

רשתות תקשורת מטלה 4

מבוא

פרויקט זה נועד להעמיק את ההבנה בשכבת הרשת ובפרוטוקולים המרכזיים המופעלים בה, תוך מימוש כלים המיועדים לאבחון ותפעול של רשתות תקשורת. המטרה העיקרית היא להכיר מקרוב את הפרוטוקולים ICMP ו-IP ואת השימושים שלהם במגוון יישומים.

הפרויקט מתמקד במימוש כלים מבוססי רשת המשתמשים בפרוטוקולים אלו לצורך אבחון תעבורה ברשת. הכלים כוללים:

- **Ping:** הכלי נועד לבדוק זמינות וקישוריות בין מחשבים ברשת על ידי שליחת בקשות ICMP Echo Request וניתוח תגובות Echo Reply. התוכנית ממומשת בשפת C ומשתמשת בפרוטוקול IPv4, עם אפשרות להגדיר כתובת יעד, סוג פרוטוקול, מספר בקשות ושליחה ללא השהיה.
- **Traceroute:** הכלי מזהה את מסלול החבילות שעוברות בין מחשב מקומי ליעד על ידי שינוי ערכי TTL והמתנה לתגובות מסוג Time Exceeded או Echo Reply. הכלי ממומש בשפת C ומדווח על הכתובות והזמנים של כל התחנות בדרך.
- **Network Scanner:** הכלי מאפשר סריקת רשת לזיהוי כתובות IP פעילות בטווח מוגדר. הוא שולח בקשות ICMP לכל כתובת בטווח ומציג את הכתובות הפעילות בלבד. מימוש הכלי כולל שימוש בפרמטרים של כתובת רשת ומסכת רשת.

בנוסף, בוצעו בדיקות באמצעות Wireshark לצורך וידוא נכונות הפעולה. במימוש כלים מבוססי רשת המשתמשים בפרוטוקולים ICMP ו-IP לצורך אבחון תעבורה ברשת.

תוכן עניינים:

| | |
|---------|--|
| 1..... | מבוא |
| 2..... | תוכן עיניים: |
| 3..... | חלק א': Ping |
| 7..... | חלק ב': Traceroute |
| 14..... | חלק ג': Network Scanner |
| 21..... | שאלות שנשאלו עם ה AI לטובת הבנת המטלה: |

Ping א': חלק

מטרות הכלי: כלי זה נועד לשלוח בקשות Echo Request לכתובת IP ולנתח את התגובות המתקבלות (Echo Reply). הכלי תומך ב: פרוטוקול IPv4 בלבד.

מבנה הפקודה: הכלי מופעל באמצעות הפקודה הבאה:

```
$ sudo ./ping -a <Ip Address> -c <num of requests> (לא חובה) -t <Ip version> -f (לא חובה)
```

פרמטרים נתמכים:

- **-a:** כתובת ה-IP של היעד שאליו מבוצעת הבדיקה (חובה).
- **-t:** גרסת פרוטוקול ה-IP או 4 או 6. (אם תתקבל מספר לא נכון נראה הודעת שגיאה ונעצור את התכנית)
- **-c:** מספר הבקשות לשליחה (אופציונלי, ברירת מחדל היא לשלוח חבילות עד עצירה ידנית).
- **-f:** שליחה מהירה (אופציונלי, ללא ההשהיה בין הבקשות).

מימוש: הקוד ממומש בשפת C תוך שימוש במבני נתונים של פרוטוקול ICMP. לדוגמה:

```
struct icmphdr header;  
header.type = ICMP_ECHO;  
header.code = 0;  
header.checksum = calculate_checksum(&header, sizeof(header));
```

הקוד כולל שימוש ב-`getopt` לצורך ניתוח פרמטרים מהשורה, כמו גם שימוש בפקודות `poll` כדי לנהל זמן המתנה לתשובות. הכלי מטפל גם בקטיעת התהליך על ידי הצגת סטטיסטיקות של החבילות שנשלחו והתקבלו.

פלט לדוגמה:

```
dor@dor-virtual-machine:~/Documents/Networks/part A$ sudo ./ping -a 1.1.1.1 -t 4 -c 5  
PING 1.1.1.1 with 64 bytes of data:  
64 bytes from 1.1.1.1: icmp_seq=0 ttl=58 time=6.32ms  
64 bytes from 1.1.1.1: icmp_seq=1 ttl=58 time=9.10ms  
64 bytes from 1.1.1.1: icmp_seq=2 ttl=58 time=8.18ms  
64 bytes from 1.1.1.1: icmp_seq=3 ttl=58 time=8.88ms  
64 bytes from 1.1.1.1: icmp_seq=4 ttl=58 time=6.78ms  
  
--- 1.1.1.1 ping statistics ---  
5 packets transmitted, 5 received , time 39.26ms  
rtt min/avg/max/mdev = 6.319/7.851/9.100/1.114 ms
```

בדיקות ב-Wireshark: לצורך וידוא נכונות המימוש, נעשה שימוש ב-Wireshark לצפייה בתעבורה. הועברו בקשות Echo Request ותועדו תגובות Echo Reply. צילומי המסך מציגים את מבנה החבילות והזמנים שנמדדו בפועל.

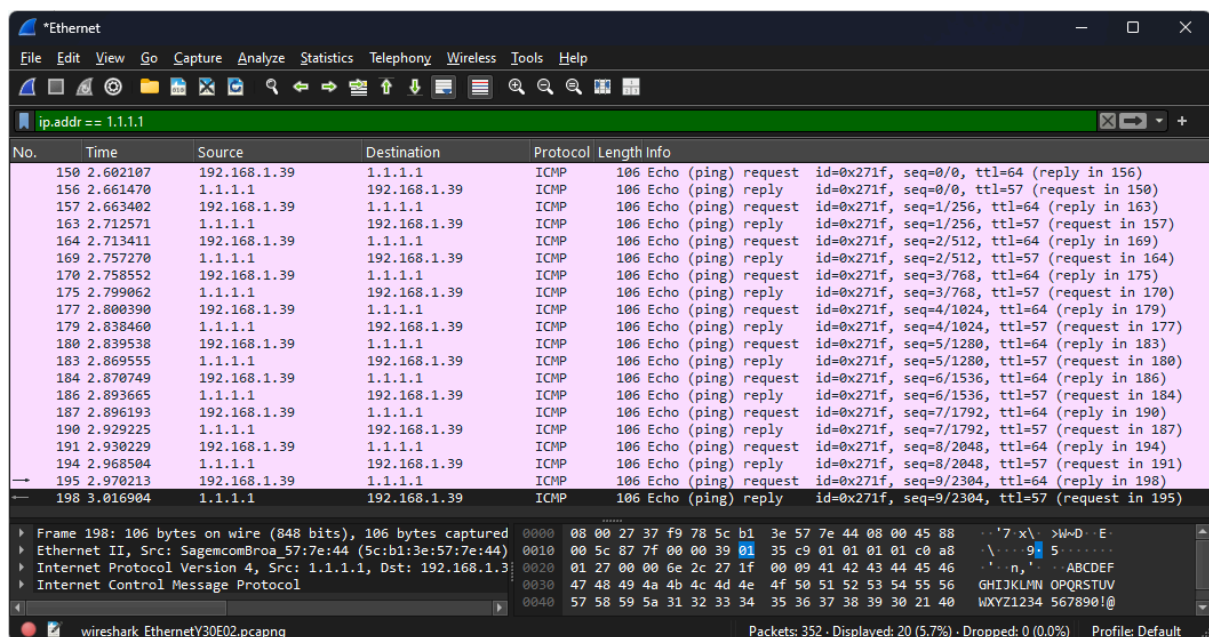
נעבור על כמה מקרים, כולל מקרי בסיס ומקרי קצה, להלן המקרים:

1. בדיקה בסיסית- שליחת פינג לכתובת 1.1.1.1, הגדרת סוג פרוטוקול ל- 4 הגבלה ל- 10 הודעות והדלקת דגל f.

נבצע בדיקה לקבלת תגובות Echo Reply. בדיקה זו מאשרת את הבסיסיות של הכלי ואת התאימות שלו לפרוטוקול.

```
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part A$ sudo ./ping -a 1.1.1.1 -c 10 -t 4 -f
PING 1.1.1.1 with 64 bytes of data:
64 bytes from 1.1.1.1: icmp_seq=0 ttl=57 time=60.67ms
64 bytes from 1.1.1.1: icmp_seq=1 ttl=57 time=50.32ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=57 time=44.86ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=57 time=41.80ms
64 bytes from 1.1.1.1: icmp_seq=4 ttl=57 time=39.09ms
64 bytes from 1.1.1.1: icmp_seq=5 ttl=57 time=30.74ms
64 bytes from 1.1.1.1: icmp_seq=6 ttl=57 time=25.59ms
64 bytes from 1.1.1.1: icmp_seq=7 ttl=57 time=34.05ms
64 bytes from 1.1.1.1: icmp_seq=8 ttl=57 time=39.34ms
64 bytes from 1.1.1.1: icmp_seq=9 ttl=57 time=48.14ms

--- 1.1.1.1 ping statistics ---
10 packets transmitted, 10 received, time 414.60ms
rtt min/avg/max/mdev = 25.594/41.460/60.675/9.673 ms
```



2. בדיקת קצה - שליחת פינג לכתובת 1.2.1.2, הגדרת סוג פרוטוקול ל - 4 הגבלה ל- 10 הודעות

שליחת בקשות ICMP לכתובת שאינה קיימת ברשת ובדיקת הופעת הודעות שגיאה מתאימות (למשל, Destination Unreachable).

