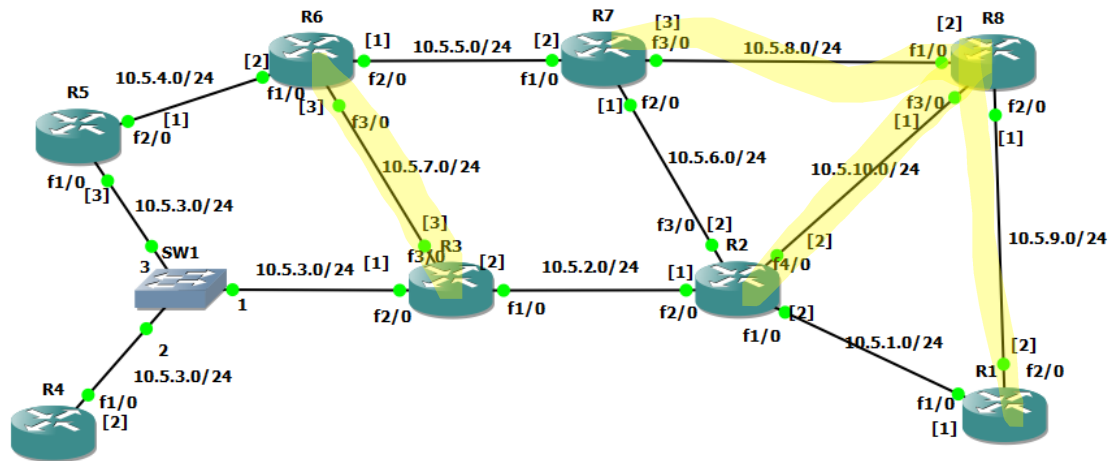


LAB 7 - Intra AS Routing: RIP

Shir Moshe - 318492667

Nadav Biran - 316468834

1.Topology 7 Configuration – Topology 7



R8

<pre>configure terminal interface FastEthernet1/0 ip address 10.5.8.2 255.255.255.0 no shutdown end</pre>	<pre>configure terminal interface FastEthernet2/0 ip address 10.5.9.1 255.255.255.0 no shutdown end</pre>	<pre>configure terminal interface FastEthernet3/0 ip address 10.5.10.1 255.255.255.0 no shutdown end</pre>
---	---	--

R1

<pre>configure terminal interface FastEthernet2/0 ip address 10.5.9.2 255.255.255.0 no shutdown end</pre>

R2

<pre>configure terminal interface FastEthernet4/0 ip address 10.5.10.2 255.255.255.0 no shutdown end</pre>
--

R7

```
configure terminal
interface FastEthernet3/0
ip address 10.5.8.1 255.255.255.0
no shutdown
end
```

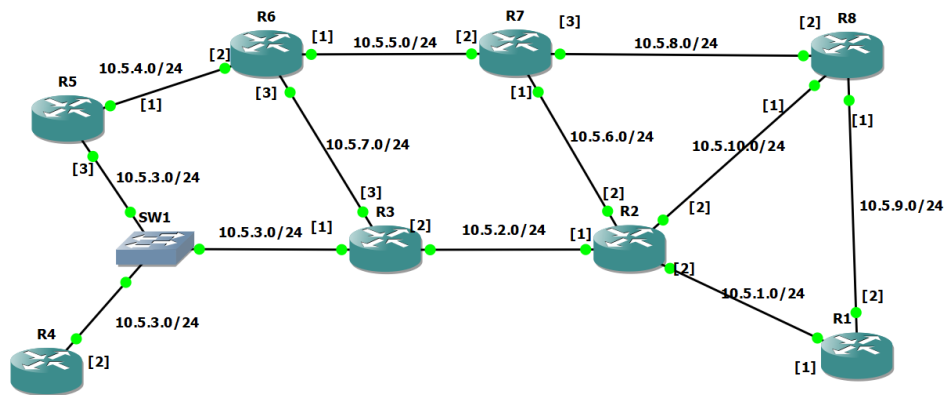
R6

```
configure terminal
interface FastEthernet3/0
ip address 10.5.7.1 255.255.255.0
no shutdown
end
```

R3

```
configure terminal
interface FastEthernet3/0
ip address 10.5.7.2 255.255.255.0
no shutdown
end
```

2.Configuring RIP version 2 on Cisco Routers – Topology 8



2.1

2.2

```
R1#show ip route
```

```
R1#show ip rip database
```

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

    10.0.0.0/24 is subnetted, 2 subnets
C       10.5.9.0 is directly connected, FastEthernet2/0
C       10.5.1.0 is directly connected, FastEthernet1/0
C       192.168.1.0/24 is directly connected, FastEthernet0/0
S       192.168.2.0/24 [1/0] via 10.5.1.2
R1#show ip rip database
R1#
```

2.4

לכתוב את זה לכל הראוטרים (לקנפג אותם שישתמשו בRIP , וישחקו עם כל מי שב 10.5.0.0 netork)

```
configure terminal
router rip
version 2
no auto-summary
network 10.5.0.0
end
```

2.6

R1: show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 10 subnets
R    10.5.10.0 [120/1] via 10.5.9.1, 00:00:24, FastEthernet2/0
     [120/1] via 10.5.1.2, 00:00:24, FastEthernet1/0
C    10.5.9.0 is directly connected, FastEthernet2/0
R    10.5.8.0 [120/1] via 10.5.9.1, 00:00:24, FastEthernet2/0
R    10.5.7.0 [120/2] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.6.0 [120/1] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.5.0 [120/2] via 10.5.9.1, 00:00:22, FastEthernet2/0
     [120/2] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.4.0 [120/3] via 10.5.9.1, 00:00:24, FastEthernet2/0
     [120/3] via 10.5.1.2, 00:00:26, FastEthernet1/0
R    10.5.3.0 [120/2] via 10.5.1.2, 00:00:26, FastEthernet1/0
R    10.5.2.0 [120/1] via 10.5.1.2, 00:00:26, FastEthernet1/0
C    10.5.1.0 is directly connected, FastEthernet1/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.2.0/24 [1/0] via 10.5.1.2
```

show ip rip database

```
R1#show ip rip database
10.0.0.0/8      auto-summary
10.5.1.0/24     directly connected, FastEthernet1/0
10.5.2.0/24     [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.3.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.4.0/24     [3] via 10.5.9.1, 00:00:29, FastEthernet2/0
                 [3] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.5.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
                 [2] via 10.5.9.1, 00:00:29, FastEthernet2/0
10.5.6.0/24     [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.7.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.8.0/24     [1] via 10.5.9.1, 00:00:29, FastEthernet2/0
10.5.9.0/24     directly connected, FastEthernet2/0
10.5.10.0/24    [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
                 [1] via 10.5.9.1, 00:00:29, FastEthernet2/0
R1#
```

R2: show ip route

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 10 subnets
C    10.5.10.0 is directly connected, FastEthernet4/0
R    10.5.9.0 [120/1] via 10.5.10.1, 00:00:55, FastEthernet4/0
     [120/1] via 10.5.1.1, 00:00:02, FastEthernet1/0
R    10.5.8.0 [120/1] via 10.5.10.1, 00:00:55, FastEthernet4/0
     [120/1] via 10.5.6.1, 00:00:00, FastEthernet3/0
R    10.5.7.0 [120/1] via 10.5.2.2, 00:00:08, FastEthernet2/0
C    10.5.6.0 is directly connected, FastEthernet3/0
R    10.5.5.0 [120/1] via 10.5.6.1, 00:00:02, FastEthernet3/0
R    10.5.4.0 [120/2] via 10.5.6.1, 00:00:02, FastEthernet3/0
     [120/2] via 10.5.2.2, 00:00:11, FastEthernet2/0
R    10.5.3.0 [120/1] via 10.5.2.2, 00:00:11, FastEthernet2/0
C    10.5.2.0 is directly connected, FastEthernet2/0
C    10.5.1.0 is directly connected, FastEthernet1/0
S    192.168.1.0/24 [1/0] via 10.5.1.1
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

```
R2#show ip rip database
10.0.0.0/8      auto-summary
10.5.1.0/24     directly connected, FastEthernet1/0
10.5.2.0/24     directly connected, FastEthernet2/0
10.5.3.0/24     [1] via 10.5.2.2, 00:00:19, FastEthernet2/0
10.5.4.0/24     [2] via 10.5.6.1, 00:00:10, FastEthernet3/0
                 [2] via 10.5.2.2, 00:00:19, FastEthernet2/0
10.5.5.0/24     [1] via 10.5.6.1, 00:00:10, FastEthernet3/0
10.5.6.0/24     directly connected, FastEthernet3/0
10.5.7.0/24     [1] via 10.5.2.2, 00:00:19, FastEthernet2/0
10.5.8.0/24     [1] via 10.5.6.1, 00:00:10, FastEthernet3/0
                 [1] via 10.5.10.1, 00:00:08, FastEthernet4/0
10.5.9.0/24     [1] via 10.5.1.1, 00:00:13, FastEthernet1/0
                 [1] via 10.5.10.1, 00:00:08, FastEthernet4/0
10.5.10.0/24    directly connected, FastEthernet4/0
R2#
```

show ip rip database

2.7

R2#show ip protocols

```
R2#show ip protocols
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 22 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
    Default version control: send version 2, receive version 2
  Interface                Send Recv Triggered RIP Key-chain
  FastEthernet1/0           2     2
  FastEthernet2/0           2     2
  FastEthernet3/0           2     2
  FastEthernet4/0           2     2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
  10.5.10.1          120           00:00:12
  10.5.6.1            120           00:00:10
  10.5.2.2            120           00:00:16
  10.5.1.1            120           00:00:09
  Distance: (default is 120)
```

2.11

R1#configure terminal

R1(config)#router rip - enable RIP.

R1(config-router)#version 2 - change the version of the protocol to 2

R1(config-router)#no auto-summary – turn off automatic summarizations

(config-router)#network 10.5.0.0 – advertise only this subnet

R1(config-router)#end

2.12

1. נעשה את הפקודה `show ip rip database` ואם נראה שם את הרשת 10.5.0.0 אז הכל התכנס

2. נשלח פינג ונראה אם הכל מחובר.

2.13

5 10.727650	10.5.1.1	224.0.0.9	RIPv2	66 Response
6 19.851555	c4:a2:5a:h4:a0:1a	c4:a2:5a:h4:a0:1a	ICMP	60 Reply

איי פי של היעד – 224.0.0.9

רק מי שרוצה לשחק מקבל את ההודעות ולא סתם כולם.

2.14

כן, אפשר לראות ב Wireshark שהתוכן בפקטות מתעדכן.

5 10.727650	10.5.1.1	224.0.0.9	RIPv2	66 Response
9 28.849642	10.5.1.2	224.0.0.9	RIPv2	66 Request

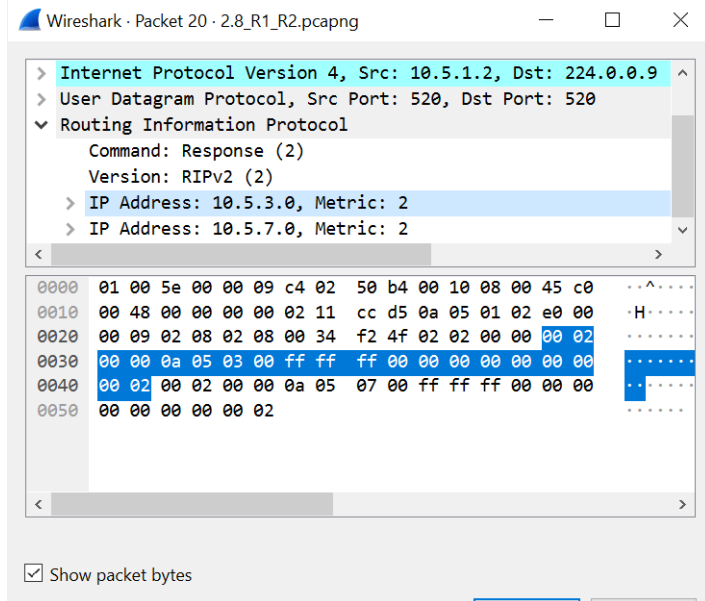
2.15

שני סוגים :

1. request - מבקש מהראוטר השכן את התוכן של הטבלת ניתוב שלו
2. response – הטבלה שכל ראוטר שולח (של עצמו) מפרסם לשכנים. נשים לב שזה מגיע כל כמה זמן ולא רק כשיש request.

2.16

כל שורה שנשלחת היא 14 בייטים.



2.17

1. Network address - היעד שאליו מגיעים

2. Subnet mask - של היעד

3. Next hop - דרך מי הוא מגיע לשם

4. Metric - כמה הופים יש בדרך

5. Timer - הזמן לפני שהראוטר יוצא מהטבלה.

- IP Address: 10.5.3.0, Metric: 2
 - Address Family: IP (2)
 - Route Tag: 0
 - IP Address: 10.5.3.0
 - Netmask: 255.255.255.0
 - Next Hop: 0.0.0.0
 - Metric: 2

1. Routing Protocol is "rip" – איזה פרוטוקול ממומש פה.
2. Outgoing update filter list for all interfaces is not set flow control – אין חוקים לשליחת הודעות של
3. Incoming update filter list for all interfaces is not set flow control – אין חוקים לשליחת הודעות של
4. Sending updates every 30 seconds, next due in 20 seconds – כל 30 שניות תשלח הודעת עדכון, הבאה תשלח בעוד 20 שניות.
5. Invalid after 180 seconds, hold down 180, flushed after 240 – אם 2R לא יקבל הודעת עדכון מ1R תוך 180 שניות – כל המסלולים מ1R יהפכו ללא שמישים. אם אחרי 240 שניות עדיין לא קיבלנו עוד עדכון – נמחק מהטבלה את המסלולים שמגיעים מ1R.
6. Default version control – שולח ומקבל בגרסה 2 של RIP.
7. FastEthernet1/0 2 2 – הפורט מוציא הודעות בגרסה 2 ומקבל הודעות של גרסה 2.
8. Automatic network summarization is not effect (לאחד שורות עם פרפיקס דומה) – אין פה קיצור של הטבלת ניתוב (נגיד
9. Maximum path: 4 – אפשר להחזיק עד 4 מסלולים לאותו יעד.
10. Routing for Networks: 10.0.0.0 – מספר איזה רשת משתתפת בפרסומים של RIP.
11. Routing Information Sources: – נותן מידע על כל כתובת איי פי והמחיר של המרחק שלה. בנוסף מתי היא התעדכנה לאחרונה.

```

R2#show ip protocols
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 22 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface        Send  Recv  Triggered RIP  Key-chain
  FastEthernet1/0    2     2
  FastEthernet2/0    2     2
  FastEthernet3/0    2     2
  FastEthernet4/0    2     2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
  10.5.10.1          120           00:00:12
  10.5.6.1           120           00:00:10
  10.5.2.2           120           00:00:16
  10.5.1.1           120           00:00:09
  Distance: (default is 120)

```


2.19

שוני בין שתי הטבלאות- טבלת ריפ וטבלת ניתוב :

הטבלה של RIP מתמלאת ע"י עדכון של הפרוטוקול של RIP בלבד, בכל מעם שמישהו שמשתתף בRIP מתעדכן הטבלה מתעדכנת.

לעומת זאת, טבלת הניתוב של הראוטר מתעדכנת ע"י כל מיני פרוטוקולים (כולל ניתוב סטטי למשל). ניתן לראות בתחילת כל שורה את סוג הפרוטוקול שאחראי על הבאת אותו מידע. (למשל R זה RIP).

נשים לב שאצלנו לדוגמא, בטבלת ניתוב כתובים המחשבים שחיברנו בצורה סטטית במעבדה הקודמת, ואילו בטבלת RIP הם לא מופיעים (כי הם לא שיקחו במשחק).

