**מטלה 4 רשתות תקשורת**

**מגישים:**

שלומה טימסיט- 332376813

איתי שגב- 315112482

**סביבת עבודה ופרטים טכניים:**

אנו עובדים ב vmware workstation pro ולא היה לנו גישה ישירה ל ipv6 ולכן אנו מוסיפים אותו ידנית כל פעם על מנת שבאמת יהיה לנו כתובת שיוצאת מהמחשב שלנו.

על מנת להריץ את התכנית נפעיל make ולאחר מכן נפעל לפי ההוראות ברידמי.

**הסברים כלליים ומושגים חשובים:**

**חלק א:**

תכנית זו היא יישום של כלי ping בשפת C התומך הן ב-IPv4 וגם בIPv6. התוכנית מאפשרת לשלוח בקשות ping לכתובת מרוחקת ולקבל מידע על זמני תגובה וסטטיסטיקות נוספות.

אנו קבענו ערך דיפולטיבי 4 פקטות אלא אם כן יש את דגל C בכדי להציג לכם את התוצאות בצורה מסודרת.

**תכונות עיקריות**

* תמיכה ב-IPv4 ו-IPv6
* אפשרות להגדרת מספר בקשות ping
* מצב flood להגברת קצב שליחת הבקשות
* חישוב וניתוח זמני תגובה (RTT)
* הצגת סטטיסטיקות מפורטות.

**חלק ב:**

תכנית זו היא יישום של כלי traceroute בשפת C הכלי מאפשר לעקוב אחר המסלול שחבילות נתונים עוברות ברשת עד ליעדן, תוך הצגת הנתבים (routers) בדרך וזמני התגובה שלהם.

**תכונות עיקריות**

* מעקב אחר מסלול חבילות ברשת
* שימוש בפרוטוקול ICMP
* מדידת זמני תגובה (RTT)
* תמיכה בכתובות IPv4
* הגבלת TTL מקסימלי ל-30 קפיצות
* תמיכה בטיימאאוט של 2 שניות לכל בקשה

**חלק א – ping :**

בחלק זה התבקשנו לממש את ping בעזרת raw socket התבקשנו כמו כן להפעיל דגלים שעליהם נפרט ובנוסף גם לתמוך בגרסאות ipv4,ipv6.

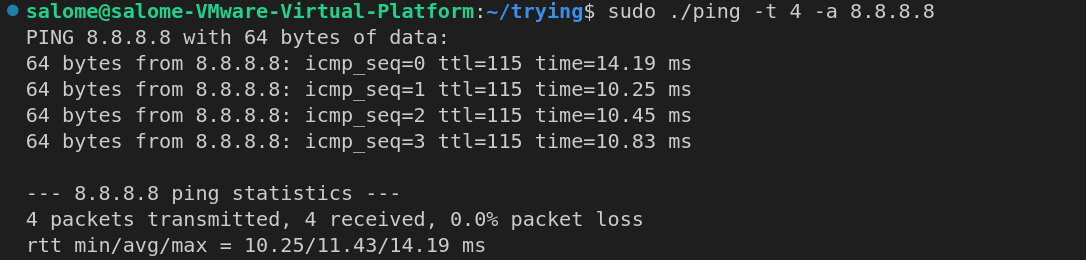
**נתחיל בכל דגל:**

**דגלי חובה:**

-a : אחרי דגל זה נכתוב את כתובת ה IP הרצוייה שעלייה נרצה לבצע ping

-t : אחרי דגל זה יבואו הספרה 4 – עבור כתובת ipv4 והספרה 6 עבור ipv6

נראה כעת איך זה נראה בתוכנית שלנו:

**עבור ipv4:**

**עבור ipv6:**

A screen shot of a computer program

Description automatically generated

**כעת נראה את 2 הדגלים הנוספים- שהם תוספת:**  
-c : דגל אשר מציין כמה חבילות אנו רוצים לשלוח ולקבל

-f : שולח את החבילות ללא דילאיי בינהם- כלומר כולן יחד

**עבור ipv4:**

**A screen shot of a computer

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**עבור ipv6:**

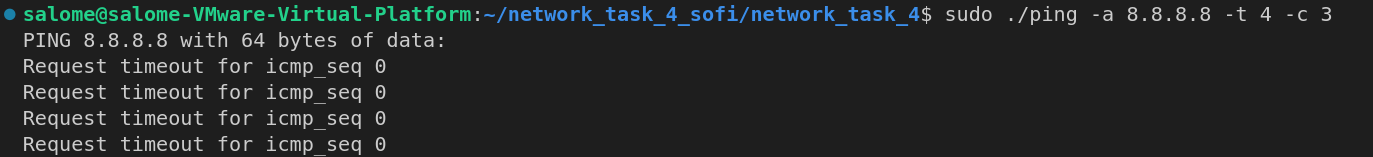
A computer screen with white text

Description automatically generated

A screen shot of a computer

Description automatically generated

נציג כאשר הקטנו את ה- TIMEOUT מאוד וקיבלנו:



כפי שהיינו מצפים.

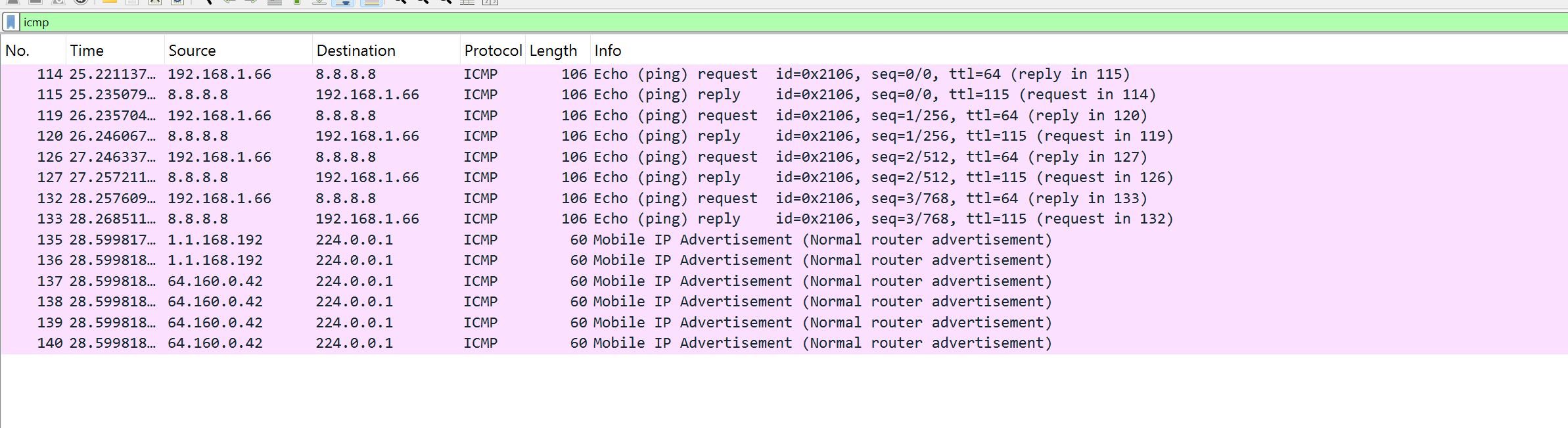
**הקלטות wireshark והסברן:**

כלל ההקלטות המלאות והסינון שלהן מצורפות לקובץ ההגשה וניתן לראות אותן- זה מחולק ל 2 תיקיות אחת שקשורה ל ipv4 ואחת שקשורה ל ipv6.

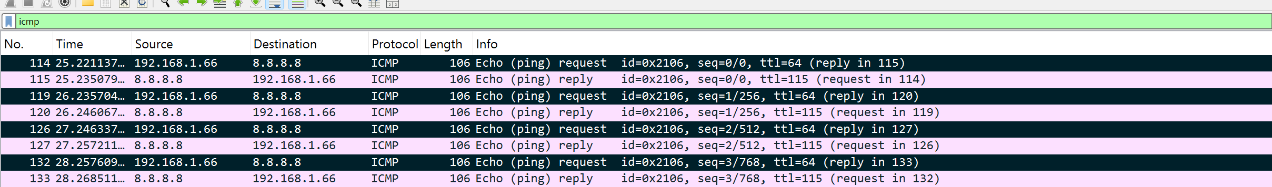
על מנת שיהיה נוח לקרוא חבילות הבקשה הן בשחור ובקשות התגובה בוורוד

