תרגיל ראשון

יובל מור, רועי פוקס

חלק א

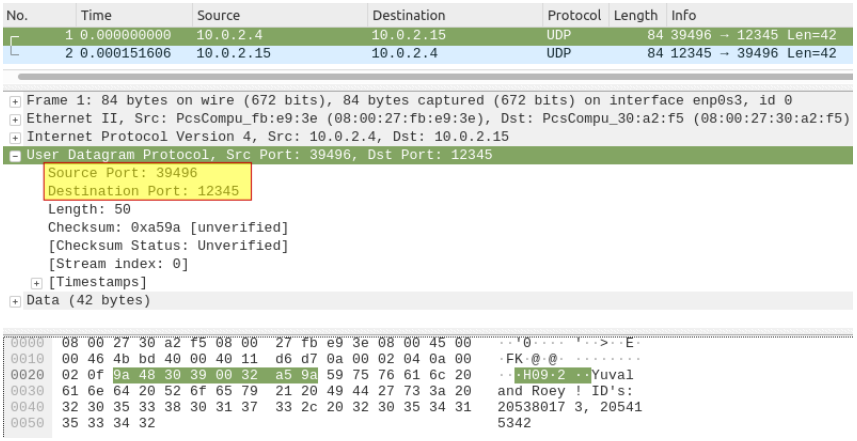
1. שינינו את הסקריפט שניתן כך שההודעה שתועבר תהיה עם השמות שלנו ומספרי הזהות. בנוסף, השתמשנו בפונקציות $encode()$ ו–$decode()$ במקום השימוש ב–b להפיכה לבייטים (והחזרה ע"י שימוש ב$str()$).

2. ביצענו את ההסנפה ממחשב השרת, לאחר הסינון נשארו שתי חבילות – הראשונה היא חבילה שהלקוח שולח לשרת, והשניה היא החבילה שהשרת שולח ללקוח.

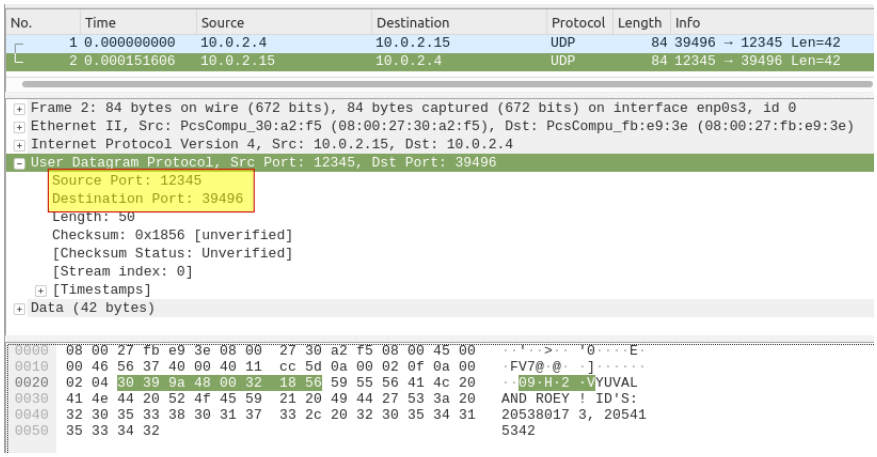
   סיננו את החבילות לפי ה–IP של הלקוח:

```
ip.dst==10.0.2.4 or ip.src==10.0.2.4
```

3. בקוד השרת נעשה bind לפורט מספר 12345, דהיינו השרת מבקש ממ"ה להשתמש בפורט זה. בצד הלקוח לא מבוצע bind ולכן מערכת ההפעלה תקצה לו פורט פנוי כלשהו. הפורט משמש כמזהה הסוקט המהווה צינור להעברת מידע לתוכנית, ונמצא בשכבת התעבורה, כפי שרואים בצילומי המסך הבאים:



From client to server



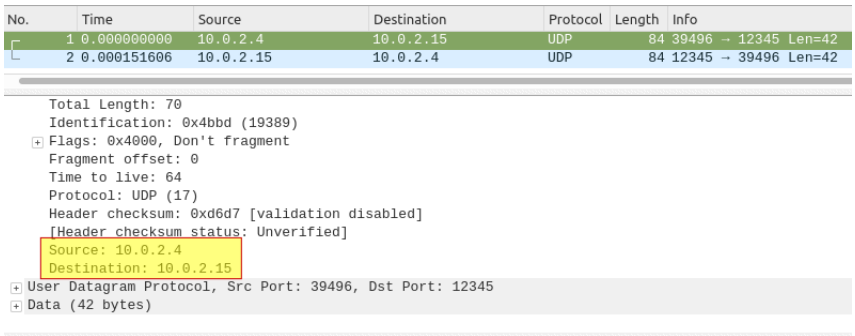
From server to client

אנחנו יכולים לראות כי הלקוח שולח הודעות מפורט 39496 לפורט 12345, ולאחר מכן השרת שולח תשובתו חזרה מ–12345 לפורט 39496. נשים לב, כי בקוד הלקוח, הגדרנו באופן מפורש את פורט השרת אליו אנו שולחים את החבילה.

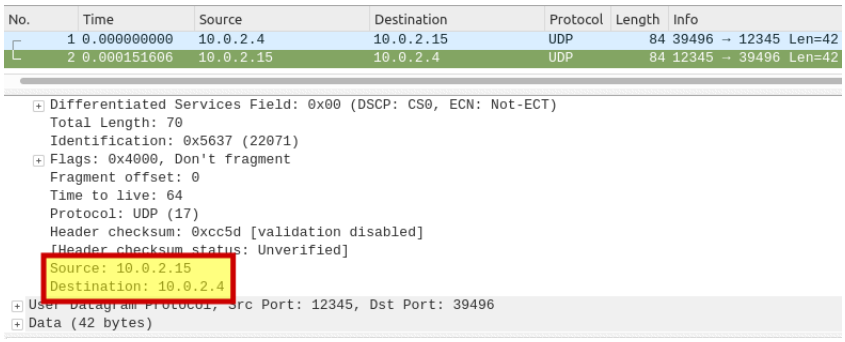
4. אנו מציגים את פלט ה–wireshark ממחשב השרת. כתובת השרת – 10.0.2.15, וכתובת הלקוח – 10.0.2.4.

הפלט הראשון, המראה את בקשת הלקוח, נשלח עם IP מקור של הלקוח ו– IP יעד של השרת, ובפלט השני, תשובת השרת, הפוך.

כתובות ה–IP המופיעות בפקודה $ifconfig$ זהות למה שהתקבל ב–wireshark. זה קורה כיוון ששני המחשבים נמצאים באותה רשת, ולכן אין שימוש בכתובות חיצוניות אלא רק בפנימיות.



From client to server



From server to client

חלק ב

השתמשנו בקבצי המיפויים שסופקו בהגדרת התרגיל, והרצנו לקוח על מכונה וירטואלית אחת, ושרת ושרת–אב במכונה וירטואלית שניה.

ביצענו את הבקשות הבאות:

- בקשת רשומה הנמצאת בקובץ המיפויים של השרת (שלבים $1, 2$).
- בקשת רשומה שאינה בקובץ המיפויים של השרת (שלבים $3, 4, 5, 6$).
- בקשת אותה רשומה שנלמדה, בטרם מעבר זמן ה–TTL (שלבים $7, 8$).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=9
2	0.000290847	10.0.2.15	10.0.2.4	UDP	65	12345 → 43113 Len=21
3	14.429221206	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
4	14.429380678	127.0.0.1	127.0.0.1	UDP	60	39392 → 12346 Len=16
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28
6	14.430066758	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28
7	21.027656018	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
8	21.028427843	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28

להלן פירוט ההרצה של הלקוח:

```
roey@roey-VB: ~/Desktop
roey@roey-VB:~$ cd Desktop/
roey@roey-VB:~/Desktop$ python3 ./client.py 10.0.2.15 12345
biu.ac.il
1.2.3.4
www.google.co.il
8.8.8.8
www.google.co.il
8.8.8.8
```

בשלב הראשון הלקוח, שנמצא בכתובת IP 10.0.2.4 ביקש לקבל את המידע מהשרת שנמצא בכתובת IP 10.0.2.15 ופורט 12345 עבור biu.ac.il.