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 10 subnets
R    10.5.10.0 [120/1] via 10.5.9.1, 00:00:24, FastEthernet2/0
      [120/1] via 10.5.1.2, 00:00:24, FastEthernet1/0
C    10.5.9.0 is directly connected, FastEthernet2/0
R    10.5.8.0 [120/1] via 10.5.9.1, 00:00:24, FastEthernet2/0
R    10.5.7.0 [120/2] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.6.0 [120/1] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.5.0 [120/2] via 10.5.9.1, 00:00:22, FastEthernet2/0
      [120/2] via 10.5.1.2, 00:00:24, FastEthernet1/0
R    10.5.4.0 [120/3] via 10.5.9.1, 00:00:24, FastEthernet2/0
      [120/3] via 10.5.1.2, 00:00:26, FastEthernet1/0
R    10.5.3.0 [120/2] via 10.5.1.2, 00:00:26, FastEthernet1/0
R    10.5.2.0 [120/1] via 10.5.1.2, 00:00:26, FastEthernet1/0
C    10.5.1.0 is directly connected, FastEthernet1/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.2.0/24 [1/0] via 10.5.1.2
```

```
R1#show ip rip database
10.0.0.0/8      auto-summary
10.5.1.0/24     directly connected, FastEthernet1/0
10.5.2.0/24     [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.3.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.4.0/24     [3] via 10.5.9.1, 00:00:29, FastEthernet2/0
                  [3] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.5.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
                  [2] via 10.5.9.1, 00:00:29, FastEthernet2/0
10.5.6.0/24     [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.7.0/24     [2] via 10.5.1.2, 00:00:05, FastEthernet1/0
10.5.8.0/24     [1] via 10.5.9.1, 00:00:29, FastEthernet2/0
10.5.9.0/24     directly connected, FastEthernet2/0
10.5.10.0/24    [1] via 10.5.1.2, 00:00:05, FastEthernet1/0
                  [1] via 10.5.9.1, 00:00:29, FastEthernet2/0
R1#
```

2.20

הפעלנו wireshark, ואז קינפגנו את כל הראוטרם להיות RIP, חיכינו שהרשת תתכנס ושמרנו את הפיק אפ:

לבדוק עם בירן אם אצלו יש פקטות כמו שצריך, צריך למצוא 2 פקטות אחת של R1 אחת של R2 (אחרי ההתכנסות – נראה שיש להן את כל המידע על כל הראוטרם), למצוא אם יש 2 IP מסויים שנמצא בשתי הטבלאות (של R1 ושל R2) ואז לראות אם אותו IP מופיע גם בטבלאות ניתוב.

נסביר את הטבלת ניתוב של R2:

דוגמא:

R 10.5.9.0 [120/1] via 10.5.10.1, 00:00:55, FastEthernet4/0

[120/1] via 10.5.1.1, 00:00:02, FastEthernet1/0

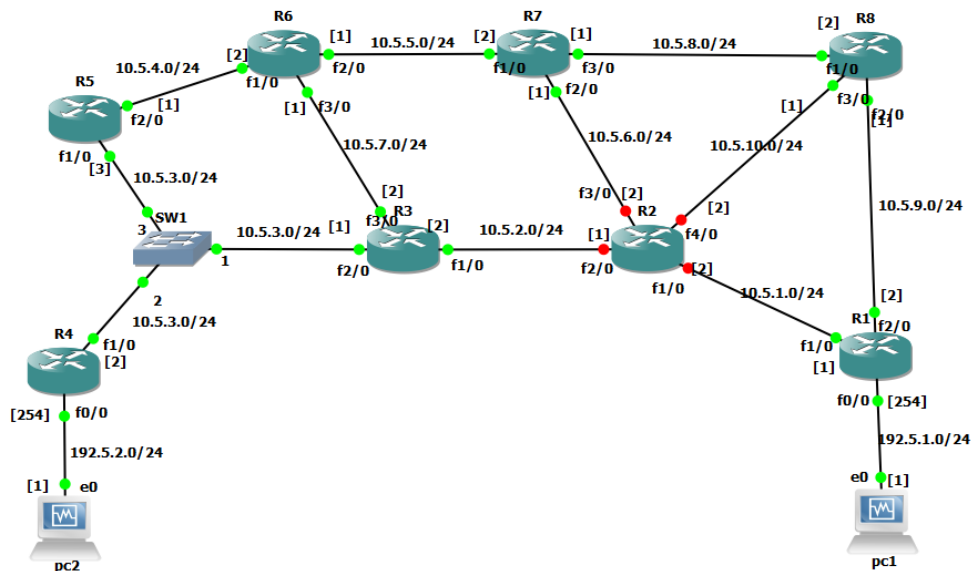
- C/R/S : מציינים באיזה פרוטוקול השתמש הראוטר כשקיבל את המידע אודות אותה רשת.
- מחובר ישירות (c), ע"י RIP (r) או הוכנס סטטי (s).
- 10.5.9.0 : אותה כתובת IP שלמדנו את הניתוב אליה.
- [120/1]:120 מצוין את רמת האמינות של הדרך, כמה שיותר נמוך יותר טוב.
- 1 מצוין כמה הופים בדרך, (10.5.9.0) זה בין 8R ל1R אפשר ושניהם מחוברים ישירות לכן אפשר להגיע אליה דרך הופ אחד). הראוטר לוקח בחשבון את שניהם כשהוא בוחר את הדרך העדיפה.
- Directly/via: איך היא מחוברת, באופן ישיר או דרך אחד IP אחרת.
- 00:00:55 – כמה זמן עבר מאז העדכון האחרון של השורה הזאת.
- FastEthernet4/0 – דרך איזה פורט לצאת.
- ** אפשר לראות שיש בטבלה דרך חלופית באותו משקל.

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 10 subnets
C    10.5.10.0 is directly connected, FastEthernet4/0
R    10.5.9.0 [120/1] via 10.5.10.1, 00:00:55, FastEthernet4/0
      [120/1] via 10.5.1.1, 00:00:02, FastEthernet1/0
R    10.5.8.0 [120/1] via 10.5.10.1, 00:00:55, FastEthernet4/0
      [120/1] via 10.5.6.1, 00:00:00, FastEthernet3/0
R    10.5.7.0 [120/1] via 10.5.2.2, 00:00:08, FastEthernet2/0
C    10.5.6.0 is directly connected, FastEthernet3/0
R    10.5.5.0 [120/1] via 10.5.6.1, 00:00:02, FastEthernet3/0
R    10.5.4.0 [120/2] via 10.5.6.1, 00:00:02, FastEthernet3/0
      [120/2] via 10.5.2.2, 00:00:11, FastEthernet2/0
R    10.5.3.0 [120/1] via 10.5.2.2, 00:00:11, FastEthernet2/0
C    10.5.2.0 is directly connected, FastEthernet2/0
C    10.5.1.0 is directly connected, FastEthernet1/0
S    192.168.1.0/24 [1/0] via 10.5.1.1
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

3. Measuring RIP version 2 Convergence Time – Topology 8.1



3.3

**** קודם למחוק את הקינפוג של המחשבים שנשאר מלפני (ממעבדה קודמת):**

למחוק רשת שמחוברת ישירות- נוציא את הפורט משימוש:

configure terminal

interface FastEthernet0/0

shutdown

no ip route 192.168.1.0 255.255.255.0

exit

3.4

נקבע IP סטטי חדש לשני המחשבים:

Pc1: <i>nano /etc/network/interfaces</i> <i>auto eth0</i> <i>iface eth0 inet static</i> <i>address 192.5.1.1</i> <i>netmask 255.255.255.0</i> ctrl+x y enter reboot <i>route add default gw 192.5.1.254</i>	Pc2: <i>nano /etc/network/interfaces</i> <i>auto eth0</i> <i>iface eth0 inet static</i> <i>address 192.5.2.1</i> <i>netmask 255.255.255.0</i> ctrl+x y enter reboot <i>route add default gw 192.5.2.254</i>
--	--

3.5

R1 (configure the network 192.5.1.0 on rip – passive, configure the port with new ip)

```
configure terminal
router rip
network 192.5.1.0
passive-interface FastEthernet 0/0
exit
interface FastEthernet 0/0
ip address 192.5.1.254 255.255.255.0
no shutdown
end
```

3.6

Pc1 (check traceroute from pc1 to R4- port 1/0)

```
traceroute 10.5.3.2
```

```
root@pc1:~# traceroute 10.5.3.2
traceroute to 10.5.3.2 (10.5.3.2), 30 hops max, 60 byte packets
 1 192.5.1.254 (192.5.1.254) 12.507 ms 22.682 ms 33.467 ms
 2 10.5.9.1 (10.5.9.1) 44.182 ms 54.734 ms 65.485 ms
 3 10.5.8.1 (10.5.8.1) 76.316 ms 86.986 ms 97.476 ms
 4 10.5.5.1 (10.5.5.1) 108.182 ms 118.913 ms 129.428 ms
 5 10.5.4.1 (10.5.4.1) 140.036 ms 150.768 ms 161.419 ms
 6 10.5.3.2 (10.5.3.2) 171.576 ms 167.504 ms 169.612 ms
root@pc1:~#
```

3.7

R1 (routing table)

```
show ip route
```

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 10 subnets
R    10.5.10.0 [120/1] via 10.5.9.1, 00:00:12, FastEthernet2/0
C    10.5.9.0 is directly connected, FastEthernet2/0
R    10.5.8.0 [120/1] via 10.5.9.1, 00:00:12, FastEthernet2/0
R    10.5.7.0 [120/3] via 10.5.9.1, 00:00:12, FastEthernet2/0
R    10.5.6.0 [120/2] via 10.5.9.1, 00:00:12, FastEthernet2/0
R    10.5.5.0 [120/2] via 10.5.9.1, 00:00:12, FastEthernet2/0
R    10.5.4.0 [120/3] via 10.5.9.1, 00:00:12, FastEthernet2/0
R    10.5.3.0 [120/4] via 10.5.9.1, 00:00:14, FastEthernet2/0
R    10.5.2.0 [120/4] via 10.5.9.1, 00:00:14, FastEthernet2/0
C    10.5.1.0 is directly connected, FastEthernet1/0
C    192.5.1.0/24 is directly connected, FastEthernet0/0
R1#
```

3.8 Start
Wireshark on link:
R1-R8, R8- R7, R4- R5
(via switch)

3.8 Wireshark -

> View -> Time Display Format -> Time Of Day.

3.10

pc1 (ping from PC1 to PC2)

```
ping 192.5.2.1
```

3.11

R4 (configure the network 192.5.2.0 on rip – passive, configure the port with new ip)

```
configure terminal
router rip
network 192.5.2.0
passive-interface FastEthernet 0/0
exit
interface FastEthernet 0/0
ip address 192.5.2.254 255.255.255.0
no shutdown
end
```

3.12 Wait for the network convergence. The ping should start working properly.

3.13 Stop the ping command using Ctrl + C.

3.14

Stop capture and save the wireshark pcaps to files.

3.15

Look again at the routing table of R1 and save the output.

R1 (routing table)

```
show ip route
```

```
10.0.0.0/24 is subnetted, 10 subnets
R    10.5.10.0 [120/1] via 10.5.9.1, 00:00:10, FastEthernet2/0
C    10.5.9.0 is directly connected, FastEthernet2/0
R    10.5.8.0 [120/1] via 10.5.9.1, 00:00:10, FastEthernet2/0
R    10.5.7.0 [120/3] via 10.5.9.1, 00:00:10, FastEthernet2/0
R    10.5.6.0 [120/2] via 10.5.9.1, 00:00:10, FastEthernet2/0
R    10.5.5.0 [120/2] via 10.5.9.1, 00:00:10, FastEthernet2/0
R    10.5.4.0 [120/3] via 10.5.9.1, 00:00:10, FastEthernet2/0
R    10.5.3.0 [120/4] via 10.5.9.1, 00:00:11, FastEthernet2/0
R    10.5.2.0 [120/4] via 10.5.9.1, 00:00:11, FastEthernet2/0
C    10.5.1.0 is directly connected, FastEthernet1/0
C    192.5.1.0/24 is directly connected, FastEthernet0/0
R    192.5.2.0/24 [120/5] via 10.5.9.1, 00:00:11, FastEthernet2/0
R1#
```

```
From 192.5.1.254 icmp_seq=26 Destination Host Unreachable
From 192.5.1.254 icmp_seq=27 Destination Host Unreachable
From 192.5.1.254 icmp_seq=28 Destination Host Unreachable
From 192.5.1.254 icmp_seq=29 Destination Host Unreachable
From 192.5.1.254 icmp_seq=30 Destination Host Unreachable
From 192.5.1.254 icmp_seq=31 Destination Host Unreachable
From 192.5.1.254 icmp_seq=32 Destination Host Unreachable
From 192.5.1.254 icmp_seq=33 Destination Host Unreachable
64 bytes from 192.5.1.254: icmp_seq=35 ttl=58 time=94.2 ms
64 bytes from 192.5.1.254: icmp_seq=36 ttl=58 time=88.8 ms
64 bytes from 192.5.1.254: icmp_seq=37 ttl=58 time=90.6 ms
64 bytes from 192.5.1.254: icmp_seq=38 ttl=58 time=94.8 ms
64 bytes from 192.5.1.254: icmp_seq=39 ttl=58 time=89.8 ms
^C
--- 192.5.2.1 ping statistics ---
40 packets transmitted, 5 received, +33 errors, 87% packet loss, time 39063ms
rtt min/avg/max/mdev = 88.860/91.688/94.819/2.416 ms
root@pc1:~#
```

3.17

כששלחנו את הפינג בהתחלה מֶסֶד 1 לֶסֶד 2 הוועדה הגיעה בדיפולט גט וואיי לֶסֶד 1. 1R זרק אותה, כי אין לֶסֶד 1 בטבלת ניתוב את הרשת 192.5.2.0, זאת משום שֶסֶד 4 אפילו לא קונפג ולא יודע שמאחוריו יש את הרשת 192.5.2.0.

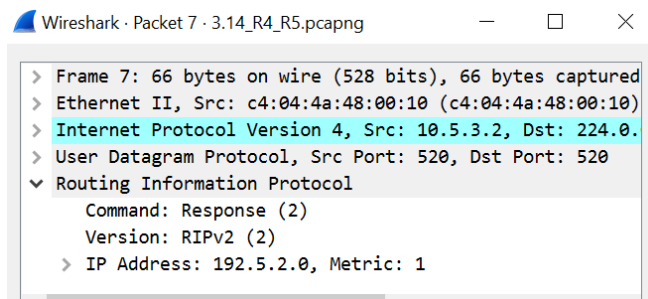
3.18

לאינטרפסיים שקינפגנו כֶסֶד passive מחוברים רק מחשבים- ללא ראוטרים. בנוסף מדובר ברשת שונה שלא כלולה במשחק של הֶסֶד RIP שהגדרנו (10.5.0.0), לכן לשלוח למחשבים את ההודעות של הֶסֶד RIP זה סתם בזבוז של רוחב פֶס. בנוסף מדובר בבעיית בטיחות אם לא רוצים שהמחשב בקצה ידע על כל המסלולים ומי הולך לאן.

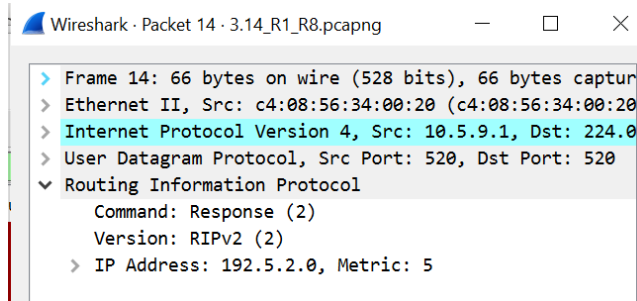
3.19

- Link between R4 – R5 (via switch): packet #7, time: 18:51:51

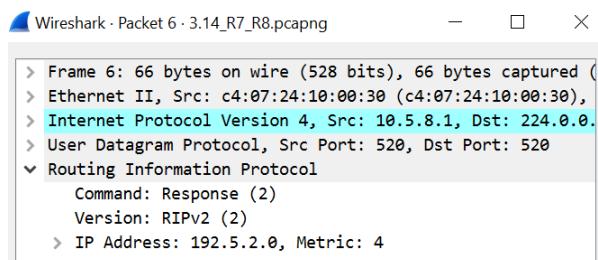
4R בפורט f1/0 מפרסם לראשונה את הרשת בשעה 18:51:51, רואים שהוא מפרסם עם מרחק = 1. (כי הרשת היא בפורט אחר שלו. זה הפרסום הראשון שמכניס את הרשת למשחק אחרי שקינפגנו את 4R.)