**Ping ipv4:**

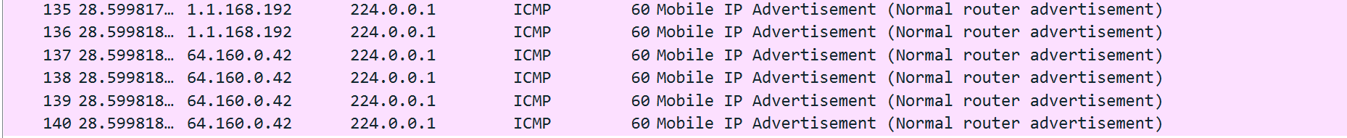
אנו נזהה שאלו הודעות שקשורות ל- ipv4 עם כתובת היעד 8.8.8.8 וכתובת המקור:192.168.1.166 שאילו כתובות ip4

כל חבילות ה ICMP:

**חבילות הבקשה והתגובה:**



ניתן לראות כי שלחנו 4 בקשות(echo ping request) וקיבלנו עליהם 4 חבילות תגובה(echo ping reply).

**הודעות נוספות שייצאו בסינון:**

ההודעות הנוספות למטה, הן הודעות שנשלחות לכתובת שידור, multitask והיא מודיעה למכשירים על הנתיבים הזמינים ברשת.

**הקלטת ping4 עם דגלי הרשות:**

הערה תכנית- סימנו את הבקשות בשחור ותשובות בוורוד

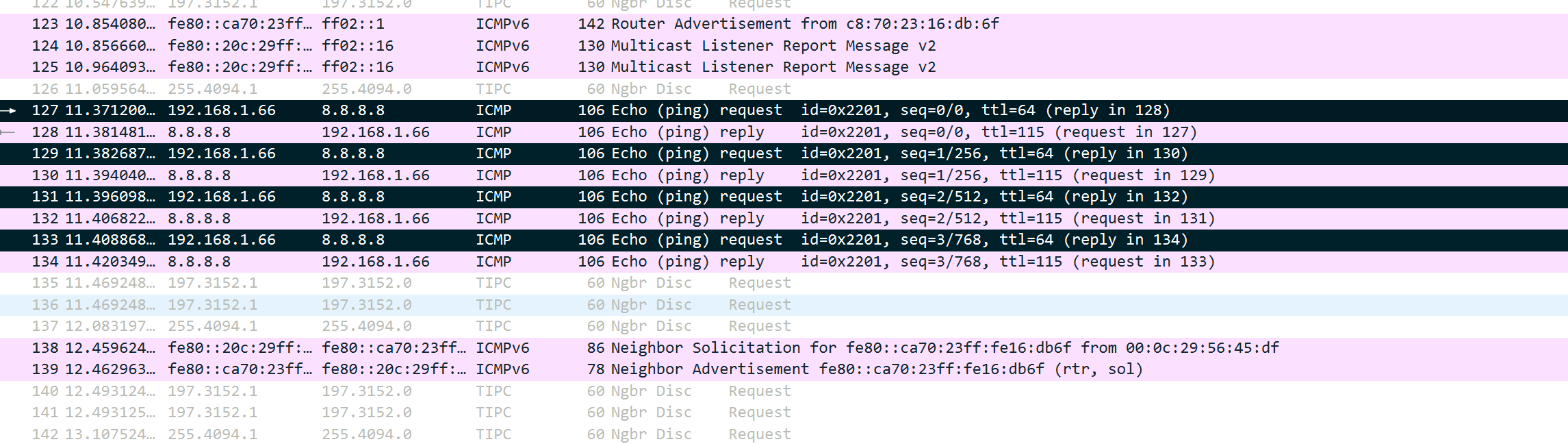
**דגל C:**

בחרנו להפעיל את דגלA screenshot of a computer

Description automatically generated זה עם 2 חבילות להלן התוצאות:

כאן ניתן לראות כי כי שלחנו 2 בקשות(echo ping request) וקיבלנו עליהם 2 חבילות תגובה(echo ping reply).

