

Thiết kế & triển khai mạng IP

Bài thực hành số 2: Dynamic routing

1 Chuẩn bị môi trường

Hướng dẫn chi tiết:

1. RIP: <https://users.soict.hust.edu.vn/hoangph/textbook/ch01-4.html>
2. OSPF: <https://users.soict.hust.edu.vn/hoangph/textbook/ch01-5.html>

2 Cài đặt routing protocol cho router

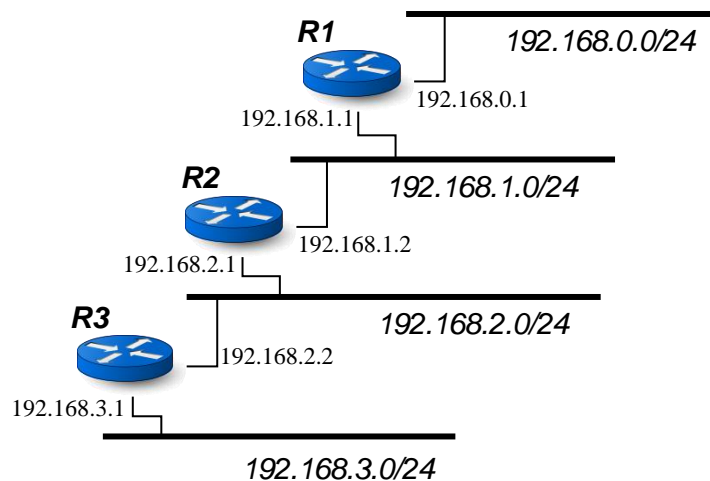
1. Thiết lập kết nối Internet cho máy router: thêm Network Adapter mới cho máy router và đặt kiểu kết nối là NAT
2. Khởi động router, kiểm tra kết nối bằng *ifconfig -a*
3. Xác định kết nối NAT vừa khởi tạo bằng cách so sánh địa chỉ MAC trong *ifconfig* và trong Virtualbox
4. Nếu kết nối mạng NAT chưa có địa chỉ IP, thực hiện gán IP động:
5. > *dhclient -s eth?*
6. Kiểm tra kết nối ra Internet:
> *ping 8.8.8.8*
7. Cài đặt trình xử lý routing động *quagga*:
> *yum install quagga*

Lưu ý: nếu xuất hiện lỗi không tìm được repository theo tên (ví dụ: Couldn't resolve host 'mirrorlist.centos.org') thì xử lý như sau (bước 8,9,10):

8. kiểm tra ping vào tên miền này
> *ping mirrorlist.centos.org*
9. Nếu ping không thành công thì có thể là lỗi DNS. Cần thiết lập lại DNS server là 8.8.8.8:
> *nano /etc/resolv.conf*
nameserver 8.8.8.8
10. Kiểm tra lại kết nối theo tên miền bằng lệnh *ping*, và chạy lại *yum* để cài đặt *quagga* (bước 7)
11. Cài đặt trình telnet để hỗ trợ làm việc với Quagga
> *yum install telnet*
12. Sau khi cài đặt thành công, shutdown router, bỏ Network Adapter kết nối Internet (NAT) để không bị ảnh hưởng khi triển khai các bước thực hành tiếp theo.

3 Thiết lập kết nối liên mạng bằng RIP

Sơ đồ mạng:



3.1 Cấu hình các router với RIP

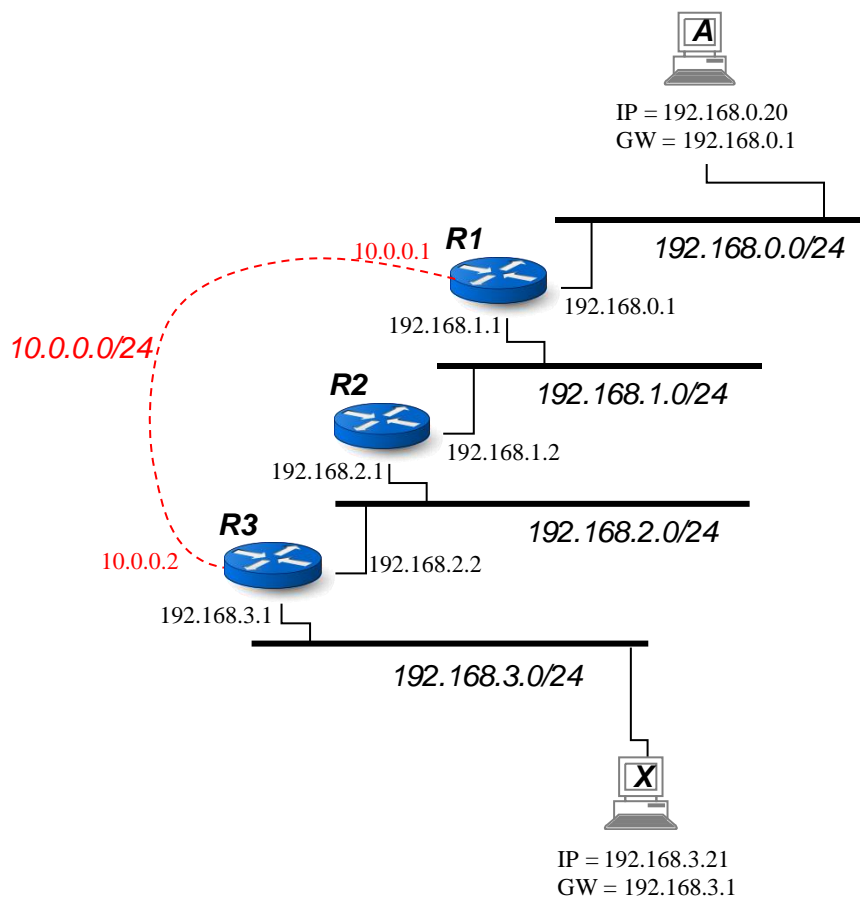
1. Cài đặt *quagga* cho tất cả các router
2. Cấu hình service *quagga*:
`> nano /etc/quagga/zebra.conf`
`hostname R2`
`password zebra`
`enable password zebra`
`log file /var/log/quagga/zebra.log`
3. Cấu hình service *ripd*:
`> nano /etc/quagga/ripd.conf`
`hostname R2`
`password zebra`
`router rip`
`network 192.168.0.0/24`
`network 192.168.1.0/24`
`network 192.168.2.0/24`
`network 192.168.3.0/24`
`log file /var/log/quagga/ripd.log`
4. Kiểm tra và cấu hình địa chỉ IP của các router bằng *ifconfig*
5. Kiểm tra kết nối giữa các router láng giềng bằng *ping*
6. Kiểm tra trạng thái *ip_forward* của các router
`> sysctl net.ipv4.ip_forward`
`net.ipv4.ip_forward = 1`
7. Tắt service *iptables* để các gói tin đi qua router không bị chặn
`> service iptables stop`
8. Bật service *zebra* và *ripd*:
`> service zebra start`
`> service ripd start`
9. Kiểm tra bảng routing, chú ý các đường định tuyến có Metric lớn hơn 1:
`> route -n`

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.3.0	192.168.1.2	255.255.255.0	UG	3	0	0	eth3
192.168.2.0	192.168.1.2	255.255.255.0	UG	2	0	0	eth3
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth3
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth2

10. Kiểm tra kết nối liên mạng bằng *ping*

3.2 Kiểm tra các tình huống đáp ứng topo mạng của RIP

Thêm kết nối trực tiếp giữa R1 và R3 và kiểm tra các thay đổi routing của RIP.



1. Thêm các trạm làm việc A trong mạng 192.168.0.0/24 và X trong mạng 192.168.3.0/24. Cấu hình IP & gateway. Kiểm tra kết nối từ các trạm này đến gateway tương ứng bằng *ping* và kiểm tra giữa A & X bằng *ping*.
2. Kiểm tra đường đi gói tin từ A sang X:

A> *tracert 192.168.3.21*

```
[root@A ~]# tracert 192.168.3.21
 1?: [LOCALHOST] pmtu 1500
 1: 192.168.0.1 (192.168.0.1) 0.487ms
 1: 192.168.0.1 (192.168.0.1) 0.410ms
 2: 192.168.1.2 (192.168.1.2) 0.833ms
 3: 192.168.2.2 (192.168.2.2) 1.034ms
 4: 192.168.3.21 (192.168.3.21) 3.371ms !H
Resume: pmtu 1500
```

3. Bổ sung Network Adapter cho router R1 & R3, đặt kết nối là Internal Network "serial00"
4. Thiếp lập địa chỉ IP mạng 10.0.0.0/24 cho các kết nối của router R1 & R3
R1> *ifconfig eth4 10.0.0.1/24*
R3> *ifconfig eth6 10.0.0.1/24*
5. Kiểm tra kết nối trực tiếp giữa R1 và R3 qua đường serial vừa kết nối bằng lệnh *ping -I eth4* (tham số -I để chỉ định kết nối mạng cho *ping* sử dụng):
R1> *ping 10.0.0.2 -I eth4*

```
[root@R1 ~]# ping 10.0.0.2 -I eth4
PING 10.0.0.2 (10.0.0.2) from 10.0.0.1 eth4: 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.467 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.680 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.485 ms
```

6. Bổ sung mạng 10.0.0.0/24 vào danh sách các mạng có thể được RIP phục vụ và khởi động lại service ripd:

```
R1> nano /etc/quagga/ripd.conf
      thêm dòng: network 10.0.0.0/24
R1> service ripd restart
```

7. Kiểm tra bảng routing được cập nhật trên R1 và R3:

```
R1> route -n
```

```
[root@R1 ~]# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.3.0 10.0.0.2 255.255.255.0 UG 2 0 0 eth4
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth4
192.168.2.0 192.168.1.2 255.255.255.0 UG 2 0 0 eth3
192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth2
```

```
R3> route -n
```

