

## תרגיל 2:

## חלק א:

1.

נציג את תהליך ההתחברות (מנקודת המבט של הלקוח):

קודם כל שלחנו אל השרת הודעה ריקה עם דגל SYN, על מנת להודיע לו שאנו רוצים לפתוח קישור TCP איתו. נתנו לו מספר SEQUENCE אקראי שבחרנו, שממנו נתחיל לספור: 3069583718.

|    |              |              |              |     |    |               |            |                |
|----|--------------|--------------|--------------|-----|----|---------------|------------|----------------|
| 10 | 23.920103442 | 10.0.2.15    | 192.168.1.20 | TCP | 76 | 39306 → 12345 | [SYN]      | Seq=3069583718 |
| 11 | 23.920799202 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [SYN, ACK] | Seq=6016001    |
| 12 | 23.920828967 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583719 |
| 31 | 29.097682713 | 10.0.2.15    | 192.168.1.20 | TCP | 70 | 39306 → 12345 | [PSH, ACK] | Seq=3069583719 |
| 32 | 29.098017313 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016002    |
| 33 | 29.098290555 | 192.168.1.20 | 10.0.2.15    | TCP | 70 | 12345 → 39306 | [PSH, ACK] | Seq=6016002    |
| 34 | 29.098298746 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583733 |
| 35 | 33.816684563 | 10.0.2.15    | 192.168.1.20 | TCP | 65 | 39306 → 12345 | [PSH, ACK] | Seq=3069583733 |
| 36 | 33.817047924 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016016    |

▶ Frame 10: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface 0  
 ▶ Linux cooked capture  
 ▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20  
 ▶ Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583718, Len: 0

בנוסף אנו מודיעים לשרת שגודל החלון שלנו הוא 64240 וגודל ה-MSS שלנו הוא 1460:

Seq=3069583718 Win=64240 Len=0 MSS=1460 SACK\_PERM=1 TSval=1594107584 TSecr=0 WS=128

בתגובה, השרת שלח לנו בחזרה הודעה ריקה עם דגלי SYN ו-ACK, על מנת להודיע לנו שהוא מקבל את בקשת פתיחת החיבור אליו. הוא שולח לנו מספר ACK 3069583719 (מעלה את מספר ה-SEQ שלנו באחד) ושולח לנו מספר SEQ אקראי שהוא בחר לעצמו: 6016001.

|    |              |              |              |     |    |               |            |                            |
|----|--------------|--------------|--------------|-----|----|---------------|------------|----------------------------|
| 10 | 23.920103442 | 10.0.2.15    | 192.168.1.20 | TCP | 76 | 39306 → 12345 | [SYN]      | Seq=3069583718 Win=64240   |
| 11 | 23.920799202 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [SYN, ACK] | Seq=6016001 Ack=3069583719 |
| 12 | 23.920828967 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583719 Ack=6016001 |
| 31 | 29.097682713 | 10.0.2.15    | 192.168.1.20 | TCP | 70 | 39306 → 12345 | [PSH, ACK] | Seq=3069583719 Ack=6016001 |
| 32 | 29.098017313 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016002 Ack=3069583719 |
| 33 | 29.098290555 | 192.168.1.20 | 10.0.2.15    | TCP | 70 | 12345 → 39306 | [PSH, ACK] | Seq=6016002 Ack=3069583719 |
| 34 | 29.098298746 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583733 Ack=6016001 |
| 35 | 33.816684563 | 10.0.2.15    | 192.168.1.20 | TCP | 65 | 39306 → 12345 | [PSH, ACK] | Seq=3069583733 Ack=6016001 |
| 36 | 33.817047924 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016016 Ack=3069583733 |

▶ Frame 11: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0  
 ▶ Linux cooked capture  
 ▶ Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15  
 ▶ Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016001, Ack: 3069583719, Len: 0  
 ▶ VSS-Monitoring ethernet trailer, Source Port: 0

בנוסף הוא מודיע לנו שגודל החלון שלו הוא 65535 וגודל ה-MSS שלו הוא גם 1460:

[SYN, ACK] Seq=6016001 Ack=3069583719 Win=65535 Len=0 MSS=1460

עכשיו אנחנו מחזירים הודעת ACK לשרת כדי לאשר קבלה עם מספר ACK 6016002, ומחכים לאפליקציה להתחיל לשלוח הודעות:

|    |              |              |              |     |    |               |            |                                      |
|----|--------------|--------------|--------------|-----|----|---------------|------------|--------------------------------------|
| 12 | 23.920828967 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583719 Ack=6016002 Win=65535 |
| 31 | 29.097682713 | 10.0.2.15    | 192.168.1.20 | TCP | 70 | 39306 → 12345 | [PSH, ACK] | Seq=3069583719 Ack=6016002           |
| 32 | 29.098017313 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016002 Ack=3069583733 Win=65535 |
| 33 | 29.098290555 | 192.168.1.20 | 10.0.2.15    | TCP | 70 | 12345 → 39306 | [PSH, ACK] | Seq=6016002 Ack=3069583733           |
| 34 | 29.098298746 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK]      | Seq=3069583733 Ack=6016016 Win=65535 |
| 35 | 33.816684563 | 10.0.2.15    | 192.168.1.20 | TCP | 65 | 39306 → 12345 | [PSH, ACK] | Seq=3069583733 Ack=6016016           |
| 36 | 33.817047924 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 12345 → 39306 | [ACK]      | Seq=6016016 Ack=3069583742 Win=65535 |

