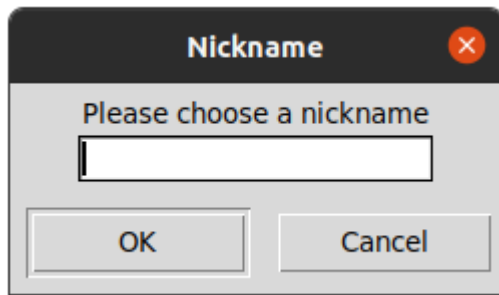
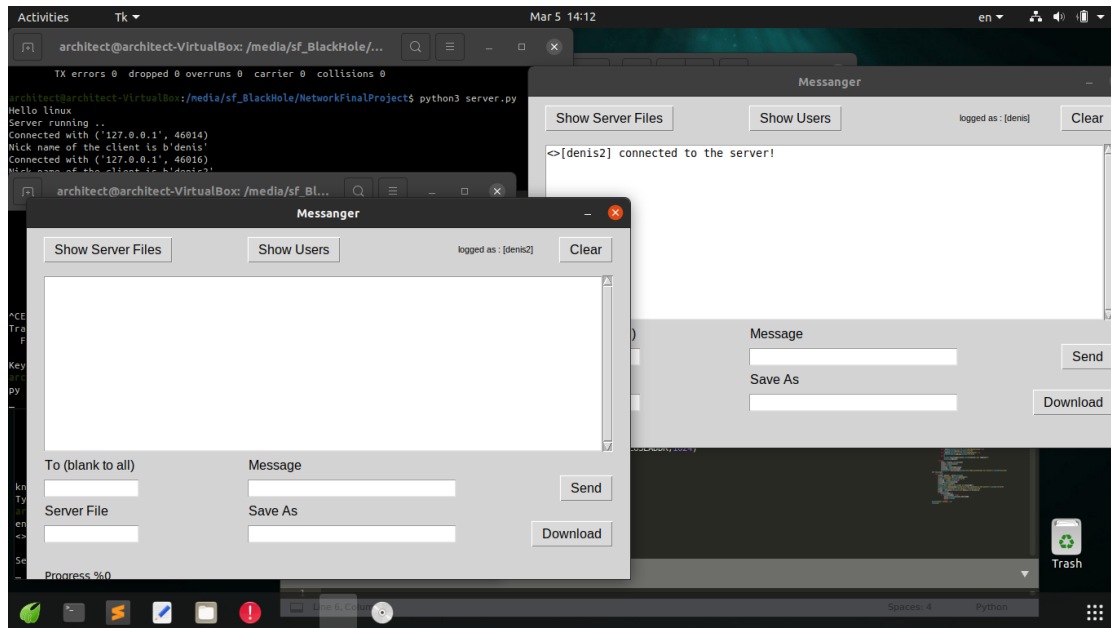


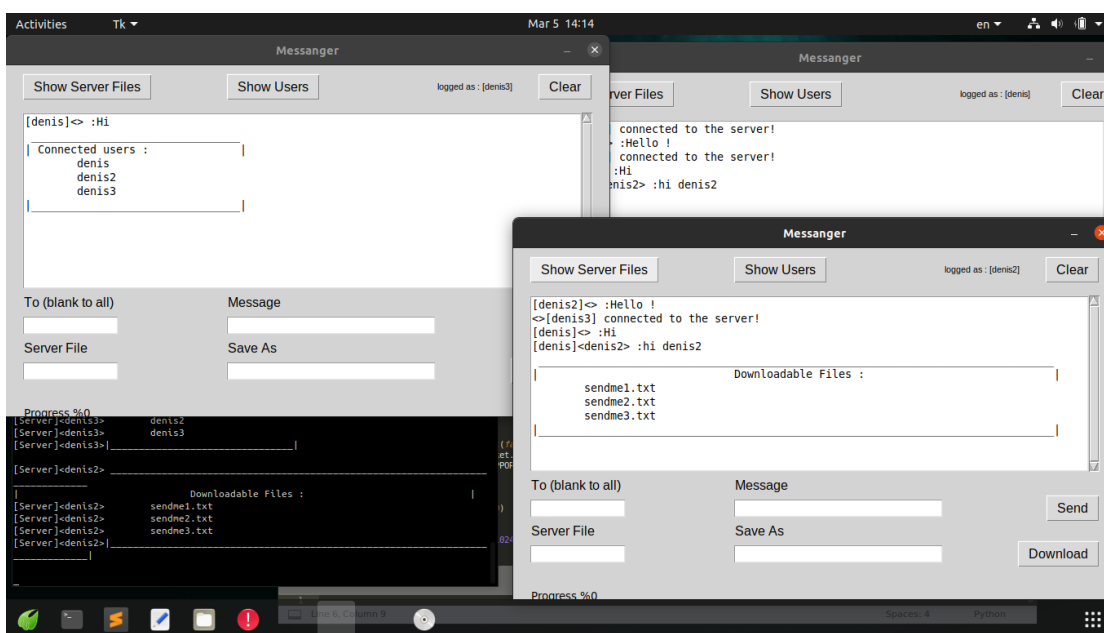
## חלק א (תמונות):

