

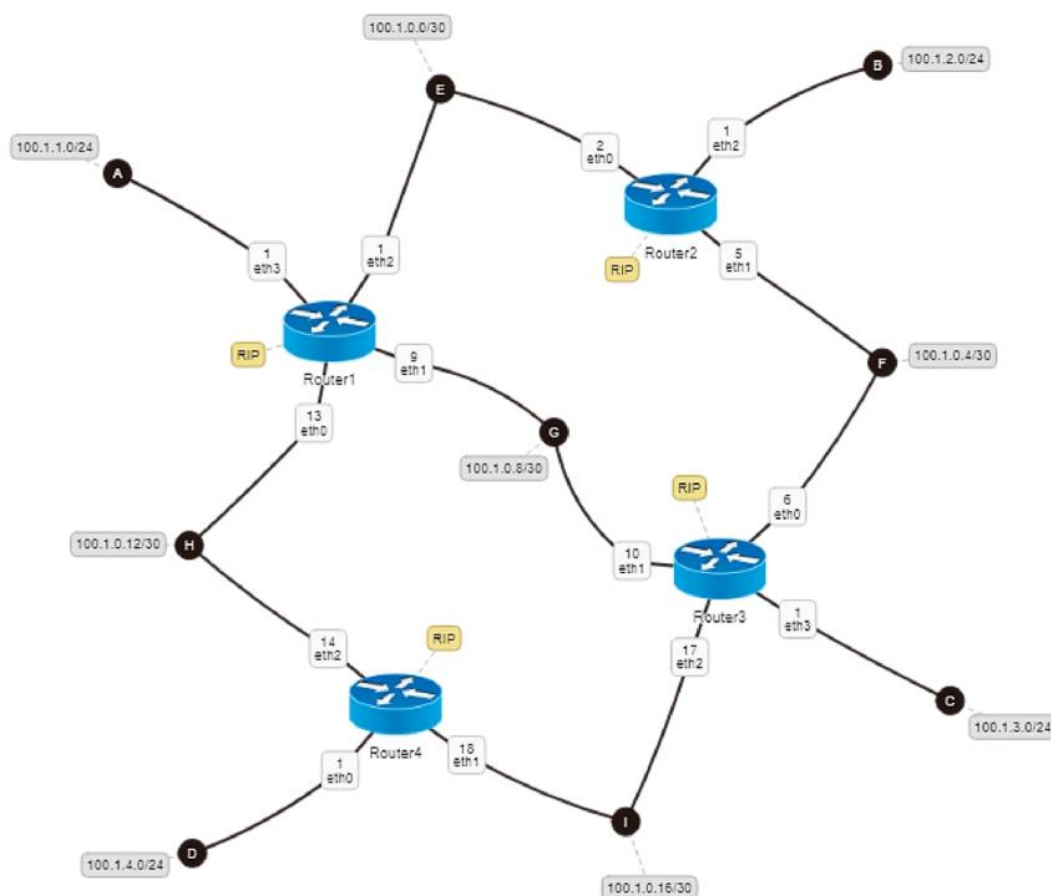
MẠNG MÁY TÍNH (CT112) - NHÓM 3

THỰC HÀNH BUỔI 4

II. BÀI TẬP THỰC HÀNH

BÀI TẬP 11: Các mạng trong miền kết nối bằng Router sử dụng giao thức RIPv2 vì router

Bước 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



Bước 2: Xây dựng cấu trúc thư mục mạng ảo (nằm dưới workspace /home/student/CT112/buoi4) với đầy đủ các thư mục con và các file cấu hình (.startup, lab.conf). Thư mục mạng ảo đặt tên là BT11. Sinh viên có thể tham khảo cấu trúc thư mục được Giảng viên hướng dẫn.

```
Terminal
student@may:~$ mkdir B2103427
student@may:~$ cd B2103427
student@may:~/B2103427$ mkdir buoi4
student@may:~/B2103427$ cd buoi4
student@may:~/B2103427/buoi4$ mkdir lab11
student@may:~/B2103427/buoi4$ cd lab11
student@may:~/B2103427/buoi4/lab11$ mkdir router1
student@may:~/B2103427/buoi4/lab11$ mkdir router2
student@may:~/B2103427/buoi4/lab11$ mkdir router3
student@may:~/B2103427/buoi4/lab11$ mkdir router4
student@may:~/B2103427/buoi4/lab11$ nano lab.conf
student@may:~/B2103427/buoi4/lab11$ nano router1.startup
student@may:~/B2103427/buoi4/lab11$ nano router2.startup
student@may:~/B2103427/buoi4/lab11$ nano router3.startup
student@may:~/B2103427/buoi4/lab11$ nano router4.startup
student@may:~/B2103427/buoi4/lab11$
```

