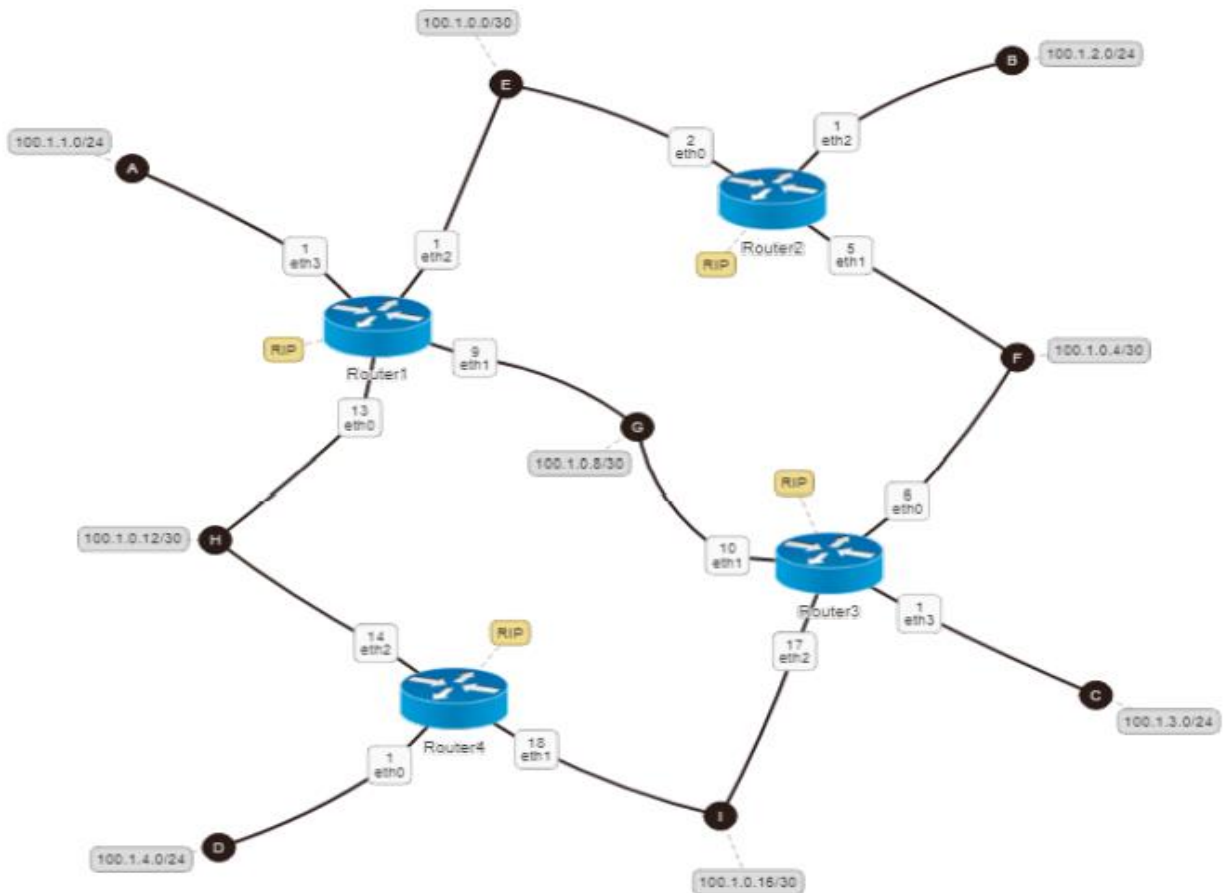


LAB 4

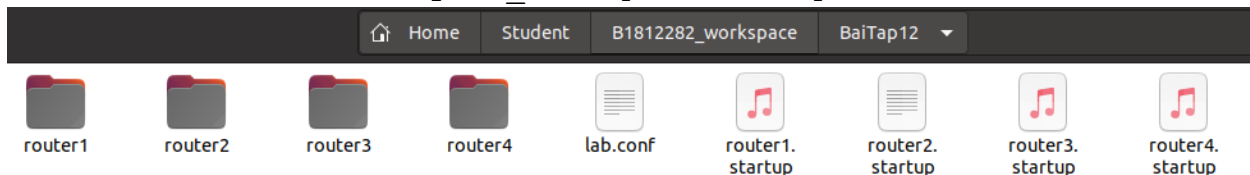
Bài tập 12:

- 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 BaiTap12 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 BaiTap12 bằng lệnh:

```
$ cd /home/student/your_workspace/BaiTap12
```



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

```
GNU nano 4.8 lab.conf Modified
router1[0]=H
router1[1]=G
router1[2]=E
router1[3]=A

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

router3[0]=F
router3[1]=G
router3[2]=I
router3[3]=C

router4[0]=D
router4[1]=I
router4[2]=H
```

4) Trên các file .startup của các router, soạn thảo nội dung cấu hình cho giao diện mạng của chúng. Nội dung **router1.startup** tham khảo:

```
GNU nano 4.8 router1.startup Modified
ifconfig eth0 100.1.0.13/30 up
ifconfig eth1 100.1.0.9/30 up
ifconfig eth2 100.1.0.1/30 up
ifconfig eth3 100.1.1.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router2.startup
ifconfig eth0 100.1.0.2/30 up
ifconfig eth1 100.1.0.5/30 up
ifconfig eth2 100.1.2.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router3.startup Modified
ifconfig eth0 100.1.0.6/30 up
ifconfig eth1 100.1.0.10/30 up
ifconfig eth2 100.1.0.17/30 up
ifconfig eth3 100.1.3.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router4.startup Modified
ifconfig eth0 100.1.4.1/24 up
ifconfig eth1 100.1.0.18/30 up
ifconfig eth2 100.1.0.14/30 up

/etc/init.d/quagga start
```

Lệnh **/etc/init.d/quagga start** sẽ khởi động dịch vụ *Quagga* trên máy ảo. Để giao thức RIPv2 thực thi trên dịch vụ Quagga của máy ảo thì cần phải cấu hình cho giao thức RIPv2 theo hướng dẫn ở 5)

5) Trong mỗi thư mục router đã tạo ra, ví dụ: trong router1, tạo cấu trúc thư mục giống như hình 4.4 dưới đây:

❖ Trong thư mục **/etc/quagga** chứa 3 files với ý nghĩa như sau:

- **zebra.conf** chứa các nội dung miêu tả thiết lập cho công cụ *Quagga* trên máy ảo. Ví dụ:
 - ✓ Username và password để sử dụng công cụ *Quagga*
 - ✓ Đường dẫn chứa nhật ký hoạt động của Quagga (log file)
- **daemons**: là nơi khai báo giao thức vạch đường nào sẽ được kích hoạt khi Quagga khởi động và giao thức sẽ không được kích hoạt. Một số giao thức được hỗ trợ bởi Quagga có thể kể đến như:
 - ✓ RIPv2 trên IPv4: đặt tên là ripd
 - ✓ OSPFv2 trên IPv4: đặt tên là ospfd
 - ✓ BGP trên IPv4 và IPv6: đặt tên là bgpd
- **ripd.conf**: chứa nội dung miêu tả hoạt động của giải thuật RIPv2 trên IPv4 trên Quagga.

6) Nếu sử dụng các thiết lập mặc nhiên của Quagga thì bỏ qua 6). Nếu không thì có thể sử dụng nội dung được miêu tả tham khảo sau dành cho file **zebra.conf**:

```
GNU nano 4.8                                zebra.conf                                Modified
hostname zebra
password zebra
enable password zebra

log file /var/log/zebra/zebra.log
```

7) Trên file **daemons**, thêm vào nội dung khai báo giao thức vạch đường sẽ sử dụng trên router. yes là sử dụng loại giao thức đó, no là không sử dụng, mặc nhiên là no. Nội dung file **daemons** có thể tham khảo như sau:

```
GNU nano 4.8                                daemons                                Modified
zebra=yes
ripd=yes

bgpd=no
ospfd=no
```

8) Trên file ripd.conf, thêm vào nội dung miêu tả hoạt động của giao thức RIPv2 trên Quagga của router. Nội dung file ripd.conf có thể tham khảo như sau:

```
GNU nano 4.8 ripd.conf
hostname ripd
password zebra
enable password zebra

router rip
network 100.1.0.0/16
redistribute connected

log file /var/log/zebra/ripd.log
```

- Lệnh router rip: chỉ ra router sẽ sử dụng RIPv2 khi vạch đường.
- Lệnh network 100.1.0.0/16: phạm vi Gửi và Nhận gói tin vạch đường RIPv2 của router
 - ✓ Phạm vi này có thể là mạng con (subnet) mà router thuộc về. Ví dụ: 100.1.0.0/24
 - ✓ Phạm vi này cũng có thể là mạng lớn chứa tất cả các router trong AS (địa chỉ mạng đại diện cho AS). Ví dụ: 100.1.0.0/16
 - ✓ Ngoài phạm vi này, router sẽ không gửi gói tin vạch đường RIPv2.
 - ✓ Câu hỏi: Tại sao trong Bài tập 12 này, địa chỉ mạng được sử dụng cho lệnh network trong file ripd.conf là 100.1.0.0/16 mà không phải là địa chỉ 100.1.0.0/24? Liệu có thể sử dụng địa chỉ 100.1.0.0/20 được hay không?
- Lệnh redistribute connected: cho phép router đóng gói các thông tin về những nhánh mạng có kết nối trực tiếp với nó thành gói tin RIPv2 và phân phối lại (redistribute) trên các ngã ra có sử dụng giao thức RIPv2 đã khai báo trước đó qua lệnh network
 - ✓ Ví dụ: router1 sẽ phân phối lại hiểu biết của nó về nhánh mạng A, E, G và H (có kết nối trực tiếp) trên tất cả kết nối của router1 thuộc 100.1.0.0/16
- Ngoài các lệnh đơn giản được sử dụng và hướng dẫn trong phần thực hành Mạng máy tính CT112 thì còn rất nhiều lệnh với các cách sử dụng khác nhau. Sinh viên có thể tham khảo thêm tại Zebra GNU Manual1.