וההודעה הראשונה שאנו רואים זה כמו שציינו מקודם, הודעה שנועדה לעדכן איזה נתבים זמינים.

**דגל F :**

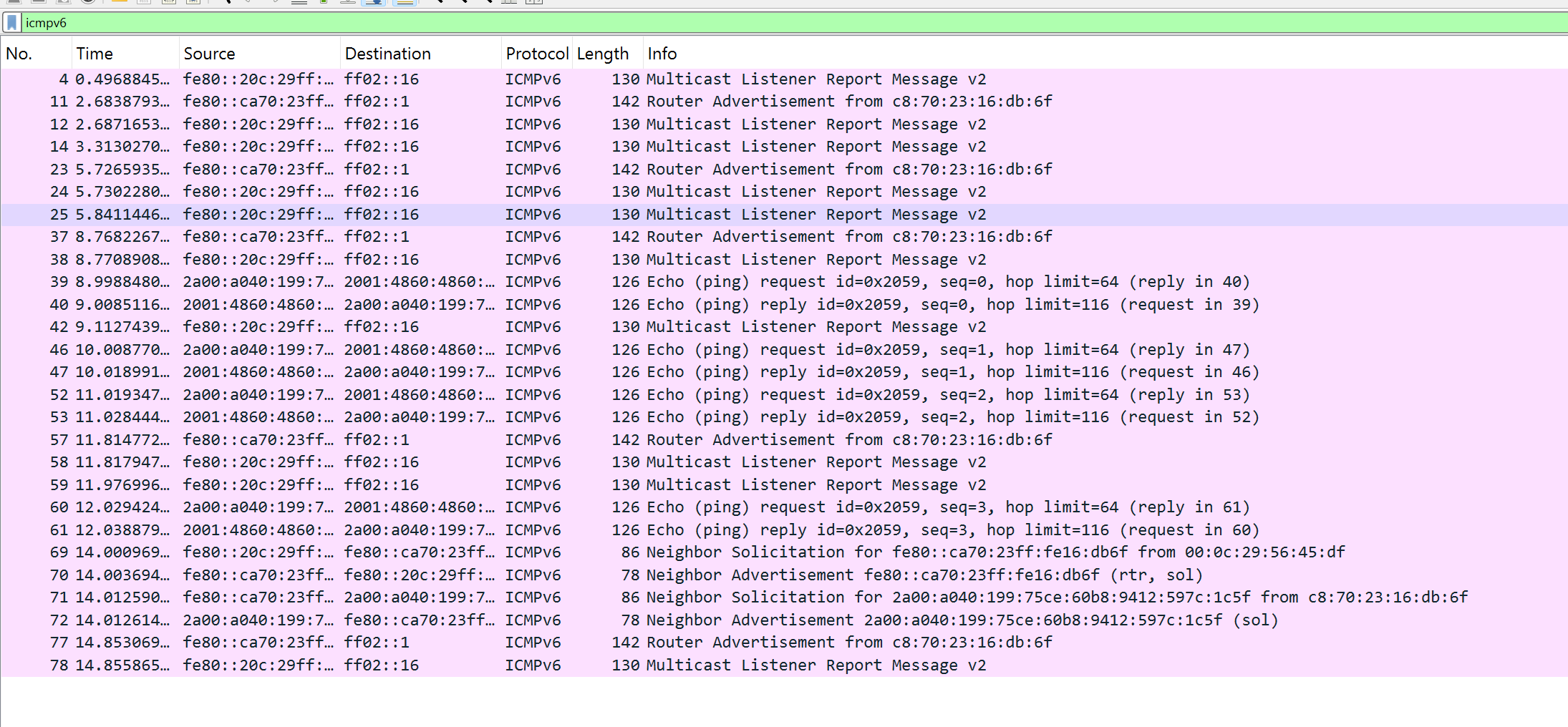
ניתן לראות כי כלל החבילות נשלחו והתקבל עליהן תשובה בסמיכות בניגוד לאחרות,

