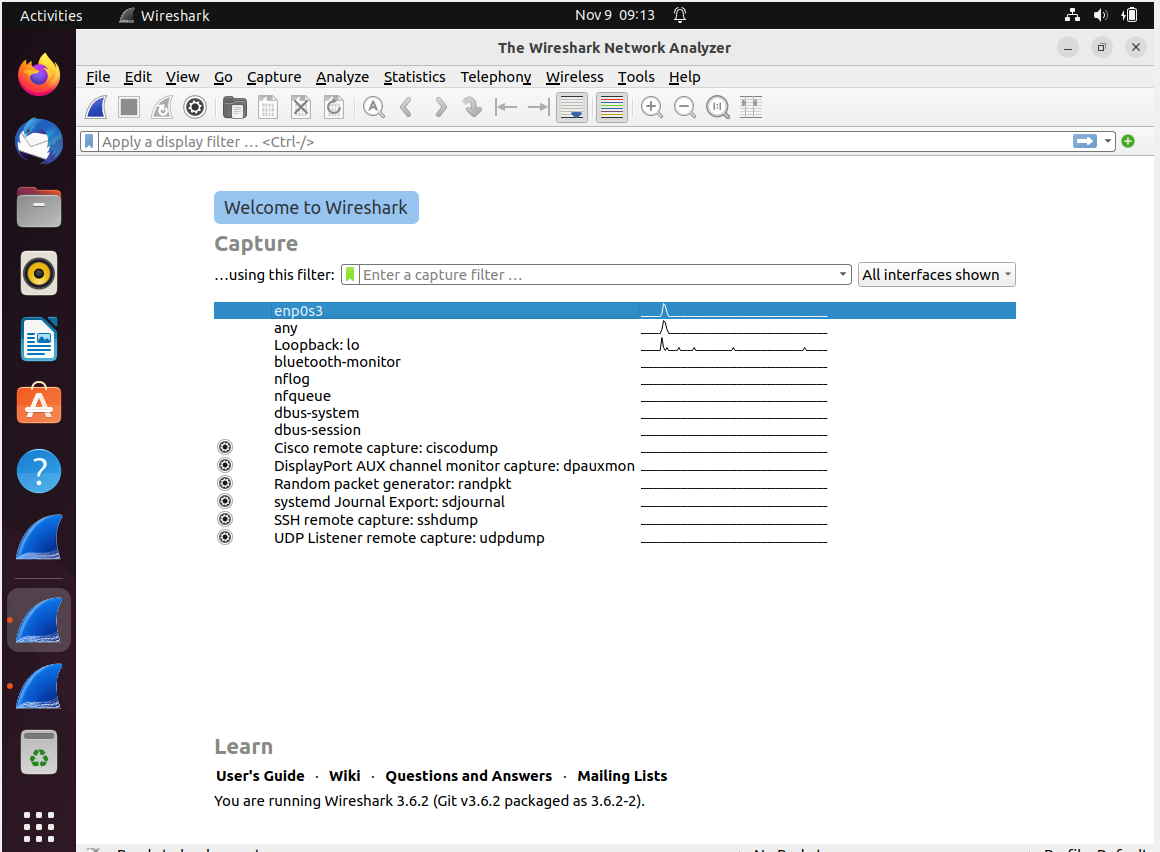
מטלה 1 – תקשורת ומחשוב

פרק 1

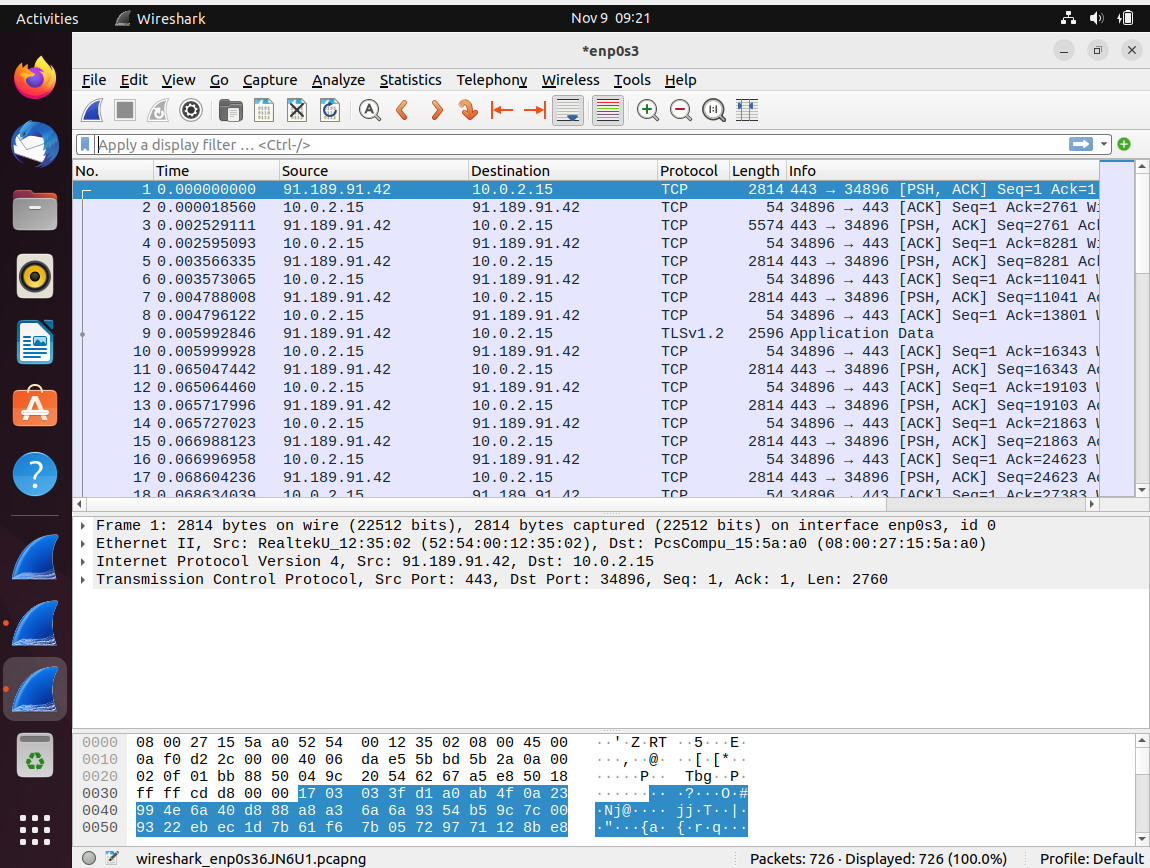
208113381 מהונד ספי

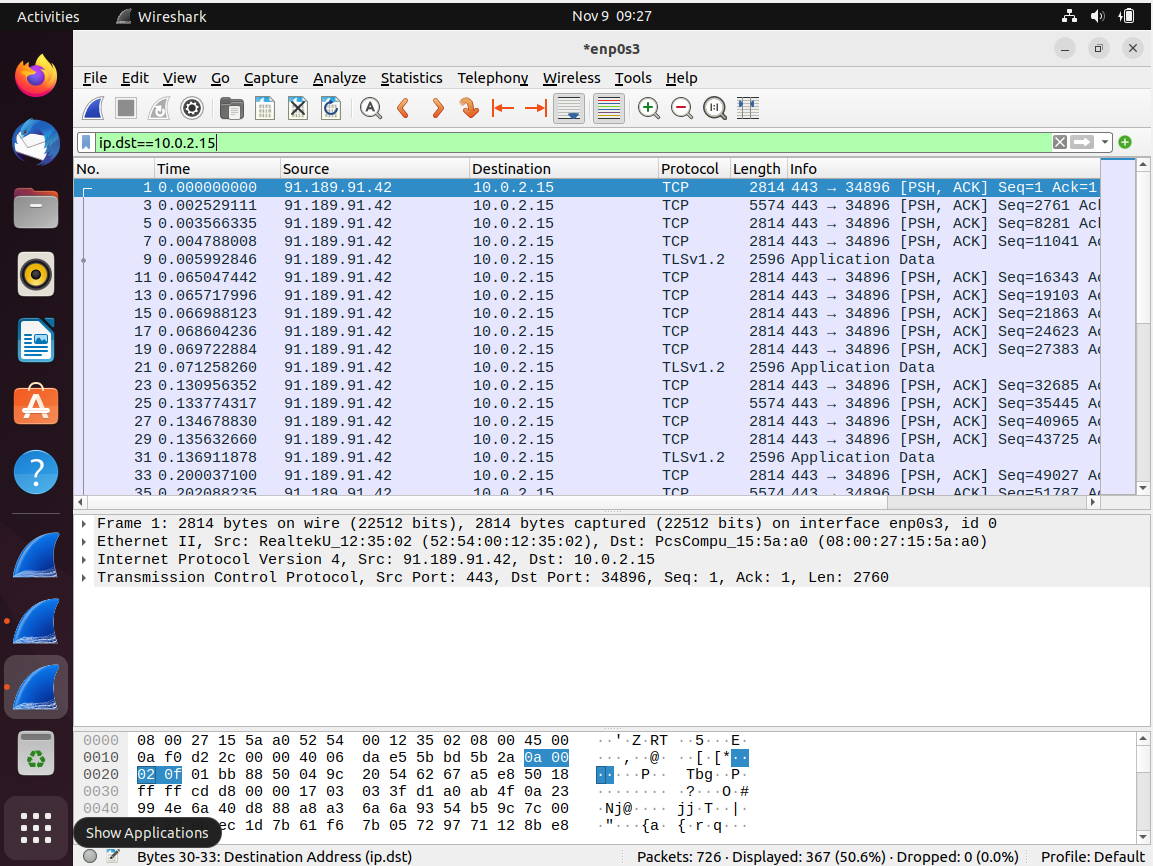
ניתאי לוי 319096251

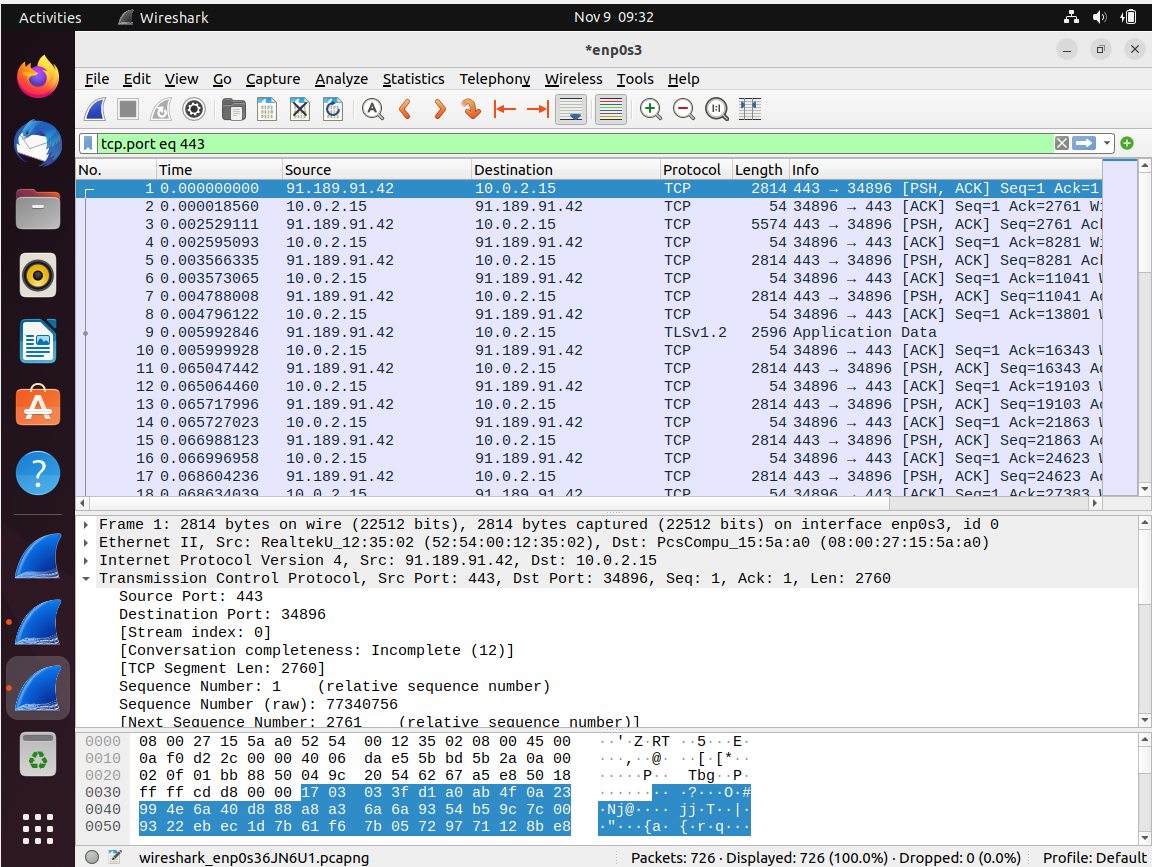
שאלה 1

צילום מסך של בחירת enp0s3.

