Assignment 5

Open Shortest Path First (OSPF)

By Ziming Song
zs2815@nyu.edu

1. Screenshot configurations of R1, R2, R3, and R4.

```
student@CN-R1:~$ sudo cat /etc/frr/frr.conf
frr version 7.1
frr defaults traditional
hostname CN-R1
log syslog informational
service integrated-vtysh-config
!
interface eth1
 ip address 10.10.10.1/29
!
router ospf
 network 10.10.10.0/29 area 0
 network 10.20.1.0/24 area 0
!
line vty
!
student@CN-R1:~$
```

Screenshot configurations of R1

```
student@CN-R2: ~
File Edit View Search Terminal Help
frr version 7.1
frr defaults traditional
hostname CN-R2
log syslog informational
service integrated-vtysh-config
!
ip route 128.238.77.0/24 10.10.10.1
!
interface eth0
 ip address 10.10.10.2/29
!
interface eth1
 ip address 10.10.11.1/30
!
interface eth2
 ip address 10.10.11.5/30
!
router ospf
 network 10.10.10.0/29 area 0
 network 10.10.11.0/28 area 1
!
line vty
!
student@CN-R2:~$
```

Screenshot configurations of R2

```
student@CN-R3:~$ sudo cat /etc/frr/frr.conf
frr version 7.1
frr defaults traditional
hostname CN-R3
log syslog informational
service integrated-vtysh-config
!
ip route 128.238.77.0/0 10.10.11.1
!
interface eth0
 ip address 10.10.11.2/30
!
interface eth1
 ip address 10.10.11.9/30
!
router ospf
 network 10.10.11.0/28 area 1
!
line vty
!
student@CN-R3:~$
```

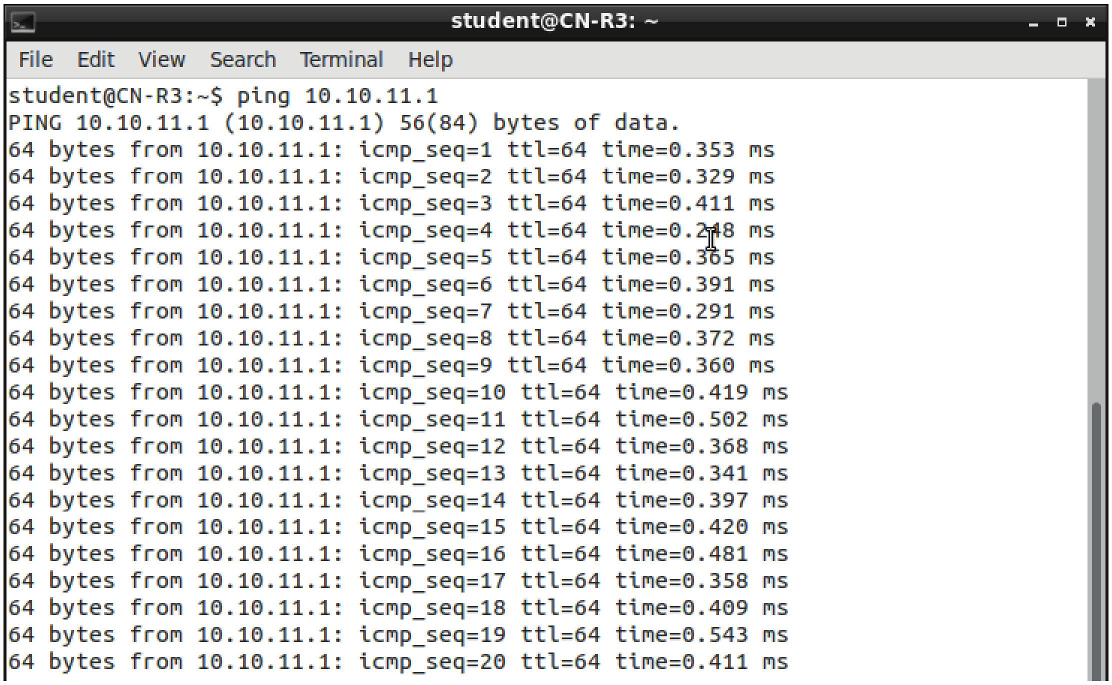
Screenshot configurations of R3

```
student@CN-R4:~$ sudo cat /etc/frr/frr.conf
frr version 7.1
frr defaults traditional
hostname CN-R4
log syslog informational
service integrated-vtysh-config
!
ip route 128.238.77.0/24 10.10.11.5
!
interface eth0
 ip address 10.10.11.10/30
!
interface eth1
 ip address 10.10.11.6/30
!
interface eth2
 ip address 10.10.11.17/28
!
router ospf
 network 10.10.11.0/28 area 1
!
line vty
!
```

