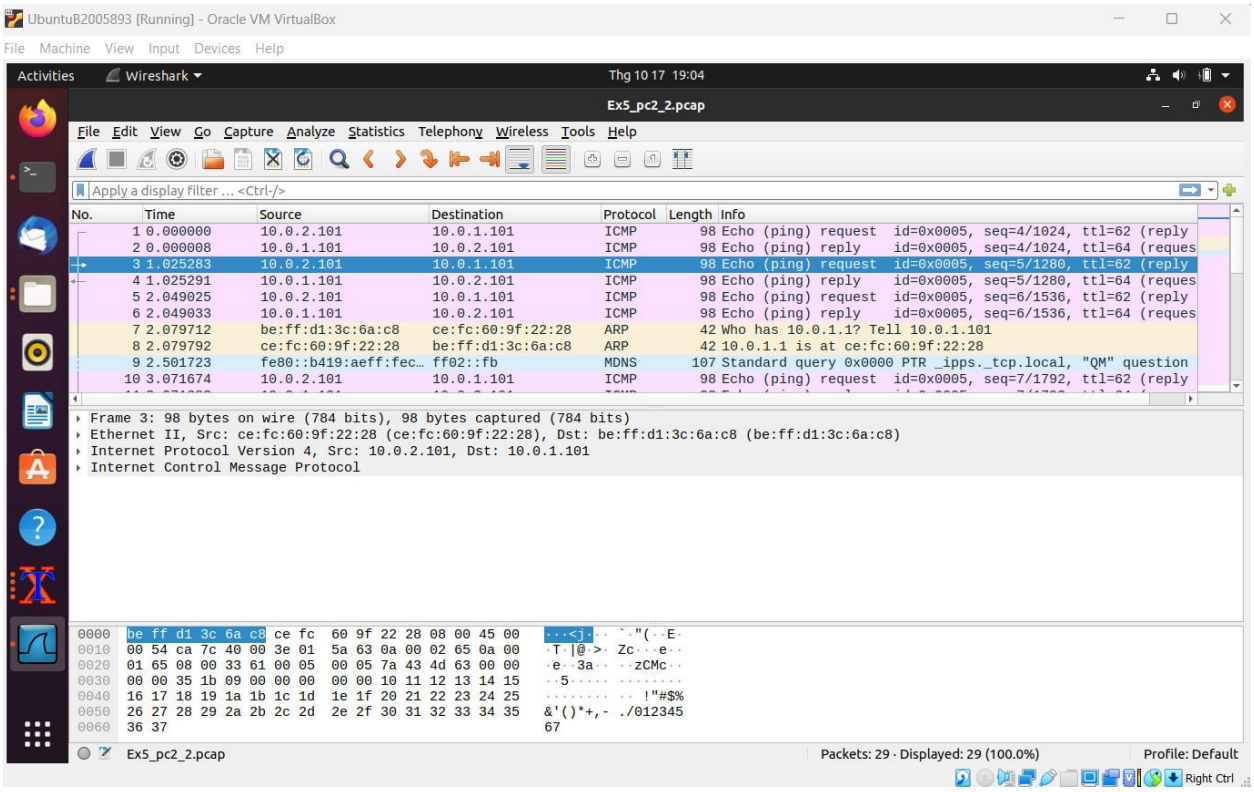


6. *ping on pc3, wait for 10s then stop pc3, pc2, router1, route2*





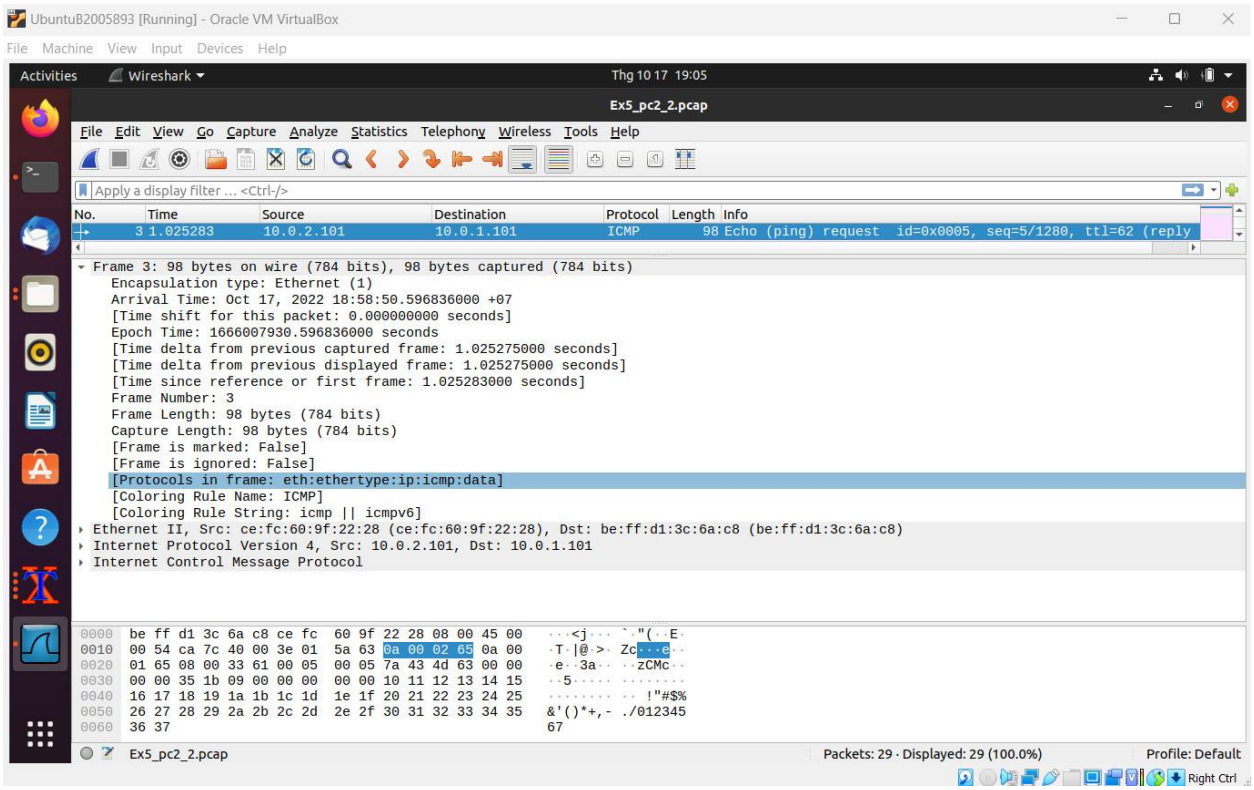
7. *Open pc2.pcap on wireshark*