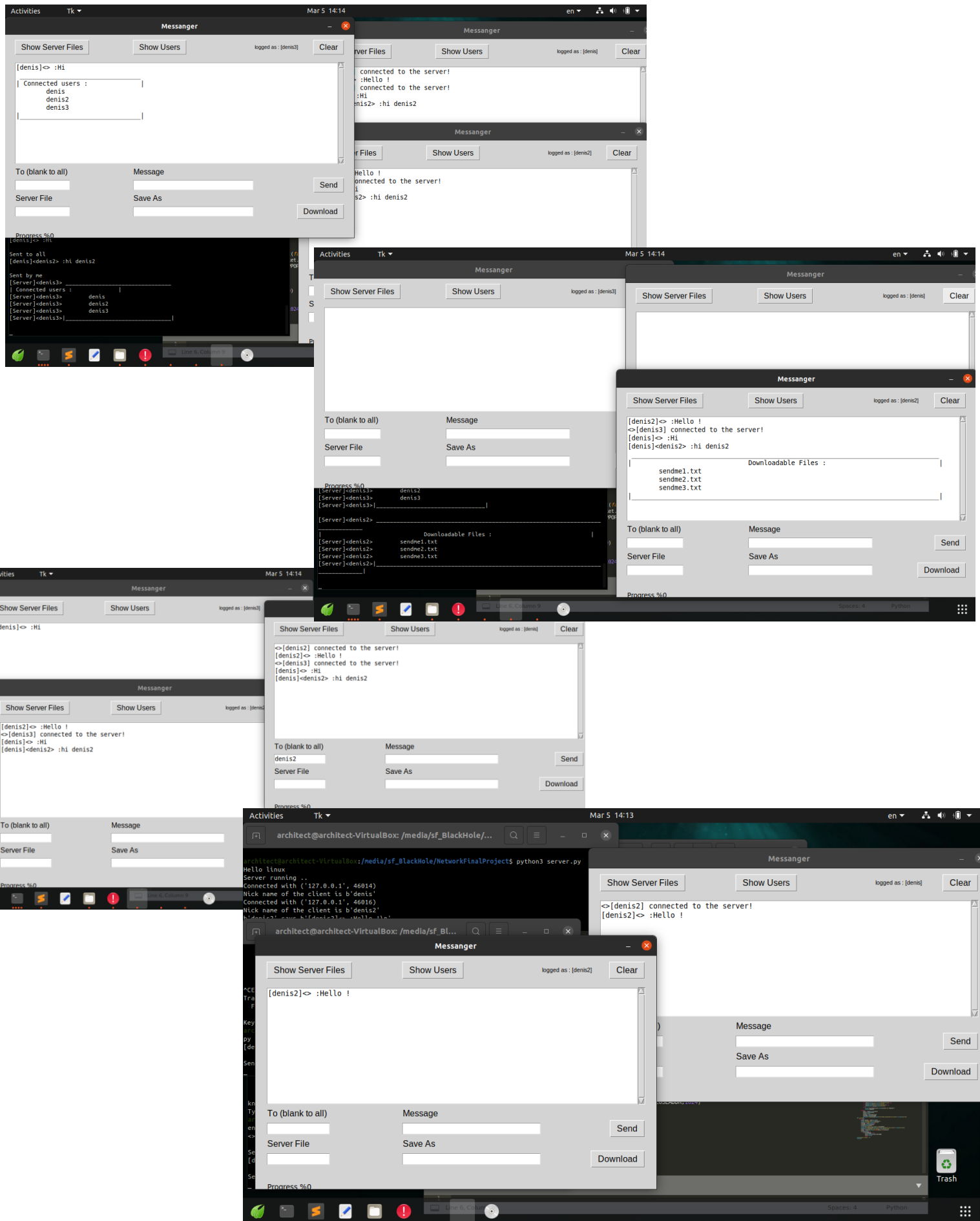


ישנה אפשרות לבחור שם משתמש  
נשלחת הודעה גם בכניסה וגם בעזיבה לחברי הצ'אט



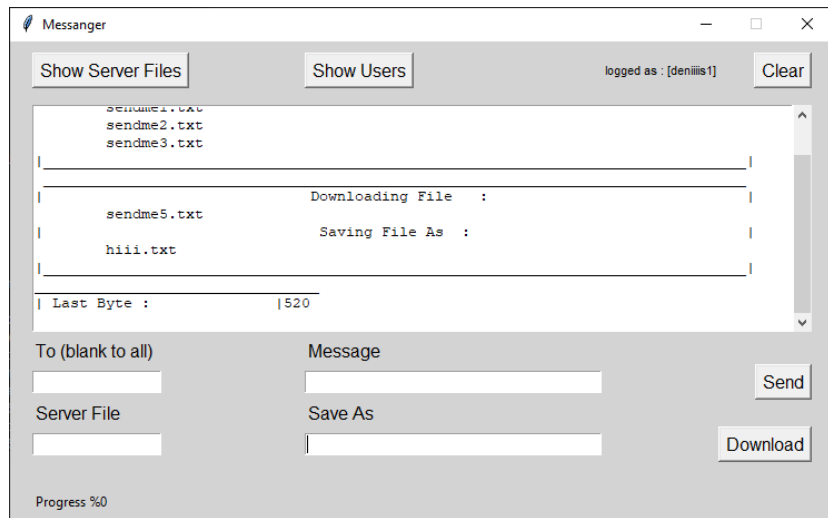
אפשר לשלוח הודעות פרטיות למשתמשים שמחוברים לקבוצה ורק אתם והם תוכלו לראות את ההודעות, בנוסף לראות אנשים שמחוברים עכשיו ובנוסף אפשר לראות