Lab 6 - final report team 10

מגישים:
זיו זידמן 208660365
גיא לייבה 208279620

Chapter 2

2.11
We start by configuring the router with OSPF protocol 1. Next, we assign it a unique Router ID that it will use to identify itself to its neighbors. Finally, we define the network that OSPF will route, using a WILDCARD mask. The wildcard is the "reverse" of the subnet mask, meaning that "0" represents the bits of the network that must match, and "1" represents the bits of hosts/subnetworks where no match is required. Additionally, we assign the network to an area (AREA).

2.12
It uses "encapsulation" of type OSPF

2.13
It starts with an Hello packet which job is to meet new neighbours and to keep alive.
If it is a new neighbour he will send him the DataBase Description with an explanation on the topology he knows.
If the neighbour needs to update part of its topology it will send a request.
The Router will answer the details with an UPDATE message, then the neighbor will answer that everything is correct with a proper ACK message.

2.14

We will write which routers are connected to the correct network

:

```
R1#show ip ospf database

            OSPF Router with ID (10.0.0.1) (Process ID 1)

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.1        10.0.0.1        1203        0x80000003 0x0013B6 2
10.0.0.2        10.0.0.2        1174        0x80000004 0x006FF5 4
10.0.0.3        10.0.0.3        1202        0x80000005 0x0095FF 3
10.0.0.4        10.0.0.4        1259        0x80000002 0x00FDF8 1
10.0.0.5        10.0.0.5        1237        0x80000003 0x0013B1 2
10.0.0.6        10.0.0.6        1191        0x80000004 0x000A81 3
10.0.0.7        10.0.0.7        1175        0x80000004 0x009AE7 3
10.0.0.8        10.0.0.8        1208        0x80000004 0x000A67 3

                Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.1.1        10.0.0.1        1203        0x80000001 0x008289
10.0.2.2        10.0.0.3        1205        0x80000001 0x007590
10.0.3.3        10.0.0.5        1247        0x80000002 0x005B90
10.0.4.1        10.0.0.5        1239        0x80000001 0x00A953
10.0.5.2        10.0.0.7        1192        0x80000001 0x009C5A
10.0.6.1        10.0.0.7        1176        0x80000001 0x006397
10.0.7.1        10.0.0.6        1366        0x80000001 0x006298
10.0.8.2        10.0.0.8        1377        0x80000001 0x008D63
10.0.9.1        10.0.0.8        1382        0x80000001 0x0038BE
10.0.10.1       10.0.0.8        1366        0x80000001 0x003BB9
R1#
```

```
R1#show ip ospf database network 10.0.3.3

            OSPF Router with ID (10.0.0.1) (Process ID 1)

                Net Link States (Area 1)

  Routing Bit Set on this LSA
  LS age: 1568
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 10.0.3.3 (address of Designated Router)
  Advertising Router: 10.0.0.5
  LS Seq Number: 80000002
  Checksum: 0x5B90
  Length: 36
  Network Mask: /24
        Attached Router: 10.0.0.5
        Attached Router: 10.0.0.3
        Attached Router: 10.0.0.4

R1#
```

```
R1#show ip ospf database router 10.0.0.3

            OSPF Router with ID (10.0.0.1) (Process ID 1)

                Router Link States (Area 1)

  LS age: 1585
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 10.0.0.3
  Advertising Router: 10.0.0.3
  LS Seq Number: 80000005
  Checksum: 0x95FF
  Length: 60
  Number of Links: 3

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 10.0.7.1
     (Link Data) Router Interface address: 10.0.7.2
      Number of TOS metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 10.0.3.3
     (Link Data) Router Interface address: 10.0.3.1
      Number of TOS metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 10.0.2.2
     (Link Data) Router Interface address: 10.0.2.2
      Number of TOS metrics: 0
       TOS 0 Metrics: 1
```

We will check which networks each router knows, in this example it is 10.0.0.3, we will notice that the router is holding a connection also to 10.0.1.1.

We found a route from our router to the designated network, there is a possibility to check routes through other routers that we saw that are connected to other networks.

2.15

The OSPF database is identical across all routers in the same area because OSPF synchronizes the database to ensure all routers have the same view of the network. Each router independently calculates its routing table so the database is the same.

2.16

R4 OSPF database:

```
R4#show ip ospf database

            OSPF Router with ID (10.0.0.4) (Process ID 1)

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.1        10.0.0.1        291         0x80000004 0x0011B7 2
10.0.0.2        10.0.0.2        288         0x80000005 0x006DF6 4
10.0.0.3        10.0.0.3        287         0x80000006 0x009301 3
10.0.0.4        10.0.0.4        378         0x80000003 0x00FBF9 1
10.0.0.5        10.0.0.5        359         0x80000004 0x0011B2 2
10.0.0.6        10.0.0.6        326         0x80000005 0x000882 3
10.0.0.7        10.0.0.7        281         0x80000005 0x0098E8 3
10.0.0.8        10.0.0.8        301         0x80000005 0x000868 3

                Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.1.1        10.0.0.1        291         0x80000002 0x00808A
10.0.2.2        10.0.0.3        287         0x80000002 0x007391
10.0.3.3        10.0.0.5        359         0x80000003 0x005991
10.0.4.1        10.0.0.5        359         0x80000002 0x00A754
10.0.5.2        10.0.0.7        283         0x80000002 0x009A5B
10.0.6.1        10.0.0.7        283         0x80000002 0x006198
10.0.7.1        10.0.0.6        328         0x80000002 0x006099
10.0.8.2        10.0.0.8        302         0x80000002 0x008B64
10.0.9.1        10.0.0.8        302         0x80000002 0x0036BF
10.0.10.1       10.0.0.8        302         0x80000002 0x0039BA
R4#
```

part 3:

3.6:

```
root@pc1: # traceroute 10.0.3.2
traceroute to 10.0.3.2 (10.0.3.2), 30 hops max, 60 byte packets
 1  192.0.1.254 (192.0.1.254)  5.757 ms  15.982 ms  26.850 ms
 2  10.0.9.1 (10.0.9.1)  36.967 ms  47.135 ms  57.863 ms
 3  10.0.8.1 (10.0.8.1)  68.589 ms  79.615 ms  89.792 ms
 4  10.0.5.1 (10.0.5.1)  110.691 ms  120.911 ms  100.013 ms
 5  10.0.7.2 (10.0.7.2)  130.905 ms  141.095 ms  150.969 ms
 6  10.0.3.2 (10.0.3.2)  161.751 ms  165.604 ms  167.007 ms
root@pc1:~# _
```

3.15
First R4 sends an update of the new link, in the form of LS update packet, which then spreads through R8 to R1. An acknowledge packet is sent to the transmitting router from every link it sent to (in the case of R4 there are two Routers connected, so we are seeing two ACKs).
R4:

```
12:35:06.636949 10.0.3.2    224.0.0.5    OSPF    110 LS Update
12:35:09.161274 10.0.3.3    224.0.0.5    OSPF     78 LS Acknowledge
12:35:09.171985 10.0.3.1    224.0.0.6    OSPF     78 LS Acknowledge
12:35:11.905611 10.0.3.1    224.0.0.5    OSPF     98 Hello Packet
12:35:11.955265 10.0.3.2    224.0.0.5    OSPF     98 Hello Packet
12:35:12.082598 10.0.3.3    224.0.0.5    OSPF     98 Hello Packet
12:35:12.621684 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=81/20736, ttl=59 (no response fou
12:35:13.624256 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=82/20992, ttl=59 (reply in 78)
12:35:13.645669 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=82/20992, ttl=63 (request in 77)
12:35:14.626603 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=83/21248, ttl=59 (reply in 80)
12:35:14.648138 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=83/21248, ttl=63 (request in 79)
12:35:15.632023 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=84/21504, ttl=59 (reply in 82)
```

