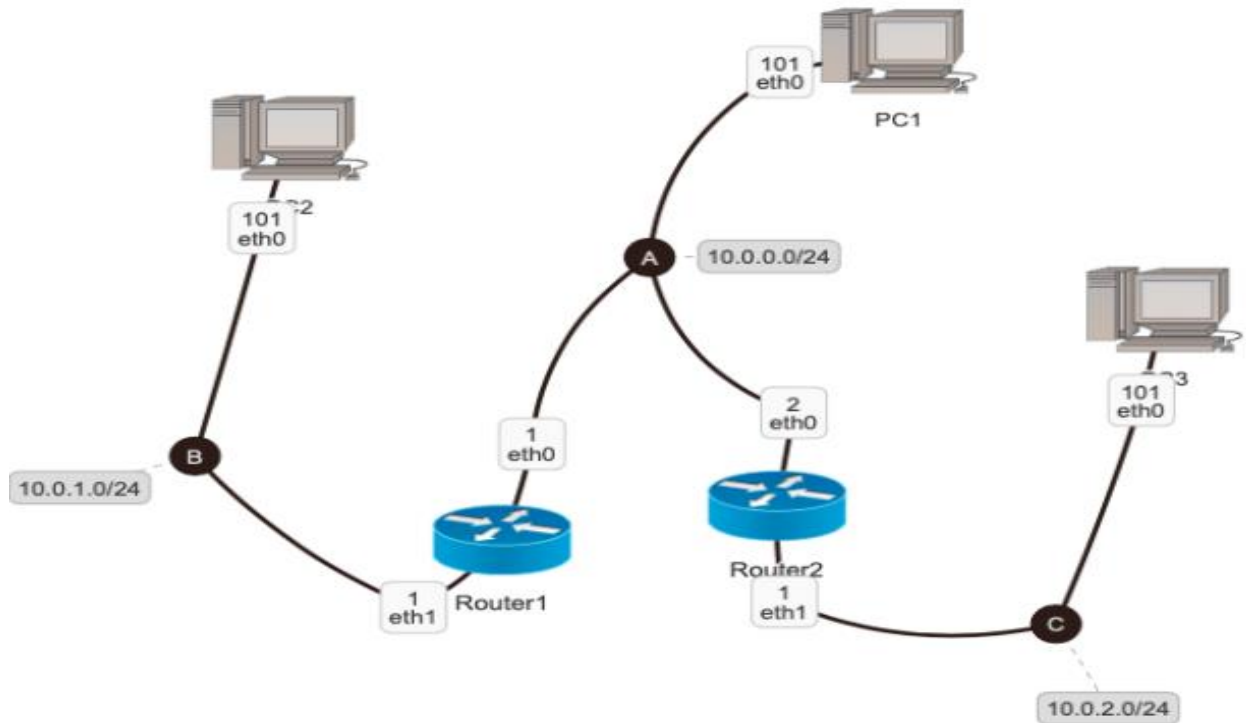


LAB 2

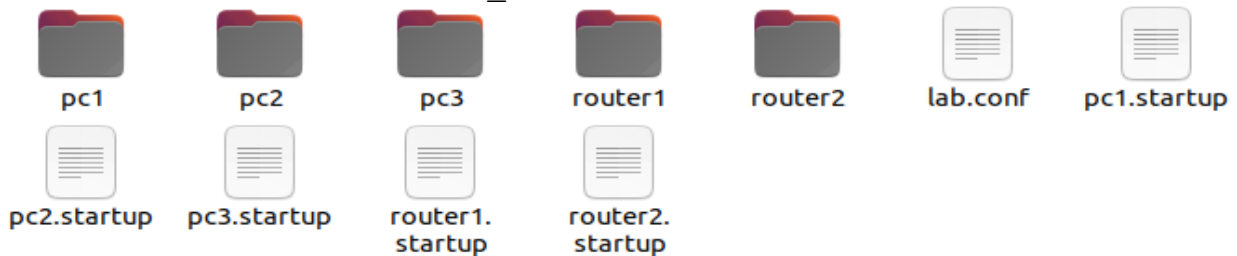
Bài 5:

- 1) Quan sát mô hình mạng cần xây dựng và nhận diện các thiết bị, giao diện với các địa chỉ IP được gán trên các máy ảo.



- 2) Tạo thư mục BaiTap5 trong workspace của sinh viên. Thư mục sẽ này chứa các thư mục con và các file cấu hình (.startup, lab.conf) theo cấu trúc quy định của Kathará. Trên máy thực, di chuyển đến thư mục BaiTap5 bằng lệnh:

```
$ cd /home/student/your_workspace/BaiTap5
```



3) Trên file **lab.conf**, soạn thảo nội dung mô tả hình thái mạng theo thiết kế:

```
GNU nano 4.8 lab.conf Modified
pc1[0]=A
pc2[0]=B
pc3[0]=C
router1[0]=A
router1[1]=B
router2[0]=A
router2[1]=C
```

4) Trên file **pc1.startup**:

```
GNU nano 4.8 pc1.startup Modified
ifconfig eth0 10.0.0.101/24 up
route add -net 10.0.1.0/24 gw 10.0.0.1
route add -net 10.0.2.0/24 gw 10.0.0.2
```

5) Thêm thông tin vạch đường đến nhánh LAN A, nhánh LAN C trên **pc2.startup** và thêm thông tin vạch đường đến nhánh LAN A, nhánh LAN B trên **pc3.startup**

```
GNU nano 4.8 pc2.startup Modified
ifconfig eth0 10.0.1.101/24 up
route add default gw 10.0.1.1
```

```
GNU nano 4.8 pc3.startup Modified
ifconfig eth0 10.0.2.101/24 up
route add default gw 10.0.2.1
```

6) Thêm thông tin vạch đường trên **router1.startup** và **router2.startup** bằng lệnh route add -net đã được hướng dẫn nhằm giúp cho router1 biết đường đi tới LAN C và router2 biết đường đi tới LAN B.

Nội dung file router1.startup có thể được trình bày như sau:

```
GNU nano 4.8 router1.startup Modified
ifconfig eth0 10.0.0.1/24 up
ifconfig eth1 10.0.1.1/24 up
route add -net 10.0.2.0/24 gw 10.0.0.2
```

```
GNU nano 4.8 router2.startup Modified
ifconfig eth0 10.0.0.2/24 up
ifconfig eth1 10.0.2.1/24 up
route add -net 10.0.1.0/24 gw 10.0.0.1
```

7) Khởi động mạng ảo BaiTap5. Kiểm tra bảng vạch đường (bằng lệnh route) trên từng thiết bị mạng (máy ảo).

\$ kathara lstart

```
root@pc1: /
--- Startup Commands Log
++ ifconfig eth0 10.0.0.101/24 up
++ route add -net 10.0.1.0/24 gw 10.0.0.1
++ route add -net 10.0.2.0/24 gw 10.0.0.2
--- End Startup Commands Log
```

```

root@pc2: /
--- Startup Commands Log
++ ifconfig eth0 10.0.1.101/24 up
++ route add default gw 10.0.1.1
--- End Startup Commands Log

root@pc3: /
--- Startup Commands Log
++ ifconfig eth0 10.0.2.101/24 up
++ route add default gw 10.0.2.1
--- End Startup Commands Log

root@router1: /
--- Startup Commands Log
++ ifconfig eth0 10.0.0.1/24 up
++ ifconfig eth1 10.0.1.1/24 up
++ route add -net 10.0.2.0/24 gw 10.0.0.2

root@router2: /
--- Startup Commands Log
++ ifconfig eth0 10.0.0.2/24 up
++ ifconfig eth1 10.0.2.1/24 up
++ route add -net 10.0.1.0/24 gw 10.0.0.1

```

Nếu bảng vạch đường của máy ảo nào đó bị sai, thực hiện:

- Kiểm tra lại cấu hình mạng của máy ảo đó bằng lệnh ifconfig.
- Tắt máy ảo (lcrash) và chỉnh sửa lại chỗ sai trước khi khởi động lại máy ảo (lstart)
- Đôi khi, việc khởi động lại toàn bộ mạng ảo (lrestart) nên được thực hiện sau khi đã chỉnh xong lỗi sai cho thao tác cấu hình của 1 máy ảo.