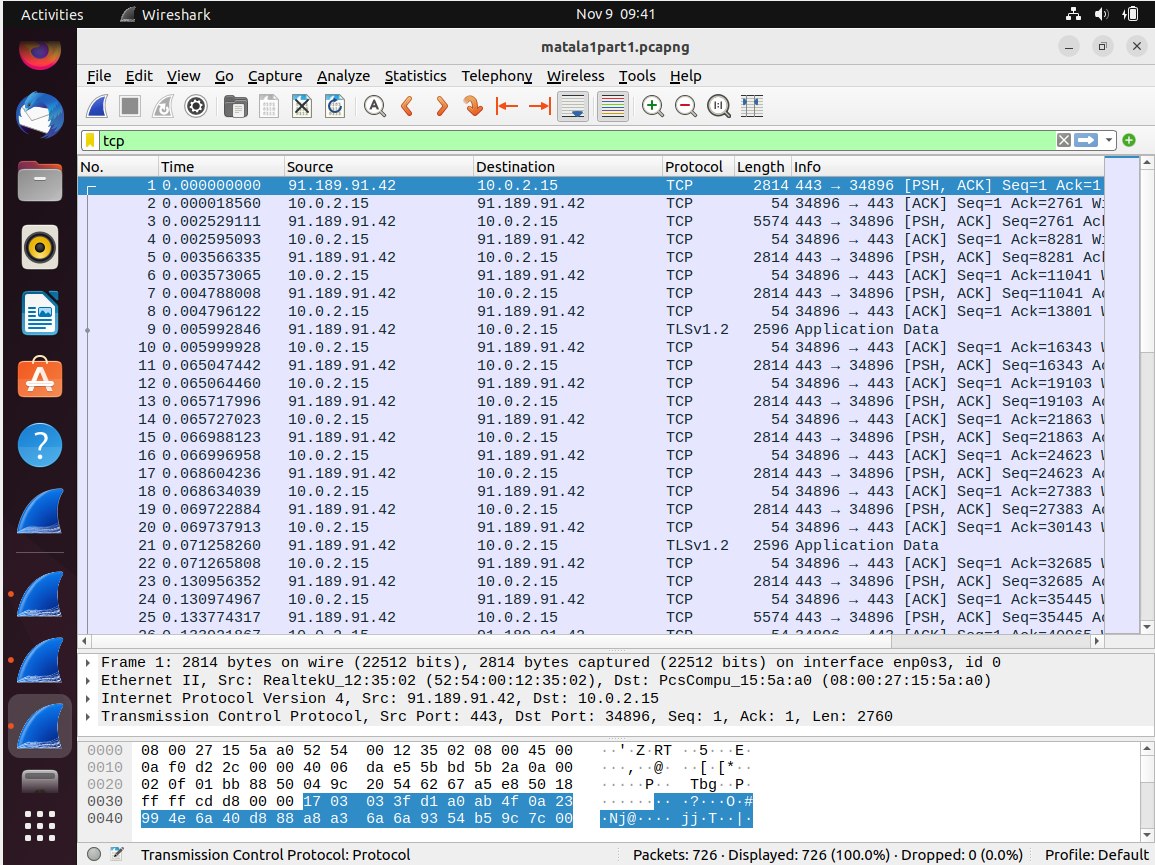
שאלה 2

צילום מסך של קליטת פאקטות.

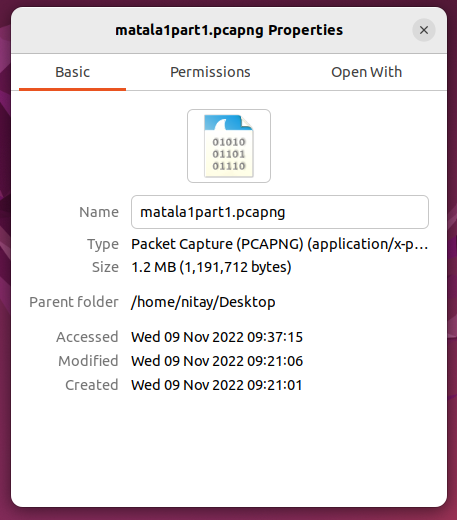
1. סינון החבילות לפי ip destination של 10.0.2.15.
2. סננו את כל החבילות לפי port source 443. הפרוטוקול הוא TCP.



1. סינון של החבילות לפי פרוטוקול TCP.