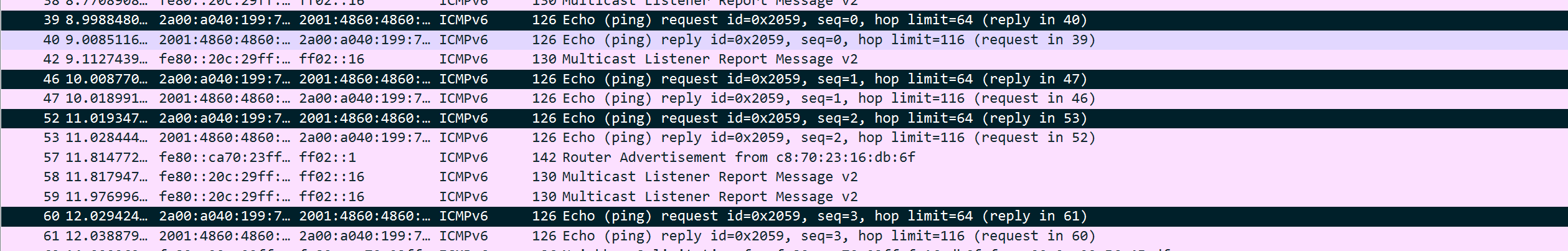
גם כאן ניתן לראות כי שלחנו 4 בקשות(echo ping request) וקיבלנו עליהם 4 חבילות תגובה(echo ping reply).

**Ping ipv6:**

אנו מזהים שאנו בכתובות של ipv6 מכתובות היעד המסופקות לכתובות המקור.

ההודעות הנוספות הן חלק מהicmpv6 על מנת להראות מי פנוי ומצב הרשת בקוד שלנו אנו סיננו אותם על מנת לקלוט אך ורק חבילות ICMP\_ECHOREPLY כל השאר לא רלוונטי לנו בהתעסקות שלנו בקוד.

הנה כל חבילות ה- icmpv6:

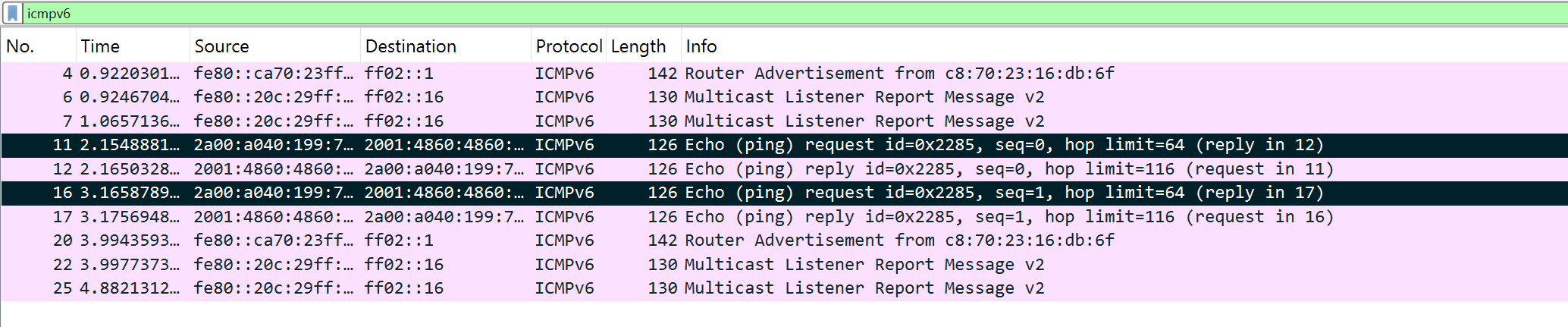
כעת נציג את חבילות הבקשה- תגובה בלבד:  
כאן ניתן לראות כי שלחנו 4 בקשות(echo ping request) וקיבלנו עליהם 4 חבילות תגובה(echo ping reply).

ההודעות הנוספות הן חלק מהicmpv6 על מנת להראות מי פנוי ומצב הרשת בקוד שלנו אנו סיננו אותם על מנת לקלוט אך ורק חבילות ICMP\_ECHOREPLY כל השאר לא רלוונטי לנו בהתעסקות שלנו בקוד.