▶ Frame 12: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0  
 ▶ Linux cooked capture  
 ▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20  
 ▶ Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583719, Ack: 6016002, Len: 0

הודעה ראשונה שאנו שולחים מהאפליקציה היא הודעה באורך 14 המכילה את השם שלי:

```

31 29.097682713 10.0.2.15 192.168.1.20 TCP 70 39306 → 12345 [PSH, ACK] Seq=3069583719 Ack=6016002 Len=14
Frame 31: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20
Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583719, Ack: 6016002, Len: 14
Data (14 bytes)
0000 00 04 00 01 00 06 08 00 27 ad a7 c0 00 00 08 00 ..... '.....
0010 45 00 00 36 a4 3c 40 00 40 06 c8 ba 0a 00 02 0f E..6.<@. @.....
0020 c0 a8 01 14 99 8a 30 39 b6 f6 21 67 00 5b cc 02 .....09...!g.[..
0030 50 18 fa f0 cd f3 00 00 4e 61 76 65 68 20 4d 61 P..... Naveh Ma
0040 72 63 68 6f 6f 6d rchoom

```

נשים לב שמספר הSEQ ומספר הACK עדיין לא עלו, אך בהודעת הACK שהתקבלה מהשרת מספר הACK עלה ב14:

```

32 29.098017313 192.168.1.20 10.0.2.15 TCP 62 12345 → 39306 [ACK] Seq=6016002 Ack=3069583733 Len=0
Frame 32: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15
Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016002, Ack: 3069583733, Len: 0
VSS-Monitoring ethernet trailer, Source Port: 0
0000 00 00 00 01 00 06 52 54 00 12 35 02 00 00 08 00 .....RT..5.....
0010 45 00 00 28 02 34 00 00 40 06 aa d1 c0 a8 01 14 E..(.4.. @.....
0020 0a 00 02 0f 30 39 99 8a 00 5b cc 02 b6 f6 21 75 .....09...[.....!u
0030 50 10 ff ff 73 7c 00 00 00 00 00 00 00 00 P...s|.....

```

בתגובה, השרת עכשיו שולח לנו את ההודעה ששלנו לו באותיות גדולות, וכמו בהודעה שאנו שלחנו, הוא עדיין לא מעלה את מספר הSEQ ומספר הACK:

```

33 29.098290555 192.168.1.20 10.0.2.15 TCP 70 12345 → 39306 [PSH, ACK] Seq=6016002 Ack=3069583733 Len=14
Frame 33: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15
Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016002, Ack: 3069583733, Len: 14
Data (14 bytes)
0000 00 00 00 01 00 06 52 54 00 12 35 02 00 00 08 00 .....RT..5.....
0010 45 00 00 36 02 35 00 00 40 06 aa c2 c0 a8 01 14 E..6.5.. @.....
0020 0a 00 02 0f 30 39 99 8a 00 5b cc 02 b6 f6 21 75 .....09...[.....!u
0030 50 18 ff ff 4f 9e 00 00 4e 41 56 45 48 20 4d 41 P...0... NAVEH MA
0040 52 43 48 4f 4f 4d RCHOOM

```

אך בהודעת הACK שאנו שולחים בתגובה אנו מעלים את מספר הACK ב14:

```

34 29.098298746 10.0.2.15 192.168.1.20 TCP 56 39306 → 12345 [ACK] Seq=3069583733 Ack=6016016 Len=0
Frame 34: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20
Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583733, Ack: 6016016, Len: 0
0000 00 04 00 01 00 06 08 00 27 ad a7 c0 00 00 08 00 ..... '.....
0010 45 00 00 28 a4 3d 40 00 40 06 c8 c7 0a 00 02 0f E..(.=>@. @.....
0020 c0 a8 01 14 99 8a 30 39 b6 f6 21 75 00 5b cc 10 .....09...!u.[..
0030 50 10 fa e2 cd ee 00 00 33 31 32 32 37 35 37 34 P..... 31227574
0040 36

```

עכשיו שלחנו לשרת תעודת הזדהות שלנו:

```

35 33.816684563 10.0.2.15 192.168.1.20 TCP 65 39306 → 12345 [PSH, ACK] Seq=3069583733 Ack=6016016 Len=9
Frame 35: 65 bytes on wire (520 bits), 65 bytes captured (520 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20
Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583733, Ack: 6016016, Len: 9
Data (9 bytes)
0000 00 04 00 01 00 06 08 00 27 ad a7 c0 00 00 08 00 ..... '.....
0010 45 00 00 31 a4 3e 40 00 40 06 c8 bd 0a 00 02 0f E..1.>@. @.....
0020 c0 a8 01 14 99 8a 30 39 b6 f6 21 75 00 5b cc 10 .....09...!u.[..
0030 50 18 fa e2 cd ee 00 00 33 31 32 32 37 35 37 34 P..... 31227574
0040 36

