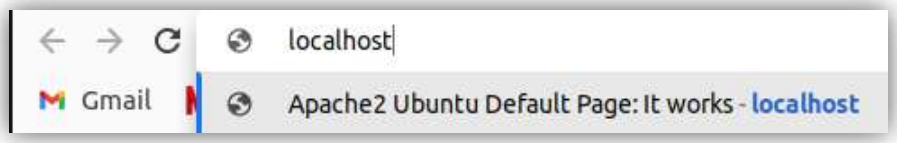
התקנו את תוכנת השרת הנקראת Apache על ידי הפקודה הנתונה בתרגיל.

```
jonathan@mousehouse:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-keybinder-3.0 libflashrom1 libftdi1-2 libkeybinder-3.0-0 python3-configobj
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2
0 upgraded, 1 newly installed, 0 to remove and 39 not upgraded.
Need to get 97.9 kB of archives.
After this operation, 546 kB of additional disk space will be used.
Get:1 http://il.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2 amd64 2.4.52-1ubuntu4.2 [97.9 kB]
Fetched 97.9 kB in 2s (60.5 kB/s)
```

בדקנו ואכן הצלחנו על ידי שימוש בדפדפן לגלוש לאתר שנמצא בכתובת של-local host.



התוצאה נמצאת בתמונה הנ"ל:

נכנסנו לתיקייה באמצעות ה- terminal. פתחנו את הקובץ באמצעות vs code וביצענו שינוי טקסטואלי – במקום שהכותרת תהיה "Apache2 Default Page" שינינו אותה ל-"Hemi King". כמו שניתן לראות בתמונות המצורפות, אכן קיבלנו את התוצאה.

```
jonathan@mousehouse:~$ cd /var/www/html
jonathan@mousehouse:/var/www/html$ ls
index.html
```

```
style="width:184px;height:146px;"
<div>
  <span style="margin-top: 1.5em;" cla
    Apache2 Default Page
  </span>
</div>
<div class="banner">
  <div id="about"></div>
  It works!
```

```
style="width:184px;height:146px;"
<div>
  <span style="margin-top: 1.5em;" cla
    <!-- Apache2 Default Page -->
    Hemi King
  </span>
</div>
<div class="banner">
  <div id="about"></div>
```

פתחנו מה terminal את Wireshark על הרשאות sudo:

```
jonathan@mousehouse:~$ sudo wireshark
[sudo] password for jonathan:
 ** (wireshark:1372343) 16:27:30.043292 [GUI WARNING] -- QStandardPaths: XDG_RUN
TIME_DIR not set, defaulting to '/tmp/runtime-root'
```

הסנפנו את התעבורה תוך כדי שפתחנו את השרת וקיבלנו את הנתונים הבאים:

127.0.0.1	127.0.0.53	DNS	85	Standard query 0xfdc9 A www.g
127.0.0.1	127.0.0.53	DNS	85	Standard query 0x8bcf AAAA w
127.0.0.53	127.0.0.1	DNS	113	Standard query response 0x8bc
127.0.0.53	127.0.0.1	DNS	349	Standard query response 0xfdc
127.0.0.1	127.0.0.1	TCP	74	60882 → 80 [SYN] Seq=0 Win=65
127.0.0.1	127.0.0.1	TCP	74	80 → 60882 [SYN, ACK] Seq=0 A
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [ACK] Seq=1 Ack=1
127.0.0.1	127.0.0.1	HTTP	505	GET / HTTP/1.1
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=1 Ack=44
127.0.0.1	127.0.0.1	HTTP	3528	HTTP/1.1 200 OK (text/html)
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [ACK] Seq=440 Ack=
127.0.0.1	127.0.0.53	DNS	101	Standard query 0x1c23 A incom
127.0.0.53	127.0.0.1	DNS	527	Standard query response 0x1c2
127.0.0.1	127.0.0.1	HTTP	447	GET /icons/ubuntu-logo.png HT
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=3463 Ack
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [FIN, ACK] Seq=821
127.0.0.1	127.0.0.1	TCP	74	60890 → 80 [SYN] Seq=0 Win=65
127.0.0.1	127.0.0.1	TCP	74	80 → 60890 [SYN, ACK] Seq=0 A
127.0.0.1	127.0.0.1	TCP	66	60890 → 80 [ACK] Seq=1 Ack=1
127.0.0.1	127.0.0.1	HTTP	437	GET /favicon.ico HTTP/1.1
127.0.0.1	127.0.0.1	TCP	66	80 → 60890 [ACK] Seq=1 Ack=37
127.0.0.1	127.0.0.1	HTTP	554	HTTP/1.1 404 Not Found (text
127.0.0.1	127.0.0.1	TCP	66	60890 → 80 [ACK] Seq=372 Ack=
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=3463 Ack
127.0.0.1	127.0.0.1	HTTP	3673	HTTP/1.1 200 OK (PNG)
127.0.0.1	127.0.0.1	TCP	54	60882 → 80 [RST] Seq=822 Win=

סיננו את התעבורה כדי להתבונן רק במידע הרלוונטי באמצעות ה filter הבא:

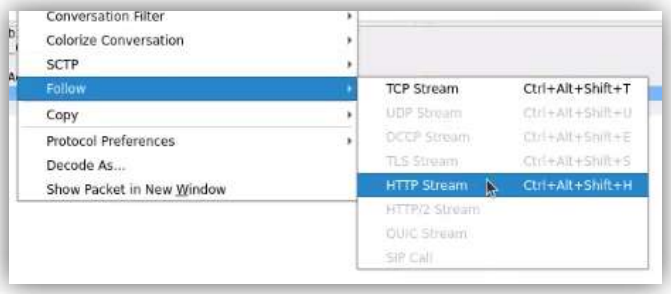
```
tcp && (tcp.port == 80)
```

ואכן כמו שציפינו קיבלנו רק את המידע הרלוונטי:

127.0.0.1	127.0.0.1	TCP	74	60882 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495
127.0.0.1	127.0.0.1	TCP	74	80 → 60882 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSv
127.0.0.1	127.0.0.1	HTTP	505	GET / HTTP/1.1
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=1 Ack=440 Win=65152 Len=0 T
127.0.0.1	127.0.0.1	HTTP	3528	HTTP/1.1 200 OK (text/html)
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [ACK] Seq=440 Ack=3463 Win=63232 Len=
127.0.0.1	127.0.0.1	HTTP	447	GET /icons/ubuntu-logo.png HTTP/1.1
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=3463 Ack=821 Win=65280 Len=
127.0.0.1	127.0.0.1	TCP	66	60882 → 80 [FIN, ACK] Seq=821 Ack=3463 Win=65536
127.0.0.1	127.0.0.1	TCP	74	60890 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495
127.0.0.1	127.0.0.1	TCP	74	80 → 60890 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=
127.0.0.1	127.0.0.1	TCP	66	60890 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSv
127.0.0.1	127.0.0.1	HTTP	437	GET /favicon.ico HTTP/1.1
127.0.0.1	127.0.0.1	TCP	66	80 → 60890 [ACK] Seq=1 Ack=372 Win=65152 Len=0 T
127.0.0.1	127.0.0.1	HTTP	554	HTTP/1.1 404 Not Found (text/html)
127.0.0.1	127.0.0.1	TCP	66	60890 → 80 [ACK] Seq=372 Ack=489 Win=65152 Len=0
127.0.0.1	127.0.0.1	TCP	66	80 → 60882 [ACK] Seq=3463 Ack=822 Win=65536 Len=
127.0.0.1	127.0.0.1	HTTP	3673	HTTP/1.1 200 OK (PNG)
127.0.0.1	127.0.0.1	TCP	54	60882 → 80 [RST] Seq=822 Win=0 Len=0

לפני שנתעמק בתעבורה מנקודת מבט של פרוטוקול HTTP, נתאר את התעבורה בקווים כללים:
שלוש החבילות הראשונות מתארות לנו את לחיצת היד המשולשת שמאפיינת את תחילת התקשורת בפרוטוקול TCP (דיברנו רבות על כך בתרגיל הקודם ולכן לא נפרט יותר). בחבילה הבאה, הלקוח (הדפדפן) מבקש מהשרת (Apache) את המשאב \ באמצעות הודעת GET ובהמשך יקבל את הקובץ index.html. השרת כמובן מאשר שקיבל את בקשת הלקוח, ולאחר מכן מחזיר הודעת 200 OK אליה מצרף את תוכן הקובץ. הלקוח מחזיר ack על שקיבל את תוכן הקובץ. לבסוף, התקשורת נחתמת על ידי שליחת הודעת סיום מטעם הלקוח (הודעה עם דגל ה-FIN דלוק) שמתלווה בתשובה והודעת סיום מצד השרת. לבסוף הלקוח מאשר שקיבל את הודעת הסיום של השרת וכך נסגר החיבור. לאחר מכן נפתח חיבור חדש ובו נשלח ה icon לאתר (התמונה שרואים למעלה בקטן כאשר פותחים אתר). בסוף גם הוא נסגר וזה סוף התעבורה הנוכחית.