Bước 3: Trên file lab.conf, soạn thảo nội dung mô tả hình thái mạng theo thiết kế. Nội dung tham khảo:

```
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
```

```
Terminal
GNU nano 6.2 lab.conf
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
```

Bước 4: Lần lượt 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:

Router1.startup

```
GNU nano 6.2 router1.startup
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
```

Router2.startup

```
GNU nano 6.2 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
```

Router3.startup

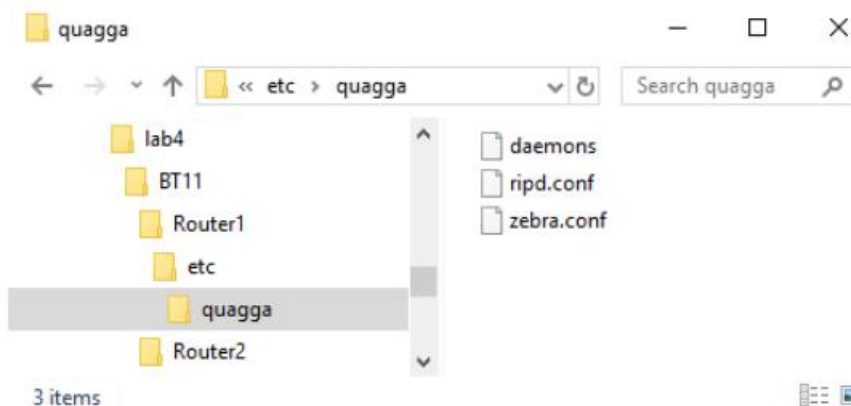
```
GNU nano 6.2 router3.startup
ifconfig eth0 100.1.0.6/30 up
ifconfig eth1 100.1.0.10/30 up
ifconfig eth2 100.1.0.17/30 up
ifconfig eth3 100.1.3.1/24 up
/etc/init.d/quagga start
```

Router4.startup

```
GNU nano 6.2 router4.startup
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
```

Ý nghĩa: Lệnh /etc/init.d/zebra start sẽ khởi động dịch vụ Zebra trên máy ảo. Để dịch vụ RIPv2 thực thi trên dịch vụ Zebra của mỗi máy ảo thì cần phải cấu hình cho dịch vụ theo hướng dẫn ở Bước 5

Bước 5: Trong mỗi thư mục máy ảo đã tạo ra, ví dụ: trong thư mục Router1, tạo cấu trúc thư mục được miêu tả như sau:



```
student@may:~/B2103427/buoi4/lab11$ cd router1
student@may:~/B2103427/buoi4/lab11/router1$ cd etc
student@may:~/B2103427/buoi4/lab11/router1/etc$ cd quagga
student@may:~/B2103427/buoi4/lab11/router1/etc/quagga$ nano daemons
student@may:~/B2103427/buoi4/lab11/router1/etc/quagga$ nano ripd.conf
student@may:~/B2103427/buoi4/lab11/router1/etc/quagga$ nano zebra.conf
```


Bước 6: Nếu sử dụng các cấu hình mặc nhiên của công cụ Zebra thì bỏ qua bước 6 này. Nếu không thì có thể sử dụng nội dung được miêu tả tham khảo sau dành cho file zebra.conf:

```
GNU nano 6.2 zebra.conf
hostname zebra
password zebra
enable password zebra
log file /var/log/zebra/zebra.log
```

Bước 7: Trên file daemons, thêm vào nội dung khai báo giải thuật vạch đường sử dụng bởi Router. Nội dung file daemons có thể tham khảo như sau. Lưu ý: ripd=yes là kích hoạt giải thuật RIPv2 trên Router.

```
GNU nano 6.2 daemons
zebra=yes
ripd=yes
```

Bước 8: Trên file ripd.conf, thêm vào nội dung cấu hình giao thức RIPv2 trên Zebra của các Router. Nội dung file ripd.conf có thể tham khảo như sau:

```
GNU nano 6.2 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
```