- 9) Khởi động mạng ảo BaiTap12. Trên các router, kiểm tra dịch vụ Quagga đã được bật lên và giao thức RIPv2 đã hoạt động hay chưa bằng lệnh:

```
$ kathara lstart
```

```
root@router1: /
--- Startup Commands Log
++ ifconfig eth0 100.1.0.13/30 up
++ ifconfig eth1 100.1.0.9/30 up
++ ifconfig eth2 100.1.0.1/30 up
++ ifconfig eth3 100.1.1.1/24 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@router2: /
--- Startup Commands Log
++ ifconfig eth0 100.1.0.2/30 up
++ ifconfig eth1 100.1.0.5/30 up
++ ifconfig eth2 100.1.2.1/24 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@router3: /
--- Startup Commands Log
++ ifconfig eth0 100.1.0.6/30 up
++ ifconfig eth1 100.1.0.10/30 up
++ ifconfig eth2 100.1.0.17/30 up
++ ifconfig eth3 100.1.3.1/24 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@router4: /
--- Startup Commands Log
++ ifconfig eth0 100.1.4.1/24 up
++ ifconfig eth1 100.1.0.18/30 up
++ ifconfig eth2 100.1.0.14/30 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log
```

```
$ /etc/init.d/quagga status
```

```
root@router1:/# /etc/init.d/quagga status
ripd watchquagga zebra
root@router1:/# █

root@router2:/# /etc/init.d/quagga status
ripd watchquagga zebra
root@router2:/# █
```

```

root@router3:/# /etc/init.d/quagga status
ripd watchquagga zebra
root@router3:/# █

root@router4:/# /etc/init.d/quagga status
ripd watchquagga zebra
root@router4:/# █

```

Nếu kết quả nhận được là:

- Không hiển thị gì cả → Dịch vụ Quagga chưa được khởi động.
- **ripd watchquagga zebra** → Mọi thiết lập cho RIPv2 trên Quagga đều chính xác.
- watchquagga → có sai sót trên **ripd.conf** hoặc **daemons**. Cần tắt và khởi động lại router cấu hình RIPv2 sai sót.
- ⇒ Thành công

10) Dợi 10 giây sau khi mạng ảo BaiTap12 đã khởi động xong

- Kiểm tra bảng vạch đường trên các router bằng: **route**. Nhận xét kết quả.

```

$ route

root@router1:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 0.0.0.0 255.255.255.252 U 0 0 0 eth2
100.1.0.4 100.1.0.2 255.255.255.252 UG 20 0 0 eth2
100.1.0.8 0.0.0.0 255.255.255.252 U 0 0 0 eth1
100.1.0.12 0.0.0.0 255.255.255.252 U 0 0 0 eth0
100.1.0.16 100.1.0.10 255.255.255.252 UG 20 0 0 eth1
100.1.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
100.1.2.0 100.1.0.2 255.255.255.0 UG 20 0 0 eth2
100.1.3.0 100.1.0.10 255.255.255.0 UG 20 0 0 eth1
100.1.4.0 100.1.0.14 255.255.255.0 UG 20 0 0 eth0
root@router1:/# █

root@router2:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 0.0.0.0 255.255.255.252 U 0 0 0 eth0
100.1.0.4 0.0.0.0 255.255.255.252 U 0 0 0 eth1
100.1.0.8 100.1.0.1 255.255.255.252 UG 20 0 0 eth0
100.1.0.12 100.1.0.1 255.255.255.252 UG 20 0 0 eth0
100.1.0.16 100.1.0.6 255.255.255.252 UG 20 0 0 eth1
100.1.1.0 100.1.0.1 255.255.255.0 UG 20 0 0 eth0
100.1.2.0 0.0.0.0 255.255.255.0 U 0 0 0 eth2
100.1.3.0 100.1.0.6 255.255.255.0 UG 20 0 0 eth1
100.1.4.0 100.1.0.6 255.255.255.0 UG 20 0 0 eth1
root@router2:/# █

root@router3:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 100.1.0.5 255.255.255.252 UG 20 0 0 eth0
100.1.0.4 0.0.0.0 255.255.255.252 U 0 0 0 eth0
100.1.0.8 0.0.0.0 255.255.255.252 U 0 0 0 eth1
100.1.0.12 100.1.0.9 255.255.255.252 UG 20 0 0 eth1
100.1.0.16 0.0.0.0 255.255.255.252 U 0 0 0 eth2
100.1.1.0 100.1.0.9 255.255.255.0 UG 20 0 0 eth1
100.1.2.0 100.1.0.5 255.255.255.0 UG 20 0 0 eth0
100.1.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
100.1.4.0 100.1.0.18 255.255.255.0 UG 20 0 0 eth2
root@router3:/# █

root@router4:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 100.1.0.13 255.255.255.252 UG 20 0 0 eth2
100.1.0.4 100.1.0.17 255.255.255.252 UG 20 0 0 eth1
100.1.0.8 100.1.0.17 255.255.255.252 UG 20 0 0 eth1
100.1.0.12 0.0.0.0 255.255.255.252 U 0 0 0 eth2
100.1.0.16 0.0.0.0 255.255.255.252 U 0 0 0 eth1
100.1.1.0 100.1.0.13 255.255.255.0 UG 20 0 0 eth2
100.1.2.0 100.1.0.17 255.255.255.0 UG 20 0 0 eth1
100.1.3.0 100.1.0.17 255.255.255.0 UG 20 0 0 eth1
100.1.4.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
root@router4:/# █

```

- Thực hiện ping giữa các router để kiểm tra tính liên thông của các mạng LAN trong mô hình BaiTap12. Ví dụ:

✓ Từ router1 đến router4:

```
$ ping 100.1.4.1
```

```
root@router1:/# ping 100.1.4.1
PING 100.1.4.1 (100.1.4.1) 56(84) bytes of data.
64 bytes from 100.1.4.1: icmp_seq=1 ttl=64 time=0.101 ms
64 bytes from 100.1.4.1: icmp_seq=2 ttl=64 time=0.081 ms
64 bytes from 100.1.4.1: icmp_seq=3 ttl=64 time=0.085 ms
64 bytes from 100.1.4.1: icmp_seq=4 ttl=64 time=0.082 ms
64 bytes from 100.1.4.1: icmp_seq=5 ttl=64 time=0.082 ms
64 bytes from 100.1.4.1: icmp_seq=6 ttl=64 time=0.078 ms
64 bytes from 100.1.4.1: icmp_seq=7 ttl=64 time=0.075 ms
64 bytes from 100.1.4.1: icmp_seq=8 ttl=64 time=0.081 ms
64 bytes from 100.1.4.1: icmp_seq=9 ttl=64 time=0.075 ms
64 bytes from 100.1.4.1: icmp_seq=10 ttl=64 time=0.081 ms
64 bytes from 100.1.4.1: icmp_seq=11 ttl=64 time=0.083 ms
64 bytes from 100.1.4.1: icmp_seq=12 ttl=64 time=0.087 ms
64 bytes from 100.1.4.1: icmp_seq=13 ttl=64 time=0.075 ms
64 bytes from 100.1.4.1: icmp_seq=14 ttl=64 time=0.084 ms
^C
--- 100.1.4.1 ping statistics ---
14 packets transmitted, 14 received, 0% packet loss, time 13291ms
rtt min/avg/max/mdev = 0.075/0.082/0.101/0.008 ms
```

✓ Từ router2 đến router3:

```
$ ping 100.1.0.10
```

```
root@router2:/# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data.
64 bytes from 100.1.0.10: icmp_seq=1 ttl=64 time=0.106 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=64 time=0.091 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=64 time=0.098 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=64 time=0.096 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=64 time=0.096 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=64 time=0.107 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=64 time=0.096 ms
64 bytes from 100.1.0.10: icmp_seq=8 ttl=64 time=0.134 ms
64 bytes from 100.1.0.10: icmp_seq=9 ttl=64 time=0.058 ms
64 bytes from 100.1.0.10: icmp_seq=10 ttl=64 time=0.100 ms
64 bytes from 100.1.0.10: icmp_seq=11 ttl=64 time=0.071 ms
64 bytes from 100.1.0.10: icmp_seq=12 ttl=64 time=0.057 ms
64 bytes from 100.1.0.10: icmp_seq=13 ttl=64 time=0.112 ms
64 bytes from 100.1.0.10: icmp_seq=14 ttl=64 time=0.076 ms
64 bytes from 100.1.0.10: icmp_seq=15 ttl=64 time=0.073 ms
64 bytes from 100.1.0.10: icmp_seq=16 ttl=64 time=0.105 ms
64 bytes from 100.1.0.10: icmp_seq=17 ttl=64 time=0.117 ms
64 bytes from 100.1.0.10: icmp_seq=18 ttl=64 time=0.080 ms
^C
--- 100.1.0.10 ping statistics ---
18 packets transmitted, 18 received, 0% packet loss, time 17405ms
rtt min/avg/max/mdev = 0.057/0.092/0.134/0.023 ms
```

- 11) Trên router bất kỳ, sử dụng lệnh tcpdump để bắt gói tin RIPv2 mà router trao đổi. Dừng lệnh tcpdump sau 20 giây. Ví dụ minh họa trên router1:

```
$ tcpdump -i any -w /shared/BaiTap12_router1.pcap
```

```
root@router1: /
root@router1:/# tcpdump -i any -w /home/BaiTap12_router1.pcap
tcpdump: listening on any, link-type LINUX_SLL (Linux cooked), capture size 2621
44 bytes
^C20 packets captured
20 packets received by filter
0 packets dropped by kernel
```

12) Trên router bất kỳ, dùng lệnh để đăng nhập vào dịch vụ RIPv2 đang chạy:

```
$ telnet localhost ripd
```

Nhập mật khẩu là zebra đã đặt ở 8)

```
root@router2: /
root@router2:/# telnet localhost ripd
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 1.0.20160315).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
ripd> █
```

13) Tại giao diện của ripd, dùng lệnh:

```
$ show ip rip
```

```
ripd> show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network      Next Hop      Metric From      Tag Time
C(i) 100.1.0.0/30    0.0.0.0        1 self          0
C(i) 100.1.0.4/30    0.0.0.0        1 self          0
R(n) 100.1.0.8/30    100.1.0.1      2 100.1.0.1      0 02:33
R(n) 100.1.0.12/30   100.1.0.1      2 100.1.0.1      0 02:33
R(n) 100.1.0.16/30   100.1.0.6      2 100.1.0.6      0 02:41
R(n) 100.1.1.0/24    100.1.0.1      2 100.1.0.1      0 02:33
C(i) 100.1.2.0/24    0.0.0.0        1 self          0
R(n) 100.1.3.0/24    100.1.0.6      2 100.1.0.6      0 02:41
R(n) 100.1.4.0/24    100.1.0.6      3 100.1.0.6      0 02:41
ripd> █
```

- So sánh kết quả hiển thị của lệnh này với lệnh **route** trên router.
- Chỉ ra một số thông tin có ích trên kết quả hiển thị.