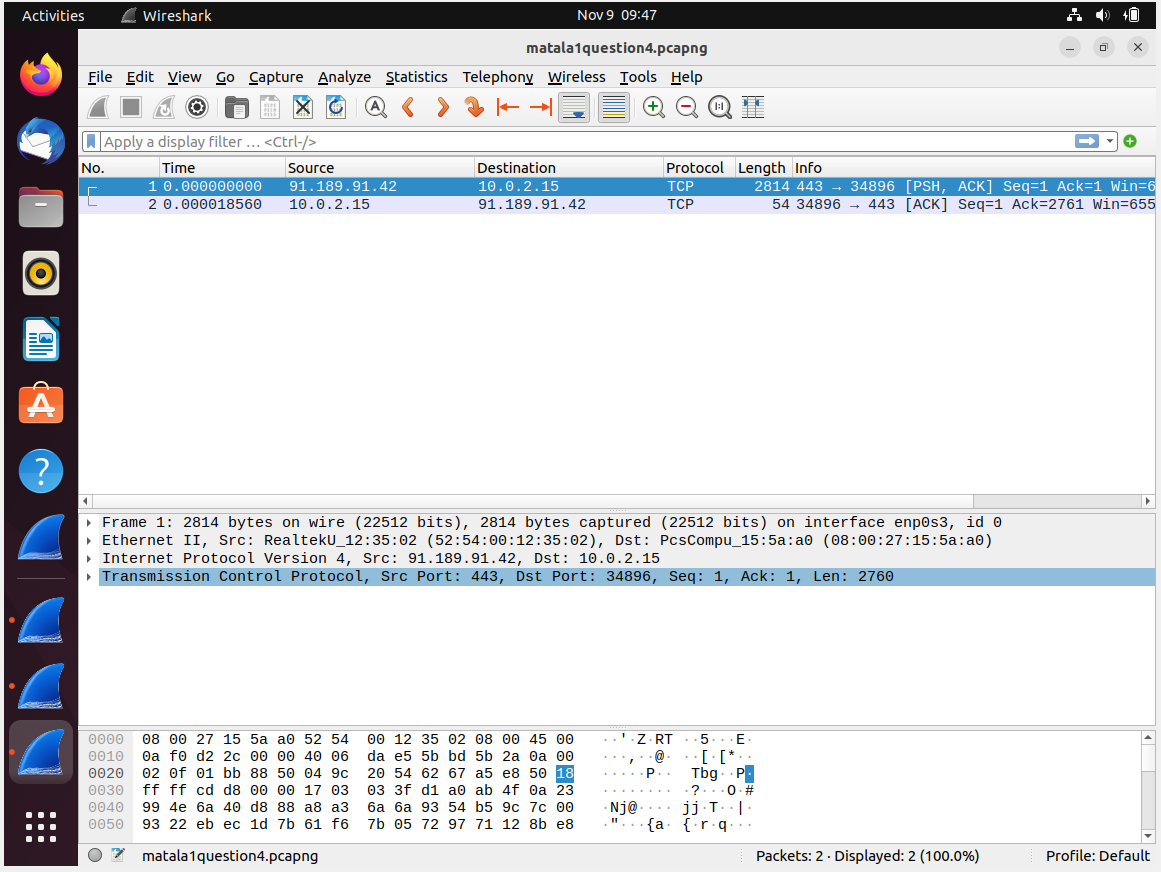


שאלה 3

פורמט ההקלטה הוא pcapng.

שאלה 4

בחלונית files יש אפשרות export specific packets, שמרו שתי פאקטות כלשהם לבחירתכם בפורמט הנ"ל.



**צעדים:** בחרנו בשני הפאקטות, לאחר מכן לחצנו ב-file ואז export specific packets וסימנו selected packets והענקנו לקובץ את השם matala1question4.

**שאלה 5**

בהגדרות קיים אפשרות להפעיל mode promiscuous, הסבירו בקצרה מה זה עושה ומה השימושים.

במצב mode promiscuous כרטיס הרשת מפסיק לפלטר רק את ה-frames שמיועדות אליו.

מצב זה אומר שהסניפר כעת מאזין לכל התעבורה שעוברת דרך כרטיס הרשת ולא רק לתעבורה שעוברת בטווח שלו, כך שהוא יכול לקבל גם את חבילות המידע שהגיעו לכרטיס הרשת של המחשב, אל לא היו מיועדות וליו , ולכן מערכת ההפעלה לא העלתה אותן לעיבוד.

במצב כזה הרחרחן עובד ברמת כרטיס הרשת ולא ברמת מערכת ההפעלה.

(כדי לאפשר לרחרחן לקבל גם תעבורה שלא מיועדת למחשב עליו מורץ בלבד).

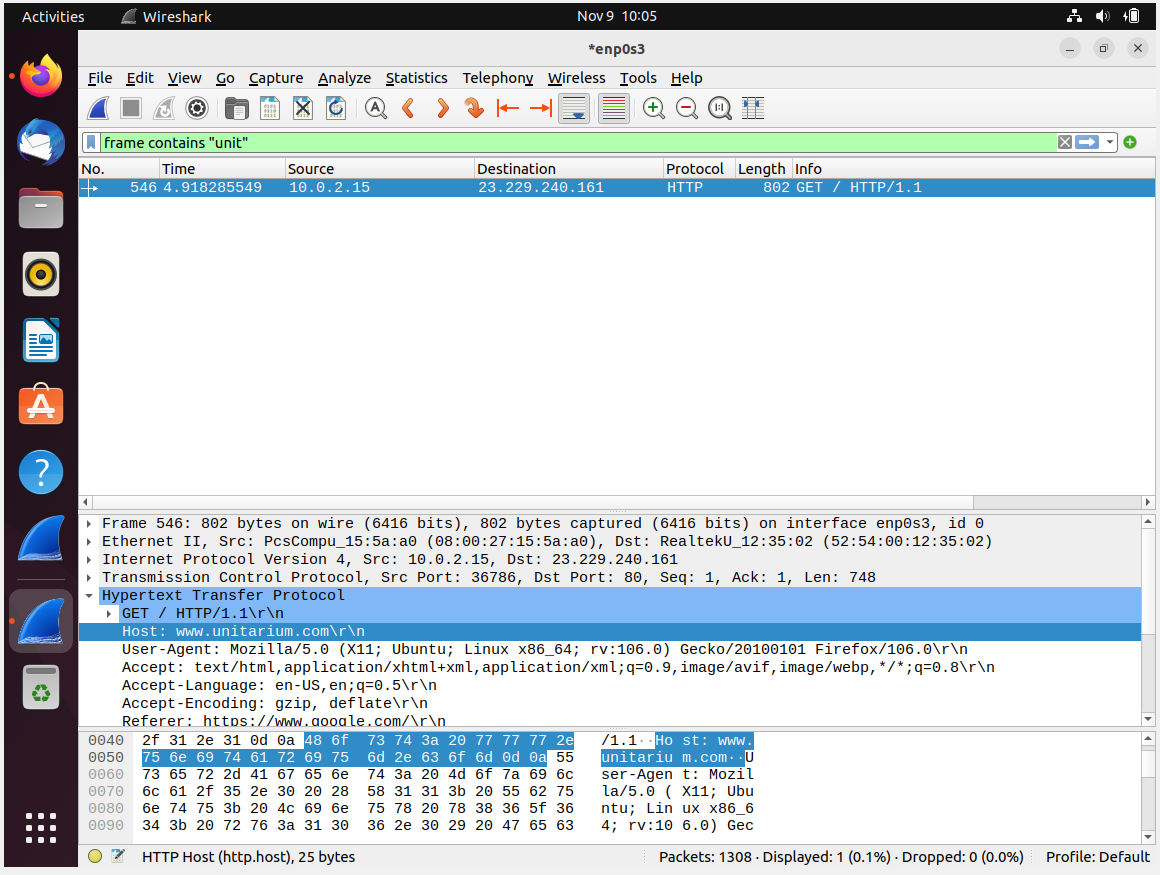
It enables a network card to capture traffic that is addressed to other devices on the network, not just the local hardware address.

Normally a network interface will only “receive” packets directly addressed to the interface.

Promiscuous mode allows the interface to receive all packets that it sees whether they are addressed to the interface or not.