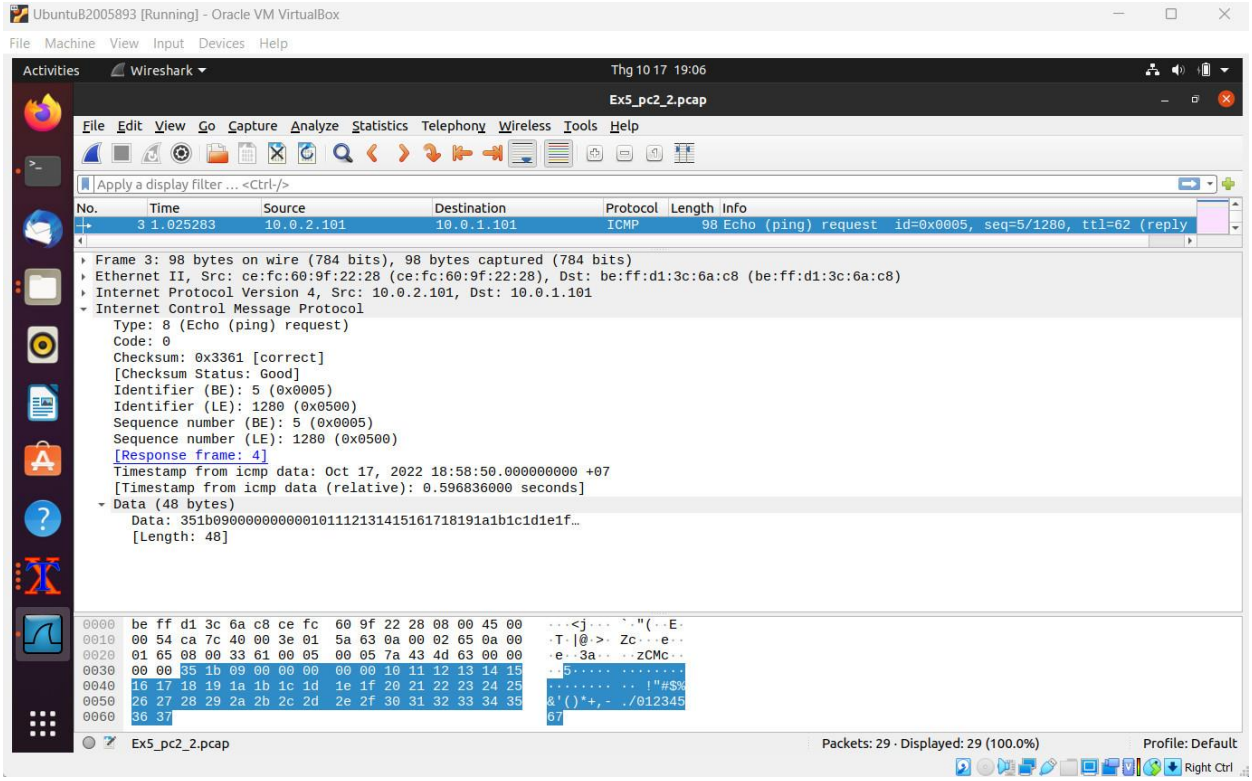
QUESTIONS

Reuse the network of Exercise 5 and answer questions. Also select the third frame of ICMP

1. Size of frame in bytes?
   ➔ 98 bytes

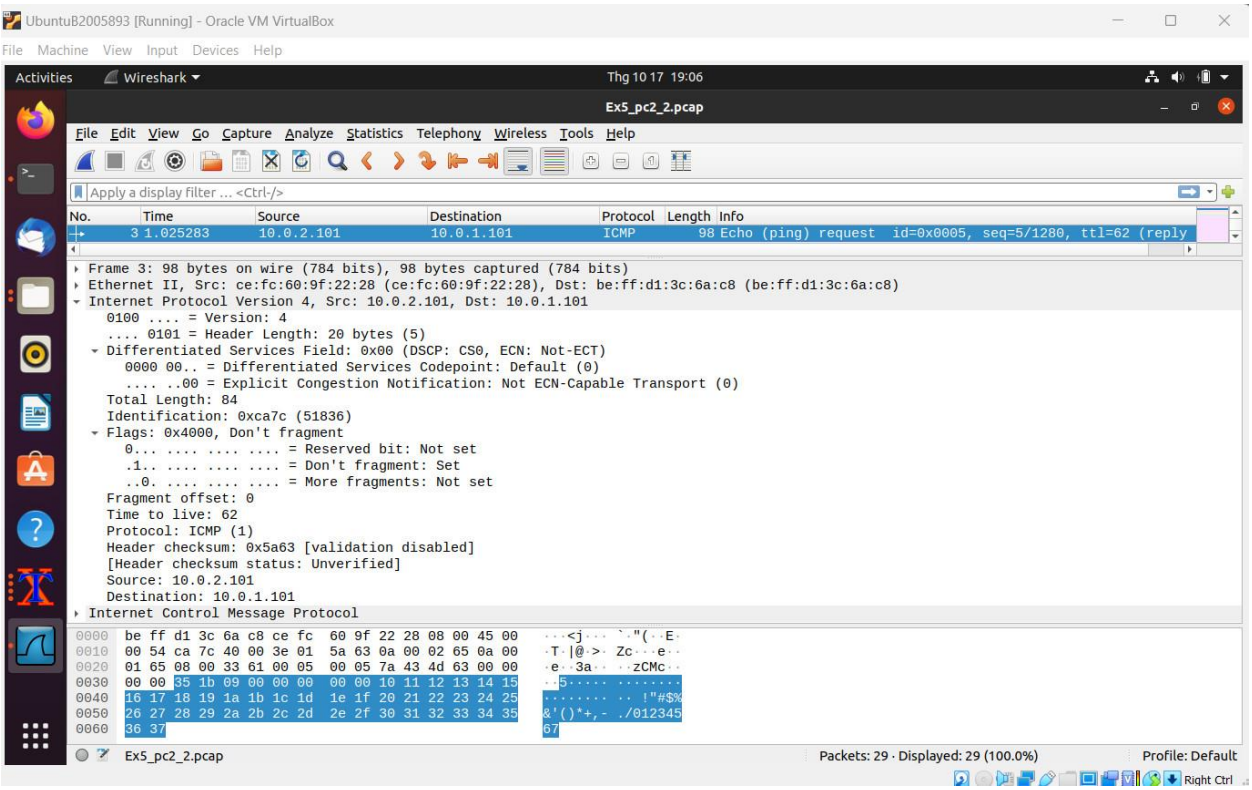


2. Select Header Internet Control Message Protocol → which protocol is using? On which layer of the OSI model does this protocol operate? What is the content of the message? How long is this message in bytes?