Bước 9: Khởi động mạng ảo BaiTap11. Trong quá trình các máy ảo khởi động, chú ý quan sát xem Zebra có được bật lên hay không và giao thức RIPv2 đã đi vào hoạt động hay chưa. Nếu chưa, quay trở lại các bước trên để kiểm tra.

```
student@may:~/B2103427/buoi4/lab11$ kathara lstart
```

Starting Network Scenario

[Deploying collision domains] 9/9

[Deploying devices] 4/4

```
student@may:~/B2103427/buoi4/lab11$
```

Bước 10: Sau khi mạng ảo khởi động khoảng 10 giây, hãy kiểm tra bảng vạch đường trên các Router bằng lệnh route. Nhận xét kết quả trên bảng vạch đường của các Router. Thực hiện một số lệnh ping giữa các giao diện của Router để kiểm tra tính liên thông của các mạng LAN.

Sử dụng lệnh route để kiểm tra vạch đường trên các máy ảo :

```
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@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        100.1.0.14      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         100.1.0.10      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: /  
--- 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@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: /

--- 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@router3:/# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
100.1.0.0         100.1.0.9       255.255.255.252 UG    20    0      0 eth1
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: /

--- 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
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         100.1.0.17      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:/#
```

Kiểm tra ping

Ping từ router1 qua router4

```
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=0.416 ms
64 bytes from 100.1.4.1: icmp_seq=2 ttl=64 time=0.234 ms
64 bytes from 100.1.4.1: icmp_seq=3 ttl=64 time=0.484 ms
64 bytes from 100.1.4.1: icmp_seq=4 ttl=64 time=0.478 ms
64 bytes from 100.1.4.1: icmp_seq=5 ttl=64 time=0.462 ms
^C
--- 100.1.4.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.234/0.414/0.484/0.093 ms
root@router1:/#
```

Ping từ router1 qua router2

```
root@router1:/# ping -c 3 100.1.0.2
PING 100.1.0.2 (100.1.0.2) 56(84) bytes of data.
64 bytes from 100.1.0.2: icmp_seq=1 ttl=64 time=0.554 ms
64 bytes from 100.1.0.2: icmp_seq=2 ttl=64 time=0.457 ms
64 bytes from 100.1.0.2: icmp_seq=3 ttl=64 time=0.438 ms

--- 100.1.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.438/0.483/0.554/0.050 ms
root@router1:/#
```

Ping từ router3 qua router1

```
root@router3:/# ping -c 3 100.1.0.13
PING 100.1.0.13 (100.1.0.13) 56(84) bytes of data.
64 bytes from 100.1.0.13: icmp_seq=1 ttl=64 time=0.457 ms
64 bytes from 100.1.0.13: icmp_seq=2 ttl=64 time=0.474 ms
64 bytes from 100.1.0.13: icmp_seq=3 ttl=64 time=0.448 ms

--- 100.1.0.13 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.448/0.459/0.474/0.010 ms
root@router3:/#
```

Ping từ router2 qua router3

```
root@router2:/# ping -c 3 100.1.0.6
PING 100.1.0.6 (100.1.0.6) 56(84) bytes of data.
64 bytes from 100.1.0.6: icmp_seq=1 ttl=64 time=0.423 ms
64 bytes from 100.1.0.6: icmp_seq=2 ttl=64 time=0.476 ms
64 bytes from 100.1.0.6: icmp_seq=3 ttl=64 time=0.468 ms

--- 100.1.0.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.423/0.455/0.476/0.023 ms
root@router2:/#
```

Bước 11: Trên Router bất kỳ, ví dụ: Router1, thực hiện lệnh:

`tcpdump -i any -w /shared/BaiTap11_Router1.pcap`

để lắng nghe các gói tin vạch đường RIPv2 được trao đổi giữa Router1 và các Router khác trong AS. Sau đó khoảng 20 giây thì dừng lại.

```
root@router1:/# tcpdump -i any -w /shared/lab11_router1.pcap
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
^C48 packets captured
48 packets received by filter
0 packets dropped by kernel
root@router1:/#
```