14) Trên máy thực, mở file BT12_router1.pcap bằng Wireshark

- Chọn gói tin RIPv2 có địa chỉ nguồn (source) là 100.1.0.10 (router3) và trả lời các câu hỏi sau:
 - ✓ Địa chỉ IP đích (destination) của khung này là bao nhiêu? Địa chỉ IP này là Broadcast, Unicast hay Multicast?


```

- Internet Protocol Version 4, Src: 100.1.0.10, Dst: 224.0.0.9
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
    Total Length: 172
    Identification: 0xeabf (60095)
  ▸ Flags: 0x4000, Don't fragment
    Fragment offset: 0
    Time to live: 1
    Protocol: UDP (17)
    Header checksum: 0x49ad [validation disabled]
    [Header checksum status: Unverified]
    Source: 100.1.0.10
    Destination: 224.0.0.9

```

Địa chỉ IP đích: 224.0.0.9

Địa chỉ IP này là: Multicast

- ✓ Giao thức sử dụng trên tầng vận chuyển là gì? Giao thức đó hoạt động ở cổng (port) bao nhiêu?

```

- User Datagram Protocol, Src Port: 520, Dst Port: 520
  Source Port: 520
  Destination Port: 520
  Length: 152
  Checksum: 0x44be [unverified]
  [Checksum Status: Unverified]
  [Stream index: 13]
  ▸ [Timestamps]

```

Giao thức UDP

Cổng Port: 520

- ✓ Chọn trường Routing Information Protocol. Trong trường này chứa thông tin của các cặp IP Address – Metric, cặp thông tin này có thể được hiểu như thế nào?

Cặp thông tin này có thể được hiểu như là một chuẩn đo lường. Đường đi càng tối ưu thì giá trị Metric càng nhỏ.

```

- Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  ▸ IP Address: 100.1.0.0, Metric: 2
  ▸ IP Address: 100.1.0.4, Metric: 1
  ▸ IP Address: 100.1.0.12, Metric: 2
  ▸ IP Address: 100.1.0.16, Metric: 1
  ▸ IP Address: 100.1.2.0, Metric: 2
  ▸ IP Address: 100.1.3.0, Metric: 1
  ▸ IP Address: 100.1.4.0, Metric: 2

```

- Ngoài gói tin RIPv2 có địa chỉ nguồn 10.1.0.10 (router3) thì router1 còn nhận dữ liệu từ các địa chỉ của những router nào nữa?

Wireshark packet capture titled "BalTap12_router1.pcap". The packet list shows a series of MDNS queries (No. 1-7) and RIPv2 responses (No. 9-16). The packet details pane shows the selected packet (No. 8) as an MDNS query for PTR records.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::54ce:57ff:fe5... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
2	0.000035	fe80::68e0:cdff:fec... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
3	0.000068	fe80::e8f6:15ff:fe0... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
4	0.000141	fe80::7c5a:c8ff:feb... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
5	0.000350	fe80::b8b1:99ff:fec... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
6	0.000384	fe80::54b8:84ff:fe7... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
7	0.000419	fe80::a455:59ff:fe8... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
8	0.000453	fe80::b8bd:b3ff:fe5... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
9	1.099504	100.1.0.13	224.0.0.9	RIPv2	168	Response
10	1.099649	100.1.0.9	224.0.0.9	RIPv2	168	Response
11	1.099755	100.1.0.1	224.0.0.9	RIPv2	168	Response
12	1.099856	100.1.1.1	224.0.0.9	RIPv2	208	Response
13	1.273773	100.1.0.14	224.0.0.9	RIPv2	128	Response
14	15.134094	100.1.0.10	224.0.0.9	RIPv2	188	Response
15	23.800444	100.1.0.2	224.0.0.9	RIPv2	148	Response
16	28.275021	100.1.0.14	224.0.0.9	RIPv2	128	Response
17	32.028689	fe80::54ce:57ff:fe5... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
18	32.028729	fe80::68e0:cdff:fec... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
19	32.028757	fe80::e8f6:15ff:fe0... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...
20	32.028784	fe80::7c5a:c8ff:feb... ff02::fb		MDNS	182	Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR ...

Từ router2 và router4

15) Trên router1, tắt đi giao diện eth1 bằng lệnh:

```
$ ifconfig eth1 down
root@router1:/# ifconfig eth1 down
```

16) Trên router1, thực hiện gửi dữ liệu đến router3 bằng lệnh

```
$ ping 100.1.0.10
```

➤ Dừng lại sau 7 giây. Có ping thành công hay không?

```
root@router1:/# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data.
^C
--- 100.1.0.10 ping statistics ---
22 packets transmitted, 0 received, 100% packet loss, time 21489ms
```

Không thành công

➤ Kiểm tra bảng vạch đường bằng lệnh route. Thông tin trong bảng có thay đổi gì hay chưa?

```
root@router1:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 0.0.0.0 255.255.255.252 U 0 0 0 eth2
100.1.0.4 100.1.0.2 255.255.255.252 UG 20 0 0 eth2
100.1.0.8 100.1.0.14 255.255.255.252 UG 20 0 0 eth0
100.1.0.12 0.0.0.0 255.255.255.252 U 0 0 0 eth0
100.1.0.16 100.1.0.14 255.255.255.252 UG 20 0 0 eth0
100.1.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
100.1.2.0 100.1.0.2 255.255.255.0 UG 20 0 0 eth2
100.1.3.0 100.1.0.14 255.255.255.0 UG 20 0 0 eth0
100.1.4.0 100.1.0.14 255.255.255.0 UG 20 0 0 eth0
root@router1:/#
```

17) Sau 30 giây, thực hiện lại lệnh:

```
$ ping 100.1.0.10
```

➤ Dừng lại sau 7 giây. Có ping thành công hay không?

```

root@router1:/# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data.
64 bytes from 100.1.0.10: icmp_seq=1 ttl=63 time=0.065 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=63 time=0.104 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=63 time=0.099 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=63 time=0.104 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=63 time=0.107 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=63 time=0.105 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=63 time=0.097 ms
^C
--- 100.1.0.10 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6150ms
rtt min/avg/max/mdev = 0.065/0.097/0.107/0.015 ms

```

- Kiểm tra bảng vạch đường bằng lệnh route. Thông tin trong bảng có thay đổi gì hay chưa? Chỉ ra thay đổi đó. Gợi ý: thông tin đường đi từ router1 đến router3 qua G được tự động cập nhật bằng 1 đường đi mới.

```

root@router1:/# route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0      0.0.0.0         255.255.255.252 U           0      0      0 eth2
100.1.0.4      100.1.0.2       255.255.255.252 UG          20      0      0 eth2
100.1.0.8      100.1.0.14      255.255.255.252 UG          20      0      0 eth0
100.1.0.12     0.0.0.0         255.255.255.252 U           0      0      0 eth0
100.1.0.16     100.1.0.14      255.255.255.252 UG          20      0      0 eth0
100.1.1.0      0.0.0.0         255.255.255.0   U           0      0      0 eth3
100.1.2.0      100.1.0.2       255.255.255.0   UG          20      0      0 eth2
100.1.3.0      100.1.0.14      255.255.255.0   UG          20      0      0 eth0
100.1.4.0      100.1.0.14      255.255.255.0   UG          20      0      0 eth0

```

18) Kết luận về hoạt động vạch đường bằng giải thuật RIPv2 trên router.

- Về gói tin vạch đường RIPv2.
- Đơn vị đo khoảng cách dùng trong vạch đường là hops.
 - Số lượng số bước nhảy tối đa là 15.
 - Chu kỳ cập nhật bảng vạch đường là 30 giây và có thể điều chỉnh lại.
 - Tầng vận chuyển sử dụng giao thức UDP với địa chỉ (port) là 520.
 - Hỗ trợ vạch đường liên miền không phân lớp CIDR.
 - Hỗ trợ cơ chế xác thực tính toàn vẹn MD5.
 - Phù hợp triển khai trên AS có kích thước nhỏ và không quá phức tạp.
 - Gói tin trao đổi thông tin vạch đường được gọi là gói tin RIPv2.
- Về chi phí (metric) sử dụng để tính toán đường đi.
- Tùy vào số lượng nhánh mạng
- Về cơ chế tự động cập nhật đường đi mới khi hình trạng mạng thay đổi
- Chu kỳ cập nhật bảng vạch đường là 30 giây và có thể điều chỉnh lại.