a. It is using the ICMP protocol.
b. ICMP (Internet Control Message Protocol) is located at the Network layer(layer3) of the OSI model
c. The message contains (Time, Source, Destination, Protocol, Length and Info) of the frame
d. This message is 48 bytes

3. Select Header Internet Protocol Version 4

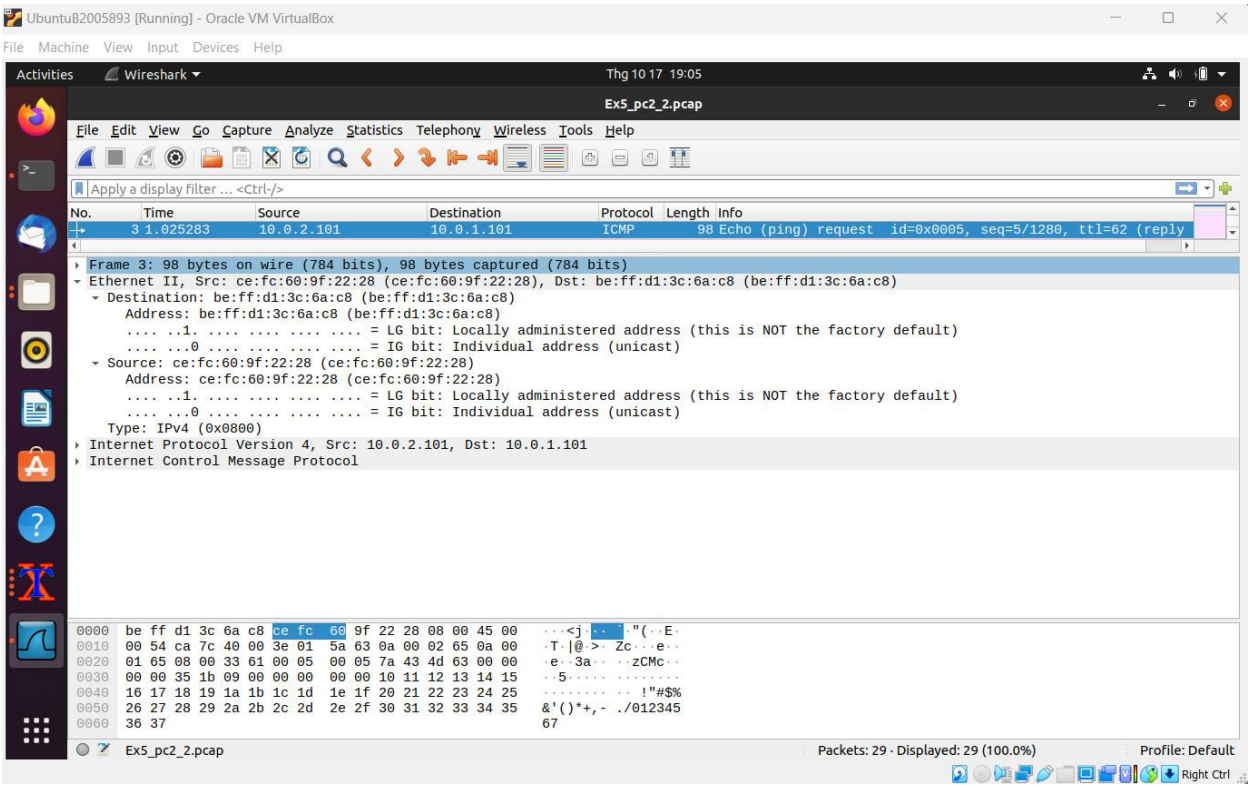


a. What are the IP addresses of the source and destination hosts?
   - ➔ Source IP address: 10.0.2.101
   - ➔ Destination IP address: 10.0.1.101

b. What is the length of the IP packet header? What fields does the Header include? How long is each field (Bytes)?
   - ➔ IP packet header is 20 bytes length
   - ➔ The Header include the source ip, destination ip, differentiated services field and flags
   - ➔ Each field is 5 bytes

c. What is the length of the Total Length field (Bytes)?
   - ➔ The total length is 84 bytes

4. Select Header Ethernet II



a. What are the MAC addresses of the source and the destination hosts?

➔ The MAC address of source is ce:fc:60:9f:22:28

➔ The MAC address of destination is be:ff:d1:3c:6a:c8

b. What is the Type value?

➔ The Type value is 0x0800