Fall 2023 CS120
Routing Protocol Simulation

Yihong Hang
hangyh2023@shanghaitech.edu.cn

Outline

1. Simulator: IMUNES

2. RIP (Routing Information Protocol)

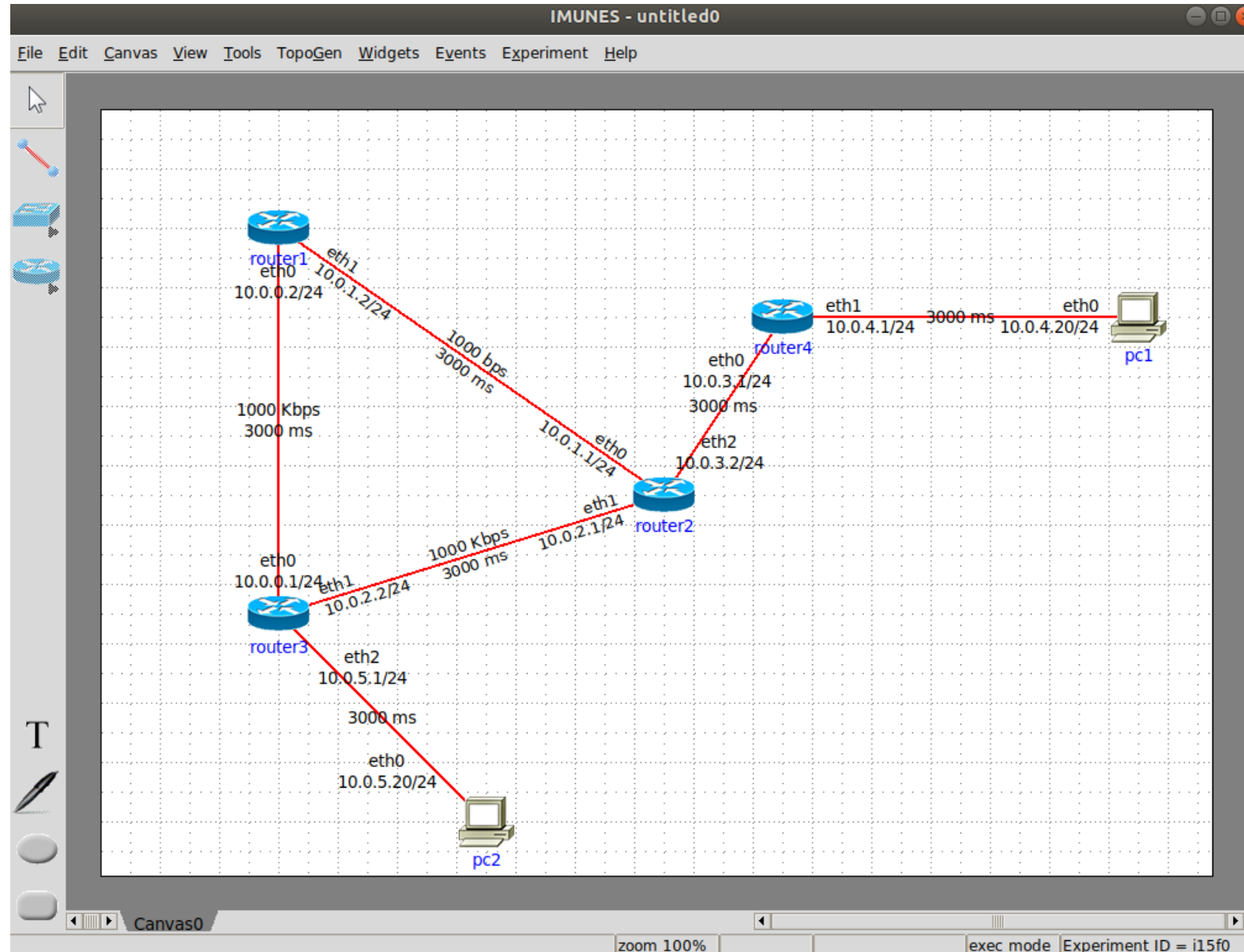3. OSPF (Open Shortest Path First)

1. Simulator: IMUNES*

*IMUNES: https://github.com/imunes/imunes, http://imunes.net/dl/guide/

1. Simulator: IMUNES

1.1. Introduction

Advantages:

1. Open-source network simulator

2. User guide manual and imunes-examples repositories

3. GUI

4. Inspect network traffic

1.2. Installation

1.2.1. install software dependencies:

Ubuntu 18.04 LTS (Mint 19, 19.1)

```
# apt install openvswitch-switch docker.io xterm wireshark \
    make imagemagick tk tcllib util-linux
```

1.2.2. install IMUNES
    $ sudo git clone https://github.com/imunes/imunes.git
    $ cd imunes
    $ sudo make install
    $ sudo imunes –p   //create a template filesystem for the topologies to work

1.2.3. Run IMUNES GUI
    $ sudo imunes
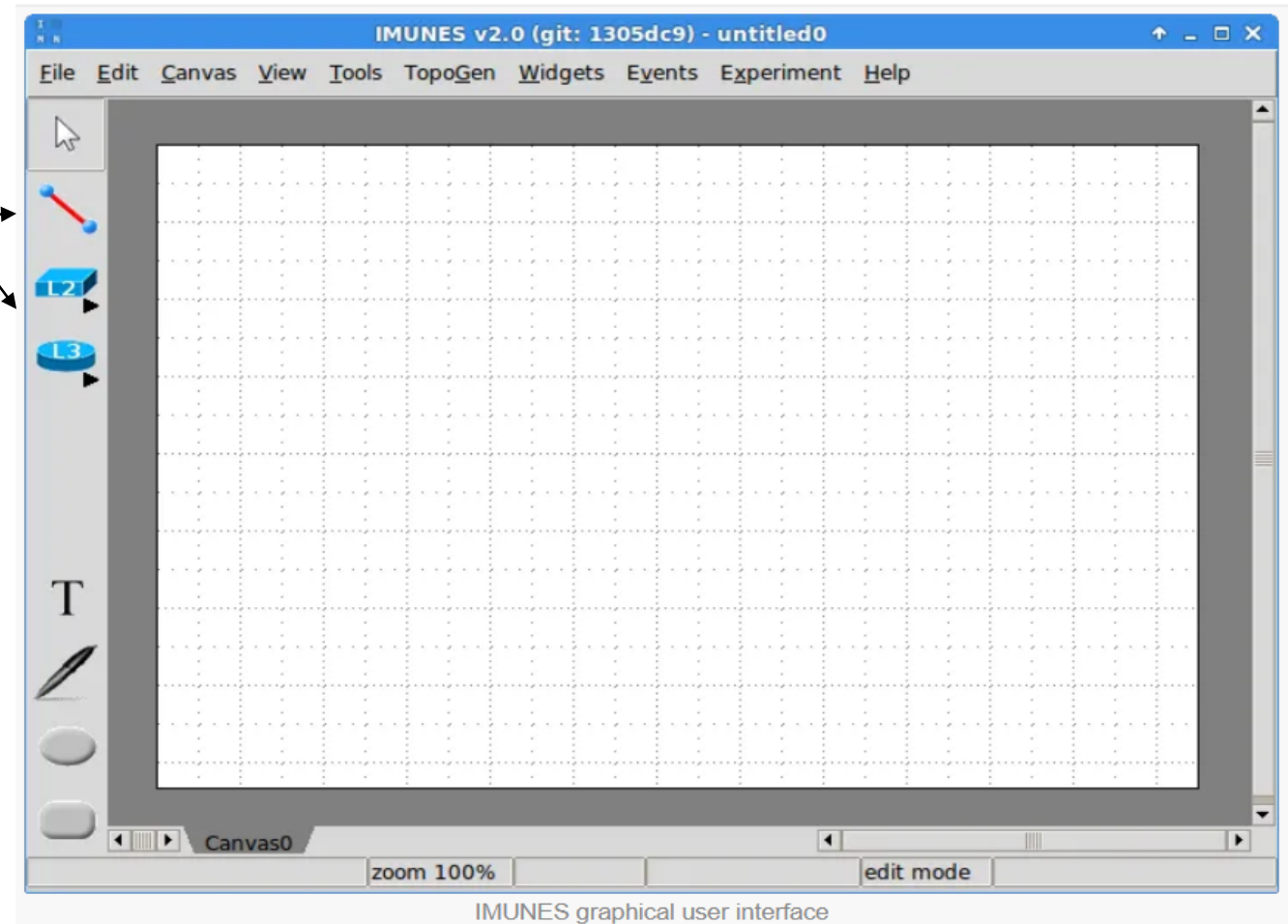
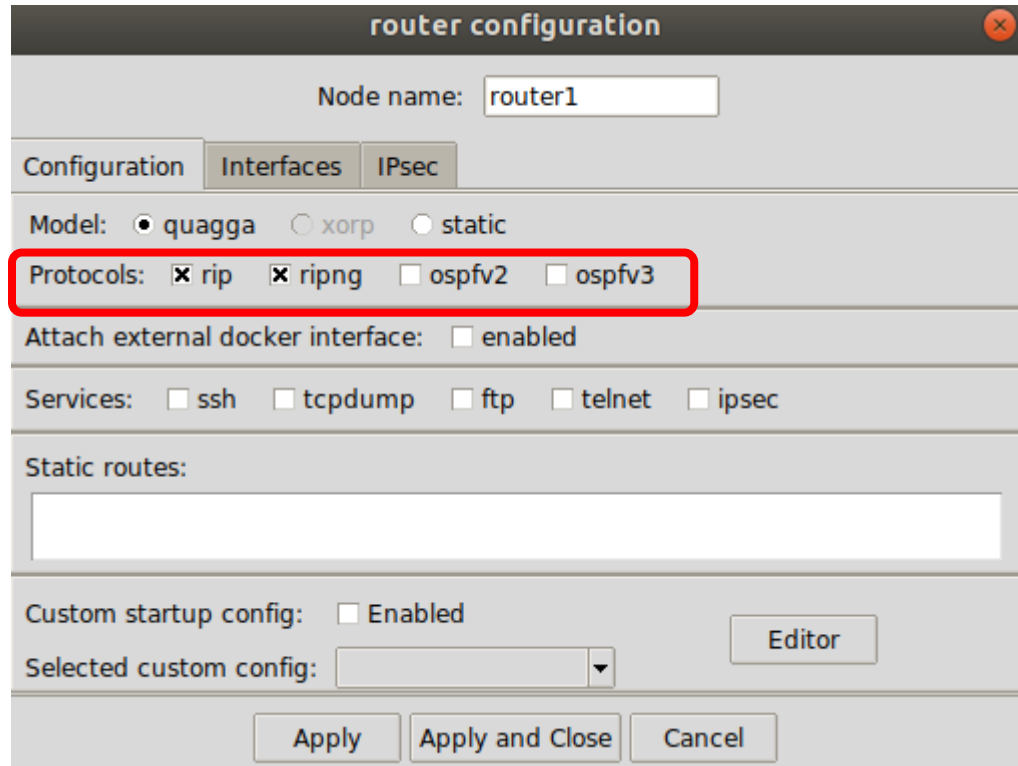# 1.3. Basic Operation