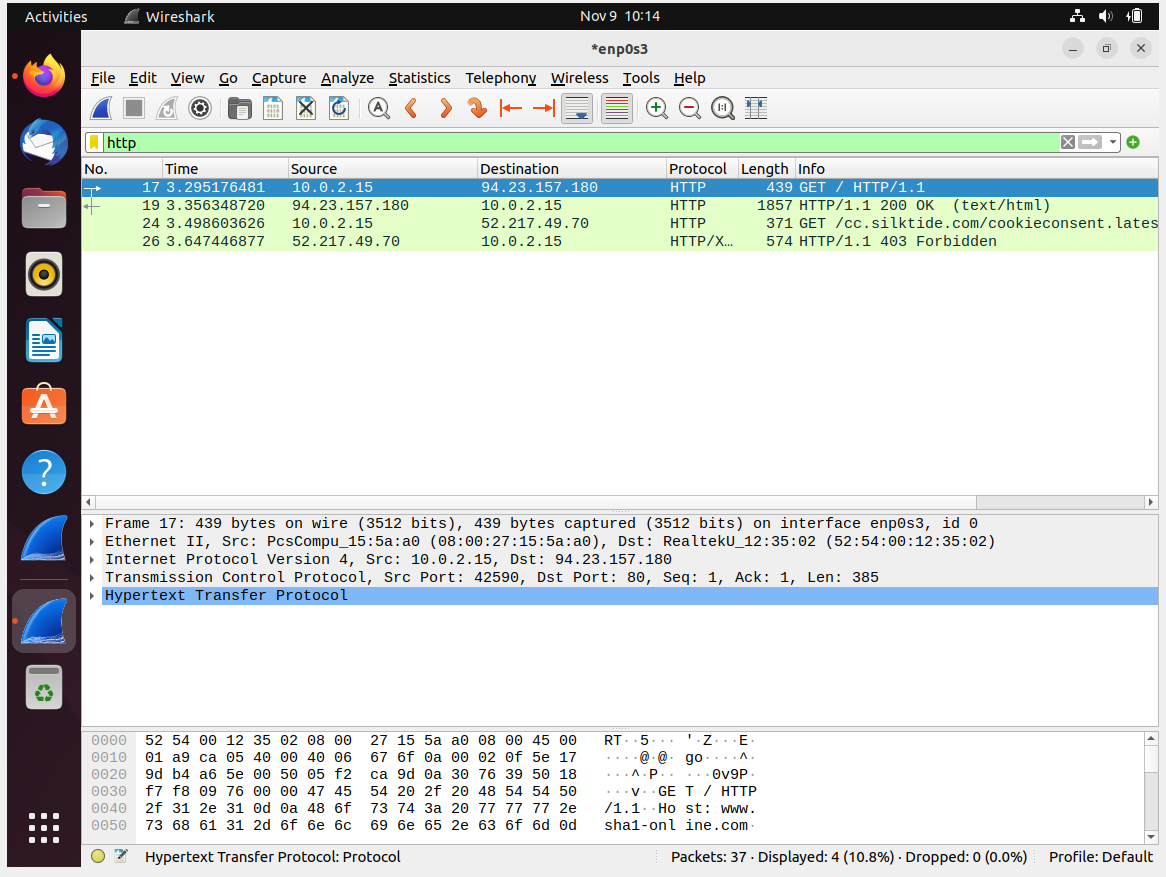
(Using the compare with mac address (physical address) for the computer and Wireshark, then Wireshark don't see it)

when a network card is in promiscuous mode it accepts all packets, even if the destination MAC of the frame does not match its own MAC. Broadcasts are accepted anyway. Without promiscuous mode frames with MACs other than the one the interface has are ignored.

שאלה 6

הסינון נותן את כל הפאקטות שמכילות בתוכן שלהם את המחרוזת "unit".

פרק 2 - HTTP

שאלה 7

לפי החץ לכיוון ימינה ולפי המילה GET ב-info ניתן לזהות שפאקטה זו היא חבילת הבקשה.

לפי החץ שמאלה ולפי שכבה 5 בה רשום [HTTP response 1/1].

שאלה 8

1. הבקשה היא פאקטה מס' 17 והתגובה היא מס' 19 וזה הזמן.

הבקשה היא פאקטה מס' 24 והתגובה היא מס' 26 וזה הזמן.



1. גרסת ה-http היא 1.1.
2. הבקשות נשלחו ממערכת ההפעלה של Ubuntu מהדפדפן firefox.



1. החבילה נמצאת בסרבר nginx במערכת הפעלה Ubuntu.



1. פורט היעד הוא 80.



שאלה 9

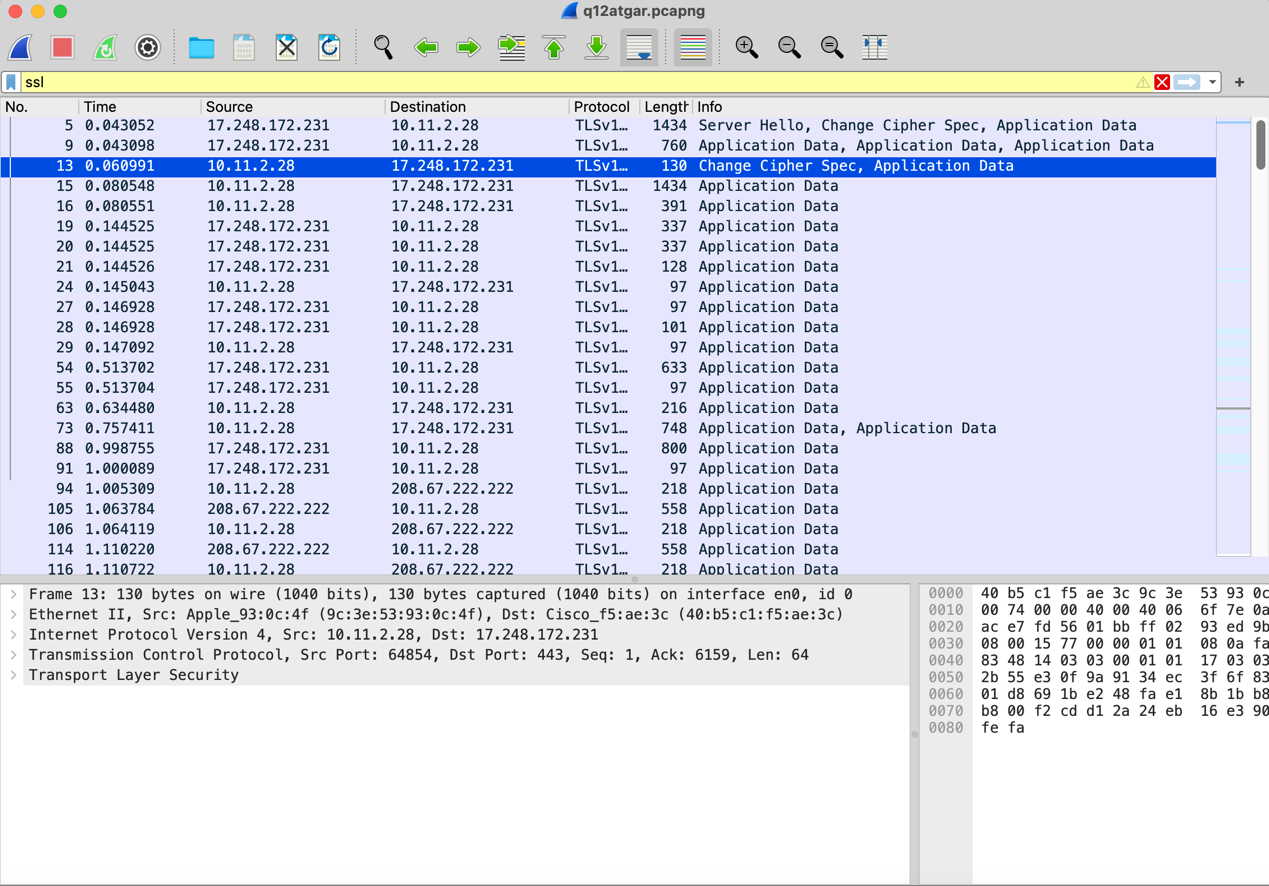
1. סטטוס קוד התגובה הוא 200, כלומר הבקשה עברה בהצלחה.



1. מהשרת nginx ומה-ip 94.23.157.180.





1. מספר חבילות ה-TCP שנדרשו כדי להרכיב את החבילה הוא 2.
2. 