נעבור על התוכן שנשלח על גבי פרוטוקול ה HTTP:



ונקבל את התוכן הבא:

```
GET / HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:108.0) Gecko/20100101 Firefox/108.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
If-Modified-Since: Fri, 13 Jan 2023 13:31:59 GMT
If-None-Match: "29a4-5f2254206b70a-gzip"

HTTP/1.1 200 OK
Date: Fri, 13 Jan 2023 14:27:05 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Fri, 13 Jan 2023 13:31:59 GMT
ETag: "29a4-5f2254206b70a-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 3123
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966004
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
```

החלק באדום הוא ההודעה שהלקוח שלח לשרת (הודעת GET ואז הקובץ המבוקש) והחלק בכחול הוא התגובה של השרת ללקוח (הודעת 200 OK ותוכן הדף).

נביט בבקשת הלקוח:

השורה הראשונה היא שורת הבקשה – בקשה מסוג GET. הלקוח מבקש מהשרת משאב כלשהו (למעשה בפועל במקרה שלנו הוא מבקש את הקובץ index.html כמו שתיארנו לעיל). שאר שורות הבקשה הן תחיליות, מטרתן לתת פרטים נוספים אודות השפה, הדפדפן, הקידוד וכו'.

למשל השורה:

User -Agent: Mozilla/5.0 (X11; Ubuntu; Linux ×86_64; rv: 108.0) Gecko/20100101 Firefox/108.0

נותנת מידע אודות הדפדפן. כמו שניתן לראות במקרה שלנו השתמשנו ב Firefox.

או השורה:

Accept-Encoding: gzip, deflate, br

נותנת מידע על שיטת/אלגוריתם הדחיסה איתן הדפדפן יודע להתמודד. כמו שניתן לראות במקרה שלנו הוא יודע להשתמש בשיטה שנקראת GZIP,deflate וגם ב br.

נשים לב כי יש בסוף \n\r\n\r. כך יודע השרת כי נגמרה ההודעה וכי אין תחיליות נוספות.

נביט בתגובת השרת:

בשורה הראשונה נתון ה Status code. תחילה HTTP/1.1 מתאר את הפרוטוקול ואת הגרסה שלו (יש צורך בכך כי השרת יכול להחזיר בגרסה שונה אולי ממה שהלקוח ביקש). בהמשך השורה כתוב 200 OK, דבר זה מסמל שהבקשה הצליחה. כמו מקודם, גם בתגובת השרת, השורות הבאות עד ירידת השורה הכפולה הבאה הן תחיליות.

למשל השורה:

Server: Apache/2.4.52 (Ubuntu)

מצוין סוג השרת והגרסה שלו, במקרה שלנו סוג השרת הוא Apache בגרסת 2.4.52 והוא רץ על מערכת הפעלה Ubuntu.

או השורות:

Content-Encoding: gzip
Content- Length: 3123

שמציינים את סוג הקידוד ואת האורך של התוכן בהתאמה.

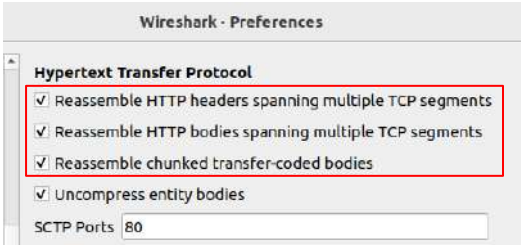
```
<div>
  <span style="margin-top: 1.5em;" class="float
    Hemi King
  </span>
</div>
```

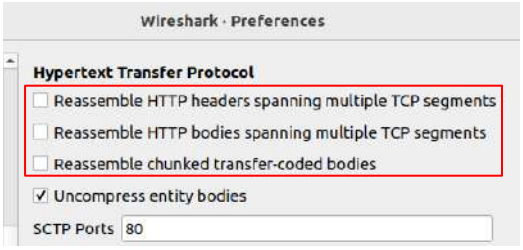
דרך אגב נשים לב כי בתוכן ההודעה ניתן לראות את השינוי שעשינו בשלב מוקדם יותר של התרגיל ובו כתוב "**חמי מלך**" (כן, נשמח מאוד לבונוס 😜)

ניגשנו למחשב שבו רץ השרת ובדקנו מה כתובת ה IP שלו:

```
amit@amit-System-Product-Name:~$ ifconfig
enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.100.102.40  netmask 255.255.255.0  broadcast 10.100.102.255
        inet6 fe80::2da6:31af:d59a:6f63  prefixlen 64  scopeid 0x20<link>
        ether 88:d7:f6:c8:84:72  txqueuelen 1000  (Ethernet)
        RX packets 290247  bytes 294201624 (294.2 MB)
        RX errors 0  dropped 4  overruns 0  frame 0
```

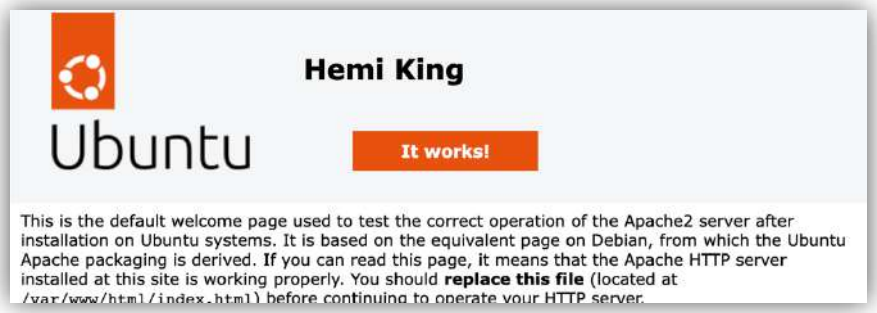
בנוסף גם שינינו את ההגדרות של Wireshark כדי שנוכל לראות את התעבורה פר חבילה ובלי ש Wireshark יעשה לנו resassemble לחבילות:





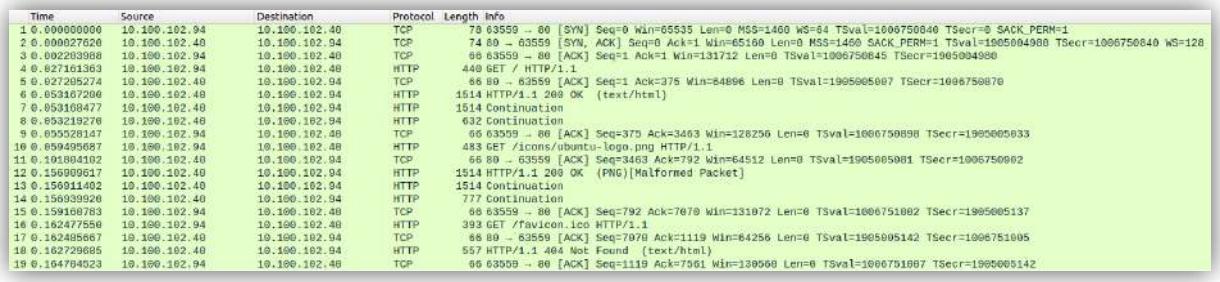
נגשנו דרך הלקוח במחשב השני על ידי שכתבנו את כתובת ה IP בדפדפן:

וקיבלנו את הדף הבא:



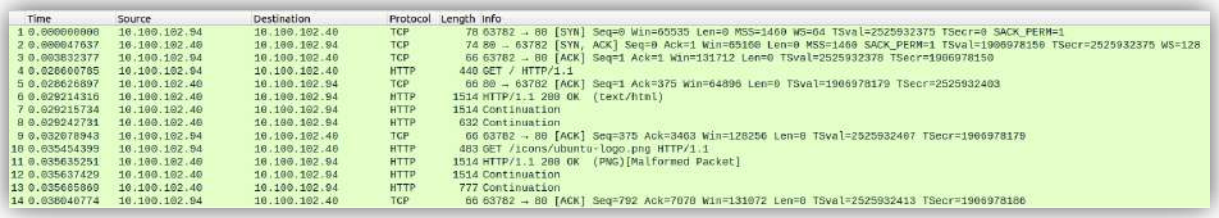
נשים לב שגם ב Macbook (המחשב שעליו הרצנו את הלקוח) חמי עדיין מלך.