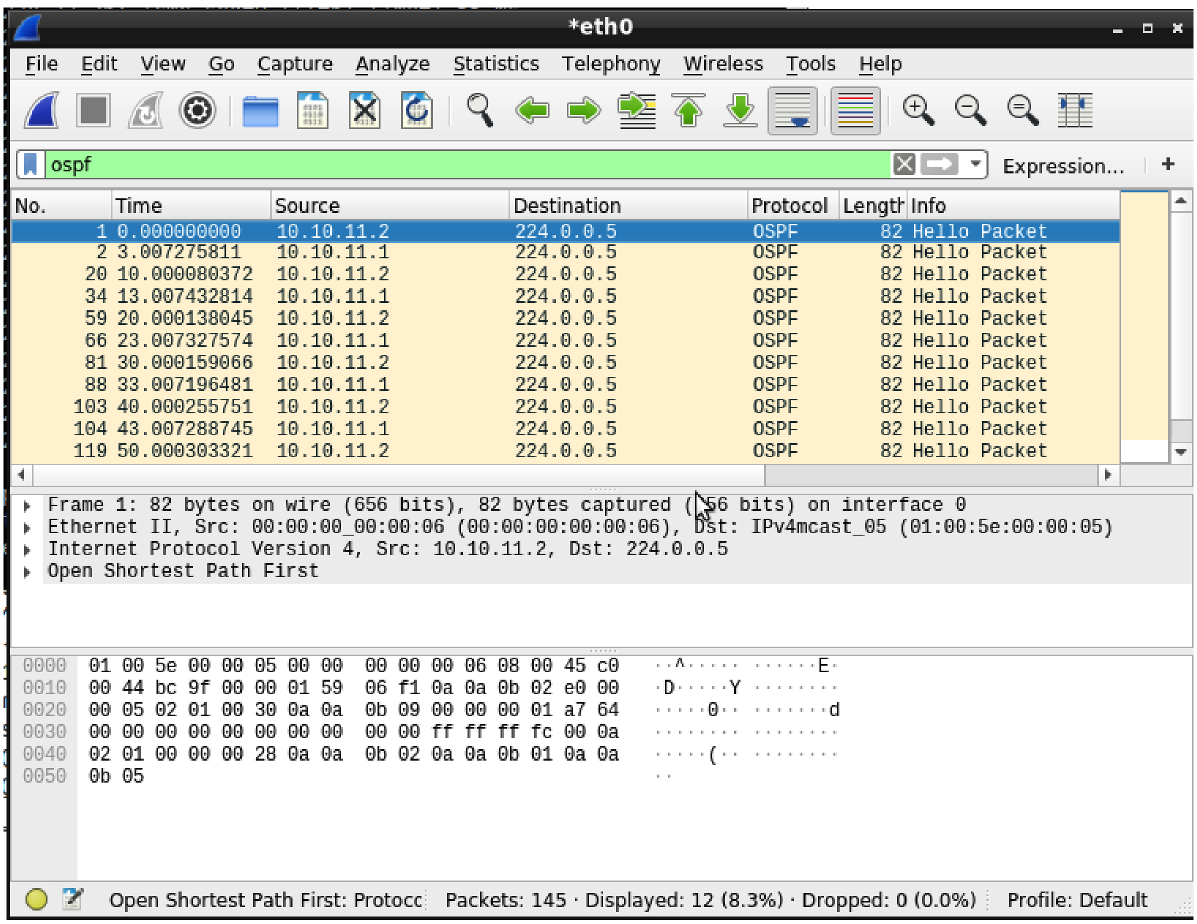
Screenshot configurations of R4