החבילה שהלקוח שולח מורכבת מהשכבות הבאות:

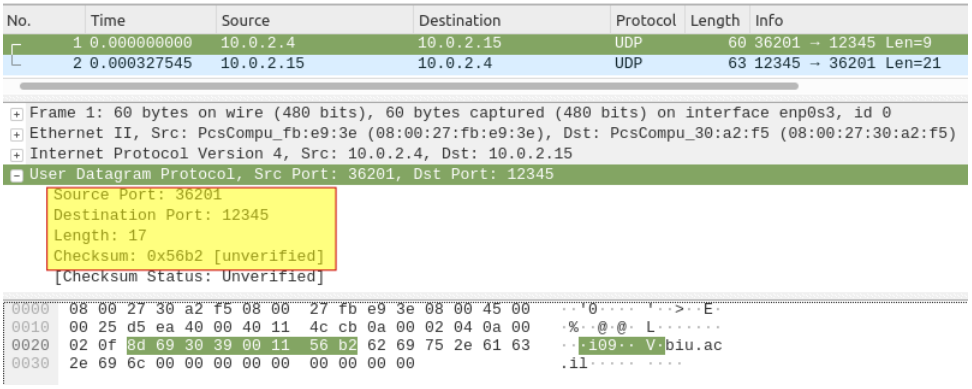
- **שכבת האפליקציה**, כוללת את המידע הגולמי שהלקוח/השרת מייצר. בדוגמה שלהלן רואים את הבקשה של הלקוח, הכוללת את הכתובת biu.ac.il.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	60	36201 → 12345 Len=9
2	0.000327545	10.0.2.15	10.0.2.4	UDP	63	12345 → 36201 Len=21

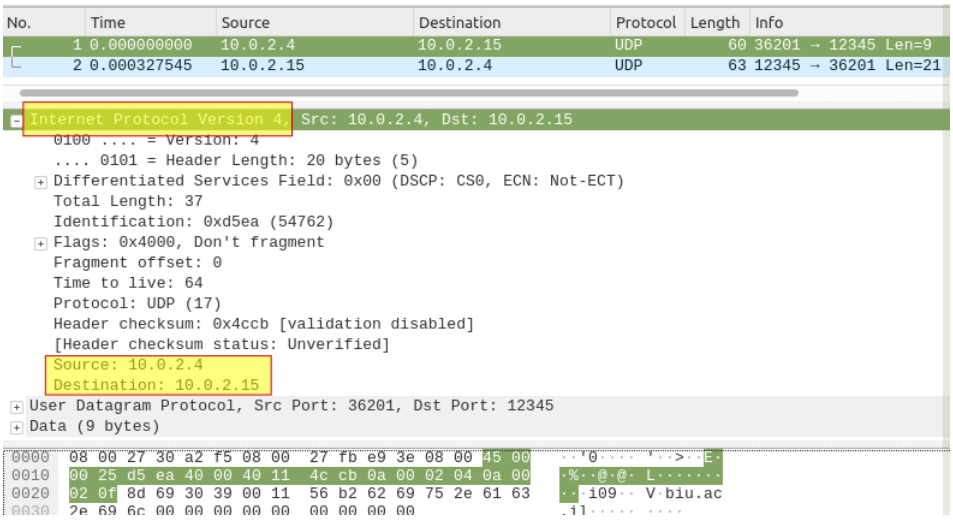
```
+ Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface enp0s3, id 0
+ Ethernet II, Src: PcsCompu_fb:e9:3e (08:00:27:fb:e9:3e), Dst: PcsCompu_30:a2:f5 (08:00:27:30:a2:f5)
+ Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.15
+ User Datagram Protocol, Src Port: 36201, Dst Port: 12345
- Data (9 bytes)
    Data: 6269752e61632e696c
    [Length: 9]

0000  08 00 27 30 a2 f5 08 00  27 fb e9 3e 08 00 45 00   ··'0····  '··>··E·
0010  00 25 d5 ea 40 00 40 11  4c cb 0a 00 02 04 0a 00   ·%··@·@·  L·******
0020  02 0f 8d 69 30 39 00 11  56 b2 62 69 75 2e 61 63   ···i09··  V·biu.ac
0030  2e 69 6c 00 00 00 00 00  00 00 00 00               .il·····  ····
```