הסנפנו את התעבורה ב Wireshark במחשב שמריץ את השרת וקיבלנו את התעבורה הבאה:



	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.100.102.94	10.100.102.40	TCP	78	63559 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1006750840 TSecr=0 SACK_PERM=1
2	0.000027620	10.100.102.40	10.100.102.94	TCP	74	80 → 63559 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1905004980 TSecr=1006750840 WS=128
3	0.002203988	10.100.102.94	10.100.102.40	TCP	66	63559 → 80 [ACK] Seq=1 Ack=1 Win=131712 Len=0 TSval=1006750845 TSecr=1905004980
4	0.027161363	10.100.102.94	10.100.102.40	HTTP	440	GET / HTTP/1.1
5	0.027205274	10.100.102.40	10.100.102.94	TCP	66	80 → 63559 [ACK] Seq=1 Ack=375 Win=64896 Len=0 TSval=1905005007 TSecr=1006750870
6	0.053167200	10.100.102.40	10.100.102.94	HTTP	1514	HTTP/1.1 200 OK (text/html)
7	0.053168477	10.100.102.40	10.100.102.94	HTTP	1514	Continuation
8	0.053219276	10.100.102.40	10.100.102.94	HTTP	632	Continuation
9	0.055528147	10.100.102.94	10.100.102.40	TCP	66	63559 → 80 [ACK] Seq=375 Ack=3463 Win=128256 Len=0 TSval=1006750898 TSecr=1905005033
10	0.059495687	10.100.102.94	10.100.102.40	HTTP	483	GET /icons/ubuntu-logo.png HTTP/1.1
11	0.101804102	10.100.102.40	10.100.102.94	TCP	66	80 → 63559 [ACK] Seq=3463 Ack=792 Win=64512 Len=0 TSval=1905005081 TSecr=1006750902
12	0.156909617	10.100.102.40	10.100.102.94	HTTP	1514	HTTP/1.1 200 OK (PNG)[Malformed Packet]
13	0.156911402	10.100.102.40	10.100.102.94	HTTP	1514	Continuation
14	0.156939920	10.100.102.40	10.100.102.94	HTTP	777	Continuation
15	0.159168783	10.100.102.94	10.100.102.40	TCP	66	63559 → 80 [ACK] Seq=792 Ack=7070 Win=131072 Len=0 TSval=1006751002 TSecr=1905005137
16	0.162477550	10.100.102.94	10.100.102.40	HTTP	393	GET /favicon.ico HTTP/1.1
17	0.162485667	10.100.102.40	10.100.102.94	TCP	66	80 → 63559 [ACK] Seq=7070 Ack=1119 Win=64256 Len=0 TSval=1905005142 TSecr=1006751005
18	0.162729685	10.100.102.40	10.100.102.94	HTTP	557	HTTP/1.1 404 Not Found (text/html)
19	0.164704523	10.100.102.94	10.100.102.40	TCP	66	63559 → 80 [ACK] Seq=1119 Ack=7561 Win=130560 Len=0 TSval=1006751007 TSecr=1905005142

הרצנו פעם נוספת בדיוק כמו בפעם הקודמת אך הפעם פתחנו בחלון גלישה "נסתרת" וקיבלנו את הנ"ל:



	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.100.102.94	10.100.102.40	TCP	78	63782 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2525932375 TSecr=0 SACK_PERM=1
2	0.000047637	10.100.102.40	10.100.102.94	TCP	74	80 → 63782 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1906978150 TSecr=2525932375 WS=128
3	0.003832377	10.100.102.94	10.100.102.40	TCP	66	63782 → 80 [ACK] Seq=1 Ack=1 Win=131712 Len=0 TSval=2525932378 TSecr=1906978150
4	0.028600785	10.100.102.94	10.100.102.40	HTTP	440	GET / HTTP/1.1
5	0.028626897	10.100.102.40	10.100.102.94	TCP	66	80 → 63782 [ACK] Seq=1 Ack=375 Win=64896 Len=0 TSval=1906978179 TSecr=2525932403
6	0.029214316	10.100.102.40	10.100.102.94	HTTP	1514	HTTP/1.1 200 OK (text/html)
7	0.029215734	10.100.102.40	10.100.102.94	HTTP	1514	Continuation
8	0.029242731	10.100.102.40	10.100.102.94	HTTP	632	Continuation
9	0.032078943	10.100.102.94	10.100.102.40	TCP	66	63782 → 80 [ACK] Seq=375 Ack=3463 Win=128256 Len=0 TSval=2525932407 TSecr=1906978179
10	0.035454399	10.100.102.94	10.100.102.40	HTTP	483	GET /icons/ubuntu-logo.png HTTP/1.1
11	0.035635251	10.100.102.40	10.100.102.94	HTTP	1514	HTTP/1.1 200 OK (PNG)[Malformed Packet]
12	0.035637429	10.100.102.40	10.100.102.94	HTTP	1514	Continuation
13	0.035685860	10.100.102.40	10.100.102.94	HTTP	777	Continuation
14	0.038040774	10.100.102.94	10.100.102.40	TCP	66	63782 → 80 [ACK] Seq=792 Ack=7070 Win=131072 Len=0 TSval=2525932413 TSecr=1906978186

נשים לב כי יש קצת הבדלים בין חלון גלישה בסתר ולבין חיבור רגיל. את ההבדלים האלו נסביר בהרחבה יותר מאוחר (לאחר שנסיים להסביר את ההבדל בין הריצה על מחשבים שונים לבין הריצה על אותו מחשב).

ההבדלים בין התעבורה כאשר נכנסו דרך אותו מחשב לבין כאשר נכנסנו דרך מחשבים שונים:
נשים לב לדבר עיקרי בו ניתן להבחין כאשר נכנסנו דרך מחשבים שונים, חלק מבקשות ה HTTP חולקו לחלקים שונים

12	0.156909617	10.100.102.40	10.100.102.94	HTTP	1514 HTTP/1.1 200 OK (PNG)[Malformed Packet]
13	0.156911402	10.100.102.40	10.100.102.94	HTTP	1514 Continuation
14	0.156939920	10.100.102.40	10.100.102.94	HTTP	777 Continuation

הסיבה לכך היא שהשתמשנו במחשבים שונים וגם שרשמנו מקודם ב Terminal את השורה הבאה:

```
ethtool -K enp3s0 tx off sg off tso off
```

נרצה להבין מדוע השורה הזאת גרמה לחלוקת החבילה. נכתוב את השורה הבאה:

```
ethtool -k enp3s0
```

ונקבל את הפלט:

```
tcp-segmentation-offload: off
        tx-tcp-segmentation: off
        tx-tcp-ecn-segmentation: off [fixed]
        tx-tcp-mangleid-segmentation: off
        tx-tcp6-segmentation: off
```

נזכר שבהרצאה דיברנו על כך ש TCP "רוצה" להימנע מפרגמנטציה ולכן הוא מבצע בעצמו סגמנטציה. בעת הפעלת הפקודה הקודמת בעצם ביטלנו את האפשרות הזאת. לכן החבילות מתפצלות לפרגמנטים שונים ואלו בעצם החבילות השונות אותם אנחנו רואים ב Wireshark. הסיבה לכך שזה קרה דווקא במחשבים השונים זה מפני שהתרחש תהליך של ניתוב והחבילה הייתה גדולה מידי לעבור במלואה ברשת הביתית.