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth6
192.168.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth5
192.168.2.0 0.0.0.0 255.255.255.0 U 0 0 0 eth4
192.168.1.0 192.168.2.1 255.255.255.0 UG 2 0 0 eth4
192.168.0.0 10.0.0.1 255.255.255.0 UG 2 0 0 eth6
```

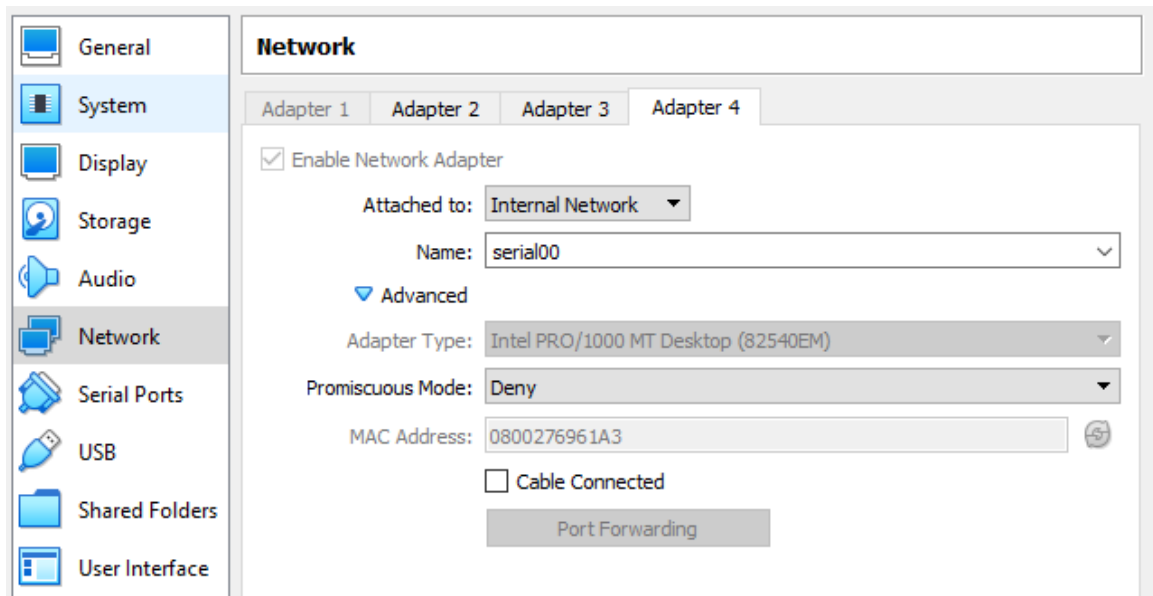
8. Kiểm tra đường đi gói tin từ A sang X:

```
A> tracepath 192.168.3.21
```

```
[root@A ~]# tracepath 192.168.3.21
1?: [LOCALHOST] pmtu 1500
1: 192.168.0.1 (192.168.0.1) 0.430ms
1: 192.168.0.1 (192.168.0.1) 0.384ms
2: 10.0.0.2 (10.0.0.2) 0.731ms
3: 192.168.3.21 (192.168.3.21) 0.997ms !H
Resume: pmtu 1500
```

➔ Đường đi từ A đến X được RIP cập nhật theo mạng 10.0.0.0/24 (thay vì qua R2 như trước)

9. Ngắt kết nối serial00 giữa R1 và R3 bằng cách vào VirtualBox, mở network setting của R1 hoặc R3, chọn kết nối serial và bỏ check "Cable Connected"



10. Đợi một khoảng thời gian để RIP cập nhật routing theo topo mạng mới, hoặc restart service *ripd*:
`R1> service ripd restart`

11. Xem bảng routing trên R1 & R3:

`R1> route -n`

```
[root@R1 ~]# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.3.0      192.168.1.2    255.255.255.0   UG      3      0      0 eth3
10.0.0.0         0.0.0.0        255.255.255.0   U        0      0      0 eth4
192.168.2.0      192.168.1.2    255.255.255.0   UG      2      0      0 eth3
192.168.1.0      0.0.0.0        255.255.255.0   U        0      0      0 eth3
192.168.0.0      0.0.0.0        255.255.255.0   U        0      0      0 eth2
```

12. Kiểm tra đường đi gói tin từ A sang X:

`A> tracepath 192.168.3.21`