1.3.1. Building a simple network
Add Nodes (router, PC, switch…)
Connect Nodes
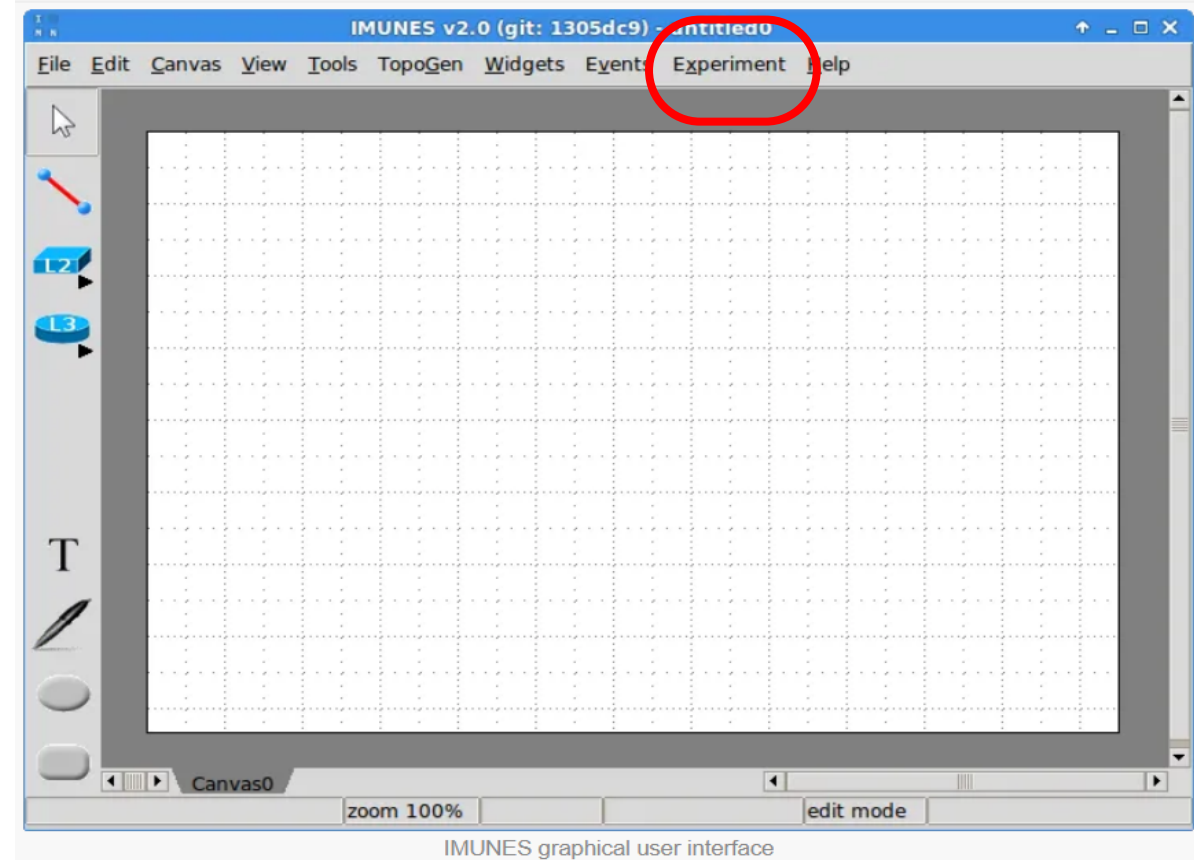

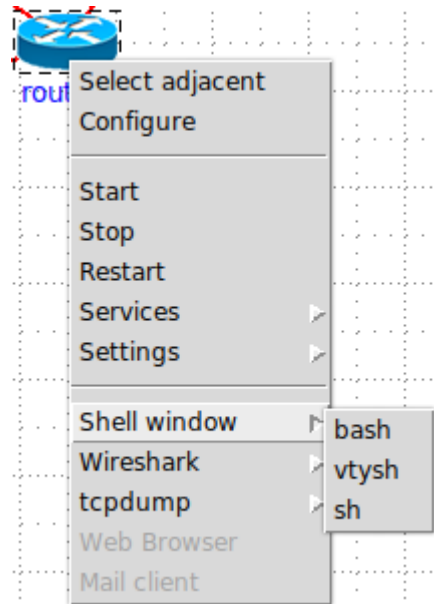IMUNES graphical user interface

1.3. Basic Operation

1.3.2. Configuring a simple network:
Double click on the network element

1.3.3. Start simulation
Experiment ->Execute

1.4. Shell Windows

```
Select adjacent
Configure

Start
Stop
Restart
Services        ▸
Settings        ▸

Shell window    ▸   bash
Wireshark       ▸   vtysh
tcpdump         ▸   sh
Web Browser
Mail client
```

bash: Unix shell. An implementation of sh.

Configure and examine the system.

vtysh: Shell for Quagga routing engine.

Configure and examine the router.

sh: Usually, /bin/sh is used to point to /bin/dash on most GNU/Linux systems. The standard command interpreter for the system.

```
root@router1:/# ls -al /bin/sh
lrwxrwxrwx 1 root root 4 Jan 24  2017 /bin/sh -> dash
root@router1:/# █
```

1.4. Shell Windows: bash

1. **ifconfig**: check or set network interface configurations
2. **ping**: check connectivity and show the delay
3. **traceroute**: trace the routing path and show the delay
4. **netstat**: examine network information (interface/routing table)
* --help will give you more information

```
root@router1:/# ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.5.1  netmask 255.255.255.0  broadcast 10.0.5.255
        inet6 fe80::4000:aaff:fe00:e  prefixlen 64  scopeid 0x20<link>
        inet6 fc00:1f::1  prefixlen 64  scopeid 0x0<global>
        ether 42:00:aa:00:00:0e  txqueuelen 1000  (Ethernet)
        RX packets 81  bytes 7657 (7.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 358  bytes 30984 (30.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
root@pc9:/# ping 10.0.6.1
PING 10.0.6.1 (10.0.6.1) 56(84) bytes of data.
64 bytes from 10.0.6.1: icmp_seq=1 ttl=64 time=0.405 ms
64 bytes from 10.0.6.1: icmp_seq=2 ttl=64 time=0.063 ms
64 bytes from 10.0.6.1: icmp_seq=3 ttl=64 time=0.065 ms
64 bytes from 10.0.6.1: icmp_seq=4 ttl=64 time=0.063 ms
^C
--- 10.0.6.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3053ms
rtt min/avg/max/mdev = 0.063/0.149/0.405/0.147 ms
```

```
root@pc2:/# traceroute 10.0.4.20
traceroute to 10.0.4.20 (10.0.4.20), 30 hops max, 60 byte packets
 1  10.0.5.1 (10.0.5.1)  0.066 ms  0.013 ms  0.006 ms
 2  10.0.2.1 (10.0.2.1)  12.865 ms  13.605 ms  14.045 ms
 3  10.0.3.1 (10.0.3.1)  14.861 ms  16.387 ms  16.747 ms
 4  10.0.4.20 (10.0.4.20)  17.285 ms  18.177 ms  19.110 ms
```

```
root@pc2:/# netstat --interface
Kernel Interface table
Iface      MTU      RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0       1500       612      0      0 0           47      0      0      0 BMRU
lo        16384        36      0      0 0           36      0      0      0 LRU
root@pc2:/# netstat --route
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         10.0.5.1        0.0.0.0         UG        0 0          0 eth0
10.0.5.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
```

1.4. Shell Windows: vtysh*

· show ip route [rip | ospf | ···]

```
router2# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O>* 10.0.0.0/24 [110/20] via 10.0.1.2, eth0, 00:02:38
 *                        via 10.0.2.2, eth1, 00:02:38
O   10.0.1.0/24 [110/10] is directly connected, eth0, 6d00h39m
C>* 10.0.1.0/24 is directly connected, eth0
O   10.0.2.0/24 [110/10] is directly connected, eth1, 6d00h07m
C>* 10.0.2.0/24 is directly connected, eth1
O   10.0.3.0/24 [110/10] is directly connected, eth2, 6d00h39m
C>* 10.0.3.0/24 is directly connected, eth2
O>* 10.0.4.0/24 [110/20] via 10.0.3.1, eth2, 6d00h27m
O>* 10.0.5.0/24 [110/20] via 10.0.2.2, eth1, 00:02:38
C>* 127.0.0.0/8 is directly connected, lo
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 01w0d01h
```

```
router2# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O>* 10.0.0.0/24 [110/20] via 10.0.1.2, eth0, 00:03:05
 *                        via 10.0.2.2, eth1, 00:03:05
O   10.0.1.0/24 [110/10] is directly connected, eth0, 6d00h39m
O   10.0.2.0/24 [110/10] is directly connected, eth1, 6d00h07m
O   10.0.3.0/24 [110/10] is directly connected, eth2, 6d00h40m
O>* 10.0.4.0/24 [110/20] via 10.0.3.1, eth2, 6d00h27m
O>* 10.0.5.0/24 [110/20] via 10.0.2.2, eth1, 00:03:05
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 01w0d01h
```