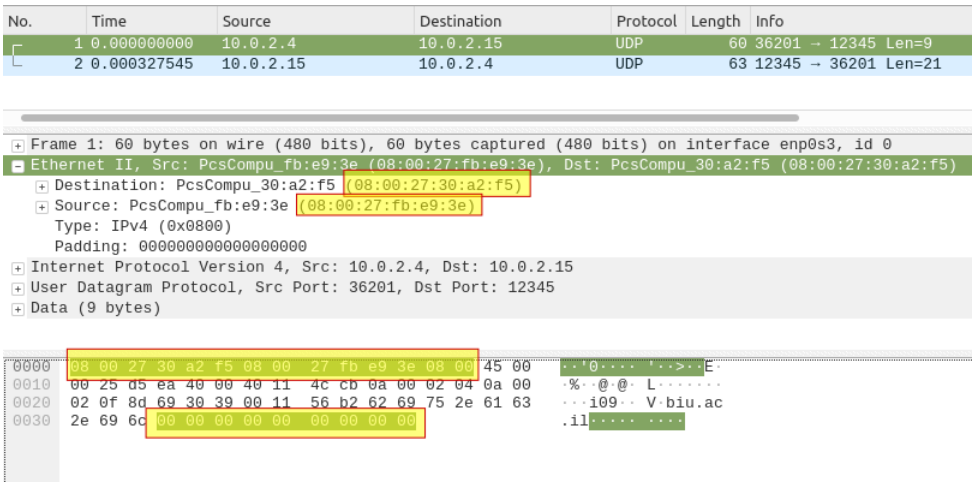
- **שכבת התעבורה**, כוללת בתוכה את הפורטים (מקור ויעד) שנועדו לסמן למ"ה לאיזה סוקט מיועד המידע. בדוגמה ניתן לראות כי פורט המקור הוא 36201 (נבחר אקראית ע"י מ"ה), ופורט היעד הוא 12345, כפי שהוגדר לתוכנית השרת כארגומנט. בנוסף, ניתן לראות כי הפרוטוקול שאנו משתמשים הוא פרוטוקול UDP ולכן מלבד מספרי הפורטים, הוא כולל בתוכו את שדות ה– length וה–checksum.



- **שכבת הרשת**, כוללת את כתובות ה–IP (מקור ויעד), המסמלות את הכתובות הלוגיות של המחשבים. בדוגמה שלנו, המחשבים יושבים באותה הרשת, ולכן נעשה שימוש בכתובות פנימיות. כתובת הלקוח הנה 10.0.2.4 וכתובת השרת היא 10.0.2.15. בנוסף, ניתן לראות כי הפרוטוקול הוא $IPv4$.



- **שכבת הערוץ**, כוללת בתוכה את הכתובות הפיזיות של המחשבים – MAC. בדוגמה שלנו ניתן לראות כי כתובת ה– MAC של הלקוח הנה 08:00:27:fb:e9:e3 ושל השרת הנה 08:00:27:30:a2:f5.
בנוסף, ניתן לראות כי השכבה מוסיפה מידע, משני צידיה של החבילה (trailer, header).



בשלב השני, השרת מחזיר את החבילה הבאה:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	60	36201 → 12345 Len=9
2	0.000327545	10.0.2.15	10.0.2.4	UDP	63	12345 → 36201 Len=21

```
+ Frame 2: 63 bytes on wire (504 bits), 63 bytes captured (504 bits) on interface enp0s3, id 0
+ Ethernet II, Src: PcsCompu_30:a2:f5 (08:00:27:30:a2:f5), Dst: PcsCompu_fb:e9:3e (08:00:27:fb:e9:3e)
+ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.4
+ User Datagram Protocol, Src Port: 12345, Dst Port: 36201
- Data (21 bytes)
    Data: 6269752e61632e696c2c312e322e332e342c313830
    [Length: 21]
```

```
0000  08 00 27 fb e9 3e 08 00  27 30 a2 f5 08 00 45 00   ··'··>·· '0····E·
0010  00 31 2d d1 40 00 40 11  f4 d8 0a 00 02 0f 0a 00   ·1-·@·@· ........
0020  02 04 30 39 8d 69 00 1d  18 41 62 69 75 2e 61 63   ··09·i·· ·Abiu.ac
0030  2e 69 6c 2c 31 2e 32 2e  33 2e 34 2c 31 38 30       .il,1.2. 3.4,180
```

החבילה מורכבת מאותן שכבות שהראנו, אך כוללת את השינויים הבאים: כתובות ה־ IP, port, MAC התחלפו בין מקור ויעד, ובשכבת האפליקציה אנו רואים את תשובת השרת (כתובת אינטרנט, כתובת IP ו–TTL) לבקשה שהתקבלה.