R8:

```
12:35:02.042143 10.0.8.1    224.0.0.5    OSPF     94 Hello Packet
12:35:06.668806 10.0.8.1    224.0.0.5    OSPF    110 LS Update
12:35:09.161274 10.0.8.2    224.0.0.5    OSPF     78 LS Acknowledge
12:35:11.670558 10.0.8.2    224.0.0.5    OSPF     94 Hello Packet
12:35:12.040727 10.0.8.1    224.0.0.5    OSPF     94 Hello Packet
12:35:12.590077 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=81/20736, ttl=62 (no response fou
12:35:13.592136 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=82/20992, ttl=62 (reply in 55)
12:35:13.677511 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=82/20992, ttl=60 (request in 54)
12:35:14.594507 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=83/21248, ttl=62 (reply in 57)
12:35:14.681028 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=83/21248, ttl=60 (request in 56)
12:35:15.599930 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=84/21504, ttl=62 (reply in 59)
12:35:15.685152 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=84/21504, ttl=60 (request in 58)
12:35:16.595970 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=85/21760, ttl=62 (reply in 61)
```

R1:

```
12:35:01.671391 10.0.9.1    224.0.0.5    OSPF     94 Hello Packet
12:35:02.232538 10.0.9.2    224.0.0.5    OSPF     94 Hello Packet
12:35:06.678935 10.0.9.1    224.0.0.5    OSPF    110 LS Update
12:35:09.158865 10.0.9.2    224.0.0.5    OSPF     78 LS Acknowledge
12:35:11.670558 10.0.9.1    224.0.0.5    OSPF     94 Hello Packet
12:35:12.231694 10.0.9.2    224.0.0.5    OSPF     94 Hello Packet
12:35:12.579721 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=81/20736, ttl=63 (no response fo
12:35:13.581430 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=82/20992, ttl=63 (reply in 63)
12:35:13.688234 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=82/20992, ttl=59 (request in 62)
12:35:14.583880 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=83/21248, ttl=63 (reply in 65)
12:35:14.691724 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=83/21248, ttl=59 (request in 64)
12:35:15.589533 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=84/21504, ttl=63 (reply in 67)
12:35:15.695215 192.0.2.1   192.0.1.1    ICMP     98 Echo (ping) reply    id=0x037d, seq=84/21504, ttl=59 (request in 66)
12:35:16.585910 192.0.1.1   192.0.2.1    ICMP     98 Echo (ping) request  id=0x037d, seq=85/21760, ttl=63 (reply in 69)
```

Then, as we can see, PINGs are exchanged.

3.16

As can be seen in the “time” column of the captures, it takes 0.041986 seconds for the OSPF info to spread through the entire network and 0.010129 seconds to go through a single hop.

3.17

OSPF calculates the metric, cost of an interface as:

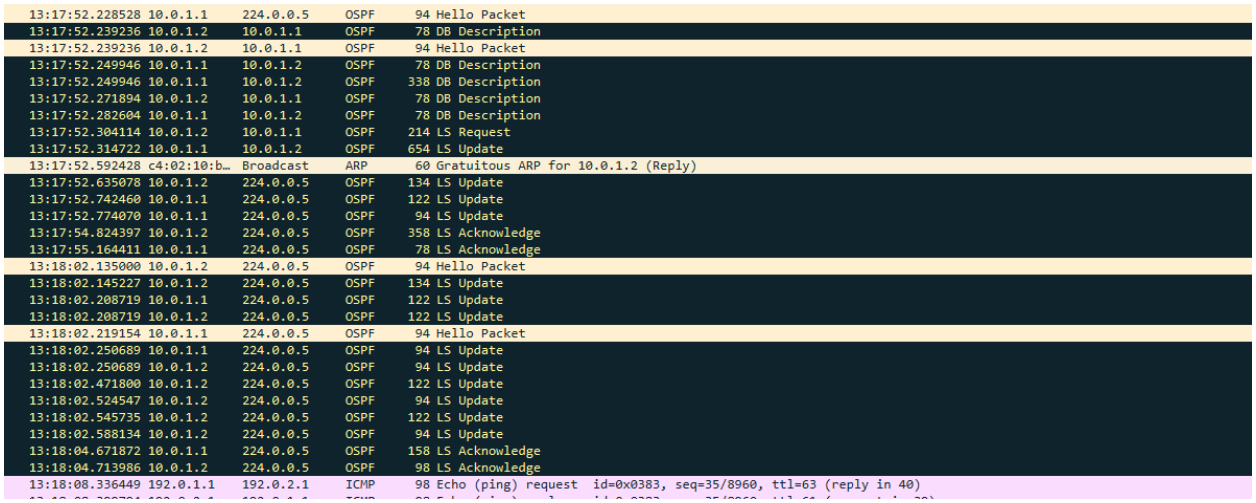
Cost = (Reference BandwidthInterface) / (Interface Bandwidth)

The interface bandwidth is usually 100 Mbs, meaning that interfaces with lower bandwidths have a higher cost. The command “show interfaces FastEthernet interface-number” helps to calculate the cost because it provides, among other things, the bandwidth of said interface.

Part 4

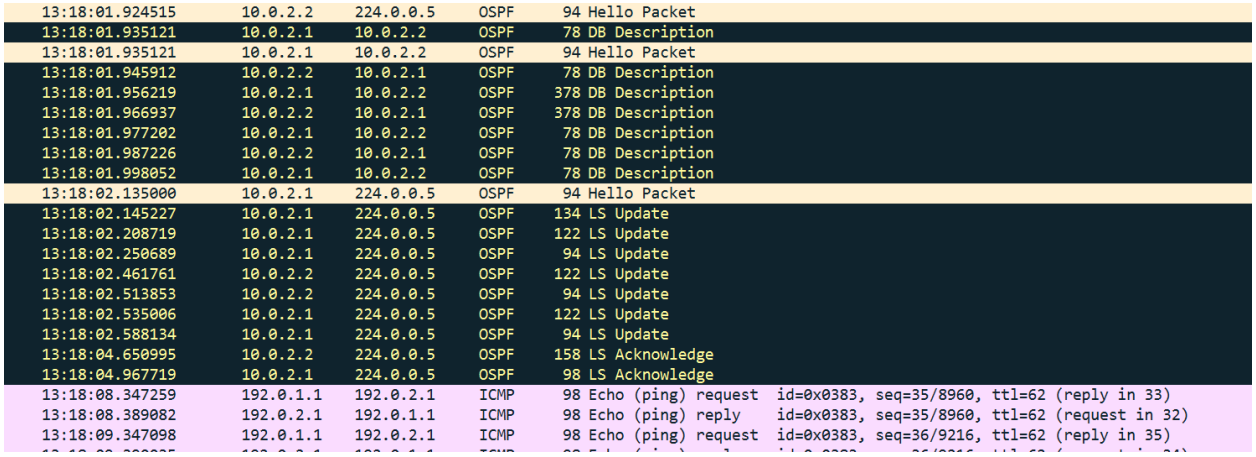
4.10

R1-R2:



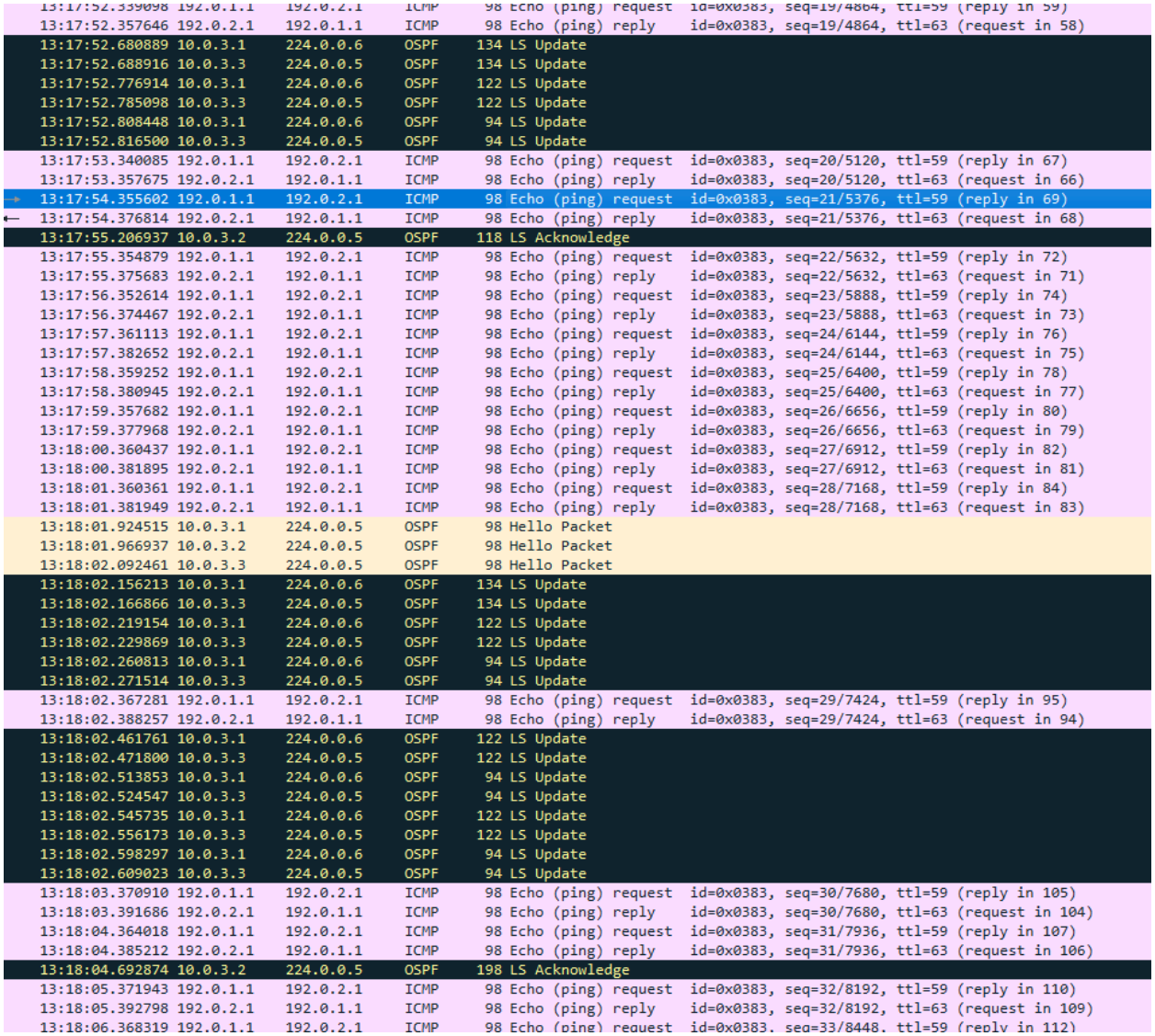
13:17:52.228528	10.0.1.1	224.0.0.5	OSPF	94	Hello Packet
13:17:52.239236	10.0.1.2	10.0.1.1	OSPF	78	DB Description
13:17:52.239236	10.0.1.2	10.0.1.1	OSPF	94	Hello Packet
13:17:52.249946	10.0.1.1	10.0.1.2	OSPF	78	DB Description
13:17:52.249946	10.0.1.1	10.0.1.2	OSPF	338	DB Description
13:17:52.271894	10.0.1.2	10.0.1.1	OSPF	78	DB Description
13:17:52.282604	10.0.1.1	10.0.1.2	OSPF	78	DB Description
13:17:52.304114	10.0.1.2	10.0.1.1	OSPF	214	LS Request
13:17:52.314722	10.0.1.1	10.0.1.2	OSPF	654	LS Update
13:17:52.592428	c4:02:10:b…	Broadcast	ARP	60	Gratuitous ARP for 10.0.1.2 (Reply)
13:17:52.635078	10.0.1.2	224.0.0.5	OSPF	134	LS Update
13:17:52.742460	10.0.1.1	224.0.0.5	OSPF	122	LS Update
13:17:52.774070	10.0.1.1	224.0.0.5	OSPF	94	LS Update
13:17:54.824397	10.0.1.2	224.0.0.5	OSPF	358	LS Acknowledge
13:17:55.164411	10.0.1.1	224.0.0.5	OSPF	78	LS Acknowledge
13:18:02.135000	10.0.1.2	224.0.0.5	OSPF	94	Hello Packet
13:18:02.145227	10.0.1.2	224.0.0.5	OSPF	134	LS Update
13:18:02.208719	10.0.1.1	224.0.0.5	OSPF	122	LS Update
13:18:02.208719	10.0.1.2	224.0.0.5	OSPF	122	LS Update
13:18:02.219154	10.0.1.1	224.0.0.5	OSPF	94	Hello Packet
13:18:02.250689	10.0.1.1	224.0.0.5	OSPF	94	LS Update
13:18:02.250689	10.0.1.2	224.0.0.5	OSPF	94	LS Update
13:18:02.471800	10.0.1.2	224.0.0.5	OSPF	122	LS Update
13:18:02.524547	10.0.1.2	224.0.0.5	OSPF	94	LS Update
13:18:02.545735	10.0.1.2	224.0.0.5	OSPF	122	LS Update
13:18:02.588134	10.0.1.2	224.0.0.5	OSPF	94	LS Update
13:18:04.671872	10.0.1.1	224.0.0.5	OSPF	158	LS Acknowledge
13:18:04.713986	10.0.1.2	224.0.0.5	OSPF	98	LS Acknowledge
13:18:08.336449	192.0.1.1	192.0.2.1	ICMP	98	Echo (ping) request id=0x0383, seq=35/8960, ttl=63 (reply in 40)

R2-R3:



13:18:01.924515	10.0.2.2	224.0.0.5	OSPF	94	Hello Packet
13:18:01.935121	10.0.2.1	10.0.2.2	OSPF	78	DB Description
13:18:01.935121	10.0.2.1	10.0.2.2	OSPF	94	Hello Packet
13:18:01.945912	10.0.2.2	10.0.2.1	OSPF	78	DB Description
13:18:01.956219	10.0.2.1	10.0.2.2	OSPF	378	DB Description
13:18:01.966937	10.0.2.2	10.0.2.1	OSPF	378	DB Description
13:18:01.977202	10.0.2.1	10.0.2.2	OSPF	78	DB Description
13:18:01.987226	10.0.2.2	10.0.2.1	OSPF	78	DB Description
13:18:01.998052	10.0.2.1	10.0.2.2	OSPF	78	DB Description
13:18:02.135000	10.0.2.1	224.0.0.5	OSPF	94	Hello Packet
13:18:02.145227	10.0.2.1	224.0.0.5	OSPF	134	LS Update
13:18:02.208719	10.0.2.1	224.0.0.5	OSPF	122	LS Update
13:18:02.250689	10.0.2.1	224.0.0.5	OSPF	94	LS Update
13:18:02.461761	10.0.2.2	224.0.0.5	OSPF	122	LS Update
13:18:02.513853	10.0.2.2	224.0.0.5	OSPF	94	LS Update
13:18:02.535006	10.0.2.1	224.0.0.5	OSPF	122	LS Update
13:18:02.588134	10.0.2.1	224.0.0.5	OSPF	94	LS Update
13:18:04.650995	10.0.2.2	224.0.0.5	OSPF	158	LS Acknowledge
13:18:04.967719	10.0.2.1	224.0.0.5	OSPF	98	LS Acknowledge
13:18:08.347259	192.0.1.1	192.0.2.1	ICMP	98	Echo (ping) request id=0x0383, seq=35/8960, ttl=62 (reply in 33)
13:18:08.389082	192.0.2.1	192.0.1.1	ICMP	98	Echo (ping) reply id=0x0383, seq=35/8960, ttl=62 (request in 32)
13:18:09.347098	192.0.1.1	192.0.2.1	ICMP	98	Echo (ping) request id=0x0383, seq=36/9216, ttl=62 (reply in 35)

R4:



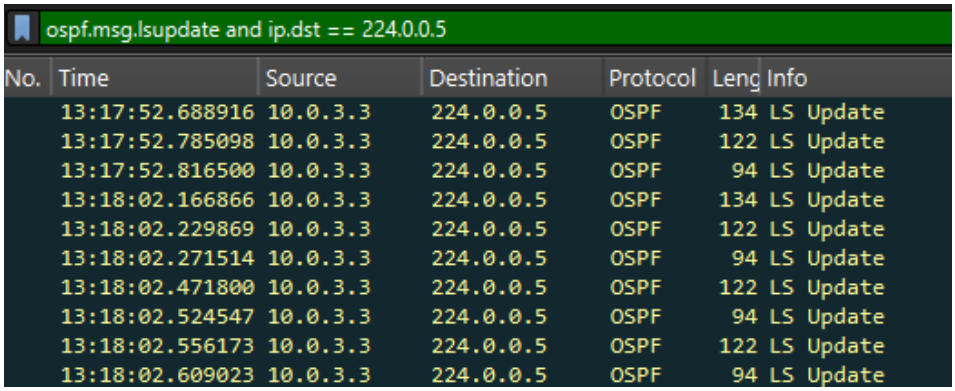
	13:17:52.339098	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=19/4864, ttl=59 (reply in 59)
	13:17:52.357646	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=19/4864, ttl=63 (request in 58)
	13:17:52.680889	10.0.3.1	224.0.0.6	OSPF	134 LS Update
	13:17:52.688916	10.0.3.3	224.0.0.5	OSPF	134 LS Update
	13:17:52.776914	10.0.3.1	224.0.0.6	OSPF	122 LS Update
	13:17:52.785098	10.0.3.3	224.0.0.5	OSPF	122 LS Update
	13:17:52.808448	10.0.3.1	224.0.0.6	OSPF	94 LS Update
	13:17:52.816500	10.0.3.3	224.0.0.5	OSPF	94 LS Update
	13:17:53.340085	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=20/5120, ttl=59 (reply in 67)
	13:17:53.357675	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=20/5120, ttl=63 (request in 66)
→	13:17:54.355602	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=21/5376, ttl=59 (reply in 69)
←	13:17:54.376814	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=21/5376, ttl=63 (request in 68)
	13:17:55.206937	10.0.3.2	224.0.0.5	OSPF	118 LS Acknowledge
	13:17:55.354879	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=22/5632, ttl=59 (reply in 72)
	13:17:55.375683	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=22/5632, ttl=63 (request in 71)
	13:17:56.352614	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=23/5888, ttl=59 (reply in 74)
	13:17:56.374467	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=23/5888, ttl=63 (request in 73)
	13:17:57.361113	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=24/6144, ttl=59 (reply in 76)
	13:17:57.382652	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=24/6144, ttl=63 (request in 75)
	13:17:58.359252	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=25/6400, ttl=59 (reply in 78)
	13:17:58.380945	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=25/6400, ttl=63 (request in 77)
	13:17:59.357682	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=26/6656, ttl=59 (reply in 80)
	13:17:59.377968	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=26/6656, ttl=63 (request in 79)
	13:18:00.360437	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=27/6912, ttl=59 (reply in 82)
	13:18:00.381895	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=27/6912, ttl=63 (request in 81)
	13:18:01.360361	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=28/7168, ttl=59 (reply in 84)
	13:18:01.381949	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=28/7168, ttl=63 (request in 83)
	13:18:01.924515	10.0.3.1	224.0.0.5	OSPF	98 Hello Packet
	13:18:01.966937	10.0.3.2	224.0.0.5	OSPF	98 Hello Packet
	13:18:02.092461	10.0.3.3	224.0.0.5	OSPF	98 Hello Packet
	13:18:02.156213	10.0.3.1	224.0.0.6	OSPF	134 LS Update
	13:18:02.166866	10.0.3.3	224.0.0.5	OSPF	134 LS Update
	13:18:02.219154	10.0.3.1	224.0.0.6	OSPF	122 LS Update
	13:18:02.229869	10.0.3.3	224.0.0.5	OSPF	122 LS Update
	13:18:02.260813	10.0.3.1	224.0.0.6	OSPF	94 LS Update
	13:18:02.271514	10.0.3.3	224.0.0.5	OSPF	94 LS Update
	13:18:02.367281	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=29/7424, ttl=59 (reply in 95)
	13:18:02.388257	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=29/7424, ttl=63 (request in 94)
	13:18:02.461761	10.0.3.1	224.0.0.6	OSPF	122 LS Update
	13:18:02.471800	10.0.3.3	224.0.0.5	OSPF	122 LS Update
	13:18:02.513853	10.0.3.1	224.0.0.6	OSPF	94 LS Update
	13:18:02.524547	10.0.3.3	224.0.0.5	OSPF	94 LS Update
	13:18:02.545735	10.0.3.1	224.0.0.6	OSPF	122 LS Update
	13:18:02.556173	10.0.3.3	224.0.0.5	OSPF	122 LS Update
	13:18:02.598297	10.0.3.1	224.0.0.6	OSPF	94 LS Update
	13:18:02.609023	10.0.3.3	224.0.0.5	OSPF	94 LS Update
	13:18:03.370910	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=30/7680, ttl=59 (reply in 105)
	13:18:03.391686	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=30/7680, ttl=63 (request in 104)
	13:18:04.364018	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=31/7936, ttl=59 (reply in 107)
	13:18:04.385212	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=31/7936, ttl=63 (request in 106)
	13:18:04.692874	10.0.3.2	224.0.0.5	OSPF	198 LS Acknowledge
	13:18:05.371943	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=32/8192, ttl=59 (reply in 110)
	13:18:05.392798	192.0.2.1	192.0.1.1	ICMP	98 Echo (ping) reply id=0x0383, seq=32/8192, ttl=63 (request in 109)
	13:18:06.368319	192.0.1.1	192.0.2.1	ICMP	98 Echo (ping) request id=0x0383, seq=33/8448, ttl=59 (reply in 112)