- Link between R1 - R8: packet #14, time: 18:52:04
8R בפורט f2/0 מפרסם לראשונה את הרשת בשעה 18:52, עם מרחק = 5.



- Link between R7 - R8: packet #6, time: 18:52:00



R7 פורט f3/0 מפרסם את הרשת לראשונה בשעה 18:52, עם מרחק = 4.

לכן מהרגע שהתחלנו לפרסם, עד שהמידע פורסם לֶסֶד 1 לקח $18:52:04 - 18:51:51 = 13_s$

הזמן שלוקח להודעה לעבור דרך הופ יחיד ניתן לחישוב ע"י: $18:52:04 - 18:52:00 = 4_s$
(הזמן שהפקטה עם המידע הראשוני על הרשת יצאה מ7R עד שיצאה לראשונה מ8r)

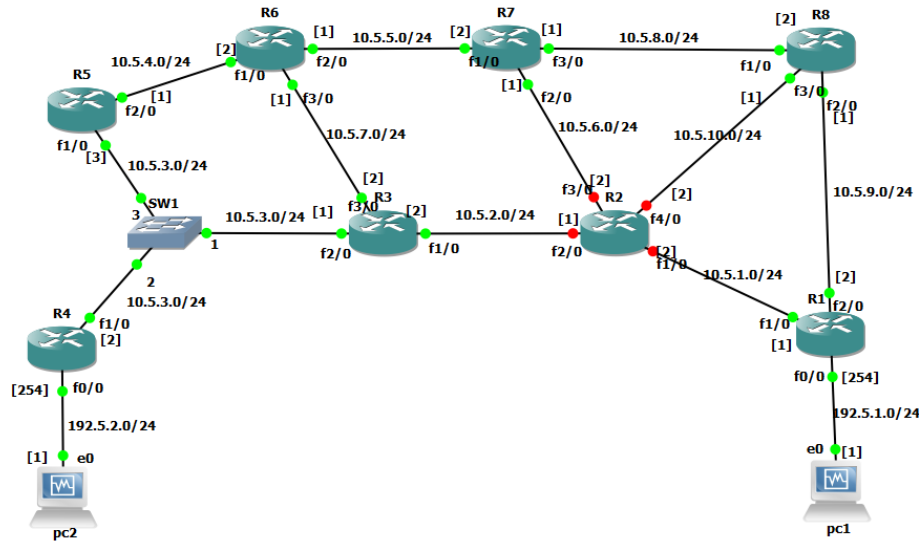
3.20

בRIP 2 משתמשים בשני דברים שעוזרים להתכנסות מהירה יותר של הרשת:

1. Poisoning – כשיש ראوتر שיוצא משימוש מי שמזהה את זה ישר מעדכן את השכנים שלו, לעומת RIP 1 שרק מוחק אותו מהטבלה שלו.
2. trigger updates – ברגע שיש למישהו עידכון בטבלה הוא ישר מפץ הודעה ולא מחכה להודעת הבאה (כל 30 שניות), בrip 1 הוא יחכה אין את העדכון המיידי הזה.

4. Good News Propagation (Adding a Router) - Topology 8.2

4.1



4.3

***verify the default gw, may need to use: route add default gw 192.5.2.254*

Pc2: traceroute from PC2 to PC1

traceroute 192.5.1.1

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 15.298 ms 26.218 ms 37.067 ms
 2 10.5.3.3 (10.5.3.3) 47.325 ms 58.288 ms 68.342 ms
 3 10.5.4.2 (10.5.4.2) 79.017 ms 89.913 ms 100.799 ms
 4 10.5.5.2 (10.5.5.2) 111.115 ms 122.096 ms 132.993 ms
 5 10.5.8.2 (10.5.8.2) 164.092 ms 175.252 ms 185.381 ms
 6 10.5.9.2 (10.5.9.2) 142.907 ms 137.713 ms 168.957 ms
 7 * * *
 8 192.5.1.1 (192.5.1.1) 136.833 ms 134.142 ms 135.944 ms
root@pc2:~#
```

היינו צריכים להריץ שוב כדי לקבל **
טבלה בלי הוכחות (לפעמים מריצים
ARP כמה פעמים עד שנלמדות הטבלאות
כמו שצריך)

4.4 Start Wireshark on link R4 -SWITCH, R3 – R2, R1 – R2

4.5 Pc2: Issue an infinite ping from PC2 to PC1

ping 192.5.1.1

4.6 R2: Start R2 router

wait for the network to converge.

4.7 R1, R4 :verify that the network was convened (הכל את מכילה את הכל)

show ip route

4.8 Pc2: Stop the ping

Ctrl+c

4.9 Stop capture and save the Wireshark pcaps to file

4.10 Pc2: Use traceroute again from PC2 to PC1

traceroute 192.5.1.1

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 13.727 ms 23.529 ms 34.020 ms
 2 10.5.3.1 (10.5.3.1) 44.637 ms 54.881 ms 65.142 ms
 3 10.5.2.1 (10.5.2.1) 75.358 ms 86.057 ms 96.734 ms
 4 10.5.1.1 (10.5.1.1) 106.997 ms 117.571 ms 128.751 ms
 5 192.5.1.1 (192.5.1.1) 138.723 ms 149.024 ms 159.607 ms
root@pc2:~#
```

4.11

נחקור את ההבדל בין המסלול PC2-PC1 לפני ואחרי הוספת 2R:

Before:

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 15.298 ms 26.218 ms 37.067 ms
 2 10.5.3.3 (10.5.3.3) 47.325 ms 58.288 ms 68.342 ms
 3 10.5.4.2 (10.5.4.2) 79.017 ms 89.913 ms 100.799 ms
 4 10.5.5.2 (10.5.5.2) 111.115 ms 122.096 ms 132.993 ms
 5 10.5.8.2 (10.5.8.2) 164.092 ms 175.252 ms 185.381 ms
 6 10.5.9.2 (10.5.9.2) 142.907 ms 137.713 ms 168.957 ms
 7 * * *
 8 192.5.1.1 (192.5.1.1) 136.833 ms 134.142 ms 135.944 ms
root@pc2:~#
```

אחרי:		לפני:	
R4	.1	4R	.1
R3	.2	5R	.2
R2	.3	6R	.3
R1	.4	7R	.4
PC1	.5	8R	.5
		1R	.6
		1PC	.7

**היינו צריכים להריץ שוב כדי לקבל טבלה בלי הכוכביות (לפעמים מריצים כמה פעמים עד שנלמדות הטבלאות ARP כמו שצריך)

After:

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 13.727 ms 23.529 ms 34.020 ms
 2 10.5.3.1 (10.5.3.1) 44.637 ms 54.881 ms 65.142 ms
 3 10.5.2.1 (10.5.2.1) 75.358 ms 86.057 ms 96.734 ms
 4 10.5.1.1 (10.5.1.1) 106.997 ms 117.571 ms 128.751 ms
 5 192.5.1.1 (192.5.1.1) 138.723 ms 149.024 ms 159.607 ms
root@pc2:~#
```

4.12

```
192.5.1.1 ping statistics:
63 packets transmitted, 63 received, 0% packet loss, time 62115ms
```

הפינג נשלח כל הזמן, גם לפני שהדלקנו את 2R וגם אחרי:

4.13

- נחפש את ההודעה הראשונה בה 2R מפרסם שהוא התחיל לעבוד:
בלינק 1R-2R נסנן הודעות RIP עם כתובת IP במקור של 2R

4.9_R1_R2.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

rip && ip.src == 10.5.1.2

No.	Time	Source	Destination	Protocol	Length	Info
10	10:11:38.719187	10.5.1.2	224.0.0.9	RIPv2	66	Request
18	10:11:42.075266	10.5.1.2	224.0.0.9	RIPv2	266	Response
41	10:12:12.336126	10.5.1.2	224.0.0.9	RIPv2	106	Response
74	10:12:29.164812	10.5.1.2	224.0.0.9	RIPv2	226	Response

rip && ip.src == 10.5.1.2

קיבלנו הודעה ראשונה בשעה 10:11:38

- נחפש את ההודעה הראשונה בה 2R מפרסם את הרשת של 2PC אבל במרחק 3 (כלומר לאחר שסיפר ל 3R שיש דרכו דרך וכו')
בלינק 1R-2R נחפש הודעת RIP עם כתובת מקור של 2R שלראשונה מפרסמת את הרשת עם
:metri=3

4.9_R4_SWITCH.pcapng

4.9_R1_R2.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools

rip && ip.src == 10.5.1.2

No.	Time	Source	Destination
10	10:11:38.719187	10.5.1.2	224.0.0.9
18	10:11:42.075266	10.5.1.2	224.0.0.9
41	10:12:12.336126	10.5.1.2	224.0.0.9
74	10:12:29.164812	10.5.1.2	224.0.0.9

Wireshark · Packet 41 · 4.9_R1_R2.pcapng

- Internet Protocol Version 4, Src: 10.5.1.2, Dst: 224.0.0.9
- User Datagram Protocol, Src Port: 520, Dst Port: 520
- Routing Information Protocol
 - Command: Response (2)
 - Version: RIPv2 (2)
 - IP Address: 10.5.3.0, Metric: 2
 - IP Address: 10.5.7.0, Metric: 2
 - IP Address: 192.5.2.0, Metric: 3

rip && ip.src == 10.5.1.2

קיבלנו הודעה ראשונה בשעה 10:12:12 :

סה"כ קיבלנו 34 שניות עד ש 2R פרסם ל 1R שהוא מכיר את הרשת:

$$42: 12: 12 - 10: 11: 38 = 34_s$$

4.14

נחפש פקטת RIP שמעדכנת לראשונה על השינוי והדרך הקצרה יותר

R4 – SWITCH:

פקטה 96, נשלחת מ R3 ומפרסמת לראשונה שיש לה דרך ל PC1 באורך 3.

לאחר ש R4 ילמד את המידע הזה הוא יעביר את הפינג דרך R3

R3 - R2:

פקטה 19, נשלחת מ R2 ומפרסמת לראשונה שיש לה מסלול ל PC1 באורך 2

R2 – R1:

פקטה 4 מ R1 לראשונה מפרסמת ל R2 שיש לה מסלול של 1 ל PC1, R2 לומד ויעביר את ההודעות

4.15

- בהתחלה 2R פרסם לראשונה הודעת RIP REQUEST עם טבלה ריקה
- אחכ קיבל מהשכנים שלו את הטבלאות שלהם
- 2R הפיץ את הטבלה שהוא למד אחרי הנתונים מ 3R, 1R
- ואז שוב השכנים מעדכנים בעדכונים שהם קיבלו
- ואז שוב 2R מעדכן מחדש את הטבלה שלו ושוב מפיץ אותה.

5.Bad News Propagation (Router Down) - Topology 8.3

5.4 Use traceroute from PC2 to PC1

***verify the default gw, may need to use: `route add default gw 192.5.2.254`*

Pc2: traceroute from PC2 to PC1

```
traceroute 192.5.1.1
```

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1  192.5.2.254 (192.5.2.254)  7.392 ms  17.487 ms  28.229 ms
 2  10.5.3.1 (10.5.3.1)  38.949 ms  49.534 ms  60.030 ms
 3  10.5.2.1 (10.5.2.1)  70.609 ms  80.797 ms  91.000 ms
 4  10.5.1.1 (10.5.1.1)  101.615 ms  112.501 ms  122.831 ms
 5  192.5.1.1 (192.5.1.1)  132.715 ms  142.737 ms  153.453 ms
root@pc2:~#
```

5.5 Start Wireshark on R4 - SWITCH, R3 -R2, R1 - R8

5.6 Pc2: Issue an infinite ping from PC2 to PC1

```
ping 192.5.1.1
```

5.7 open computer's clock, stop the R2 write down the “stop time” (in resolution of seconds):

06:13:39

5.8 When the router stops, the ping fails. Wait for the network to converge until the ping returns to work properly (6 minutes).

```
64 bytes from 192.5.1.1: icmp_seq=7 ttl=60 time=76.5 ms
64 bytes from 192.5.1.1: icmp_seq=8 ttl=60 time=53.8 ms
64 bytes from 192.5.1.1: icmp_seq=9 ttl=60 time=63.4 ms
64 bytes from 192.5.1.1: icmp_seq=10 ttl=60 time=100 ms
From 192.5.1.1 icmp_seq=296 Destination Host Unreachable
From 192.5.1.1 icmp_seq=297 Destination Host Unreachable
From 192.5.1.1 icmp_seq=298 Destination Host Unreachable
64 bytes from 192.5.1.1: icmp_seq=458 ttl=58 time=125 ms
64 bytes from 192.5.1.1: icmp_seq=459 ttl=58 time=125 ms
64 bytes from 192.5.1.1: icmp_seq=460 ttl=58 time=121 ms
64 bytes from 192.5.1.1: icmp_seq=461 ttl=58 time=92.4 ms
64 bytes from 192.5.1.1: icmp_seq=462 ttl=58 time=76.1 ms
64 bytes from 192.5.1.1: icmp_seq=463 ttl=58 time=80.0 ms
64 bytes from 192.5.1.1: icmp_seq=464 ttl=58 time=96.9 ms
^C
```

5.9 Pc2: Stop the ping

Ctrl+c

5.10 Stop capture and save the wireshark pcaps to files

5.11 Pc2: traceroute from PC2 to PC1

traceroute 192.5.1.1

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 3.966 ms 14.121 ms 24.296 ms
 2 10.5.3.3 (10.5.3.3) 34.869 ms 45.409 ms 55.925 ms
 3 10.5.4.2 (10.5.4.2) 66.810 ms 77.312 ms 87.207 ms
 4 10.5.5.2 (10.5.5.2) 97.439 ms 108.277 ms 118.828 ms
 5 10.5.8.2 (10.5.8.2) 129.077 ms 138.625 ms 149.344 ms
 6 10.5.9.2 (10.5.9.2) 160.498 ms 167.160 ms 167.976 ms
 7 192.5.1.1 (192.5.1.1) 167.954 ms 167.068 ms 167.861 ms
root@pc2:~#
```

5.12

נתאר את המסלול מ 2PC ל 1PC לפני ואחרי הכיבוי של 2R:

Before:

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 7.392 ms 17.487 ms 28.229 ms
 2 10.5.3.1 (10.5.3.1) 38.949 ms 49.534 ms 60.030 ms
 3 10.5.2.1 (10.5.2.1) 70.609 ms 80.797 ms 91.000 ms
 4 10.5.1.1 (10.5.1.1) 101.615 ms 112.501 ms 122.831 ms
 5 192.5.1.1 (192.5.1.1) 132.715 ms 142.737 ms 153.453 ms
root@pc2:~#
```

לפני:		אחרי:	
R4	.1	4R	.1
R3	.2	5R	.2
R2	.3	6R	.3
R1	.4	7R	.4
PC1	.5	8R	.5
		1R	.6
		1PC	.7

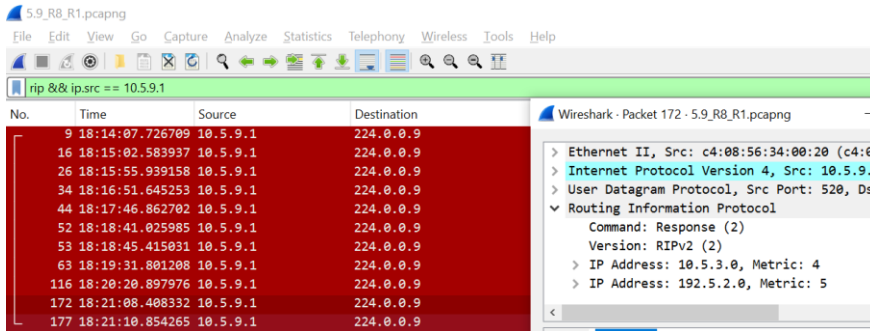
After:

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 3.966 ms 14.121 ms 24.296 ms
 2 10.5.3.3 (10.5.3.3) 34.869 ms 45.409 ms 55.925 ms
 3 10.5.4.2 (10.5.4.2) 66.810 ms 77.312 ms 87.207 ms
 4 10.5.5.2 (10.5.5.2) 97.439 ms 108.277 ms 118.828 ms
 5 10.5.8.2 (10.5.8.2) 129.077 ms 138.625 ms 149.344 ms
 6 10.5.9.2 (10.5.9.2) 160.498 ms 167.160 ms 167.976 ms
 7 192.5.1.1 (192.5.1.1) 167.954 ms 167.068 ms 167.861 ms
root@pc2:~#
```

5.13

נבדוק כמה זמן לקח לרשת להתכנס מהרגע שכיבנו את 2R:

נחפש בקצבים של wireshark את ההודעת RIP שגרמה לפינג להצליח להגיע ל PC1 וגם לחזור ממנו.