1.4. Shell Windows: vtysh

· show ip ospf neighbor (see 3.1)
· show ip ospf database (see 3.1)

· show ip ospf interface

```
router2# show ip ospf interface eth0
eth0 is up
  ifindex 80, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.1.1/24, Broadcast 10.0.1.255, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 10.0.3.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.0.3.2, Interface Address 10.0.1.1
  Backup Designated Router (ID) 10.0.1.2, Interface Address 10.0.1.2
  Saved Network-LSA sequence number 0x80000006
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 7.208s
  Neighbor Count is 1, Adjacent neighbor count is 1
```

1.4. Shell Windows: vtysh

Configure the cost of OSPF (also see 3.3)

· config terminal

· interface INTFACE_NAME

· bandwidth BW / ospf cost COST

```
router2# configure terminal
router2(config)# interface eth0
router2(config-if)# ospf cost 66
router2(config-if)# bandwidth 666
router2(config-if)# exit
router2(config)# exit
router2# show ip ospf interface
eth0 is up
  ifindex 80, MTU 1500 bytes, BW 666 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.1.1/24, Broadcast 10.0.1.255, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 10.0.3.2, Network Type BROADCAST, Cost: 66
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.0.3.2, Interface Address 10.0.1.1
  Backup Designated Router (ID) 10.0.1.2, Interface Address 10.0.1.2
  Saved Network-LSA sequence number 0x80000006
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 9.812s
  Neighbor Count is 1, Adjacent neighbor count is 1
```
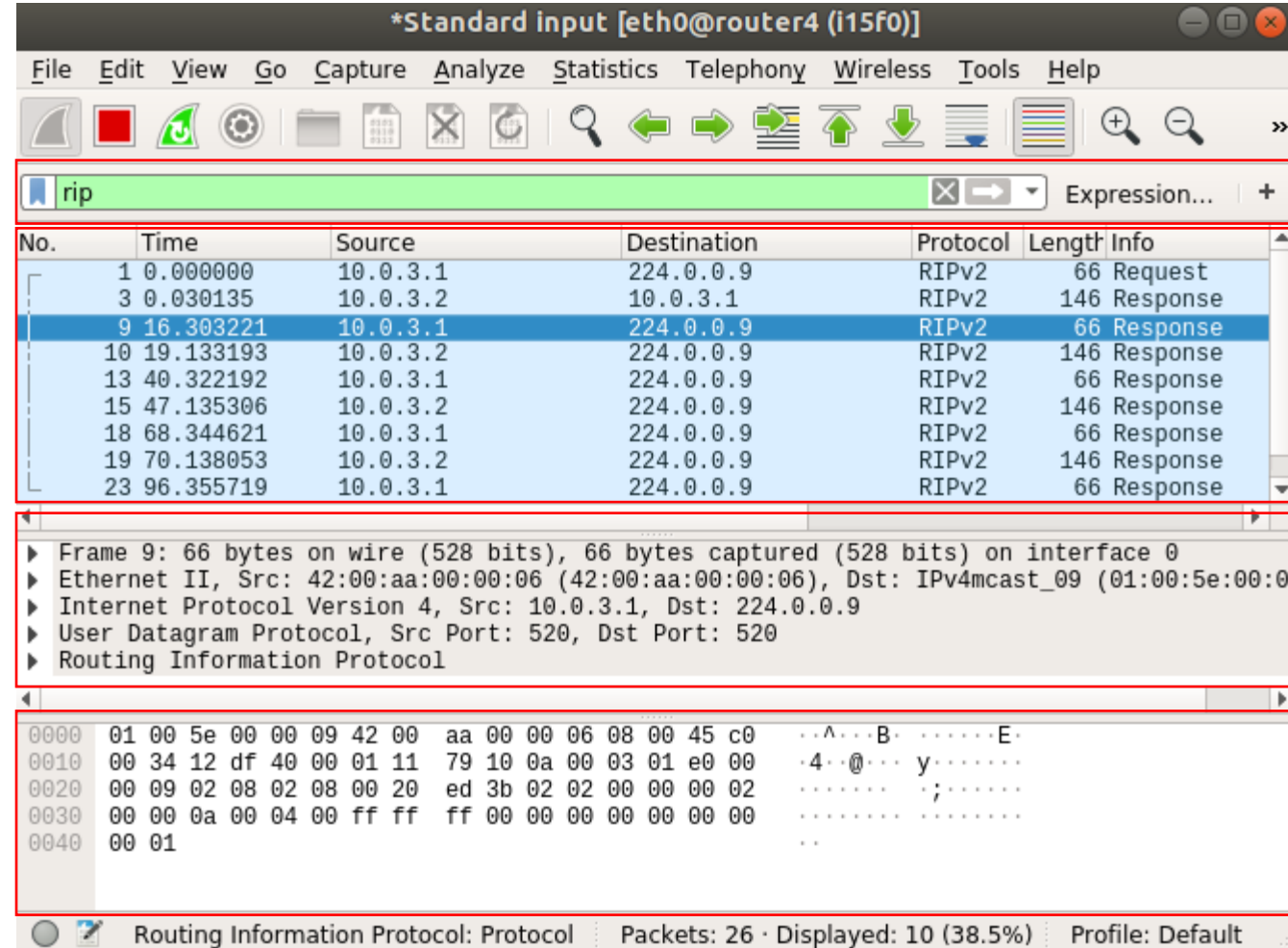
1.5. Wireshark[*]
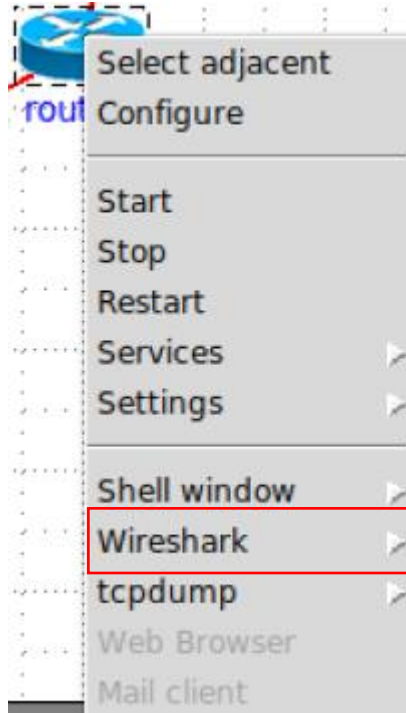
Filter: Protocol type / (Src/Dst) IP address / port …

List of captured packet

Packet details

Packet in Hex

*Wireshark (user guide): https://www.wireshark.org/docs/wsug_html_chunked/
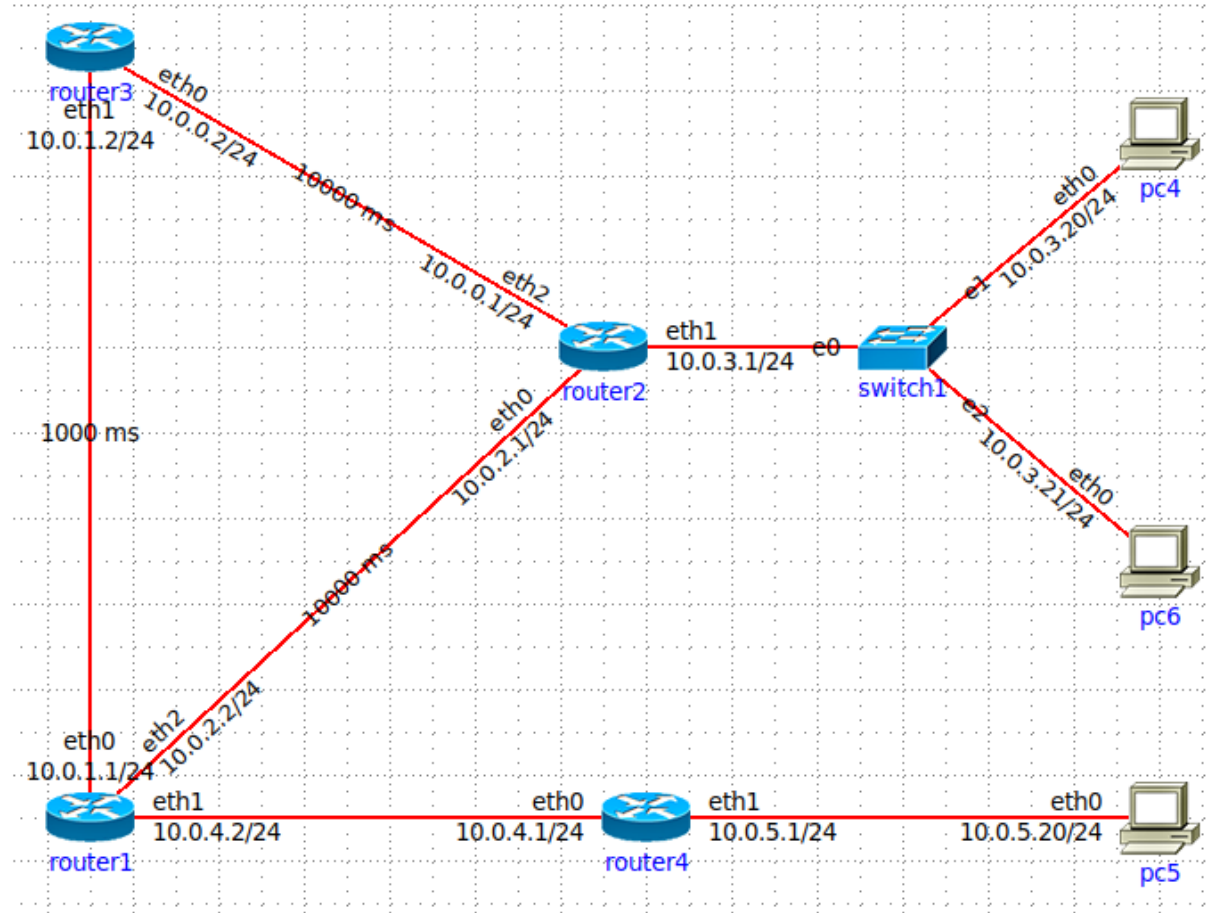
2. RIP* (Routing Information Protocol)

2.1. Stable Network

2.2. Encounter Network Node Failure

2.3. Count to Infinity Problem

2.4. Silent Failure

2. RIP (Routing Information Protocol)

*The RIP topology file is rip.imn, open it in the IMUNES.