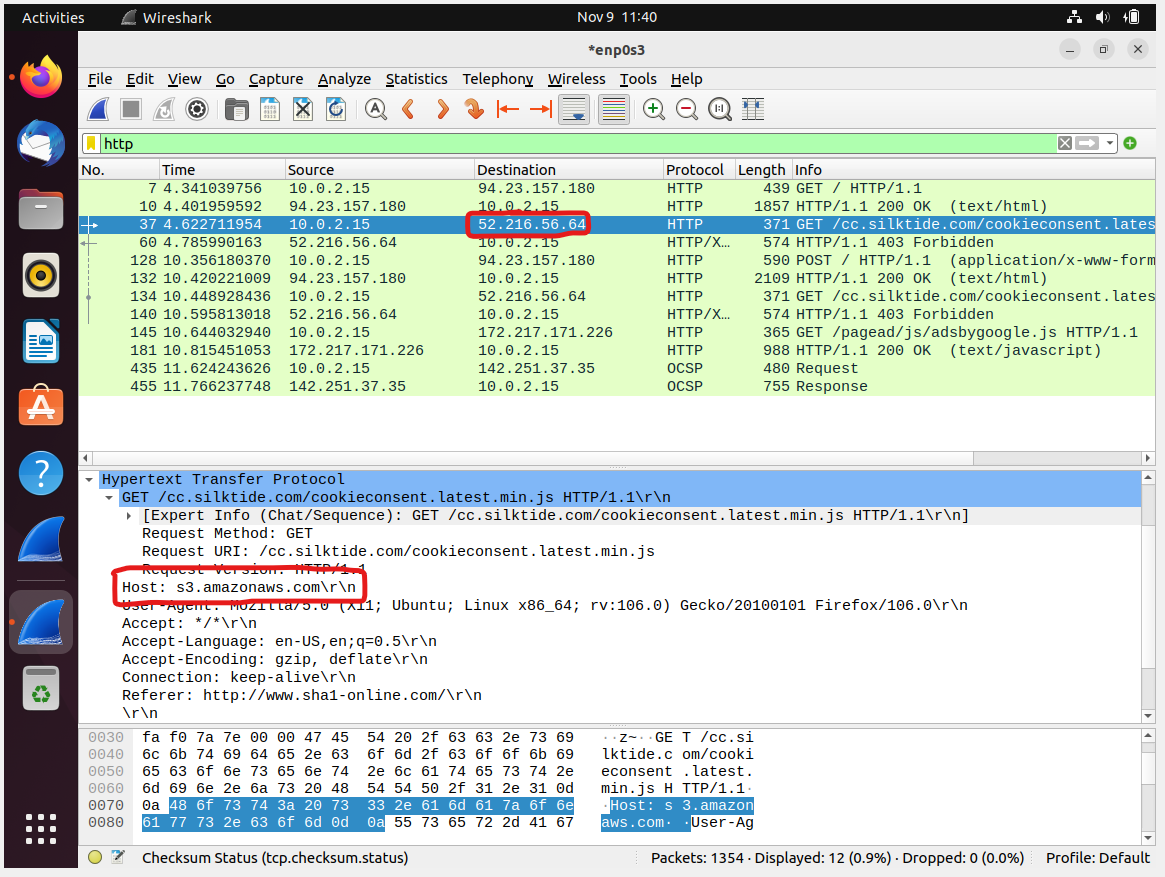
1. סוג החיבור בין השרת ללקוח הוא presistent http.



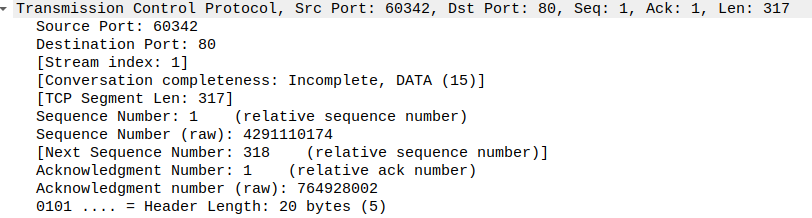
חיבור זה משמעו לאפשר קשר TCP יחיד שנשאר פתוח עבור העברות רבות של אוביקטים מהשרת ללקוח ולהפך, במקום כל פעם לפתוח ולסגור את החיבור לשרת.

שאלה 10

1. נשים לב שהחישוב נעשה בשרת אחר לפי זה שהשרת התחלף ל- 52.216.56.64 וה-host התחלף.



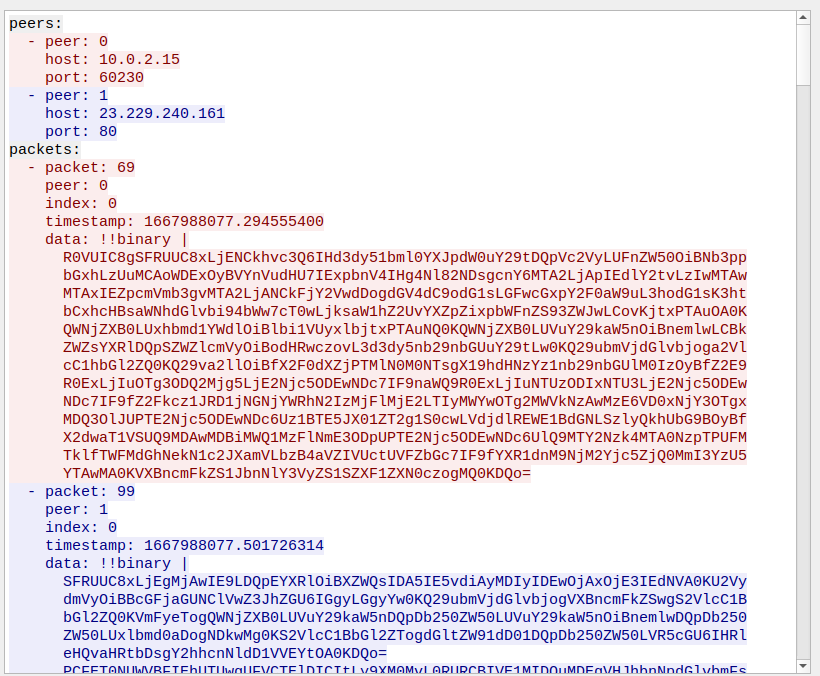
1. נשלח כתובת השולח וכתובת שרת היעד. בנוסף, מידע שעליו אנו רוצים לבצע את החישוב.



1. כדי לא להעמיס על השרת חישובים רבים מסוג זה הוא שולח לשרת אחר שיעשה זאת עבורו באופן יעיל יותר. לכן היה צורך לבצע תקשורת http ולא לבצע את החישוב בצד בלקוח.
2. סיכון אחד הוא זמן ביצוע איטי יותר כי אנו נדרשים ליצור קשר כל פעם על מנת לבצע חישוב זה. בנוסף, במקרה שהשרת שמבצע את פעולת החישוב עבורנו לא פועל בגלל תקלה למשל, לא נוכל לבצע את החישוב כי אנו תלויים בו.

שאלה 11

1. נשלחו 69 חבילות מהלקוח ו-99 מהשרת. השרת החזיר אובייקטים של האתר ואת המידע שביקשנו.



שאלה 12

Graphical user interface, table

Description automatically generated

הסבר:-

לאחר התחלת ההקלטה של חבילות , נכנסת לאתר יוטיוב , ואז עשיתי פילטר של ssl , שזה בעצם את האבטחה שנוספה לפרוטוקול http ו להפוך ל https

זה ייתן לי את כל החבילות עם פורט 443 שזה אומר לאתרים שהם מושדרגים עם אבטחה.

קצת הסבר על האבטחה הנוספה:-

הלקוח מבקש משרת לכנס לאתר ואז השרת יישלח לו קוד ומידע כדי לאמת איתו לזהות שהוא הלקוח האמיתי , אחר כך אם מצליח אז עושה קישר (ובטח שכל זה דרך מפחת ספציפי לשרת ומפתח פופלרי ללקוח)

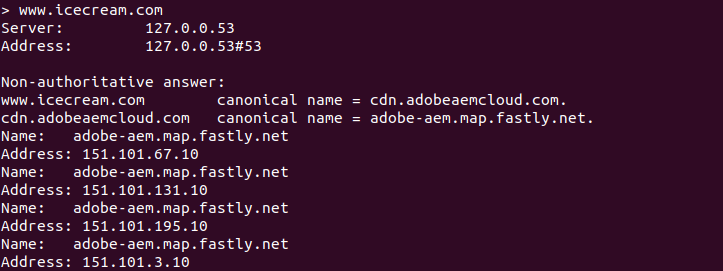
וכן הלא...