No, the PING application was not affected and no PING packets were lost. We can see that the TTL of the pings improved, before the update ping reached their destination with TTL=59. And after TTL of 61, meaning less hops on their way.

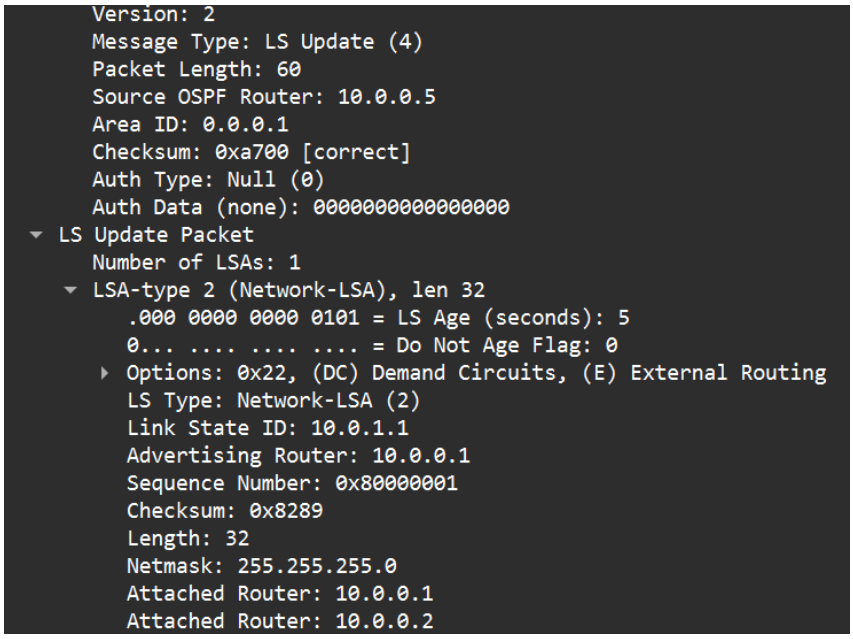
4.11

You can tell how long it took the network to converge by looking at the first update that R2 is sending, at 13:17:52.23, and when R4 sends and LS ACK packet to show that it received the update, at 13:18:04.69. Meaning that it took the network about 12.4 seconds to converge.

4.12



R5 is sending update messages to inform about changes in the network, for example:



An early message to inform the network that there is a network 10.0.1.0/24 between R1 and R2.

Part 5

```
root@pc2:~#
root@pc2:~#
root@pc2:~#
root@pc2:~#
root@pc2:~#
root@pc2:~#
root@pc2:~# ping 192.0.1.1
PING 192.0.1.1 (192.0.1.1) 56(84) bytes of data.
64 bytes from 192.0.1.1: icmp_seq=1 ttl=60 time=77.3 ms
64 bytes from 192.0.1.1: icmp_seq=2 ttl=60 time=78.7 ms
64 bytes from 192.0.1.1: icmp_seq=3 ttl=60 time=77.3 ms
64 bytes from 192.0.1.1: icmp_seq=4 ttl=60 time=77.9 ms
64 bytes from 192.0.1.1: icmp_seq=5 ttl=60 time=79.4 ms
64 bytes from 192.0.1.1: icmp_seq=6 ttl=60 time=77.2 ms
64 bytes from 192.0.1.1: icmp_seq=7 ttl=60 time=76.9 ms
64 bytes from 192.0.1.1: icmp_seq=8 ttl=60 time=74.4 ms
64 bytes from 192.0.1.1: icmp_seq=9 ttl=60 time=83.0 ms
64 bytes from 192.0.1.1: icmp_seq=10 ttl=60 time=78.7 ms
64 bytes from 192.0.1.1: icmp_seq=50 ttl=58 time=120 ms
64 bytes from 192.0.1.1: icmp_seq=51 ttl=58 time=121 ms
64 bytes from 192.0.1.1: icmp_seq=52 ttl=58 time=128 ms
64 bytes from 192.0.1.1: icmp_seq=53 ttl=58 time=128 ms
64 bytes from 192.0.1.1: icmp_seq=54 ttl=58 time=127 ms
64 bytes from 192.0.1.1: icmp_seq=55 ttl=58 time=124 ms
64 bytes from 192.0.1.1: icmp_seq=56 ttl=58 time=121 ms
^C
--- 192.0.1.1 ping statistics ---
56 packets transmitted, 17 received, 69% packet loss, time 55133ms
rtt min/avg/max/mdev = 74.418/97.227/128.126/22.979 ms
root@pc2:~#
```