```

ובתגובה הוא שלח לנו הודעת ACK:

|   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |                 |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------------|
| 36 33.817047924 192.168.1.20 10.0.2.15 TCP 62 12345 → 39306 [ACK] Seq=6016016 Ack=3069583742  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |                 |
| Frame 36: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0<br>Linux cooked capture<br>Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15<br>Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016016, Ack: 3069583742, Len: 0<br>VSS-Monitoring ethernet trailer, Source Port: 0 |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |                 |
| 00  | 00 | 00 | 00 | 01 | 00 | 06 | 52 | 54 | 00 | 12 | 35 | 02 | 00 | 00 | 08 | 00 | .....RT..5..... |
| 10  | 45 | 00 | 00 | 28 | 02 | 36 | 00 | 00 | 40 | 06 | aa | cf | c0 | a8 | 01 | 14 | E..(·6..@.....  |
| 20  | 0a | 00 | 02 | 0f | 30 | 39 | 99 | 8a | 00 | 5b | cc | 10 | b6 | f6 | 21 | 7e | ....09..[....!~ |
| 30  | 50 | 10 | ff | ff | 73 | 65 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | P...se..        |

בה כרגיל הוא מעלה את מספר הACK ב9,

ועכשיו הוא שולח לנו את מספר תעודת הזהות שלנו בתגובה:

|  |              |              |           |     |    |       |    |       |            |             |                |    |    |    |    |    |                  |
|--|--------------|--------------|-----------|-----|----|-------|----|-------|------------|-------------|----------------|----|----|----|----|----|------------------|
| 37   | 33.817344432 | 192.168.1.20 | 10.0.2.15 | TCP | 65 | 12345 | -  | 39306 | [PSH, ACK] | Seq=6016016 | Ack=3069583742 |    |    |    |    |    |                  |
| Frame 37: 65 bytes on wire (520 bits), 65 bytes captured (520 bits) on interface 0<br>Linux cooked capture<br>Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15<br>Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016016, Ack: 3069583742, Len: 9<br>Data (9 bytes) |              |              |           |     |    |       |    |       |            |             |                |    |    |    |    |    |                  |
| 00   | 00           | 00           | 00        | 01  | 00 | 06    | 52 | 54    | 00         | 12          | 35             | 02 | 00 | 00 | 08 | 00 | .....RT..5.....  |
| 10   | 45           | 00           | 00        | 31  | 02 | 37    | 00 | 00    | 40         | 06          | aa             | c5 | c0 | a8 | 01 | 14 | E..1·7..@.....   |
| 20   | 0a           | 00           | 02        | 0f  | 30 | 39    | 99 | 8a    | 00         | 5b          | cc             | 10 | b6 | f6 | 21 | 7e | ....09..[....!~  |
| 30   | 50           | 18           | ff        | ff  | 69 | 87    | 00 | 00    | 33         | 31          | 32             | 32 | 37 | 35 | 37 | 34 | P...i...31227574 |
| 40   | 36           |              |           |     |    |       |    |       |            |             |                |    |    |    |    |    | 6                |

ואנו מחזירים ACK עם מספר ACK גדול ב9:

|  |              |           |              |     |    |               |       |                |                         |                |
|--|--------------|-----------|--------------|-----|----|---------------|-------|----------------|-------------------------|----------------|
| 38   | 33.817353391 | 10.0.2.15 | 192.168.1.20 | TCP | 56 | 39306 → 12345 | [ACK] | Seq=3069583742 | Ack=6016025             |                |
|  |              |           |              |     |    |               |       |                |                         |                |
| Frame 38: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0                     |              |           |              |     |    |               |       |                |                         |                |
| Linux cooked capture   |              |           |              |     |    |               |       |                |                         |                |
| Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20   |              |           |              |     |    |               |       |                |                         |                |
| Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583742, Ack: 6016025, Len: 0 |              |           |              |     |    |               |       |                |                         |                |
|  |              |           |              |     |    |               |       |                |                         |                |
| 00   | 00           | 04        | 00           | 01  | 00 | 06            | 08    | 00             | 27 ad a7 c0 00 00 08 00 | .....'.....    |
| 10   | 45           | 00        | 00           | 28  | a4 | 3f            | 40    | 00             | 40 06 c8 c5 0a 00 02 0f | E..(·?@·@..... |
| 20   | c0           | a8        | 01           | 14  | 99 | 8a            | 30    | 39             | b6 f6 21 7e 00 5b cc 19 | ....09..!~[... |
| 30   | 50           | 10        | fa           | d9  | cd | e5            | 00    | 00             |                         | P.....         |