2.1. Stable Network

RIP Routing Table

① Destination Network

② Next Hop

③ Convergent Costs

④ Time-to-live

RIP advertisement - periodic update

*Multicast address: https://www.iana.org/assignments/multicast-addresses/multicast-addresses.xhtml

2.1. Stable Network

RIP Routing Table

RIP advertisement - periodic update

Why does not Router4 advertise its whole RIP routing table?

2.1. Stable Network

Split Horizon*

A node **CANNOT** send those routes it learned from each neighbor back to that neighbor.

E.g., A with (C, 2, B) will not send route (C, 2) in routing update to B

To prevent **routing loops** that involve **only two gateways.**

> **Without Split Horizon:**
> Count to infinity after node C is down
> → Node B sets (C,inf)
> → Node B receives (C,2) from A
> → Node B sets (C, 3, A) ...
>
> A ⟷ B

sending info about C router to A router, from B router

sending info about C router to B router, from A router
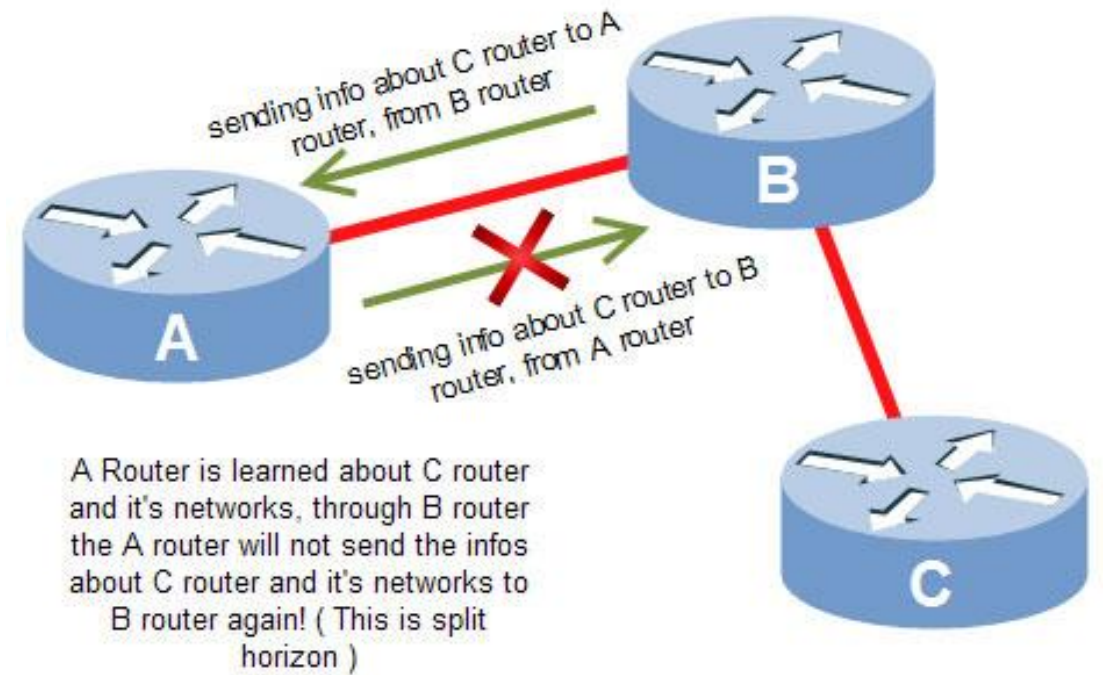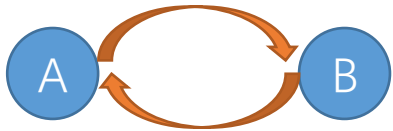
A Router is learned about C router and it's networks, through B router the A router will not send the infos about C router and it's networks to B router again! (This is split horizon)

2.2. Normal Case
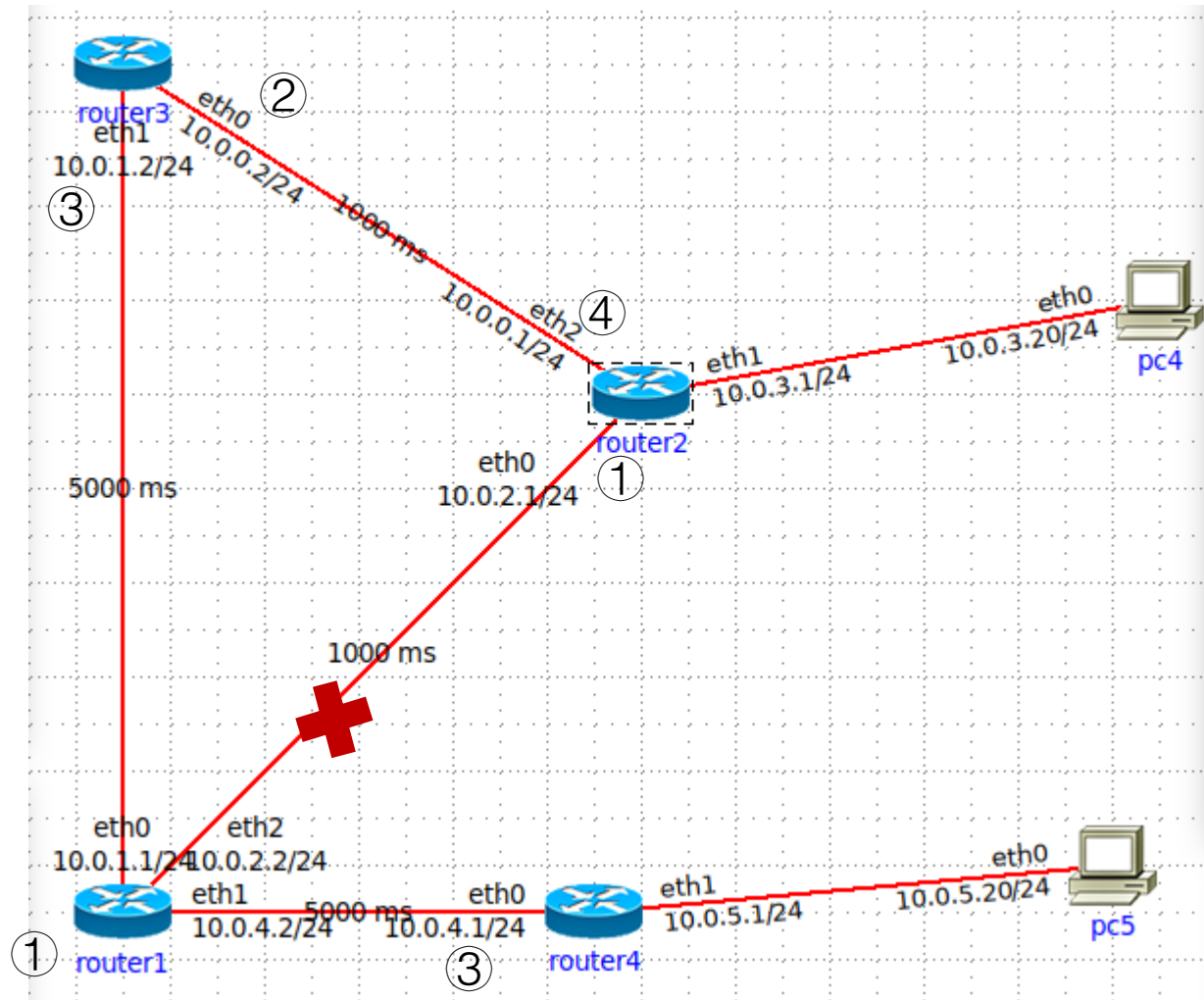
Triggered Updates

① **T=0**: **Router1&2** detect link failure
→ update their routing tables
→ send RIP advertisement to their neighbors

② **T=1s**: **Router3** eth0 receives RIP response
from router2 eth2
→ update routing table

③ **T=5s**: **Router3** eth1 & **Router4** eth0 receive
RIP response from router1 eth0 & router1 eth1
→ update routing table

④ **T=6s**: **Router2** eth2 receive RIP response
from router3 eth0
→ update routing table

....