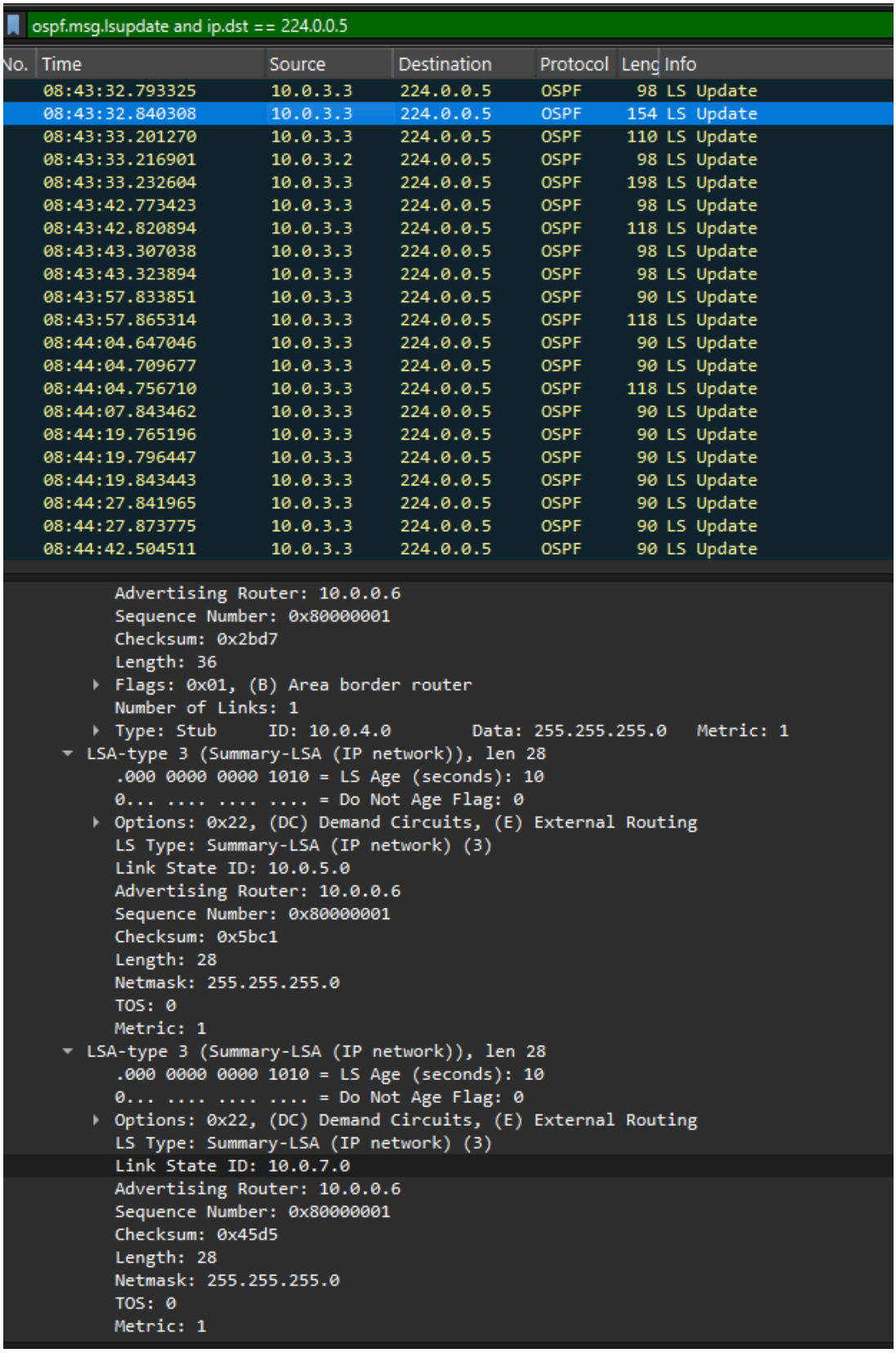
5.9

We turned R2 off about 8 seconds after the start of the PING sending, and saw the ACKs stop. Than, about 39 seconds later, ACKs started arriving again, taking more time to arrive (120 ms in compare to about 80 ms).

5.10

The waiting makes sense given the dead-interval protocol, which dictates that routers wait for usually 40 seconds between the last “hello” packet to declare a route as down. After the route is declared down, the network updates and finds the new shortest path, thats when we see the new ACK messages start arriving.