ואז זה עוזר שהמידע המועבר בין השרת והלקוח יהיה סגור למישהו אחר, כלומר לא יכלו לקרא את המידע (נגיד כשפה אחרת רק שני הקצוות יודעים אותה)

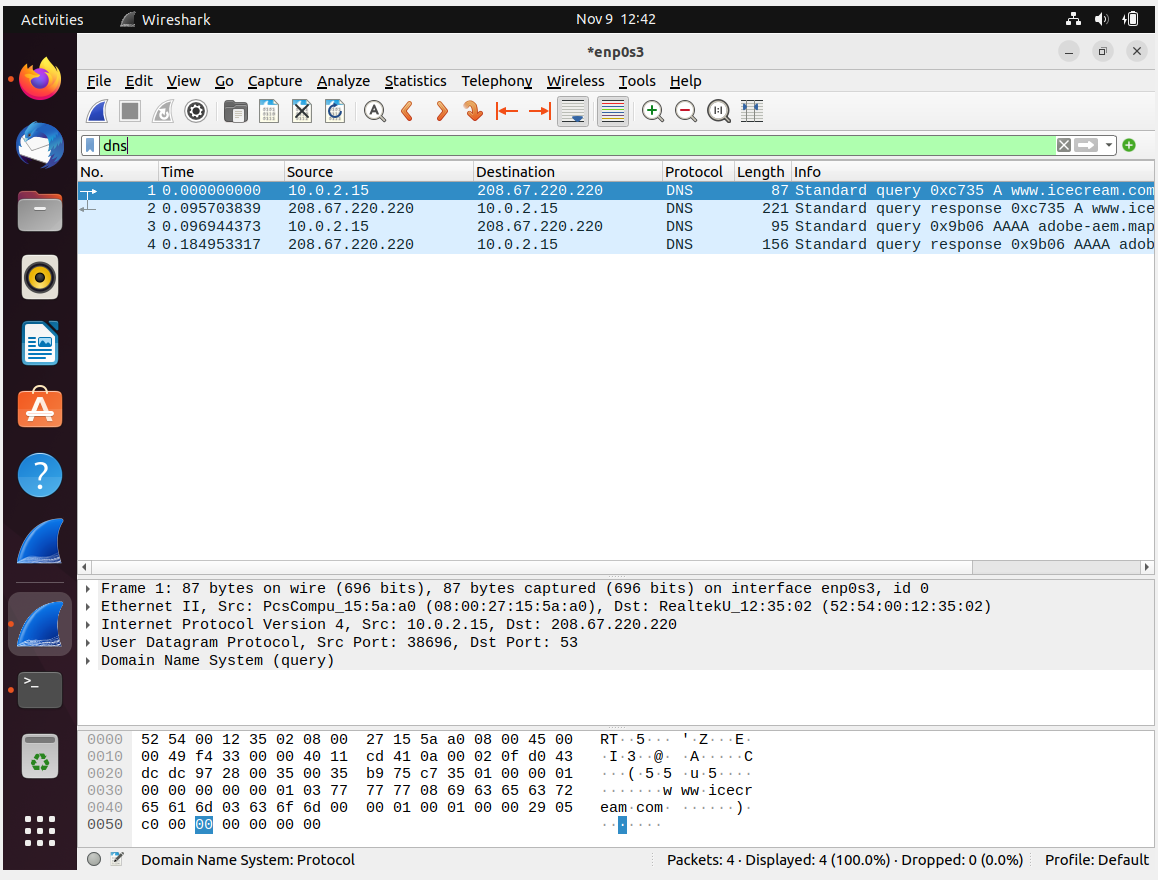
ואז זה עוזר למנוע את ההתקפות...

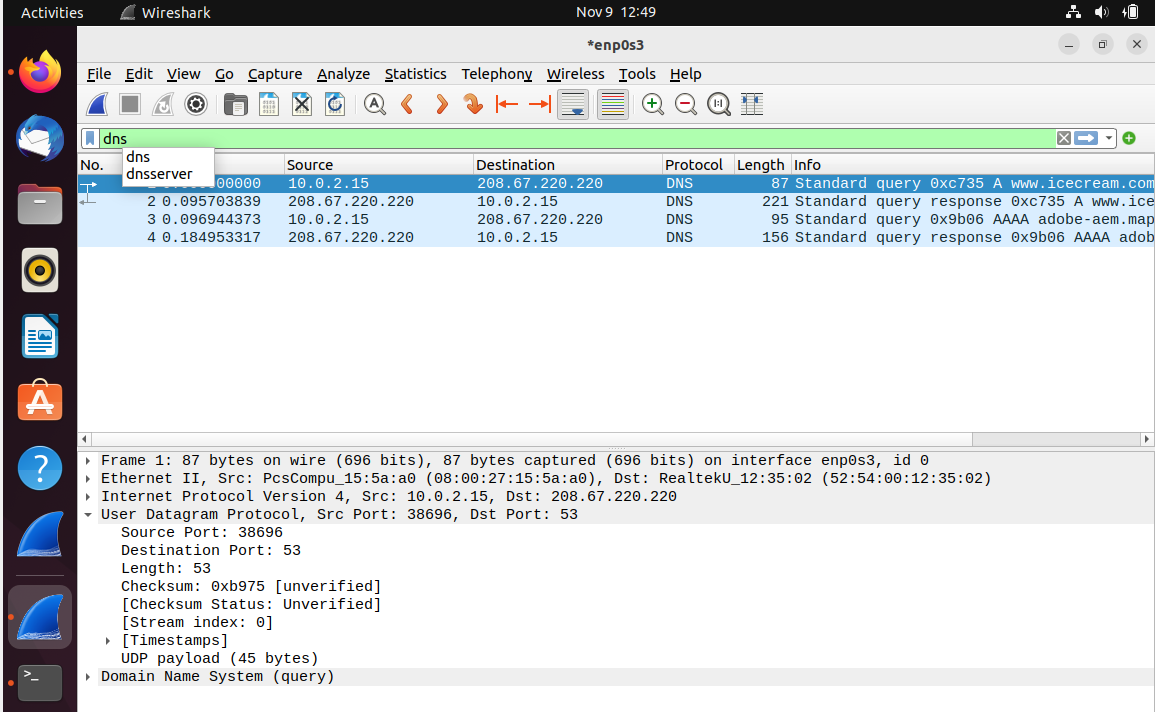
שאלה 13

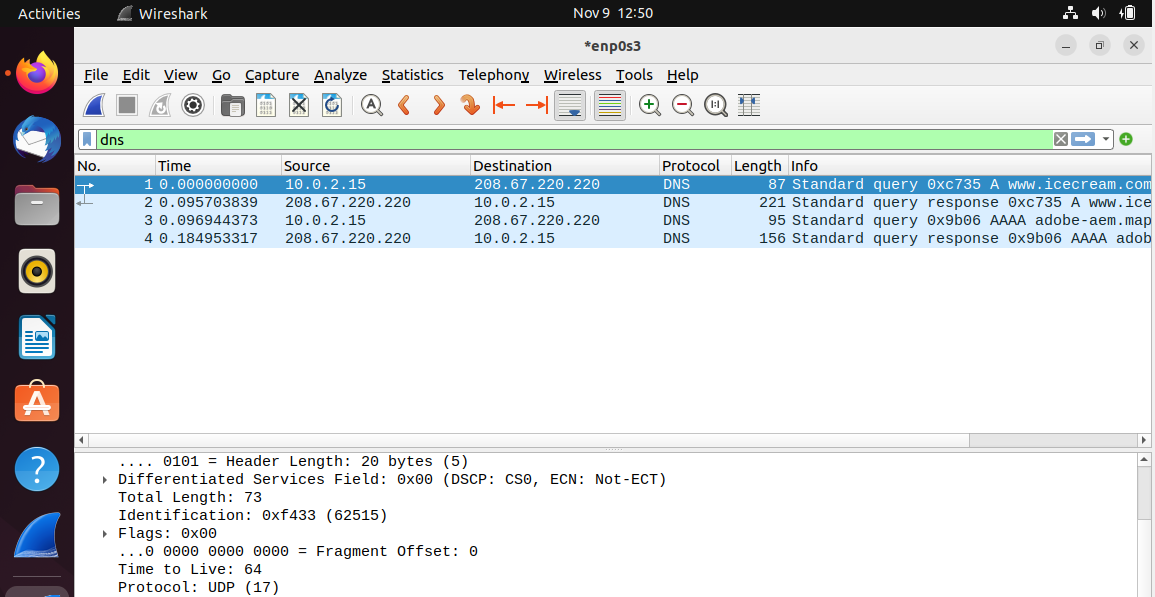
שם השרת הוא adobe-aem.map.fastly.net

 הוא לא שרת מהימן. אחת מכתובות ה-ip שלו היא 151.101.67.10.