```
[root@A ~]# tracepath 192.168.3.21
 1?: [LOCALHOST] pmtu 1500
 1: 192.168.0.1 (192.168.0.1) 0.487ms
 1: 192.168.0.1 (192.168.0.1) 0.410ms
 2: 192.168.1.2 (192.168.1.2) 0.833ms
 3: 192.168.2.2 (192.168.2.2) 1.034ms
 4: 192.168.3.21 (192.168.3.21) 3.371ms !H
Resume: pmtu 1500
```

➔ đường đi từ R1 đến mạng 192.168.3.0/24 được cập nhật lại theo đường qua R2

3.3 Bắt các gói tin RIP với iptables trên router R2

1. Bật service *iptables* trên R2 để bắt gói tin
`R2> service iptables start`
2. Kiểm tra các luật firewall (cấm gói tin) của *iptables*

R2> iptables -L --line-number

Chain INPUT (policy ACCEPT)						
num	target	prot	opt	source	destination	
1	ACCEPT	all	--	anywhere	anywhere	state RELATED,ESTABLISHED
2	ACCEPT	icmp	--	anywhere	anywhere	
3	ACCEPT	all	--	anywhere	anywhere	
4	ACCEPT	tcp	--	anywhere	anywhere	state NEW tcp dpt:ssh
5	REJECT	all	--	anywhere	anywhere	reject-with icmp-host-prohibited
Chain FORWARD (policy ACCEPT)						
num	target	prot	opt	source	destination	
1	REJECT	all	--	anywhere	anywhere	reject-with icmp-host-prohibited
Chain OUTPUT (policy ACCEPT)						
num	target	prot	opt	source	destination	

3. Có 2 luật cấm gói tin trong chain FORWARD và INPUT cần xóa (RIP hoạt động kiểu lan tỏa – propagation - các bảng routing nên cần bỏ cả luật cấm trong chain INPUT)

R2> iptables -D FORWARD 1

R2> iptables -D INPUT 5

4. Khởi động lại service *ripd* và check bảng routing đã được RIP xây dựng:

R2> service ripd restart

R2> route -n

Kernel IP routing table						
Destination	Gateway	Genmask	Flags	Metric	Ref	Use Iface
192.168.3.0	192.168.2.2	255.255.255.0	UG	2	0	0 eth2
192.168.2.0	0.0.0.0	255.255.255.0	U	0	0	0 eth2
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0 eth3
192.168.0.0	192.168.1.1	255.255.255.0	UG	2	0	0 eth3

5. Thêm luật log gói tin ở chain INPUT (chú ý là RIP gửi & nhận gói tin giữa các router láng giềng chứ không forward qua router, vậy nên log gói tin cần đặt tại điểm INPUT chứ không phải FORWARD):

R2> iptables -A INPUT -j LOG

6. Bắt gói tin RIP trong /var/log/messages:

R2> tail -f /var/log/messages

```
Apr  3 18:07:49 R1 kernel: IN=eth3 OUT= MAC= SRC=192.168.1.2 DST=224.0.0.9 LEN=72 TOS=0x00 PREC=0xC0
TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=52
Apr  3 18:07:57 R1 kernel: IN=eth3 OUT= MAC=01:00:5e:00:00:09:08:00:27:d4:68:44:08:00 SRC=192.168.1.
1 DST=224.0.0.9 LEN=52 TOS=0x00 PREC=0xC0 TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=32
Apr  3 18:08:13 R1 kernel: IN=eth2 OUT= MAC= SRC=192.168.2.1 DST=224.0.0.9 LEN=72 TOS=0x00 PREC=0xC0
TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=52
Apr  3 18:08:13 R1 kernel: IN=eth3 OUT= MAC= SRC=192.168.1.2 DST=224.0.0.9 LEN=72 TOS=0x00 PREC=0xC0
TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=52
Apr  3 18:08:21 R1 kernel: IN=eth3 OUT= MAC=01:00:5e:00:00:09:08:00:27:d4:68:44:08:00 SRC=192.168.1.
1 DST=224.0.0.9 LEN=52 TOS=0x00 PREC=0xC0 TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=32
Apr  3 18:08:23 R1 kernel: IN=eth2 OUT= MAC=01:00:5e:00:00:09:08:00:27:42:c9:60:08:00 SRC=192.168.2.
2 DST=224.0.0.9 LEN=52 TOS=0x00 PREC=0xC0 TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=32
Apr  3 18:08:44 R1 kernel: IN=eth2 OUT= MAC= SRC=192.168.2.1 DST=224.0.0.9 LEN=72 TOS=0x00 PREC=0xC0
TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=52
Apr  3 18:08:44 R1 kernel: IN=eth3 OUT= MAC= SRC=192.168.1.2 DST=224.0.0.9 LEN=72 TOS=0x00 PREC=0xC0
TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=52
Apr  3 18:08:45 R1 kernel: IN=eth2 OUT= MAC=01:00:5e:00:00:09:08:00:27:42:c9:60:08:00 SRC=192.168.2.
2 DST=224.0.0.9 LEN=52 TOS=0x00 PREC=0xC0 TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=32
Apr  3 18:08:46 R1 kernel: IN=eth2 OUT= MAC=01:00:5e:00:00:09:08:00:27:42:c9:60:08:00 SRC=192.168.2.
2 DST=224.0.0.9 LEN=52 TOS=0x00 PREC=0xC0 TTL=1 ID=0 DF PROTO=UDP SPT=520 DPT=520 LEN=32
```

➔ các gói tin RIP là các gói UDP với cổng = 520 được gửi & nhận giữa các router láng giềng. Do iptables chỉ log phần header mà gói tin RIP nằm ở tầng Application (trên UDP) nên không nhìn thấy được nội dung các RIP message

3.4 Bắt các gói tin RIP với log của service ripd trên router R2

Thêm dòng cấu hình bật chức năng log gói tin RIP trong file cấu hình của service *ripd* trên R2