Bước 12: Trên Router bất kỳ, gõ lệnh: telnet localhost ripd. Nhập mật khẩu là zebra đã đặt ở Bước 8.

```
root@router3:/# telnet localhost ripd
Trying ::1...
Connection failed: Connection refused
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 1.2.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

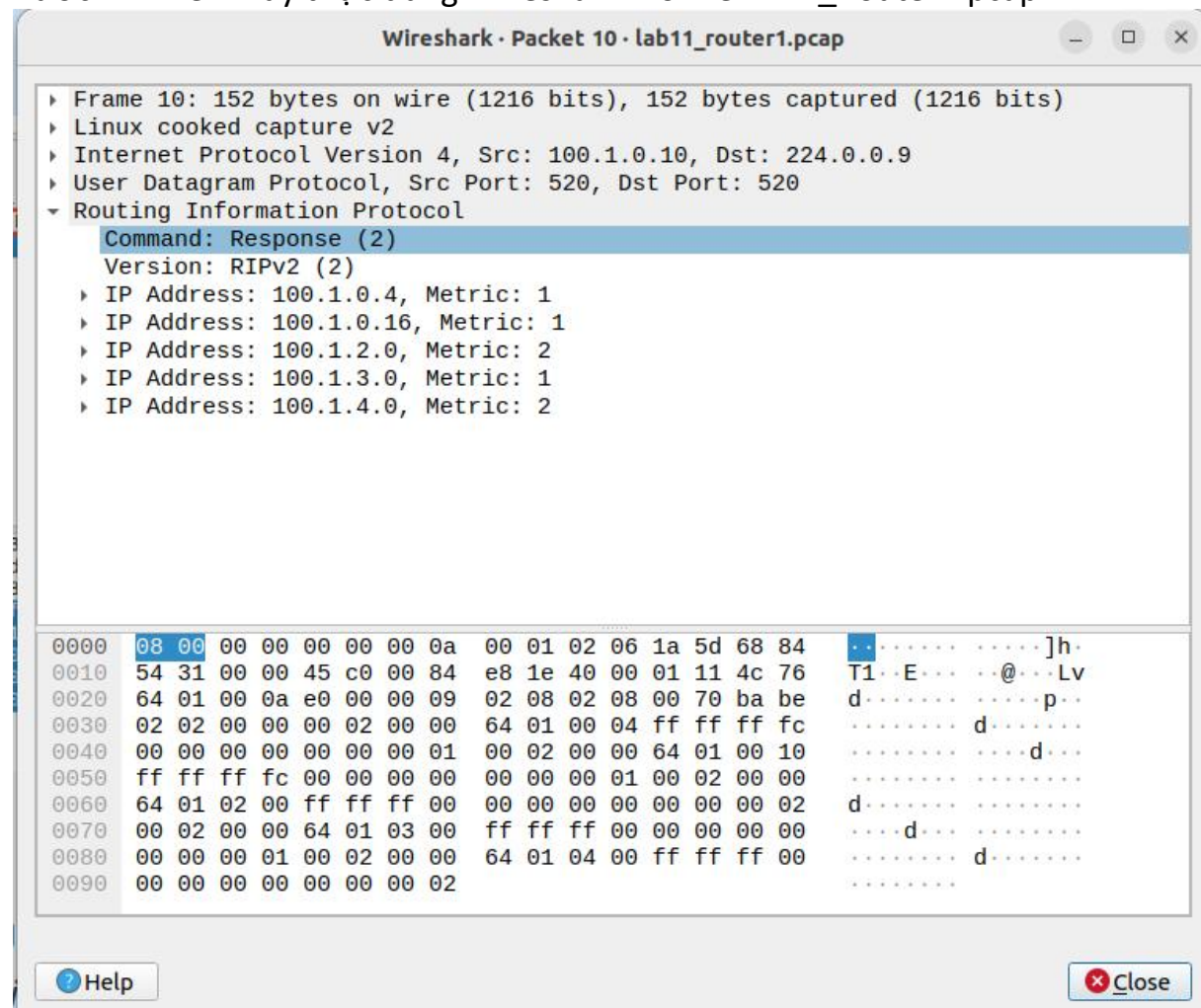
Password:
ripd> show ip rip
```