הבדל נוסף הינו כתובות ה IP שכן בפעם הקודמת השרת והלקוח היו בכתובת 127.0.0.1 ועכשיו לא, אבל זה דבר טריוויאלי בשלב הזה של הקורס ולכן לא נפרט יותר מידי. עוד הבדל טריוויאלי כמובן הוא שמספר הפורט של הלקוח השתנה, שכן הלקוח מקבל פורט שיכול להשתנות ממערכת ההפעלה ולכן בהרצה אחרת הגיוני שנקבל פורט שונה.

נשים לב לעוד דבר מעניין, כאשר התחברנו משני מחשבים שונים, היה חיבור אחד בין השרת ללקוח. כאשר התחברנו מאותו מחשב היו שני חיבורים (כמו שכתבנו מקודם, אחד לאתר ואחד בקובץ האייקון).

נדבר עכשיו גם על ההבדלים בתוכן עצמו של ההודעות בשכבת האפליקציה:

אותו מחשב

Protocol	Length	Info
TCP	74	49788 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval
TCP	74	80 → 49788 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_P
TCP	66	49788 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=4124494335 TSec
HTTP	597	GET / HTTP/1.1
TCP	66	80 → 49788 [ACK] Seq=1 Ack=532 Win=65024 Len=0 TSval=4124494335 TS
HTTP	3528	HTTP/1.1 200 OK (text/html)
TCP	66	49788 → 80 [ACK] Seq=532 Ack=3463 Win=63232 Len=0 TSval=4124494336
TCP	66	49788 → 80 [FIN, ACK] Seq=532 Ack=3463 Win=65536 Len=0 TSval=41244
TCP	66	80 → 49788 [FIN, ACK] Seq=3463 Ack=533 Win=65536 Len=0 TSval=41244
TCP	66	49788 → 80 [ACK] Seq=533 Ack=3464 Win=65536 Len=0 TSval=4124499337

V

S

מחשבים שונים

Protocol	Length	Info
TCP	78	63559 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1006750840 TSecr=0 SACK_PERM=1
TCP	74	80 → 63559 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1905004980 TSecr=1006750840 WS=128
TCP	66	63559 → 80 [ACK] Seq=1 Ack=1 Win=131712 Len=0 TSval=1006750845 TSecr=1905004980
HTTP	440	GET / HTTP/1.1
TCP	66	80 → 63559 [ACK] Seq=1 Ack=375 Win=64896 Len=0 TSval=1905005007 TSecr=1006750870
HTTP	1514	HTTP/1.1 200 OK (text/html)
HTTP	1514	Continuation
HTTP	632	Continuation
TCP	66	63559 → 80 [ACK] Seq=375 Ack=3463 Win=128256 Len=0 TSval=1006750898 TSecr=1905005033
HTTP	483	GET /icons/ubuntu-logo.png HTTP/1.1
TCP	66	80 → 63559 [ACK] Seq=3463 Ack=792 Win=64512 Len=0 TSval=1905005081 TSecr=1006750902
HTTP	1514	HTTP/1.1 200 OK (PNG)[Malformed Packet]
HTTP	1514	Continuation
HTTP	777	Continuation
TCP	66	63559 → 80 [ACK] Seq=792 Ack=7070 Win=131072 Len=0 TSval=1006751002 TSecr=1905005137
HTTP	393	GET /favicon.ico HTTP/1.1
TCP	66	80 → 63559 [ACK] Seq=7070 Ack=1119 Win=64256 Len=0 TSval=1905005142 TSecr=1006751005
HTTP	557	HTTP/1.1 404 Not Found (text/html)
TCP	66	63559 → 80 [ACK] Seq=1119 Ack=7561 Win=130560 Len=0 TSval=1006751007 TSecr=1905005142

ההבדל הראשון הינו בתחילית Host, כאשר הרצנו על אותו מחשב ה Host היה localhost (מאוד הגיוני כי השרת נמצא באותו מחשב כמובן) וכאשר הרצנו על מחשבים שונים ה Host היה 10.100.102.40 שכן זה כתובת ה IP שבה רץ השרת.

אותו מחשב:

```
GET / HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (
Accept: text/html,applica
Accept-Language: en-US,en
```

מחשבים שונים:

```
GET / HTTP/1.1
Host: 10.100.102.40
Upgrade-Insecure-Request
Accept: text/html,applic
User-Agent: Mozilla/5.0
```

הבדלים שנובעים משימוש בדפדפנים שונים:

ההבדל השני הינו בתחילית User-Agent. כאשר הרצנו על אותו מחשב, היה כתוב בתחילית הזאת שהדפדפן בו אנחנו משתמשים הינו Firefox וכאשר הרצנו על מחשבים שונים הדפדפן היה Safari. זה כמובן תואם את הדרך שבא פתחנו את האתר.

אותו מחשב:

```
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:108.0) Gecko/20100101 Firefox/108.0
```

מחשבים שונים:

```
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.2 Safari/605.1.15
```

הבדל נוסף הוא בתחילית Accept. תחילית זאת בעצם מצביעה על כל הטיפוסים מסוג MIME שהלקוח יודע לקבל. נשים לב ש - Firefox יודע לקבל יותר דברים מ - Safari וזה מאוד הגיוני שכן Safari נבנה על ידי אפל.

אותו מחשב:

```
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
```

מחשבים שונים:

```
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
```

נתבונן בעוד הבדל, הפעם בתחילית Accepted-Encoding. נראה כי Firefox יודע לקבל קידוד שכווץ על ידי פורמט gzip, deflate וגם ע"י br בעוד ש Safari יודע לקבל קידוד שכווץ רק על ידי פורמט Gzip וגם Deflate.

אותו מחשב:

```
Accept-Encoding: gzip, deflate, br
```

מחשבים שונים:

```
Accept-Encoding: gzip, deflate
```

גם בתחילית של השפות יש הבדלים קטנים. כאשר השתמשנו במחשבים שונים אנחנו הלקוח יודע להבין את השפה האנגלית בארצות הברית בדיוק כמו במקרה השני אך הוא גם יודע להבין את האנגלית של הבריטים.

אותו מחשב:

```
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
```

מחשבים שונים:

```
Accept-Language: en-US,en;q=0.5
```

נשים לב שכאשר השתמשנו במחשבים שונים, אנחנו לא רואים בכלל את התחיליות הבאות שכן הופיעו כאשר הרצנו על אותו מחשב:

```
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
```

כצפוי בתגובת השרת השינוי היחיד הינו תאריך וזמן השליחה של ההודעה. שאר הדברים לא השתנו וגם אין סיבה שישתנו – אנחנו שולחים את אותו אתר ולכן קובץ ה HTML ישאר זהה וכך גם כל הגדרות השרת.

נעבור לדבר על ההבדלים כאשר קראנו לשרת ממצב גלישה "נסתרת" וכאשר קראנו לשרת מכרטיסיה רגילה: עקרונית, במקרה שלנו אין הרבה הבדלים גדולים בין הריצה במצב גלישה בסתר ובין הגלישה הרגילה. הדבר היה לנו די מוזר שכן לא סתם אמרו לנו להסביר את ההבדלים. לכן בדקנו קצת באינטרנט וגילינו שזה כנראה מפני ש Safari בצורה דיפולטיבית חוסם לנו Cookies ועוד כל מיני אפשרויות כאלו.

הבדל אחד שכן היה בין גלישה "נסתרת" לגלישה רגילה זה שבגלישה נסתרת, הלקוח לא ביקש את ה – favicon.ico. בדקנו את המקרה וגילינו שהסיבה לכך היא שלמרות שאנחנו במצב גלישה "נסתרת", עדיין פתחנו את האתר לפני מחלון רגיל. כאשר עשינו את זה התמונה favicon.ico נשמרה ב cache של הדפדפן. כאשר נכנסנו דרך חלון הגלישה בסתר, הדפדפן ראה שיש לו את התמונה ב cache וטען אותה משם.