8) Trường hợp bảng vạch đường của các thiết bị đều đúng, trên **pc2**, **router1** và **router2** lần lượt thực hiện lệnh tcpdump với cú pháp như sau:

\$ tcpdump -s 1536 -w /hostlab/BT5_pc2.pcap (trên pc2)

```

root@pc2:/# tcpdump -s 1536 -w /hostlab/BT5_pc2.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

\$ tcpdump -s 1536 -w /hostlab/BT5_router1.pcap (trên router1)

```

root@router1:/# tcpdump -s 1536 -w /hostlab/BT5_router1.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

\$ tcpdump -s 1536 -w /hostlab/BT5_router2.pcap (trên router2)

```

root@router2:/# tcpdump -s 1536 -w /hostlab/BT5_router2.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

9) Trên **pc3** thực hiện gửi dữ liệu đến **pc2** bằng lệnh:

```
$ ping 10.0.1.101
```

```
root@pc3:/# ping 10.0.1.101
PING 10.0.1.101 (10.0.1.101) 56(84) bytes of data.
64 bytes from 10.0.1.101: icmp_seq=1 ttl=62 time=0.117 ms
64 bytes from 10.0.1.101: icmp_seq=2 ttl=62 time=0.129 ms
64 bytes from 10.0.1.101: icmp_seq=3 ttl=62 time=0.128 ms
64 bytes from 10.0.1.101: icmp_seq=4 ttl=62 time=0.131 ms
64 bytes from 10.0.1.101: icmp_seq=5 ttl=62 time=0.133 ms
64 bytes from 10.0.1.101: icmp_seq=6 ttl=62 time=0.075 ms
64 bytes from 10.0.1.101: icmp_seq=7 ttl=62 time=0.121 ms
64 bytes from 10.0.1.101: icmp_seq=8 ttl=62 time=0.110 ms
64 bytes from 10.0.1.101: icmp_seq=9 ttl=62 time=0.076 ms
64 bytes from 10.0.1.101: icmp_seq=10 ttl=62 time=0.126 ms
```

và chờ khoảng 10 giây, sau đó dừng lệnh ping lại.

Dừng các lệnh tcpdump trên pc2, router1 và router2 lại.

10) Trên máy thực, mở file **BT5_pc2.pcap** bằng Wireshark. Chọn khung vật lý (physical frame) số 3 và trả lời các câu hỏi sau đây:

❖ Toàn bộ khung số 3 có kích thước là bao nhiêu (Bytes)? Có kích thước 196 bytes

No.	Time	Source	Destination	Protocol	Len
4	6.277690	fe80::486:a5ff:fece...	ff02::fb	MDNS	
5	35.524202	02:42:ac:d1:00:03	Broadcast	ARP	
6	35.524206	02:42:ac:d1:00:02	02:42:ac:d1:00:03	ARP	
7	35.524211	10.0.2.101	10.0.1.101	ICMP	
8	35.524218	10.0.1.101	10.0.2.101	ICMP	
9	36.544655	10.0.2.101	10.0.1.101	ICMP	
10	36.544665	10.0.1.101	10.0.2.101	ICMP	

▶ Frame 9: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
 ▶ Ethernet II, Src: 02:42:ac:d1:00:03 (02:42:ac:d1:00:03), Dst: 02:42:ac:d1:00:02
 ▶ Internet Protocol Version 4, Src: 10.0.2.101, Dst: 10.0.1.101
 ▶ Internet Control Message Protocol

❖ Chọn **Header Internet Control Message Protocol** trong khung và cho biết:

- Gói tin này sử dụng giao thức gì? Giao thức này hoạt động trên tầng nào của mô hình OSI?
 - Gói tin này sử dụng giao thức ICMP, giao thức này hoạt động trên Tầng Mạng
- Nội dung thông điệp của giao thức này là gì? Thông điệp này có độ dài bao nhiêu (Bytes)?

▼ Data (48 bytes)

Data: 07cd0a0000000000101112131415161718191a1b1c1d1e1f...
[Length: 48]

- Nội dung của thông điệp: một thông điệp chứa gói dữ liệu khi ping
- Thông điệp có độ dài: 48 bytes

❖ Chọn **Header Internet Protocol Version 4** và cho biết:

- Địa chỉ IP của máy gửi dữ liệu là bao nhiêu? Địa chỉ IP này là của máy tính nào trong mạng?
 - Địa chỉ IP của máy gửi dữ liệu: 10.0.2.101, đây là địa chỉ IP của PC3

- Địa chỉ IP của máy nhận dữ liệu là bao nhiêu? Địa chỉ IP này là của máy tính nào trong mạng?
 - Địa chỉ IP của máy nhận dữ liệu là: 10.0.1.101, đây là địa chỉ IP của PC2
- Định danh (ID) của gói tin IP này là bao nhiêu (dạng hexadecimal). Định danh của 1 gói tin có ý nghĩa gì trong thông điệp IP?
 - Định danh ID = 0x9c45
 - Định danh cho gói tin được lựa chọn bởi nguồn gửi gói tin. Nếu gói tin IPv4 bị phân mảnh, mọi phân mảnh sẽ giữ lại giá trị trường định danh này, mục đích để nút đích có thể nhóm lại các mảnh, phục vụ cho việc phục hồi lại gói tin.
- Độ dài phần Header của thông điệp IP là bao nhiêu? Phần Header bao gồm những trường (fields) nào? Mỗi trường có độ dài bao nhiêu?

02	42	ac	d1	00	02	02	42	ac	d1	00	03	08	00	45	00
00	54	b6	93	40	00	3e	01	6e	4c	0a	00	02	65	0a	00
01	65	08	00	72	00	00	38	00	02	f4	c5	c0	5f	00	00
00	00	07	cd	0a	00	00	00	00	00	10	11	12	13	14	15

- Độ dài phần Header của thông điệp IP là: 160 bit (20 bytes)
- Phần Header gồm những trường(fields):
 - Version 4 bits
 - IHL 4 bits
 - Differentiated Services 8 bits
 - Total Length 16 bits
 - Identification 16 bits
 - Flags 3 bits
 - Fragment Offset 13 bits
 - Time To Live 8 bit
 - Protocol 8 bits
 - Header Checksum 16 bits
 - Source IP Address 32 bits
 - Destination IP Address 32 bits
- Trường Total Length có độ dài là bao nhiêu (Bytes). Hãy lý giải tại sao có độ dài như vậy?
 - Trường Total Length có độ dài: 84 bits = (10.5 bytes ~ 11bytes)
 - Đây là (Độ dài của tiêu đề IP + Độ dài của payload)

❖ Chọn **Header Ethernet II** và cho biết:

- Địa chỉ MAC của máy gửi dữ liệu là bao nhiêu? Có phải là địa chỉ MAC của máy tính có địa chỉ IP (source) đã tìm được trong câu trên không? Nếu không, hãy lý giải và cho biết địa chỉ MAC này là của máy tính nào trong mạng?
 - 02:42:ac:d1:00:03 là địa chỉ MAC của pc3
- Địa chỉ MAC của máy nhận dữ liệu là bao nhiêu? Có phải là địa chỉ MAC của máy tính có địa chỉ IP (destination) đã tìm được trong câu trên không? Nếu không, hãy lý giải và cho biết địa chỉ MAC này là của máy tính nào trong mạng?
 - 02:42:ac:d1:00:02 là địa chỉ MAC của pc2
- Trường Type mang giá trị (hexadecimal) bằng bao nhiêu? Thông tin thể hiện là gì?
 - 0X0800
 - Thông tin xác định các giao thức khác nhau của lớp mạng
- Hãy chỉ ra trường Payload của khung Ethernet II? Trường Payload này có độ dài bằng bao nhiêu (Bytes)?

02	42	ac	d1	00	02	02	42	ac	d1	00	03	08	00	45	00
00	54	b6	93	40	00	3e	01	6e	4c	0a	00	02	65	0a	00
01	65	08	00	72	00	00	38	00	02	f4	c5	c0	5f	00	00
00	00	07	cd	0a	00	00	00	00	00	10	11	12	13	14	15

Ethernet (eth), 14 bytes Packets: 28 · Displayed

- Có độ dài 14 bytes

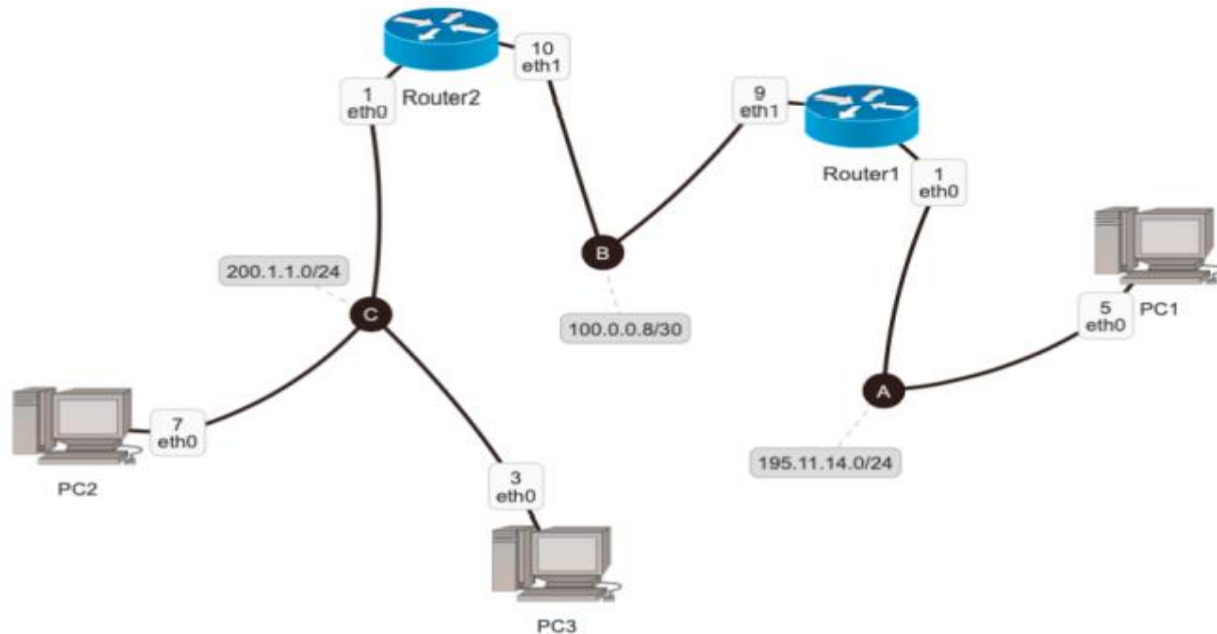
11) Hủy mạng ảo sau khi đã thực hiện xong Bài tập 5

\$ kathara wipe