קבצים להורדה, כל ההודעות צד שרת לקוח (ולא לקוח לקוח) מופיעות אך ורק אצל אותם המחוברים שמבקשים את השירות



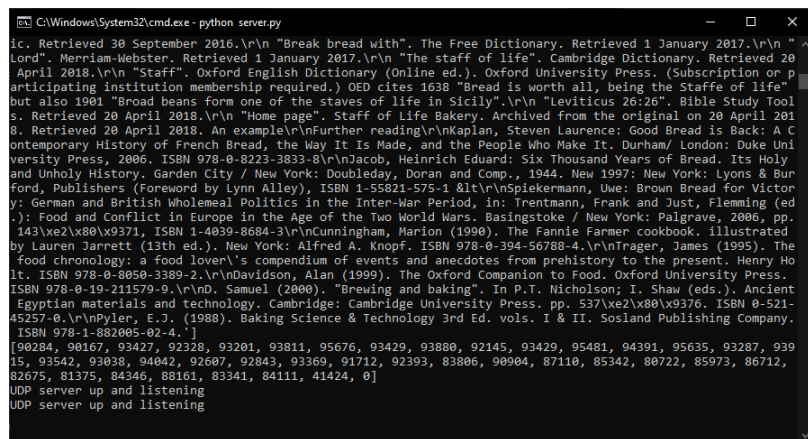


כמו כן אפשר לנקות את הצ'אט של המשתמש שמבקש לעשות את זה

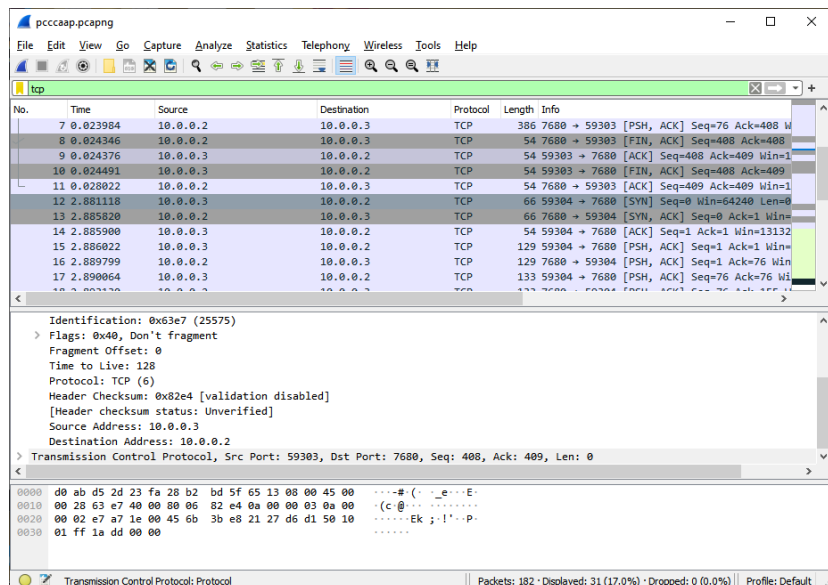
## אפשר לראות את גודל הבייט האחרון שצריך להישלח