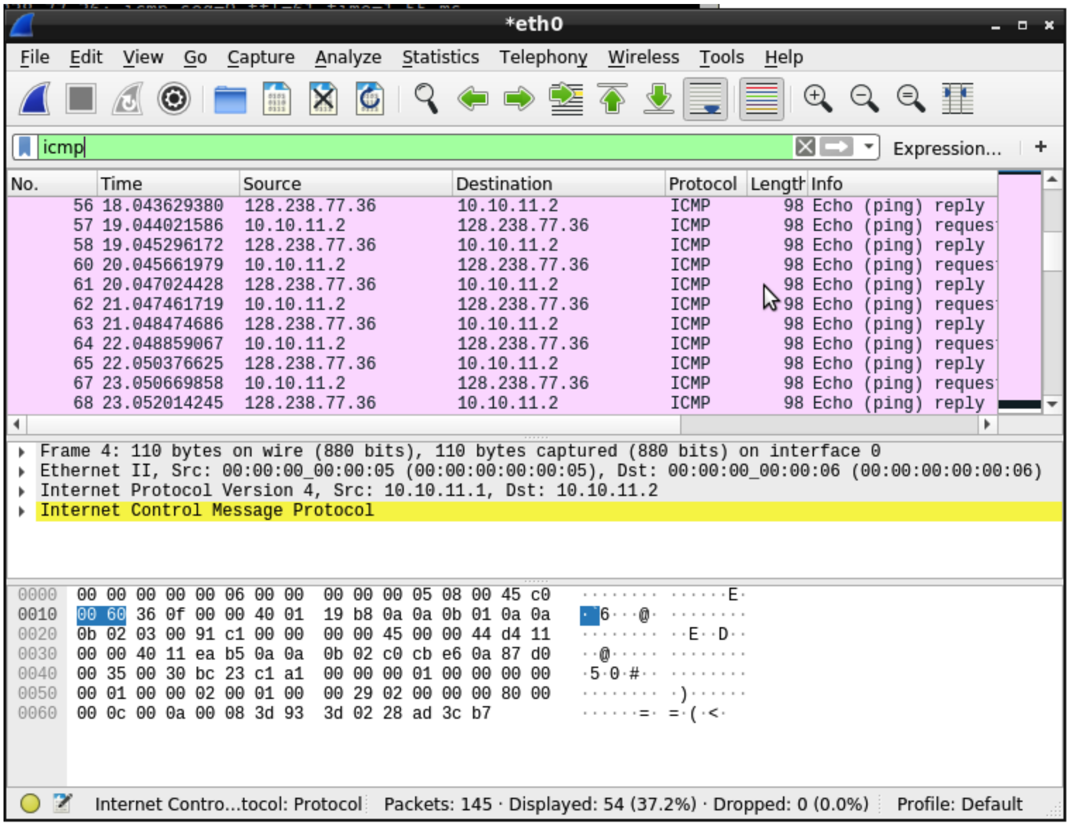
2. ICMP results from R3 to R1.