**הקלטת ping4 עם דגלי הרשות:**

**דגל C:**

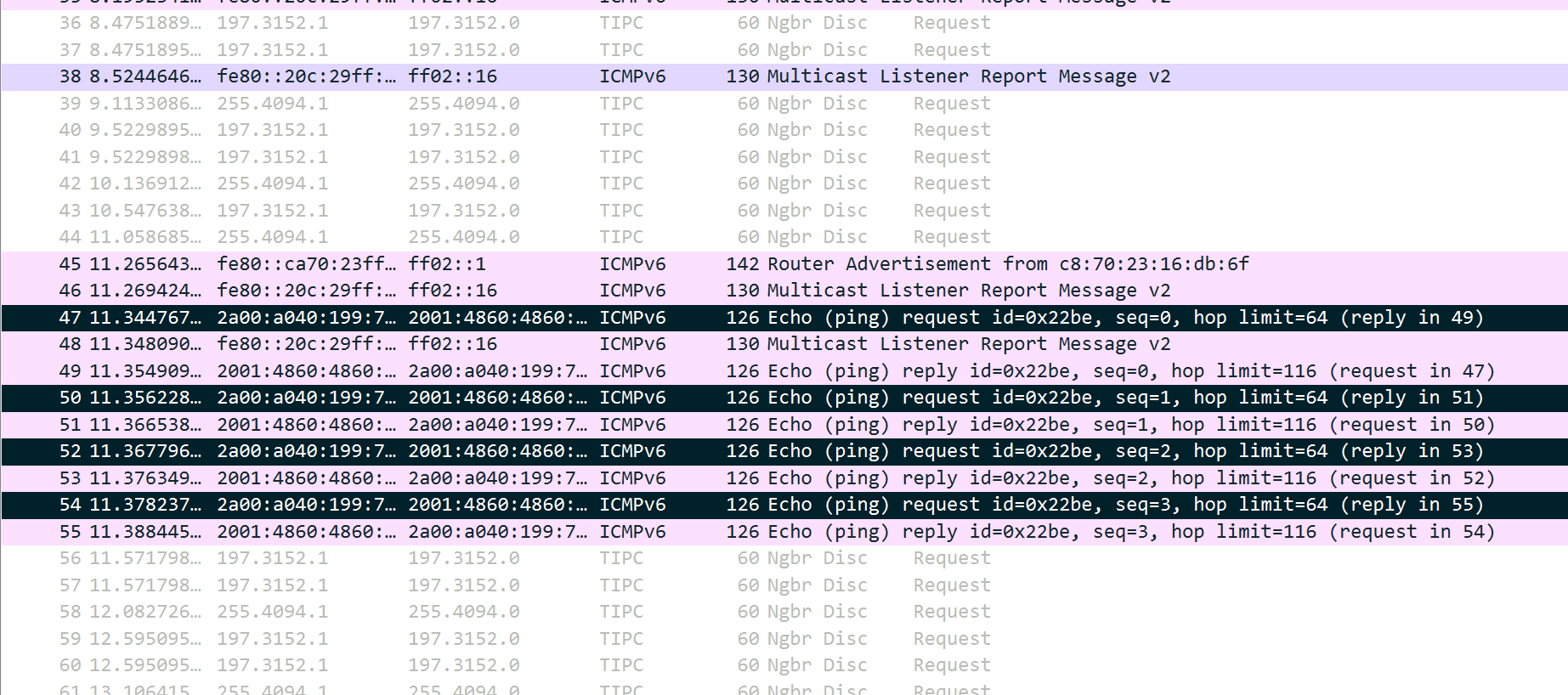
השתמשנו בדגל ונתנו לו ערך 2- כלומר אנו שולחים רק 2 חבילות,



כאן ניתן לראות כי כי שלחנו 2 בקשות(echo ping request) וקיבלנו עליהם 2 חבילות תגובה(echo ping reply).

A screenshot of a computer

Description automatically generatedעל מנת להראות את השוני מדגל F בצפיפות נראה שהחבילות אינן צמודות:

**דגל F:**

ניתן לראות כי ההודעות מרוכזות יחד מכיוון שהן נשלחות בצמוד זו לזו בניגוד לאחרות.