19) Hủy mạng ảo, sau khi đã thực hiện xong Bài tập 12

\$ kathara wipe

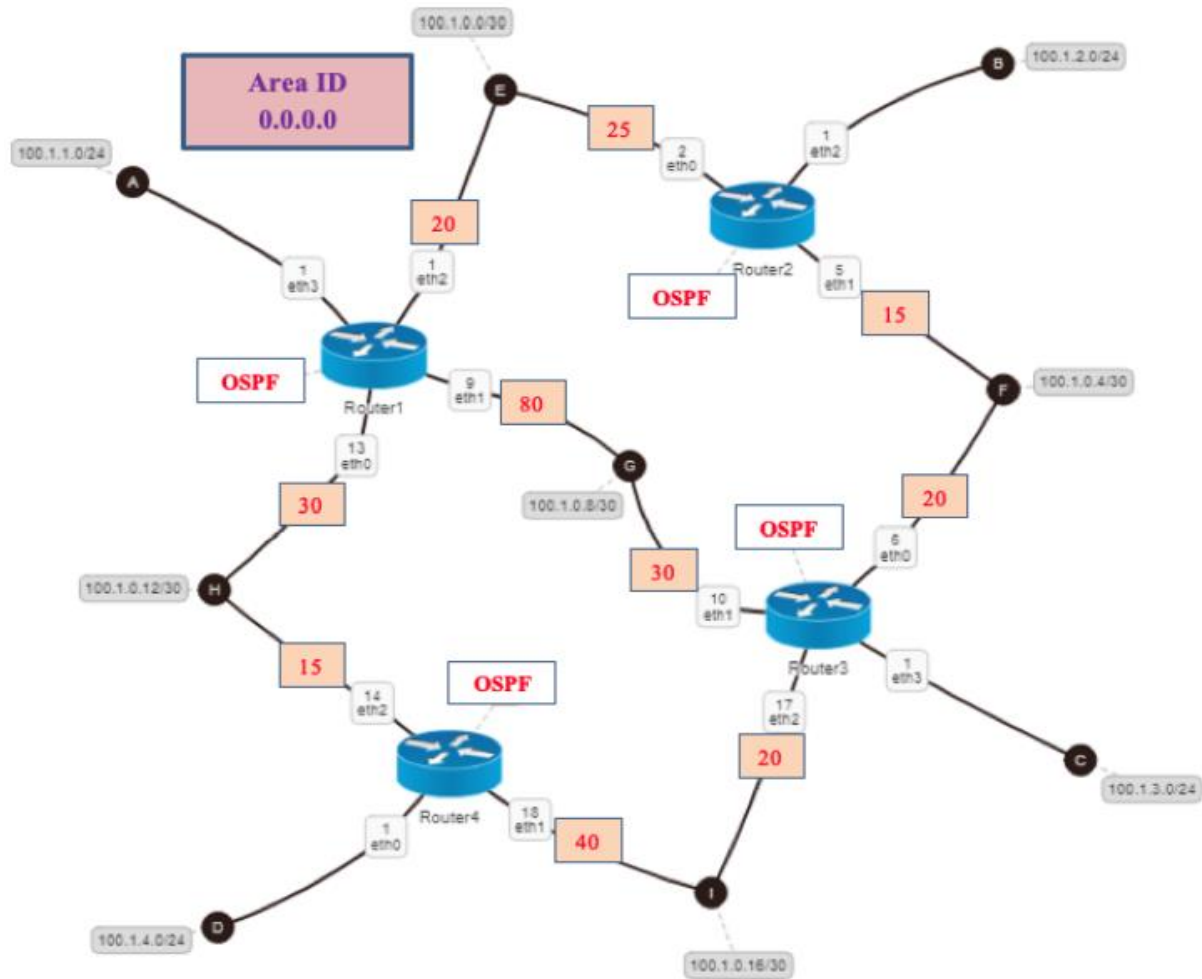
```

b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap12$ kathara wipe
Are you sure to wipe Kathara? (y/n) y

```

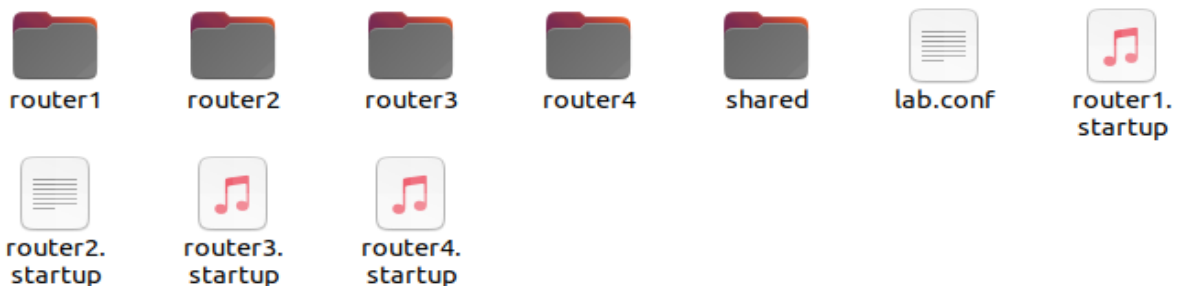
Bài tập 13:

- 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 BaiTap13 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 BaiTap13 bằng lệnh:

```
$ cd /home/student/your_workspace/BaiTap13
```



- 3) Trên file **lab.conf**, soạn thảo nội dung mô tả hình thái mạng theo thiết kế giống 3) của Bài tập 12.

```
GNU nano 4.8 lab.conf Modified
router1[0]=H
router1[1]=G
router1[2]=E
router1[3]=A

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

router3[0]=F
router3[1]=G
router3[2]=I
router3[3]=C

router4[0]=D
router4[1]=I
router4[2]=H
```

- 4) Trên các file .startup của các router, soạn thảo nội dung cấu hình cho giao diện mạng của chúng giống 4) của Bài tập 12.

```
GNU nano 4.8 router1.startup Modified
ifconfig eth0 100.1.0.13/30 up
ifconfig eth1 100.1.0.9/30 up
ifconfig eth2 100.1.0.1/30 up
ifconfig eth3 100.1.1.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router2.startup Modified
ifconfig eth0 100.1.0.2/30 up
ifconfig eth1 100.1.0.5/30 up
ifconfig eth2 100.1.2.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router3.startup Modified
ifconfig eth0 100.1.0.6/30 up
ifconfig eth1 100.1.0.10/30 up
ifconfig eth2 100.1.0.17/30 up
ifconfig eth3 100.1.3.1/24 up

/etc/init.d/quagga start

GNU nano 4.8 router4.startup Modified
ifconfig eth0 100.1.4.1/24 up
ifconfig eth1 100.1.0.18/30 up
ifconfig eth2 100.1.0.14/30 up

/etc/init.d/quagga start
```

- 5) Trong mỗi thư mục router đã tạo ra, ví dụ : trong router1, tạo cấu trúc thư mục giống như hình 4.9 dưới đây:

Các file trong thư mục /etc/quagga có ý nghĩa đã được giải thích ở 5) của Bài tập 12. Chỉ có 1 file khác là:

➤ **ospfd.conf**: chứa nội dung miêu tả hoạt động của giải thuật OSPFv2 trên IPv4 trên Quagga.

- 6) Miêu tả các thiết lập cho dịch vụ Quagga qua file zebra.conf giống với 6) của Bài tập 12. Bước 6) này cũng có thể bỏ qua nếu như muốn sử dụng các thiết lập mặc nhiên sẵn có mà Quagga trên Kathará đã cung cấp.

- 7) Trên file daemons, thêm vào nội dung khai báo giao thức vạch đường sẽ sử dụng trên router. Nội dung file daemons có thể tham khảo như sau:

```
GNU nano 4.8 daemons Modified
zebra=yes
ospfd=yes

bgpd=no
ripd=no
```

- 8) Trên file ospfd.conf, thêm vào nội dung miêu tả hoạt động của giao thức OSPFv2 trên Quagga của router. Nội dung file ospfd.conf của router1 có thể tham khảo như sau:

```
GNU nano 4.8 ospfd.conf
hostname ospfd
password zebra
enable password zebra

interface eth0
ospf cost 30
interface eth1
ospf cost 80
interface eth2
ospf cost 20

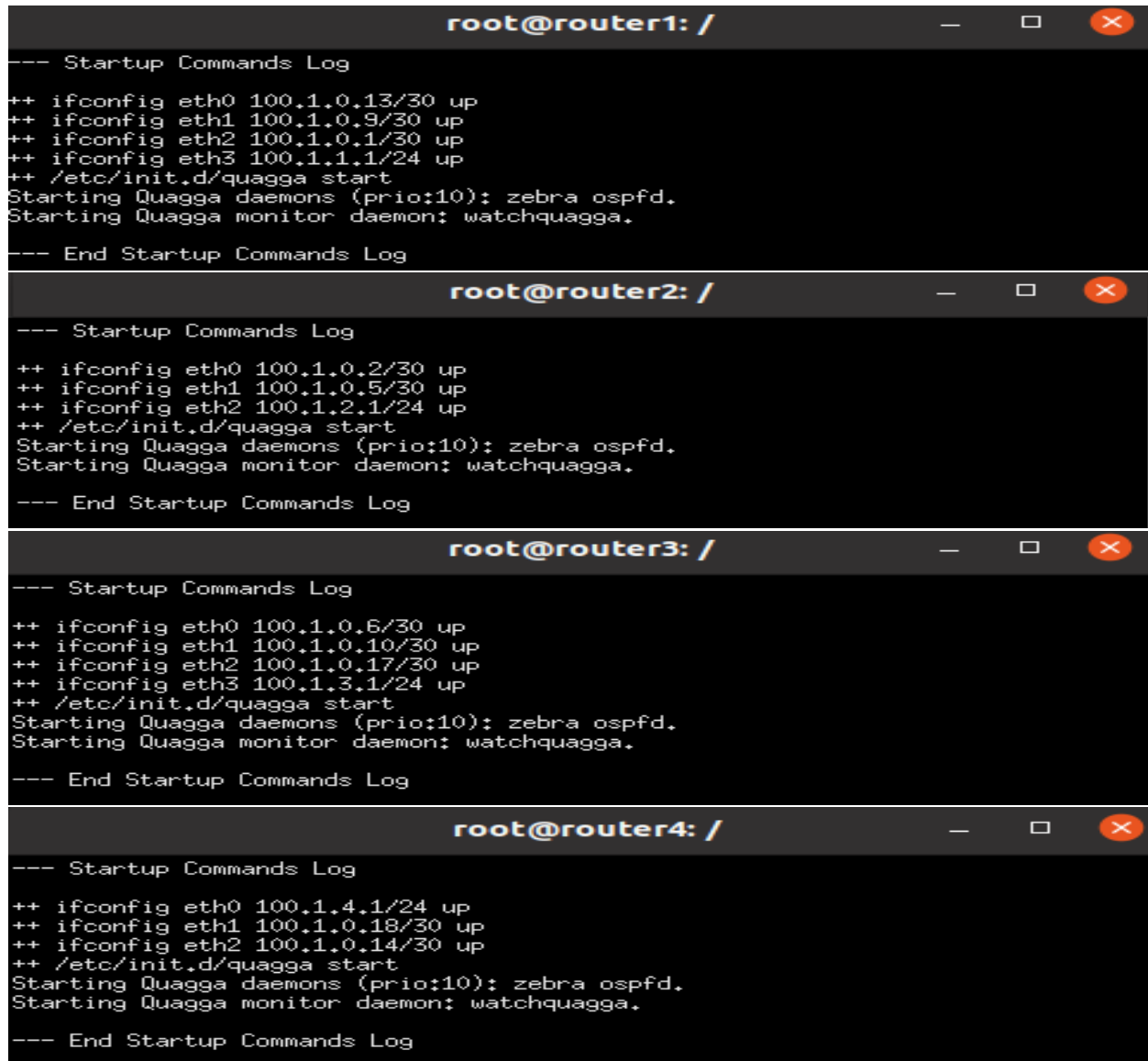
router ospf
network 100.1.0.0/16 area 0.0.0.0
redistribute connected 55

log file /var/log/zebra/ospfd.log
```