Bước 13: Gõ lệnh: show ip rip. So sánh kết quả hiển thị của lệnh này với lệnh route (Bảng vạch đường) trên Router. Chỉ ra một số thông tin có ích trên kết quả hiển thị.

```
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

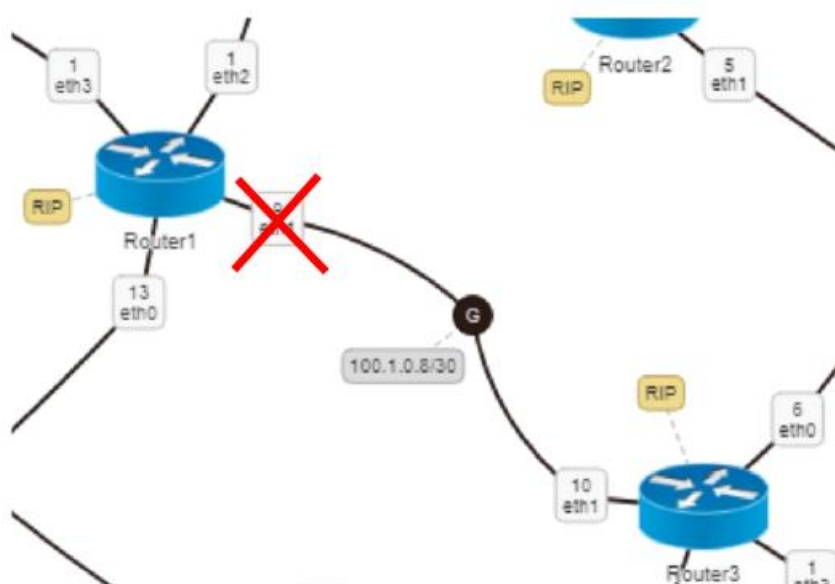
  Network          Next Hop          Metric From          Tag Time
R(n) 100.1.0.0/30   100.1.0.9         2 100.1.0.9          0 02:59
C(i) 100.1.0.4/30   0.0.0.0           1 self              0
C(i) 100.1.0.8/30   0.0.0.0           1 self              0
R(n) 100.1.0.12/30  100.1.0.9         2 100.1.0.9          0 02:59
C(i) 100.1.0.16/30  0.0.0.0           1 self              0
R(n) 100.1.1.0/24   100.1.0.9         2 100.1.0.9          0 02:59
R(n) 100.1.2.0/24   100.1.0.5         2 100.1.0.5          0 02:31
C(i) 100.1.3.0/24   0.0.0.0           1 self              0
R(n) 100.1.4.0/24   100.1.0.18        2 100.1.0.18         0 02:31
ripd>
```

Bước 14: Trên máy thực dùng Wireshark mở file BT11_Router1.pcap.



- Địa chỉ IP đích của khung này là 224.0.0.9, nó là địa chỉ Multicast.
- Gói tin sử dụng giao thức trên tầng vận chuyển là user datagram, và hoạt động ở cổng 520.
- Trong trường Routing Information Protocol đây là bộ định tuyến sử dụng giải thuật RIPv2.
- Ngoài gói tin RIPv2 có địa chỉ 100.1.0.10 thì router3 còn nhận dữ liệu của các router 1, routere2 và router4.

Bước 15: Trên Router1, tắt đi giao diện eth1 bằng lệnh: ifconfig eth1 down. Hình ảnh minh họa cho việc tắt đi giao diện eth1:



lệnh: `ifconfig eth1 down`.

```
0 packets dropped by kernel
root@router1:/# ifconfig eth1 down
root@router1:/#
```

Bước 16: Trên Router1, thực hiện lệnh ping đến 100.1.0.10 (Router3) và dừng lại sau 5 giây. Nhận xét kết quả và giải thích bằng kết quả hiển thị từ lệnh `route`.

```
root@router1:/# ping -c 3 100.1.0.10
PING 100.1.0.10 (100.1.0.10) 56(84) bytes of data.

--- 100.1.0.10 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2028ms

root@router1:/#
```

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

- Giao diện eth2 của router1 sẽ thay thế giao diện eth1 đã bị tắt đi

Bước 17: Đợi khoảng 1 phút và thực hiện lại lệnh ping đến 100.1.0.10 trên Router1. Nhận xét kết quả và giải thích bằng kết quả hiển thị từ lệnh `route`. Hãy cho biết lúc này để gửi dữ liệu từ Router1 đến

100.1.0.10 thì giao diện nào (eth0, eth2 hay eth3) trên Router1 đã được sử dụng thay thế cho giao diện eth1 bị tắt đi?

Bước 18: Kết luận về hoạt động vạch đường bằng giải thuật RIPv2 trên Router khi hình trạng mạng thay đổi, chẳng hạn: 1 giao diện bị tắt, 1 Router ngưng hoạt động...

=> Giải thuật RIPv2 trên router tự động cập nhật lại đường đi khi cấu trúc mạng thay đổi để đảm bảo dữ liệu đường truyền tải với đường đi ngắn nhất và không bị mất mát dữ liệu.