2.3 Count to Infinity Case
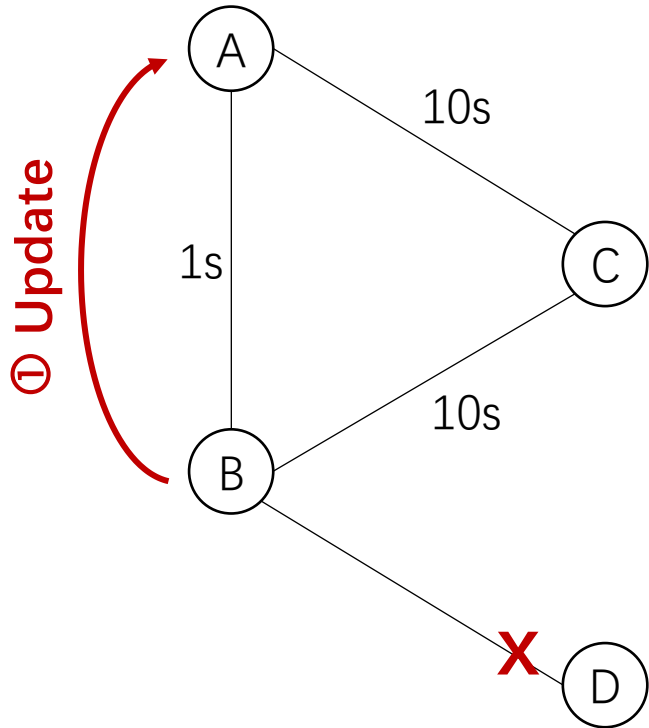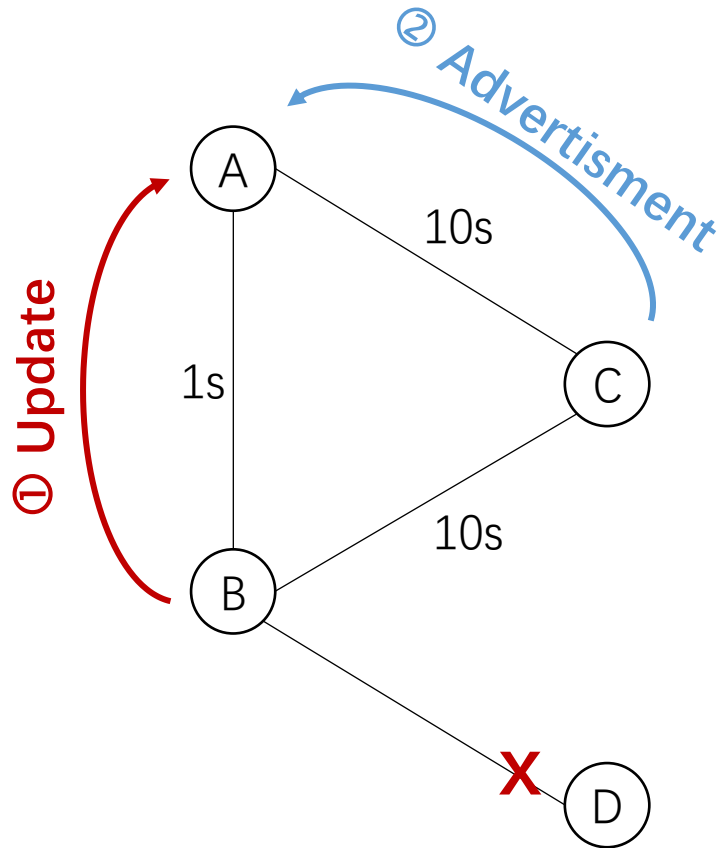
T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2

2.3 Count to Infinity Case

T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2
n=3	16	16	2

2.3 Count to Infinity Case

T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2
n=3	16	16	2
10	16	16	16
n+10	3	16	16
n+11	3	4	16
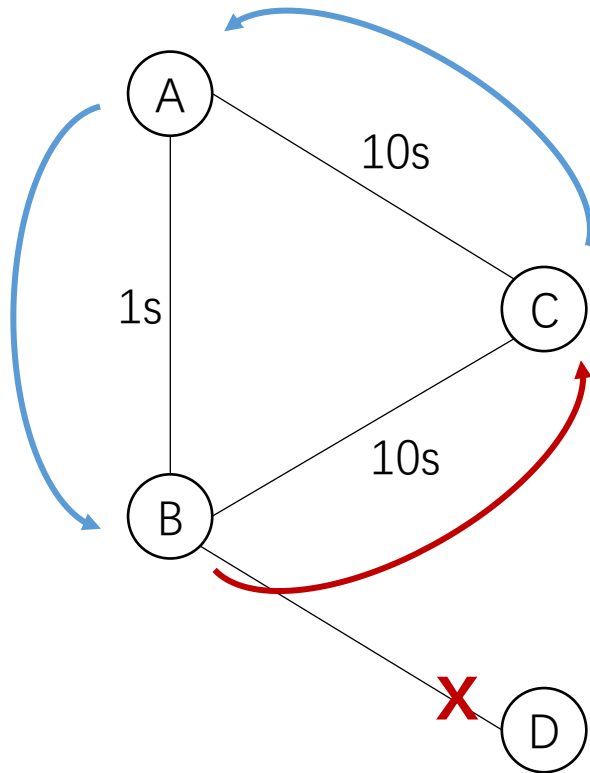
2.3 Count to Infinity Case
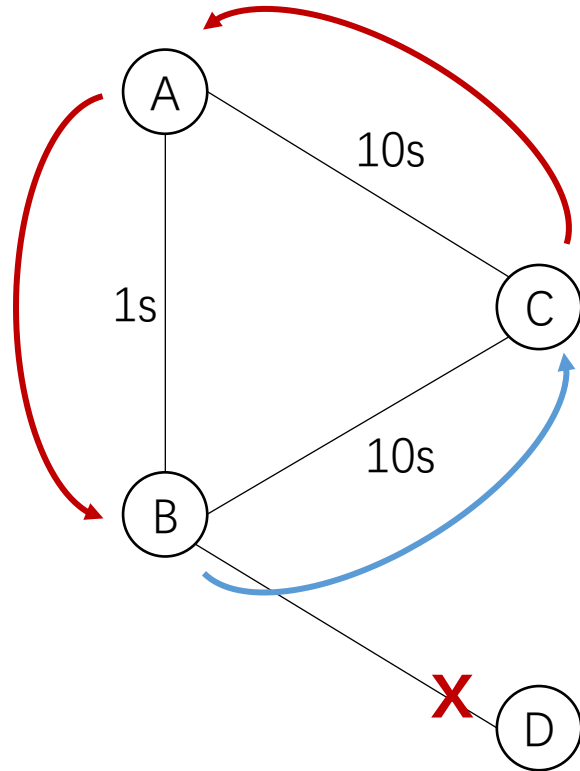
T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2
n=3	16	16	2
10	16	16	16
n+10	3	16	16
n+11	3	4	16
20	16	4	16
21	16	16	16
n+11 +10	16	16	5

2.3 Count to Infinity Case
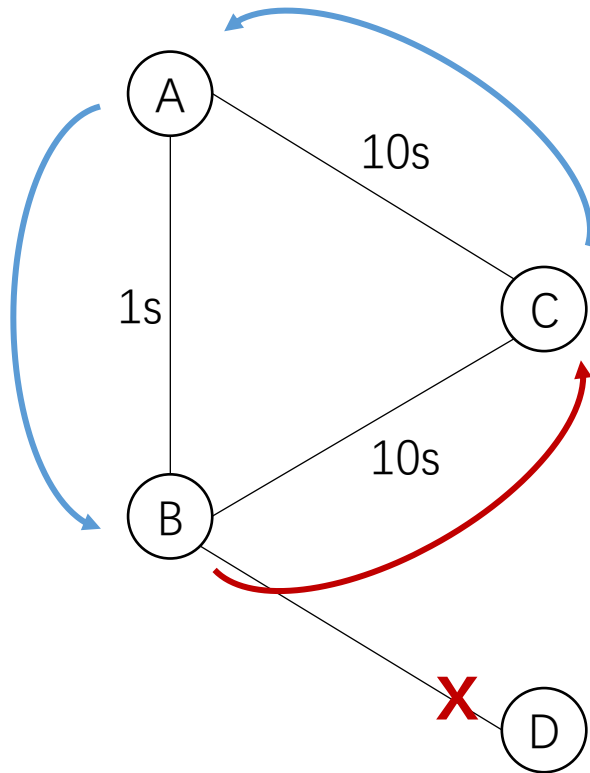
T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2
n=3	16	16	2
10	16	16	16
n+10	3	16	16
n+11	3	4	16
20	16	4	16
21	16	16	16
n+11 +10	16	16	5
31	16	16	16
n+31	6	16	16
n+32	6	7	16

2.3 Count to Infinity Case
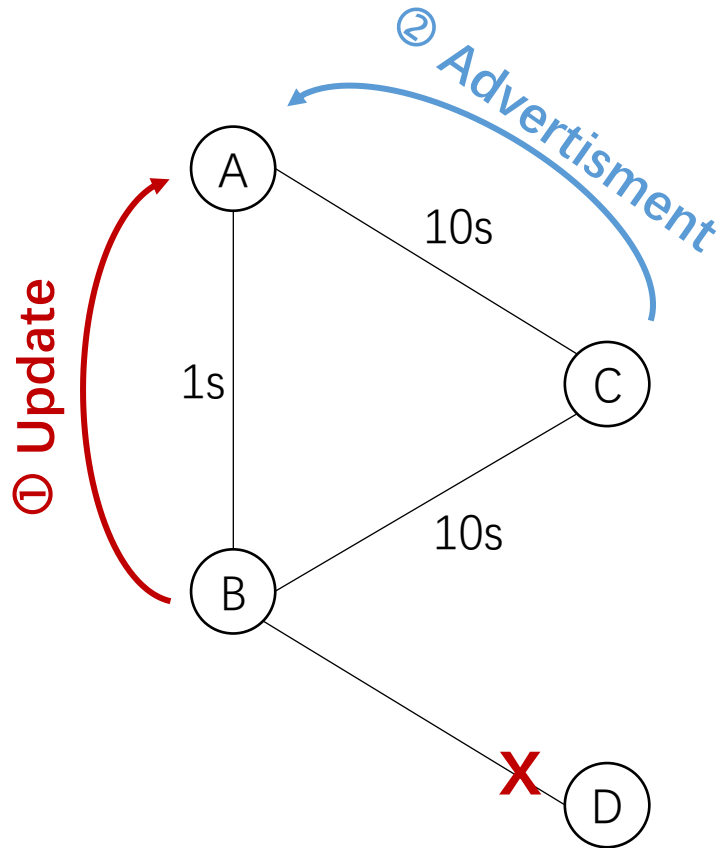
T	A	B	C
<0	2	1	2
0	2	16	2
1	16	16	2
n=3	16	16	2
10	16	16	16
n+10	3	16	16
n+11	3	4	16
20	16	4	16
21	16	16	16
n+11 +10	16	16	5
31	16	16	16
n+31	6	16	16
n+32	6	7	16