- Lệnh interface chỉ ra giao diện của router sẽ chạy giải thuật OSPFv2
➤ Lệnh ospf cost chỉ ra chi phí để Gửi gói tin OSPFv2 trên giao diện đã được khai báo bởi lệnh interface

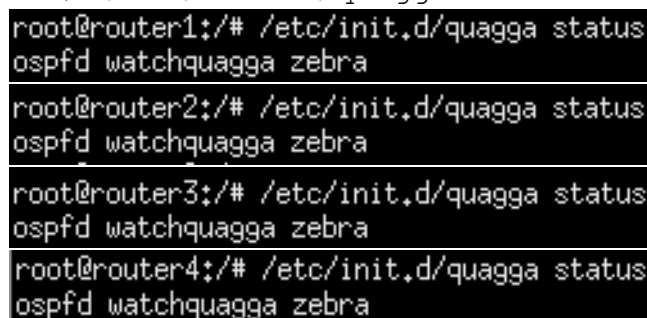
- 9) Khởi động mạng ảo BaiTap13. Trên các router, kiểm tra dịch vụ Quagga đã được bật lên và giao thức RIPv2 đã hoạt động hay chưa bằng lệnh:

```
$ kathara lstart
```



```
root@router1: /  
--- Startup Commands Log  
++ ifconfig eth0 100.1.0.13/30 up  
++ ifconfig eth1 100.1.0.9/30 up  
++ ifconfig eth2 100.1.0.1/30 up  
++ ifconfig eth3 100.1.1.1/24 up  
++ /etc/init.d/quagga start  
Starting Quagga daemons (prio:10): zebra ospfd.  
Starting Quagga monitor daemon: watchquagga.  
--- End Startup Commands Log  
  
root@router2: /  
--- Startup Commands Log  
++ ifconfig eth0 100.1.0.2/30 up  
++ ifconfig eth1 100.1.0.5/30 up  
++ ifconfig eth2 100.1.2.1/24 up  
++ /etc/init.d/quagga start  
Starting Quagga daemons (prio:10): zebra ospfd.  
Starting Quagga monitor daemon: watchquagga.  
--- End Startup Commands Log  
  
root@router3: /  
--- Startup Commands Log  
++ ifconfig eth0 100.1.0.6/30 up  
++ ifconfig eth1 100.1.0.10/30 up  
++ ifconfig eth2 100.1.0.17/30 up  
++ ifconfig eth3 100.1.3.1/24 up  
++ /etc/init.d/quagga start  
Starting Quagga daemons (prio:10): zebra ospfd.  
Starting Quagga monitor daemon: watchquagga.  
--- End Startup Commands Log  
  
root@router4: /  
--- Startup Commands Log  
++ ifconfig eth0 100.1.4.1/24 up  
++ ifconfig eth1 100.1.0.18/30 up  
++ ifconfig eth2 100.1.0.14/30 up  
++ /etc/init.d/quagga start  
Starting Quagga daemons (prio:10): zebra ospfd.  
Starting Quagga monitor daemon: watchquagga.  
--- End Startup Commands Log
```

```
$ /etc/init.d/quagga status
```



```
root@router1:/# /etc/init.d/quagga status  
ospfd watchquagga zebra  
  
root@router2:/# /etc/init.d/quagga status  
ospfd watchquagga zebra  
  
root@router3:/# /etc/init.d/quagga status  
ospfd watchquagga zebra  
  
root@router4:/# /etc/init.d/quagga status  
ospfd watchquagga zebra
```


10) Đợi 30 giây sau khi mạng ảo BaiTap13 đã khởi động xong.

➤ Kiểm tra bảng vạch đường trên các router bằng: **route**. Nhận xét kết quả

\$ route

```
root@router1:/# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0        0.0.0.0        255.255.255.252 U        0      0      0 eth2
100.1.0.4        100.1.0.2      255.255.255.252 UG       20     0      0 eth2
100.1.0.8        0.0.0.0        255.255.255.252 U        0      0      0 eth1
100.1.0.12       0.0.0.0        255.255.255.252 U        0      0      0 eth0
100.1.0.16       100.1.0.10     255.255.255.252 UG       20     0      0 eth1
100.1.1.0        0.0.0.0        255.255.255.0   U        0      0      0 eth3
100.1.2.0        100.1.0.2      255.255.255.0   UG       20     0      0 eth2
100.1.3.0        100.1.0.10     255.255.255.0   UG       20     0      0 eth1
100.1.4.0        100.1.0.14     255.255.255.0   UG       20     0      0 eth0
```

```
root@router2:/# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0        0.0.0.0        255.255.255.252 U        0      0      0 eth0
100.1.0.4        0.0.0.0        255.255.255.252 U        0      0      0 eth1
100.1.0.8        100.1.0.1      255.255.255.252 UG       20     0      0 eth0
100.1.0.12       100.1.0.1      255.255.255.252 UG       20     0      0 eth0
100.1.0.16       100.1.0.6      255.255.255.252 UG       20     0      0 eth1
100.1.1.0        100.1.0.1      255.255.255.0   UG       20     0      0 eth0
100.1.2.0        0.0.0.0        255.255.255.0   U        0      0      0 eth2
100.1.3.0        100.1.0.6      255.255.255.0   UG       20     0      0 eth1
100.1.4.0        100.1.0.1      255.255.255.0   UG       20     0      0 eth0
```

```
root@router3:/# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0        100.1.0.5      255.255.255.252 UG       20     0      0 eth0
100.1.0.4        0.0.0.0        255.255.255.252 U        0      0      0 eth0
100.1.0.8        0.0.0.0        255.255.255.252 U        0      0      0 eth1
100.1.0.12       100.1.0.18     255.255.255.252 UG       20     0      0 eth2
100.1.0.16       0.0.0.0        255.255.255.252 U        0      0      0 eth2
100.1.1.0        100.1.0.18     255.255.255.0   UG       20     0      0 eth2
100.1.2.0        100.1.0.5      255.255.255.0   UG       20     0      0 eth0
100.1.3.0        0.0.0.0        255.255.255.0   U        0      0      0 eth3
100.1.4.0        100.1.0.18     255.255.255.0   UG       20     0      0 eth2
```

```
root@router4:/# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0        100.1.0.13     255.255.255.252 UG       20     0      0 eth2
100.1.0.4        100.1.0.17     255.255.255.252 UG       20     0      0 eth1
100.1.0.8        100.1.0.13     255.255.255.252 UG       20     0      0 eth2
100.1.0.12       0.0.0.0        255.255.255.252 U        0      0      0 eth2
100.1.0.16       0.0.0.0        255.255.255.252 U        0      0      0 eth1
100.1.1.0        100.1.0.13     255.255.255.0   UG       20     0      0 eth2
100.1.2.0        100.1.0.13     255.255.255.0   UG       20     0      0 eth2
100.1.3.0        100.1.0.17     255.255.255.0   UG       20     0      0 eth1
100.1.4.0        0.0.0.0        255.255.255.0   U        0      0      0 eth0
```

➤ Thực hiện ping giữa các router để kiểm tra tính liên thông của các mạng LAN trong mô hình BaiTap13. Ví dụ:

✓ Từ router1 đến router4:


```
$ ping 100.1.4.1
```

```
root@router1:/# ping 100.1.4.1
PING 100.1.4.1 (100.1.4.1) 56(84) bytes of data.
64 bytes from 100.1.4.1: icmp_seq=1 ttl=64 time=0.049 ms
64 bytes from 100.1.4.1: icmp_seq=2 ttl=64 time=0.110 ms
64 bytes from 100.1.4.1: icmp_seq=3 ttl=64 time=0.082 ms
64 bytes from 100.1.4.1: icmp_seq=4 ttl=64 time=0.076 ms
64 bytes from 100.1.4.1: icmp_seq=5 ttl=64 time=0.125 ms
64 bytes from 100.1.4.1: icmp_seq=6 ttl=64 time=0.076 ms
64 bytes from 100.1.4.1: icmp_seq=7 ttl=64 time=0.080 ms
64 bytes from 100.1.4.1: icmp_seq=8 ttl=64 time=0.082 ms
64 bytes from 100.1.4.1: icmp_seq=9 ttl=64 time=0.043 ms
64 bytes from 100.1.4.1: icmp_seq=10 ttl=64 time=0.081 ms
64 bytes from 100.1.4.1: icmp_seq=11 ttl=64 time=0.077 ms
64 bytes from 100.1.4.1: icmp_seq=12 ttl=64 time=0.082 ms
64 bytes from 100.1.4.1: icmp_seq=13 ttl=64 time=0.083 ms
64 bytes from 100.1.4.1: icmp_seq=14 ttl=64 time=0.081 ms
64 bytes from 100.1.4.1: icmp_seq=15 ttl=64 time=0.065 ms
```

✓ Từ router2 đến router3:

```
$ ping 100.1.0.10
```

```
root@router2:/# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data.
64 bytes from 100.1.0.10: icmp_seq=1 ttl=62 time=0.074 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=62 time=0.110 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=62 time=0.111 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=62 time=0.116 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=62 time=0.123 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=62 time=0.110 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=62 time=0.210 ms
64 bytes from 100.1.0.10: icmp_seq=8 ttl=62 time=0.115 ms
64 bytes from 100.1.0.10: icmp_seq=9 ttl=62 time=0.122 ms
64 bytes from 100.1.0.10: icmp_seq=10 ttl=62 time=0.123 ms
64 bytes from 100.1.0.10: icmp_seq=11 ttl=62 time=0.123 ms
64 bytes from 100.1.0.10: icmp_seq=12 ttl=62 time=0.113 ms
64 bytes from 100.1.0.10: icmp_seq=13 ttl=62 time=0.088 ms
64 bytes from 100.1.0.10: icmp_seq=14 ttl=62 time=0.118 ms
64 bytes from 100.1.0.10: icmp_seq=15 ttl=62 time=0.103 ms
```