```
b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap5$ kathara wipe
Are you sure to wipe Kathara? (y/n) y
```

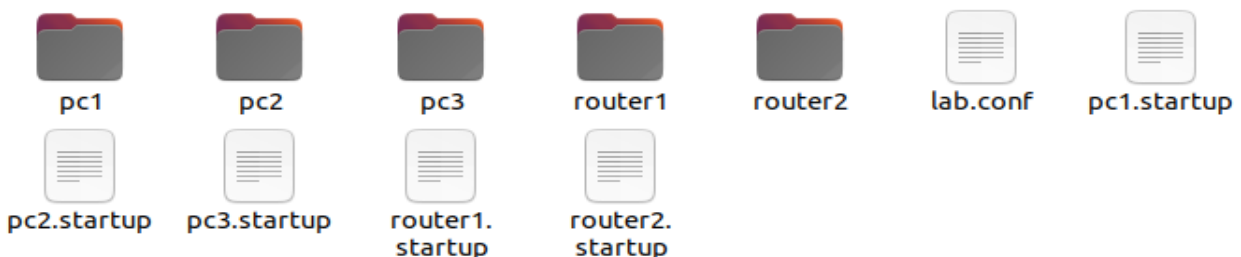
Bài 6:

- 1) Quan sát mô hình mạng cần xây dựng và nhận diện các thiết bị, giao diện với các địa chỉ IP được gán trên các máy ảo.



- 2) Tạo thư mục BaiTap6 trong workspace của sinh viên. Thư mục sẽ này chứa các thư mục con và các file cấu hình (.startup, lab.conf) theo cấu trúc quy định của Kathará. Trên máy thực, di chuyển đến thư mục BaiTap6 bằng lệnh:

```
$ cd /home/student/your_workspace/BaiTap6
```



- 3) Trên file **lab.conf**, soạn thảo nội dung mô tả hình thái mạng theo thiết kế :

```
GNU nano 4.8 lab.conf Modified
pc1[0]=A
pc2[0]=C
pc3[0]=C

router1[0]=A
router1[1]=B
router2[0]=C
router2[1]=B
```


- 4) Trên file **pc1.startup**, **pc2.startup** và **pc3.startup** để vạch đường mặc nhiên thì sẽ chứa nội dung được miêu tả như sau:

```
GNU nano 4.8 pc1.startup
ifconfig eth0 195.11.14.5/24 up
route add default gw 195.11.14.1
```

```
GNU nano 4.8 pc2.startup Modified
ifconfig eth0 200.1.1.7/24 up
route add default gw 200.1.1.1
```

```
GNU nano 4.8 pc3.startup Modified
ifconfig eth0 200.1.1.3/24 up
route add default gw 200.1.1.1
```

- 5) Trên file **router1.startup** và **router2.startup** cũng thực hiện thêm thông tin vạch đường tĩnh sao cho router1 biết hướng đi tới LAN C và router2 biết hướng đi tới LAN A.

Nội dung file **router1.startup** có thể được trình bày như sau:

```
GNU nano 4.8 router1.startup Modified
ifconfig eth0 195.11.14.1/24 up
ifconfig eth1 100.0.0.9/30 up
route add -net 200.1.1.0/24 gw 100.0.0.10
```

```
GNU nano 4.8 router2.startup Modified
ifconfig eth0 200.1.1.1/24 up
ifconfig eth1 100.0.0.10/30 up
route add -net 195.11.14.0/24 gw 100.0.0.9
```

- 6) Khởi động mạng ảo BaiTap6. Kiểm tra bảng vạch đường (lệnh route) và địa chỉ IP của các giao diện mạng (lệnh ifconfig) trên từng máy ảo để đảm bảo tính đúng đắn của mô hình mạng Bài Tập 6. Nếu có sai sót, thực hiện các thao tác điều chỉnh theo hướng dẫn đã trình bày trong 7) của Bài Tập 5.

```
$ kathara lstart
```

```
root@pc1: /
--- Startup Commands Log
++ ifconfig eth0 195.11.14.5/24 up
++ route add default gw 195.11.14.1
--- End Startup Commands Log
```

```
root@pc2: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.7/24 up
++ route add default gw 200.1.1.1
--- End Startup Commands Log
```



```
root@pc3: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.3/24 up
++ route add default gw 200.1.1.1
--- End Startup Commands Log
```

```
root@router1: /
--- Startup Commands Log
++ ifconfig eth0 195.11.14.1/24 up
++ ifconfig eth1 100.0.0.9/30 up
++ route add -net 200.1.1.0/24 gw 100.0.0.10
--- End Startup Commands Log
```

```
root@router2: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.1/24 up
++ ifconfig eth1 100.0.0.10/30 up
++ route add -net 195.11.14.0/24 gw 100.0.0.9
--- End Startup Commands Log
```

2.3.2.1. Giao thức ARP giữa 2 thiết bị trong cùng nhánh mạng LAN C

7) Trên máy ảo **pc3**, **pc2** và **router2**, lần lượt dùng lệnh **arp**. Nhận xét kết quả:

```
root@pc2:/# arp
root@pc2:/# █
```

```
root@pc3:/# arp
root@pc3:/# █
```

```
root@router2:/# arp
root@router2:/# █
```

8) Lần lượt thực hiện lệnh **tcpdump** với cú pháp như sau:

\$ **tcpdump -s 1536 -w /hostlab/BT6_pc2_A.pcap** (trên pc2)