נשים לב, כי לשרת יש את המידע עבור biu.ac.il בקובץ שקיבל, ולכן לא היה צריך לפנות לשרת האב והחזיר במיידית את התשובה.

בשלב השלישי, הלקוח מבקש את ה–IP של הכתובת www.google.co.il.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=9
2	0.000290847	10.0.2.15	10.0.2.4	UDP	65	12345 → 43113 Len=21
3	14.429221206	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16

```
+ Frame 3: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface any, id 0
+ Linux cooked capture
+ Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.15
+ User Datagram Protocol, Src Port: 43113, Dst Port: 12345
- Data (16 bytes)
    Data: 7777772e676f6f676c652e636f2e696c
```

```
0000  00 00 00 01 00 06 08 00  27 fb e9 3e 00 00 08 00   ........ '··>····
0010  45 00 00 2c 15 08 40 00  40 11 0d a7 0a 00 02 04   E··,··@· @·......
0020  0a 00 02 0f a8 69 30 39  00 18 d6 28 77 77 77 2e   ·····i09 ···(www.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c 00 00          google.c o.il··
```

בשלב הרביעי, כיוון שהכתובת אינה נמצאת אצל השרת הוא פונה בבקשה לקבלת המידע לשרת האב.

ניתן לראות בתמונה כי גם כתובת היעד וגם כתובת המקור היא 127.0.0.1. זאת כיוון ששרת האב והשרת הרגיל מורצים על אותה מכונה. נשים לב כי הפורט השולח הוא פורט שנבחר באקראי, שכן בקוד שכתבנו לשרת יש סוקט נפרד המשמש לתקשורת עם שרת האב, ואילו הפורט של שרת האב (פורט היעד) הוא 12346, שהוזן אליו כארגומנט.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=9
2	0.000290847	10.0.2.15	10.0.2.4	UDP	65	12345 → 43113 Len=21
3	14.429221206	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
4	14.429380678	127.0.0.1	127.0.0.1	UDP	60	39392 → 12346 Len=16
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28

```
+ Frame 4: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface any, id 0
+ Linux cooked capture
+ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
+ User Datagram Protocol, Src Port: 39392, Dst Port: 12346
- Data (16 bytes)
    Data: 7777772e676f6f676c652e636f2e696c
    [Length: 16]
```

```
0000  00 00 03 04 00 06 00 00  00 00 00 00 00 00 08 00   ........ ........
0010  45 00 00 2c 34 06 40 00  40 11 08 b9 7f 00 00 01   E··,4·@· @·......
0020  7f 00 00 01 99 e0 30 3a  00 18 fe 2b 77 77 77 2e   ......0: ···+www.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c                google.c o.il
```

בשלב החמישי, שרת האב מחזיר תשובה לשרת הרגיל, בפורמט שנקבע (שם, כתובת IP ו–TTL). שוב, כיוון ששניהם נמצאים על אותה המכונה, שרת האב יחזיר תשובה לכתובת ה–IP 127.0.0.1. מספרי הפורטים יהיו כפי שהיו בעת שהשרת שאל את שרת האב שאלה, אך בהיפוך תפקיד (מקור הופך ליעד ויעד למקור).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=9
2	0.000290847	10.0.2.15	10.0.2.4	UDP	65	12345 → 43113 Len=21
3	14.429221206	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
4	14.429380678	127.0.0.1	127.0.0.1	UDP	60	39392 → 12346 Len=16
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28
6	14.430066758	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28

```
Frame 5: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface any, id 0
Linux cooked capture
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
User Datagram Protocol, Src Port: 12346, Dst Port: 39392
Data (28 bytes)
```

```
0000  00 00 03 04 00 06 00 00  00 00 00 00 00 00 08 00   ........ ........
0010  45 00 00 38 34 07 40 00  40 11 08 ac 7f 00 00 01   E··84·@· @·......
0020  7f 00 00 01 30 3a 99 e0  00 24 fe 37 77 77 77 2e   ····0:·· ·$·7www.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c 2c 38 2e 38   google.c o.il,8.8
0040  2e 38 2e 38 2c 33 30 30                            .8.8,300
```