11) Trên router1, thực hiện lần lượt các lệnh sau:

```
$ traceroute 100.1.0.5 (giao diện eth1 của router2)
```

```
root@router1:/# traceroute 100.1.0.5
traceroute to 100.1.0.5 (100.1.0.5), 30 hops max, 60 byte packets
1 100.1.0.5 (100.1.0.5) 0.054 ms 0.008 ms 0.007 ms
```

```
$ traceroute 100.1.0.17 (giao diện eth2 của router3)
```

```
root@router1:/# traceroute 100.1.0.17
traceroute to 100.1.0.17 (100.1.0.17), 30 hops max, 60 byte packets
1 100.1.0.17 (100.1.0.17) 0.213 ms 0.159 ms 0.150 ms
```

```
$ traceroute 100.1.0.10 (giao diện eth1 của router3)
```

```
root@router1:/# traceroute 100.1.0.10
traceroute to 100.1.0.10 (100.1.0.10), 30 hops max, 60 byte packets
1 100.1.0.10 (100.1.0.10) 0.210 ms 0.160 ms 0.149 ms
```

Dựa trên các kết quả nhận được, hãy cho biết từ router1 đi đến các địa chỉ này sẽ đi qua lần lượt những nhánh mạng nào?

12) Trên router1 (hoặc 1 router bất kỳ nếu muốn), thực hiện lệnh:

```
$ tcpdump -i any -w /hosthome/BaiTap13_router1.pcap
```

để bắt các gói tin OSPFv2 được trao đổi giữa router1 và các router khác trong AS. Sau khoảng 30 giây thì dừng lại lệnh tcpdump lại.

Các phần 13), 14) và 15) dưới đây giúp sinh viên tự tìm hiểu thêm về OSPFv2 trên Quagga. Sinh viên không bắt buộc phải thực hiện phần này.

13) Trên router bất kỳ, dùng lệnh:

```
$ telnet localhost ospfd
```

Nhập mật khẩu là zebra đã đặt ở 8)

14) Lần lượt thực hiện các lệnh sau. Sau đó tự tìm hiểu và trả lời về kết quả hiển thị của các lệnh đó.

```
$ show ip ospf route
```

```
$ show ip ospf database
```

15) Trên máy thực, mở file BT13_router1.pcap bằng Wireshark.

- Chọn gói tin OSPFv2 có địa chỉ nguồn (source) bất kỳ, chẳng hạn: 100.1.0.2 và trả lời các câu hỏi sau:
 - ✓ Địa chỉ IP đích (destination) của khung này là bao nhiêu? Địa chỉ IP này là Broadcast, Unicast hay Multicast?
 - ✓ Gói tin này có sử dụng giao thức gì trên tầng vận chuyển hay không?
- Trong trường Open Shortest Path First, hãy tự tìm hiểu thêm về 2 trường con bên trong, lần lượt là: OSPF Header và OSPF Hello Packet; trả lời các câu hỏi sau:
 - ✓ Tìm tên trường hiển thị phiên bản (version) của OSPF.
 - ✓ Tìm hiểu và giải thích ý nghĩa của trường Active Neighbor.
 - ✓ Tìm hiểu và giải thích ý nghĩa của trường Designated Router và Backup Designated Router.

16) Trên router1, tắt đi giao diện eth1 tương tự 15) của Bài tập 12:

```
$ ifconfig eth1 down
```

```
root@router1:/# ifconfig eth1 down
root@router1:/#
```

17) Trên router1, gửi dữ liệu đến router3 tương tự 16) của Bài tập 12:

- Dừng lại sau 7 giây. Có ping thành công hay không?

```
root@router1:/# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data:
64 bytes from 100.1.0.10: icmp_seq=1 ttl=63 time=0.072 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=63 time=0.092 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=63 time=0.088 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=63 time=0.095 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=63 time=0.105 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=63 time=0.103 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=63 time=0.090 ms
^C
--- 100.1.0.10 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6126ms
rtt min/avg/max/mdev = 0.072/0.092/0.105/0.011 ms
```

- Kiểm tra bảng vạch đường bằng lệnh route. Thông tin trong bảng có thay đổi gì hay chưa?

```

root@router1:~# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data:
64 bytes from 100.1.0.10: icmp_seq=1 ttl=63 time=0.072 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=63 time=0.092 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=63 time=0.088 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=63 time=0.095 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=63 time=0.105 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=63 time=0.103 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=63 time=0.090 ms
^C
--- 100.1.0.10 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6126ms
rtt min/avg/max/mdev = 0.072/0.092/0.105/0.011 ms

```

18) Sau 30 giây, thực hiện lại lệnh

```
$ ping 100.1.0.10
```

➤ Dừng lại sau 7 giây. Có ping thành công hay không?

```

root@router1:~# ping 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data:
64 bytes from 100.1.0.10: icmp_seq=1 ttl=63 time=0.066 ms
64 bytes from 100.1.0.10: icmp_seq=2 ttl=63 time=0.093 ms
64 bytes from 100.1.0.10: icmp_seq=3 ttl=63 time=0.110 ms
64 bytes from 100.1.0.10: icmp_seq=4 ttl=63 time=0.093 ms
64 bytes from 100.1.0.10: icmp_seq=5 ttl=63 time=0.056 ms
64 bytes from 100.1.0.10: icmp_seq=6 ttl=63 time=0.092 ms
64 bytes from 100.1.0.10: icmp_seq=7 ttl=63 time=0.094 ms
64 bytes from 100.1.0.10: icmp_seq=8 ttl=63 time=0.099 ms
64 bytes from 100.1.0.10: icmp_seq=9 ttl=63 time=0.093 ms
^C
--- 100.1.0.10 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8178ms
rtt min/avg/max/mdev = 0.056/0.088/0.110/0.018 ms
root@router1:~#

```

➤ Nếu không, nhận xét gì về thời gian cập nhật bảng vạch đường của OSPFv2 so với RIPv2 khi hình trạng mạng có thay đổi như đã nêu ở 16)

19) Đợi thêm 30 giây nữa rồi kiểm tra bảng vạch đường bằng lệnh route. Thông tin trong bảng có thay đổi gì hay chưa? Chỉ ra thay đổi đó. Gợi ý: thông tin đường đi từ router1 đến router3 qua G được tự động cập nhật bằng 1 đường đi mới.

```

root@router1:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
100.1.0.0 0.0.0.0 255.255.255.252 U 0 0 0 eth2
100.1.0.4 100.1.0.2 255.255.255.252 UG 20 0 0 eth2
100.1.0.8 100.1.0.2 255.255.255.252 UG 20 0 0 eth2
100.1.0.12 0.0.0.0 255.255.255.252 U 0 0 0 eth0
100.1.0.16 100.1.0.14 255.255.255.252 UG 20 0 0 eth0
100.1.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
100.1.2.0 100.1.0.2 255.255.255.0 UG 20 0 0 eth2
100.1.3.0 100.1.0.2 255.255.255.0 UG 20 0 0 eth2
100.1.4.0 100.1.0.14 255.255.255.0 UG 20 0 0 eth0
root@router1:~#

```

20) Hủy mạng ảo, sau khi đã thực hiện xong Bài tập 13.

```
$ kathara wipe
```

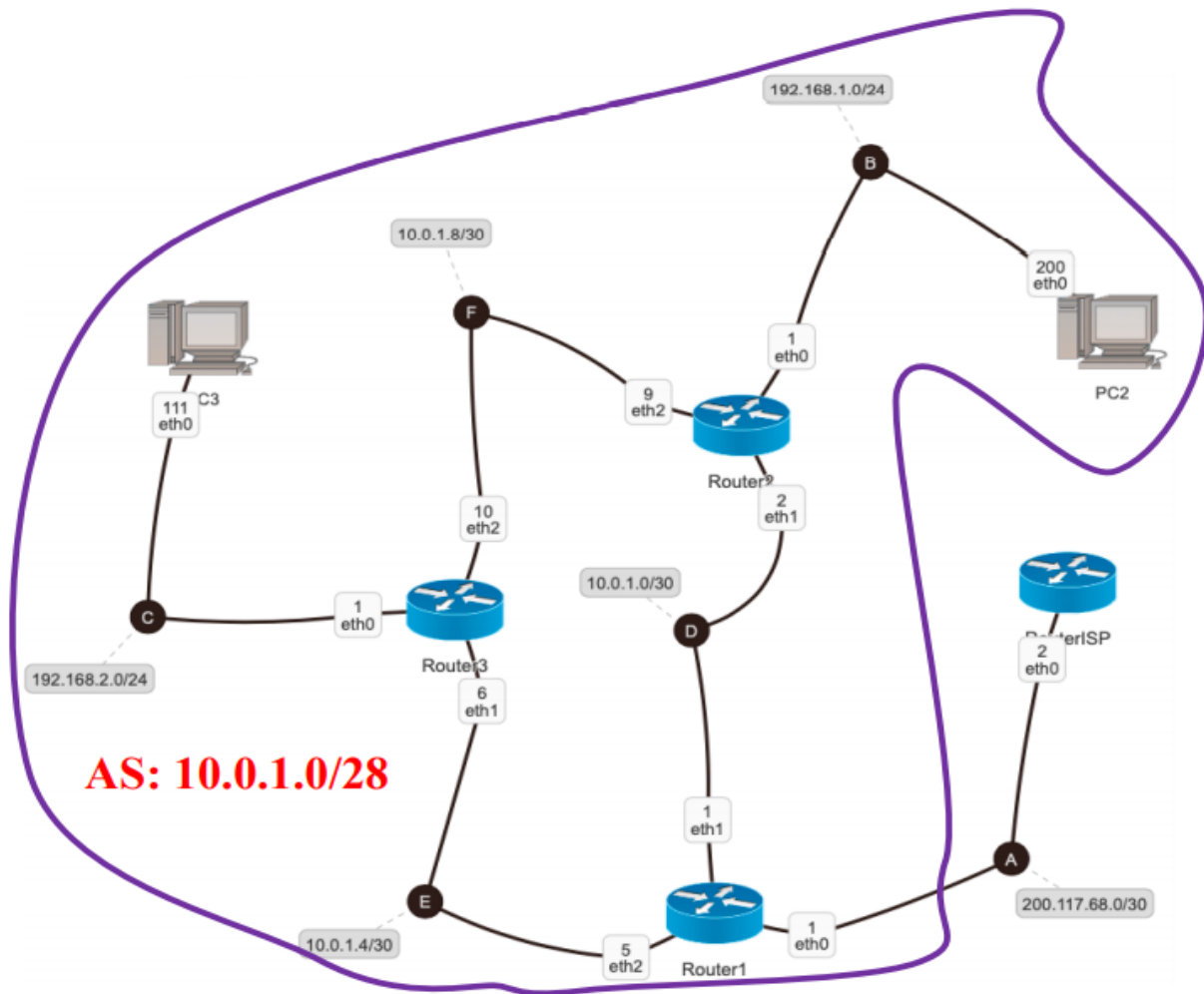
```

b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap13$ kathara wipe
Are you sure to wipe Kathara? (y/n) y