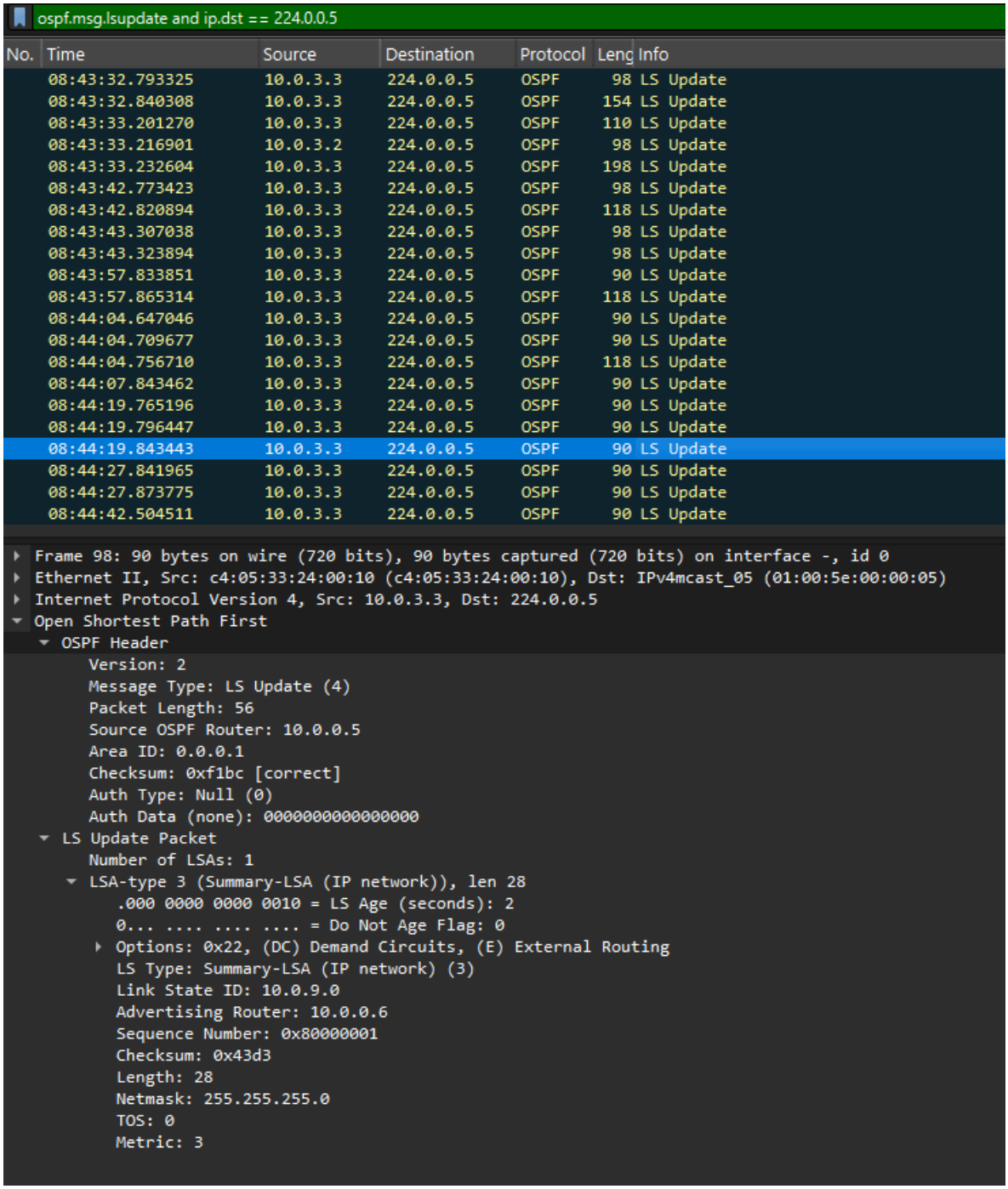
Part 6


ospf.msg.lsupdate and ip.dst == 224.0.0.5
No. Time Source Destination Protocol Leng Info
08:43:32.793325 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:32.840308 10.0.3.3 224.0.0.5 OSPF 154 LS Update
08:43:33.201270 10.0.3.3 224.0.0.5 OSPF 110 LS Update
08:43:33.216901 10.0.3.2 224.0.0.5 OSPF 98 LS Update
08:43:33.232604 10.0.3.3 224.0.0.5 OSPF 198 LS Update
08:43:42.773423 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:42.820894 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:43:43.307038 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:43.323894 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:57.833851 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:43:57.865314 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:44:04.647046 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:04.709677 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:04.756710 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:44:07.843462 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.765196 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.796447 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.843443 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:27.841965 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:27.873775 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:42.504511 10.0.3.3 224.0.0.5 OSPF 90 LS Update
Advertising Router: 10.0.0.6
Sequence Number: 0x80000001
Checksum: 0x2bd7
Length: 36
Flags: 0x01, (B) Area border router
Number of Links: 1
Type: Stub ID: 10.0.4.0 Data: 255.255.255.0 Metric: 1
LSA-type 3 (Summary-LSA (IP network)), len 28
.000 0000 0000 1010 = LS Age (seconds): 10
0... .... = Do Not Age Flag: 0
Options: 0x22, (DC) Demand Circuits, (E) External Routing
LS Type: Summary-LSA (IP network) (3)
Link State ID: 10.0.5.0
Advertising Router: 10.0.0.6
Sequence Number: 0x80000001
Checksum: 0x5bc1
Length: 28
Netmask: 255.255.255.0
TOS: 0
Metric: 1
LSA-type 3 (Summary-LSA (IP network)), len 28
.000 0000 0000 1010 = LS Age (seconds): 10
0... .... = Do Not Age Flag: 0
Options: 0x22, (DC) Demand Circuits, (E) External Routing
LS Type: Summary-LSA (IP network) (3)
Link State ID: 10.0.7.0
Advertising Router: 10.0.0.6
Sequence Number: 0x80000001
Checksum: 0x45d5
Length: 28
Netmask: 255.255.255.0
TOS: 0
Metric: 1

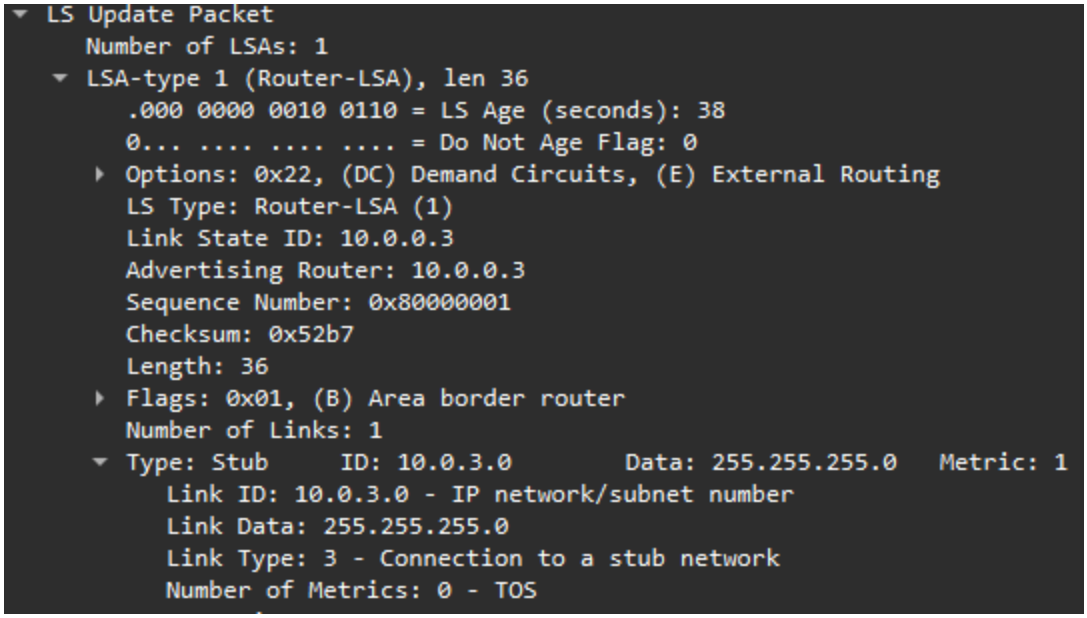
6.11

For the backbone area - R4 receives information about the router OSPF ip of R6 and R3 only, and the networks each is connected to - as can be seen in the picture above - R6 is 10.0.0.6 and connected to 10.0.5.0/24 and 10.0.7.0/24.

For area 2 R4 receives routing for the next router - but not information about the routers themselves, or the components in those networks. For example - to reach 10.0.9.0/24 go to R6:


ospf.msg.lsupdate and ip.dst == 224.0.0.5
No. Time Source Destination Protocol Leng Info
08:43:32.793325 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:32.840308 10.0.3.3 224.0.0.5 OSPF 154 LS Update
08:43:33.201270 10.0.3.3 224.0.0.5 OSPF 110 LS Update
08:43:33.216901 10.0.3.2 224.0.0.5 OSPF 98 LS Update
08:43:33.232604 10.0.3.3 224.0.0.5 OSPF 198 LS Update
08:43:42.773423 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:42.820894 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:43:43.307038 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:43.323894 10.0.3.3 224.0.0.5 OSPF 98 LS Update
08:43:57.833851 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:43:57.865314 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:44:04.647046 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:04.709677 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:04.756710 10.0.3.3 224.0.0.5 OSPF 118 LS Update
08:44:07.843462 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.765196 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.796447 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:19.843443 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:27.841965 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:27.873775 10.0.3.3 224.0.0.5 OSPF 90 LS Update
08:44:42.504511 10.0.3.3 224.0.0.5 OSPF 90 LS Update
Frame 98: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface -, id 0
Ethernet II, Src: c4:05:33:24:00:10 (c4:05:33:24:00:10), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
Internet Protocol Version 4, Src: 10.0.3.3, Dst: 224.0.0.5
Open Shortest Path First
OSPF Header
Version: 2
Message Type: LS Update (4)
Packet Length: 56
Source OSPF Router: 10.0.0.5
Area ID: 0.0.0.1
Checksum: 0xf1bc [correct]
Auth Type: Null (0)
Auth Data (none): 0000000000000000
LS Update Packet
Number of LSAs: 1
LSA-type 3 (Summary-LSA (IP network)), len 28
.000 0000 0000 0010 = LS Age (seconds): 2
0... .... = Do Not Age Flag: 0
Options: 0x22, (DC) Demand Circuits, (E) External Routing
LS Type: Summary-LSA (IP network) (3)
Link State ID: 10.0.9.0
Advertising Router: 10.0.0.6
Sequence Number: 0x80000001
Checksum: 0x43d3
Length: 28
Netmask: 255.255.255.0
TOS: 0
Metric: 3

6.12



R3 advertises that it is the connection to 10.0.3.0/24 network.