בשלב השישי, לאחר שהשרת מקבל את תשובת שרת האב, הוא יחזיר את התשובה ללקוח. ניתן לראות שמדובר בשרת האב ובלקוח לפי כתובות ה–IP.

No.	Time	Source	Destination	Protocol	Length	Info
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28
6	14.430066758	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28
7	21.027656018	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
8	21.028427843	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28

```
Frame 6: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface any, id 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.4
User Datagram Protocol, Src Port: 12345, Dst Port: 43113
Data (28 bytes)
    Data: 7777772e676f6f676c652e636f2e696c2c382e382e382e38…
    [Length: 28]
```

```
0000  00 04 00 01 00 06 08 00  27 30 a2 f5 00 00 08 00   ........ '0......
0010  45 00 00 38 24 87 40 00  40 11 fe 1b 0a 00 02 0f   E··8$·@· @·......
0020  0a 00 02 04 30 39 a8 69  00 24 18 48 77 77 77 2e   ····09·i ·$·Hwww.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c 2c 38 2e 38   google.c o.il,8.8
0040  2e 38 2e 38 2c 33 30 30                            .8.8,300
```

בשלב השביעי, לפני שעוברות 300 השניות (ה–TTL של הרשומה) הלקוח מבקש אותה בשנית (www.google.co.il)

No.	Time	Source	Destination	Protocol	Length	Info
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28
6	14.430066758	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28
7	21.027656018	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
8	21.028427843	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28

```
+ Frame 7: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface any, id 0
+ Linux cooked capture
+ Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.15
+ User Datagram Protocol, Src Port: 43113, Dst Port: 12345
- Data (16 bytes)
    Data: 7777772e676f6f676c652e636f2e696c
    [Length: 16]
+ VSS Monitoring Ethernet trailer, Source Port: 0

0000  00 00 00 01 00 06 08 00  27 fb e9 3e 00 00 08 00   ........ '..>....
0010  45 00 00 2c 1b 6a 40 00  40 11 07 45 0a 00 02 04   E..,.j@. @..E....
0020  0a 00 02 0f a8 69 30 39  00 18 d6 28 77 77 77 2e   .....i09 ...(www.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c 00 00         google.c o.il..
```

בשלב השמיני, השרת מחזיר מיידית את התשובה, ללא פניה לשרת האב משום שלא חלף הזמן שהוגדר ב–TTL.

No.	Time	Source	Destination	Protocol	Length	Info
5	14.429505349	127.0.0.1	127.0.0.1	UDP	72	12346 → 39392 Len=28
6	14.430066758	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28
7	21.027656018	10.0.2.4	10.0.2.15	UDP	62	43113 → 12345 Len=16
8	21.028427843	10.0.2.15	10.0.2.4	UDP	72	12345 → 43113 Len=28

```
+ Frame 8: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface any, id 0
+ Linux cooked capture
+ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.4
+ User Datagram Protocol, Src Port: 12345, Dst Port: 43113
- Data (28 bytes)
    Data: 7777772e676f6f676c652e636f2e696c2c382e382e382e38…
    [Length: 28]

0000  00 04 00 01 00 06 08 00  27 30 a2 f5 00 00 08 00   ........ '0......
0010  45 00 00 38 29 1b 40 00  40 11 f9 87 0a 00 02 0f   E..8).@. @.......
0020  0a 00 02 04 30 39 a8 69  00 24 18 48 77 77 77 2e   ....09.i .$.Hwww.
0030  67 6f 6f 67 6c 65 2e 63  6f 2e 69 6c 2c 38 2e 38   google.c o.il,8.8
0040  2e 38 2e 38 2c 33 30 30                            .8.8,300
```