```

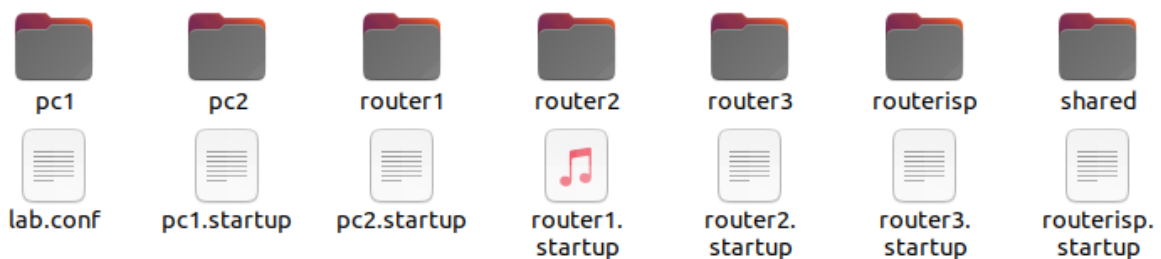
Bài tập 14:

- 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 BaiTap14 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 BaiTap14 bằng lệnh:

```
$ cd /home/student/your_workspace/BaiTap14
```



3) Một số điểm lưu ý khi xây dựng mô hình mạng Bài tập 14:

```
GNU nano 4.8 lab.conf
pc1[0]=B
pc2[0]=C

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

router2[0]=B
router2[1]=D
router2[2]=F

router3[0]=C
router3[1]=E
router3[2]=F

routerisp[0]=A

GNU nano 4.8 pc1.startup
ifconfig eth0 192.168.1.200/24 up
route add default gw 192.168.1.1

GNU nano 4.8 pc2.startup
ifconfig eth0 192.168.2.111/24 up
route add default gw 192.168.2.1

GNU nano 4.8 router1.startup
ifconfig eth0 200.117.68.1/30 up
ifconfig eth1 10.0.1.1/30 up
ifconfig eth2 10.0.1.5/30 up

/etc/init.d/quagga start

GNU nano 4.8 router2.startup
ifconfig eth0 192.168.1.1/24 up
ifconfig eth1 10.0.1.2/30 up
ifconfig eth2 10.0.1.9/30 up

/etc/init.d/quagga start

GNU nano 4.8 router3.startup
ifconfig eth0 192.168.2.1/24 up
ifconfig eth1 10.0.1.6/30 up
ifconfig eth2 10.0.1.10/30 up

/etc/init.d/quagga start
```

```
GNU nano 4.8 routerisp.startup
ifconfig eth0 200.117.68.2/30 up
route add default gw 200.117.68.1

GNU nano 4.8 zebra.conf
hostname zebra
password zebra
enable password zebra

log file /var/log/zebra/zebra.log

GNU nano 4.8 daemons
zebra=yes
ripd=yes

bgpd=no
ospfd=no

GNU nano 4.8 ripd.conf
hostname ripd
password zebra
enable password zebra

router rip
network 10.0.1.0/28
redistribute connected

log file /var/log/zebra/ripd.log
```

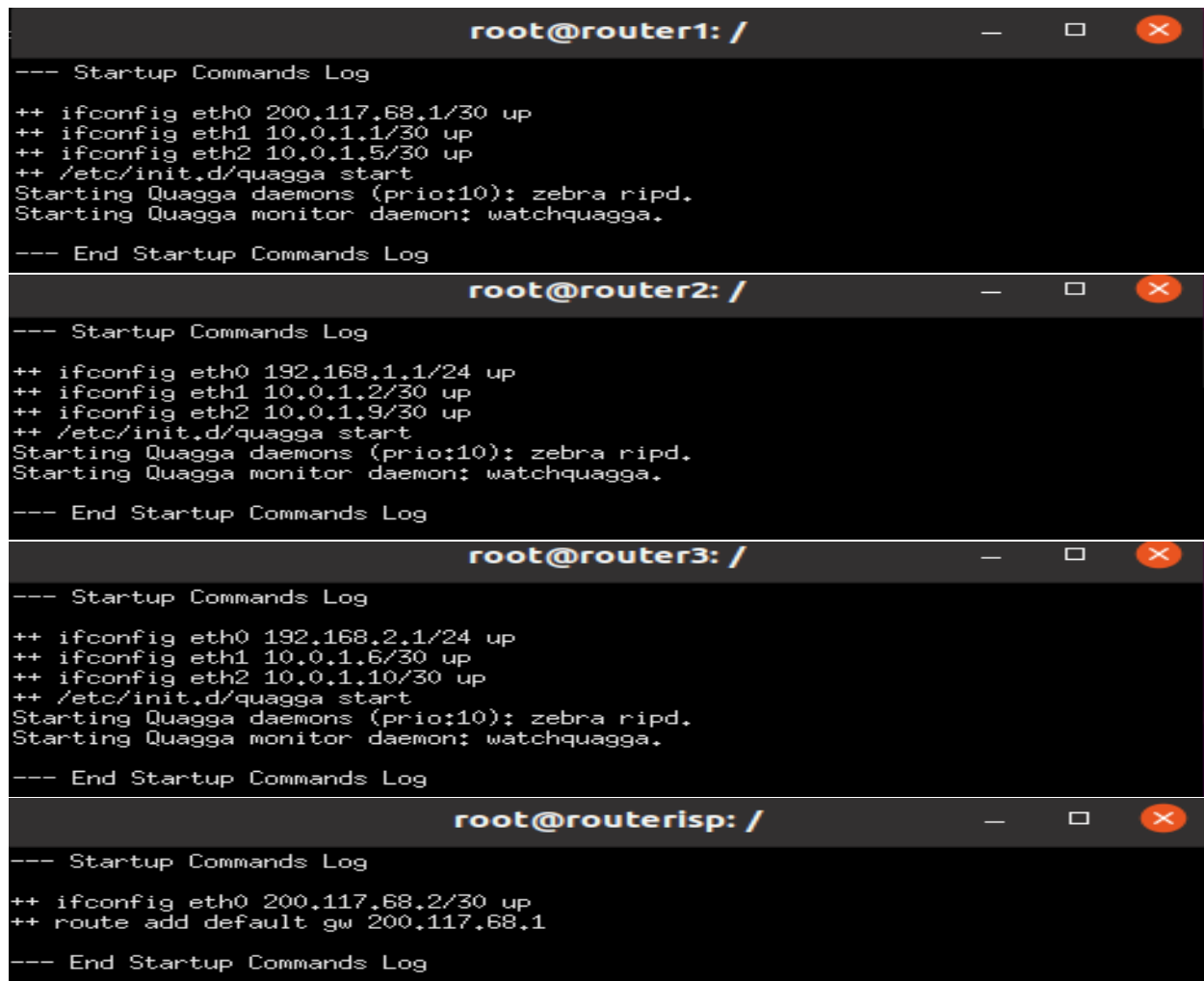
- Các router vạch đường RIPv2 trong miền 10.0.1.0/28 bao gồm các mạng: D, F, E.
- Mạng A, B và C KHÔNG nằm trong phạm vi truyền tải gói tin RIPv2 của các router. Vì vậy, thông tin về A (do router1 biết), B (do router2 biết) và C (do router3 biết) phải được phân phối lại cho các router khác trong mạng dưới dạng gói tin RIPv2 bằng lệnh: redistribute connected
- routerISP KHÔNG chạy giải thuật RIPv2.

4) Kết quả của Bài tập 14 phải thỏa mãn các điều kiện sau:

```
$ kathara lstart
```

```
root@pc1: /
--- Startup Commands Log
++ ifconfig eth0 192.168.1.200/24 up
++ route add default gw 192.168.1.1
--- End Startup Commands Log

root@pc2: /
--- Startup Commands Log
++ ifconfig eth0 192.168.2.111/24 up
++ route add default gw 192.168.2.1
--- End Startup Commands Log
```



```
root@router1: /
--- Startup Commands Log
++ ifconfig eth0 200.117.68.1/30 up
++ ifconfig eth1 10.0.1.1/30 up
++ ifconfig eth2 10.0.1.5/30 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@router2: /
--- Startup Commands Log
++ ifconfig eth0 192.168.1.1/24 up
++ ifconfig eth1 10.0.1.2/30 up
++ ifconfig eth2 10.0.1.9/30 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@router3: /
--- Startup Commands Log
++ ifconfig eth0 192.168.2.1/24 up
++ ifconfig eth1 10.0.1.6/30 up
++ ifconfig eth2 10.0.1.10/30 up
++ /etc/init.d/quagga start
Starting Quagga daemons (prio:10): zebra ripd.
Starting Quagga monitor daemon: watchquagga.
--- End Startup Commands Log

root@routerisp: /
--- Startup Commands Log
++ ifconfig eth0 200.117.68.2/30 up
++ route add default gw 200.117.68.1
--- End Startup Commands Log
```

- Các router trong AS 10.0.1.0/28 ping thành công đến routerISP và ngược lại.
 - ✓ router1, router2 và router3 ping đến routerISP