עכשיו, אחרי ששלחנו לאפליקציה הודעת quit, אנחנו שולחים לשרת הודעה ריקה עם דגל FIN:

|  |              |           |              |     |    |       |   |       |            |                |             |
|--|--------------|-----------|--------------|-----|----|-------|---|-------|------------|----------------|-------------|
| 39   | 36.392012496 | 10.0.2.15 | 192.168.1.20 | TCP | 56 | 39306 | - | 12345 | [FIN, ACK] | Seq=3069583742 | Ack=6016025 |
| Frame 39: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0                     |              |           |              |     |    |       |   |       |            |                |             |
| Linux cooked capture   |              |           |              |     |    |       |   |       |            |                |             |
| Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.20   |              |           |              |     |    |       |   |       |            |                |             |
| Transmission Control Protocol, Src Port: 39306, Dst Port: 12345, Seq: 3069583742, Ack: 6016025, Len: 0 |              |           |              |     |    |       |   |       |            |                |             |

  

|      |                         |                         |                 |
|------|-------------------------|-------------------------|-----------------|
| 0000 | 00 04 00 01 00 06 08 00 | 27 ad a7 c0 00 00 08 00 | .....'.....     |
| 0010 | 45 00 00 28 a4 40 40 00 | 40 06 c8 c4 0a 00 02 0f | E..(·00·@.....  |
| 0020 | c0 a8 01 14 99 8a 30 39 | b6 f6 21 7e 00 5b cc 19 | ....09..!~[...] |
| 0030 | 50 11 fa d9 cd e5 00 00 |                         | P.....          |

ומקבלים הודעת ACK מהשרת, ונשים לב שמספר הACK גדל באחד (למרות שההודעה ריקה):

|  |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
|--|--------------|--------------|-----------|-----|----|-------|----|-------|-------|-------------|----------------|----|----|----|----|----|-----------------|
| 40   | 36.392359510 | 192.168.1.20 | 10.0.2.15 | TCP | 62 | 12345 | -  | 39306 | [ACK] | Seq=6016025 | Ack=3069583743 |    |    |    |    |    |                 |
| .....  |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| Frame 40: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0                     |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| Linux cooked capture   |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15   |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016025, Ack: 3069583743, Len: 0 |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| VSS-Monitoring ethernet trailer, Source Port: 0  |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| .....  |              |              |           |     |    |       |    |       |       |             |                |    |    |    |    |    |                 |
| 000  | 00           | 00           | 00        | 01  | 00 | 06    | 52 | 54    | 00    | 12          | 35             | 02 | 00 | 00 | 08 | 00 | .....RT..5..... |
| 010  | 45           | 00           | 00        | 28  | 02 | 38    | 00 | 00    | 40    | 06          | aa             | cd | c0 | a8 | 01 | 14 | E..(·8..@.....  |
| 020  | 0a           | 00           | 02        | 0f  | 30 | 39    | 99 | 8a    | 00    | 5b          | cc             | 19 | b6 | f6 | 21 | 7f | ....09..[....!~ |
| 030  | 50           | 10           | ff        | ff  | 73 | 5b    | 00 | 00    | 00    | 00          | 00             | 00 | 00 | 00 | 00 | 00 | P...s[.....     |

אך מאחר והשרת פעל על python3 ועבור python3 יש הבדל בין "" לבין "b, אז השרת לא מצליח להגיע לפקודת closen, ולכן ברגע שסגרנו את השרת ידנית הוא שלח לנו הודעת RST:

|  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|
| 41 38.577509633 192.168.1.20 10.0.2.15 TCP 62 12345 → 39306 [RST, ACK] Seq=6016025 Ack=30695           |  |  |  |  |  |  |  |  |  |  |  |
| Frame 40: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0                     |  |  |  |  |  |  |  |  |  |  |  |
| Linux cooked capture   |  |  |  |  |  |  |  |  |  |  |  |
| Internet Protocol Version 4, Src: 192.168.1.20, Dst: 10.0.2.15   |  |  |  |  |  |  |  |  |  |  |  |
| Transmission Control Protocol, Src Port: 12345, Dst Port: 39306, Seq: 6016025, Ack: 3069583743, Len: 0 |  |  |  |  |  |  |  |  |  |  |  |
| VSS-Monitoring ethernet trailer, Source Port: 0  |  |  |  |  |  |  |  |  |  |  |  |
| .....  |  |  |  |  |  |  |  |  |  |  |  |
| 000 00 00 00 01 00 06 52 54 00 12 35 02 00 00 08 00 .....RT..5.....                                    |  |  |  |  |  |  |  |  |  |  |  |
| 010 45 00 00 28 02 38 00 00 40 06 aa cd c0 a8 01 14 E..(8..@.....                                      |  |  |  |  |  |  |  |  |  |  |  |
| 020 0a 00 02 0f 30 39 99 8a 00 5b cc 19 b6 f6 21 7f ...09...[...!..                                    |  |  |  |  |  |  |  |  |  |  |  |
| 030 50 10 ff ff 73 5b 00 00 00 00 00 00 00 00 P..s[... ..  |  |  |  |  |  |  |  |  |  |  |  |

2.