,where –9 < n < 10

Simulation

2.4. Silent Failure Case

Kill RIP process of Router4

2.4. Silent Failure Case

Kill RIP process of Router4

```
Every 1.0s: vtysh -c "show ip rip"          router1: Thu Oct 29 11:12:54 2020

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) 10.0.0.0/24      10.0.2.1             2 10.0.2.1        0 02:45
C(i) 10.0.1.0/24      0.0.0.0              1 self            0
C(i) 10.0.2.0/24      0.0.0.0              1 self            0
R(n) 10.0.3.0/24      10.0.2.1             2 10.0.2.1        0 02:45
C(i) 10.0.4.0/24      0.0.0.0              1 self            0
R(n) 10.0.5.0/24      10.0.4.1             2 10.0.4.1        0 02:53
```
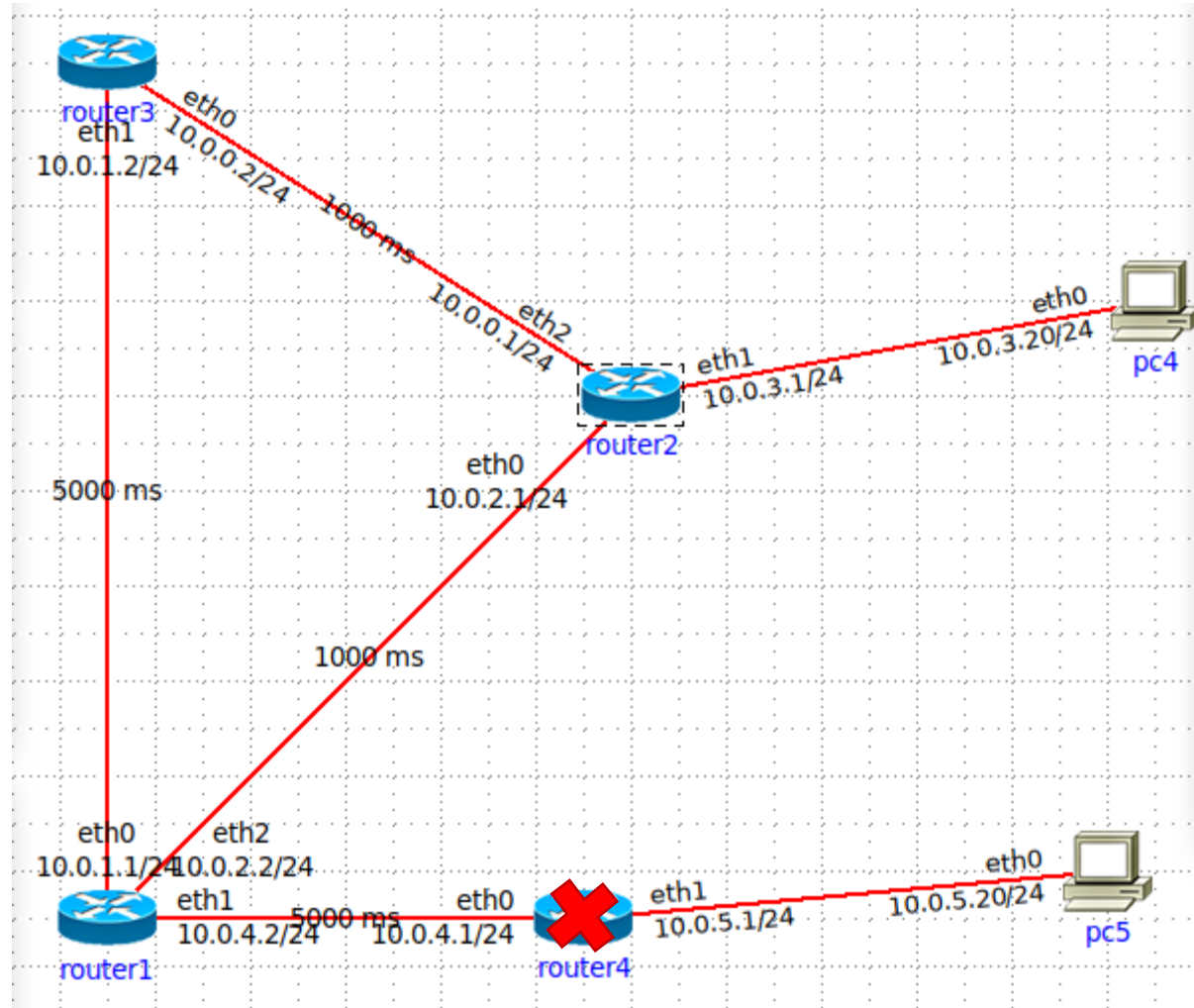① (highlighted: R(n) 10.0.5.0/24 row)

```
Every 1.0s: vtysh -c "show ip rip"          router1: Thu Oct 29 11:15:56 2020

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) 10.0.0.0/24      10.0.2.1             2 10.0.2.1        0 02:47
C(i) 10.0.1.0/24      0.0.0.0              1 self            0
C(i) 10.0.2.0/24      0.0.0.0              1 self            0
R(n) 10.0.3.0/24      10.0.2.1             2 10.0.2.1        0 02:47
C(i) 10.0.4.0/24      0.0.0.0              1 self            0
R(n) 10.0.5.0/24      10.0.4.1            16 10.0.4.1        0 01:54
```
③ (highlighted: R(n) 10.0.5.0/24 row)

```
Every 1.0s: vtysh -c "show ip rip"          router1: Thu Oct 29 11:15:44 2020

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) 10.0.0.0/24      10.0.2.1             2 10.0.2.1        0 02:53
C(i) 10.0.1.0/24      0.0.0.0              1 self            0
C(i) 10.0.2.0/24      0.0.0.0              1 self            0
R(n) 10.0.3.0/24      10.0.2.1             2 10.0.2.1        0 02:53
C(i) 10.0.4.0/24      0.0.0.0              1 self            0
R(n) 10.0.5.0/24      10.0.4.1             2 10.0.4.1        0 00:05
```
② (highlighted: R(n) 10.0.5.0/24 row)

Detect failure: 180s
+
Delete infinity entry: 120s

3. OSPF* (Open Shortest Path First)

3.1. Stable Network

3.2. Add New Node

3.3. Cost in OSPF

3.4. Silent Failure

3.1. Stable Network

router4
eth1
10.0.4.1/24
eth0
10.0.4.20/24
pc1

eth0
10.0.3.1/24

eth2
10.0.3.2/24

eth1
10.0.2.1/24
router2

1000 Kbps
5.000 ms

eth1
10.0.2.2/24

router3

eth2
10.0.5.1/24

eth0
10.0.5.20/24

pc2
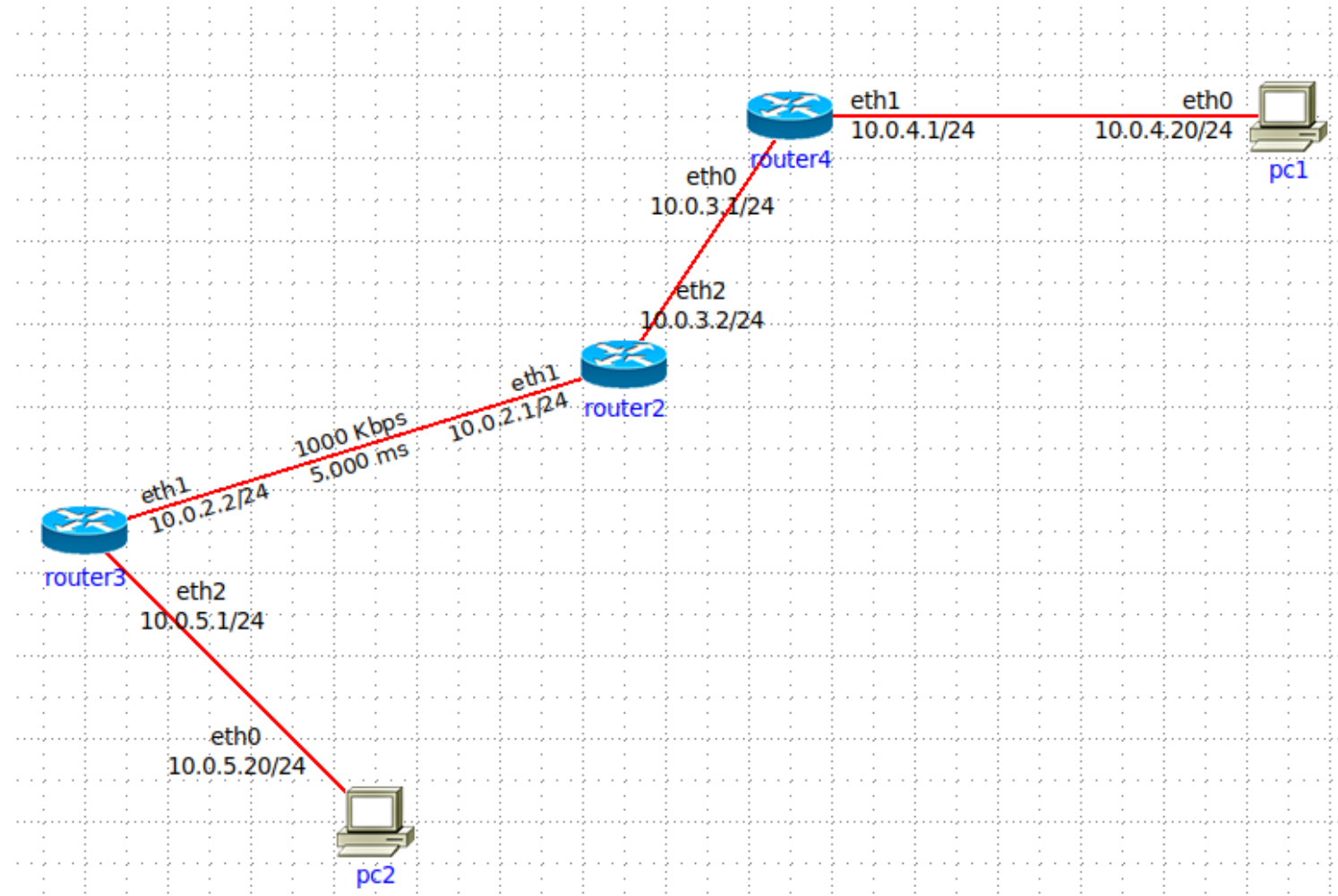
*The OSPF topology file is ospf.imn, open it in the IMUNES.