```
root@pc2: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.7/24 up
++ route add default gw 200.1.1.1
--- End Startup Commands Log
root@pc2:/# arp
root@pc2:/# tcpdump -s 1536 -w /hostlab/BT6_pc2_A.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes
█
```

\$ tcpdump -s 1536 -w /hostlab/BT6_router1_A.pcap (trên router1)

```

root@router1: /
--- Startup Commands Log
++ ifconfig eth0 195.11.14.1/24 up
++ ifconfig eth1 100.0.0.9/30 up
++ route add -net 200.1.1.0/24 gw 100.0.0.10
--- End Startup Commands Log

root@router1:/# tcpdump -s 1536 -w /hostlab/BT6_router1_A.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

\$ tcpdump -s 1536 -w /hostlab/BT6_router2_A.pcap (trên router2)

```

root@router2: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.1/24 up
++ ifconfig eth1 100.0.0.10/30 up
++ route add -net 195.11.14.0/24 gw 100.0.0.9
--- End Startup Commands Log

root@router2:/# arp
root@router2:/# tcpdump -s 1536 -w /hostlab/BT6_router2_A.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

9) Trên **pc3** thực hiện gửi dữ liệu đến **pc2** bằng lệnh:

\$ ping 200.1.1.7

và chờ khoảng 10 giây

sau đó dừng lệnh ping trên **pc3** lại. Dừng các lệnh **tcpdump** trên **pc2**, **router1** và **router2** lại.

```

root@pc3: /
--- Startup Commands Log
++ ifconfig eth0 200.1.1.3/24 up
++ route add default gw 200.1.1.1
--- End Startup Commands Log

root@pc3:/# arp
root@pc3:/# ping 200.1.1.7
PING 200.1.1.7 (200.1.1.7) 56(84) bytes of data:
64 bytes from 200.1.1.7: icmp_seq=1 ttl=64 time=0.073 ms
64 bytes from 200.1.1.7: icmp_seq=2 ttl=64 time=0.080 ms
64 bytes from 200.1.1.7: icmp_seq=3 ttl=64 time=0.079 ms
64 bytes from 200.1.1.7: icmp_seq=4 ttl=64 time=0.075 ms
64 bytes from 200.1.1.7: icmp_seq=5 ttl=64 time=0.079 ms
64 bytes from 200.1.1.7: icmp_seq=6 ttl=64 time=0.055 ms
64 bytes from 200.1.1.7: icmp_seq=7 ttl=64 time=0.098 ms
64 bytes from 200.1.1.7: icmp_seq=8 ttl=64 time=0.082 ms
64 bytes from 200.1.1.7: icmp_seq=9 ttl=64 time=0.082 ms
64 bytes from 200.1.1.7: icmp_seq=10 ttl=64 time=0.078 ms

```

10) Trên **pc3** thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Lưu ý sự thay đổi so với kết quả ở 7). Lý giải cho sự thay đổi này. Ghi nhận kết quả hiển thị để so sánh với 10) ở mục 2.3.2.2.

```

root@pc3:/# arp
Address          Hwtype  Hwaddress      Flags Mask      Iface
200.1.1.7        ether    fe:19:75:77:0c:02 C                eth0
root@pc3:/#

```

11) Trên **pc2**, thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Lưu ý sự thay đổi so với kết quả ở bước số 7). Lý giải cho sự thay đổi này.

```
root@pc2:/# arp
Address          HWtype  HWaddress      Flags Mask    Iface
200.1.1.3        ether    82:d3:51:9b:83:a8  C             eth0
root@pc2:/#
```

12) Trên **router2**, thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Ghi nhận kết quả hiển thị để so sánh với bước 12) ở mục 2.3.2.2.

```
root@router2:/# arp
root@router2:/#
```

13) Trên máy thực, dùng Wireshark mở file **BT6_router2_A.pcap**, chọn khung vật lý số thứ tự 1. Trả lời các câu hỏi sau đây:

❖ Toàn bộ khung số thứ tự 1 có kích thước là bao nhiêu (Bytes)?

No.	Time	Source	Destination	Protocol	Len
1	0.000000	02:42:ac:14:00:03	Broadcast	ARP	
2	0.000026	02:42:ac:14:00:02	02:42:ac:14:00:03	ARP	
3	0.000051	200.1.1.3	200.1.1.7	ICMP	
4	0.000077	200.1.1.7	200.1.1.3	ICMP	
5	1.023147	200.1.1.3	200.1.1.7	ICMP	

▶ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
 ▶ Ethernet II, Src: 02:42:ac:14:00:03 (02:42:ac:14:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 ▶ Address Resolution Protocol (request)

- Có kích thước 84 bytes

❖ Chọn **Header Address Resolution Protocol** và cho biết:

- Trường Opcode có giá trị (hexadecimal) là bao nhiêu? Giá trị của trường này thể hiện thông tin gì? Trường Opcode này còn có thể có giá trị (hexadecimal) là bao nhiêu nữa và thể hiện thông tin gì?
 - 00 01, thể hiện thông tin gửi yêu cầu
- Địa chỉ IP và địa chỉ MAC của máy gửi dữ liệu? Đây là địa chỉ IP và MAC của máy tính nào trong mạng?

Sender MAC address: 02:42:ac:14:00:03 (02:42:ac:14:00:03)	
Sender IP address: 200.1.1.3	
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)	
Target IP address: 200.1.1.7	
0000	ff ff ff ff ff ff 02 42 ac 14 00 03 08 06 00 01
0010	08 00 06 04 00 01 02 42 ac 14 00 03 c8 01 01 03
0020	00 00 00 00 00 00 c8 01 01 07

- Máy pc3

- Địa chỉ IP và địa chỉ MAC của máy nhận dữ liệu? Đây là địa chỉ IP và MAC của máy tính nào trong mạng? Nhận xét về cặp địa chỉ IP và MAC của máy nhận dữ liệu.

Sender IP address: 200.1.1.3	
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)	
Target IP address: 200.1.1.7	
0000	ff ff ff ff ff ff 02 42 ac 14 00 03 08 06 00 01
0010	08 00 06 04 00 01 02 42 ac 14 00 03 c8 01 01 03
0020	00 00 00 00 00 00 c8 01 01 07

- Máy pc2
- Khác nhau

❖ Chọn **Header Ethernet II** và cho biết:

- Địa chỉ MAC của máy gửi dữ liệu là bao nhiêu? Địa chỉ MAC này là của máy tính nào trong mạng?

▼ Ethernet II, Src: 02:42:ac:14:00:03 (02:42:ac:14:00:03), Dst:
▶ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
▶ Source: 02:42:ac:14:00:03 (02:42:ac:14:00:03)
Type: ARP (0x0806)
▶ Address Resolution Protocol (request)

0000	ff ff ff ff ff ff 02 42 ac 14 00 03 08 06 00 01
0010	08 00 06 04 00 01 02 42 ac 14 00 03 c8 01 01 03

- Máy pc3

- Địa chỉ MAC của máy nhận dữ liệu là bao nhiêu? Địa chỉ MAC này là của máy tính nào trong mạng? Nhận xét về địa chỉ MAC này và địa chỉ MAC của máy nhận dữ liệu đã quan sát được ở phần Header Address Resolution Protocol

▼ Ethernet II, Src: 02:42:ac:14:00:03 (02:42:ac:14:00:03), Dst:
▶ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
▶ Source: 02:42:ac:14:00:03 (02:42:ac:14:00:03)
Type: ARP (0x0806)
▶ Address Resolution Protocol (request)

0000	ff ff ff ff ff ff 02 42 ac 14 00 03 08 06 00 01
0010	08 00 06 04 00 01 02 42 ac 14 00 03 c8 01 01 03
0020	00 00 00 00 00 00 c8 01 01 07

- Đây là địa chỉ broadcast.

- Trường Type mang giá trị (hexadecimal) bằng bao nhiêu? Thông tin thể hiện là gì?

▶ Source: 02:42:ac:14:00:03 (02:42:ac:14:00:03)

Type: ARP (0x0806)

▶ Address Resolution Protocol (request)

- 0x0806, thể hiện truyền bằng giao thức ARP

14) Hủy mạng ảo sau khi đã thực hiện xong phần 2.3.2.1.

\$ kathara wipe

```
b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap6$ kathara wipe
Are you sure to wipe Kathara? (y/n) y
```

2.3.2.2. Giao thức ARP giữa 2 thiết bị khác nhánh mạng LAN

7) Mở lại mạng ảo. Trên máy ảo **pc1** và **router1**, lần lượt dùng lệnh **arp**, nhận xét kết quả hiển thị.