צילום wireshark:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|-------------|--------------------|-------------|----------|--------|---|
| 22 | 4.167958610 | dor-virtual-mac... | 1.2.1.2 | ICMP | 106 | Echo (ping) request id=0x3f79, seq=0/0, ttl=64 (no response found!) |
| 28 | 6.169640975 | dor-virtual-mac... | 1.2.1.2 | ICMP | 106 | Echo (ping) request id=0x3f79, seq=0/0, ttl=64 (no response found!) |
| 36 | 8.172299034 | dor-virtual-mac... | 1.2.1.2 | ICMP | 106 | Echo (ping) request id=0x3f79, seq=0/0, ttl=64 (no response found!) |

נסתכל על אחת הפקטות כי כולם אותו הדבר וננסה להבין מה היא מכילה, נסתכל על חלק ה ICMP

Internet Control Message Protocol

Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0x4ddb [correct]
[Checksum Status: Good]
Identifier (BE): 16249 (0x3f79)
Identifier (LE): 31039 (0x793f)
Sequence Number (BE): 0 (0x0000)
Sequence Number (LE): 0 (0x0000)
[No response seen]
[Expert Info (Warning/Sequence): No response seen to ICMP request]

Type 8 - Echo Request

סוג ההודעה הוא "Echo Request". זהו חלק מפרוטוקול ה-ICMP.

Code: 0: קוד 0, משמעותו היא שהודעת ה-Echo Request תקינה.

Checksum correct: בדיקת תקינות עבור ההודעה. במקרה זה, ה-Checksum תקין.

Identifiers: ה-Identifier הוא מזהה ייחודי להודעה זו. הוא משמש להתאמת תגובות Echo Reply להודעות Echo Request.

Sequence Number: מספר סידורי המאפשר לעקוב אחרי ההודעות שנשלחו והתקבלו. במקרה זה המספר הוא 0 וכך נמשיך עד הסוף..

[No response seen]: כאן Wireshark מתריע שלא נצפתה תגובה עבור בקשת ה-Echo Request. ייתכן שהיעד לא זמין, או שהחבילה אבדה ברשת.

[Expert Info]: הודעה שמספקת מידע נוסף על כך שלא נצפתה תגובה.

צילום Terminal:


```
❶ dor@dor-virtual-machine:~/Documents/Networks/part A$ sudo ./ping -a 1.2.1.2 -t 4 -c 10
[sudo] password for dor:
PING 1.2.1.2 with 64 bytes of data:
Request timeout for icmp_seq 1, retrying...
Request timeout for icmp_seq 1, retrying...
Request timeout for icmp_seq 1, aborting.

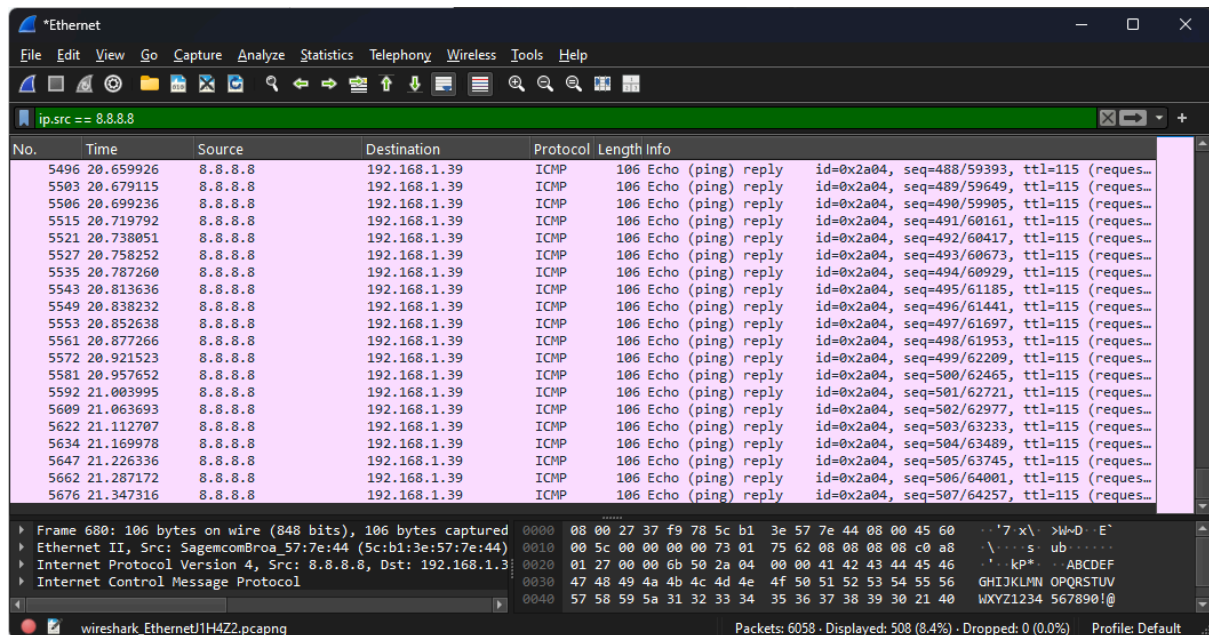
--- 1.2.1.2 ping statistics ---
1 packets transmitted, 0 received, time 0.00ms
No packets received, no RTT statistics available.
```

3. בדיקת עומס - שליחת פינג לכתובת 8.8.8.8, הגדרת סוג פרוטוקול ל - 4 לא מוגבל להודעות והדלקת דגל f.

הבדיקה מספקת תובנות לגבי יציבות הכלי בתנאי עומס גבוה. בבדיקה ניתן לראות מימוש של עצירת התוכנית עם `ctrl + c` וקבלת סטטיסטיקה של ההודעות כנדרש.

מכיוון שגודל ההודעות ארוך מהרגיל, הוספנו קובץ `txt` עם התגובות שקיבלנו (בדיקה של לפחות 500 הודעות).

 `ping_8.8.8_extreme_case_3.txt`



| No. | Time | Source | Destination | Protocol | Length | Info |
|------|-----------|---------|--------------|----------|--------|--|
| 5496 | 20.659926 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=488/59393, ttl=115 (request...) |
| 5503 | 20.679115 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=489/59649, ttl=115 (request...) |
| 5506 | 20.699236 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=490/59905, ttl=115 (request...) |
| 5515 | 20.719792 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=491/60161, ttl=115 (request...) |
| 5521 | 20.738051 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=492/60417, ttl=115 (request...) |
| 5527 | 20.758252 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=493/60673, ttl=115 (request...) |
| 5535 | 20.787260 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=494/60929, ttl=115 (request...) |
| 5543 | 20.813636 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=495/61185, ttl=115 (request...) |
| 5549 | 20.838232 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=496/61441, ttl=115 (request...) |
| 5553 | 20.852638 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=497/61697, ttl=115 (request...) |
| 5561 | 20.877266 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=498/61953, ttl=115 (request...) |
| 5572 | 20.921523 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=499/62209, ttl=115 (request...) |
| 5581 | 20.957652 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=500/62465, ttl=115 (request...) |
| 5592 | 21.003995 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=501/62721, ttl=115 (request...) |
| 5609 | 21.063693 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=502/62977, ttl=115 (request...) |
| 5622 | 21.112707 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=503/63233, ttl=115 (request...) |
| 5634 | 21.169978 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=504/63489, ttl=115 (request...) |
| 5647 | 21.226336 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=505/63745, ttl=115 (request...) |
| 5662 | 21.287172 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=506/64001, ttl=115 (request...) |
| 5676 | 21.347316 | 8.8.8.8 | 192.168.1.39 | ICMP | 106 | Echo (ping) reply id=0x2a04, seq=507/64257, ttl=115 (request...) |

```
sudo ./ping -a 8.8.8.8 -t 4 -f
```

```
64 bytes from 8.8.8.8: icmp_seq=495 ttl=115 time=26.42ms
64 bytes from 8.8.8.8: icmp_seq=496 ttl=115 time=24.90ms
64 bytes from 8.8.8.8: icmp_seq=497 ttl=115 time=14.09ms
64 bytes from 8.8.8.8: icmp_seq=498 ttl=115 time=25.00ms
64 bytes from 8.8.8.8: icmp_seq=499 ttl=115 time=42.94ms
64 bytes from 8.8.8.8: icmp_seq=500 ttl=115 time=36.00ms
64 bytes from 8.8.8.8: icmp_seq=501 ttl=115 time=48.79ms
64 bytes from 8.8.8.8: icmp_seq=502 ttl=115 time=55.88ms
64 bytes from 8.8.8.8: icmp_seq=503 ttl=115 time=48.40ms
64 bytes from 8.8.8.8: icmp_seq=504 ttl=115 time=57.18ms
64 bytes from 8.8.8.8: icmp_seq=505 ttl=115 time=56.51ms
64 bytes from 8.8.8.8: icmp_seq=506 ttl=115 time=63.22ms
^C
--- 8.8.8.8 ping statistics ---
508 packets transmitted, 507 received , time 17978.24ms
rtt min/avg/max/mdev = 10.722/35.460/78.845/15.782 ms
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part A$
```

חלק ב': Traceroute

מטרות הכלי: כלי זה נועד לזהות את המסלול שעוברות החבילות מהרשת המקומית לכתובת היעד. המטרה היא להבין אילו נתבים או התקנים חוצות החבילות בדרכן ליעד, תוך שימוש בערך Time To Live (TTL) משתנה.