הזמן בו כיבנו את 2R: 18:13:38

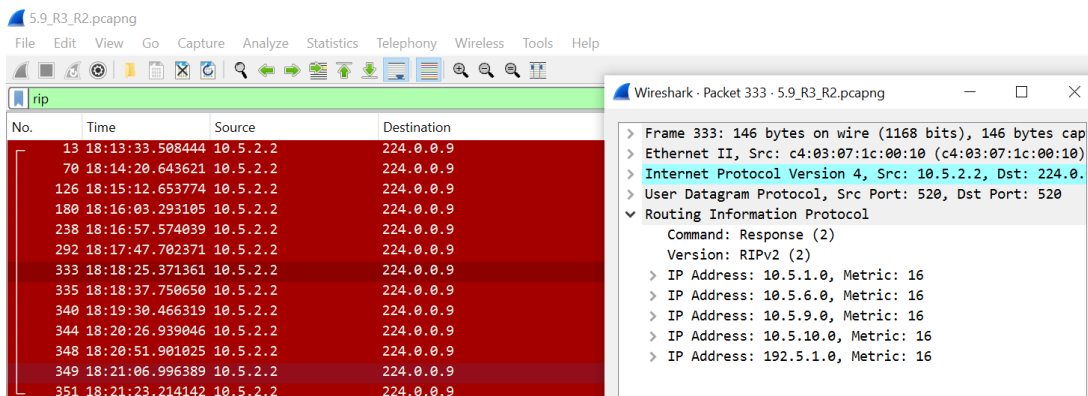
8R מפרסם את 2PC עם מטריקה 5

בפעם הראשונה ב: 18:21:08

כלומר לקח לרשת להתכנס:

$$18:21:08 - 18:13:38 = 7:30_m$$

5.14



בלינק 2R-3R צריך ללכוד את הפקטה בה 3R הבין ש 2R אבד. ונמדוד כמה זמן לקח לו להבין את זה:

כאשר 2R נכבה 3R עדיין המשיך לשלוח דרכו הודעות במשך זמן ה holddown (180 שניות) ע"מ לאפשר לו להתאושש אם זה היה סתם נפילה נקודתית.

אחרי שעבר ה holddown 3R מבין שאין לו גישה דרך 2R לכל מי שעד עכשיו היה מאחוריו ומפרסם שהוא במרחק 16 מהם – כלומר לא נגיש.

הזמן בו נשלחת ההודעה הזו: 18:18:25 אז סה"כ לקח לו להבין $18:18:25 - 18:13:38 = 2:47_m$

5.15

*** צריך להשתמש בסטטיסטיקה של הפינג, אחרי הקונטרול c אבל לא צילמתי את זה, אז בדקנו ב wireshark *** - אם רוצים סטטיסטיקה צריך:

- נקבל את המידע שאומר כמה פינגים נפלו מתוך כל הפינגים שנשלחו

- וזאת כמות השניות שבה הפינג לא עבד, כלומר הזמן בו הרשת עוד לא התכנסה

24	18:13:38.252752	192.5.2.1	192.5.1.1	ICMP	98	Echo (ping) request	id=0x037a, seq=10/2560, ttl=62 (reply in 25)
25	18:13:38.293853	192.5.1.1	192.5.2.1	ICMP	98	Echo (ping) reply	id=0x037a, seq=10/2560, ttl=62 (request in 24)
26	18:13:39.235345	192.5.2.1	192.5.1.1	ICMP	98	Echo (ping) request	id=0x037a, seq=11/2816, ttl=62 (no response found!)
27	18:13:40.242451	192.5.2.1	192.5.1.1	ICMP	98	Echo (ping) request	id=0x037a, seq=12/3072, ttl=62 (no response found!)

עכשיו נבדוק את זמן ההתכנסות בעזרת הפינג, פינג נשלח כל שניה, נחפש מתי הפינג התחיל ליפול ומתי

הוא חזר לעבוד כראוי:
נסתכל על הלינק 3R-2R:

אפשר לראות שהפינג נשלח כראוי עד 18:13:39 שם כבר לא התקבלה תגובה לפינג, וזה אכן הזמן בו כיבינו את הראוטר.

בלינק של 1R-8R אפשר לראות שהפינג חזר לעבוד בשעה 18:21:09 כשאר הרשת מצאה מסלול חלופי.

170	18:21:07.208769	192.5.2.1	192.5.1.1	ICMP	98 Echo (ping) request	id=0x037a, seq=456/51201, ttl=59 (no response found!)
171	18:21:08.216730	192.5.2.1	192.5.1.1	ICMP	98 Echo (ping) request	id=0x037a, seq=457/51457, ttl=59 (no response found!)
172	18:21:08.408332	10.5.9.1	224.0.0.9	RIPv2	86 Response	
173	18:21:09.234535	192.5.2.1	192.5.1.1	ICMP	98 Echo (ping) request	id=0x037a, seq=458/51713, ttl=59 (reply in 174)
174	18:21:09.256088	192.5.1.1	192.5.2.1	ICMP	98 Echo (ping) reply	id=0x037a, seq=458/51713, ttl=63 (request in 173)
175	18:21:10.235614	192.5.2.1	192.5.1.1	ICMP	98 Echo (ping) request	id=0x037a, seq=459/51969, ttl=59 (reply in 176)
176	18:21:10.257433	192.5.1.1	192.5.2.1	ICMP	98 Echo (ping) reply	id=0x037a, seq=459/51969, ttl=63 (request in 175)

נחשב את ההפרש:

$$18:21:09 - 18:13:39 = 7:30_s$$

5.16

Explain, what in the protocol supports the waiting time you calculated?

6.Bad News Propagation (Interface Down) – Topology 8.4

6.3

***verify the default gw, may need to use: route add default gw 192.5.2.254*

Pc2: traceroute from PC2 to PC1

```
tracert 192.5.1.1
```

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254) 6.970 ms 17.530 ms 28.440 ms
 2 10.5.3.1 (10.5.3.1) 38.733 ms 49.469 ms 60.449 ms
 3 10.5.2.1 (10.5.2.1) 71.265 ms 81.718 ms 92.575 ms
 4 10.5.1.1 (10.5.1.1) 103.326 ms 113.685 ms 124.353 ms
 5 192.5.1.1 (192.5.1.1) 135.056 ms 145.723 ms 156.548 ms
root@pc2:~# s
```

6.4 Pc2: Issue an infinite ping from PC2 to PC1

```
ping 192.5.1.1
```

6.5 Shut down the interface f2/0 of router R2.

R2:

```
configure terminal
interface FastEthernet2/0
shutdown
no ip route 10.5.2.0 255.255.255.0
exit
```


6.6 Wait for the network to converge until the ping returns to work properly.

6.7 Pc2: Stop the ping

```
Ctrl+c
^C
--- 192.5.1.1 ping statistics ---
407 packets transmitted, 51 received, +19 errors, 87% packet loss, time 408545ms
rtt min/avg/max/mdev = 44.611/76.728/113.242/15.343 ms
root@pc2:~#
```

6.8 Pc2: traceroute from PC2 to PC1

```
traceroute 192.5.1.1
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1  192.5.2.254 (192.5.2.254)  4.180 ms  14.776 ms  24.987 ms
 2  10.5.3.3 (10.5.3.3)  35.728 ms  45.670 ms  56.502 ms
 3  10.5.4.2 (10.5.4.2)  67.950 ms  78.786 ms  89.393 ms
 4  10.5.5.2 (10.5.5.2)  99.546 ms  110.395 ms  121.965 ms
 5  10.5.6.2 (10.5.6.2)  132.547 ms  142.349 ms  152.688 ms
 6  10.5.1.1 (10.5.1.1)  163.303 ms  169.350 ms  170.767 ms
 7  192.5.1.1 (192.5.1.1)  171.292 ms  170.657 ms  172.015 ms
root@pc2:~#
```

6.9

מרגע שסגרנו את האינטרפייס אף פינג לא יכל לעבור, הזמן שלקח בשניות לרשת להתכנס שוב שווה למספר הפיניגים שנפלו שכן כל שניה נשלח פינג

$$407 - 51 = 365_{[sec]}$$

6.10

```
root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1  192.5.2.254 (192.5.2.254)  6.970 ms  17.530 ms  28.440 ms
 2  10.5.3.1 (10.5.3.1)  38.733 ms  49.469 ms  60.449 ms
 3  10.5.2.1 (10.5.2.1)  71.265 ms  81.718 ms  92.575 ms
 4  10.5.1.1 (10.5.1.1)  103.326 ms  113.685 ms  124.353 ms
 5  192.5.1.1 (192.5.1.1)  135.056 ms  145.723 ms  156.548 ms
root@pc2:~# s
```

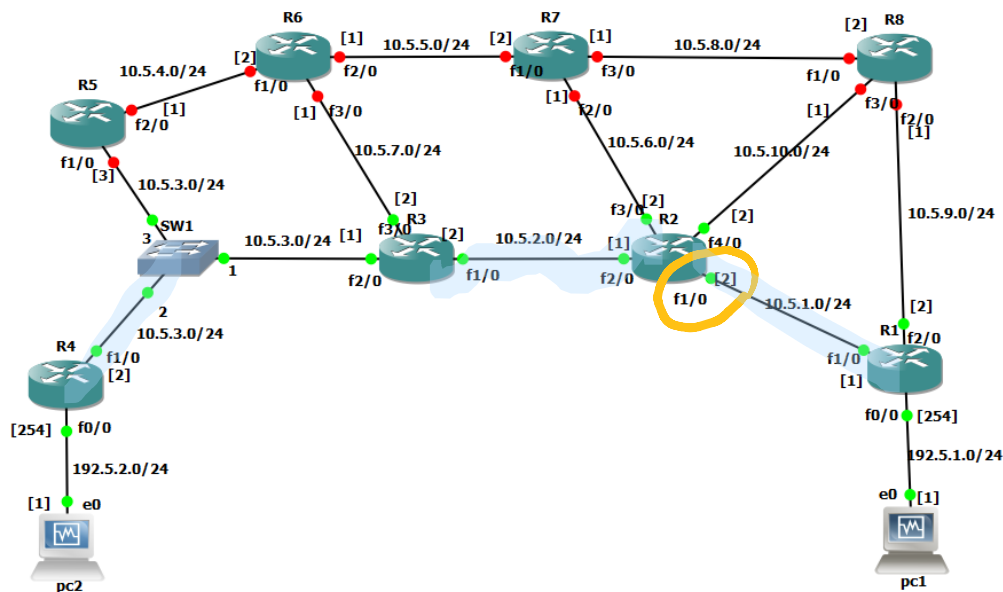
```

root@pc2:~# traceroute 192.5.1.1
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254)  4.180 ms  14.776 ms  24.987 ms
 2 10.5.3.3 (10.5.3.3)  35.728 ms  45.670 ms  56.502 ms
 3 10.5.4.2 (10.5.4.2)  67.950 ms  78.786 ms  89.393 ms
 4 10.5.5.2 (10.5.5.2)  99.546 ms  110.395 ms  121.965 ms
 5 10.5.6.2 (10.5.6.2)  132.547 ms  142.349 ms  152.688 ms
 6 10.5.1.1 (10.5.1.1)  163.303 ms  169.350 ms  170.767 ms
 7 192.5.1.1 (192.5.1.1)  171.292 ms  170.657 ms  172.015 ms
root@pc2:~#

```

אחרי:	לפני:
4R .1	R4 .1
5R .2	R3 .2
6R .3	R2 .3
2R .4	R1 .4
1R .5	PC1 .5
1PC .6	

7.Mechanisms to prevent "Count to Infinity Problem" – Topology 8.5



7.2. Start only: PC1, PC2, R1, R2, R3, R4

7.3. **verify the default gw, may need to use: `route add default gw 192.5.2.254`

Pc2: traceroute from PC2 to PC1

```
traceroute 192.5.1.1
```

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1  192.5.2.254 (192.5.2.254)  7.733 ms  17.561 ms  27.945 ms
 2  10.5.3.1 (10.5.3.1)  38.508 ms  48.512 ms  59.108 ms
 3  10.5.2.1 (10.5.2.1)  70.344 ms  80.476 ms  91.944 ms
 4  10.5.1.1 (10.5.1.1)  101.144 ms  111.530 ms  122.384 ms
 5  192.5.1.1 (192.5.1.1)  133.033 ms  144.031 ms  154.750 ms
root@pc2:~#
```

7.4. Start Wireshark on R4- SWITCH , R3 -R2 , and on R1- R2

7.5. Pc2: Issue an infinite ping from PC2 to PC1

```
ping 192.5.1.1
```

7.6. Open computer's clock, Shut down the interface f1/0 of router R2. 11:13:41

R2:

```
configure terminal
interface FastEthernet1/0
shutdown
no ip route 10.5.2.0 255.255.255.0
exit
```

7.7. The ping will fail. After several attempts that result in "Destination Host Unreachable", stop the ping.

Pc2: Stop the ping

```
Ctrl+c
```

```
From 192.5.1.1 icmp_seq=42 Destination Host Unreachable
From 192.5.1.1 icmp_seq=43 Destination Host Unreachable
From 192.5.1.1 icmp_seq=44 Destination Host Unreachable
From 192.5.1.1 icmp_seq=45 Destination Host Unreachable
From 192.5.1.1 icmp_seq=46 Destination Host Unreachable
From 192.5.1.1 icmp_seq=47 Destination Host Unreachable
^C
--- 192.5.1.1 ping statistics ---
47 packets transmitted, 23 received, +24 errors, 51% packet loss, time 46064ms
rtt min/avg/max/mdev = 42.798/76.486/83.298/7.845 ms
root@pc2:~#
```

7.8. Stop capture and save the wireshark pcaps to files.

7.9.

updates – מספר ישר ולא מחכה ל30 שניות של העדכון הקבוע

split horizon – לא מדווח למי שעובר דרכו על המסלול הטוב שלו

route poisoning – מדווח למי שהמסלול עובר דרכו שהמרחק הוא 16

אם היינו משתמשים בRIPV1 שאין לו את המרכיבים האלו מה היה קורה:

- בגלל שאין **trigger update** ברגע שה interface נפל 2R לא מעדכן אף אחד ומחכה לעדכון הבא.
- בינתיים 3R ממשיך לחשוב שהדרך ל 1PC היא דרך 2R והיא באורך 2.
- בגלל שאין **split horizon** 3R פרסם קודם ל 1R שיש לו דרך ל 1PC באורך של 2.
- ברגע שהיינו מכבים את ה interface של 2R אז הניתוב של 2R ל 1PC היה מתבטל והוא היה מחפש דרך חלופית טובה יותר.
- בגלל ש 3R פרסם לו שיש לו דרך באורך 2, 2R היה מנתב את החבילות ל 1PC דרך 3R.
- 3R היה מחזיר אותן ל 2R כי הוא לא יודע על השינוי אצל 2R (כי אין **trigger update**)
- שוב 2R היה מחזיר אותן ל 3R וככה היינו נתקעים בלופ עד העדכון הבא של 2R שידווח שאין לו קשר ל 1PC.
- **** אם היו המגנונים **split horizon** וגם **route poisoning** אז 3R היה מפרסם ל 2R שהדרך שלו ל 1PC היא באורך 16. ואז 2R בכלל לא היה חושב על האופציה להחליף את הניתוב להיות דרך 3R.