חזרנו למכונה עליה מותקן שרת האינטרנט והתקנו שרת DNS באופן הבא:

```
jonathan@mousehouse:~$ sudo apt install bind9
[sudo] password for jonathan:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  bind-doc resolvconf
The following NEW packages will be installed:
  bind9
0 upgraded, 1 newly installed, 0 to remove and 47 not upgraded.
Need to get 251 kB of archives.
After this operation, 928 kB of additional disk space will be used.
Get:1 http://il.archive.ubuntu.com/ubuntu jammy-updates/main amd64 bind9 amd64 1:9.18.1-1ubuntu1.2 [251 kB]
Fetched 251 kB in 6s (39.5 kB/s)
Selecting previously unselected package bind9.
(Reading database ... 378307 files and directories currently installed.)
Preparing to unpack .../bind9_1%3a9.18.1-1ubuntu1.2_amd64.deb ...
Unpacking bind9 (1:9.18.1-1ubuntu1.2) ...
Setting up bind9 (1:9.18.1-1ubuntu1.2) ...
Adding group `bind' (GID 142) ...
Done.
Adding system user `bind' (UID 131) ...
Adding new user `bind' (UID 131) with group `bind' ...
Not creating home directory `/var/cache/bind'.
wrote key file "/etc/bind/rndc.key"
named-resolvconf.service is a disabled or a static unit, not starting it.
Created symlink /etc/systemd/system/bind9.service → /lib/systemd/system/named.se
rvice.
Created symlink /etc/systemd/system/multi-user.target.wants/named.service → /llb
/systemd/system/named.service.

Processing triggers for man-db (2.10.2-1) ...#######.......................]
Processing triggers for ufw (0.36.1-4build1) ...
```

בתום ההורדה, נכנסנו לתיקייה etc/bind כדי לעדכן את הגדרות השרת.

```
jonathan@mousehouse:~$ cd /etc/bind
jonathan@mousehouse:/etc/bind$ code .
```

עדכנו את הקובץ named.conf.options כך שבכל פעם שישאלו את השרת שאילתת DNS, הוא יעביר אותה לשרת DNS של גוגל, יקבל את התשובה בחזרה מגוגל, ישמור אותה ב cache שלו ואז יחזיר את התשובה למי ששאל אותו.

```
// If your ISP provided one or more IP addresses for stable
// nameservers, you probably want to use them as forwarders.
// Uncomment the following block, and insert the addresses replacing
// the all-0's placeholder.

// forwarders {
//      0.0.0.0;
// };

//========================================================================
// If BIND logs error messages about the root key being expired,
// you will need to update your keys.  See https://www.isc.org/bind-keys
//========================================================================
dnssec-validation auto;

listen-on-v6 { any; };
```

```
// If your ISP provided one or more IP addresses for stable
// nameservers, you probably want to use them as forwarders.
// Uncomment the following block, and insert the addresses replacing
// the all-0's placeholder.

forwarders {
    8.8.8.8;
    8.8.4.4;
};

//========================================================================
// If BIND logs error messages about the root key being expired,
// you will need to update your keys.  See https://www.isc.org/bind-keys
//========================================================================
dnssec-validation no;

listen-on-v6 { any; };
```

שמרנו את השינויים, ואתחלנו את השרת כדי שהשינויים יכנסו לתוקף על ידי:

```
jonathan@mousehouse:~$ sudo service bind9 restart
```

לאחר מכן, הגדרנו את המחשב שלנו לעבוד מול שרת ה – DNS. נכנסנו לקובץ etc/resolv.conf ושם הגדרנו את כתובת ה IP של שרת ה DNS שלנו כ nameserver . בנוסף, השארנו את הכתובת הקודמת בהערה כדי שנוכל לחזור אליה בתום המשימה.

```
jonathan@mousehouse:~$ code /etc/resolv.conf
```

```
# through the symlink at /etc/resolv.conf. To
# different way, replace this symlink by a st
#
# See man:systemd-resolved.service(8) for det
# operation for /etc/resolv.conf.

nameserver 192.168.1.30 # 127.0.0.53
options edns0 trust-ad
search .
```

```
# different way, replace this symlink by a s
#
# See man:systemd-resolved.service(8) for de
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
options edns0 trust-ad
search .
```

כעת, השרת היה מוכן לתפעול. התחלנו להסניף את התעבורה בWireShark, ותוך כדי ביצענו nslookup ל-github.com - פעולה המצריכה ביצוע DNS.

```
jonathan@mousehouse:~$ nslookup github.com
Server:         192.168.1.30
Address:        192.168.1.30#53

Non-authoritative answer:
Name:   github.com
Address: 140.82.121.4
```

להלן תוצאות ההסנפה:

1	0.000000000	192.168.1.49	192.168.1.49	DNS	72	Standard query 0xe929 A github.com
2	0.003511356	192.168.1.49	8.8.8.8	DNS	95	Standard query 0x72e5 A github.com OPT
3	0.064828809	8.8.8.8	192.168.1.49	DNS	99	Standard query response 0x72e5 A github.com A 140.82.121.4 OPT
4	0.066664743	192.168.1.49	192.168.1.49	DNS	88	Standard query response 0xe929 A github.com A 140.82.121.4
5	0.068715131	192.168.1.49	192.168.1.49	DNS	72	Standard query 0xfb26 AAAA github.com
6	0.069878378	192.168.1.49	8.8.4.4	DNS	95	Standard query 0x79fa AAAA github.com OPT
7	0.177906102	8.8.4.4	192.168.1.49	DNS	148	Standard query response 0x79fa AAAA github.com SOA dns1.p08.nsone.net OPT
8	0.181168124	192.168.1.49	192.168.1.49	DNS	146	Standard query response 0xfb26 AAAA github.com SOA dns1.p08.nsone.net

כפי שניתן לראות, שתי החבילות הראשונות שנתפסו בהסנפה הן שאילתת ה-DNS. ראשית, שאילתת ה-DNS נשלחה מהמחשב שלנו אל שרת ה-DNS המוגדר במחשב. אבל כאמור, הגדרנו את שרת זה להיות השרת שהורדנו. לכן, הבקשה נשלחת מהמחשב שלנו לעצמו! לאחר מכן, שרת ה-DNS שהגדרנו במחשב מעביר את הבקשה לשרת של גוגל בכתובת 8.8.8.8. למעשה, המחשב שלנו מתפקד כאן כריזולבר.

בחבילה השלישית שרת ה-DNS של גוגל מחזיר את התגובה המתאימה לשאילתה, ושולח אותה לשרת ה-DNS על המחשב שלנו (הריזולבר). נבחין כי בפירוט של של התגובה ב-WireShark, מתואר המצב הבא:

```
▾ Domain Name System (response)
    Transaction ID: 0x72e5
  ▸ Flags: 0x8190 Standard query response, No error
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 1
  ▸ Queries
  ▾ Answers
    ▸ github.com: type A, class IN, addr 140.82.121.4
  ▸ Additional records
    [Request In: 2]
    [Time: 0.061309444 seconds]
```

קיבלנו מהשרתים של גוגל את הכתובת של github.com, בגרסה IPv4.

לאחר מכן, בחבילה הרביעית שרת ה-DNS המקומי שלנו מחזיר את כתובת זו ללקוח (שנמצא על המחשב שלנו).

בחבילה החמישית הלקוח שוב מבקש את הכתובת ש github.com רק הפעם בגרסה IPv6. הבקשה נשלחת מהלקוח (המחשב שלנו) לשרת ה DNS המקומי (שגם הוא רץ על המחשב שלנו):

```
▾ Domain Name System (query)
    Transaction ID: 0xfb26
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▾ Queries
    ▾ github.com: type AAAA, class IN
        Name: github.com
        [Name Length: 10]
        [Label Count: 2]
        Type: AAAA (IPv6 Address) (28)
        Class: IN (0x0001)
    [Response In: 8]
```

החבילה השישית דומה לחבילה השנייה, וגם כאן השרת המקומי שלנו (הריזולבר) מעביר את שאילתת ה-DNS של הלקוח לשרת של גוגל, רק הפעם אל השרת ב-8.8.4.4:

6	0.069878378	192.168.1.49	8.8.4.4

בחבילה השביעית השרתים של גוגל מחזירים את הכתובת של github.com בגירסה IPv6 אל השרת המקומי, וכעת כאשר בידו התשובות, בחבילה השמינית שרת ה-DNS שלנו מחזיר את הכתובות שקיבל אל הלקוח (כלומר - אלינו):