מבנה הפקודה: הכלי מופעל באמצעות הפקודה הבאה:

```
$ sudo ./traceroute -a <IP ADDRESS>
```

פרמטרים נתמכים:

- **a-** כתובת ה-IP של היעד שאליו מבוצעת הבדיקה.

מימוש: הכלי שולח בכל שלב שלוש בקשות ICMP עם ערכי TTL עוקבים, מתחיל ב-TTL=1 ומעלה את הערך בכל סבב. כאשר החבילה מגיעה לנתב ביניים, הנתב מחזיר תגובת Time Exceeded, ו-כשהחבילה מגיעה ליעד, מתקבלת תגובת Echo Reply.

במהלך המימוש נעשה שימוש בפרוטוקול IPv4 בלבד, תוך התמקדות בקבלת מידע מכל תחנת ביניים כמו כתובת ה-IP וזמן התגובה (RTT). התהליך מסתיים כאשר החבילה מגיעה ליעד או כאשר ערך ה-TTL מגיע למקסימום שהוגדר מראש (30) במידה והחבילה לא הגיעה ליעד תודפס הודעה בהתאם.

פלט לדוגמה:

```
dor@dor-virtual-machine:~/Documents/Networks/part B$ sudo ./traceroute -a 1.1.1.1
[sudo] password for dor:
traceroute to 1.1.1.1 with 30 hops max:
 1  10.100.102.1    4.324ms    1.763ms    2.248ms
 2  212.143.208.34   6.233ms    4.534ms    5.663ms
 3  212.143.203.219  5.399ms    6.212ms    7.773ms
 4  212.143.12.13    7.405ms    8.350ms    8.127ms
 5  212.143.7.64     7.901ms    8.797ms    8.522ms
 6  212.143.232.3    8.295ms    20.818ms   7.443ms
 7  1.1.1.1          6.562ms    9.980ms    8.282ms
```

בדיקות ב-Wireshark: ב-Wireshark תועדו חבילות ICMP עם ערכי TTL משתנים. ניתן היה לזהות את התגובות מכל נתב לאורך המסלול. צילומי המסך מציגים את הנתונים שנאספו ואת מבנה החבילות. ב-Wireshark תועדו חבילות ICMP עם ערכי TTL משתנים. ניתן היה לזהות את התגובות מכל נתב לאורך המסלול. צילומי המסך מציגים את הנתונים שנאספו ואת מבנה החבילות.

1.בדיקה בסיסית:ביצוע traceroute לכתובת 8.8.8.8

ביצוע Traceroute לכתובת ידועה ובדיקת כל תחנה במסלול דרך Wireshark. הבדיקה כוללת אימות תגובות Time Exceeded ו-Echo Reply.

צילום Terminal:

```
● dor@dor-virtual-machine:~/Documents/Networks/part B$ sudo ./traceroute -a 8.8.8.8
[sudo] password for dor:
traceroute to 8.8.8.8 with 30 hops max:

 1  10.100.102.1    2.922ms    1.558ms    1.634ms
 2  212.143.208.34  4.987ms    6.803ms    6.940ms
 3  212.143.203.223 6.792ms    5.167ms    8.114ms
 4  212.143.25.168  4.519ms    8.890ms    6.504ms
 5  192.178.69.74   6.986ms    5.454ms    7.238ms
 6  108.170.229.65  6.812ms    6.715ms    6.938ms
 7  142.251.228.197 4.878ms    7.458ms    7.275ms
 8  8.8.8.8         6.001ms    5.820ms    6.145ms
```

צילום wireshark:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|--------------|-----------------|----------------|----------|--------|--|
| 11 | 4.642319418 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=0/0, ttl=1 (no response found!) |
| 12 | 4.644193958 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 13 | 4.644430475 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=1/256, ttl=1 (no response found!) |
| 14 | 4.646000587 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 15 | 4.6464000587 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=2/512, ttl=1 (no response found!) |
| 16 | 4.647440436 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 17 | 4.647800325 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=3/768, ttl=2 (no response found!) |
| 18 | 4.652516767 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 19 | 4.652797259 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=4/1024, ttl=2 (no response found!) |
| 20 | 4.659387075 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 21 | 4.659612867 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=5/1280, ttl=2 (no response found!) |
| 22 | 4.666215384 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 23 | 4.666654213 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=6/1536, ttl=3 (no response found!) |
| 24 | 4.673237846 | 212.143.203.223 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 25 | 4.673455104 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=7/1792, ttl=3 (no response found!) |
| 26 | 4.6742433309 | 212.143.203.223 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 27 | 4.678635313 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=8/2048, ttl=3 (no response found!) |
| 28 | 4.684480620 | 212.143.203.223 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 29 | 4.686816222 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=9/2304, ttl=4 (no response found!) |
| 30 | 4.691192935 | 212.143.25.168 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 31 | 4.691336969 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=10/2560, ttl=4 (no response found!) |
| 32 | 4.698085850 | 212.143.25.168 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 33 | 4.700277845 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=11/2816, ttl=4 (no response found!) |
| 34 | 4.706584259 | 212.143.25.168 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 35 | 4.706829243 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=12/3072, ttl=5 (no response found!) |
| 36 | 4.713537129 | 192.178.69.74 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 37 | 4.713830117 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=13/3328, ttl=5 (no response found!) |
| 38 | 4.718992571 | 192.178.69.74 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 39 | 4.719285231 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=14/3584, ttl=5 (no response found!) |
| 40 | 4.726300282 | 192.178.69.74 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 41 | 4.726585897 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=15/3840, ttl=6 (no response found!) |
| 42 | 4.733106299 | 108.170.229.65 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 43 | 4.733430751 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=16/4096, ttl=6 (no response found!) |
| 44 | 4.739922537 | 108.170.229.65 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 45 | 4.740559683 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=17/4352, ttl=6 (no response found!) |
| 46 | 4.747164558 | 108.170.229.65 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 47 | 4.747646225 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=18/4608, ttl=7 (no response found!) |
| 48 | 4.752260618 | 142.251.228.197 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 49 | 4.752571225 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=19/4864, ttl=7 (no response found!) |
| 50 | 4.759613868 | 142.251.228.197 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 51 | 4.760047465 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=20/5120, ttl=7 (no response found!) |
| 52 | 4.760608950 | 142.251.228.197 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 53 | 4.767366581 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=21/5376, ttl=8 (reply in 54) |
| 54 | 4.773233776 | 8.8.8.8 | 10.100.102.121 | ICMP | 106 | Echo (ping) reply id=0x42a1, seq=21/5376, ttl=110 (request in 53) |
| 55 | 4.773370860 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=22/5632, ttl=8 (reply in 56) |
| 56 | 4.779012504 | 8.8.8.8 | 10.100.102.121 | ICMP | 106 | Echo (ping) reply id=0x42a1, seq=22/5632, ttl=110 (request in 55) |
| 57 | 4.779202150 | 10.100.102.1 | 8.8.8.8 | ICMP | 106 | Echo (ping) request id=0x42a1, seq=23/5888, ttl=8 (reply in 58) |
| 58 | 4.785008919 | 8.8.8.8 | 10.100.102.121 | ICMP | 106 | Echo (ping) reply id=0x42a1, seq=23/5888, ttl=110 (request in 57) |

בצילום wireshark מודגם מעקב אחר מסלול של הודעה שיוצאת מהמחשב שלי עד שהיא מגיעה לכתובת 8.8.8.8.

בכל איטרציה של תהליך ה-Traceroute, ה-TTL גדל באחד בכל סבב של שליחת חבילות. TTL הוא שדה בכתרת ה-IP שמגדיר את מספר הקפיצות (hops) שחבילה יכולה לעבור לפני ש"פג תוקפה".

התהליך מתחיל ב-TTL של 1, כך שהחבילה תגיע לנתב הראשון (hop הראשון). כאשר הנתב מזהה שה-TTL של החבילה הוא 1, הוא מחזיר תגובת ICMP מסוג "Time-to-live Exceeded". לאחר קבלת התגובה, התוכנית מגדילה את ערך ה-TTL ב-1 ומנסה לשלוח את החבילה שוב.