```
root@pc1:/# arp
root@pc1:/#
```

```
root@router1:/# arp
root@router1:/#
```

8) Lần lượt thực hiện lệnh **tcpdump** với cú pháp như sau:

\$ tcpdump -s 1536 -w /hostlab/BT6_pc1_B.pcap (trên pc1)

```
root@pc1:/# tcpdump -s 1536 -w /hostlab/BT6_pc1_B.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

\$ tcpdump -s 1536 -w /hostlab/BT6_router1_B.pcap (trên router1)

```
root@router1:/# tcpdump -s 1536 -w /hostlab/BT6_router1_B.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

\$ tcpdump -s 1536 -w /hostlab/BT6_router2_B.pcap (trên router2)

```
root@router2:/# tcpdump -s 1536 -w /hostlab/BT6_router2_B.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1536 bytes

```

9) Trên **pc3** thực hiện gửi dữ liệu đến **pc1** bằng lệnh:

```
$ ping 195.11.14.5
```

và chờ khoảng 10 giây, sau đó dừng lệnh ping trên pc3 lại.

Dừng các lệnh tcpdump trên pc1, router1 và router2.

```
root@pc3:/# ping 195.11.14.5
PING 195.11.14.5 (195.11.14.5) 56(84) bytes of data:
64 bytes from 195.11.14.5: icmp_seq=1 ttl=62 time=0.127 ms
64 bytes from 195.11.14.5: icmp_seq=2 ttl=62 time=0.073 ms
64 bytes from 195.11.14.5: icmp_seq=3 ttl=62 time=0.074 ms
64 bytes from 195.11.14.5: icmp_seq=4 ttl=62 time=0.082 ms
64 bytes from 195.11.14.5: icmp_seq=5 ttl=62 time=0.086 ms
64 bytes from 195.11.14.5: icmp_seq=6 ttl=62 time=0.073 ms
64 bytes from 195.11.14.5: icmp_seq=7 ttl=62 time=0.097 ms
64 bytes from 195.11.14.5: icmp_seq=8 ttl=62 time=0.074 ms
64 bytes from 195.11.14.5: icmp_seq=9 ttl=62 time=0.078 ms
64 bytes from 195.11.14.5: icmp_seq=10 ttl=62 time=0.087 ms
```

10) Trên **pc3** thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Lưu ý: so sánh với 10) ở phần 2.3.2.1.

```
root@pc3:/# arp
Address          HWtype  HWaddress      Flags Mask    Iface
200.1.1.1        ether    62:17:51:2b:ab:d1  C             eth0
```

11) Trên **router2**, thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Lưu ý: so sánh với 12) ở phần 2.3.2.1.

```
root@router2:/# arp
Address          HWtype  HWaddress      Flags Mask    Iface
100.0.0.9        ether    4e:75:c8:4c:dc:eb  C             eth1
200.1.1.3        ether    be:0c:94:aa:a3:b3  C             eth0
```

12) Trên **router1**, thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị.

```
root@router1:/# arp
Address          HWtype  HWaddress      Flags Mask    Iface
100.0.0.10       ether    a2:d7:58:bd:d7:fd  C             eth1
195.11.14.5      ether    a2:20:26:d6:81:1e  C             eth0
```

13) Trên **pc1**, thực hiện lại lệnh **arp** và nhận xét kết quả hiển thị. Lưu ý: so sánh với thông tin hiển thị của máy pc1 tại 7) ở phần 2.3.2.2.

```
root@pc1:/# arp
Address          HWtype  HWaddress      Flags Mask    Iface
195.11.14.1      ether    c2:8c:23:ce:44:30  C             eth0
```

14) Trên máy thực, dùng Wireshark mở file **BT6_router1_B.pcap**, chọn khung vật lý số thứ tự 1.

- ❖ Trả lời các câu hỏi giống như 13) ở phần 2.3.2.1 đã trình bày.
- ❖ Toàn bộ khung số thứ tự 1 có kích thước là bao nhiêu (Bytes)? Có kích thước 84 bytes

No.	Time	Source	Destination	Protocol	Len
1	0.000000	02:42:ac:13:00:03	Broadcast	ARP	
2	0.000009	02:42:ac:13:00:02	02:42:ac:13:00:03	ARP	
3	0.000010	200.1.1.3	195.11.14.5	ICMP	
4	0.000024	195.11.14.5	200.1.1.3	ICMP	

▶ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
 ▶ Ethernet II, Src: 02:42:ac:13:00:03 (02:42:ac:13:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 ▶ Address Resolution Protocol (request)

- ❖ Chọn **Header Address Resolution Protocol** và cho biết:

- Trường Opcode có giá trị (hexadecimal) là bao nhiêu? Giá trị của trường này thể hiện thông tin gì? Trường Opcode này còn có thể có giá trị (hexadecimal) là bao nhiêu nữa và thể hiện thông tin gì?

Protocol Size: 4															
Opcode: request (1)															
Sender MAC address: 02:42:ac:13:00:03 (02:42:ac:13:00:03)															
0000	ff	ff	ff	ff	ff	ff	02	42	ac	13	00	03	08	06	00 01
0010	08	00	06	04	00	01	02	42	ac	13	00	03	c3	0b	0e 01
0020	00	00	00	00	00	00	c3	0b	0e	05					

- 00 01, thể hiện thông tin yêu cầu
- Địa chỉ IP và địa chỉ MAC của máy gửi dữ liệu? Đây là địa chỉ IP và MAC của máy tính nào trong mạng?

Sender MAC address: 02:42:ac:13:00:03 (02:42:ac:13:00:03)															
Sender IP address: 195.11.14.1															
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)															
Target IP address: 195.11.14.5															
0000	ff	ff	ff	ff	ff	ff	02	42	ac	13	00	03	08	06	00 01
0010	08	00	06	04	00	01	02	42	ac	13	00	03	c3	0b	0e 01

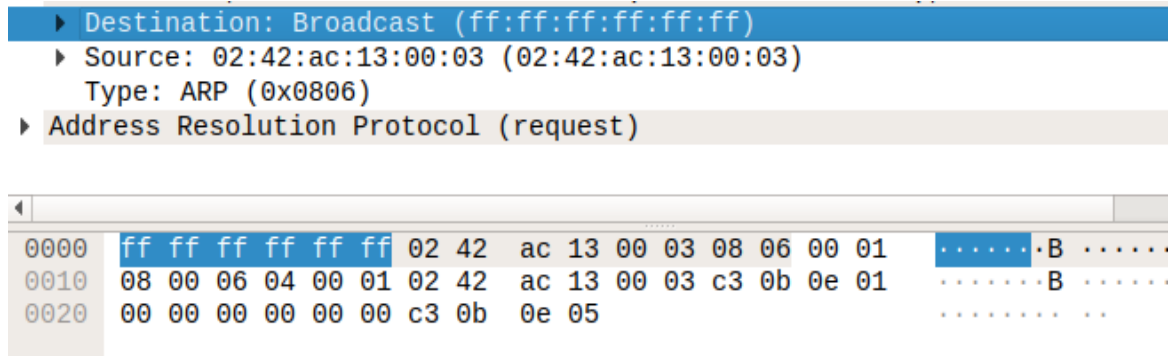
- Máy pc3
- Địa chỉ IP và địa chỉ MAC của máy nhận dữ liệu? Đây là địa chỉ IP và MAC của máy tính nào trong mạng? Nhận xét về cặp địa chỉ IP và MAC của máy nhận dữ liệu.

Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)															
Target IP address: 195.11.14.5															
0000	ff	ff	ff	ff	ff	ff	02	42	ac	13	00	03	08	06	00 01
0010	08	00	06	04	00	01	02	42	ac	13	00	03	c3	0b	0e 01
0020	00	00	00	00	00	00	c3	0b	0e	05					

- Máy pc1
- ❖ Chọn **Header Ethernet II** và cho biết:
 - Địa chỉ MAC của máy gửi dữ liệu là bao nhiêu? Địa chỉ MAC này là của máy tính nào trong mạng?
 - ▼ Ethernet II, Src: 02:42:ac:13:00:03 (02:42:ac:13:00:03), Dst: Broadcast
 - Destination: Broadcast (ff:ff:ff:ff:ff:ff)
 - Source: 02:42:ac:13:00:03 (02:42:ac:13:00:03)
 - Type: ARP (0x0806)
 - Address Resolution Protocol (request)

0000	ff	ff	ff	ff	ff	ff	02	42	ac	13	00	03	08	06	00 01
0010	08	00	06	04	00	01	02	42	ac	13	00	03	c3	0b	0e 01

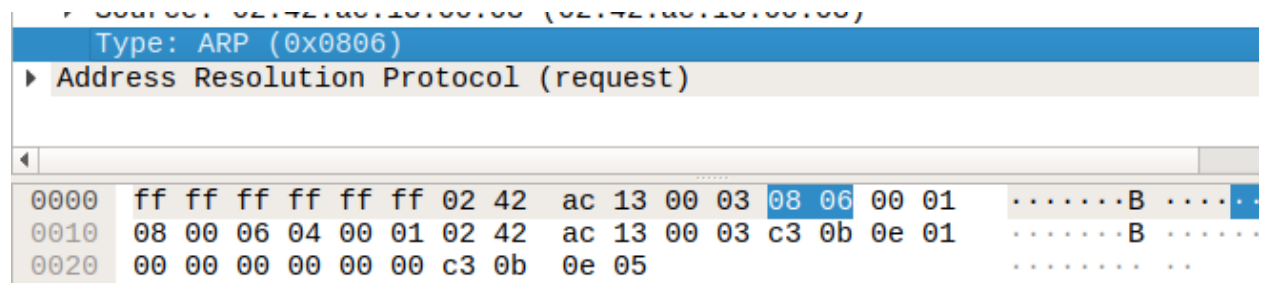
- Máy pc3
- Địa chỉ MAC của máy nhận dữ liệu là bao nhiêu? Địa chỉ MAC này là của máy tính nào trong mạng? Nhận xét về địa chỉ MAC này và địa chỉ MAC của máy nhận dữ liệu đã quan sát được ở phần Header Address Resolution Protocol



Pc1

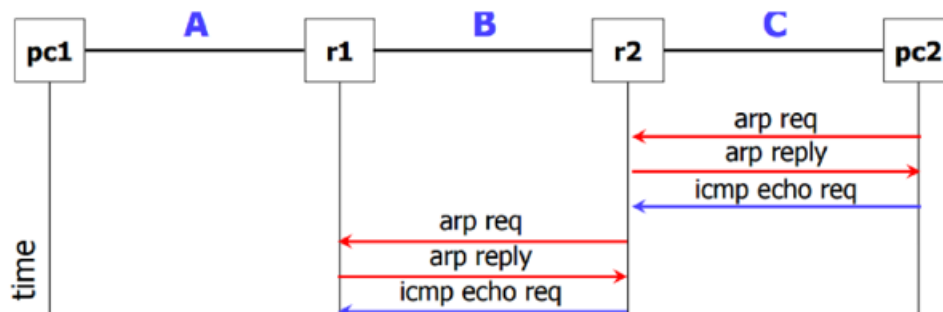
Đây là địa chỉ broadcast

- Trường Type mang giá trị (hexadecimal) bằng bao nhiêu? Thông tin thể hiện là gì?



08 06, thể hiện đây là giao thức ARP

- ❖ Vẽ sơ đồ tuần tự (sequence diagram) thể hiện vai trò của giao thức ARP trong việc truyền tải dữ liệu từ pc3 đến pc1 bằng lệnh ping.



15) Hủy mạng ảo, sau khi đã thực hiện xong 2.3.2.2