7.10.

- צריך לזהות ב wireshark 2 סיטואציות שונות שנגרמו כתוצאה מהשימוש ב **trigger update** (נדגיש את הזמן בן נשלחה ההודעה)
 - בשעה 11:13:41 כיבינו את ה interface, אפשר לראות ששניה אח"כ 2R פרסם דרך שאר ה interfaces שלו שנפל לו החיבור ל 1PC (packet#63)
 - בלינק 3R-2R אפשר לראות שאיך 2R מפרסם (packet#63) שאין לו גישה ל 1PC ישר (שניה אח"כ) נשלחת הודעה מ 3R (packet#64) שאומרת שגם לה אין גישה ל 1PC

תמונה מלינק 3R-2R:

No.	Time	Source	Destination
5	11:12:57.958263	10.5.2.2	224.0.0.9
8	11:13:13.263306	10.5.2.1	224.0.0.9
63	11:13:42.027747	10.5.2.1	224.0.0.9
64	11:13:44.715672	10.5.2.2	224.0.0.9
65	11:13:43.455177	10.5.2.2	224.0.0.9
68	11:13:47.256484	10.5.2.1	224.0.0.9
74	11:14:16.248908	10.5.2.1	224.0.0.9
80	11:14:42.213611	10.5.2.2	224.0.0.9
81	11:14:43.218053	10.5.2.1	224.0.0.9
87	11:15:09.859942	10.5.2.1	224.0.0.9

Wireshark - Packet 64 · 7.8_R2_R3.pcapng

```

Command: Response (2)
Version: RIPv2 (2)
> IP Address: 10.5.1.0, Metric: 16
> IP Address: 10.5.3.0, Metric: 1
> IP Address: 10.5.7.0, Metric: 1
> IP Address: 10.5.9.0, Metric: 16
> IP Address: 192.5.1.0, Metric: 16
> IP Address: 192.5.2.0, Metric: 2
  
```

Wireshark - Packet 63 · 7.8_R2_R3.pcapng

```

> Internet Protocol Version 4, Src: 10.5.2.1, Dst: 224.0.0.9
> User Datagram Protocol, Src Port: 520, Dst Port: 520
  > Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
    > IP Address: 10.5.1.0, Metric: 16
    > IP Address: 10.5.9.0, Metric: 16
    > IP Address: 192.5.1.0, Metric: 16
  
```

2. סיטואציה אחת שמראה שימוש ב **split horizon**

נסתכל על הודעה ששולח 3R, אפשר לראות שאין שם את הניתוב ל 1PC כי ההודעה הזאת

No.	Time	Source	Destination
5	11:12:57.958263	10.5.2.2	224.0.0.9
8	11:13:13.263306	10.5.2.1	224.0.0.9
63	11:13:42.027747	10.5.2.1	224.0.0.9
64	11:13:44.715672	10.5.2.2	224.0.0.9
65	11:13:45.459175	10.5.2.2	224.0.0.9
68	11:13:47.256484	10.5.2.1	224.0.0.9
74	11:14:16.248908	10.5.2.1	224.0.0.9
80	11:14:42.213611	10.5.2.2	224.0.0.9
81	11:14:43.218053	10.5.2.1	224.0.0.9
87	11:15:09.859942	10.5.2.1	224.0.0.9

Wireshark - Packet 5 - 7.8_R2_R3.pcapng

- Internet Protocol Version 4, Src: 10.5.2.2, Dst: 224.0.0.9
- User Datagram Protocol, Src Port: 520, Dst Port: 520
- Routing Information Protocol
 - Command: Response (2)
 - Version: RIPv2 (2)
 - IP Address: 10.5.3.0, Metric: 1
 - IP Address: 10.5.7.0, Metric: 1
 - IP Address: 192.5.2.0, Metric: 2

רכו הניתוב עובר, לכן 3R לא מפרסם לו את זה.

3. סיטואציה אחת שמראה שימוש ב route poisoning כש 2R מפרסם ל 3R שנפל לו הקשר ל 1PC אז הוא מפרסם את כל הניתובים דרכו באורך 16.

No.	Time	Source	Destination
5	11:12:57.958263	10.5.2.2	224.0.0.9
8	11:13:13.263306	10.5.2.1	224.0.0.9
63	11:13:42.027747	10.5.2.1	224.0.0.9
64	11:13:44.715672	10.5.2.2	224.0.0.9
65	11:13:45.459175	10.5.2.2	224.0.0.9
68	11:13:47.256484	10.5.2.1	224.0.0.9
74	11:14:16.248908	10.5.2.1	224.0.0.9
80	11:14:42.213611	10.5.2.2	224.0.0.9
81	11:14:43.218053	10.5.2.1	224.0.0.9
87	11:15:09.859942	10.5.2.1	224.0.0.9

Wireshark - Packet 63 - 7.8_R2_R3.pcapng

- Internet Protocol Version 4, Src: 10.5.2.1, Dst: 224.0.0.9
- User Datagram Protocol, Src Port: 520, Dst Port: 520
- Routing Information Protocol
 - Command: Response (2)
 - Version: RIPv2 (2)
 - IP Address: 10.5.1.0, Metric: 16
 - IP Address: 10.5.9.0, Metric: 16
 - IP Address: 192.5.1.0, Metric: 16

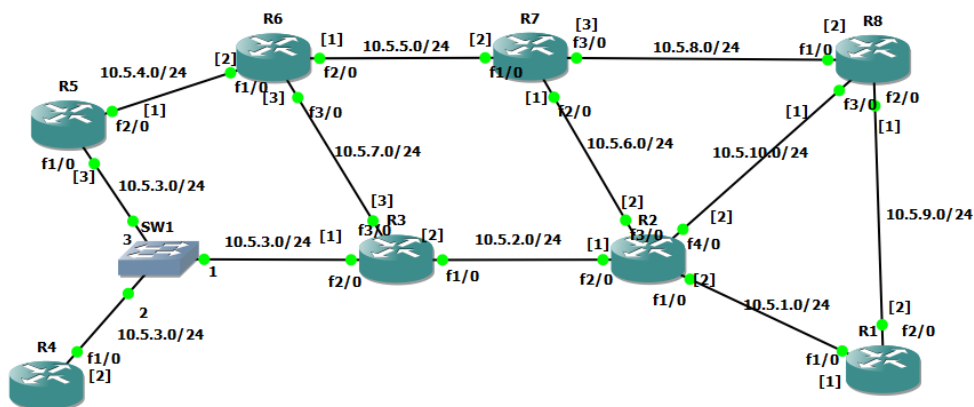
7.11. For each such situation, explain how the relevant mechanism contributes with prevention of count-to-infinity problem.

1.
 - 1.1 ברגע שיש תקלה 2R ישר מעדכן, מונע מהשכנים שלו להעביר לו הודעות סתם למקום שכבר אין לו חיבור אליו.
 - 1.2 ברגע 3R מבין שיש תקלה ל 2R הוא ישר מעדכן גם אצלו וגם את השכנים שלו שכל מי שרוצה להגיע למקומות שעד עכשיו נותבו דרך 2R שלא ינסה דרכו. בגלל במיידיות אנחנו חוסכים את ההודעות המיותרות שיעברו בפינג פונג כי אף אחד חוץ מ 2R לא יודע שהייתה תקלה.

2. אם 3R היה מספר ל 2R שיש לו ניתוב באורך 2 ל 1PC אז ברגע שה interface 2R היה מעביר את ההודעות ל 3R, 3R היה מחזיר לו אותן והיה נוצר פינג פונג.
3. אם 2R לא היה מפרסם ל 3R שכל הניתובים שנפלו הם באורך 16, אז 2R היה מגדיל את המרחק שלו ל 3, ואז 3R היה מגדיל ל 4 וככה הפינג פונג היה נמשך לנצח. בגלל שהוא אמר 16 – כלומר לא ישיג, אז 3R לא יכול עוד להגדיל את המרחק שלו ל 1PC בפעם הראשונה כבר בפינג פונג והוא היה נעצר.

Intra AS Routing (Part B): OSPF

8.Configuring OSPF on Cisco Routers – Topology 9



8.4. Start Wireshark on R4 – SWITCH

8.5. Enable OSPF on all routers:

R1 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.1</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	R2 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.2</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	R3 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.3</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>
R4 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.4</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	R5 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.5</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	R6 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.6</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>
R7 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.7</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	R8 <i>configure terminal</i> <i>router ospf 1</i> <i>router-id 10.5.0.8</i> <i>network 10.5.0.0 0.0.255.255</i> <i>area 1</i> <i>end</i>	

8.6. Wait for the network convergence.

8.7. Stop capture and save the Wireshark pcap to file.

8.8. You should now be able to ping any router from any other router, **R1**:

R2	<i>ping 10.5.1.2</i>
R3	<i>ping 10.5.2.2</i>
R4	<i>ping 10.5.3.2</i>
R5	<i>ping 10.5.3.3</i>
R6	<i>ping 10.5.4.2</i>
R7	<i>ping 10.5.6.1</i>
R8	<i>ping 10.5.9.1</i>

8.9. Perform "write" on each router, save the topology (as Topology 9) and ZIP it.

<i>write</i>

8.10. Leave the network running, in order to answer the exercise questions.

8.11. Explain the meaning of each command from step 5:

configure terminal

router ospf 1 - מספר התהליך בו הראוטר משתתף

router-id 10.5.0.1 - ID של הראוטר בתוך התהליך של ה OSPF

network 10.5.0.0 0.0.255.255 area 1 - הראוטר מפרסם באזור 1, בו המשתתפים ילמדו על הרשת
10.5.0.0/16 בהודעות OSPF

end

8.12.

OSPF משתמש ישירות בשכבת ה network, ושולח פקטות של IP

8.13.

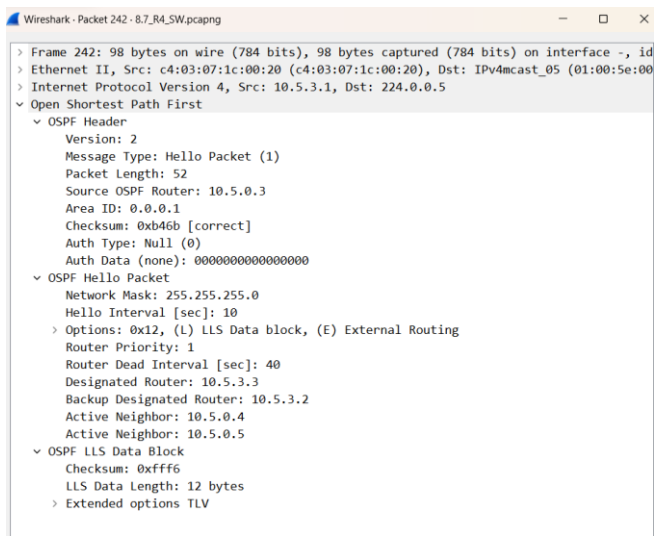
נחפש ב wireshark את פקטת ה OSPF הכי גדולה מכל סוג.

לכל פקטה נסביר מה המטרה שלה, ונסביר מה הפרוטוקול לומד ממנה (main field):

1. Hello packet

משמשת לתחזוק וזיהוי של השכנים, כל אחד שולח הודעות כאלה לשכנים שלו במחזורי זמן קבועים. (10 שניות)
הפקטות האלה מכילות מידע על ה network של המכשיר, על האזור שלו, ועל ה ID.

הודעה מ R3:



2. Link State Ack – בפקטות אלו מועבר המידע על מצב הלינקים. (כתגובה ל update) 4R פרסם הודעה עם כל הלינקים שהוא מכיר:

```
Wireshark - Packet 98 - 8.7_R4_SW.pcapng
> Frame 98: 378 bytes on wire (3024 bits), 378 bytes captured (3024 bits) on interface -, id 0
> Ethernet II, Src: c4:04:4a:48:00:10 (c4:04:4a:48:00:10), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
> Internet Protocol Version 4, Src: 10.5.3.2, Dst: 224.0.0.5
  Open Shortest Path First
    OSPF Header
    LSA-type 1 (Router-LSA), len 48
      .000 0000 0000 0101 = LS Age (seconds): 5
      0... .... = Do Not Age Flag: 0
    Options: 0x22, (DC) Demand Circuits, (E) External Routing
      LS Type: Router-LSA (1)
      Link State ID: 10.5.0.1
      Advertising Router: 10.5.0.1
      Sequence Number: 0x80000003
      Checksum: 0x1099
      Length: 48
    LSA-type 1 (Router-LSA), len 48
    LSA-type 1 (Router-LSA), len 72
    LSA-type 1 (Router-LSA), len 60
```

3. Link State Requests - אם רוצים לבקש מידע ספציפי מהשכן (כלומר לבקש הועדת LSA ספציפית), אם לאותו שכן אין את המידע הוא יכול להעביר את הבקשה הלאה.

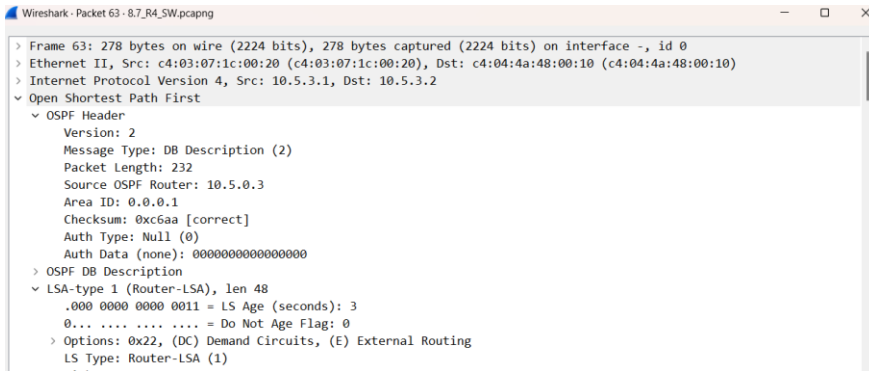
```
Wireshark - Packet 71 - 8.7_R4_SW.pcapng
> Frame 71: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface -, id 0
> Ethernet II, Src: c4:04:4a:48:00:10 (c4:04:4a:48:00:10), Dst: c4:03:07:1c:00:20 (c4:03:07:1c:00:20)
> Internet Protocol Version 4, Src: 10.5.3.2, Dst: 10.5.3.1
  Open Shortest Path First
    OSPF Header
    Link State Request
      LS Type: Router-LSA (1)
      Link State ID: 10.5.0.1
      Advertising Router: 10.5.0.1
    Link State Request
      LS Type: Router-LSA (1)
      Link State ID: 10.5.0.2
      Advertising Router: 10.5.0.2
    Link State Request
    Link State Request
    Link State Request
    Link State Request
```

0000	c4 03 07 1c 00 20	c4 04 4a 48 00 10 08 00 45 c0H....E.
0010	00 a4 00 15 00 00 01 59	9e 20 0a 05 03 02 0a 05Y -
0020	03 01 02 03 00 90 0a 05	00 04 00 00 00 01 0e b1

4. Link State Updates - מפרסם את המרחקים שלמד משאר הראוטרים ברשת, מי שמקבל הודעה כזאת מעדכן את עצמו.

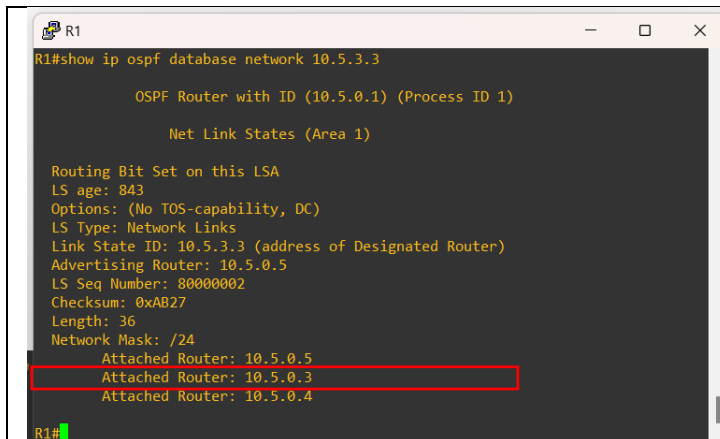
```
Wireshark - Packet 79 - 8.7_R4_SW.pcapng
> Frame 79: 522 bytes on wire (4176 bits), 522 bytes captured (4176 bits) on interface -, id 0
> Ethernet II, Src: c4:03:07:1c:00:20 (c4:03:07:1c:00:20), Dst: c4:04:4a:48:00:10 (c4:04:4a:48:00:10)
> Internet Protocol Version 4, Src: 10.5.3.1, Dst: 10.5.3.2
  Open Shortest Path First
    OSPF Header
    LS Update Packet
      Number of LSAs: 10
      LSA-type 1 (Router-LSA), len 48
      LSA-type 1 (Router-LSA), len 72
      LSA-type 1 (Router-LSA), len 60
      LSA-type 1 (Router-LSA), len 60
      LSA-type 1 (Router-LSA), len 60
      LSA-type 2 (Network-LSA), len 32
      LSA-type 2 (Network-LSA), len 32
      LSA-type 2 (Network-LSA), len 32
      LSA-type 2 (Network-LSA), len 32
      LSA-type 2 (Network-LSA), len 32
```

5. DBD - מוודא שיש לראוטרם את כל המידע ולא פספסו כלום



8.14.