השלב הזה חוזר על עצמו, כאשר בכל סבב ערך ה-TTL גדל, מה שמאפשר לחבילה לעבור לנתב הבא במסלול. כך ממשיכים עד שהחבילה מגיעה ליעד הסופי (במקרה זה, 8.8.8.8), שבו מתקבלת תגובת ICMP מסוג "Echo Reply".

הודות לגישה זו, ניתן למפות את כל התחנות (נתבים) שהחבילה עברה דרכן בדרך ליעד.

בדיקת קצה: ביצוע Traceroute לכתובת 192.0.2.1

Traceroute לכתובת בלתי גנישה ברשת (למשל, 192.0.2.1) ואימות הופעת שגיאות או כשל תגובה במעבר דרך הנתבים.

צילום טרמינל:

```
dor@dor-virtual-machine:~/Documents/Networks/part B$ sudo ./traceroute -a 192.0.2.1
traceroute to 192.0.2.1 with 30 hops max:

 1  10.100.102.1    3.674ms    2.553ms    2.269ms

 2  212.143.208.34   6.612ms    7.353ms    18.779ms

 3  *      *      *

 4  *      *      *

 5  *      *      *

 6  *      *      *

 7  *      *      *

 8  *      *      *

 9  *      *      *
```

...

...

```
28 *      *      *

29 *      *      *

30 *      *      *
Destination Unreachable.
```

צילום wireshark:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|--------------|----------------|----------------|----------|--------|---|
| 14 | 2.128568478 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=9/8, ttl=1 (no response found) |
| 15 | 2.132071357 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (time to live exceeded in transit) |
| 16 | 2.132878777 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=1/256, ttl=1 (no response found) |
| 17 | 2.141580311 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (time to live exceeded in transit) |
| 18 | 2.134839396 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=2/512, ttl=1 (no response found) |
| 19 | 2.140700622 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (time to live exceeded in transit) |
| 20 | 2.137193595 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=3/768, ttl=2 (no response found) |
| 21 | 2.144181488 | 212.143.208.34 | 10.100.102.121 | ICMP | 118 | Time-to-live exceeded (time to live exceeded in transit) |
| 22 | 2.144531533 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=4/1024, ttl=2 (no response found) |
| 23 | 2.151105042 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (time to live exceeded in transit) |
| 24 | 2.151708326 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=5/1280, ttl=2 (no response found) |
| 25 | 2.146169462 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (time to live exceeded in transit) |
| 26 | 2.170011483 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=6/1536, ttl=3 (no response found) |
| 27 | 2.679954639 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=7/1792, ttl=3 (no response found) |
| 28 | 3.181380425 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=8/2048, ttl=3 (no response found) |
| 29 | 3.682897121 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=9/2304, ttl=4 (no response found) |
| 30 | 4.184992212 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=10/2560, ttl=4 (no response found) |
| 31 | 4.605377685 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=11/2816, ttl=4 (no response found) |
| 33 | 5.186844885 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=12/3072, ttl=5 (no response found) |
| 41 | 5.688155642 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=13/3328, ttl=5 (no response found) |
| 44 | 6.189128086 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=14/3584, ttl=5 (no response found) |
| 45 | 6.699590959 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=15/3840, ttl=6 (no response found) |
| 49 | 7.191516130 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=16/4096, ttl=6 (no response found) |
| 56 | 7.692915176 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=17/4352, ttl=6 (no response found) |
| 59 | 8.194222699 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=18/4608, ttl=7 (no response found) |
| 66 | 8.695137354 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=19/4864, ttl=7 (no response found) |
| 67 | 9.196133552 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=20/5120, ttl=7 (no response found) |
| 68 | 9.697636341 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=21/5376, ttl=8 (no response found) |
| 71 | 10.198988485 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=22/5632, ttl=8 (no response found) |
| 72 | 10.78944438 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=23/5888, ttl=8 (no response found) |
| 75 | 11.201609069 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=24/6144, ttl=9 (no response found) |
| 76 | 11.703164112 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=25/6400, ttl=9 (no response found) |
| 81 | 12.204929785 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=26/6656, ttl=9 (no response found) |
| 82 | 12.705136778 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=27/6912, ttl=10 (no response found) |
| 84 | 13.206860534 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=28/7168, ttl=10 (no response found) |
| 88 | 13.707162587 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=29/7424, ttl=10 (no response found) |
| 89 | 14.208393156 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=30/7680, ttl=11 (no response found) |
| 90 | 14.709384827 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=31/7936, ttl=11 (no response found) |
| 92 | 15.212744877 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=32/8192, ttl=11 (no response found) |
| 110 | 15.717415430 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=33/8448, ttl=12 (no response found) |
| 112 | 16.218333999 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=34/8704, ttl=12 (no response found) |
| 115 | 16.719170587 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=35/8960, ttl=12 (no response found) |
| 117 | 17.220880480 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=36/9216, ttl=13 (no response found) |
| 118 | 17.722196162 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=37/9472, ttl=13 (no response found) |
| 119 | 18.223138888 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=38/9728, ttl=13 (no response found) |
| 120 | 18.725402354 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=39/9984, ttl=14 (no response found) |
| 121 | 19.226768411 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=40/10240, ttl=14 (no response found) |
| 126 | 19.728660389 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=41/10496, ttl=14 (no response found) |
| 129 | 20.225969832 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=42/10752, ttl=15 (no response found) |
| 131 | 20.730385451 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=43/11008, ttl=15 (no response found) |
| 135 | 21.231184972 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=44/11264, ttl=15 (no response found) |
| 136 | 21.732525411 | 10.100.102.1 | 192.0.2.1 | ICMP | 106 | Echo (ping) request id=0x4555, seq=45/11520, ttl=16 (no response found) |

(צירפתי חלק מהחבילות מכיוון ששאר החבילות בהמשך חוזרות על עצמן עד שהttl מגיע ל 30 בקובץ עצמו יופיעו שאר החבילות)

במהלך תהליך ה-Traceroute, נשלחות חבילות ICMP מסוג Echo Request מכתובת המקור (10.100.102.121) אל כתובת היעד (192.0.2.1). בכל פעם, החבילות נשלחות עם ערך TTL שמתחיל ב-1. ה-TTL מגדיר כמה קפיצות (hops) החבילה יכולה לבצע לפני שתוקפה יפוג. כאשר TTL פג, הנתב שמטפל בחבילה מחזיר הודעת ICMP מסוג "Time-to-live Exceeded", שאותן אנחנו "תופסים" בתוכנית על מנת לדעת היכן החבילה נעצרה.

החבילות נשלחות שלוש פעמים לכל ערך TTL, ולאחר מכן ערך ה-TTL מוגדל ב-1. הדבר מאפשר לחבילה לעבור לתחנה הבאה במסלול ולהמשיך עד שהיא מגיעה ליעד או עד שמתקבלת תגובה אחרת. אם TTL מספיק גדול, והחבילה מגיעה ליעד, היעד מחזיר תגובת Echo Reply.

במקרה של כתובת היעד (192.0.2.1), נראה כי החבילה עוברת לdefault gateway בכתובת 10.100.102.1 ומשם אל הכתובת 212.143.208.34 של חברת cellcom ומשם ממשיכה להישלח אך אינה מקבלת תגובת Echo Reply. הכתובת שבחרנו הינה כתובת פנימית שלא נמצאת ברשת הפנימית שלנו ולכן לא נוכל למצוא אותה ונקבל שגיאת destination unreachable.

תהליך זה מאפשר לזהות את התחנות (נתבים) שבדרך, ומדגיש היכן החבילות עוברות במסלול שלהן. באמצעות המידע שמוחזר על ידי הנתבים (במקרה של TTL שפג) ניתן למפות את הדרך מהמקור ליעד. כל שינוי ב-TTL מייצג תחנה חדשה במסלול, והשליחה החוזרת מאמתת את המידע המתקבל.

3.בדיקת מסלול מורכב: ביצוע traceroute לכתובת 5.160.139.194 באיראן:

בדיקת מסלול עם מספר תחנות גדול ואימות תזמונים וכתובות, שדרגנו קצת את הסעיף , נשלח פינג לכתובת שנמצאת באיראן , כלומר נעבור מספר תחנות בדרך עד שנקבל חסימה עקב זה שלא ניתן לבצע תקשורת מישראל לאיראן.