```
R2> nano /etc/quagga/ripd.conf
```

```
debug rip event
```

```
debug rip package
```

```
log file /var/log/quagga/ripd.log
```

```
....
```

1. Khởi động lại service *ripd* để áp dụng file cấu hình mới trên R2

```
R2> service ripd restart
```

2. Xem các message RIP trên router R2

```
R2> tailf -f /var/log/quagga/ripd.log
```

```
2020/04/03 21:19:03 RIP: RECV RESPONSE version 2 packet size 64
2020/04/03 21:19:03 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:03 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:03 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:19:17 RIP: RECV RESPONSE version 2 packet size 64
2020/04/03 21:19:17 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:17 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:19:17 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:23 RIP: rip_send_packet 192.168.2.1 > 224.0.0.9 (eth2)
2020/04/03 21:19:23 RIP: SEND RESPONSE version 2 packet size 44
2020/04/03 21:19:23 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:19:23 RIP: 192.168.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:23 RIP: rip_send_packet 10.0.4.15 > 224.0.0.9 (eth2)
2020/04/03 21:19:23 RIP: SEND RESPONSE version 2 packet size 64
2020/04/03 21:19:23 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:19:23 RIP: 192.168.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:23 RIP: 192.168.2.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:23 RIP: rip_send_packet 192.168.1.2 > 224.0.0.9 (eth3)
2020/04/03 21:19:23 RIP: SEND RESPONSE version 2 packet size 64
2020/04/03 21:19:23 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:19:23 RIP: 192.168.2.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:19:23 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
```

➔ các message RIP được gửi & nhận giữa R2 và R1,R3. RIP version 2 được sử dụng, từng RTE (Route Table Entry) được gửi qua lại

3.5 Phân tích xử lý Route Poisoning

1. Thiết lập topo mạng bao gồm có đường kết nối serial giữa R1 và R3, kiểm tra đường đi serial này được RIP thiết lập trên các router R1 & R3 (xem lại mục 3.2).

```
R1> route -n
```

```
[root@R1 ~]# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.3.0 10.0.0.2 255.255.255.0 UG 2 0 0 eth4
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth4
192.168.2.0 192.168.1.2 255.255.255.0 UG 2 0 0 eth3
192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth3
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth2
```

```
R3> route -n
```

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth6
192.168.3.0	0.0.0.0	255.255.255.0	U	0	0	0	eth5
192.168.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth4
192.168.1.0	192.168.2.1	255.255.255.0	UG	2	0	0	eth4
192.168.0.0	10.0.0.1	255.255.255.0	UG	2	0	0	eth6

- Ngắt kết nối serial giữa R1 & R3 và theo dõi log message RIP trên R2 (có thể phải đợi một khoảng thời gian để R1 và R3 phát hiện hết nối bị hỏng):

R2> tailf -f /var/log/quagga/ripd

```

....
2020/04/03 21:21:37 RIP: RECV RESPONSE version 2 packet size 64
2020/04/03 21:21:37 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:37 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
2020/04/03 21:21:37 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:37 RIP: RECV RESPONSE version 2 packet size 64
2020/04/03 21:21:37 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:37 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:37 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
2020/04/03 21:21:42 RIP: RECV RESPONSE version 2 packet size 44
2020/04/03 21:21:42 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:42 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
2020/04/03 21:21:42 RIP: RECV RESPONSE version 2 packet size 64
2020/04/03 21:21:42 RIP: 10.0.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
2020/04/03 21:21:42 RIP: 192.168.0.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
2020/04/03 21:21:42 RIP: 192.168.3.0/24 -> 0.0.0.0 family 2 tag 0 metric 1

```

→ xuất hiện các RTE có metric=16 được gửi từ R1 và R3 đến cho R2.

3.6 RIP Advance (bài thực hành tham khảo, không bắt buộc)

Link tham khảo: <https://www.quagga.net/docs/quagga.html#Zebra>
<https://www.quagga.net/docs/quagga.html#RIP>

- Quagga & Ripd hỗ trợ các cấu hình bổ sung để hỗ trợ RIP xử lý các vấn đề nâng cao như bảo mật, khai báo các mạng nghiệp vụ, tối ưu không gian RIP, v.v... Để truy nhập đến các chức năng này, cần cài đặt trình telnet và login vào service zebra & ripd để thực hiện các lệnh (giống các lệnh trên thiết bị router).
- Cài đặt trình *telnet* để làm việc với các service:
 > `yum install telnet`
- Các thông số *password* trong file cấu hình zebra và ripd dùng để login vào giao diện Vty và cấu hình online cho các service này. Zebra sử dụng cổng 2601, ripd sử dụng cổng 2602. Sau khi login vào giao diện Vty này, ta có thể cấu hình router hoặc xem trạng thái router với các lệnh router:


```
R2> telnet 127.0.0.1 2601
```

```
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.15).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