```

b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap6$ kathara wipe
Are you sure to wipe Kathara? (y/n) y
    
```

2.3.2.3. Giao thức ARP với địa chỉ không nằm trong bảng vạch đường

- Trên **pc1**, gửi dữ liệu đến **Google DNS** (địa chỉ ngoài mạng ảo) bằng lệnh:

\$ ping 8.8.8.8

```

root@pc1: /
++ ifconfig eth0 195.11.14.5/24 up
++ route add default gw 195.11.14.1
--- End Startup Commands Log

root@pc1:/# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 195.11.14.1 icmp_seq=1 Destination Net Unreachable
From 195.11.14.1 icmp_seq=2 Destination Net Unreachable
From 195.11.14.1 icmp_seq=3 Destination Net Unreachable
From 195.11.14.1 icmp_seq=4 Destination Net Unreachable
From 195.11.14.1 icmp_seq=5 Destination Net Unreachable
From 195.11.14.1 icmp_seq=6 Destination Net Unreachable
From 195.11.14.1 icmp_seq=7 Destination Net Unreachable
From 195.11.14.1 icmp_seq=8 Destination Net Unreachable
From 195.11.14.1 icmp_seq=9 Destination Net Unreachable
^C
--- 8.8.8.8 ping statistics ---
9 packets transmitted, 0 received, +9 errors, 100% packet loss, time 8184ms

```

- Ghi nhận kết quả hiển thị khi dùng lệnh **arp** trên **pc1** và **router1**. Nhận xét kết quả này.

```

root@pc1:/# arp
Address          Hwtype  Hwaddress      Flags Mask      Iface
195.11.14.1      ether   4a:ae:4d:c8:7f:4c  C              eth0
root@pc1:/# █

root@router1:/# arp
Address          Hwtype  Hwaddress      Flags Mask      Iface
195.11.14.5      ether   de:08:0d:2a:a2:63  C              eth0
root@router1:/# █

```

2.3.2.4. Giao thức ARP với địa chỉ thuộc nhánh mạng LAN không sử dụng

- Trên **pc1**, gửi dữ liệu đến 195.11.14.200 (địa chỉ không tồn tại có thể được cấp phát trong nhánh LAN A) bằng lệnh:

\$ ping 195.11.14.200

```

root@pc1: /
root@pc1:/# ping 195.11.14.200
PING 195.11.14.200 (195.11.14.200) 56(84) bytes of data.
From 195.11.14.5 icmp_seq=1 Destination Host Unreachable
From 195.11.14.5 icmp_seq=2 Destination Host Unreachable
From 195.11.14.5 icmp_seq=3 Destination Host Unreachable
From 195.11.14.5 icmp_seq=4 Destination Host Unreachable
From 195.11.14.5 icmp_seq=5 Destination Host Unreachable
From 195.11.14.5 icmp_seq=6 Destination Host Unreachable
^C
--- 195.11.14.200 ping statistics ---
9 packets transmitted, 0 received, +6 errors, 100% packet loss, time 8197ms
pipe 4

```

- Ghi nhận kết quả hiển thị khi dùng lệnh **arp** trên **pc1** và **router1**. Nhận xét kết quả này.