צילום Terminal:

```

dor@dor-virtual-machine:~/Documents/Networks/part B$ sudo ./traceroute -a 5.160.139.194
traceroute to 5.160.139.194 with 30 hops max:
 1  10.100.102.1  3.243ms  3.406ms  2.173ms
 2  212.143.208.34  5.475ms  9.677ms  11.510ms
 3  212.143.203.221  7.757ms  7.167ms  6.654ms
 4  212.143.203.9  10.886ms  12.568ms  11.359ms
 5  212.143.12.66  45.502ms  56.649ms  46.111ms
 6  10.10.120.1  51.290ms  49.638ms  49.929ms
 7  80.231.200.53  66.708ms  58.820ms  54.962ms
 8  195.219.187.2  65.683ms  63.818ms  62.659ms
 9  195.219.187.7  64.935ms  66.454ms  65.365ms
10  195.219.87.56  58.198ms  57.034ms  58.702ms
11  195.219.87.199  130.838ms  132.300ms  131.122ms
12  85.132.90.170  139.486ms  138.170ms  141.286ms
13  * * *
14  * * *
```

...

...

```

28  * * *
29  * * *
30  * * *
Destination Unreachable.
```

צילום wireshark:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|-------------|-----------------|----------------|----------|--------|---|
| 19 | 1.522640786 | 10.100.102.1 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=0/0, ttl=1 (no response found) |
| 20 | 1.524070352 | 10.100.102.1 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 21 | 1.525484402 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=1/256, ttl=1 (no response found) |
| 22 | 1.526990192 | 10.100.102.1 | 10.100.102.121 | ICMP | 134 | Time-to-live exceeded (Time to live exceeded in transit) |
| 23 | 1.528998198 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=2/512, ttl=1 (no response found) |
| 24 | 1.530976922 | 10.100.102.1 | 10.100.102.121 | ICMP | 114 | Time-to-live exceeded (Time to live exceeded in transit) |
| 25 | 1.531313562 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=3/768, ttl=2 (no response found) |
| 26 | 1.535901070 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 27 | 1.537231593 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=4/1024, ttl=2 (no response found) |
| 28 | 1.545389945 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 29 | 1.546599978 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=5/1280, ttl=2 (no response found) |
| 30 | 1.550971150 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 31 | 1.553169331 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=6/1536, ttl=3 (no response found) |
| 32 | 1.555390991 | 212.143.203.221 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 33 | 1.555883069 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=7/1792, ttl=3 (no response found) |
| 34 | 1.572304770 | 212.143.203.221 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 35 | 1.573953747 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=8/2048, ttl=3 (no response found) |
| 36 | 1.578548588 | 212.143.203.221 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 37 | 1.579771211 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=9/2304, ttl=4 (no response found) |
| 38 | 1.585010090 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 39 | 1.596670518 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=10/2560, ttl=4 (no response found) |
| 40 | 1.602784206 | 212.143.203.9 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 41 | 1.603252349 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=11/2816, ttl=4 (no response found) |
| 42 | 1.613350097 | 212.143.208.34 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 43 | 1.614712334 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=12/3072, ttl=5 (no response found) |
| 44 | 1.659989765 | 212.143.12.66 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 45 | 1.660215593 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=13/3328, ttl=5 (no response found) |
| 46 | 1.675591970 | 212.143.12.66 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 47 | 1.716854387 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=14/3584, ttl=5 (no response found) |
| 48 | 1.762757588 | 212.143.12.66 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 49 | 1.763910125 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=15/3840, ttl=6 (no response found) |
| 50 | 1.814952029 | 10.10.120.1 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 51 | 1.815349427 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=16/4096, ttl=6 (no response found) |
| 52 | 1.864047010 | 10.10.120.1 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 53 | 1.864860549 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=17/4352, ttl=6 (no response found) |
| 54 | 1.914594750 | 10.10.120.1 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 55 | 1.914975633 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=18/4608, ttl=7 (no response found) |
| 56 | 1.931000510 | 80.231.200.53 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 57 | 1.931704167 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=19/4864, ttl=7 (no response found) |
| 58 | 2.040463372 | 80.231.200.53 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|-------------|----------------|----------------|----------|--------|---|
| 62 | 2.040305772 | 80.231.200.53 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 63 | 2.040395954 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=20/5120, ttl=7 (no response found) |
| 64 | 2.095321570 | 80.231.200.53 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 65 | 2.095559781 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=21/5376, ttl=8 (no response found) |
| 66 | 2.101074687 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 67 | 2.101262020 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=22/5632, ttl=8 (no response found) |
| 68 | 2.122109135 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 69 | 2.225139798 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=23/5888, ttl=8 (no response found) |
| 70 | 2.287584068 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 71 | 2.288934512 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=24/6144, ttl=9 (no response found) |
| 72 | 2.352773421 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 73 | 2.352981154 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=25/6400, ttl=9 (no response found) |
| 74 | 2.418495932 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 75 | 2.419440302 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=26/6656, ttl=9 (no response found) |
| 76 | 2.484460296 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 77 | 2.484929718 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=27/6912, ttl=10 (no response found) |
| 78 | 2.545260991 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 81 | 2.545132068 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=28/7168, ttl=10 (no response found) |
| 82 | 2.599910766 | 195.210.187.2 | 10.100.102.121 | ICMP | 70 | Time-to-live exceeded (Time to live exceeded in transit) |
| 83 | 2.600181803 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=29/7424, ttl=10 (no response found) |
| 84 | 2.650081917 | 195.210.187.2 | 10.100.102.121 | ICMP | 102 | Time-to-live exceeded (Time to live exceeded in transit) |
| 85 | 2.659680529 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=30/7680, ttl=11 (no response found) |
| 86 | 2.789669323 | 195.210.187.2 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 87 | 2.789916667 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=31/7936, ttl=11 (no response found) |
| 88 | 2.824500551 | 195.210.187.2 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 109 | 2.922426519 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=32/8192, ttl=11 (no response found) |
| 111 | 3.058999042 | 195.210.187.2 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 112 | 3.053537802 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=33/8448, ttl=12 (no response found) |
| 114 | 3.192775105 | 85.132.90.178 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 115 | 3.193043233 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=34/8704, ttl=12 (no response found) |
| 116 | 3.331010810 | 85.132.90.178 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 117 | 3.331335233 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=35/8960, ttl=12 (no response found) |
| 118 | 3.472352716 | 85.132.90.178 | 10.100.102.121 | ICMP | 110 | Time-to-live exceeded (Time to live exceeded in transit) |
| 119 | 3.472620178 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=36/9216, ttl=13 (no response found) |
| 127 | 3.974137063 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=37/9472, ttl=13 (no response found) |
| 129 | 4.475470469 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=38/9728, ttl=13 (no response found) |
| 130 | 4.976590409 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=39/9984, ttl=14 (no response found) |
| 131 | 5.477559089 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=40/10240, ttl=14 (no response found) |
| 134 | 5.978796086 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=41/10496, ttl=14 (no response found) |
| 139 | 6.479712920 | 10.100.102.121 | 5.160.139.194 | ICMP | 106 | Echo (ping) request id=0x4918, seq=42/10752, ttl=15 (no response found) |

(יש עוד פקטות אך כולן פקטות של no response found לאחר שלא מגיעה תשובה לאחר שנחסמה הגישה בדרך. השאר נמצאות בקובץ ההקלטה.)

במהלך ביצוע ה-Traceroute לכתובת באיראן מהמחשב שממוקם בישראל, נשלחות חבילות ICMP מסוג Echo Request אל היעד. תהליך ה-Traceroute מתחיל בערך TTL=1 ומעלה את הערך בהדרגה כדי לעבור בין תחנות (נתבים) במסלול.

בתחילת התהליך ניתן לראות שהחבילות הגיעו בהצלחה למספר תחנות ביניים, שכל אחת מהן השיבה בהודעת ICMP מסוג "Time-to-live Exceeded". תחנות אלו כוללות כתובות IP מישראל (12.143.208.34) ותחנות בינלאומיות נוספות שממשיכות את המסלול לכיוון היעד.

לאחר מספר קפיצות, כאשר החבילות מתקדמות לרשתות רחוקות יותר, מתברר שלא מתקבלות תגובות מהיעד באיראן(החל מחבילה 119) הדבר נובע ממדיניות חסימה ברמת הרשת האיראנית. לכן, למרות שהחבילות מגיעות לנתבים שונים במסלול, לא מתאפשר להן לעבור עד ליעד הסופי.

בפועל, תהליך ה-Traceroute מדגיש כיצד החבילות עוברות בהצלחה דרך תחנות מסוימות אך נעצרות לפני שהן מגיעות לרשת היעד. עובדה זו ניכרת מהופעת הודעות No response found בחלקים מתקדמים של המסלול. התוצאה משקפת את מגבלות הגישה בין ישראל לאיראן, כפי שהן משתקפות במאפייני הרשת.

חלק ג': Network Scanner

מטרות הכלי: סריקת רשת לצורך זיהוי כתובות IP פעילות בטווח נתון.

מבנה הפקודה: הכלי מופעל באמצעות הפקודה הבאה:

```
$ sudo ./discovery -a <IP ADDRESS> -c <SUBNET MASK>
```

פרמטרים נתמכים:

- **-a:** כתובת הרשת שעליה תתבצע הסריקה (חובה).
- **-c:** subnet-mask בפורמט מספרי לצורך חישוב הטווח לסריקה (חובה).