R1: 10.5.3.0/24



show ip ospf database network 10.5.3.3

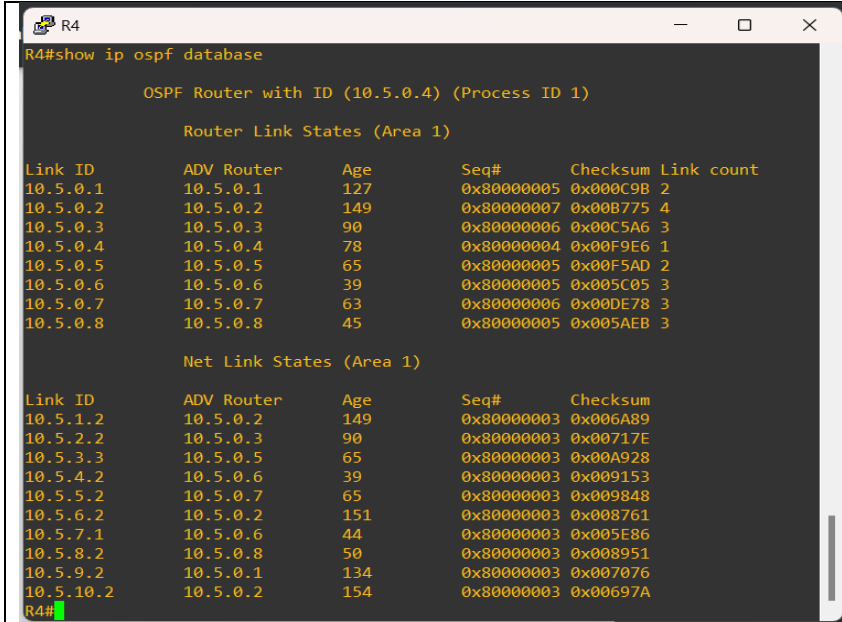
באמצעות הפקודה למעלה ניתן לראות השיוך בין הרשת לראוטרם.

<pre> R1 R1#show ip ospf database router 10.5.0.3 OSPF Router with ID (10.5.0.1) (Process ID 1) Router Link States (Area 1) LS age: 1152 Options: (No TOS-capability, DC) LS Type: Router Links Link State ID: 10.5.0.3 Advertising Router: 10.5.0.3 LS Seq Number: 80000005 Checksum: 0xC7A5 Length: 60 Number of Links: 3 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.7.1 (Link Data) Router Interface address: 10.5.7.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.3.3 (Link Data) Router Interface address: 10.5.3.1 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.2.2 (Link Data) Router Interface address: 10.5.2.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 </pre>	<p><i>show ip ospf database router 10.5.0.3</i></p> <p>בשלב זה מסתכלים לאיזה רשתות ראوتر 3 מתחבר</p>
<pre> R1#show ip ospf database network 10.5.2.2 OSPF Router with ID (10.5.0.1) (Process ID 1) Net Link States (Area 1) Routing Bit Set on this LSA LS age: 1314 Options: (No TOS-capability, DC) LS Type: Network Links Link State ID: 10.5.2.2 (address of Designated Router) Advertising Router: 10.5.0.3 LS Seq Number: 80000002 Checksum: 0x737D Length: 32 Network Mask: /24 Attached Router: 10.5.0.3 Attached Router: 10.5.0.2 </pre>	<p><i>show ip ospf database network 10.5.2.2</i></p>

<pre> R1#show ip ospf database router 10.5.0.2 OSPF Router with ID (10.5.0.1) (Process ID 1) Router Link States (Area 1) LS age: 1384 Options: (No TOS-capability, DC) LS Type: Router Links Link State ID: 10.5.0.2 Advertising Router: 10.5.0.2 LS Seq Number: 80000006 Checksum: 0xB974 Length: 72 Number of Links: 4 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.10.2 (Link Data) Router Interface address: 10.5.10.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.6.2 (Link Data) Router Interface address: 10.5.6.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.2.2 (Link Data) Router Interface address: 10.5.2.1 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.1.2 (Link Data) Router Interface address: 10.5.1.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 </pre>	<pre>show ip ospf database router 10.5.0.2</pre>
<pre> R1#show ip ospf database network 10.5.1.2 OSPF Router with ID (10.5.0.1) (Process ID 1) Net Link States (Area 1) Routing Bit Set on this LSA LS age: 1508 Options: (No TOS-capability, DC) LS Type: Network Links Link State ID: 10.5.1.2 (address of Designated Router) Advertising Router: 10.5.0.2 LS Seq Number: 80000002 Checksum: 0x6C88 Length: 32 Network Mask: /24 Attached Router: 10.5.0.2 Attached Router: 10.5.0.1 </pre>	<pre>show ip ospf database network 10.5.1.2</pre>
<pre> R1#show ip ospf database router 10.5.0.1 OSPF Router with ID (10.5.0.1) (Process ID 1) Router Link States (Area 1) LS age: 1654 Options: (No TOS-capability, DC) LS Type: Router Links Link State ID: 10.5.0.1 Advertising Router: 10.5.0.1 LS Seq Number: 80000004 Checksum: 0xE9A Length: 48 Number of Links: 2 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.9.2 (Link Data) Router Interface address: 10.5.9.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.1.2 (Link Data) Router Interface address: 10.5.1.1 Number of TOS metrics: 0 TOS 0 Metrics: 1 </pre>	<pre>show ip ospf database router 10.5.0.1</pre> <p>חקרנו אחורה מהיעד אל ראوتر 1 בחזרה דרך המידע על גבי ראوتر בלבד. מהרשת לנתב בכל איטרציה, התקדמנו הופ אחר הופ עד שהגענו לנקודת המוצא.</p>

8.15.

בשיטת OSPF אין היררכיה ועל כן כל המידע נמצא ברשות כל הראוטרים המשתתפים כל הזמן (כל עוד הרשת מכונסת)



```
R4#show ip ospf database

OSPF Router with ID (10.5.0.4) (Process ID 1)

Router Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum Link count
10.5.0.1     10.5.0.1     127        0x80000005  0x000C9B  2
10.5.0.2     10.5.0.2     149        0x80000007  0x00B775  4
10.5.0.3     10.5.0.3     90         0x80000006  0x00C5A6  3
10.5.0.4     10.5.0.4     78         0x80000004  0x00F9E6  1
10.5.0.5     10.5.0.5     65         0x80000005  0x00F5AD  2
10.5.0.6     10.5.0.6     39         0x80000005  0x005C05  3
10.5.0.7     10.5.0.7     63         0x80000006  0x00DE78  3
10.5.0.8     10.5.0.8     45         0x80000005  0x005AEB  3

Net Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum
10.5.1.2     10.5.0.2     149        0x80000003  0x006A89
10.5.2.2     10.5.0.3     90         0x80000003  0x00717E
10.5.3.3     10.5.0.5     65         0x80000003  0x00A928
10.5.4.2     10.5.0.6     39         0x80000003  0x009153
10.5.5.2     10.5.0.7     65         0x80000003  0x009848
10.5.6.2     10.5.0.2     151        0x80000003  0x008761
10.5.7.1     10.5.0.6     44         0x80000003  0x005E86
10.5.8.2     10.5.0.8     50         0x80000003  0x008951
10.5.9.2     10.5.0.1     134        0x80000003  0x007076
10.5.10.2    10.5.0.2     154        0x80000003  0x00697A

R4#
```

```
show ip ospf database
```

9. Measuring OSPF Convergence Time

9.4. Add and configure the *PCs* according to the figure. Set appropriate default gateway for each *PC*.

**** ייתכן ויהיה צורך למחוק בR2 את הקינפוג של המחשבים שנשאר מלפני (ממעבדה קודמת)
לבדוק לפני אם הקישור קיים בעזרת: show ip route**

```
configure terminal
interface FastEthernet0/0
ip address 192.168.1.254 255.255.255.0
shutdown

end
```

end נקבע IP סטטי לשני המחשבים, ונגדיר default gateway עבורם.
את הראוטרים נקבע בסעיפים הבאים

PC1	PC2
<pre>nano /etc/network/interfaces auto eth0 iface eth0 inet static address 192.5.1.1 netmask 255.255.255.0 ctrl+x y enter reboot route add default gw 192.5.1.254</pre>	<pre>nano /etc/network/interfaces auto eth0 iface eth0 inet static address 192.5.2.1 netmask 255.255.255.0 ctrl+x y enter reboot route add default gw 192.5.2.254</pre>

9.5. show OSPF routing process.

```
R1:

configure terminal
router ospf 1
network 192.5.1.0 0.0.0.255 area 1
passive-interface FastEthernet 0/0
interface FastEthernet 0/0
ip address 192.5.1.254 255.255.255.0
no shutdown
end
```

9.6. Use *traceroute* from *PC1* in order to verify that your network convened.

```
traceroute 10.5.3.2
```

```
root@pc1:~# traceroute 10.5.3.2
traceroute to 10.5.3.2 (10.5.3.2), 30 hops max, 60 byte packets
 1  192.5.1.254 (192.5.1.254)  8.298 ms  18.839 ms  30.373 ms
 2  10.5.9.1 (10.5.9.1)  40.538 ms  50.781 ms  61.787 ms
 3  10.5.8.1 (10.5.8.1)  71.599 ms  82.860 ms  93.852 ms
 4  10.5.5.1 (10.5.5.1)  103.470 ms  113.871 ms  123.757 ms
 5  10.5.4.1 (10.5.4.1)  134.194 ms  145.559 ms  155.457 ms
 6  10.5.3.2 (10.5.3.2)  165.863 ms  167.894 ms  170.811 ms
root@pc1:~# _
```

9.7. Start Wireshark on *R1* interface *f2/0*, on *R8* interface *f1/0* and on *R4* interface *f1/0*.

9.8. Change the time display format of the *Wireshark's* time Column, to "Time of day".

9.9. Issue an infinite *ping* from *PC1* to *PC2*. You should see that the ping failed.

```
ping 192.5.2.1
```

9.10. Similarly, to *R1*, now configure the network interface *f0/0* of *R4* and set as *passive-interface*. Add the new network to the OSPF routing process.

```
R4:

configure terminal
router ospf 1
```

```

network 192.5.2.0 0.0.0.255 area 1
passive-interface FastEthernet 0/0
interface FastEthernet 0/0
ip address 192.5.2.254 255.255.255.0
no shutdown
end

```

9.11. Wait for the network convergence. The ping should start working properly.

9.12. Stop the ping command.

Ctrl+C

```

64 bytes from 192.5.1.254: icmp_seq=87 ttl=58 time=110 ms
^C
--- 192.5.2.1 ping statistics ---
87 packets transmitted, 22 received, +64 errors, 74% packet loss, time 86180ms
rtt min/avg/max/mdev = 80.189/113.248/127.511/12.694 ms
root@pc1:~# _

```

9.13. Stop capture and save the *wireshark pcaps* to files.

9.14. Perform "write" on routers R1 and R4, and save the topology for later exercises.

9.14.1. Keep the project open for question 9.17 .

9.15. Describe the *OSPF* distribution process of the information about the new added subnet. Pay attention to times and to the chronological order of the packets. Refer to information that the packets contain.

OSPF UPDATE ברגע ש R4 מקבל את המידע שהסאבנט של PC2 מחובר אליו, הוא מייד שולח הודעה על כך ליתר הראוטרים

47	14:41:01.743658	10.5.3.2	224.0.0.5	OSPF	110 LS Update
48	14:41:04.264866	10.5.3.3	224.0.0.5	OSPF	78 LS Acknowledge
49	14:41:04.266220	10.5.3.1	224.0.0.6	OSPF	78 LS Acknowledge
50	14:41:04.612844	c4:04:4a:48:00:10	c4:04:4a:48:00:10	LOOP	60 Reply
51	14:41:06.559153	10.5.3.1	224.0.0.5	OSPF	98 Hello Packet

בתוכן ההודעה הוא מספר להם על הלינק החדש:

> Flags: SxTx

Number of Links: 2

> Type: Stub ID: 192.5.2.0 Data: 255.255.255.0 Metric: 10
 > Type: Transit ID: 10.5.3.3 Data: 10.5.3.2 Metric: 1

המידע זורם ברשת, ועובר בין הראוטרים השונים:

R7-r8

50	14:41:01.775746	10.5.8.1	224.0.0.5	OSPF	110	LS Update
51	14:41:04.275018	10.5.8.2	224.0.0.5	OSPF	78	LS Acknowledge
52	14:41:04.612844	c4:08:56:34:00:10	c4:08:56:34:00:10	LOOP	60	Reply
pc1-r1						
56	14:41:01.786201	10.5.9.1	224.0.0.5	OSPF	110	LS Update
57	14:41:04.306044	10.5.9.2	224.0.0.5	OSPF	78	LS Acknowledge
58	14:41:05.480763	c4:08:56:34:00:20	c4:08:56:34:00:20	LOOP	60	Reply
R4 לבסוף מכל העדכון הנל חוזרים AKC						
47	14:41:01.743658	10.5.3.2	224.0.0.5	OSPF	110	LS Update
48	14:41:04.264866	10.5.3.3	224.0.0.5	OSPF	78	LS Acknowledge
49	14:41:04.266220	10.5.3.1	224.0.0.6	OSPF	78	LS Acknowledge
50	14:41:04.612844	c4:04:4a:48:00:10	c4:04:4a:48:00:10	LOOP	60	Reply
51	14:41:06.559153	10.5.3.1	224.0.0.5	OSPF	98	Hello Packet
לאחר כל העדכונים אפשר לראות שהפינגים מתחילים לזרום:						
47	14:41:01.743658	10.5.3.2	224.0.0.5	OSPF	110	LS Update
48	14:41:04.264866	10.5.3.3	224.0.0.5	OSPF	78	LS Acknowledge
49	14:41:04.266220	10.5.3.1	224.0.0.6	OSPF	78	LS Acknowledge
50	14:41:04.612844	c4:04:4a:48:00:10	c4:04:4a:48:00:10	LOOP	60	Reply
51	14:41:06.559153	10.5.3.1	224.0.0.5	OSPF	98	Hello Packet
52	14:41:06.917286	10.5.3.2	224.0.0.5	OSPF	98	Hello Packet
53	14:41:07.329464	10.5.3.3	224.0.0.5	OSPF	98	Hello Packet
54	14:41:07.652122	192.5.1.1	192.5.2.1	ICMP	98	Echo (ping) request id=0x036f, seq=65/16640, ttl=59 (no response found!)
55	14:41:08.251160	c4:03:07:1c:00:20	CDP/VTP/DTP/PagP/UD...	CDP	351	Device ID: R3 Port ID: FastEthernet2/0
56	14:41:08.652594	192.5.1.1	192.5.2.1	ICMP	98	Echo (ping) request id=0x036f, seq=66/16896, ttl=59 (reply in 57)
57	14:41:08.672669	192.5.2.1	192.5.1.1	ICMP	98	Echo (ping) reply id=0x036f, seq=66/16896, ttl=63 (request in 56)
58	14:41:08.809469	c4:04:4a:48:00:10	CDP/VTP/DTP/PagP/UD...	CDP	351	Device ID: R4 Port ID: FastEthernet1/0

9.16. Measure how long it takes for *OSPF* information to spread throughout the entire network, and how long it takes for the information to pass through a single hop.