7	0.177966102	8.8.4.4	192.168.1.49	DNS	148 Standard query response 0x79fa AAAA github.com SOA dns1.p08.nsone.net OPT
8	0.181108124	192.168.1.49	192.168.1.49	DNS	146 Standard query response 0xfb26 AAAA github.com SOA dns1.p08.nsone.net

נבצע שוב את ה-nslookup ל-github.com:

```
jonathan@mousehouse:~$ nslookup github.com
Server:         192.168.1.30
Address:        192.168.1.30#53

Non-authoritative answer:
Name:   github.com
Address: 140.82.121.4
```

להלן התוצאות:

1	0.000000000	192.168.1.49	192.168.1.49	DNS	72	Standard query 0x072e A github.com
2	0.000329641	192.168.1.49	192.168.1.49	DNS	88	Standard query response 0x072e A github.com A 140.82.121.3
3	0.001880607	192.168.1.49	192.168.1.49	DNS	72	Standard query 0xaab0 AAAA github.com
4	0.002208478	192.168.1.49	192.168.1.49	DNS	146	Standard query response 0xaab0 AAAA github.com SOA dns1.p08.nsone.net

כפי שניתן לראות, הפעם התקשורת לוקלית לחלוטין. הלקוח (כלומר, אנחנו) פונים לשרת ה-DNS המקומי שהגדרנו על המחשב, והוא שולח אלינו את התשובה ישירות. הוא לא פונה לשרתים של גוגל תוך כדי למצוא את הכתובות. אכן - הסיבה לכך היא שהשרת ענה מה-cache, הוא לא היה צריך לפנות לגוגל!

כדי לוודא זאת, ייצאנו את הרשומות השמורות ב-cache לפני ואחרי ה-nslookup הראשון.

```
jonathan@mousehouse:~$ sudo rndc dumpdb -cache
[sudo] password for jonathan:
```

לפני ה-nslookup הראשון ה-cache היה ריק, שכן לא בוצעו אף שאילתות DNS:

```
; Unassociated entries
;
;   8.8.4.4 [srtt 48005] [flags 00004000] [edns 1/0] [plain 0/0] [udpsize 512] [ttl 1799]
;   8.8.8.8 [srtt 29] [flags 00000000] [edns 0/0] [plain 0/0] [ttl 1799]
;
; Bad cache
;
;
; SERVFAIL cache
;
;
; Start view _bind
;
;
; Cache dump of view '_bind' (cache _bind)
;
; using a 0 second stale ttl
$DATE 20230114175216
;
; Address database dump
;
; [edns success/timeout]
; [plain success/timeout]
;
;
; Unassociated entries
;
;
; Bad cache
;
;
; SERVFAIL cache
;
; Dump complete
```

לעומת זאת, לאחר ה-nslookup הראשון, התווספה רשומה עבור github.com ל-cache:

```
; answer
github.com.     52  A   140.82.121.3
```

התחלנו להפוך את שרת ה-DNS להיות שרת DNS אוטורטיבי. תחילה, ערכנו את הקובץ etc/bind/named.conf.local:

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "biu.ac.il" {
type master;
file "/etc/bind/db.biu.ac.il";
};
```

לאחר מכן, הרצנו את הפקודה הבאה לביצוע שכפול התבנית אל הקובץ החדש שלנו:

```
jonathan@mousehouse:~$ sudo cp /etc/bind/db.local /etc/bind/db.biu.ac.il
```

את הקובץ החדש שיצרנו שינינו בהתאם להוראות שניתנו, וקיבלנו:

```
  GNU nano 6.4                                    db.biu.ac.il *
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     biu.ac.il. root.biu.ac.il. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns.biu.ac.il.
www.biu.ac.il IN A 192.168.1.49
@       IN      MX      10 mail.biu.ac.il
mail.biu.ac.il IN A     192.168.1.49
ns      IN      A       192.168.1.49
@       IN      AAAA    ::1

File Name to Write: db.biu.ac.il
^G Help          M-D DOS Format    M-A Append        M-B Backup File
^C Cancel        M-M Mac Format    M-P Prepend       ^T Browse
```

שמרנו את השינויים, ואתחלנו את השרת עם:

```
jonathan@mousehouse:~$ sudo systemctl restart bind9.service
```

כעת, בתום הכנת השרת התפננו לתפעל אותו. השתמשנו ב-nslookup כדי לבצע שאילתת DNS מול השרת שהגדרנו:

```
jonathan@mousehouse:~$ nslookup -type=ns biu.al.il 192.168.1.49
```

תוך כדי, הסנפנו את התעבורה:

→	1	0.000000000	192.168.1.49	192.168.1.49	DNS	71 Standard query 0xf0b6 NS biu.ac.il
←	2	0.000127206	192.168.1.49	192.168.1.49	DNS	104 Standard query response 0xf0b6 NS biu.ac.il NS ns.biu.ac.il A 192.168.1.49

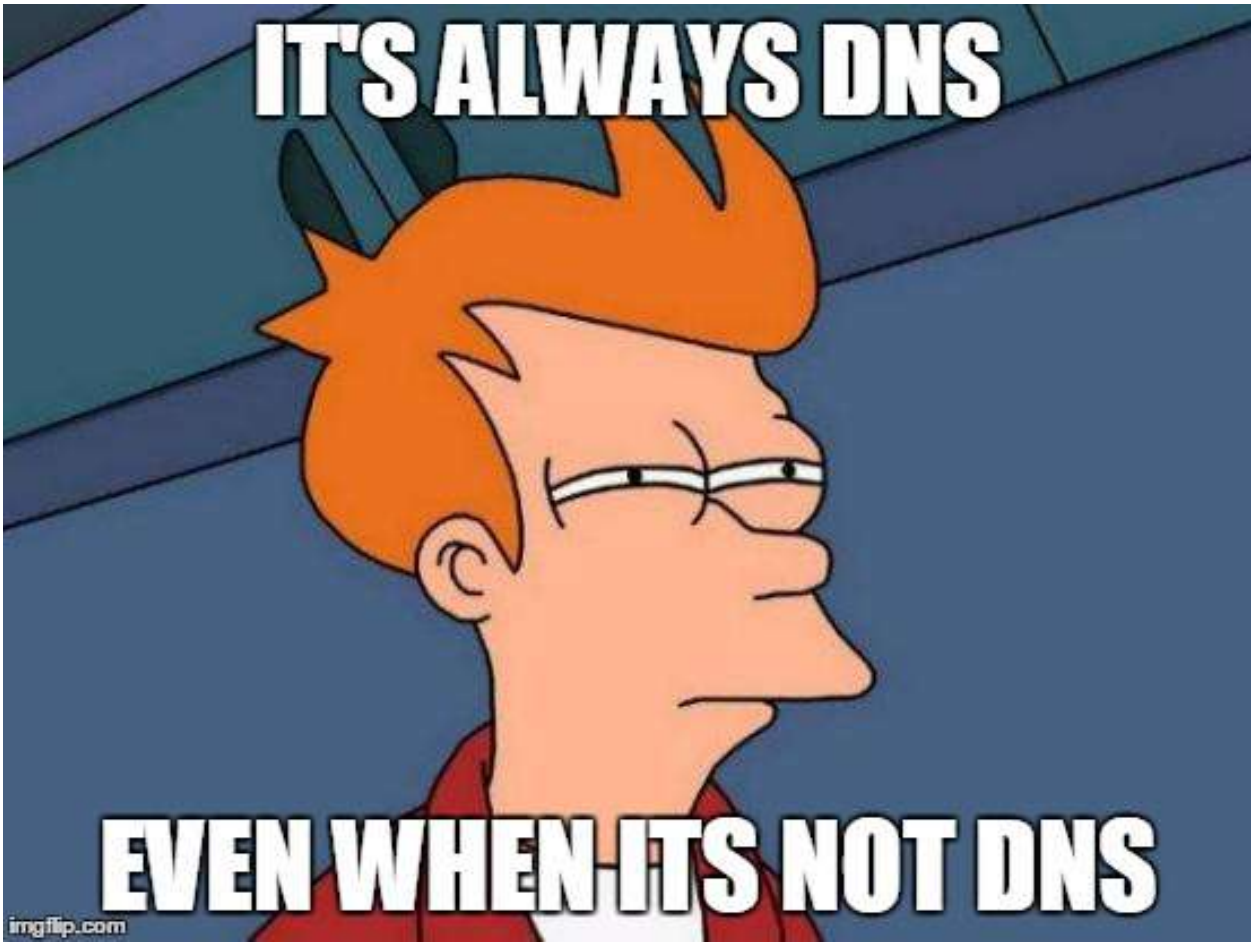
בחבילה הראשונה נשלחת שאילתת ה-DNS.