מימוש: הכלי מחשב בעזרת אלגוריתם פנימי את כתובת ה-network בהתבסס על הכתובת הנתונה ומסכת הרשת (subnet mask). לאחר מכן הוא עובר על כל הכתובות בטווח הרשת, שולח בקשות ICMP לכל כתובת, ומציג את הכתובות הפעילות בלבד. בנוסף, התהליך כולל מנגנון טיפול בשגיאות, כמו כתובות לא זמינות או חריגה מזמן המתנה. בסיום הסריקה, תוצג הודעה המאשרת את השלמת הפעולה.

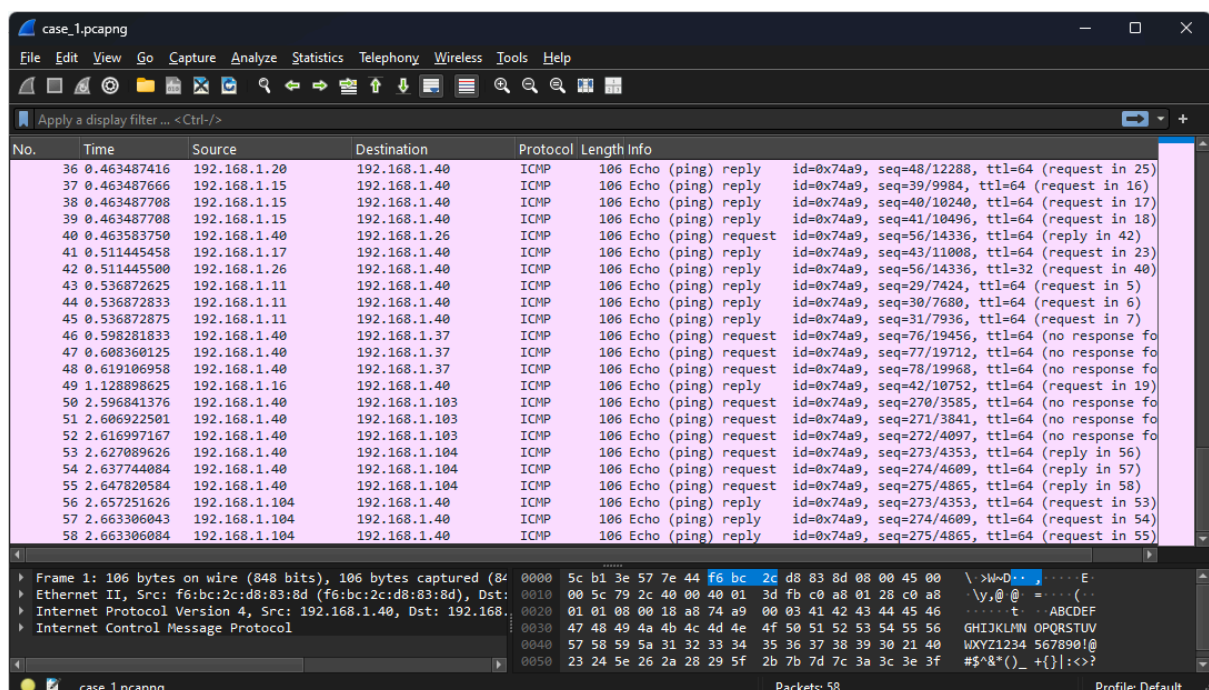
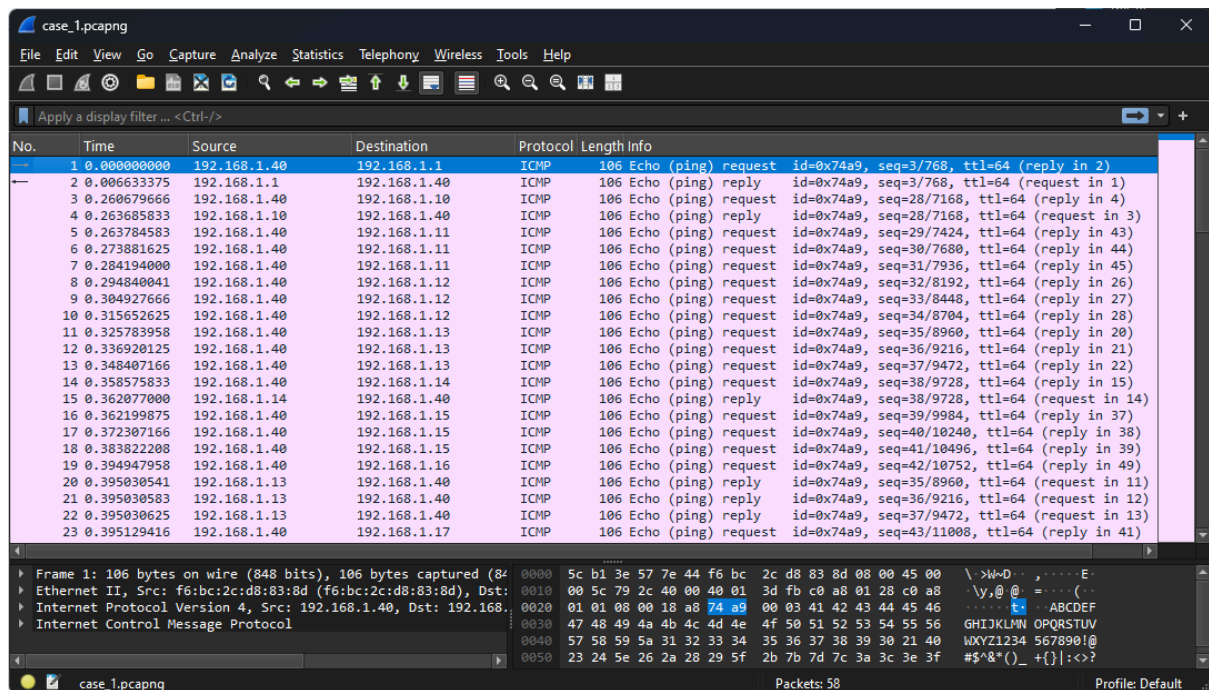
פלט לדוגמה:

```
● dor@dor-virtual-machine:~/Documents/Networks/part C$ sudo ./discovery -a 10.100.102.64 -c 24
[sudo] password for dor:
scanning 10.100.102.0 /24 for 255 addresses
10.100.102.1
10.100.102.6
10.100.102.5
10.100.102.7
10.100.102.36
10.100.102.38
10.100.102.60
10.100.102.97
10.100.102.121
Scan Complete!
```

בדיקות ב-Wireshark: לצורך וידוא נכונות המימוש, נעשה שימוש ב-Wireshark לתיעוד בקשות ICMP שנשלחו לכל הכתובות בטווח. התגובות שזוהו אישרו את פעילות הכתובות ברשת. צילומי מסך מצורפים להצגת מבנה החבילות והזמנים שנמדדו.

1. סריקת טווח קטן

- תיאור: סריקה של רשת קטנה מאוד (למשל, טווח 24/2, שמכיל רק 255 כתובות).
נבדוק ברשת הלוקלית של הבית המקומי (192.168.1.0) נזין כתובת 192.168.1.231 כדי להראות שהאלגוריתם שבנינו מזהה את subnet של 24, יציג אותו למסך ויתחיל לסרוק את תת הרשת שדרשנו.
- מטרה: לבדוק אם הכלי מתמודד עם טווח קטן ויודע לזהות נכון את הכתובות הפעילות.