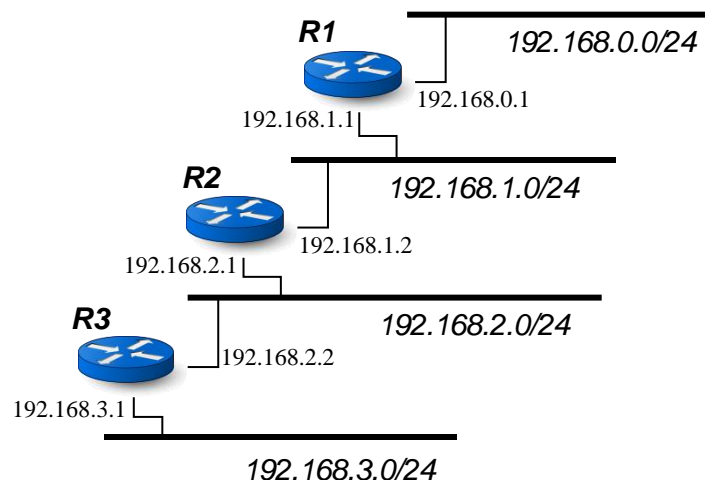
User Access Verification

Password:
R1> show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

R>* 10.0.0.0/24 [120/2] via 192.168.2.2, eth2, 00:28:51
C>* 10.0.4.0/24 is directly connected, eth2
C>* 127.0.0.0/8 is directly connected, lo
R>* 192.168.0.0/24 [120/2] via 192.168.1.1, eth3, 00:28:51
C>* 192.168.1.0/24 is directly connected, eth3
```

4 Kết nối liên mạng với OSPF Single Area

Tiếp tục sử dụng sơ đồ mạng như trong phần trên:



Phần này cần thao tác nhiều với các kết nối mạng của router nên xử lý qui tắc đặt tên cho dễ nhớ:

- Đặt qui tắc gán tên 4 kết nối mạng của máy ảo theo địa chỉ MAC:
> nano /etc/udev/rules.d/70-persistent-net.rules
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="?:?:?:?:?:?:01", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="?:?:?:?:?:?:02", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="?:?:?:?:?:?:03", NAME="eth3"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="?:?:?:?:?:?:04", NAME="eth4"
- Sửa địa chỉ MAC của các router theo qui tắc 00:00:00:00:0x:0y trong đó x là mã số router, y là mã số kết nối mạng. Ví dụ:
R1.Network Adapter 1 → 00:00:00:00:01:01

R2.Network Adapter 3 ➔ 00:00:00:00:02:03

3. Reboot lại router, các kết nối mạng sẽ được đặt tên theo đúng qui tắc
4. Thiết lập địa chỉ IP tự động cho từng kết nối mạng:

```
R2> nano /etc/sysconfig/networ-scripts/ifcfg-eth1
```

```
DEVICE = eth1
```

```
NAME = eth1
```

```
HWADDR=00:00:00:00:02:01
```

```
ONBOOT=yes
```

```
IPADDR=...
```

```
NETMASK=...
```

```
R2> service network restart
```

5. Thiết lập tên máy ảo:

```
R2> nano /etc/sysconfig/network
```

```
NETWORKING=yes
```

```
HOSTNAME=R2
```

4.1 Cấu hình các router OSPF trong một area

1. Cấu hình service *ospfd* trên từng router:

```
> nano /etc/quagga/ospfd.conf
```

```
hostname R1
```

```
password zebra
```

```
router ospf
```

```
ospf router-id 1.1.1.1
```

```
network 192.168.0.0/24 area 1
```

```
network 192.168.1.0/24 area 1
```

```
network 192.168.2.0/24 area 1
```

```
network 192.168.3.0/24 area 1
```

```
network 10.0.0.0/24 area 1
```

```
debug ospf event
```

```
debug ospf packet all
```

```
log file /var/log/quagga/ospfd.log
```

2. Tắt service *iptables* & *ripd*

```
> service iptables stop
```

```
> service ripd stop
```

3. Bật service *zebra* và *ospfd*:

```
> service zebra start
```

```
> service ospfd start
```

4. Kiểm tra bảng routing, chú ý giá trị Metric trên các đường định tuyến:

```
> route -n
```

Kernel IP routing table						
Destination	Gateway	Genmask	Flags	Metric	Ref	Use Iface
192.168.3.0	10.0.0.2	255.255.255.0	UG	20	0	0 eth2
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0 eth2
192.168.2.0	10.0.0.2	255.255.255.0	UG	20	0	0 eth2
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0 eth3
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0 eth4

4.2 Kiểm tra tính đáp ứng link state của OSPF

- Với routing table của R1 như trên, đường đi gói tin từ R1 đến mạng 192.168.3.0/24 là đi qua R3 link serial (địa chỉ 10.0.0.2). Tổng cost đến mạng đích này là Metric = 20.
Giải thích: các kết nối mạng Internal Network trong VirtualBox mặc định thiết lập băng thông 10Mbps (cost = 10), đường đi từ R1 đến mạng nghiệp vụ 192.168.3.0/24 phải qua 2 bước:
 - R1 == (serial line) == > R3: cost = 10 (LSA type=1, link type point-to-point)
 - R3 == (stub network) == > net 192.168.3.0/24: cost = 10 (LSA type=1, link type stub net)→ Tổng cost đường đi = 10 + 10 = 20
- Sử dụng telnet để thiết lập cost cho serial link trên R1 theo cấu hình đường T1 (1.544 Mbps – cost = 64)