\$ ping 200.117.68.2

```
root@router1:/# ping 200.117.68.2
PING 200.117.68.2 (200.117.68.2) 56(84) bytes of data.
64 bytes from 200.117.68.2: icmp_seq=1 ttl=64 time=0.064 ms
64 bytes from 200.117.68.2: icmp_seq=2 ttl=64 time=0.077 ms
64 bytes from 200.117.68.2: icmp_seq=3 ttl=64 time=0.085 ms
64 bytes from 200.117.68.2: icmp_seq=4 ttl=64 time=0.082 ms
64 bytes from 200.117.68.2: icmp_seq=5 ttl=64 time=0.084 ms
64 bytes from 200.117.68.2: icmp_seq=6 ttl=64 time=0.084 ms
64 bytes from 200.117.68.2: icmp_seq=7 ttl=64 time=0.083 ms
64 bytes from 200.117.68.2: icmp_seq=8 ttl=64 time=0.075 ms
64 bytes from 200.117.68.2: icmp_seq=9 ttl=64 time=0.079 ms
64 bytes from 200.117.68.2: icmp_seq=10 ttl=64 time=0.083 ms
^C
--- 200.117.68.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9196ms
rtt min/avg/max/mdev = 0.064/0.079/0.085/0.011 ms
```

```
root@router2:/# ping 200.117.68.2
PING 200.117.68.2 (200.117.68.2) 56(84) bytes of data.
64 bytes from 200.117.68.2: icmp_seq=1 ttl=63 time=0.064 ms
64 bytes from 200.117.68.2: icmp_seq=2 ttl=63 time=0.097 ms
64 bytes from 200.117.68.2: icmp_seq=3 ttl=63 time=0.095 ms
64 bytes from 200.117.68.2: icmp_seq=4 ttl=63 time=0.104 ms
64 bytes from 200.117.68.2: icmp_seq=5 ttl=63 time=0.105 ms
64 bytes from 200.117.68.2: icmp_seq=6 ttl=63 time=0.108 ms
64 bytes from 200.117.68.2: icmp_seq=7 ttl=63 time=0.105 ms
64 bytes from 200.117.68.2: icmp_seq=8 ttl=63 time=0.104 ms
64 bytes from 200.117.68.2: icmp_seq=9 ttl=63 time=0.107 ms
64 bytes from 200.117.68.2: icmp_seq=10 ttl=63 time=0.103 ms
^C
--- 200.117.68.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9223ms
rtt min/avg/max/mdev = 0.064/0.099/0.108/0.013 ms
```

```
root@router3:/# ping 200.117.68.2
PING 200.117.68.2 (200.117.68.2) 56(84) bytes of data.
64 bytes from 200.117.68.2: icmp_seq=1 ttl=63 time=0.066 ms
64 bytes from 200.117.68.2: icmp_seq=2 ttl=63 time=0.105 ms
64 bytes from 200.117.68.2: icmp_seq=3 ttl=63 time=0.099 ms
64 bytes from 200.117.68.2: icmp_seq=4 ttl=63 time=0.093 ms
64 bytes from 200.117.68.2: icmp_seq=5 ttl=63 time=0.103 ms
64 bytes from 200.117.68.2: icmp_seq=6 ttl=63 time=0.106 ms
64 bytes from 200.117.68.2: icmp_seq=7 ttl=63 time=0.099 ms
64 bytes from 200.117.68.2: icmp_seq=8 ttl=63 time=0.095 ms
64 bytes from 200.117.68.2: icmp_seq=9 ttl=63 time=0.104 ms
64 bytes from 200.117.68.2: icmp_seq=10 ttl=63 time=0.105 ms
^C
--- 200.117.68.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9220ms
rtt min/avg/max/mdev = 0.066/0.097/0.106/0.015 ms
```

routerISP ping đến các router còn lại

\$ ping 200.117.68.1(router1)

```
root@routerisp:/# ping 200.117.68.1
PING 200.117.68.1 (200.117.68.1) 56(84) bytes of data.
64 bytes from 200.117.68.1: icmp_seq=1 ttl=64 time=0.068 ms
64 bytes from 200.117.68.1: icmp_seq=2 ttl=64 time=0.080 ms
64 bytes from 200.117.68.1: icmp_seq=3 ttl=64 time=0.079 ms
64 bytes from 200.117.68.1: icmp_seq=4 ttl=64 time=0.083 ms
64 bytes from 200.117.68.1: icmp_seq=5 ttl=64 time=0.080 ms
64 bytes from 200.117.68.1: icmp_seq=6 ttl=64 time=0.200 ms
64 bytes from 200.117.68.1: icmp_seq=7 ttl=64 time=0.083 ms
64 bytes from 200.117.68.1: icmp_seq=8 ttl=64 time=0.081 ms
64 bytes from 200.117.68.1: icmp_seq=9 ttl=64 time=0.080 ms
64 bytes from 200.117.68.1: icmp_seq=10 ttl=64 time=0.082 ms
^X64 bytes from 200.117.68.1: icmp_seq=11 ttl=64 time=0.084 ms
64 bytes from 200.117.68.1: icmp_seq=12 ttl=64 time=0.079 ms
^C
--- 200.117.68.1 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11241ms
rtt min/avg/max/mdev = 0.068/0.083/0.200/0.035 ms
```


\$ ping 192.168.1.1(router2)

```
root@routerisp:/# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=0.066 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=0.101 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=63 time=0.099 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=63 time=0.103 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=63 time=0.103 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=63 time=0.100 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=63 time=0.102 ms
64 bytes from 192.168.1.1: icmp_seq=8 ttl=63 time=0.103 ms
64 bytes from 192.168.1.1: icmp_seq=9 ttl=63 time=0.100 ms
64 bytes from 192.168.1.1: icmp_seq=10 ttl=63 time=0.103 ms
^X64 bytes from 192.168.1.1: icmp_seq=11 ttl=63 time=0.101 ms
^C
--- 192.168.1.1 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10217ms
rtt min/avg/max/mdev = 0.066/0.098/0.103/0.012 ms
```

\$ ping 192.168.2.1(router3)

```
root@routerisp:/# ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=63 time=0.065 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=63 time=0.098 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=63 time=0.094 ms
64 bytes from 192.168.2.1: icmp_seq=4 ttl=63 time=0.060 ms
64 bytes from 192.168.2.1: icmp_seq=5 ttl=63 time=0.072 ms
64 bytes from 192.168.2.1: icmp_seq=6 ttl=63 time=0.127 ms
64 bytes from 192.168.2.1: icmp_seq=7 ttl=63 time=0.054 ms
64 bytes from 192.168.2.1: icmp_seq=8 ttl=63 time=0.093 ms
64 bytes from 192.168.2.1: icmp_seq=9 ttl=63 time=0.093 ms
64 bytes from 192.168.2.1: icmp_seq=10 ttl=63 time=0.091 ms
^C
--- 192.168.2.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9211ms
rtt min/avg/max/mdev = 0.054/0.084/0.127/0.023 ms
```

➤ pc1 ping thành công đến pc2 và ngược lại.

\$ ping 192.168.2.111(pc2)

```
root@pc1:/# ping 192.168.2.111
PING 192.168.2.111 (192.168.2.111) 56(84) bytes of data.
64 bytes from 192.168.2.111: icmp_seq=1 ttl=62 time=0.102 ms
64 bytes from 192.168.2.111: icmp_seq=2 ttl=62 time=0.116 ms
64 bytes from 192.168.2.111: icmp_seq=3 ttl=62 time=0.126 ms
64 bytes from 192.168.2.111: icmp_seq=4 ttl=62 time=0.125 ms
64 bytes from 192.168.2.111: icmp_seq=5 ttl=62 time=0.128 ms
64 bytes from 192.168.2.111: icmp_seq=6 ttl=62 time=0.118 ms
64 bytes from 192.168.2.111: icmp_seq=7 ttl=62 time=0.124 ms
64 bytes from 192.168.2.111: icmp_seq=8 ttl=62 time=0.164 ms
64 bytes from 192.168.2.111: icmp_seq=9 ttl=62 time=0.125 ms
64 bytes from 192.168.2.111: icmp_seq=10 ttl=62 time=0.115 ms
^C
--- 192.168.2.111 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9215ms
rtt min/avg/max/mdev = 0.102/0.124/0.164/0.017 ms
root@pc1:/#
```

```
$ ping 192.168.1.200 (pc1)
root@pc2:/# ping 192.168.1.200
PING 192.168.1.200 (192.168.1.200) 56(84) bytes of data:
64 bytes from 192.168.1.200: icmp_seq=1 ttl=62 time=0.072 ms
64 bytes from 192.168.1.200: icmp_seq=2 ttl=62 time=0.125 ms
64 bytes from 192.168.1.200: icmp_seq=3 ttl=62 time=0.176 ms
64 bytes from 192.168.1.200: icmp_seq=4 ttl=62 time=0.117 ms
64 bytes from 192.168.1.200: icmp_seq=5 ttl=62 time=0.129 ms
64 bytes from 192.168.1.200: icmp_seq=6 ttl=62 time=0.128 ms
64 bytes from 192.168.1.200: icmp_seq=7 ttl=62 time=0.123 ms
64 bytes from 192.168.1.200: icmp_seq=8 ttl=62 time=0.123 ms
64 bytes from 192.168.1.200: icmp_seq=9 ttl=62 time=0.119 ms
64 bytes from 192.168.1.200: icmp_seq=10 ttl=62 time=0.112 ms
^C
--- 192.168.1.200 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9217ms
rtt min/avg/max/mdev = 0.072/0.122/0.176/0.025 ms
root@pc2:/#
```

- pc1 và pc2 gửi được ICMP echo request đến routerISP; routerISP KHÔNG gửi được ICMP echo reply đến pc2 và pc3. Sinh viên tìm hiểu và lý giải nguyên nhân điều này. Gợi ý: kiểm tra bảng vạch đường trên routerISP, pc2 và pc3.

5) Kết thúc Bài Tập 14. Dùng lệnh `lwiipe` để hủy mạng ảo

```
$ kathara wipe
b1812282@b1812282-VirtualBox:~/Student/B1812282_workspace/BaiTap14$ kathara wipe
Are you sure to wipe Kathara? (y/n) y
```