קוד השרת של v1 פותח שרת TCP על כתובת IP קבועה (שאותה שיניתי ל 192.168.1.20) ועל מספר פורט שאותו הוא מקבל כפרמטר למחלקת main. הוא מחכה לחיבור לקוח וכאשר לקוח מתחבר הוא מדפיס הודעה עם כתובת הלקוח. הוא מחכה להודעות מהלקוח עד שהוא יקבל הודעה ריקה, ועבור כל הודעה שהוא מקבל הוא שולח אותה בחזרה ללקוח באותיות גדולות. ברגע שהוא מקבל הודעה ריקה מהלקוח הוא סוגר את החיבור.

קוד הלקוח של v1 פותח חיבור TCP לשרת שאת כתובת ה IP ומספר ה port שלו הוא מקבל כפרמטרים למחלקת main, ולאחר מכן שולח הודעת "Hello, World!" אליו ומחכה לתשובה. ברגע שהוא מקבל תשובה הוא מדפיס אותה למשתמש וסוגר את החיבור.

נראה את רצף ההודעות בv1:

|  |             |              |              |     |    |       |   |       |            |       |
|--|-------------|--------------|--------------|-----|----|-------|---|-------|------------|-------|
| 16   | 8.604567643 | 10.0.2.15    | 192.168.1.20 | TCP | 76 | 40956 | → | 5402  | [SYN]      | Seq=2 |
| 17   | 8.605153056 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402  | → | 40956 | [SYN, ACK] |       |
| 18   | 8.605176648 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40956 | → | 5402  | [ACK]      | Seq=2 |
| 19   | 8.605276837 | 10.0.2.15    | 192.168.1.20 | TCP | 69 | 40956 | → | 5402  | [PSH, ACK] |       |
| 20   | 8.605450451 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402  | → | 40956 | [ACK]      | Seq=2 |
| 21   | 8.605758877 | 192.168.1.20 | 10.0.2.15    | TCP | 69 | 5402  | → | 40956 | [PSH, ACK] |       |
| 22   | 8.605765902 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40956 | → | 5402  | [ACK]      | Seq=2 |
| 23   | 8.605939376 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40956 | → | 5402  | [FIN, ACK] |       |
| 24   | 8.606099712 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402  | → | 40956 | [ACK]      | Seq=2 |
| 25   | 8.606118482 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402  | → | 40956 | [FIN, ACK] |       |
| 26   | 8.606127141 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40956 | → | 5402  | [ACK]      | Seq=2 |
| [SYN] Seq=277376446 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1597563522 TSecr=0 WS=128 |             |              |              |     |    |       |   |       |            |       |
| [SYN, ACK] Seq=26880001 Ack=277376447 Win=65535 Len=0 MSS=1460                           |             |              |              |     |    |       |   |       |            |       |
| [ACK] Seq=277376447 Ack=26880002 Win=64240 Len=0   |             |              |              |     |    |       |   |       |            |       |
| [PSH, ACK] Seq=277376447 Ack=26880002 Win=64240 Len=13                                   |             |              |              |     |    |       |   |       |            |       |
| [ACK] Seq=26880002 Ack=277376460 Win=65535 Len=0   |             |              |              |     |    |       |   |       |            |       |
| [PSH, ACK] Seq=26880002 Ack=277376460 Win=65535 Len=13                                   |             |              |              |     |    |       |   |       |            |       |
| [ACK] Seq=277376460 Ack=26880015 Win=64227 Len=0   |             |              |              |     |    |       |   |       |            |       |
| [FIN, ACK] Seq=277376460 Ack=26880015 Win=64227 Len=0                                    |             |              |              |     |    |       |   |       |            |       |
| [ACK] Seq=26880015 Ack=277376461 Win=65535 Len=0   |             |              |              |     |    |       |   |       |            |       |
| [FIN, ACK] Seq=26880015 Ack=277376461 Win=65535 Len=0                                    |             |              |              |     |    |       |   |       |            |       |
| [ACK] Seq=277376461 Ack=26880016 Win=64227 Len=0   |             |              |              |     |    |       |   |       |            |       |

- שלחנו הודעת SYN לשרת
- קיבלנו SYN,ACK מהשרת
- החזרנו ACK לשרת על הSYN
- שלחנו הודעת Hello, World! לשרת
- קיבלנו ACK מהשרת
- קיבלנו הודעת HELLO, WORLD! מהשרת
- החזרנו ACK לשרת
- שלחנו הודעת FIN לשרת
- קיבלנו ACK מהשרת
- קיבלנו הודעת FIN מהשרת
- שלחנו ACK לשרת, וסיימנו את החיבור.
- רצף רגיל מאוד עבור חיבור TCP.