Ping R1 from R3

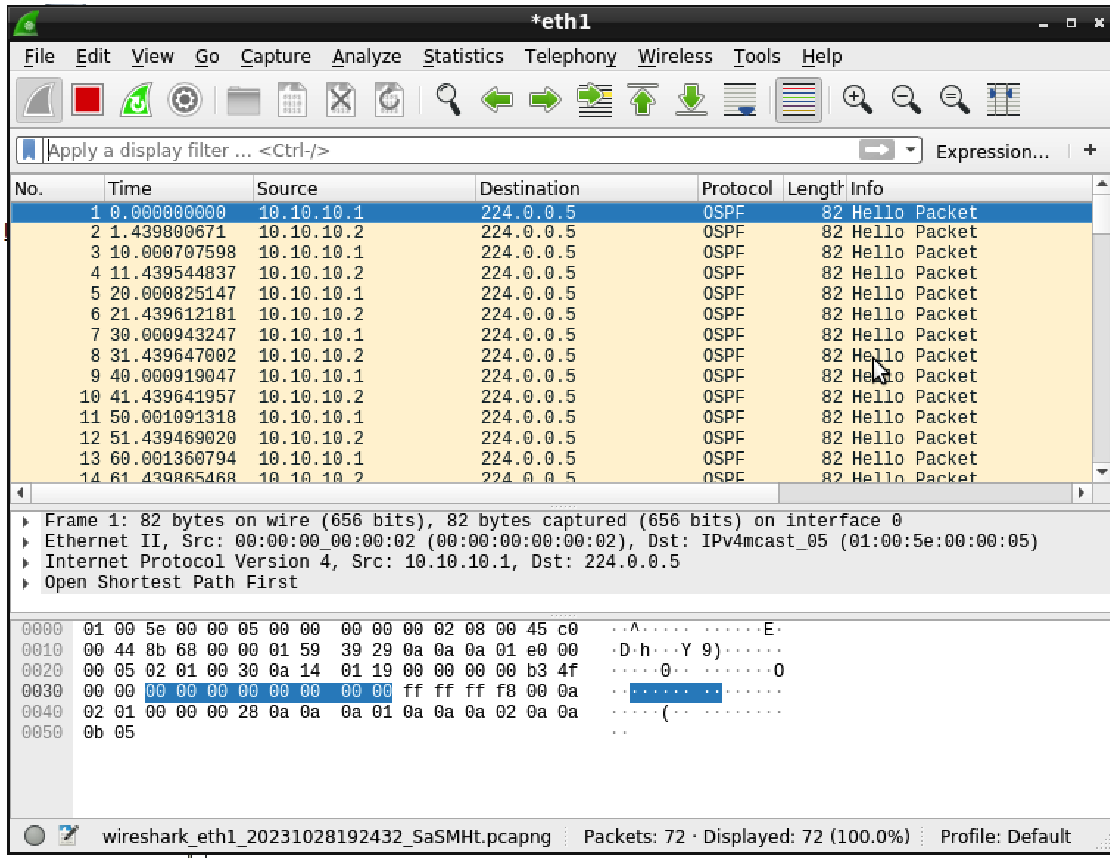


Wireshark on R3




*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression...
No. Time Source Destination Protocol Length Info
56 18.043629380 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
57 19.044021586 10.10.11.2 128.238.77.36 ICMP 98 Echo (ping) reques
58 19.045296172 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
60 20.045661979 10.10.11.2 128.238.77.36 ICMP 98 Echo (ping) reques
61 20.047024428 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
62 21.047461719 10.10.11.2 128.238.77.36 ICMP 98 Echo (ping) reques
63 21.048474686 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
64 22.048859067 10.10.11.2 128.238.77.36 ICMP 98 Echo (ping) reques
65 22.050376625 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
67 23.050669858 10.10.11.2 128.238.77.36 ICMP 98 Echo (ping) reques
68 23.052014245 128.238.77.36 10.10.11.2 ICMP 98 Echo (ping) reply
Frame 4: 110 bytes on wire (880 bits), 110 bytes captured (880 bits) on interface 0
Ethernet II, Src: 00:00:00_00:00:05 (00:00:00:00:00:05), Dst: 00:00:00_00:00:06 (00:00:00:00:00:06)
Internet Protocol Version 4, Src: 10.10.11.1, Dst: 10.10.11.2
Internet Control Message Protocol
0000 00 00 00 00 00 06 00 00 00 00 00 05 08 00 45 c0
0010 00 60 36 0f 00 00 40 01 19 b8 0a 0a 0b 01 0a 0a
0020 0b 02 03 00 91 c1 00 00 00 00 45 00 00 44 d4 11
0030 00 00 40 11 ea b5 0a 0a 0b 02 c0 cb e6 0a 87 d0
0040 00 35 00 30 bc 23 c1 a1 00 00 00 01 00 00 00 00
0050 00 01 00 00 02 00 01 00 00 29 02 00 00 00 80 00
0060 00 0c 00 0a 00 08 3d 93 3d 02 28 ad 3c b7
Internet Contro...tocol: Protocol
Packets: 145 · Displayed: 54 (37.2%) · Dropped: 0 (0.0%)
Profile: Default

3. Wireshark screenshots on R1.



Wireshark screenshot on R1