<p>מרגע שליחת העדכון ועד חזרת האק האחרון זה הזמן שלקח לרשת להתעדכן הוא 2.5225 שניות.</p> <p>אילו הכוונה היא לכמה זמן לקח לעדכון להגיע להופ הרחוק ביותר אז</p> $1.7862 - 1.74386 = 0.04234sec$
<p>נסתכל על הזמן שלקח לראוטר 8 להעביר את העדכון לראוטר 1:</p> <p>1.7757 עד 1.7862</p> <p>נקבל 0.0105 שניות בין הופסה להופלה</p>

9.17. Find out how *OSPF* calculates the *metric* for an interface. Explain what causes the difference in *metric* between the advertised entries. For that purpose, use the following command and explain how it can help to find the reason. *show interfaces FastEthernet interface-number*

<p>להלן שלושת הממשקים של ראוטר 1.</p> <p><i>show interfaces</i></p>	<pre> R1#show interfaces FastEthernet0/0 FastEthernet0/0 is up, line protocol is up Hardware is Gt96k FE, address is c401.1b7c.0000 (bia c401.1b7c.0000) Internet address is 192.5.1.254/24 MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, reliability 255/255, txload 1/255, rxload 1/255 Encapsulation ARPA, loopback not set FastEthernet1/0 is up, line protocol is up Hardware is AmdFE, address is c401.1b7c.0010 (bia c401.1b7c.0010) Internet address is 10.5.1.1/24 MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255 FastEthernet2/0 is up, line protocol is up Hardware is AmdFE, address is c401.1b7c.0020 (bia c401.1b7c.0020) Internet address is 10.5.9.2/24 MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255, rxload 1/255 </pre>
<p><i>show ip ospf database router</i></p> <p>The Cost formula is reference bandwidth divided by interface bandwidth. The default reference bandwidth of 100 Mbps is used for OSPF cost calculation.</p> <p>מהנוסחא ה"נ ניתן לראות כי המטריקה אל המחשב תהא 10 וגדולה פי 1 מהמטריקו בין הראוטרם.</p>	<pre> R1#show ip ospf database router OSPF Router with ID (10.5.0.1) (Process ID 1) Router Link States (Area 1) LS age: 605 Options: (No TOS-capability, DC) LS Type: Router Links Link State ID: 10.5.0.1 Advertising Router: 10.5.0.1 LS Seq Number: 80000007 Checksum: 0x4C8B Length: 60 Number of Links: 3 Link connected to: a Stub Network (Link ID) Network/subnet number: 192.5.1.0 (Link Data) Network Mask: 255.255.255.0 Number of TOS metrics: 0 TOS 0 Metrics: 10 Link connected to: a Transit Network (Link ID) Designated Router address: 10.5.9.1 (Link Data) Router Interface address: 10.5.9.2 Number of TOS metrics: 0 TOS 0 Metrics: 1 Link connected to: a Stub Network (Link ID) Network/subnet number: 10.5.1.0 (Link Data) Network Mask: 255.255.255.0 Number of TOS metrics: 0 TOS 0 Metrics: 1 </pre>

10. Good News Propagation (Adding a Router)

First make sure default gw is set:

PC1	PC2
<code>route add default gw 192.5.1.254</code>	<code>route add default gw 192.5.2.254</code>

10.2. Start all the network components, **except for R2**.

10.3. Use *tracert* from PC2 to PC1 in order to verify that your network converged.

```
tracert 192.5.1.1
```

```
tracert to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254)  9.721 ms  20.701 ms  31.656 ms
 2 10.5.3.3 (10.5.3.3)  42.351 ms  52.728 ms  62.906 ms
 3 10.5.4.2 (10.5.4.2)  73.127 ms  83.769 ms  92.633 ms
 4 10.5.5.2 (10.5.5.2)  103.467 ms  113.687 ms  123.828 ms
 5 10.5.8.2 (10.5.8.2)  133.795 ms  143.628 ms  154.406 ms
 6 10.5.9.2 (10.5.9.2)  165.781 ms  164.610 ms  166.704 ms
 7 192.5.1.1 (192.5.1.1)  166.041 ms  164.773 ms  165.754 ms
root@pc2:~#
```

10.4. Start *Wireshark* on R4 interface *f1/0*, on R3 interface *f1/0* and on R1 interface *f1/0*.

10.5. Issue an infinite *ping* from PC2 to PC1.

```
ping 192.5.1.1
```

10.6. Start router R2 and wait for the network to converge. Notice that when we start R2, we provide a better route between the two PCs. Observe *Wireshark* on links R3-R2 and R2-R1 to see when the ICMP packets (the Request and the reply) start to pass through the new route.

10.7. Look at the routing table of R1 and R4 and verify that the network was converged.

R1	R4
<pre>10.0.0.0/24 is subnetted, 10 subnets O 10.5.10.0 [110/2] via 10.5.9.1, 00:07:41, FastEthernet2/0 [110/2] via 10.5.1.2, 00:03:15, FastEthernet1/0 C 10.5.9.0 is directly connected, FastEthernet2/0 O 10.5.8.0 [110/2] via 10.5.9.1, 00:07:41, FastEthernet2/0 O 10.5.7.0 [110/3] via 10.5.1.2, 00:03:15, FastEthernet1/0 O 10.5.6.0 [110/2] via 10.5.1.2, 00:03:15, FastEthernet1/0 O 10.5.5.0 [110/3] via 10.5.9.1, 00:07:41, FastEthernet2/0 [110/3] via 10.5.1.2, 00:03:16, FastEthernet1/0 O 10.5.4.0 [110/4] via 10.5.9.1, 00:07:42, FastEthernet2/0 [110/4] via 10.5.1.2, 00:03:16, FastEthernet1/0 O 10.5.3.0 [110/3] via 10.5.1.2, 00:03:16, FastEthernet1/0 O 10.5.2.0 [110/2] via 10.5.1.2, 00:03:16, FastEthernet1/0 C 10.5.1.0 is directly connected, FastEthernet1/0 C 192.5.1.0/24 is directly connected, FastEthernet0/0 O 192.5.2.0/24 [110/13] via 10.5.1.2, 00:03:22, FastEthernet1/0 S 192.168.2.0/24 [1/0] via 10.5.1.2 R1#</pre>	<pre>10.0.0.0/24 is subnetted, 10 subnets O 10.5.10.0 [110/3] via 10.5.3.1, 00:03:06, FastEthernet1/0 O 10.5.9.0 [110/4] via 10.5.3.1, 00:03:06, FastEthernet1/0 O 10.5.8.0 [110/4] via 10.5.3.3, 00:07:32, FastEthernet1/0 [110/4] via 10.5.3.1, 00:07:32, FastEthernet1/0 O 10.5.7.0 [110/2] via 10.5.3.1, 00:07:32, FastEthernet1/0 O 10.5.6.0 [110/3] via 10.5.3.1, 00:03:06, FastEthernet1/0 O 10.5.5.0 [110/3] via 10.5.3.3, 00:07:32, FastEthernet1/0 [110/3] via 10.5.3.1, 00:07:33, FastEthernet1/0 O 10.5.4.0 [110/2] via 10.5.3.3, 00:07:43, FastEthernet1/0 C 10.5.3.0 is directly connected, FastEthernet1/0 O 10.5.2.0 [110/2] via 10.5.3.1, 00:07:43, FastEthernet1/0 O 10.5.1.0 [110/3] via 10.5.3.1, 00:03:07, FastEthernet1/0 O 192.5.1.0/24 [110/13] via 10.5.3.1, 00:03:25, FastEthernet1/0 C 192.5.2.0/24 is directly connected, FastEthernet0/0 R4#</pre>

10.8. Stop the ping command using `ctrl+c` and save the ping statistics.

10.9. Stop capture and save the *wireshark pcaps* to files

10.10. Did the network changes affect the ping application? If so, how many packets were lost?

הפיגים לא נאבדו

```
--- 192.5.1.1 ping statistics ---
41 packets transmitted, 41 received, 0% packet loss, time 40066ms
rtt min/avg/max/mdev = 58.772/103.665/128.342/20.387 ms
root@pc2:~#
```

זאת משום שבכל רגע נתון הם הטבלאת ניתוב מכילה מסלולים מלאים.

10.11. Using the *Wireshark pcaps*, measure how long it takes the network to converge. Explain, how did you calculate this?

נמדוד זמן בין הודעת העדכון הראשונה של ראטר 4:

No.	Time	Source	Destination	Protocol	Length	Info
96	20:26:47.389930	10.5.3.1	224.0.0.6	OSPF	134	LS Update
97	20:26:47.400712	10.5.3.3	224.0.0.5	OSPF	122	LS Update
98	20:26:47.400712	10.5.3.3	224.0.0.5	OSPF	242	LS Update
99	20:26:47.453157	10.5.3.3	224.0.0.5	OSPF	94	LS Update
100	20:26:47.474905	10.5.3.3	224.0.0.5	OSPF	94	LS Update
101	20:26:47.474905	10.5.3.3	224.0.0.5	OSPF	94	LS Update
102	20:26:47.519105	10.5.3.3	224.0.0.5	OSPF	94	LS Update
103	20:26:47.561735	10.5.3.3	224.0.0.5	OSPF	94	LS Update
104	20:26:47.624068	10.5.3.3	224.0.0.5	OSPF	94	LS Update
105	20:26:47.655375	10.5.3.3	224.0.0.5	OSPF	94	LS Update

ויד האק האחרון שהוא שלח:

No.	Time	Source	Destination	Protocol	Length	Info
114	20:26:49.086725	192.5.1.1	192.5.2.1	ICMP	98	Echo (ping) reply id=0
115	20:26:49.914223	10.5.3.2	224.0.0.5	OSPF	238	LS Acknowledge
116	20:26:49.987830	192.5.2.1	192.5.1.1	ICMP	98	Echo (ping) request id=0

זמן עד להתכנסות המערכת:

$$49.914 - 47.389 = 2.024_{[sec]}$$

10.12. At the pcap of subnet 10.x.3.0, follow the update packets that spread in the network because the router started. Use the display filter: “*ospf.msg.lsupdate and ip.dst == 224.0.0.5*”. Describe the update packets which spread in the network, and the reason for each one.

10_R4_SW.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ospf.msg.lsupdate and ip.dst == 224.0.0.5

No.	Time	Source	Destination	Protocol	Length	Info
97	20:26:47.400712	10.5.3.3	224.0.0.5	OSPF	134	LS Update
99	20:26:47.453157	10.5.3.3	224.0.0.5	OSPF	122	LS Update
101	20:26:47.474905	10.5.3.3	224.0.0.5	OSPF	242	LS Update
103	20:26:47.519105	10.5.3.3	224.0.0.5	OSPF	94	LS Update
104	20:26:47.561735	10.5.3.3	224.0.0.5	OSPF	94	LS Update
107	20:26:47.624068	10.5.3.3	224.0.0.5	OSPF	94	LS Update
109	20:26:47.655375	10.5.3.3	224.0.0.5	OSPF	94	LS Update

97: ראוטר 5 מעביר הודעת עדכון על השכנים של ראוטר 2 (בניהם ראוטר 3 שממהר מיד אחר כך להעביר את הבשורה)

```
Advertising Router: 10.5.0.2
Sequence Number: 0x80000001
Checksum: 0x75c0
Length: 72
> Flags: 0x00
Number of Links: 4
> Type: Transit ID: 10.5.10.1      Data: 10.5.10.2      Metric: 1
> Type: Transit ID: 10.5.6.1      Data: 10.5.6.2      Metric: 1
> Type: Transit ID: 10.5.2.2      Data: 10.5.2.1      Metric: 1
> Type: Transit ID: 10.5.1.1      Data: 10.5.1.2      Metric: 1
```

99: ראוטר 5 מעביר עדכון מראוטר 3 על השכנים החדשים שלו

```
Advertising Router: 10.5.0.3
Sequence Number: 0x80000004
Checksum: 0xc9a4
Length: 60
> Flags: 0x00
Number of Links: 3
> Type: Transit ID: 10.5.7.1      Data: 10.5.7.2      Metric: 1
> Type: Transit ID: 10.5.3.3      Data: 10.5.3.1      Metric: 1
> Type: Transit ID: 10.5.2.2      Data: 10.5.2.2      Metric: 1
```

101: כנ"ל מראוטר 7

```
Advertising Router: 10.5.0.7
Sequence Number: 0x80000004
Checksum: 0xcc8d
Length: 60
> Flags: 0x00
Number of Links: 3
> Type: Transit ID: 10.5.8.2      Data: 10.5.8.1      Metric: 1
> Type: Transit ID: 10.5.6.1      Data: 10.5.6.1      Metric: 1
> Type: Transit ID: 10.5.5.2      Data: 10.5.5.2      Metric: 1
```

וכן הלאה... משידועות המטריקות של השכנים לראוטר 4 יש מספיק מידע בשביל לנתב את עצמו באופן המהיר ביותר.

12. Bad News Propagation (Router Down)

Now, you **turn off a router** and see how *OSPF* reacts and how the network converges accordingly.

Do:

11.3. Start all the network components.

11.4. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened.

```
traceroute 192.5.1.1
```

```
traceroute to 192.5.1.1 (192.5.1.1), 30 hops max, 60 byte packets
 1 192.5.2.254 (192.5.2.254)  8.490 ms  19.271 ms  29.115 ms
 2 10.5.3.1 (10.5.3.1)      39.359 ms  50.470 ms  61.137 ms
 3 10.5.2.1 (10.5.2.1)      71.962 ms  82.699 ms  92.150 ms
 4 10.5.1.1 (10.5.1.1)     102.835 ms 113.690 ms 124.286 ms
 5 192.5.1.1 (192.5.1.1)   133.713 ms 144.331 ms 155.575 ms
root@pc2:~# _
```

11.5. Issue an infinite *ping* from *PC2* to *PC1*.

```
ping 192.5.1.1
```

11.6. Stop the router *R2*.

11.7. When the router stops, the ping fails. Wait for the network to converge until the ping returns to work properly. That will happen when an alternate path is found and may take several minutes.

11.8. Stop the ping command using **ctrl+c** and save the ping statistics

```
root@pc2:~# ping 192.5.1.1
PING 192.5.1.1 (192.5.1.1) 56(84) bytes of data.
64 bytes from 192.5.1.1: icmp_seq=1 ttl=60 time=81.4 ms
64 bytes from 192.5.1.1: icmp_seq=2 ttl=60 time=81.5 ms
64 bytes from 192.5.1.1: icmp_seq=3 ttl=60 time=83.6 ms
64 bytes from 192.5.1.1: icmp_seq=4 ttl=60 time=84.3 ms
64 bytes from 192.5.1.1: icmp_seq=45 ttl=58 time=119 ms
64 bytes from 192.5.1.1: icmp_seq=46 ttl=58 time=122 ms
64 bytes from 192.5.1.1: icmp_seq=47 ttl=58 time=124 ms
64 bytes from 192.5.1.1: icmp_seq=48 ttl=58 time=129 ms
64 bytes from 192.5.1.1: icmp_seq=49 ttl=58 time=111 ms
64 bytes from 192.5.1.1: icmp_seq=50 ttl=58 time=112 ms
^C
--- 192.5.1.1 ping statistics ---
50 packets transmitted, 10 received, 80% packet loss, time 49282ms
rtt min/avg/max/mdev = 81.456/105.028/129.299/18.861 ms
root@pc2:~# _
```

11.9. Using the ping statistics, calculate the time it took *OSPF* to converge.

אנו יודעים כי פינג נשלח בכל שניה, וכן ידוע לנו כי נשלחו עשרה פינגים לאורך 50 שניות. אזי שלא נשלחו פינגים במשך 40 שניות אז זה הזמן אשר לקח למערכת להתכנס מחדש.