```
R1> telnet 127.0.0.1 2604
```

```
R1> password: zebra
```

```
R1> enable
```

← bật chế độ cấu hình router

```
R1> password: zebra
```

```
R1#> configure terminal
```

← chọn cấu hình kết nối mạng

```
R1(config)> interface eth2
```

← chú ý kiểm tra kết nối mạng serial, ở đây là eth2

```
R1(config-if)> ospf cost 64
```

← thiết lập cost của liên kết này là 64

```
R1> exit
```

- Kiểm tra bảng routing, chú ý đường đi từ R1 đến mạng 192.168.3.0/24:

```
R1> route -n
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.3.0	192.168.1.2	255.255.255.0	UG	30	0	0	eth3
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth2
192.168.2.0	192.168.1.2	255.255.255.0	UG	20	0	0	eth3
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth3
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth4

→ đường đi từ R1 đến mạng 192.168.3.0/24 đã được thay đổi, không qua link serial nữa mà qua R2 (192.168.1.2) và cost=30.

Giải thích: khi thiết lập lại cost của đường serial thành 64, đường đi cũ từ R1 đến net#3 (qua R3 serial line) có tổng cost là 64 + 10 = 74, lớn hơn đường đi qua R2 với tổng cost là 30 (R1 == > R2 - cost = 10, R2 == > R3 - cost = 10, R3 == > stub net - cost = 10)

- Ngắt kết nối của R2 đến lan01 hoặc lan02 rồi đợi một khoảng thời gian để ospfd cập nhật lại link state DB, hoặc restart service ospfd trên R2:

```
R2> service ospfd restart
```

- Kiểm tra bảng routing trên R1:

```
R1> route -n
```

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.3.0	10.0.0.2	255.255.255.0	UG	74	0	0	eth2
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth2
192.168.2.0	192.168.1.2	255.255.255.0	UG	20	0	0	eth3
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth3
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth4

→ đường đi từ R1 đến mạng 192.168.3.0/24 đã được thay đổi, không qua R2 nữa mà quay lại link serial (10.0.0.2) và cost=74 (64 + 10)

4.3 Xem các dữ liệu OSPF

1. Kết nối telnet vào router bất kỳ và xem bảng routing với route cost :

R1> telnet 127.0.0.1 2604

R1> password: zebra

R1> show ip ospf route

```
===== OSPF network routing table =====
N   10.0.0.0/24          [64] area: 0.0.0.1
    directly attached to eth2
N   192.168.0.0/24      [10] area: 0.0.0.1
    directly attached to eth4
N   192.168.1.0/24      [10] area: 0.0.0.1
    directly attached to eth3
N   192.168.2.0/24      [74] area: 0.0.0.1
    via 10.0.0.2, eth2
N   192.168.3.0/24      [74] area: 0.0.0.1
    via 10.0.0.2, eth2

===== OSPF router routing table =====
===== OSPF external routing table =====
```

2. Xem thông tin các router láng giềng, chú ý router ID được tự động thiết lập bằng giá trị địa chỉ IP lớn nhất trong các network interface của nó:

R1> show ip ospf neighbor

.....

3. Xem thông tin các kết nối mạng của router, chú ý kiểm tra loại kết nối (link type) của từng kết nối mạng. Mặc định các kết nối mạng ban đầu đều được thiết lập là broadcast, có thể đổi sang point-to-point như khi đổi cost của kết nối mạng.

R1> show ip ospf interface

.....

4. Xem thông tin văn bản link-state database, chú ý các dữ liệu LS age được tăng theo từng giây, LS sequence chỉ tăng khi có phiên bản LSA mới:

R1> show ip ospf database

.....