```

root@pc1:/# arp
Address          Hwtype  Hwaddress      Flags Mask      Iface
195.11.14.1      ether   4a:ae:4d:c8:7f:4c  C              eth0
195.11.14.200    ether   (incomplete)      C              eth0
root@pc1:/# █

root@router1:/# arp
Address          Hwtype  Hwaddress      Flags Mask      Iface
195.11.14.5      ether   de:08:0d:2a:a2:63  C              eth0
root@router1:/# █

```

Do không tồn tại trong mạng LAN

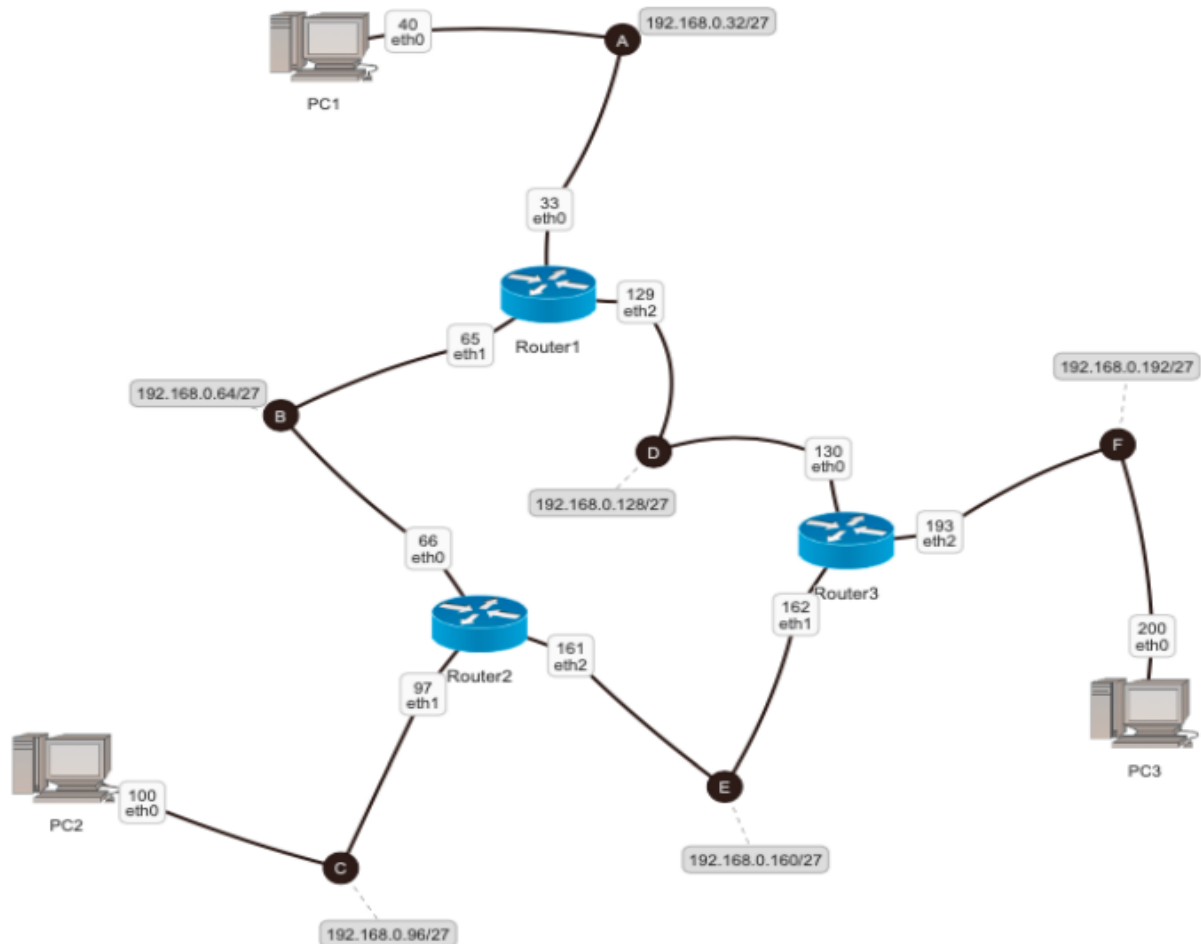
16) Hủy mạng ảo bằng lệnh `lwipe` sau khi đã thực hiện xong Bài tập 6.

```
$ kathara wipe
```

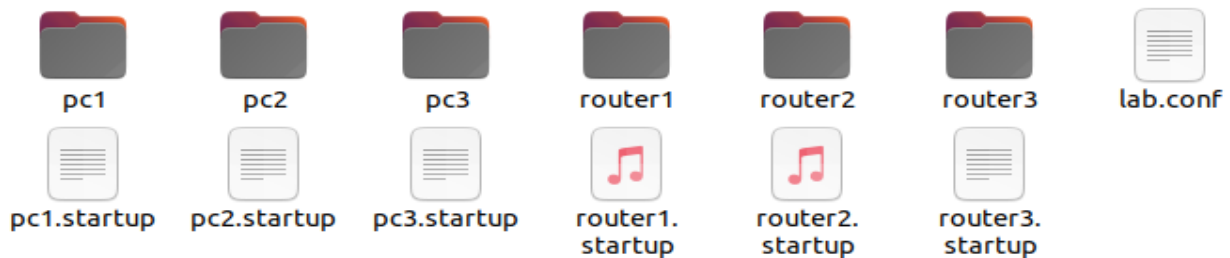
```
b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap6$ kathara wipe  
Are you sure to wipe Kathara? (y/n) y
```

Bài 7:

- 1) Quan sát mô hình mạng cần xây dựng. Nhận diện các thiết bị (PC, Router...), giao diện (eth0, eth1...) với các địa chỉ IP được gán.



2) Tạo thư mục BaiTap7 trong workspace của sinh viên.



3) Soạn thảo nội dung mô tả hình thái mạng theo thiết kế trên file **lab.conf**

```
GNU nano 4.8 lab.conf Modified
pc1[0]=A
pc2[0]=C
pc3[0]=F

router1[0]=A
router1[1]=B
router1[2]=D

router2[0]=B
router2[1]=C
router2[2]=E

router3[0]=D
router3[1]=E
router3[2]=F
```

4) Đối với các file **pc1.startup**, **pc2.startup** và **pc3.startup**: thực hiện vạch đường mặc nhiên thông qua các Router tương ứng trong nhánh mạng.

```
GNU nano 4.8 pc1.startup Modified
ifconfig eth0 192.168.0.40/27 up
route add default gw 192.168.0.33
```

```
GNU nano 4.8 pc2.startup Modified
ifconfig eth0 192.168.0.100/27 up
route add default gw 192.168.0.97
```

```
GNU nano 4.8 pc3.startup Modified
ifconfig eth0 192.168.0.200/27 up
route add default gw 192.168.0.193
```

5) Đối với các file **router1.startup**, **router2.startup** và **router3.startup**: thực hiện vạch đường tĩnh và vạch đường mặc nhiên (nếu cần)

```
GNU nano 4.8 router1.startup Modified
ifconfig eth0 192.168.0.33/27 up
ifconfig eth1 192.168.0.65/27 up
ifconfig eth2 192.168.0.129/27 up

route add -net 192.168.0.96/27 gw 192.168.0.66
route add -net 192.168.0.192/27 gw 192.168.0.130
```

```
GNU nano 4.8 router2.startup Modified
ifconfig eth0 192.168.0.66/27 up
ifconfig eth1 192.168.0.97/27 up
ifconfig eth2 192.168.0.161/27 up

route add -net 192.168.0.32/27 gw 192.168.0.65
route add -net 192.168.0.192/27 gw 192.168.0.162
```

```
GNU nano 4.8 router3.startup Modified
ifconfig eth0 192.168.0.130/27 up
ifconfig eth1 192.168.0.162/27 up
ifconfig eth2 192.168.0.193/27 up

route add -net 192.168.0.32/27 gw 192.168.0.129
route add -net 192.168.0.96/27 gw 192.168.0.161
```

- 6) Khởi động mạng ảo BaiTap7. Kiểm tra bảng vạch đường (bằng route) và địa chỉ IP của các giao diện mạng (bằng ifconfig) trên từng máy ảo để đảm bảo tính đúng đắn của mô hình mạng Bài Tập 7. Kiểm tra tính liên thông giữa pc1, pc2 và pc3 trong mạng (bằng ping).