נסתכל עכשיו על v2 וננסה לראות אם משהו השתנה:

קוד השרת של v2 פועל באופן זהה לפעולתו של v1, עם שינוי יחיד: גודל הבאפר של הקריאה מהלקוח הוא 5 בייט בלבד. קוד הלקוח זהה לקוד הלקוח של v1 גם פה עם שינוי יחיד, ההודעה שאותה שולח הלקוח לשרת היא "World! Hello, World!". נריץ את הקוד ונראה את ההתנהגות

wireshark:

|        |             |              |              |     |    |              |            |      |
|--------|-------------|--------------|--------------|-----|----|--------------|------------|------|
| Number | 000000      | 10.0.2.15    | 192.168.1.20 | TCP | 76 | 40968 → 5402 | [SYN]      | Seq= |
|        | 489967      | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [SYN, ACK] |      |
| 3      | 0.000517482 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |
| 4      | 0.004723364 | 10.0.2.15    | 192.168.1.20 | TCP | 76 | 40968 → 5402 | [PSH, ACK] |      |
| 5      | 0.004980440 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [ACK]      | Seq= |
| 6      | 0.005117472 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [PSH, ACK] |      |
| 7      | 0.005127667 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |
| 8      | 0.005177530 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [PSH, ACK] |      |
| 9      | 0.005186118 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |
| 10     | 0.005340246 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [PSH, ACK] |      |
| 11     | 0.005348311 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |
| 12     | 0.005392359 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [PSH, ACK] |      |
| 13     | 0.005399585 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |
| 14     | 0.005573968 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [FIN, ACK] |      |
| 15     | 0.005715136 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [ACK]      | Seq= |
| 16     | 0.005814100 | 192.168.1.20 | 10.0.2.15    | TCP | 62 | 5402 → 40968 | [FIN, ACK] |      |
| 17     | 0.005861151 | 10.0.2.15    | 192.168.1.20 | TCP | 56 | 40968 → 5402 | [ACK]      | Seq= |

  

|            |                |                |           |          |             |                  |         |        |
|------------|----------------|----------------|-----------|----------|-------------|------------------|---------|--------|
| [SYN]      | Seq=3848888091 | Win=64240      | Len=0     | MSS=1460 | SACK_PERM=1 | TSval=1598786696 | TSecr=0 | WS=128 |
| [SYN, ACK] | Seq=31488001   | Ack=3848888092 | Win=65535 | Len=0    | MSS=1460    |                  |         |        |
| [ACK]      | Seq=3848888092 | Ack=31488002   | Win=64240 | Len=0    |             |                  |         |        |
| [PSH, ACK] | Seq=3848888092 | Ack=31488002   | Win=64240 | Len=20   |             |                  |         |        |
| [ACK]      | Seq=31488002   | Ack=3848888112 | Win=65535 | Len=0    |             |                  |         |        |
| [PSH, ACK] | Seq=31488002   | Ack=3848888112 | Win=65535 | Len=5    |             |                  |         |        |
| [ACK]      | Seq=3848888112 | Ack=31488007   | Win=64235 | Len=0    |             |                  |         |        |
| [PSH, ACK] | Seq=31488007   | Ack=3848888112 | Win=65535 | Len=5    |             |                  |         |        |
| [ACK]      | Seq=3848888112 | Ack=31488012   | Win=64230 | Len=0    |             |                  |         |        |
| [PSH, ACK] | Seq=31488012   | Ack=3848888112 | Win=65535 | Len=5    |             |                  |         |        |
| [ACK]      | Seq=3848888112 | Ack=31488017   | Win=64225 | Len=0    |             |                  |         |        |
| [PSH, ACK] | Seq=31488017   | Ack=3848888112 | Win=65535 | Len=5    |             |                  |         |        |
| [ACK]      | Seq=3848888112 | Ack=31488022   | Win=64220 | Len=0    |             |                  |         |        |
| [FIN, ACK] | Seq=3848888112 | Ack=31488022   | Win=64220 | Len=0    |             |                  |         |        |
| [ACK]      | Seq=31488022   | Ack=3848888113 | Win=65535 | Len=0    |             |                  |         |        |
| [FIN, ACK] | Seq=31488022   | Ack=3848888113 | Win=65535 | Len=0    |             |                  |         |        |
| [ACK]      | Seq=3848888113 | Ack=31488023   | Win=64219 | Len=0    |             |                  |         |        |

- ביצענו את ההתחברות לשרת על ידי ה-SYN, קיבלנו SYN,ACK ושלחנו ACK.
- עכשיו שלחנו לשרת את ההודעה "World! Hello, World!" באורך 20 בתים, וקיבלנו ACK מהשרת.
- השרת עכשיו מחזיר לנו את ההודעה בחזרה, אך במקום לשלוח אותה בהודעה אחת כמו בv1 הוא מחלק אותה ל4 הודעות שונות, כל אחת בגודל 5 בתים.
- סיימנו את ההעברה ושלחנו הודעת FIN לשרת, קיבלנו ACK, קיבלנו FIN,ACK והחזרנו ACK.