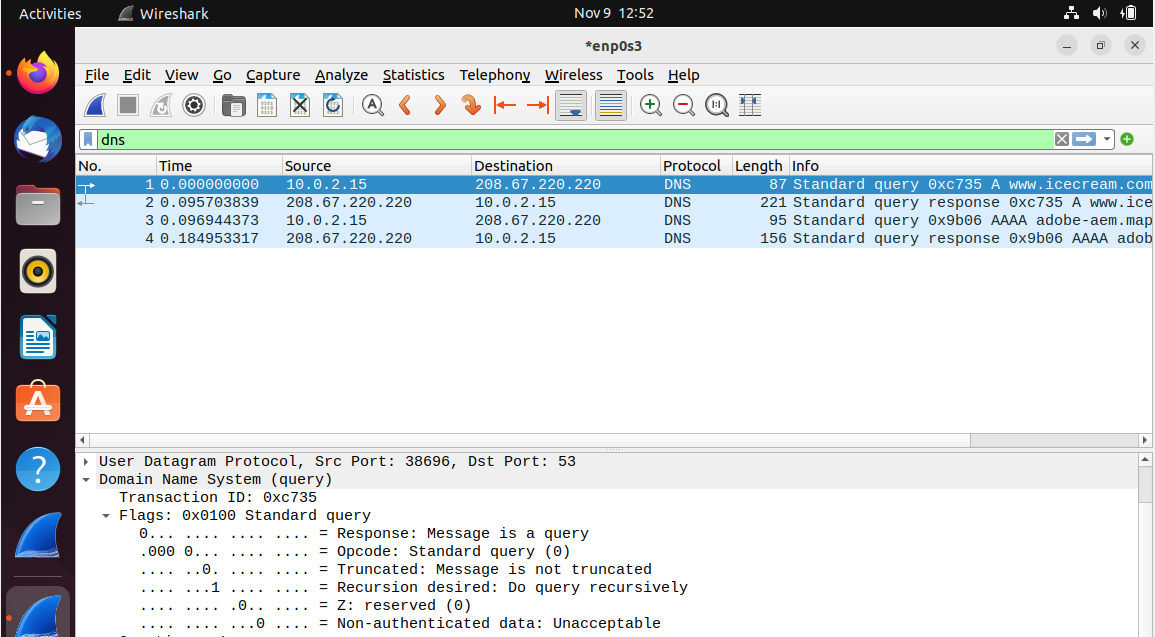
שאלה 14



1. ארבע פאקטות.
2. שני השאילתות הראשונות עובדות עם אינטרנט IPV4 והשניים אחרי כן עובדות עם IPV6.
3. לפורט 53.
4. UDP

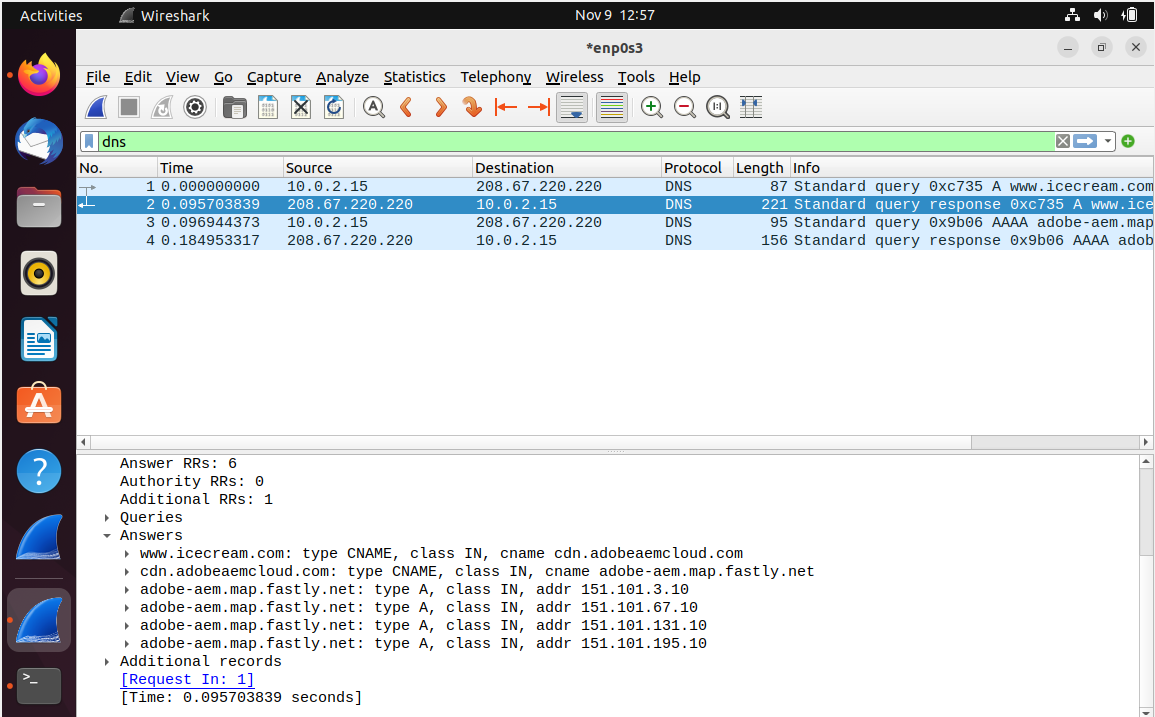


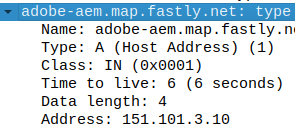
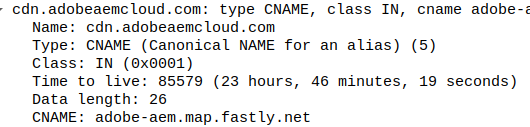
1. בצורה רקורסיבית. מכיוון שרשום do query recursively והמספר 1 כלומר שהוא מופעל.



1. בתגובה לשאילתה הראשונה התקבלו 6 תשובות. ההבדל בין התשובות הוא כתובת ה-ip ששונה בכל אחד, ה-data length ו-Timetolive.

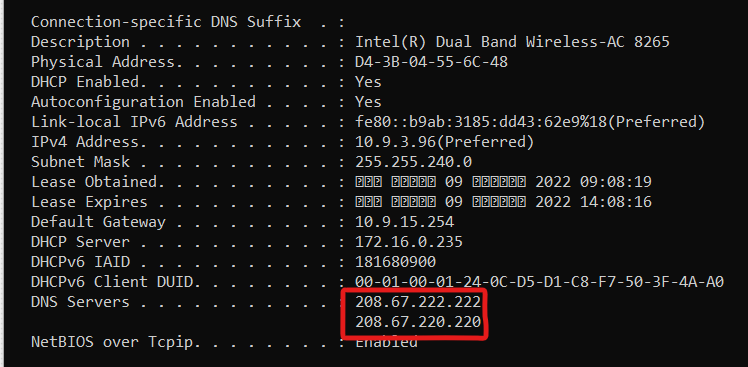
בשאר השאילתות לא התקבלו כלל תשובות.





1. ההבדל הוא סוג ה-IPV: כאשר type הוא A הוא שייך ל-IPV4. כאשר type הוא AAAA הוא שייך ל-IPV6. ההבדל הוא למעשה בכמות הביטים.

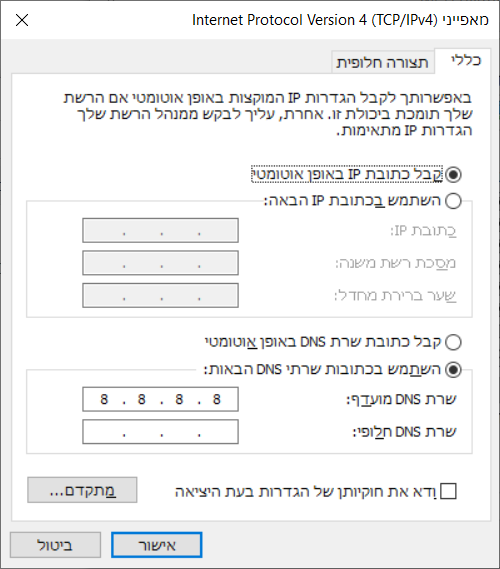
שאלות 15 ו-16

מס' שרתי ה-DNS הם 208.67.222.222 וגם 208.67.220.220.

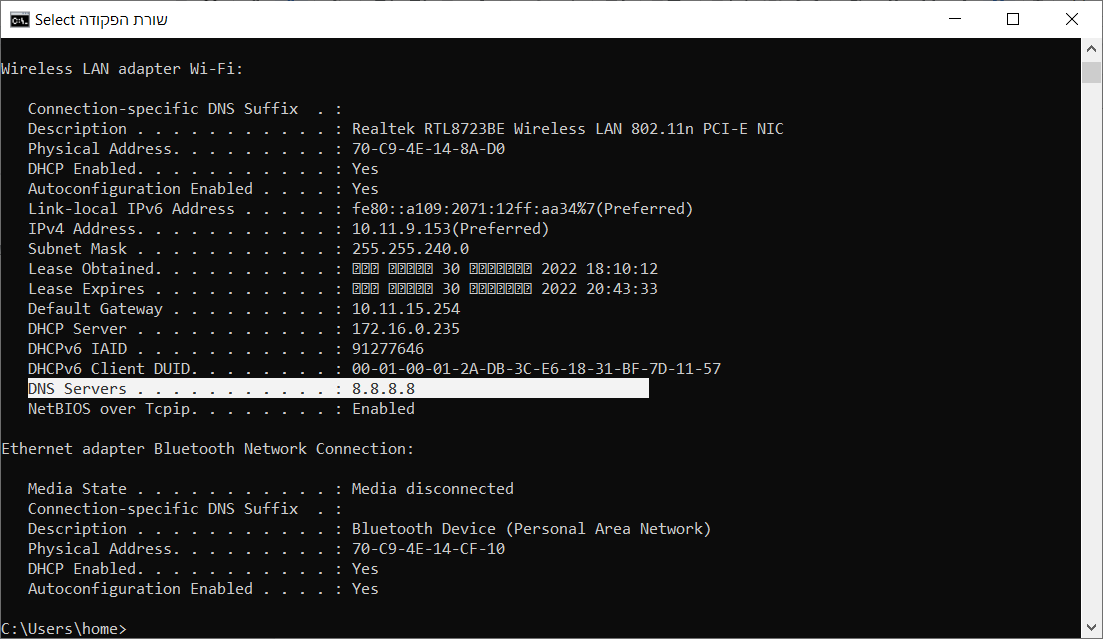
בנוסף, ה-DNS פועל גם על IPv6 ששם השרת שלו מופיע בריבוע העליון מצד ימין.