גם כאן ניתן לראות כי שלחנו 4 בקשות(echo ping request) וקיבלנו עליהם 4 חבילות תגובה(echo ping reply).

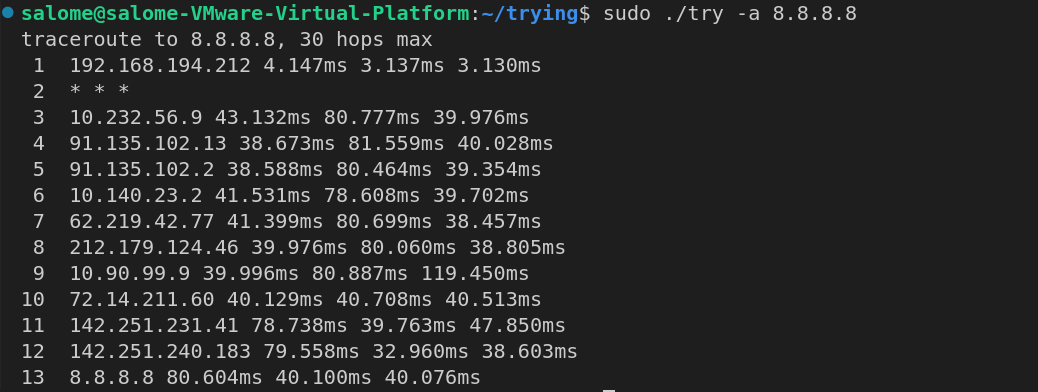
**חלק ב- traceroute:**

התבקשנו לממש תכנית אשר מיישמת את ה- traceroute בעזרת ה- raw socket בחלק זה השתמשנו בחלק מהקוד של ping והוספנו את הפונקציונליות של מה שהיינו צריכים.

השתמשנו כאן בדגל אחד:

a : אחרי דגל זה נכתוב את כתובת ה IP הרצוייה שעלייה נרצה לבצע ping

התחברנו דרך הטלפון על מנת שייראה לנו הרבה קפיצות, הנה הoutput שלנו:

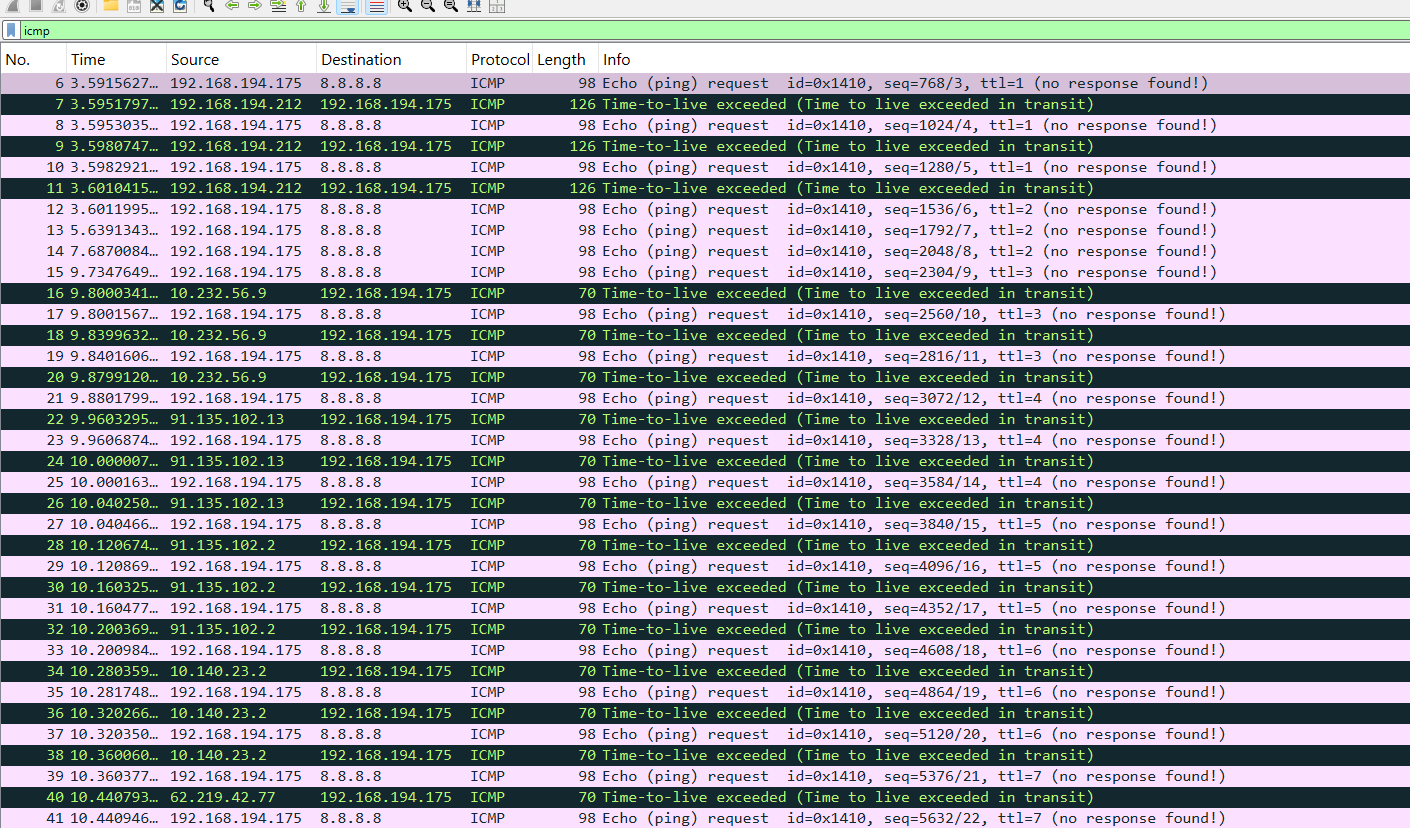


בתמונה אנחנו רואים את הפלט של הרצת תוכנת ה-traceroute שכתבנו כשניסינו להגיע לכתובת של שרת ה-DNS של Google (8.8.8.8).הפלט מראה את המסלול המדויק שהחבילה עברה ברשת עד להגעה ליעד-החבילה התחילה מהרשת שלנו(192.168.194.212) עם זמן תגובה של 5.79 מילישניות. לאחר מכן היה timeout בקפיצה השנייה, כלומר לא התקבלה תשובה מהנתב בנקודה זו. החבילה המשיכה דרך מספר נתבים שונים, כשכל אחד מהם מציג את כתובת ה-IP שלו וזמן התגובה. אנחנו יכולים לראות שהחבילה עברה דרך 13 נתבים שונים בדרך, כשבסופו של דבר הגיעה ליעד הסופי - 8.8.8.8, עם זמן תגובה של 50 מילישניות. ההודעה "Destination reached" מאשרת שהגענו בהצלחה ליעד המבוקש.

**הקלטות wireshark:**

התחברנו מהטלפון הנייד על מנת להמחיש כמות גדולה של קפיצות.

**כל חבילות ה- icmp:**

****

**A screenshot of a computer screen

Description automatically generated**

**כעת ננתח ונסביר:**

ההודעות בוורוד(חוץ מהאחרונה) אילו חבילות בקשה שעליהן לא התקבלו תשובה,ניתן לראות שרוב הבקשות מסומנות כ"no response found", מה שאומר שלא התקבלה תשובה עבורן, וזה מתואם עם ה-timeouts שראינו בפלט של ה-traceroute.

על כל הודעה נשלח 3 probes כלומר 3 נסיונות להגיע עם טיימאוטים שונים.

ההודעות השחורות זה החבילות שמסומנות כאן בשחור שמציינות שה- TTL שלהן עבר ובתוכנית שלנו אנחנו קלטנו את ICMP\_TIME\_EXCEEDED על מנת שזה יציג לנו את הhop בדרך. בעמודת ה-Info אנחנו יכולים לראות פרטים נוספים על כל חבילה, כמו מספר הרצף (seq)וה-TTL.בהודעה האחרונה אנו מקבלים תשובה שאכן הגענו ליעד שלנו שהוא 8.8.8.8.