Bước 19: Kết thúc Bài tập 11 và hủy đi mạng ảo bằng lệnh kathara lclean.

```
student@may:~/B2103427/buoi4/lab11$ kathara lclean
```

Stopping Network Scenario

[Deleting devices]

4/4

[Deleting collision domains]

9/9

```
student@may:~/B2103427/buoi4/lab11$
```

BÀI TẬP 12: Xây dựng mô hình mạng sử dụng OSPFv2 Bước 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.

The diagram illustrates a network topology for OSPFv2 configuration. It features four routers: Router1, Router2, Router3, and Router4. Router1 is connected to Router2, Router3, and Router4. Router2 is connected to Router3. Router3 is connected to Router4. Each router has multiple interfaces (eth0, eth1, eth2, eth3) connected to various PCs (A, B, C, D, E, F, G, H, I) with specific IP addresses. OSPF is configured on all routers. A box labeled 'Area ID 0.0.0.0' is shown near Router1.

Router	Interface	Connected Device	IP Address
Router1	eth3	A	100.1.1.0/24
Router1	eth2	E	100.1.0.0/30
Router1	eth1	G	100.1.0.8/30
Router1	eth0	H	100.1.0.12/30
Router2	eth2	B	100.1.2.0/24
Router2	eth1	F	100.1.0.4/30
Router3	eth0	F	100.1.0.4/30
Router3	eth1	G	100.1.0.8/30
Router3	eth2	I	100.1.0.16/30
Router3	eth3	C	100.1.3.0/24
Router4	eth2	D	100.1.4.0/24
Router4	eth1	I	100.1.0.16/30

Bước 2: Xây dựng cấu trúc thư mục mạng ảo (nằm dưới workspace /home/student/your_workspace) với đầy đủ các thư mục con và các file cấu hình (.startup, lab.conf). Thư mục mạng ảo đặt tên là BaiTap12. Sinh viên có thể tham khảo cấu trúc thư mục được Giảng viên hướng dẫn.

```
student@may:~/B2103427/buoi4$ mkdir lab12
student@may:~/B2103427/buoi4$ cd lab12
student@may:~/B2103427/buoi4/lab12$ mkdir router1
student@may:~/B2103427/buoi4/lab12$ mkdir router2
student@may:~/B2103427/buoi4/lab12$ mkdir router3
student@may:~/B2103427/buoi4/lab12$ mkdir router4
student@may:~/B2103427/buoi4/lab12$ nano lab.conf
student@may:~/B2103427/buoi4/lab12$ nano router1.startup
student@may:~/B2103427/buoi4/lab12$ nano router2.startup
student@may:~/B2103427/buoi4/lab12$ nano router3.startup
student@may:~/B2103427/buoi4/lab12$ nano router4.startup
student@may:~/B2103427/buoi4/lab12$
```

Bước 3, 4: Thực hiện giống Bước 3 và Bước 4 của Bài Tập 11

File lab.conf

```
GNU nano 6.2
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
```

Router1.startup

```
GNU nano 6.2 router1.startup
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
```

Router2.startup

```
GNU nano 6.2 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
```

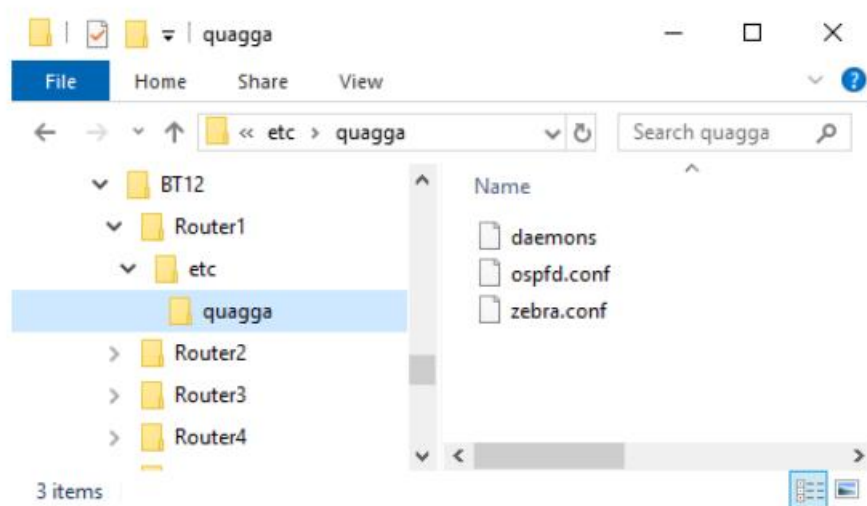
Router3.startup

```
GNU nano 6.2 router3.startup
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
```