```
root@pc1: /
--- Startup Commands Log
++ ifconfig eth0 192.168.0.40/27 up
++ route add default gw 192.168.0.33
--- End Startup Commands Log
root@pc1: /#
```

```
root@pc2: /
--- Startup Commands Log
++ ifconfig eth0 192.168.0.100/27 up
++ route add default gw 192.168.0.97
--- End Startup Commands Log
```

```
root@pc3: /
--- Startup Commands Log
++ ifconfig eth0 192.168.0.200/27 up
++ route add default gw 192.168.0.193
--- End Startup Commands Log
```

```
root@router3: /
--- Startup Commands Log
++ ifconfig eth0 192.168.0.130/27 up
++ ifconfig eth1 192.168.0.162/27 up
++ ifconfig eth2 192.168.0.193/27 up
++ route add -net 192.168.0.32/27 gw 192.168.0.129
++ route add -net 192.168.0.96/27 gw 192.168.0.161
--- End Startup Commands Log
```

```
root@router1: /  
--- Startup Commands Log  
++ ifconfig eth0 192.168.0.33/27 up  
++ ifconfig eth1 192.168.0.65/27 up  
++ ifconfig eth2 192.168.0.129/27 up  
++ route add -net 192.168.0.96/27 gw 192.168.0.66  
++ route add -net 192.168.0.192/27 gw 192.168.0.130  
--- End Startup Commands Log  
  
root@router2: /  
--- Startup Commands Log  
++ ifconfig eth0 192.168.0.66/27 up  
++ ifconfig eth1 192.168.0.97/27 up  
++ ifconfig eth2 192.168.0.161/27 up  
++ route add -net 192.168.0.32/27 gw 192.168.0.65  
++ route add -net 192.168.0.192/27 gw 192.168.0.162  
--- End Startup Commands Log
```

Từ pc1 ping đến pc2, pc3

\$ ping 192.168.0.100 (địa chỉ IP của pc2)

\$ ping 192.168.0.200 (địa chỉ IP của pc3)

```
root@pc1:/# ping 192.168.0.100  
PING 192.168.0.100 (192.168.0.100) 56(84) bytes of data.  
64 bytes from 192.168.0.100: icmp_seq=1 ttl=62 time=0.140 ms  
64 bytes from 192.168.0.100: icmp_seq=2 ttl=62 time=0.113 ms  
64 bytes from 192.168.0.100: icmp_seq=3 ttl=62 time=0.086 ms  
64 bytes from 192.168.0.100: icmp_seq=4 ttl=62 time=0.067 ms  
64 bytes from 192.168.0.100: icmp_seq=5 ttl=62 time=0.109 ms  
64 bytes from 192.168.0.100: icmp_seq=6 ttl=62 time=0.130 ms  
64 bytes from 192.168.0.100: icmp_seq=7 ttl=62 time=0.111 ms  
64 bytes from 192.168.0.100: icmp_seq=8 ttl=62 time=0.123 ms  
64 bytes from 192.168.0.100: icmp_seq=9 ttl=62 time=0.112 ms  
64 bytes from 192.168.0.100: icmp_seq=10 ttl=62 time=0.115 ms  
^C  
--- 192.168.0.100 ping statistics ---  
10 packets transmitted, 10 received, 0% packet loss, time 9202ms  
rtt min/avg/max/mdev = 0.067/0.110/0.140/0.022 ms  
root@pc1:/# ping 192.168.0.200  
PING 192.168.0.200 (192.168.0.200) 56(84) bytes of data.  
64 bytes from 192.168.0.200: icmp_seq=1 ttl=62 time=0.100 ms  
64 bytes from 192.168.0.200: icmp_seq=2 ttl=62 time=0.110 ms  
64 bytes from 192.168.0.200: icmp_seq=3 ttl=62 time=0.107 ms  
64 bytes from 192.168.0.200: icmp_seq=4 ttl=62 time=0.104 ms  
64 bytes from 192.168.0.200: icmp_seq=5 ttl=62 time=0.115 ms  
64 bytes from 192.168.0.200: icmp_seq=6 ttl=62 time=0.110 ms  
64 bytes from 192.168.0.200: icmp_seq=7 ttl=62 time=0.115 ms  
^C  
--- 192.168.0.200 ping statistics ---  
7 packets transmitted, 7 received, 0% packet loss, time 6128ms  
rtt min/avg/max/mdev = 0.100/0.108/0.115/0.013 ms  
root@pc1:/#
```

Từ pc2 ping đến pc1, pc3

\$ ping 192.168.0.40 (địa chỉ IP của pc1)

\$ ping 192.168.0.200 (địa chỉ IP của pc3)

```
root@pc2:/# ping 192.168.0.40
bash: ping: command not found
root@pc2:/# ping 192.168.0.40
PING 192.168.0.40 (192.168.0.40) 56(84) bytes of data.
64 bytes from 192.168.0.40: icmp_seq=1 ttl=62 time=0.076 ms
64 bytes from 192.168.0.40: icmp_seq=2 ttl=62 time=0.108 ms
64 bytes from 192.168.0.40: icmp_seq=3 ttl=62 time=0.091 ms
64 bytes from 192.168.0.40: icmp_seq=4 ttl=62 time=0.110 ms
64 bytes from 192.168.0.40: icmp_seq=5 ttl=62 time=0.110 ms
64 bytes from 192.168.0.40: icmp_seq=6 ttl=62 time=0.115 ms
64 bytes from 192.168.0.40: icmp_seq=7 ttl=62 time=0.107 ms
^C
--- 192.168.0.40 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6142ms
rtt min/avg/max/mdev = 0.076/0.102/0.115/0.015 ms
root@pc2:/# ping 192.168.0.200
PING 192.168.0.200 (192.168.0.200) 56(84) bytes of data.
64 bytes from 192.168.0.200: icmp_seq=1 ttl=62 time=0.075 ms
64 bytes from 192.168.0.200: icmp_seq=2 ttl=62 time=0.117 ms
64 bytes from 192.168.0.200: icmp_seq=3 ttl=62 time=0.107 ms
64 bytes from 192.168.0.200: icmp_seq=4 ttl=62 time=0.122 ms
64 bytes from 192.168.0.200: icmp_seq=5 ttl=62 time=0.109 ms
64 bytes from 192.168.0.200: icmp_seq=6 ttl=62 time=0.113 ms
64 bytes from 192.168.0.200: icmp_seq=7 ttl=62 time=0.107 ms
^C
--- 192.168.0.200 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6147ms
rtt min/avg/max/mdev = 0.075/0.107/0.122/0.015 ms
root@pc2:/#
```

Từ pc3 ping đến pc1, pc2

\$ ping 192.168.0.40 (địa chỉ IP của pc1)

\$ ping 192.168.0.100 (địa chỉ IP của pc2)

```
root@pc3:/# ping 192.168.0.40
PING 192.168.0.40 (192.168.0.40) 56(84) bytes of data.
64 bytes from 192.168.0.40: icmp_seq=1 ttl=62 time=0.074 ms
64 bytes from 192.168.0.40: icmp_seq=2 ttl=62 time=0.076 ms
64 bytes from 192.168.0.40: icmp_seq=3 ttl=62 time=0.128 ms
64 bytes from 192.168.0.40: icmp_seq=4 ttl=62 time=0.114 ms
64 bytes from 192.168.0.40: icmp_seq=5 ttl=62 time=0.213 ms
64 bytes from 192.168.0.40: icmp_seq=6 ttl=62 time=0.153 ms
64 bytes from 192.168.0.40: icmp_seq=7 ttl=62 time=0.120 ms
64 bytes from 192.168.0.40: icmp_seq=8 ttl=62 time=0.062 ms
^C
--- 192.168.0.40 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7168ms
rtt min/avg/max/mdev = 0.062/0.117/0.213/0.047 ms
root@pc3:/# ping 192.168.0.100
PING 192.168.0.100 (192.168.0.100) 56(84) bytes of data.
64 bytes from 192.168.0.100: icmp_seq=1 ttl=62 time=0.089 ms
64 bytes from 192.168.0.100: icmp_seq=2 ttl=62 time=0.115 ms
64 bytes from 192.168.0.100: icmp_seq=3 ttl=62 time=0.112 ms
64 bytes from 192.168.0.100: icmp_seq=4 ttl=62 time=0.105 ms
64 bytes from 192.168.0.100: icmp_seq=5 ttl=62 time=0.083 ms
64 bytes from 192.168.0.100: icmp_seq=6 ttl=62 time=0.119 ms
64 bytes from 192.168.0.100: icmp_seq=7 ttl=62 time=0.105 ms
^C
--- 192.168.0.100 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6145ms
rtt min/avg/max/mdev = 0.083/0.104/0.119/0.012 ms
root@pc3:/#
```


7) Hủy mạng ảo bằng lệnh lwipe sau khi đã thực hiện xong Bài tập 7

```
b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap7$ kathara wipe  
Are you sure to wipe Kathara? (y/n) y
```