4. Screenshots depicting successful ping requests to the SFTP server (128.238.77.36) from R1, R2, R3, and R4.

```
student@CN-R1:~$ ping 128.238.77.36
PING 128.238.77.36 (128.238.77.36) 56(84) bytes of data.
64 bytes from 128.238.77.36: icmp_seq=1 ttl=63 time=4.23 ms
64 bytes from 128.238.77.36: icmp_seq=2 ttl=63 time=0.569 ms
64 bytes from 128.238.77.36: icmp_seq=3 ttl=63 time=0.655 ms
64 bytes from 128.238.77.36: icmp_seq=4 ttl=63 time=0.552 ms
64 bytes from 128.238.77.36: icmp_seq=5 ttl=63 time=0.646 ms
64 bytes from 128.238.77.36: icmp_seq=6 ttl=63 time=0.607 ms
```

Screenshot of ping SFTP server from R1

```
student@CN-R2:~$ ping 128.238.77.36
PING 128.238.77.36 (128.238.77.36) 56(84) bytes of data.
64 bytes from 128.238.77.36: icmp_seq=1 ttl=62 time=1.17 ms
64 bytes from 128.238.77.36: icmp_seq=2 ttl=62 time=1.02 ms
64 bytes from 128.238.77.36: icmp_seq=3 ttl=62 time=1.04 ms
64 bytes from 128.238.77.36: icmp_seq=4 ttl=62 time=0.905 ms
64 bytes from 128.238.77.36: icmp_seq=5 ttl=62 time=7.86 ms
64 bytes from 128.238.77.36: icmp_seq=6 ttl=62 time=1.07 ms
```

Screenshot of ping SFTP server from R2

```
student@CN-R3:~$ ping 128.238.77.36
PING 128.238.77.36 (128.238.77.36) 56(84) bytes of data.
64 bytes from 128.238.77.36: icmp_seq=1 ttl=61 time=1.31 ms
64 bytes from 128.238.77.36: icmp_seq=2 ttl=61 time=8.00 ms
64 bytes from 128.238.77.36: icmp_seq=3 ttl=61 time=1.40 ms
64 bytes from 128.238.77.36: icmp_seq=4 ttl=61 time=1.36 ms
64 bytes from 128.238.77.36: icmp_seq=5 ttl=61 time=1.40 ms
64 bytes from 128.238.77.36: icmp_seq=6 ttl=61 time=1.36 ms
64 bytes from 128.238.77.36: icmp_seq=7 ttl=61 time=1.37 ms
64 bytes from 128.238.77.36: icmp_seq=8 ttl=61 time=1.10 ms
64 bytes from 128.238.77.36: icmp_seq=9 ttl=61 time=1.31 ms
```