Router4.startup

```
GNU nano 6.2 router4.startup
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
```

Bước 5: Trong mỗi thư mục máy ảo đã tạo ra, ví dụ : trong thư mục Router1, tạo cấu trúc thư mục được miêu tả như sau:



Bước 6: Tương tự Bước 6 trong Bài Tập 11. Lưu ý: Bước này có thể bỏ qua nếu muốn sử dụng các thiết lập mặc nhiên trên công cụ Zebra.

```
GNU nano 6.2 zebra.conf
hostname zebra
password zebra
enable password zebra

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

Bước 7: Trên file daemons, thêm vào nội dung khai báo giải thuật vạch đường sử dụng bởi Router. Nội dung file daemons có thể tham khảo như sau. Lưu ý: ospfd=yes là kích hoạt giải thuật OSPFv2 trên Router.

```
GNU nano 6.2 daemons
zebra=yes
ospfd=yes

bgpd=no
ospf6d=no
ripngd=no
```

Bước 8: Trên file ospfd.conf, thêm vào cấu hình giao thức OSPFv2 trên Zebra của các Router1. Nội dung file ospfd.conf của Router1 có thể tham khảo như sau. File ospfd.conf của các Router khác sinh viên tự thực hiện.

Router1

```
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@router1:/# /etc/init.d/quagga status

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:/#
```

Router2

```
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@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@router2:/#
```

Bước 11: Trên Router1, thực hiện lần lượt các lệnh sau:
tracert 100.1.0.5 (giao diện eth1 của Router2)
tracert 100.1.0.17 (giao diện eth2 của Router3)
tracert 100.1.0.10 (giao diện eth1 của Router3)

```
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)  0.207 ms  0.346 ms  0.343 ms
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)  0.754 ms  1.353 ms  1.305 ms
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)  0.742 ms  1.386 ms  1.340 ms
root@router1:/#
```

Quan sát và nhận xét về đường đi của dữ liệu từ Router1 đến các địa chỉ này dựa trên các chi phí trên giao diện của các Router đã thiết lập trước đó.

Nhận xét:

Nhờ giải thuật OSPFv2 tăng khả năng mở rộng và hội tụ nhanh mà router1 đi qua các nhánh mạng của router2 và router3 không có nhánh mạng trung gian nào.

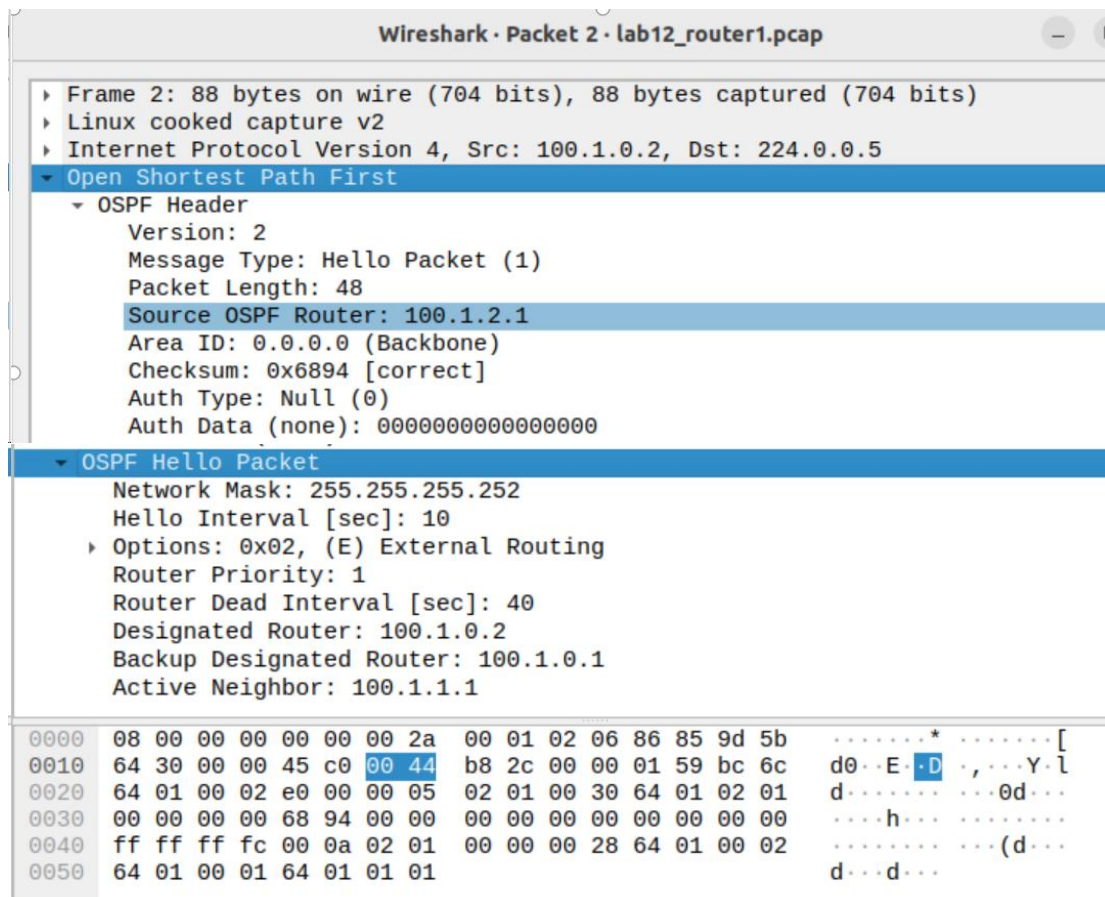
Bước 12: Trên Router bất kỳ, ví dụ: Router1, thực hiện lệnh:
tcpdump -i any -w /shared/BaiTap12_Router1.pcap
để lắng nghe các gói tin vạch đường OSPFv2 được trao đổi giữa Router1 và các Router khác trong AS. Sau đó khoảng 30 giây thì dừng lại.