עכשיו נסתכל על v3:

קוד השרת בv3 פותח חיבור TCP על כתובת IP קבועה ומספר פורט שנקבע על ידי המשתמש, מחכה לחיבור משתמש, ברגע שמתקבל חיבור הוא מדפיס הודעה עם פרטי הלקוח, מחכה להודעה מהלקוח, וכל עוד לא קיבל הודעה ריקה מחזיר את ההודעות שקיבל ללקוח עם אותיות גדולות, מוכפלות ב1000.

קוד הלקוח מתחבר לשרת עם כתובת IP ופורט אותו קיבל כפרמטרים, שולח הודעת "Hello, World!" לשרת, ועכשיו קורא שני הודעות מהשרת כל אחת באורך מקסימלי של 1024 בתים, ומדפיס אותם.

נריץ ונראה את פעולתו על wireshark:

|   |             |              |              |     |      |              |            |       |
|---|-------------|--------------|--------------|-----|------|--------------|------------|-------|
| 1   | 0.000000000 | 10.0.2.15    | 192.168.1.20 | TCP | 76   | 40974 → 5402 | [SYN]      | Seq=4 |
| 2   | 0.000655937 | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40974 | [SYN, ACK] |       |
| 3   | 0.000678607 | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40974 → 5402 | [ACK]      | Seq=4 |
| 4   | 0.000779408 | 10.0.2.15    | 192.168.1.20 | TCP | 69   | 40974 → 5402 | [PSH, ACK] |       |
| 5   | 0.000943249 | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40974 | [ACK]      | Seq=3 |
| 6   | 0.001292915 | 192.168.1.20 | 10.0.2.15    | TCP | 2... | 5402 → 40974 | [ACK]      | Seq=3 |
| 7   | 0.001300777 | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40974 → 5402 | [ACK]      | Seq=4 |
| 8   | 0.001584602 | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40974 → 5402 | [RST, ACK] |       |
| 9   | 0.003991948 | 192.168.1.20 | 10.0.2.15    | TCP | 4... | 5402 → 40974 | [ACK]      | Seq=3 |
| 10  | 0.004010694 | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40974 → 5402 | [RST]      | Seq=4 |
| 11  | 0.004861433 | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40974 | [RST, ACK] |       |
| [SYN] Seq=4236563249 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1599654265 TSecr=0 WS=128 |             |              |              |     |      |              |            |       |
| [SYN, ACK] Seq=33152001 Ack=4236563250 Win=65535 Len=0 MSS=1460                           |             |              |              |     |      |              |            |       |
| [ACK] Seq=4236563250 Ack=33152002 Win=64240 Len=0   |             |              |              |     |      |              |            |       |
| [PSH, ACK] Seq=4236563250 Ack=33152002 Win=64240 Len=13                                   |             |              |              |     |      |              |            |       |
| [ACK] Seq=33152002 Ack=4236563263 Win=65535 Len=0   |             |              |              |     |      |              |            |       |
| [ACK] Seq=33152002 Ack=4236563263 Win=65535 Len=2840                                      |             |              |              |     |      |              |            |       |
| [ACK] Seq=4236563263 Ack=33154842 Win=62480 Len=0   |             |              |              |     |      |              |            |       |
| [RST, ACK] Seq=4236563263 Ack=33154842 Win=63900 Len=0                                    |             |              |              |     |      |              |            |       |
| [ACK] Seq=33154842 Ack=4236563263 Win=65535 Len=4260                                      |             |              |              |     |      |              |            |       |
| [RST] Seq=4236563263 Win=0 Len=0  |             |              |              |     |      |              |            |       |
| [RST, ACK] Seq=0 Ack=4236563263 Win=0 Len=0   |             |              |              |     |      |              |            |       |

- החיבור התחיל כרגיל עם הודעת SYN, תשובה SYN,ACK ותשובה ACK.
- הלקוח שלח לשרת הודעת "Hello, World!" באורך 13, וקיבל ACK בתשובה מהשרת.
- השרת עכשיו מנסה לשלוח הודעה באורך 2840 עליה הלקוח מחזיר ACK. נשים לב שגודל ה MSS הוא 1460, ולכן אנחנו יכולים להניח שמדובר בכמה הודעות שחוברו.
- הלקוח שקרא שני הודעות מהבאפר מנסה לסגור את החיבור, אך משום מה לא נשלחת הודעת FIN וישר הוא שולח הודעת RST שעליה השרת במקום לסגור מנסה לשלוח עוד חבילות.
- הלקוח רואה שלא קיבל תשובה, ושולח הודעת RST נוספת.
- השרת מקבל ושולח RST,ACK ונסגר החיבור.