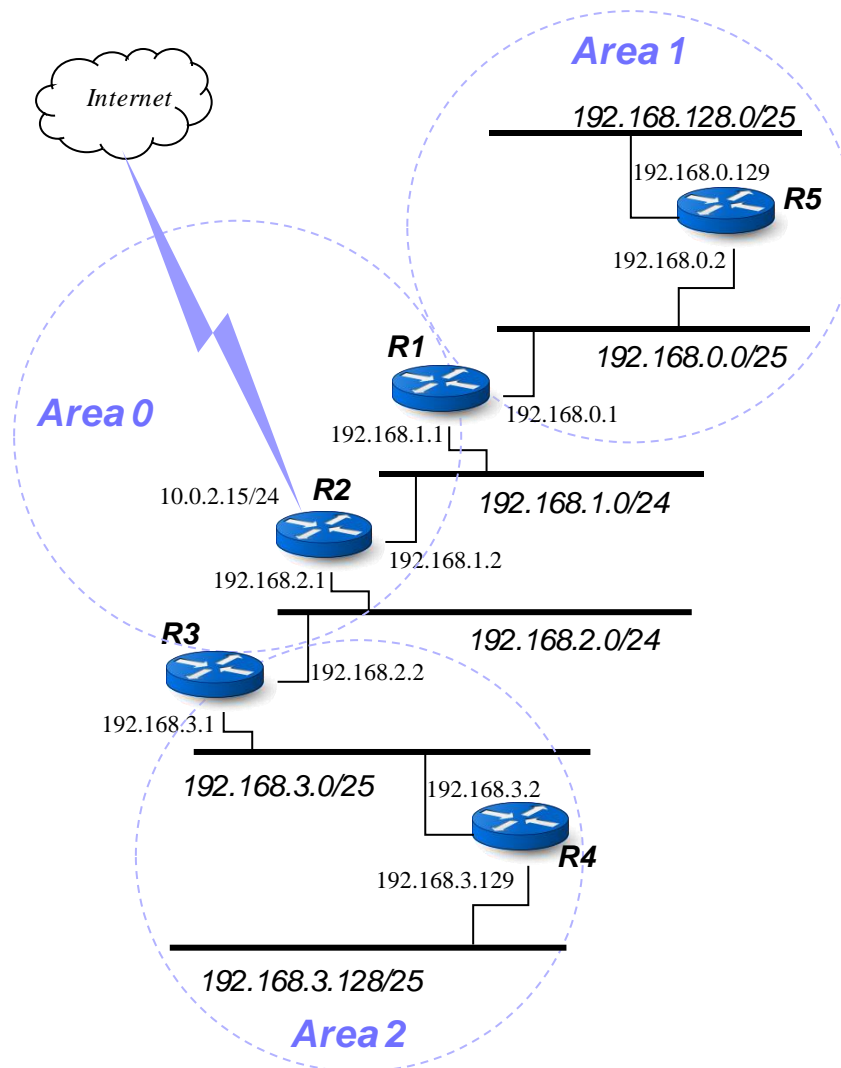
5. Xem thông tin chi tiết link-state database theo từng router, chú ý số lượng link của mỗi router và thông tin trạng thái (cost) của từng link này

R1> show ip ospf database router

.....

5 Thiết lập kết nối liên mạng bằng OSPF Multi Area

Sơ đồ mạng:



1. R2 sử dụng Host-only Network Adapter kết nối với máy host để giả lập External Network (Internet):
Host-only network: 192.168.56.0/24. R2: 192.168.56.101. Host machine: 192.168.56.1
2. Cấu hình ospf trên R2 với kết nối external (redistribute connected):

```
R2> nano /etc/quagga/ospfd.conf
router ospf
  ospf router-id 2.2.2.2
  redistribute connected
  . . . . .
```
3. Kiểm tra External-LSA được lan truyền & save trong OSPF LS Database:

```
R2> telnet 127.0.0.1 2604
. . . . .
show ip ospf database external
```

```

R2> show ip ospf database external

      OSPF Router with ID (2.2.2.2)

                AS External Link States

LS age: 252
Options: 0x2 : *I-I-I-I-I-EI*
LS Flags: 0xb
LS Type: AS-external-LSA
Link State ID: 192.168.56.0 (External Network Number)
Advertising Router: 2.2.2.2
LS Seq Number: 80000006
Checksum: 0x49c6
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 20
    Forward Address: 0.0.0.0
    External Route Tag: 0

```

4. Thiết lập Stub Area và check External-LSA được lan truyền vào bên trong (không thấy có). Thay vào đó là Summary-LSA với Link State ID = 0.0.0.0 được kích hoạt từ ABR của Stub Area để tạo RTE dạng default gateway

```

R4> nano /etc/quagga/ospfd.conf
router ospf
    ospf router-id 4.4.4.4
    redistribute connected
    network 192.168.3.128/25 area 3
area 3 stub
...

```

```

R4> service restart ospfd
R4> telnet 127.0.0.1 2604
....

```

```

R4> show ip ospf database external
R4> show ip ospf database external

```

```

      OSPF Router with ID (4.4.4.4)

                AS External Link States

```

```

R4> show ip ospf database summary

```

```

Summary Link States (Area 0.0.0.3 [Stub])

LS age: 420
Options: 0x0 : *I-I-I-I-I-I-I*
LS Flags: 0x6
LS Type: summary-LSA
Link State ID: 0.0.0.0 (summary Network Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000001
Checksum: 0x3919
Length: 28
Network Mask: /0
TOS: 0 Metric: 1

```

R4> show ip ospf route

```

R4> show ip ospf route
===== OSPF network routing table =====
N IA 0.0.0.0/0 [11] area: 0.0.0.3
via 192.168.3.1, eth7
N IA 192.168.0.0/24 [40] area: 0.0.0.3
via 192.168.3.1, eth7
N IA 192.168.1.0/24 [30] area: 0.0.0.3

```