3.1. Stable Network

OSPF Routing Table

```
router1# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route
        ①                ②              ③                  ④
O    10.0.0.0/24  [110/10] is directly connected, eth0, 00:02:45
O    10.0.1.0/24  [110/10] is directly connected, eth1, 00:02:45
O>* 10.0.3.0/24  [110/20] via 10.0.1.1, eth1, 00:01:57
O>* 10.0.4.0/24  [110/30] via 10.0.1.1, eth1, 00:01:56
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 00:02:45
router1#
router1# show ip ospf neighbor
```

① Destination Network

② Administrative Distance* / Cost

③ Next Hop

④ Existence Time

OSPF Neighbour Table

```
Neighbor ID     Pri State         Dead Time Address        Interface
   RXmtL RqstL DBsmL
10.0.3.2          1 Full/DR        30.135s 10.0.1.1         eth1:10.0.1.2
       0      0       0
router1#
router1# show ip ospf database
```

OSPF Link State Database

```
        OSPF Router with ID (10.0.1.2)

        Router Link States (Area 0.0.0.0)

Link ID        ADV Router        Age  Seq#        CkSum  Link count
10.0.1.2       10.0.1.2          133 0x80000007 0x76e3 3
10.0.3.2       10.0.3.2          136 0x80000008 0x3a0a 3
10.0.4.1       10.0.4.1          139 0x80000006 0x1c34 3

        Net Link States (Area 0.0.0.0)

Link ID        ADV Router        Age  Seq#        CkSum
10.0.1.1       10.0.3.2          136 0x80000001 0x7ba7
10.0.3.1       10.0.4.1          138 0x80000001 0x7da1
```
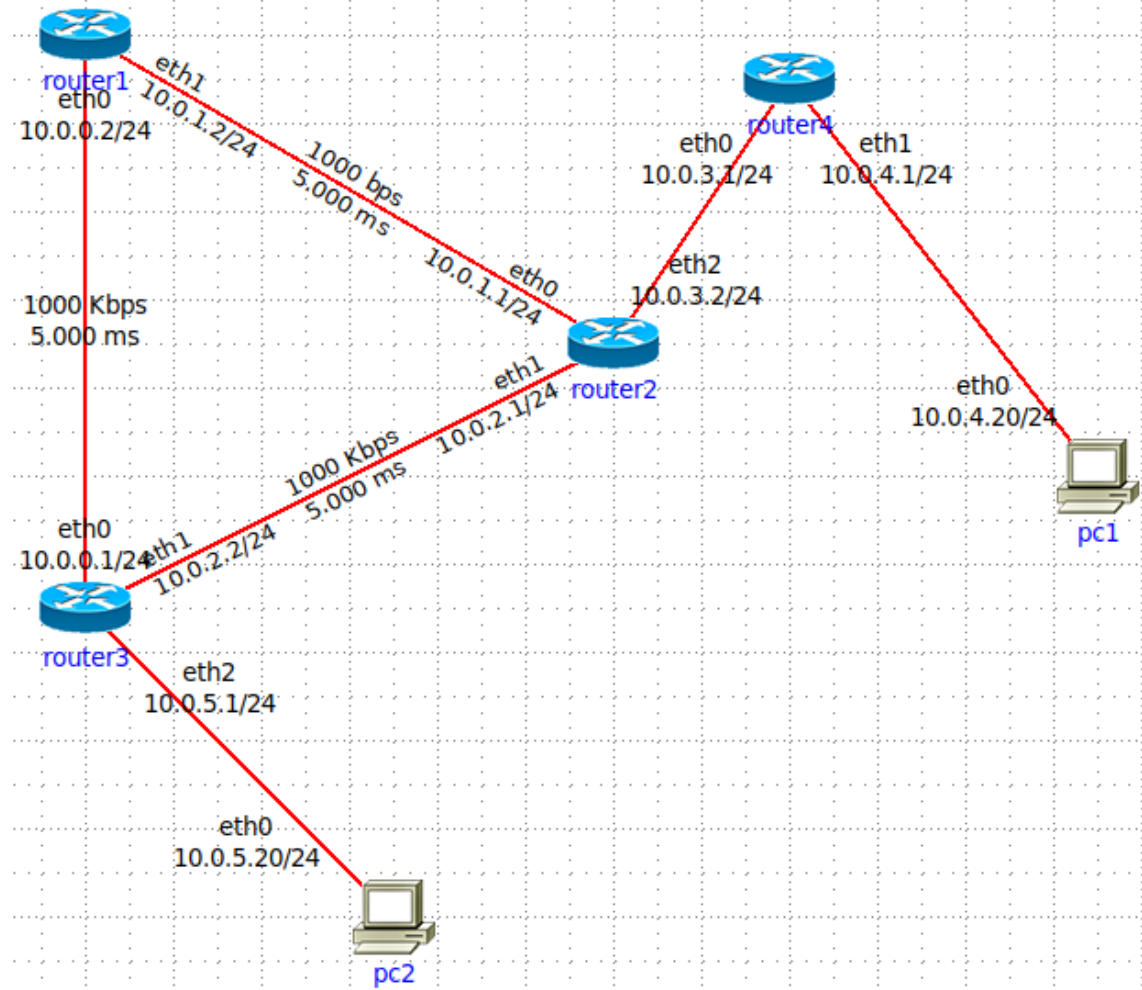
*Administrative Distance: https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/15986-admin-distance.html

3.2. Add New Node

*The OSPF topology file is ospf.imn, open it in the IMUNES.

3.2. Add New Node

```
router1# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O   10.0.0.0/24 [110/10] is directly connected, eth0, 00:03:18
O   10.0.1.0/24 [110/10] is directly connected, eth1, 00:02:31
O>* 10.0.2.0/24 [110/20] via 10.0.0.1, eth0, 00:02:27
  *             via 10.0.1.1, eth1, 00:02:27
O>* 10.0.3.0/24 [110/20] via 10.0.1.1, eth1, 00:02:31
O>* 10.0.4.0/24 [110/30] via 10.0.1.1, eth1, 00:02:26
O>* 10.0.5.0/24 [110/20] via 10.0.0.1, eth0, 00:02:27
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 00:03:20
router1#
router1#
router1# show ip ospf neighbor

Neighbor ID     Pri State        Dead Time Address      Interface
10.0.5.1          1 Full/DR         36.902s 10.0.0.1    eth0:10.0.0.2
10.0.3.2          1 Full/DR         36.092s 10.0.1.1    eth1:10.0.1.2
router1#
router1#
router1# show ip ospf database

         OSPF Router with ID (10.0.1.2)

              Router Link States (Area 0.0.0.0)

Link ID         ADV Router       Age  Seq#       CkSum  Link count
10.0.1.2        10.0.1.2         184 0x80000009 0xf754 3
10.0.3.2        10.0.3.2         180 0x8000000c 0x14f7 4
10.0.4.1        10.0.4.1         186 0x80000006 0x1c34 3
10.0.5.1        10.0.5.1         185 0x80000009 0x36e4 4
```
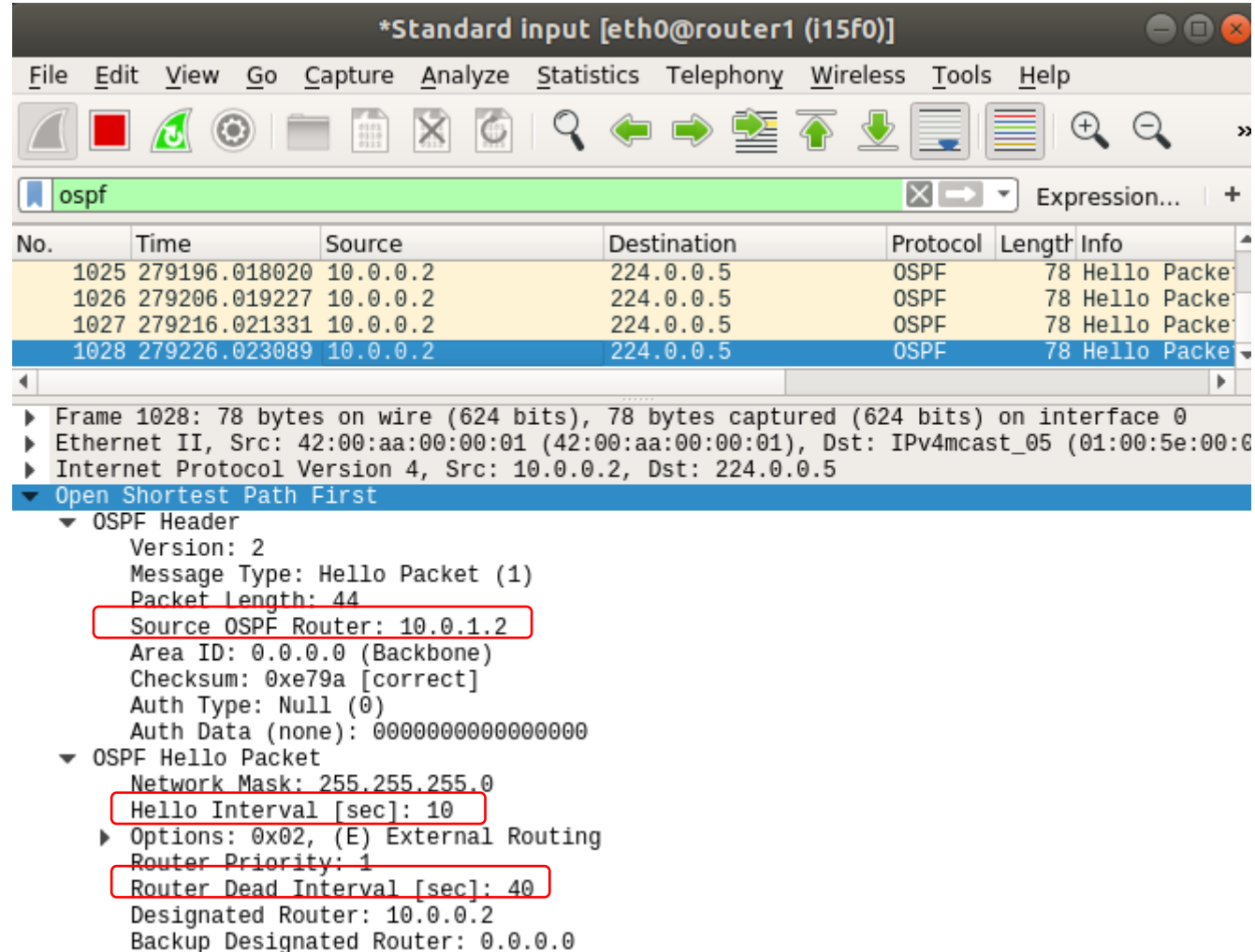
*Administrative Distance: https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/15986-admin-distance.html