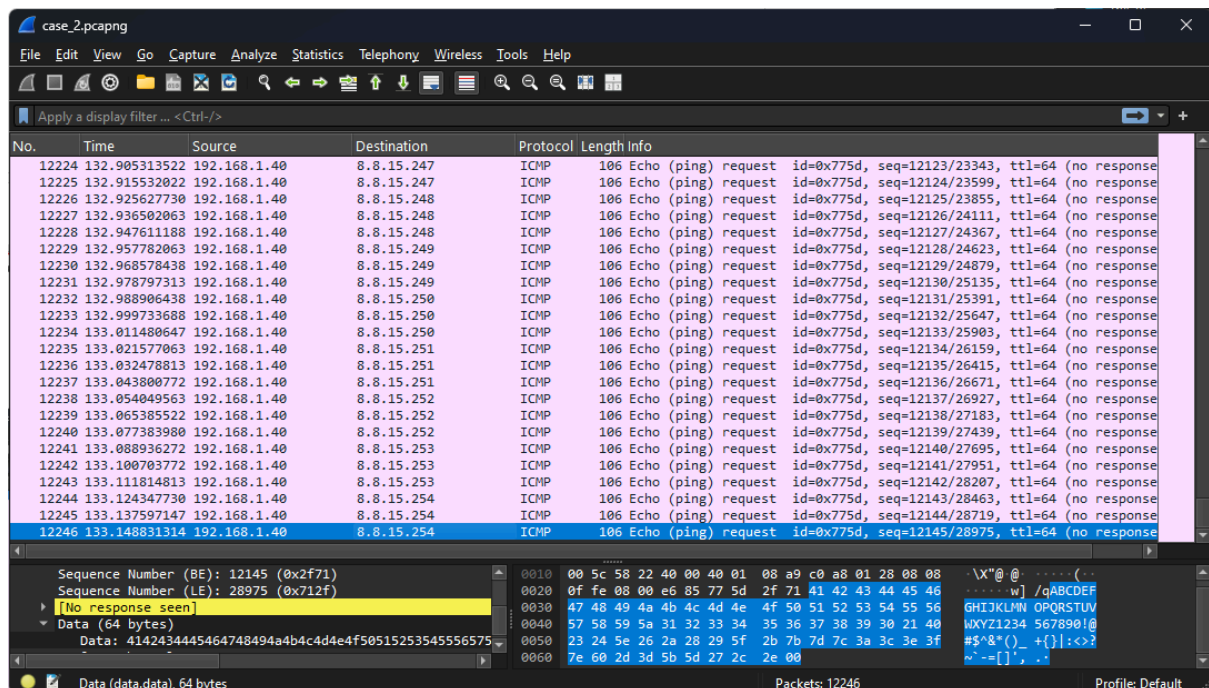
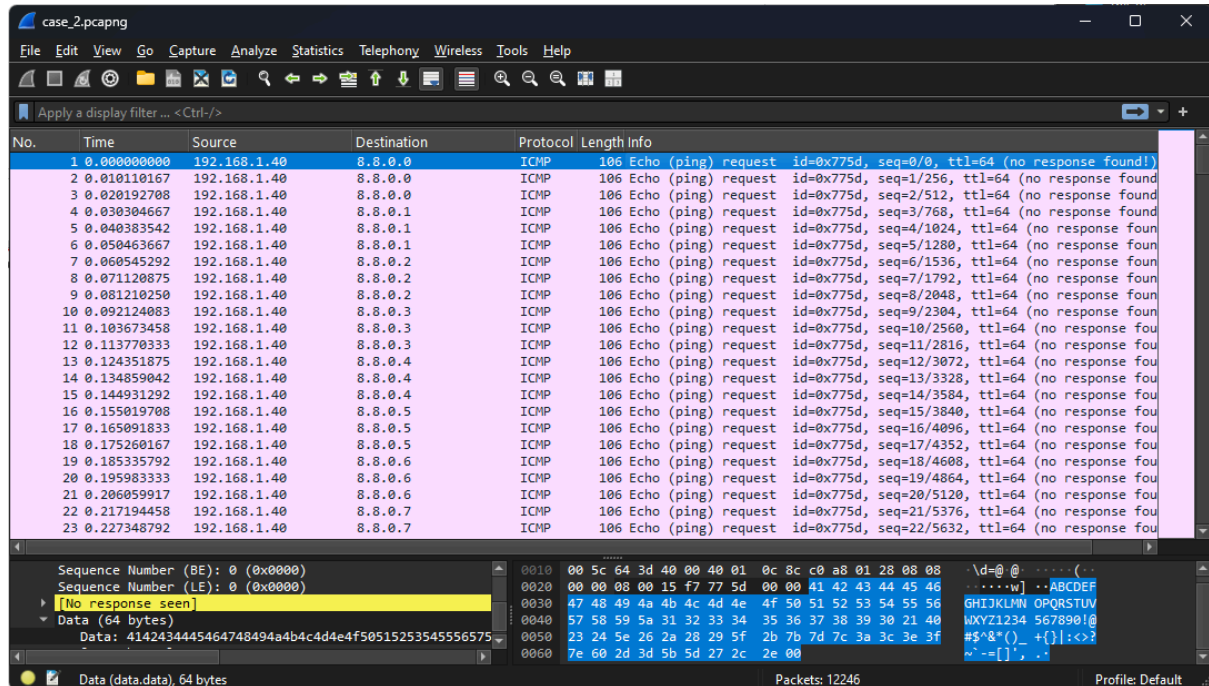


Terminal Output:

```
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part C$ sudo ./discovery -a 192.168.1.231 -c 24
scanning 192.168.1.0 /24 for 255 addresses
192.168.1.1
192.168.1.12
192.168.1.13
192.168.1.14
192.168.1.15
192.168.1.16
192.168.1.17
192.168.1.18
192.168.1.19
192.168.1.21
192.168.1.23
192.168.1.24
192.168.1.25
192.168.1.26
192.168.1.28
192.168.1.34
192.168.1.35
192.168.1.41
192.168.1.47
192.168.1.104
192.168.1.105
Scan Complete!
```


2. סריקה של טווח רשת גדול

- תיאור: סריקה של טווח 20/ (כ-4095 כתובות IP).
- טווח הכתובות הוא 8.8.0.0/20, המכסה כתובות מ-8.8.0.0 עד 8.8.15.255.
- מטרה: לבדוק את עמידות הכלי וביצועיו בטווחים גדולים מאוד.



Terminal Output:

```
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part C$ sudo ./discovery -a 8.8.8.8 -c 20  
scanning 8.8.0.0 /20 for 4095 addresses
```

```
8.8.1.10  
8.8.1.11  
8.8.1.12  
8.8.1.13  
8.8.1.14  
8.8.1.15  
8.8.1.16  
8.8.1.17  
8.8.1.18  
8.8.1.19  
8.8.1.20  
8.8.1.21  
8.8.1.22  
8.8.1.23  
8.8.1.24  
8.8.1.27  
8.8.1.28  
8.8.1.29  
8.8.1.30  
8.8.1.31  
8.8.1.32  
8.8.1.70  
8.8.1.71  
8.8.1.72  
8.8.1.73  
8.8.1.74  
8.8.1.75  
8.8.1.76  
8.8.1.77  
8.8.1.78  
8.8.1.79  
8.8.1.80  
8.8.1.81  
8.8.1.82  
8.8.1.83  
8.8.1.84  
8.8.1.102  
8.8.1.103  
8.8.1.104  
8.8.1.105  
8.8.1.106  
8.8.1.107  
8.8.1.108  
8.8.1.109  
8.8.1.110  
8.8.1.111  
8.8.1.112  
8.8.1.113  
8.8.1.114  
8.8.1.115  
8.8.1.116  
8.8.4.4  
8.8.4.5  
8.8.5.6  
8.8.5.7  
8.8.5.8  
8.8.6.22  
8.8.6.23  
8.8.6.24  
8.8.6.33  
8.8.6.34  
8.8.6.35
```

```
8.8.6.74  
8.8.6.75  
8.8.6.76  
8.8.8.8  
8.8.8.9  
8.8.8.10  
8.8.11.6  
8.8.11.7  
8.8.11.8  
8.8.11.15  
8.8.11.17  
8.8.11.18  
8.8.11.19  
8.8.11.20  
8.8.11.21  
8.8.11.127  
8.8.11.128  
8.8.11.129  
8.8.12.1  
8.8.12.2  
8.8.12.3  
8.8.12.4  
8.8.12.5  
8.8.12.6  
Scan Complete!
```

3. כתובת רשת לא תקינה

- תיאור: הזנה של כתובת רשת שאינה חוקית (למשל, 192.168.1.300).
- מטרה: לבדוק אם הכלי מתמודד עם קלט לא תקין ומציג הודעת שגיאה מתאימה.

```
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part C$ sudo ./discovery  
-a 192.168.1.300 -c 20  
[sudo] password for ubuntu:  
Invalid IP address
```