11.10. Explain, what in the protocol supports the waiting time you calculated?

בשיטת OSPF בין הראוטרם נשלח hello בכל 10 שניות. dead interval הוא פרק זמן ארוך פי 4 כלומר 40 שניות שאם בהן ראוטר מסויים לא מעביר hello אז הוא בטח מת או הלך לישון או שהוא בעונש, בכל אופן צריך למצוא לו אלטרנטיבה. 40 שניות שקיבלנו להתכנסות כנראה בגלל שכיבינו את הראוטר ממש רגע לפני השליחה הבאה של ה hello שלו, וזה "קיצר" אז זמן האינטרוול המת. שכן ראינו שמספיקות לנו שניות מעטות מרגע הבנת הבעיה ועד להתכנסות בפועל.

13. Hierarchical Routing in OSPF

DO:

12.2. Start *Wireshark* on *R4* interface *f1/0*.

12.3. Enable *OSPF* on all routersConfigure the routers in this order: *R4, R5, R3, R6, R7, R2, R8, R1*. Note that the configuration of the border routers is a little different. Follow the examples.

In *OSPF*, networks can be divided into areas, to significantly reduce the amount of topological information routers have to learn. In *OSPF* all areas must be connected to *Area 0* which is known as the *backbone* area.

You will now define *Hierarchical OSPF* with two areas (*area 1* and *area 2*) connected through a *backbone* area (*area 0*). Routers that connect two areas are called *area border routers*.

R4 router inside area 1:

```
configure terminal
router ospf 1
router-id 10.5.0.4
network 10.5.0.0 0.0.255.255 area 1
end
```

R5 router inside area 1:

```
configure terminal
router ospf 1
router-id 10.5.0.5
network 10.5.0.0 0.0.255.255 area 1
end
```

R3 border router:

```
configure terminal
router ospf 1
router-id 10.5.0.3
network 10.5.3.0 0.0.0.255 area 1
network 10.5.2.0 0.0.0.255 area 0
network 10.5.7.0 0.0.0.255 area 0
end
```

R6 border router:

```
configure terminal
router ospf 1
router-id 10.5.0.6
network 10.5.4.0 0.0.0.255 area 1
network 10.5.5.0 0.0.0.255 area 0
network 10.5.7.0 0.0.0.255 area 0
end
```


R7 border router: <pre> configure terminal router ospf 1 router-id 10.5.0.7 network 10.5.5.0 0.0.0.255 area 0 network 10.5.6.0 0.0.0.255 area 0 network 10.5.8.0 0.0.0.255 area 2 end </pre>
R2 border router: <pre> configure terminal router ospf 1 router-id 10.5.0.2 network 10.5.2.0 0.0.0.255 area 0 network 10.5.6.0 0.0.0.255 area 0 network 10.5.1.0 0.0.0.255 area 2 network 10.5.10.0 0.0.0.255 area 2 end </pre>
R8 router inside area 2: <pre> configure terminal router ospf 1 router-id 10.5.0.8 network 10.5.0.0 0.0.255.255 area 2 end </pre>
R1 router inside area 2: <pre> configure terminal router ospf 1 router-id 10.5.0.1 network 10.5.0.0 0.0.255.255 area 2 end </pre>

12.6. Wait for the network convergence.

12.7. Stop capture and save the *wireshark pcap* to file.

12.8. You should now be able to ping any router from any other router. Test your network by using *ping* from some router to all the others.

R1	<i>ping 10.5.1.1</i>
R2	<i>ping 10.5.1.2</i>
R3	<i>ping 10.5.2.2</i>
R4	<i>ping 10.5.3.2</i>
R5	<i>ping 10.5.3.3</i>
R6	<i>ping 10.5.4.2</i>
R7	<i>ping 10.5.6.1</i>
R8	<i>ping 10.5.9.1</i>

12.9. Perform "write" on each router, save the topology (as *Topology 10*) and ZIP it.

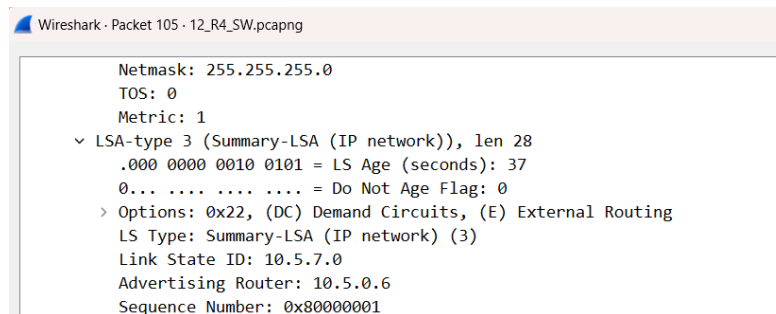
ענה על השאלות הבאות הדבק לדוח:

Look at the *Wireshark pcap* using the display filter:

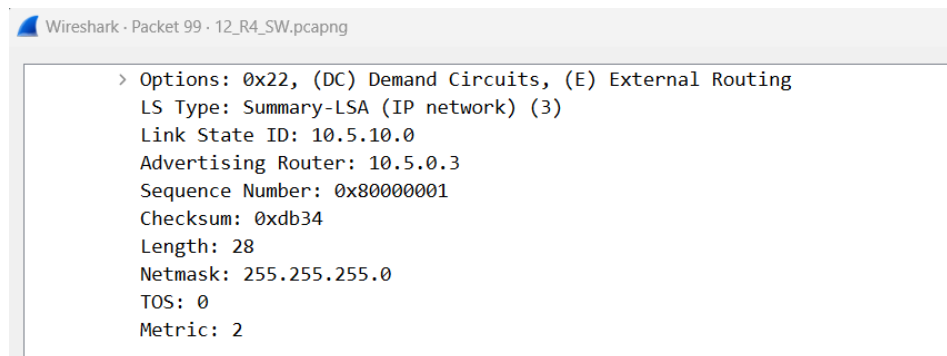
"*ospf.msg.lsupdate and ip.dst == 224.0.0.5*".

12.11. Using the *Wireshark pcap*, describe which information routers in area 1 received about area 2 and which information they received about the backbone area.

בפקטה הנ"ל ניתן לראות שראוטר 5 מקבל מידע אודות לינק 10.5.7.0 שאליו יש להגיע דרך ראוטר 6, הפקטה אינה מכילה מידע אודות קיומה של רשת 0.



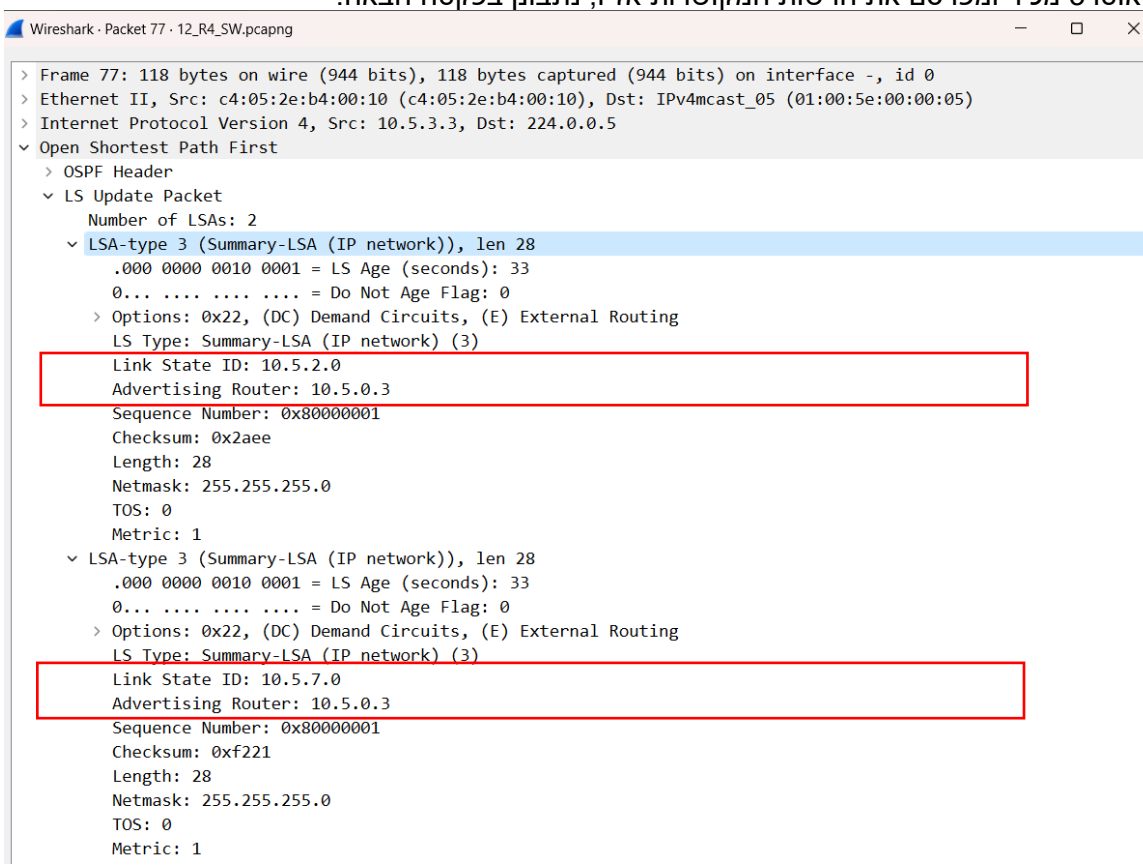
נסתכל על פקטה אחרת שמדווחת על לינק 10.5.10.3 השייך לאיזור 2, הפקטה מלמד כי אל הלינק יש להגיע דרך ראוטר 3 אך אינה מכירה את איזור 2 ואינה מכירה את הראוטרים השייכים אליו



נמצא כי ראוטרים אשר אינם שייכים לאותו אזור לא מכירים זה את זה או אזורים אחרים, הם יודעים רק דרך איזה ראוטר עליהם לגשת ללינקים אחרים.

12.12. Locate on the *Wireshark pcap* the *Router-LSA* update that *R3* advertised. Which interfaces it advertises? Why?

ראוטר 3 מכיר ומפרסם את הרשות המקושרות אליו, נתבון בפקטה הבאה:



Wireshark · Packet 77 · 12_R4_SW.pcapng

- > Frame 77: 118 bytes on wire (944 bits), 118 bytes captured (944 bits) on interface -, id 0
- > Ethernet II, Src: c4:05:2e:b4:00:10 (c4:05:2e:b4:00:10), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
- > Internet Protocol Version 4, Src: 10.5.3.3, Dst: 224.0.0.5
- > Open Shortest Path First
 - > OSPF Header
 - > LS Update Packet
 - Number of LSAs: 2
 - > LSA-type 3 (Summary-LSA (IP network)), len 28
 - .000 0000 0010 0001 = LS Age (seconds): 33
 - 0... = Do Not Age Flag: 0
 - > Options: 0x22, (DC) Demand Circuits, (E) External Routing
 - LS Type: Summary-LSA (IP network) (3)
 - Link State ID: 10.5.2.0
 - Advertising Router: 10.5.0.3
 - Sequence Number: 0x80000001
 - Checksum: 0x2aee
 - Length: 28
 - Netmask: 255.255.255.0
 - TOS: 0
 - Metric: 1
 - > LSA-type 3 (Summary-LSA (IP network)), len 28
 - .000 0000 0010 0001 = LS Age (seconds): 33
 - 0... = Do Not Age Flag: 0
 - > Options: 0x22, (DC) Demand Circuits, (E) External Routing
 - LS Type: Summary-LSA (IP network) (3)
 - Link State ID: 10.5.7.0
 - Advertising Router: 10.5.0.3
 - Sequence Number: 0x80000001
 - Checksum: 0xf221
 - Length: 28
 - Netmask: 255.255.255.0
 - TOS: 0
 - Metric: 1

הראוטר מפרסם את הרשות המחוברות אליו דרך ממשקים 1/0 ו 3/0

למה?

12.13. Display the *OSPF database* of R4 and compare it to the one you saved earlier, at exercise “Configuring OSPF on Cisco Routers”. Discuss the differences.

Configuring OSPF on Cisco Routers:

```

R4#show ip ospf database

        OSPF Router with ID (10.5.0.4) (Process ID 1)

          Router Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum Link count
10.5.0.1     10.5.0.1     127         0x80000005  0x000C9B  2
10.5.0.2     10.5.0.2     149         0x80000007  0x00B775  4
10.5.0.3     10.5.0.3     90          0x80000006  0x00C5A6  3
10.5.0.4     10.5.0.4     78          0x80000004  0x00F9E6  1
10.5.0.5     10.5.0.5     65          0x80000005  0x00F5AD  2
10.5.0.6     10.5.0.6     39          0x80000005  0x005C05  3
10.5.0.7     10.5.0.7     63          0x80000006  0x00DE78  3
10.5.0.8     10.5.0.8     45          0x80000005  0x005AEB  3

          Net Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum
10.5.1.2     10.5.0.2     149         0x80000003  0x006A89
10.5.2.2     10.5.0.3     90          0x80000003  0x00717E
10.5.3.3     10.5.0.5     65          0x80000003  0x00A928
10.5.4.2     10.5.0.6     39          0x80000003  0x009153
10.5.5.2     10.5.0.7     65          0x80000003  0x009848
10.5.6.2     10.5.0.2     151         0x80000003  0x008761
10.5.7.1     10.5.0.6     44          0x80000003  0x005E86
10.5.8.2     10.5.0.8     50          0x80000003  0x008951
10.5.9.2     10.5.0.1     134         0x80000003  0x007076
10.5.10.2    10.5.0.2     154         0x80000003  0x00697A

```

Hierarchical Routing in OSPF:

```

R4#show ip ospf database

        OSPF Router with ID (10.5.0.4) (Process ID 1)

          Router Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum Link count
10.5.0.3     10.5.0.3     869         0x80000003  0x0001E2  1
10.5.0.4     10.5.0.4     912         0x80000003  0x00FBE5  1
10.5.0.5     10.5.0.5     839         0x80000004  0x00F7AC  2
10.5.0.6     10.5.0.6     805         0x80000003  0x00EAF0  1

          Net Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum
10.5.3.3     10.5.0.5     839         0x80000003  0x00A928
10.5.4.2     10.5.0.6     805         0x80000002  0x009352

          Summary Net Link States (Area 1)

Link ID      ADV Router   Age         Seq#         Checksum
10.5.1.0     10.5.0.3     869         0x80000002  0x003DDA
10.5.1.0     10.5.0.6     805         0x80000002  0x0035DE
10.5.2.0     10.5.0.3     869         0x80000002  0x0028EF
10.5.2.0     10.5.0.6     805         0x80000002  0x0020F3
10.5.5.0     10.5.0.3     872         0x80000002  0x001103
10.5.5.0     10.5.0.6     808         0x80000002  0x00F41D
10.5.6.0     10.5.0.3     872         0x80000002  0x00060D
10.5.6.0     10.5.0.6     808         0x80000003  0x00F11D
10.5.7.0     10.5.0.3     872         0x80000002  0x00F022
10.5.7.0     10.5.0.6     808         0x80000002  0x00DE31
10.5.8.0     10.5.0.3     872         0x80000002  0x00F916
10.5.8.0     10.5.0.6     808         0x80000002  0x00DD30
10.5.9.0     10.5.0.3     872         0x80000002  0x00EE20
10.5.9.0     10.5.0.6     808         0x80000002  0x00DC2F
10.5.10.0    10.5.0.3     872         0x80000002  0x00D935
10.5.10.0    10.5.0.6     808         0x80000002  0x00D139

```

ההבדלים:

1. כאשר ה-OSPF הוא היררכי ראوتر4 אינו מכיר באופן אישי את כל הראוטרים ברשת, הוא מכיר רק את אלו שחולקים לינקים מהאיזור שלו.
2. הניתוב שהראوتر מכיר הוא עד לנקודה בה מתחלף האיזור, הראوتر מכיר את הניתוב לכל הלינקים עד הנקודה בה החבילה יוצאת מהאיזור שלו.