שאלה 17

**לאחר שינוי את  שרת ה dns ל 8.8.8.8**

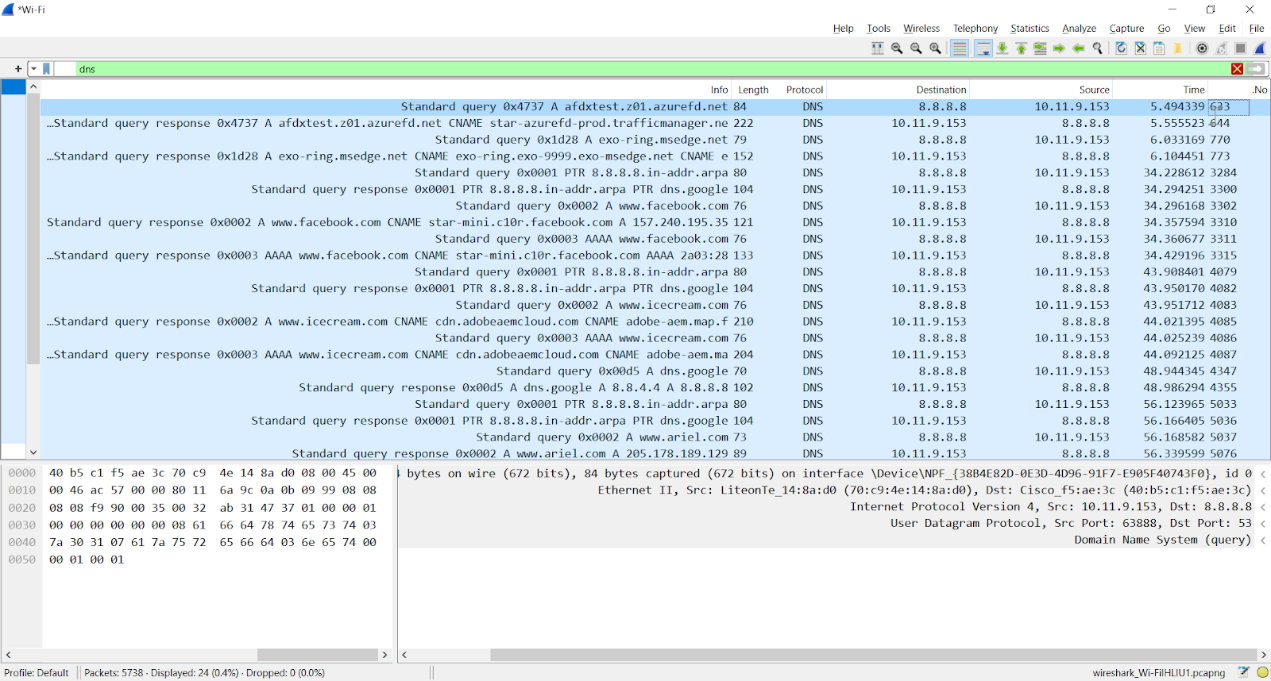
****

**בדיקה בעזרת ipconfig/all**

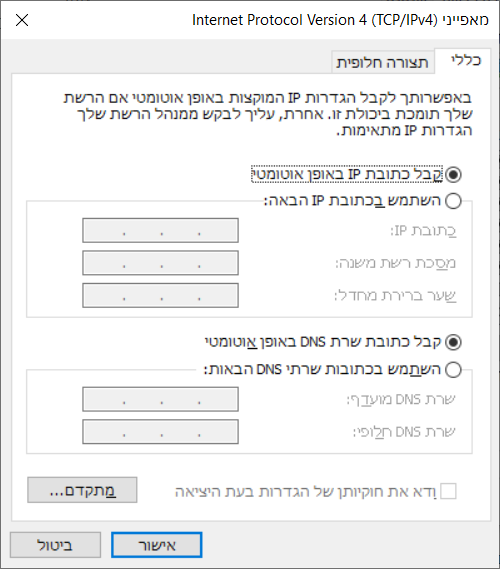
****

**לאחר נכנסת לכמה אתרים , רואים את 8.8.8.8**

****

****

**החזרה להקצאה אוטומטית**

****

שאלה 18

חבילות או שאילתות התגובה של dns , הן יותר כבדות משאילתות הבקשה.

הסיבה , שבתגובה יוחזר הרבה מידע , על מני התכליך על היריכה של dns כדי לקבל את הדרוש , בנוף ה data , כל זה יועטף עם הכתובות מאיזה שרת קיבלנו ולאן ולמי,ומה הזמן הרבה שדות יותר מחבילת הבקשה, ואז יוחזר ללקוח.

שאלה 19

1.

****

2.

****

3.

****

**Time to live:232**

שאלה 20

ידוע שבתקשורת אינטרנט לא ניתן להתחבר אל דומיין, אלא דרושה כתובת IP

נגיד שרוצים לכנס לאתאר [www.google.com](http://www.google.com) מה שקורה כך:

שלב התרגום – תרגום שם של דומיין )במקרה זה – .DNS פרוטוקול באמצעות נעשה זה תרגום שם של הדומיין ל ip המתאים לו באמצעות פרוטוקול dns ,לדוגמא שלנו יהיה ip(8.8.8.8).

פרוטוקול DNS פועל באמצעות בקשה שנקראת גם שאילתה ותשובה.

בטרם הדפדפן ניגש אל האתר המבוקש , על מערכת ההפעלה למצוא את כתובת ה -IP ה רלוונטית.

לשם כך, המחשב שלנו מתשאל שרת DNS

**כעת, כאשר ללקוח יש את כתובת ה -IP של שרת היעד, הוא יכול לפנות אליו באמצעות פרוטוקול .HTTP**

ואז הוא יכול להשתמש בה בכדי לתקשר עם השרת.

**כעת, באמצע המעבר מ dns ו http , יש את המעבר של מטמון cache**

המטמון עוזר לנו במקרה למשל , אם נכנס לאתר האוניברסיטה ביום עשרות פעמים לעומת שאנחנו נכנסין לאותא אתר בלי שום שינוי ואז כל המידע מחדש יועבר מהשרת ללקוח , אז כאן הבעיה והעבודה של מטמון.

זה גורם לתעבורה "מיותרת", שהרי המידע היה כבר בדפדפן שלכם בעבר, ומאז הוא לא השתנה! התעבורה המיותרת גורמת גם לבזבוז של עלויות )רוחב הפס וכמות המידע שעוברת על גביו(, וגם של זמן )בהמתנה למשאב שיגיע(.

הרעיון של מנגנון ה -Cache מטמון הוא לשמור משאבים כאלה על המחשב של הלקוח, וכל עוד הם לא משתנים, לטעון אותם מהדיסק המקומי ולא על גבי הרשת. המנגנון שמאפשר לבצע זאת בפרוטוקול HTTP נקרא Get-Conditional בקשת GET מותנית

. משמעות השם: בקשת ה-GET תתבצע רק בתנאי שלא קיים עותק מקומי ועדכני של המשאב.

בשלב הראשון, הלקוח פונה בבקשת משאב מסוים. הלקוח מציין שברצונו לקבל את המשאב, רק אם זה שונה מתאריך מס וים.

הלקוח מבקש את המשאב html.file .הלקוח מבקש שהשרת ישלח לו את קובץ זה, אך ורק אם הוא השתנה מאז ה-2014.01.01 .הסיבה היא, שהלקוח שמר בתאריך זה את הקובץ html.file ל -Cache שלו. לכן, אם הקובץ לא השתנה מאז ה-2014.01.01 ,הרי שהעותק שעל השרת זהה לעותק שנמצא ב-Cache של הלקוח. במקרה שבו הקובץ לא השתנה, השרת עונה בתשובת Modified Not 304 H.

בשלב זה, הלקוח יכול לטעון את המשאב מתוך ה-Cache ,ולא לקבל אותו מהשרת.

במקרה אחר, י י תכן שהקובץ התעדכן, נאמר ב -2014/03/01 .השרת יחזיר את הק ובץ המעודכן בתשובה.

במקרה זה, הלקוח מבין שהעותק השמור אצלו אינו עדכני. הוא טוען את הקובץ מהשרת, ומחליף את הקובץ הקיים אצלו ב -Cache בקובץ החדש.