Screenshot of ping SFTP server from R3

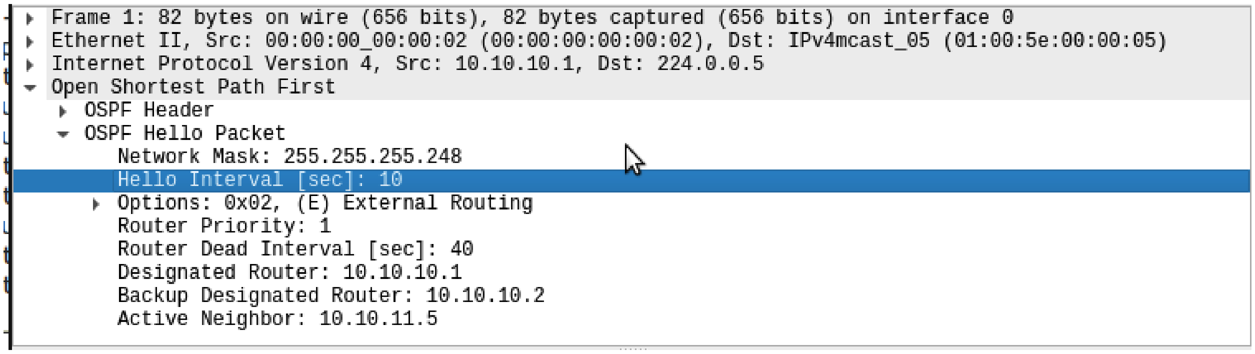
```
student@CN-R4:~$ ping 128.238.77.36
PING 128.238.77.36 (128.238.77.36) 56(84) bytes of data.
64 bytes from 128.238.77.36: icmp_seq=1 ttl=61 time=1.28 ms
64 bytes from 128.238.77.36: icmp_seq=2 ttl=61 time=1.60 ms
64 bytes from 128.238.77.36: icmp_seq=3 ttl=61 time=1.55 ms
64 bytes from 128.238.77.36: icmp_seq=4 ttl=61 time=1.48 ms
64 bytes from 128.238.77.36: icmp_seq=5 ttl=61 time=1.31 ms
64 bytes from 128.238.77.36: icmp_seq=6 ttl=61 time=1.36 ms
64 bytes from 128.238.77.36: icmp_seq=7 ttl=61 time=1.42 ms
```