3.2. Add New Node

OSPF Hello - periodic update

3.3. Cost in OSPF

Cost = 10^8 / (BW of interface in bps)

router1
eth1
eth0
10.0.1.2/24
10.0.0.2/24
5.000 ms

1000 Kbps
5.000 ms

router4
eth0
10.0.3.1/24

10.0.1.1/24
eth0

eth2
10.0.3.2/24

?
800.00 Kbps
500 ms
10.0.2.1/24
eth1
router2

No!

eth0
10.0.0.1/24
eth1
10.0.2.2/2

router3
eth2
10.0.5.1/24

eth0
10.0.5.20/24

pc2

```
eth1 is up
   ifindex 79, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
   Internet Address 10.0.1.2/24, Broadcast 10.0.1.255, Area 0.0.0.0
   MTU mismatch detection:enabled
   Router ID 10.0.1.2, Network Type BROADCAST, Cost: 10
   Transmit Delay is 1 sec, State Backup, Priority 1
   Designated Router (ID) 10.0.3.2, Interface Address 10.0.1.1
   Backup Designated Router (ID) 10.0.1.2, Interface Address 10.0.1.2
   Saved Network-LSA sequence number 0x80000005
   Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
   Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
      Hello due in 8.439s
   Neighbor Count is 1, Adjacent neighbor count is 1

eth1 is up
   ifindex 79, MTU 1500 bytes, BW 1000 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
   Internet Address 10.0.1.2/24, Broadcast 10.0.1.255, Area 0.0.0.0
   MTU mismatch detection:enabled
   Router ID 10.0.1.2, Network Type BROADCAST, Cost: 100
   Transmit Delay is 1 sec, State Backup, Priority 1

eth1 is up
   ifindex 79, MTU 1500 bytes, BW 1000000 Kbit <UP,BROADCAST,RUNNING,MULTICAST
   Internet Address 10.0.1.2/24, Broadcast 10.0.1.255, Area 0.0.0.0
   MTU mismatch detection:enabled
   Router ID 10.0.1.2, Network Type BROADCAST, Cost: 1
   Transmit Delay is 1 sec, State Backup, Priority 1
```
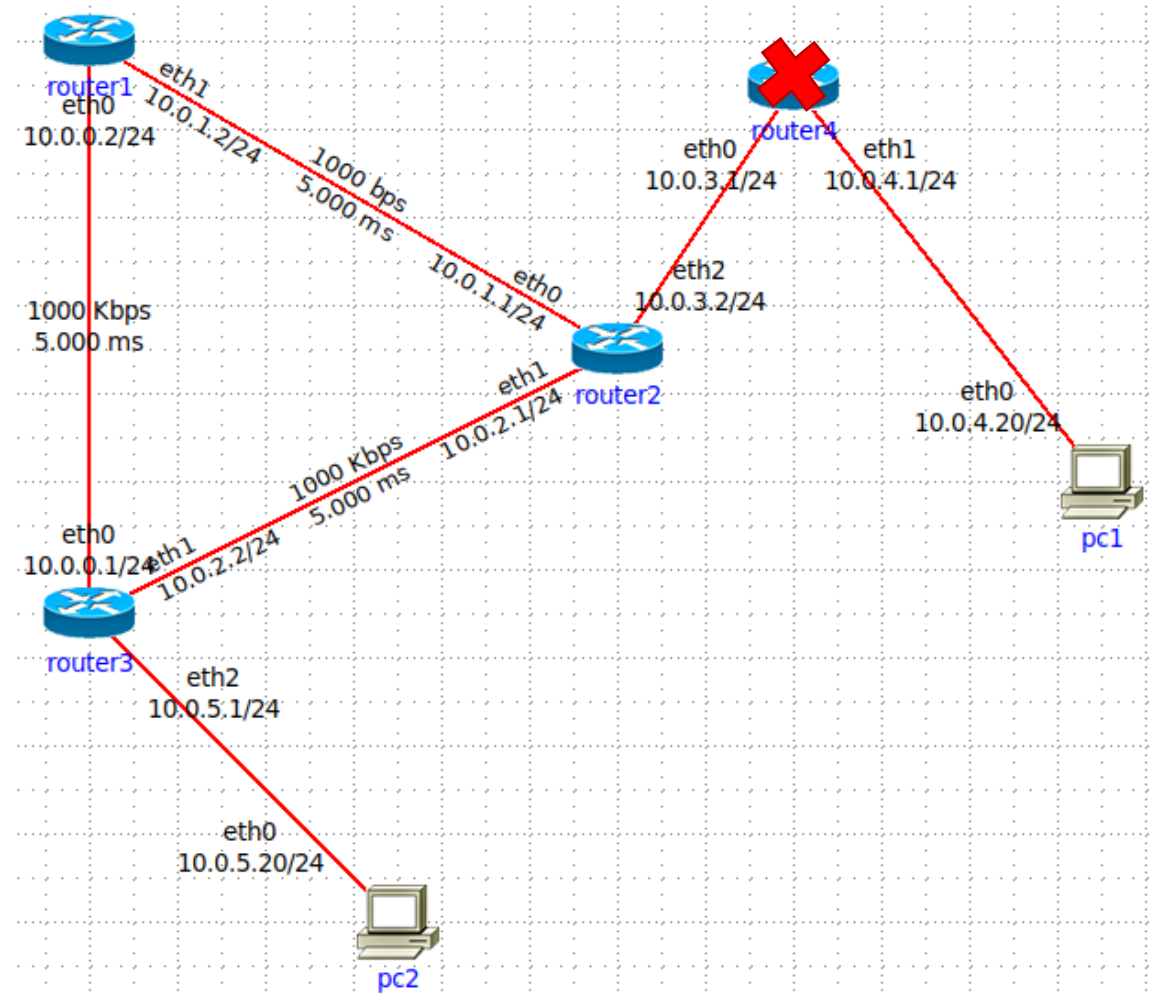
3.4. Silent Failure

Kill OSPF process of Router4

3.4. Silent Failure

Kill OSPF process of Router4

①
```
O>* 10.0.0.0/24 [110/20] via 10.0.1.2, eth0, 00:05:39
 *                      via 10.0.2.2, eth1, 00:05:39
O    10.0.1.0/24 [110/10] is directly connected, eth0, 00:06:23
O    10.0.2.0/24 [110/10] is directly connected, eth1, 00:07:30
O    10.0.3.0/24 [110/10] is directly connected, eth2, 00:07:30
O>* 10.0.4.0/24 [110/20] via 10.0.3.1, eth2, 00:02:20
O>* 10.0.5.0/24 [110/20] via 10.0.2.2, eth1, 00:05:39
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 00:07:30
Every 1.0s: vtysh -c "show ip ospf neighbor"          router2: Thu

Neighbor ID     Pri State         Dead Time Address        Interface
10.0.1.2          1 Full/DR          31.372s 10.0.1.2       eth0:10.0.1.1
10.0.5.1          1 Full/Backup      37.787s 10.0.2.2       eth1:10.0.2.1
10.0.4.1          1 Full/Backup      38.950s 10.0.3.1       eth2:10.0.3.2
```

②
```
O>* 10.0.0.0/24 [110/20] via 10.0.1.2, eth0, 00:06:15
 *                      via 10.0.2.2, eth1, 00:06:15
O    10.0.1.0/24 [110/10] is directly connected, eth0, 00:06:57
O    10.0.2.0/24 [110/10] is directly connected, eth1, 00:08:04
O    10.0.3.0/24 [110/10] is directly connected, eth2, 00:08:04
O>* 10.0.4.0/24 [110/20] via 10.0.3.1, eth2, 00:02:54
O>* 10.0.5.0/24 [110/20] via 10.0.2.2, eth1, 00:06:15
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 00:08:04
Every 1.0s: vtysh -c "show ip ospf neighbor"          router2: Thu

Neighbor ID     Pri State         Dead Time Address        Interface
10.0.1.2          1 Full/DR          33.580s 10.0.1.2       eth0:10.0.1.1
10.0.5.1          1 Full/Backup      34.792s 10.0.2.2       eth1:10.0.2.1
10.0.4.1          1 Full/Backup       5.954s 10.0.3.1       eth2:10.0.3.2
```

③
```
O>* 10.0.0.0/24 [110/20] via 10.0.1.2, eth0, 00:06:22
 *                      via 10.0.2.2, eth1, 00:06:22
O    10.0.1.0/24 [110/10] is directly connected, eth0, 00:07:03
O    10.0.2.0/24 [110/10] is directly connected, eth1, 00:08:15
O    10.0.3.0/24 [110/10] is directly connected, eth2, 00:08:15
O>* 10.0.5.0/24 [110/20] via 10.0.2.2, eth1, 00:06:22
O>* 127.0.0.1/32 [110/0] is directly connected, lo, 00:08:15
Every 1.0s: vtysh -c "show ip ospf neighbor"          router2: Thu

Neighbor ID     Pri State         Dead Time Address        Interface
10.0.1.2          1 Full/DR          39.292s 10.0.1.2       eth0:10.0.1.1
10.0.5.1          1 Full/Backup      37.579s 10.0.2.2       eth1:10.0.2.1
```

4 Hello packets (40s) to detect failure.
Delete the table entry immediately.

Simulation

Questions?

Source Code of Routing Algorithms

1. RIP

   https://github.com/Quagga/quagga/tree/master/ripd

2. OSPF

   https://github.com/Quagga/quagga/tree/master/ospfd