6.13

```
R4#show ip ospf database

            OSPF Router with ID (10.0.0.4) (Process ID 1)

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.3        10.0.0.3        67          0x80000003 0x0001F6 1
10.0.0.4        10.0.0.4        60          0x80000003 0x00FBF9 1
10.0.0.5        10.0.0.5        76          0x80000003 0x0013B1 2
10.0.0.6        10.0.0.6        61          0x80000003 0x00E010 1

                Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.3.3        10.0.0.5        76          0x80000003 0x005991
10.0.4.1        10.0.0.5        76          0x80000002 0x00A754

                Summary Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.1.0        10.0.0.3        67          0x80000002 0x00A180
10.0.1.0        10.0.0.6        61          0x80000002 0x009984
10.0.2.0        10.0.0.3        67          0x80000002 0x008C95
10.0.2.0        10.0.0.6        61          0x80000002 0x008499
10.0.5.0        10.0.0.3        72          0x80000002 0x0075A8
10.0.5.0        10.0.0.6        65          0x80000002 0x0059C2
10.0.6.0        10.0.0.3        72          0x80000002 0x006AB2
10.0.6.0        10.0.0.6        65          0x80000003 0x0056C2
10.0.7.0        10.0.0.3        72          0x80000002 0x0055C7
10.0.7.0        10.0.0.6        65          0x80000002 0x0043D6
10.0.8.0        10.0.0.3        72          0x80000002 0x005EBB
10.0.8.0        10.0.0.6        65          0x80000002 0x0042D5
10.0.9.0        10.0.0.3        72          0x80000003 0x0051C6
10.0.9.0        10.0.0.6        65          0x80000002 0x0041D4
10.0.10.0       10.0.0.3        72          0x80000002 0x003EDA
10.0.10.0       10.0.0.6        65          0x80000002 0x0036DE
R4#
```

This time R4, which is not a border router, does not hold the knowledge of the entire network.

R4 knows the networks in area 1 and which router holds them(10.0.3.3 / 10.0.4.1 - R5), knows what routers are connected to area 1 in area 0 (R6,R3), and knows of the existence of all other networks in area 0 and area 2, and through which router in area 0 to reach these networks - but it isn't aware of the existence of other routers in area 0 or any in area 2.

6.14

```
R4#show ip ospf database summary 10.0.1.0

          OSPF Router with ID (10.0.0.4) (Process ID 1)

              Summary Net Link States (Area 1)

 Routing Bit Set on this LSA
 LS age: 924
 Options: (No TOS-capability, DC, Upward)
 LS Type: Summary Links(Network)
 Link State ID: 10.0.1.0 (summary Network Number)
 Advertising Router: 10.0.0.3
 LS Seq Number: 80000002
 Checksum: 0xA180
 Length: 28
 Network Mask: /24
       TOS: 0  Metric: 2

 LS age: 917
 Options: (No TOS-capability, DC, Upward)
 LS Type: Summary Links(Network)
 Link State ID: 10.0.1.0 (summary Network Number)
 Advertising Router: 10.0.0.6
 LS Seq Number: 80000002
 Checksum: 0x9984
 Length: 28
 Network Mask: /24
       TOS: 0  Metric: 3
```

```
R4#show ip ospf database router 10.0.0.3

            OSPF Router with ID (10.0.0.4) (Process ID 1)

                Router Link States (Area 1)

  Routing Bit Set on this LSA
  LS age: 1064
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 10.0.0.3
  Advertising Router: 10.0.0.3
  LS Seq Number: 80000003
  Checksum: 0x1F6
  Length: 36
  Area Border Router
  Number of Links: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 10.0.3.3
     (Link Data) Router Interface address: 10.0.3.1
      Number of TOS metrics: 0
       TOS 0 Metrics: 1
```

From the given commands we can see that to reach network 10.0.1.0/24 we need to go through R3, 10.0.0.3. As we can tell from the show ip ospf database router 10.0.0.3 command, R3 is a link router, meaning it is connecting area 1 and area 0.

The given commands do not show us any information about other routers, such as R1 or R2 because this information does not exist in the R4 database

6.15

```
R7#show ip ospf database

            OSPF Router with ID (10.0.0.7) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.2        10.0.0.2        1930        0x80000004 0x0006C2 2
10.0.0.3        10.0.0.3        1982        0x80000004 0x00269D 2
10.0.0.6        10.0.0.6        1973        0x80000004 0x005861 2
10.0.0.7        10.0.0.7        1949        0x80000004 0x004C6C 2

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.2.2        10.0.0.3        1982        0x80000002 0x007391
10.0.5.2        10.0.0.7        1949        0x80000002 0x009A5B
10.0.6.1        10.0.0.7        1949        0x80000002 0x006198
10.0.7.1        10.0.0.6        1973        0x80000002 0x006099

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.1.0        10.0.0.2        1930        0x80000002 0x009D86
10.0.1.0        10.0.0.7        1949        0x80000002 0x009389
10.0.3.0        10.0.0.3        1988        0x80000002 0x00819F
10.0.3.0        10.0.0.6        1980        0x80000002 0x0079A3
10.0.4.0        10.0.0.3        1988        0x80000002 0x00809E
10.0.4.0        10.0.0.6        1980        0x80000002 0x0064B8
10.0.8.0        10.0.0.2        1937        0x80000002 0x005AC1
10.0.8.0        10.0.0.7        1956        0x80000002 0x0032E5
10.0.9.0        10.0.0.2        1937        0x80000002 0x004FCB
10.0.9.0        10.0.0.7        1956        0x80000002 0x0031E4
10.0.10.0       10.0.0.2        1937        0x80000002 0x003AE0
10.0.10.0       10.0.0.7        1956        0x80000002 0x0026EE

                Router Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.0.0.1        10.0.0.1        1881        0x80000004 0x0027A0 2
10.0.0.2        10.0.0.2        1938        0x80000004 0x0017AA 2
10.0.0.7        10.0.0.7        1956        0x80000003 0x001BCC 1
10.0.0.8        10.0.0.8        1944        0x80000005 0x00E58B 3

                Net Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.1.2        10.0.0.2        1940        0x80000002 0x006C9C
10.0.8.1        10.0.0.7        1959        0x80000002 0x009F52
10.0.9.1        10.0.0.8        1946        0x80000002 0x0036BF
10.0.10.1       10.0.0.8        1946        0x80000002 0x0039BA

                Summary Net Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.2.0        10.0.0.2        1941        0x80000002 0x009290
10.0.2.0        10.0.0.7        1959        0x80000003 0x007C9F
10.0.3.0        10.0.0.2        1941        0x80000002 0x00918F
10.0.3.0        10.0.0.7        1959        0x80000002 0x007D9D
10.0.4.0        10.0.0.2        1941        0x80000002 0x00908E
10.0.4.0        10.0.0.7        1959        0x80000002 0x0068B2
10.0.5.0        10.0.0.2        1941        0x80000003 0x0079A4
10.0.5.0        10.0.0.7        1959        0x80000002 0x0053C7
10.0.6.0        10.0.0.2        1941        0x80000002 0x0066B8
10.0.6.0        10.0.0.7        1959        0x80000002 0x0048D1
10.0.7.0        10.0.0.2        1943        0x80000002 0x0065B7
10.0.7.0        10.0.0.7        1961        0x80000002 0x0047D0
R7#
```

The first thing you can tell is that R7's database is much larger. It holds information about all of area 0 and area 2 which includes all the networks and routers, and information about area 1 through area 0 links.

he advantage is the division into areas, which reduces the amount of data exchanged between routers. Each router only knows the routers within its area and the border routers. As a result, updates within a single area are faster (they pass through fewer routers and are smaller in size).

On the other hand, this is also a disadvantage—there’s no way to know the full topology, only the routers within the same area. If a border router fails and there’s no alternative connection to the backbone (or the desired area), communication will be disrupted, potentially causing significant issues.