נסתכל על v4:

קוד השרת של v4 פותח חיבור TCP, מחכה ללקוח וברגע שיש חיבור עם לקוח, כל 5 שניות מנסה לקרוא מהלקוח, וכל עוד קיבל הודעה לא ריקה שולח אותה בחזרה ללקוח עם אותיות גדולות וחוזר לחכות 5 שניות נוספות וחוזר חלילה עד הגעת הודעה ריקה. קוד הלקוח של v4 פותח חיבור TCP לשרת ושולח 4 פעמים הודעות באורך 130 בתים ("Hello, World!" \* 10), קורא תשובה מהשרת ואז סוגר את החיבור.

נראה את ההרצה:

|    |              |              |              |     |      |              |            |       |
|----|--------------|--------------|--------------|-----|------|--------------|------------|-------|
| 1  | 0.000000000  | 10.0.2.15    | 192.168.1.20 | TCP | 76   | 40982 → 5402 | [SYN]      | Seq=1 |
| 2  | 0.000625670  | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [SYN, ACK] |       |
| 3  | 0.000652744  | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40982 → 5402 | [ACK]      | Seq=1 |
| 4  | 0.000760161  | 10.0.2.15    | 192.168.1.20 | TCP | 1... | 40982 → 5402 | [PSH, ACK] |       |
| 5  | 0.000909981  | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [ACK]      | Seq=1 |
| 6  | 0.000978485  | 10.0.2.15    | 192.168.1.20 | TCP | 1... | 40982 → 5402 | [PSH, ACK] |       |
| 7  | 0.001188927  | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [ACK]      | Seq=1 |
| 8  | 0.001197848  | 10.0.2.15    | 192.168.1.20 | TCP | 3... | 40982 → 5402 | [PSH, ACK] |       |
| 9  | 0.001326447  | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [ACK]      | Seq=1 |
| 10 | 5.001710231  | 192.168.1.20 | 10.0.2.15    | TCP | 5... | 5402 → 40982 | [PSH, ACK] |       |
| 11 | 5.001746886  | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40982 → 5402 | [ACK]      | Seq=1 |
| 12 | 5.002010535  | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40982 → 5402 | [FIN, ACK] |       |
| 13 | 5.006471890  | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [ACK]      | Seq=1 |
| 14 | 10.002360232 | 192.168.1.20 | 10.0.2.15    | TCP | 62   | 5402 → 40982 | [FIN, ACK] |       |
| 15 | 10.002385413 | 10.0.2.15    | 192.168.1.20 | TCP | 56   | 40982 → 5402 | [ACK]      | Seq=1 |

```

[SYN] Seq=2468301588 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1600541405 TSecr=0 WS=128
[SYN, ACK] Seq=34944001 Ack=2468301589 Win=65535 Len=0 MSS=1460
[ACK] Seq=2468301589 Ack=34944002 Win=64240 Len=0
[PSH, ACK] Seq=2468301589 Ack=34944002 Win=64240 Len=130
[ACK] Seq=34944002 Ack=2468301719 Win=65535 Len=0
[PSH, ACK] Seq=2468301719 Ack=34944002 Win=64240 Len=130
[ACK] Seq=34944002 Ack=2468301849 Win=65535 Len=0
[PSH, ACK] Seq=2468301849 Ack=34944002 Win=64240 Len=260
[ACK] Seq=34944002 Ack=2468302109 Win=65535 Len=0
[PSH, ACK] Seq=34944002 Ack=2468302109 Win=65535 Len=520
[ACK] Seq=2468302109 Ack=34944522 Win=63784 Len=0
[FIN, ACK] Seq=2468302109 Ack=34944522 Win=63784 Len=0
[ACK] Seq=34944522 Ack=2468302110 Win=65535 Len=0
[FIN, ACK] Seq=34944522 Ack=2468302110 Win=65535 Len=0
[ACK] Seq=2468302110 Ack=34944523 Win=63784 Len=0

```

- א. הלקוח מבצע חיבור לשרת על ידי SYN,ACK <- ACK.
- ב. הלקוח שולח לשרת הודעה באורך 130 ומקבל ACK מהשרת.
- ג. הלקוח שולח לשרת הודעה נוספת באורך 130 ומקבל ACK מהשרת.
- ד. הלקוח שולח הודעה באורך 260 ומקבל ACK מהשרת.
- נשים לב שלמרות שבאפליקציה שלחנו שני חבילות נפרדות, שכבת התעבורה החליטה לחבר אותן לחבילה אחת.
- ה. השרת שולח ללקוח הודעה אחת באורך 520 בתים ומקבל ACK מהלקוח.
- ו. הלקוח והשרת מסיימים את החיבור על ידי ACK<-FIN , ACK <- FIN.

נשים לב שלמרות שהלקוח שלח 5 הודעות נפרדות, השרת שלח ללקוח הודעה אחת בלבד, מאחר שהוא חיכה 5 שניות, אז בינתיים הלקוח הספיק לשלוח את כל ההודעות שלו, שחיכו לשרת בבאפר עד שקרא אותן באפליקציה.