```
root@router1:/# tcpdump -i any -w /shared/lab12_router1.pcap
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
^C56 packets captured
56 packets received by filter
0 packets dropped by kernel
root@router1:/#
```

Bước 12 (Không bắt buộc thực hiện): Trên Router bất kỳ, gõ lệnh: telnet localhost ospfd. Nhập mật khẩu là zebra đã đặt ở Bước 8.

Bước 13 (Không bắt buộc thực hiện): Gõ lệnh: show ip ospf route và show ip ospf database. Sinh viên tự tìm hiểu kết quả hiển thị từ 2 lệnh này.

Bước 14: Trên máy thực dùng Wireshark mở file BT12_Router1.pcap. Chọn một gói tin OSPFv2 đến từ địa chỉ đến từ một địa chỉ (giao diện) bất kỳ, chẳng hạn: 100.1.0.2 và trả lời các câu hỏi:



Địa chỉ Multicast nhận dữ liệu của khung này là bao nhiêu? Địa chỉ IP này được gọi là gì (Broadcast, Multicast...)

=> Địa chỉ IP của khung này là 224.0.0.5

=> Đây là địa chỉ Multicast

o Gói tin OSPFv2 này có sử dụng giao thức gì trên tầng vận chuyển hay không?

=> Gói tin này dùng giao thức Open Shortest Path First trên tầng vận chuyển

o Nội dung thông điệp mà của gói tin OSPFv2 này truyền đi là gì?

o Dựa vào phần nội dung thông điệp, hãy cho biết các hàng xóm đang hoạt động bình thường (Active Neighbor) của giao diện gửi đi gói tin. Đây cũng chính là thông tin thể hiện cho trạng thái nối kết (Link State) trên một giao diện mạng của Router.

=> Hiện thuật toán OSPF đang sử dụng là version 2

=> Active Neighbor là danh sách các bộ định tuyến lân cận hoặc liền kề.

=> Designated Router là bộ định tuyến liền kề với bộ định tuyến hiện tại

=>Backup Designated Router phục vụ như một chế độ chờ để nhận cập nhật các bộ định tuyến hiện tại.

Bước 15 (Không bắt buộc): Kiểm tra việc cập nhật bảng vạch đường bằng giải thuật OSPFv2 của các Router khi hình trạng mạng thay đổi, chẳng hạn: tắt đi 1 giao diện trên 1 Router, tắt đi 1 Router...

Bước 16: Kết thúc Bài tập 12 và hủy đi mạng ảo bằng lệnh kithara lclean.

--HẾT--