```
▾ Domain Name System (query)
    Transaction ID: 0xf0b6
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▾ Queries
    ▸ biu.ac.il: type NS, class IN
    [Response In: 2]
```

ניתן לראות כי בחבילה השנייה שרת ה-DNS המקומי עונה כי ה-NS של biu.ac.il הוא ns.biu.ac.il. כמו כן, תחת "Additional Records" מצורפת כתובת ה-IP.

```
▾ Domain Name System (response)
    Transaction ID: 0xf0b6
  ▸ Flags: 0x8580 Standard query response, No error
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 1
  ▸ Queries
  ▾ Answers
    ▸ biu.ac.il: type NS, class IN, ns ns.biu.ac.il
  ▾ Additional records
    ▸ ns.biu.ac.il: type A, class IN, addr 192.168.1.49
    [Request In: 1]
    [Time: 0.000127206 seconds]
```

תחת Answers מצורפת התשובה לשאילתה - שרת ה-NS. הסיבה שכתובת ה-IP של השרת מצורפת תחת Additional Records היא שלא ביקשנו אותה באופן מפורש. אכן, התשובות מתאימות לקונפיגרציות של השרת אותן הגדרנו מקודם.



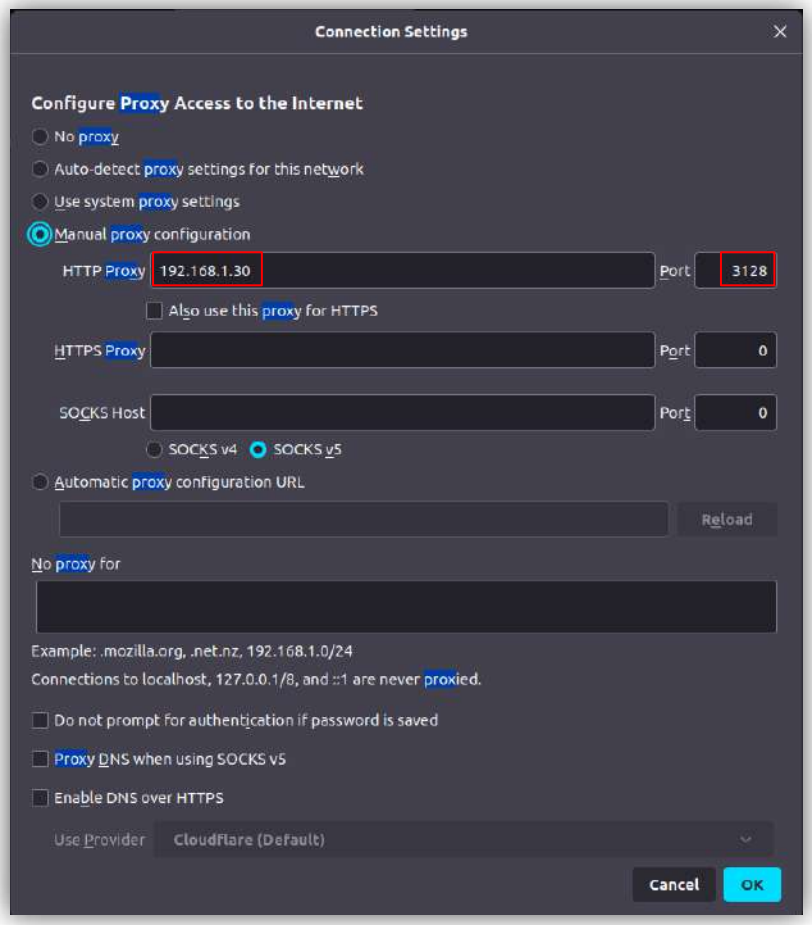
התחלנו בלהתקין שרת פרוקסי שנקרא squid על אחד המחשבים שלנו בעזרת הפקודה הבאה:

```
jonathan@mousehouse:~$ sudo apt install squid
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  squidclient squid-cgi squid-purge resolvconf smbclient winbind
The following NEW packages will be installed:
  squid
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 2,799 kB of archives.
After this operation, 8,540 kB of additional disk space will be used.
Get:1 http://il.archive.ubuntu.com/ubuntu jammy-updates/main amd64 squid amd64 5.2-1ubuntu4.2 [2,799 kB]
Fetched 2,799 kB in 2s (1,563 kB/s)
Selecting previously unselected package squid.
```

והרצנו את השרת בעזרת הפקודה הנתונה בפירוט התרגיל:

```
jonathan@mousehouse:~$ sudo service squid start
[sudo] password for jonathan:
```

לאחר מכן, כמו שחמי המלך ביקש בתרגיל, הלכנו להגדרות האינטרנט שלנו במחשב השני ושינינו את הגדרות ה proxy על מנת שהיה מחובר לשרת:



לאחר שיצרנו את שרת ה Proxy שלנו והגדרנו את הדפדפן שלנו כדי שיוכל להתחבר אליו, פתחנו בדפדפן אתר אינטרנט המאפשר גישה אליו באמצעות פרוטוקול HTTP.

נשים לב שנכשלנו מלהיכנס בדיוק כמו שחמי אמר (וכמובן שצדק!) כאשר כתב את התרגיל.

נתקן את התוצאה בכך שניכנס לתיקייה etc/squid/ ונשנה בקובץ את השורה כמתואר בתרגיל:

```
jonathan@mousehouse:~$ cd /etc/squid
jonathan@mousehouse:/etc/squid$
```

```
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
#http_access allow localnet
http_access allow localhost

# And finally deny all other access to this proxy
http_access deny all

#  TAG: adapted_http_access
#       Allowing or Denying access based on defined access lists
#
#       Essentially identical to http_access, but runs after redirectors
#       and ICAP/eCAP adaptation. Allowing access control based on their
#       output.
#
```

```
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
#http_access allow localnet
http_access allow localhost

# And finally deny all other access to this proxy
http_access allow all

#  TAG: adapted_http_access
#       Allowing or Denying access based on defined access lists
#
#       Essentially identical to http_access, but runs after redirectors
#       and ICAP/eCAP adaptation. Allowing access control based on their
#       output.
#
```

ואתחלנו את השרת בעזרת הפקודה הנ"ל:

```
jonathan@mousehouse:/etc/squid$ sudo service squid restart
```

ניסינו להתחבר לאתר מחדש וקיבלנו את התוצאה הבאה בדפדפן:

Not Secure — info.cern.ch

http://info.cern.ch - home of the first website

From here you can:

- Browse the first website
- Browse the first website using the line-mode browser simulator
- Learn about the birth of the web
- Learn about CERN, the physics laboratory where the web was born

נשים לב שעכשיו לאחר ההגדרות החדשות הצלחנו להתחבר לשרת.

הערה: על מנת להגיע לאתר שבחרנו לחץ כאן.

הערת אגב: נשים לב כי ליד כתובת האתר כתוב Not Secure. על מנת לדעת מדוע נסביר את ההבדל בין פרוטוקול Http ו Https.

בדומה ל http, -גם https הוא פרוטוקול תקשורת אינטרנטי שמסייע בהעברת אותם נתונים על גבי רשת האינטרנט.
ההבדל הוא ברמת האבטחה של העברת הנתונים.

על ידי שימוש בפרוטוקול https, נתונים המוזנים באתר יוצפנו לפני שיועברו לגורם צד שלישי.
דוגמה נפוצה היא באתרי קניות, כאשר בתהליך הרכישה הקונים מתבקשים להזין פרטי אשראי.
פרוטוקול זה הוא פרוטוקול חדשני, מאובטח ובשנים האחרונות נכנס כפרוטוקול סטנדרט לאתרים שמופיעים בתוצאות החיפוש של גוגל.
הפירוש של https הוא Hypertext Transfer Protocol Secure, ובתרגום חופשי: פרוטוקול אובטח להעברת תמלול.

מקור: האתר של רותם קנון

לאחר שסיימנו את כל ההכנות, ניתן להסניף את התעבורה ב Wireshark.
להלן התעבורה בעת התחברות לאתר http://neverssl.com:

No.	Time	Source	Destination	Protocol	Length	Info
1047	7.687263449	192.168.1.30	192.117.235.237	DNS	83	Standard query 0xa4a9 A neverssl.com OPT
1048	7.687319253	192.168.1.30	192.117.235.237	DNS	83	Standard query 0xf3de AAAA neverssl.com OPT
1051	7.763798108	192.117.235.237	192.168.1.30	DNS	99	Standard query response 0xa4a9 A neverssl.com A 34.223.124.45 OPT
1053	7.823631594	192.117.235.237	192.168.1.30	DNS	111	Standard query response 0xf3de AAAA neverssl.com AAAA 2600:1f13:37c:1400:ba21:7165:5fc7:736e OPT
1081	8.260223583	192.168.1.30	192.117.235.237	DNS	103	Standard query 0x1ed6 A oldyoungwholesunset.neverssl.com OPT
1082	8.260343293	192.168.1.30	192.117.235.237	DNS	103	Standard query 0x5c2c AAAA oldyoungwholesunset.neverssl.com OPT
1085	8.288982041	192.117.235.237	192.168.1.30	DNS	131	Standard query response 0x5c2c AAAA oldyoungwholesunset.neverssl.com AAAA 2600:1f13:37c:1400:ba21:7165:5fc7:736e OPT
1091	8.367621499	192.117.235.237	192.168.1.30	DNS	119	Standard query response 0x1ed6 A oldyoungwholesunset.neverssl.com A 34.223.124.45 OPT

כתובת שרת הפרוקסי היא כתובת המחשב שלנו (שהרי אנו מריצים אותו באופן לוקאלי) - 192.169.1.30. אכן, אפשר לראות ברשומות שהתעבורה הוסנפה דרך המחשב עליו התקנו את שרת הפרוקסי. כדי להיווכח שאכן שרת הפרוקסי מתפקד כמו שצריך ושהלקוח פונה לשרת הפרוקסי, נתבונן בתעבורה: נבחין כי שאילתות ה-DNS והתשובות שלהן עוברות דרך שרת הפרוקסי שלנו, ואינן נשלחות ומתקבלות ישר מהלקוח אל השרתים של גוגל .

```
1047 7.687263449   192.168.1.30   192.117.235.237   DNS   83 Standard query 0xa4a9 A neverssl.com OPT

▸ Frame 1047: 83 bytes on wire (664 bits), 83 bytes captured (664 bits) on interface wlp2s0, id 0
▸ Ethernet II, Src: CloudNet_8f:6a:13 (dc:e9:94:8f:6a:13), Dst: DrayTek_9c:b8:40 (00:1d:aa:9c:b8:40)
▸ Internet Protocol Version 4, Src: 192.168.1.30, Dst: 192.117.235.237
▸ User Datagram Protocol, Src Port: 36815, Dst Port: 53
▸ Domain Name System (query)
```

```
1045 7.686678383   192.168.1.35   192.168.1.30   HTTP   419 GET http://neverssl.com/ HTTP/1.1
1046 7.686703246   192.168.1.30   192.168.1.35   TCP     54 3128 → 63198 [ACK] Seq=1 Ack=366 W
1071 8.208715137   192.168.1.30   192.168.1.35   TCP    476 3128 → 63198 [PSH, ACK] Seq=1 Ack=
1072 8.208756064   192.168.1.30   192.168.1.35   TCP   1514 3128 → 63198 [ACK] Seq=423 Ack=366
1073 8.208761791   192.168.1.30   192.168.1.35   HTTP   494 HTTP/1.1 200 OK  (text/html)

▸ Frame 1045: 419 bytes on wire (3352 bits), 419 bytes captured (3352 bits) on interface wlp2s0, id 0
▸ Ethernet II, Src: IntelCor_7d:71:93 (18:26:49:7d:71:93), Dst: CloudNet_8f:6a:13 (dc:e9:94:8f:6a:13)
▸ Internet Protocol Version 4, Src: 192.168.1.35, Dst: 192.168.1.30
▸ Transmission Control Protocol, Src Port: 63198, Dst Port: 3128, Seq: 1, Ack: 1, Len: 365
```

ניתן גם לראות כי החיבור של שרת הפרוקסי עם הלקוח מצד השרת הוא מפורט מספר 3128, אך התקשורת של שרת הפרוקסי עם השרת החיצוני ששומר את המידע של האתר (גם מצד השרת), היא מפורט 63198 - כלומר שרת הפרוקסי מתקשר משני חיבורים שונים במהלך הבקשה.

כעת נתבונן בתמונות להלן:

```
1045 7.686678383   192.168.1.35   192.168.1.30   HTTP   419 GET http://neverssl.com/ HTTP/1.1
▸ Frame 1045: 419 bytes on wire (3352 bits), 419 bytes captured (3352 bits) on interface wlp2s0, id 0
▸ Ethernet II, Src: IntelCor_7d:71:93 (18:26:49:7d:71:93), Dst: CloudNet_8f:6a:13 (dc:e9:94:8f:6a:13)
▸ Internet Protocol Version 4, Src: 192.168.1.35, Dst: 192.168.1.30
▸ Transmission Control Protocol, Src Port: 63198, Dst Port: 3128, Seq: 1, Ack: 1, Len: 365
▾ Hypertext Transfer Protocol
  ▸ GET http://neverssl.com/ HTTP/1.1\r\n
    Host: neverssl.com\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/109.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\n
    Accept-Language: en-US,en;q=0.5\r\n
    Accept-Encoding: gzip, deflate\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    \r\n
    [Full request URI: http://neverssl.com/]
    [HTTP request 1/4]
    [Response in frame: 1073]
    [Next request in frame: 1079]
```

```
1073 8.208761791   192.168.1.30   192.168.1.35   HTTP   494 HTTP/1.1 200 OK  (text/html)
▸ Frame 1073: 494 bytes on wire (3952 bits), 494 bytes captured (3952 bits) on interface wlp2s0, id 0
▸ Ethernet II, Src: CloudNet_8f:6a:13 (dc:e9:94:8f:6a:13), Dst: IntelCor_7d:71:93 (18:26:49:7d:71:93)
▸ Internet Protocol Version 4, Src: 192.168.1.30, Dst: 192.168.1.35
▸ Transmission Control Protocol, Src Port: 3128, Dst Port: 63198, Seq: 1883, Ack: 366, Len: 440
▸ [3 Reassembled TCP Segments (2322 bytes): #1071(422), #1072(1460), #1073(440)]
▾ Hypertext Transfer Protocol
  ▸ HTTP/1.1 200 OK\r\n
    Date: Thu, 19 Jan 2023 17:52:50 GMT\r\n
    Server: Apache/2.4.54 ()\r\n
    Last-Modified: Wed, 29 Jun 2022 00:23:33 GMT\r\n
    ETag: "f79-5e28b29d38e93-gzip"\r\n
    Accept-Ranges: bytes\r\n
    Vary: Accept-Encoding\r\n
    Content-Encoding: gzip\r\n
  ▸ Content-Length: 1900\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    X-Cache: MISS from mousehouse\r\n
    X-Cache-Lookup: MISS from mousehouse:3128\r\n
    Via: 1.1 mousehouse (squid/5.2)\r\n
    Connection: keep-alive\r\n
    \r\n
    [HTTP response 1/4]
    [Time since request: 0.522083408 seconds]
    [Request in frame: 1045]
    [Next request in frame: 1079]
    [Next response in frame: 1159]
    [Request URI: http://neverssl.com/]
    Content-encoded entity body (gzip): 1900 bytes -> 3961 bytes
    File Data: 3961 bytes
▸ Line-based text data: text/html (131 lines)
```

כפי שהניתן לראות, הלקוח מבקש משרת הפרוקסי את דף האינטרנט, כלומר הלוקח מבקש את המשאבים של הדף מהשרת הלוקאלי – ולא ישירות מהשרתים החיצוניים. בדיוק כמו שאנחנו מצפים, שרת הפרוקסי מוריד אליו את תוכן הדף כיוון שתוכן הדף עובר דרך שרת הפרוקסי ונשלח ללקוח עם תגובת 200 OK לאחר מכן.