4. סריקה בתת-רשת (Subnet) מורכבת

- תיאור: סריקה של כתובת רשת עם מסכה לא רגילה, כלומר נסרוק את הרשת 208.67.222.216 עם תת רשת 29, תת רשת של 29 תתן לנו תוצאה של 7 רשתות סה"כ.
- מטרה: לבדוק את היכולת של הכלי לחשב נכונה את טווח הכתובות ולבצע סריקה מדויקת.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|-------------|----------------|----------------|----------|--------|---|
| 1 | 0.000000000 | 192.168.1.40 | 208.67.222.216 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=0/0, ttl=64 (no response found!) |
| 2 | 0.013174708 | 192.168.1.40 | 208.67.222.216 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=1/256, ttl=64 (no response found!) |
| 3 | 0.024114875 | 192.168.1.40 | 208.67.222.216 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=2/512, ttl=64 (no response found!) |
| 4 | 0.034897875 | 192.168.1.40 | 208.67.222.217 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=3/768, ttl=64 (no response found!) |
| 5 | 0.045014625 | 192.168.1.40 | 208.67.222.217 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=4/1024, ttl=64 (no response found!) |
| 6 | 0.055165334 | 192.168.1.40 | 208.67.222.217 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=5/1280, ttl=64 (no response found!) |
| 7 | 0.065271709 | 192.168.1.40 | 208.67.222.218 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=6/1536, ttl=64 (no response found!) |
| 8 | 0.075443750 | 192.168.1.40 | 208.67.222.218 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=7/1792, ttl=64 (no response found!) |
| 9 | 0.087091042 | 192.168.1.40 | 208.67.222.218 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=8/2048, ttl=64 (no response found!) |
| 10 | 0.097774084 | 192.168.1.40 | 208.67.222.219 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=9/2304, ttl=64 (no response found!) |
| 11 | 0.108543584 | 192.168.1.40 | 208.67.222.219 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=10/2560, ttl=64 (no response found!) |
| 12 | 0.119411792 | 192.168.1.40 | 208.67.222.219 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=11/2816, ttl=64 (no response found!) |
| 13 | 0.130087250 | 192.168.1.40 | 208.67.222.220 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=12/3072, ttl=64 (reply in 16) |
| 14 | 0.140169042 | 192.168.1.40 | 208.67.222.220 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=13/3328, ttl=64 (reply in 18) |
| 15 | 0.151067709 | 192.168.1.40 | 208.67.222.220 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=14/3584, ttl=64 (reply in 21) |
| 16 | 0.152193584 | 208.67.222.220 | 192.168.1.40 | ICMP | 106 | Echo (ping) reply id=0x7ccd, seq=12/3072, ttl=53 (request in 13) |
| 17 | 0.152301917 | 192.168.1.40 | 208.67.222.221 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=15/3840, ttl=64 (no response found!) |
| 18 | 0.159770917 | 208.67.222.220 | 192.168.1.40 | ICMP | 106 | Echo (ping) reply id=0x7ccd, seq=13/3328, ttl=53 (request in 14) |
| 19 | 0.159873917 | 192.168.1.40 | 208.67.222.222 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=16/4096, ttl=64 (reply in 22) |
| 20 | 0.169997125 | 192.168.1.40 | 208.67.222.222 | ICMP | 106 | Echo (ping) request id=0x7ccd, seq=17/4352, ttl=64 (reply in 23) |
| 21 | 0.172564834 | 208.67.222.220 | 192.168.1.40 | ICMP | 106 | Echo (ping) reply id=0x7ccd, seq=14/3584, ttl=53 (request in 15) |
| 22 | 0.181709459 | 208.67.222.222 | 192.168.1.40 | ICMP | 106 | Echo (ping) reply id=0x7ccd, seq=16/4096, ttl=53 (request in 19) |
| 23 | 0.193490334 | 208.67.222.222 | 192.168.1.40 | ICMP | 106 | Echo (ping) reply id=0x7ccd, seq=17/4352, ttl=53 (request in 20) |

Terminal Output:

```
ubuntu@ubuntu:~/Downloads/211896279_208526012 (1)/part C$ sudo ./discovery -a 208.67.222.222 -c 29
scanning 208.67.222.216 /29 for 7 addresses
208.67.222.220
208.67.222.221
208.67.222.222
Scan Complete!
```

שאלות שנשאלו עם ה AI לטובת הבנת המטלה:

Question 1:

can you explain to me what is inet_pton and inet_hntop

Answer 1:

The functions `inet_pton` and `inet_ntop` are part of the POSIX library and are commonly used for converting IP addresses between textual and binary forms in network programming. Here's a breakdown:

1. `inet_pton` (Presentation to Numeric)

Purpose: Converts an IP address from its standard text representation (e.g., "192.168.0.1" or "2001:db8::ff00:42:8329") into its binary form for use in network functions.

Prototype:

C

CopyEdit

```
int inet_pton(int af, const char *src, void *dst);
```

- - **Parameters:**
 - `af`: Address family (`AF_INET` for IPv4, `AF_INET6` for IPv6).
 - `src`: Pointer to a null-terminated string containing the IP address in text format.
 - `dst`: Pointer to a buffer where the binary form of the IP address will be stored.
 - **Returns:**
 - `1`: Success (address converted and stored in `dst`).
 - `0`: Input address is not valid for the specified address family.
 - `-1`: Error (e.g., invalid `af`).
-

2. `inet_ntop` (Numeric to Presentation)

Purpose: Converts an IP address from its binary form to a readable text representation.

Prototype:

C

CopyEdit

```
const char *inet_ntop(int af, const void *src, char *dst, socklen_t size);
```

-
- **Parameters:**
 - `af`: Address family (`AF_INET` for IPv4, `AF_INET6` for IPv6).
 - `src`: Pointer to the binary representation of the IP address.
 - `dst`: Buffer where the textual representation will be stored.
 - `size`: Size of the `dst` buffer (should be large enough to hold the resulting string).
- **Returns:**
 - Pointer to `dst` on success.
 - `NULL`: Error (e.g., invalid `af`, insufficient buffer size).

Question 2:

can you make for me in C checksum for IPV6

Answer 2:

Here is an implementation of a checksum function for IPv6 in C, based on the requirements of your project:

```
unsigned short int calculate_checksum_icmp6(void *data, unsigned int bytes,
                                           struct in6_addr *src, struct
in6_addr *dest, uint8_t next_header) {
    struct pseudo_header {
        struct in6_addr src_addr;
        struct in6_addr dest_addr;
        uint32_t length;
        uint8_t zeros[3];
        uint8_t next_header;
    } psh;

    memset(&psh, 0, sizeof(psh));
    psh.src_addr = *src;
    psh.dest_addr = *dest;
    psh.length = htonl(bytes);
    psh.next_header = next_header;

    unsigned int total_sum = 0;
    unsigned short int *data_pointer = (unsigned short int *)&psh;

    // Add pseudo-header to checksum
    for (unsigned int i = 0; i < sizeof(psh) / 2; i++) {
        total_sum += *data_pointer++;
    }

    // Add ICMPv6 header and payload to checksum
    data_pointer = (unsigned short int *)data;
    while (bytes > 1) {
        total_sum += *data_pointer++;
        bytes -= 2;
    }

    // Add left-over byte, if any
    if (bytes > 0) {
        total_sum += *((unsigned char *)data_pointer);
    }

    // Fold 32-bit sum to 16 bits
    while (total_sum >> 16) {
        total_sum = (total_sum & 0xFFFF) + (total_sum >> 16);
    }

    // Return the one's complement of the result
    return (~(unsigned short int)total_sum));
}
```

Question 3:

how to get the ttl value out of the ip socket

Answer 3:

To retrieve the **TTL (Time to Live)** value from an IP socket, you can use the `getsockopt()` function. This allows you to query the current TTL value of the socket.

Explanation

1. `getsockopt()` Parameters:

- **sock**: The socket file descriptor.
- **IPPROTO_IP**: The protocol level for IP options.
- **IP_TTL**: The option you want to retrieve (in this case, the TTL value).
- **&ttl**: A pointer to the variable where the retrieved TTL value will be stored.
- **&optlen**: A pointer to the length of the buffer `ttl`. This must initially hold the size of the buffer and will be updated to reflect the size of the returned value.

2. Return Value:

- `getsockopt()` returns `0` on success.
- On failure, it returns `-1`, and `errno` is set to indicate the error.

Question 4:

'int ret = poll(fds, 1, TIMEOUT);' explain to me this line of code

Answer 4:

The line `int ret = poll(fds, 1, TIMEOUT);` is using the `poll()` function, a system call in Linux and UNIX-like operating systems, to monitor file descriptors for events. Here's a detailed breakdown:

Understanding `poll()`

`poll()` is used to wait for events on a set of file descriptors. It is often used in network programming to monitor multiple sockets or file descriptors for readiness to perform I/O.

Arguments in Detail

1. `fds`:
 - This is a pointer to an array of `struct pollfd`.
 - Each `pollfd` struct specifies a file descriptor to monitor and the events of interest.
2. `1`:
 - This is the number of file descriptors in the `fds` array to monitor.
 - In this case, it's `1`, meaning only one file descriptor (`fds[0]`) is being monitored.
3. `TIMEOUT`:
 - Specifies the maximum amount of time (in milliseconds) to wait for an event.
 - If `TIMEOUT` is `0`, `poll()` returns immediately.
 - If `TIMEOUT` is `-1`, `poll()` waits indefinitely.

Return Value

The result of `poll()` is stored in `ret`:

- `ret > 0`: Indicates the number of file descriptors with events.
- `ret == 0`: Timeout occurred (no events within the specified time).
- `ret < 0`: An error occurred. The global `errno` is set to indicate the error.

Question 5:

what should i do with the -f argument according to the pdf

Answer 5:

According to the instructions in the provided PDF, the `-f` flag is meant to enable "flood mode" in your custom ping implementation. When this flag is specified:

- The program will send ICMP packets back-to-back without any delay between them, effectively "flooding" the network with packets.

This flag is optional when running the program but must be implemented in your code. Make sure the program also supports other required features like statistics display and handling a timeout for replies ex4_2025_new.

Question 6:

תוכל לתת לי כתובן באיראן?

Answer 6:

IP דרך מספר תחנות רבות, ניתן להשתמש בכתובת (Traceroute) כדי לבצע בדיקת מסלול מורכב משויכת לספק שירותי האינטרנט **5.160.139.194** IP-ציבורית באיראן. לדוגמה, כתובת ה **Respina Networks & Beyond PJSC** באיראן.