

תרגיל 1:

הערה עבור שני החלקים: מאחר והמחשב שלי לא מספיק חזק על מנת להריץ שני מכונות וירטואליות במקביל, נאלצתי להריץ גם את השרת וגם את הלקוח על אותה מכונה וירטואלית, ולכן ה-IP זהה עבור שניהם.

חלק א:

על מנת לסנן את החבילות השתמשתי בפילטר

`udp.port == 5402 && ip.src == 10.0.2.15`

שבעצם מציג את כל ההודעות שנשלחו מפורט 5402 או לפורט 5402 מה-IP 10.0.2.15 אם הייתי מריץ את השרת ואת הלקוח על שני מכונות וירטואליות שונות, אזי הייתי צריך להוסיף לפילטר

`|| ip.src == 10.0.2.16`

בהנחה שזהו ה-IP של המכונה הוירטואלית השנייה.

ההבדל בשימוש במספרי הפורט בין השרת ללקוח הוא שהשרת רץ על הפורט שניתן לו בקוד, ולעומת זאת, הלקוח מקבל פורט אקראי מהמערכת, אך שולח את ההודעות שלו לפורט שניתן לו בקוד.

נסתכל לדוגמא על ההודעה שנשלחה מהלקוח לשרת עם השם שלי:

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
udp.port == 5402 && ip.src == 10.0.2.15						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	10.0.2.15	UDP	46	60481 → 5402 Len=2
2	0.000137310	10.0.2.15	10.0.2.15	UDP	46	5402 → 60481 Len=2
3	59.273997960	10.0.2.15	10.0.2.15	UDP	58	60481 → 5402 Len=14
4	59.274077808	10.0.2.15	10.0.2.15	UDP	58	5402 → 60481 Len=14
5	65.245391508	10.0.2.15	10.0.2.15	UDP	57	60481 → 5402 Len=13
6	65.245477617	10.0.2.15	10.0.2.15	UDP	57	5402 → 60481 Len=13

Frame 3: 58 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface 0 Linux cooked capture Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.15 User Datagram Protocol, Src Port: 60481, Dst Port: 5402 Data (14 bytes)	
--	--

0000	00 00 03 04 00 06 00 00 00 00 00 00 00 08 00
0010	45 00 00 2a f4 eb 40 00 40 11 2d ba 0a 00 02 0f	E.....@.....
0020	0a 00 02 0f ec 41 15 1a 00 16 18 45 4e 61 76 65A.....ENave
0030	68 20 4d 61 72 63 68 6f 6f 6d	h Marcho om

נשים לב שבשכבת ה-transport ה-dst port הוא אכן הפורט שנתנו לו בקוד, אך לעומת זאת ה-src port הוא פורט אקראי. נזכיר כי התפקיד של הפורט הוא לתת אינדיקציה לאיזה אפליקציה מיועדת ההודעה המועברת, לכן אף שהשרת והלקוח רצים על אותה מכונה וירטואלית, דרושים להם שני מספרי פורט שונים.

מאחר והרצנו את השרת והלקוח על אותה מכונה וירטואלית, הם רצים על אותו IP, אותו מצאנו על ידי הרצת פקודת `ifconfig`:

```
naveh@Naveh-VB:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::56c2:15e7:49df:2787 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:ad:a7:c0 txqueuelen 1000 (Ethernet)
    RX packets 192764 bytes 217357754 (217.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 48382 bytes 2989977 (2.9 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 163 bytes 13493 (13.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 163 bytes 13493 (13.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

חלק ב:

כשה-wireshark עובד ברקע הרצתי את השרת ושני ללקוחות בצורה הבאה:

```
naveh@Naveh-VB:~/Desktop$ python3 chatclient.py localhost 5402
Enter a message to send:1 Naveh
Enter a message to send:5
Naveh has joined the chat
Enter a message to send:3 ID:312275746
Enter a message to send:5
Naveh has joined the chat
Marchoom has joined the chat
Marchoom: Hello
Naveh has changed his name to ID:312275746
Enter a message to send:4

naveh@Naveh-VB: ~/Desktop
File Edit View Search Terminal Help
naveh@Naveh-VB:~/Desktop$ python3 chatclient.py localhost 5402
Enter a message to send:Hello
illegal request
Enter a message to send:1 Marchoom
Enter a message to send:2 Hello
Enter a message to send:5
Marchoom has joined the chat
Marchoom: Hello
Enter a message to send:5
Marchoom has joined the chat
Marchoom: Hello
Naveh has changed his name to ID:312275746
ID:312275746 has left the group
Enter a message to send:4

naveh@Naveh-VB: ~/Desktop
File Edit View Search Terminal Help
naveh@Naveh-VB:~/Desktop$ python3
chatserver.py 5402
```

ננסה לעבור על ההודעות שנשלחו מהלקוחות לשרת ומהשרת ללקוחות עבור כל אחת מהתופעות.
נראה עבור הלקוח הראשון את ההתחברות לשרת עם ההודעה 1: Naveh

```

▶ Frame 17: 51 bytes on wire (408 bits), 51 bytes captured (408 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 47412, Dst Port: 5402
▶ Data (7 bytes)

```

```

0000  00 00 03 04 00 06 00 00 00 00 00 00 00 00 08 00  .....
0010  45 00 00 23 3c b0 40 00 40 11 00 18 7f 00 00 01  E..#<.@. @.....
0020  7f 00 00 01 b9 34 15 1a 00 0f fe 22 31 20 4e 61  .....4.. '"1 Na
0030  76 65 68                                           veh

```

נשים לב שהפורט של הלקוח הוא 47412, כמו בחלק א' הוא נבחר באקראי.
נראה שבתגובה השרת שולח בחזרה ללקוח את ההודעה הבאה:

```

▶ Frame 18: 45 bytes on wire (360 bits), 45 bytes captured (360 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 5402, Dst Port: 47412
▶ Data (1 byte)

```

```

0000  00 00 03 04 00 06 00 00 00 00 00 00 00 00 08 00  .....
0010  45 00 00 1d 3c b1 40 00 40 11 00 1d 7f 00 00 01  E..<.@. @.....
0020  7f 00 00 01 15 1a b9 34 00 09 fe 1c 00           .....4.. 00

```

אני רוצה להבהיר שהדרך שבה בחרתי שבה השרת מודיע ללקוח לאיזה מצב לעבור היא ע"י שליחת סוג של הודעה. אם הלקוח מקבל הודעה ריקה, אז הוא יודע שלהתחיל להאזין לשרת ולהדפיס את ההודעות שהוא מקבל, עד קבלה של ההודעה הבאה - אם הוא מקבל הודעה עם בית 1 שמכיל 0, הוא יודע שהוא צריך לחזור למצב קליטת קלט מהמשתמש. אם הלקוח מקבל הודעה עם שני בתים שמכילים 0, הוא שולח הודעת *Illegal Request* למשתמש וחוזר לקלוט קלט מהמשתמש.

נראה שכאשר הלקוח שלח את ההודעה 5, שמבקשת לקבל את כל ההודעות שנשלחו בצ'אט עד כה, הוא יקבל את ההודעה הראשונה (הודעה ריקה):
ההודעה שהלקוח שלח לשרת:

```

▶ Frame 19: 45 bytes on wire (360 bits), 45 bytes captured (360 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 47412, Dst Port: 5402
▶ Data (1 byte)

```

```

0000  00 00 03 04 00 06 00 00 00 00 00 00 00 00 08 00  .....
0010  45 00 00 1d 3e 59 40 00 40 11 fe 74 7f 00 00 01  E..>Y@. @..t....
0020  7f 00 00 01 b9 34 15 1a 00 09 fe 1c 35           .....4.. 5

```

ההודעות שקיבל בתגובה מהשרת:

ההודעה הריקה שאומרת לו להתחיל לקבל הודעות מהשרת להדפיס אותן:

```

▶ Frame 20: 44 bytes on wire (352 bits), 44 bytes captured (352 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 5402, Dst Port: 47412

```

```

0000  00 00 03 04 00 06 00 00 00 00 00 00 00 00 08 00  .....
0010  45 00 00 1c 3e 5a 40 00 40 11 fe 74 7f 00 00 01  E..>Z@. @..t....
0020  7f 00 00 01 15 1a b9 34 00 08 fe 1b             .....4..

```

ההודעה היחידה שנשלחה עד כה בצ'אט (הודעת ההתחברות של הלקוח):

```

▶ Frame 21: 69 bytes on wire (552 bits), 69 bytes captured (552 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 5402, Dst Port: 47412
▶ Data (25 bytes)

```

0010	45 00 00 35 3e 5b 40 00	40 11 fe 5a 7f 00 00 01	E..5>[@. @. Z....
0020	7f 00 00 01 15 1a b9 34	00 21 fe 34 4e 61 76 654 .!.4Nave
0030	68 20 68 61 73 20 6a 6f	69 6e 65 64 20 74 68 65	h has jo ined the
0040	20 63 68 61 74		chat

ולאחר מכן ההודעה עם הבית הבודד שאומרת ללקוח לחזור לקלט מהמשתמש:

```

▶ Frame 22: 45 bytes on wire (360 bits), 45 bytes captured (360 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 5402, Dst Port: 47412
▶ Data (1 byte)

```

0000	00 00 03 04 00 06 00 00	00 00 00 00 00 00 08 00
0010	45 00 00 1d 3e 5c 40 00	40 11 fe 71 7f 00 00 01	E...>\ @. @. q....
0020	7f 00 00 01 15 1a b9 34	00 09 fe 1c 004

מאחר ו3 ו4 עובדים פחות או יותר באותה דרך, אין צורך להסביר אותם בנוסף. רק אני רוצה להוסיף, שלאחר שהשרת מקבל את הודעה 4 במצב חוקי (כאשר המשתמש רשום לצ'אט) אז הוא רק מוריד את הלקוח מרשימת המשתמשים, ולא שולח שום תגובה בחזרה ללקוח, ולכן הלקוח לא חוזר למצב קליטה מהמשתמש לאחר התנתקות.

נראה מצב של קלט לא חוקי: ההודעה הראשונה ששולח הלקוח השני היא *Hello* שלא עומדת בתנאים שנקבעו עבור הודעות לשרת, ולכן נראה איך השרת מגיב: ההודעה מהלקוח לשרת:

```

▶ Frame 23: 49 bytes on wire (392 bits), 49 bytes captured (392 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 54813, Dst Port: 5402
▶ Data (5 bytes)

```

0000	00 00 03 04 00 06 00 00	00 00 00 00 00 00 08 00
0010	45 00 00 21 49 d1 40 00	40 11 f2 f8 7f 00 00 01	E..!I. @. @.
0020	7f 00 00 01 d6 1d 15 1a	00 0d fe 20 48 65 6c 6c Hell
0030	6f		o

ובתגובה השרת מחזיר הודעה עם שני בתים מאופסים:

```

▶ Frame 24: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▶ User Datagram Protocol, Src Port: 5402, Dst Port: 54813
▶ Data (2 bytes)

```

0000	00 00 03 04 00 06 00 00	00 00 00 00 00 00 08 00
0010	45 00 00 1e 49 d2 40 00	40 11 f2 fa 7f 00 00 01	E...I. @. @.
0020	7f 00 00 01 15 1a d6 1d	00 0a fe 1d 00 00

מה שקבעתי כסימן לבקשה לא חוקית.