Screenshot of ping SFTP server from R4

5. Questions.

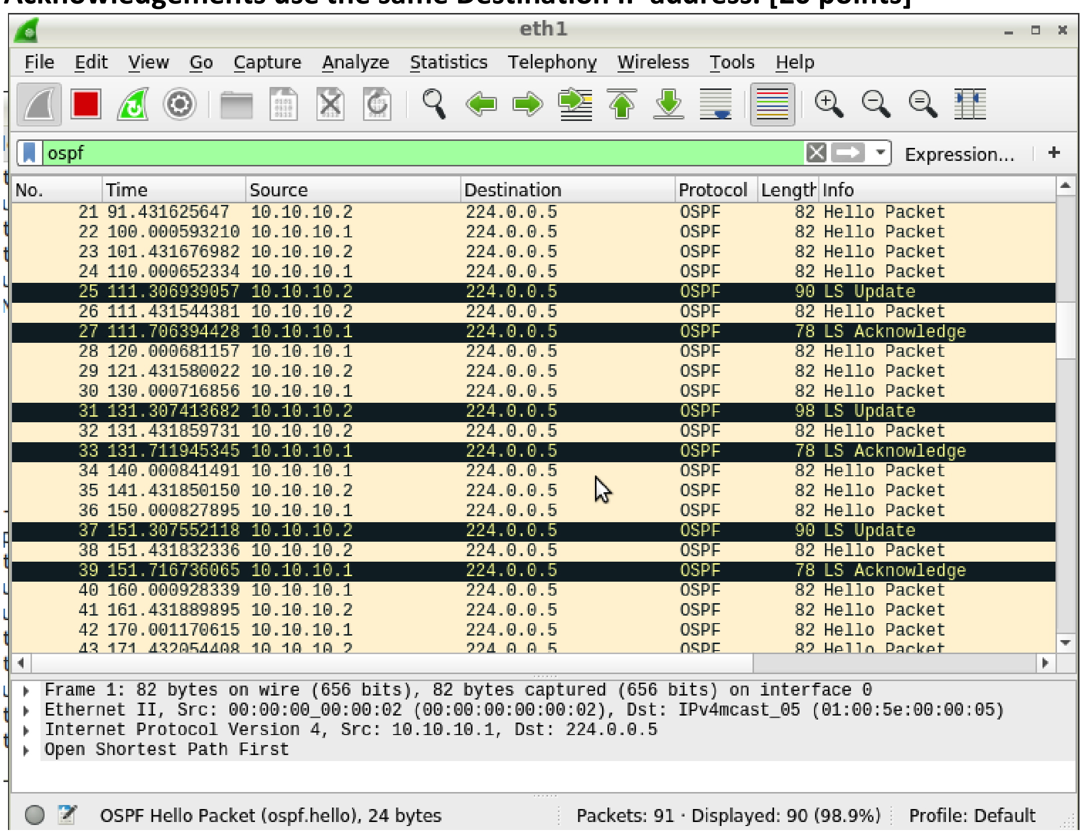
a. Power on all routers and run Wireshark on R1. Apply a filter for OSPF, and look at the Hello Packets. How frequently are these packets sent, and why must they be sent periodically? [10 points]

The Router sends these OSPF HELLO packets every 10 seconds periodically according to the screenshot below.



This is done to maintain neighbor relationships and ensure network stability. First, it helps routers discover and establish neighbor relationships, facilitating the exchange of routing information. Second, it allows routers to monitor the health and status of their neighbors. If a router doesn't receive Hello packets from a neighbor within a specified time, it considers the neighbor as unreachable, triggering a network convergence process to update routing tables and maintain network integrity. This periodic communication is vital for OSPF's dynamic and efficient routing operation.

b. Continue running Wireshark and turn off R4. You should now see new OSPF packet types captured on R1. Explain why Hello, Link State Update, and Link State Acknowledgements use the same Destination IP address. [20 points]



In the screenshot, the destination ip address of Hello packets, Link State Update packets, and Link State Acknowledgement packets in OSPF are same, 224.0.0.5.

Here are reasons why using the same destination IP address:
1. Group communication: OSPF routers use same destination IP address (multicast address) to efficiently distribute OSPF messages to multiple routers in a network segment.
2. Reduction of configuration complexity: Using a consistent multicast address for these different types of OSPF packets simplifies router configuration. Routers don't need to be explicitly configured to send different types of OSPF messages to different destinations. They can simply use the common multicast address for OSPF communication.
3. Minimizing network traffic: By sending Hello packets, Link State Update packets, and Link State Acknowledgements to the same multicast address, routers can efficiently exchange OSPF information without the need for separate routing entries or additional address assignments.

c. **Continue Based on the above steps, explain why we do not see DB Descriptions and LS Requests on R1. Is there a situation in which we get all OSPF packet types on R1? [20 points]**
DB Descriptions and LS Requests packets are primarily exchanged during initial neighbor setup and when there are significant topology changes. However, in this case, R1 already has a fully established OSPF network, and the initial synchronization has already occurred. So, R1 no longer needs to request or describe the database because it already has a complete and up-to-date database. We do not see DB Descriptions and LS Requests on R1 because R1 and R2 lie on the same area(area 0) and R4 router from area 1 was shut off.

Reference

[1] https://www.juniper.net/documentation/us/en/software/junos/ospf/topics/topic-map/ospf-overview.html
[2] chatgpt