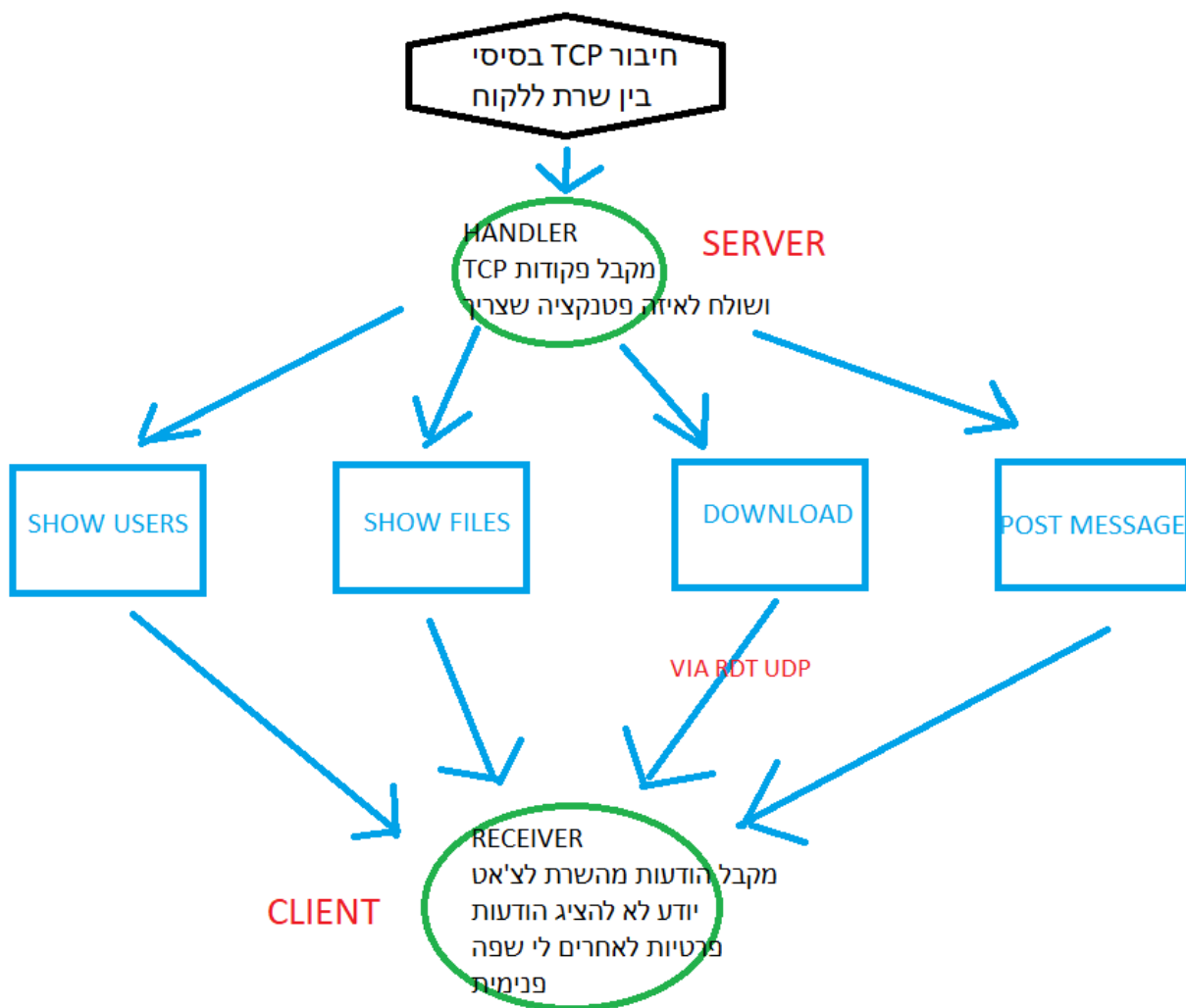
ורואים את רשימת הבייטים שצריכים לשלוח "קובץ טקסט , מחרוזת  
לקראת סוף הטקסט אפשר להבחין ברשימת צ'אקסאם שמיועדת לשליחה ללקוח

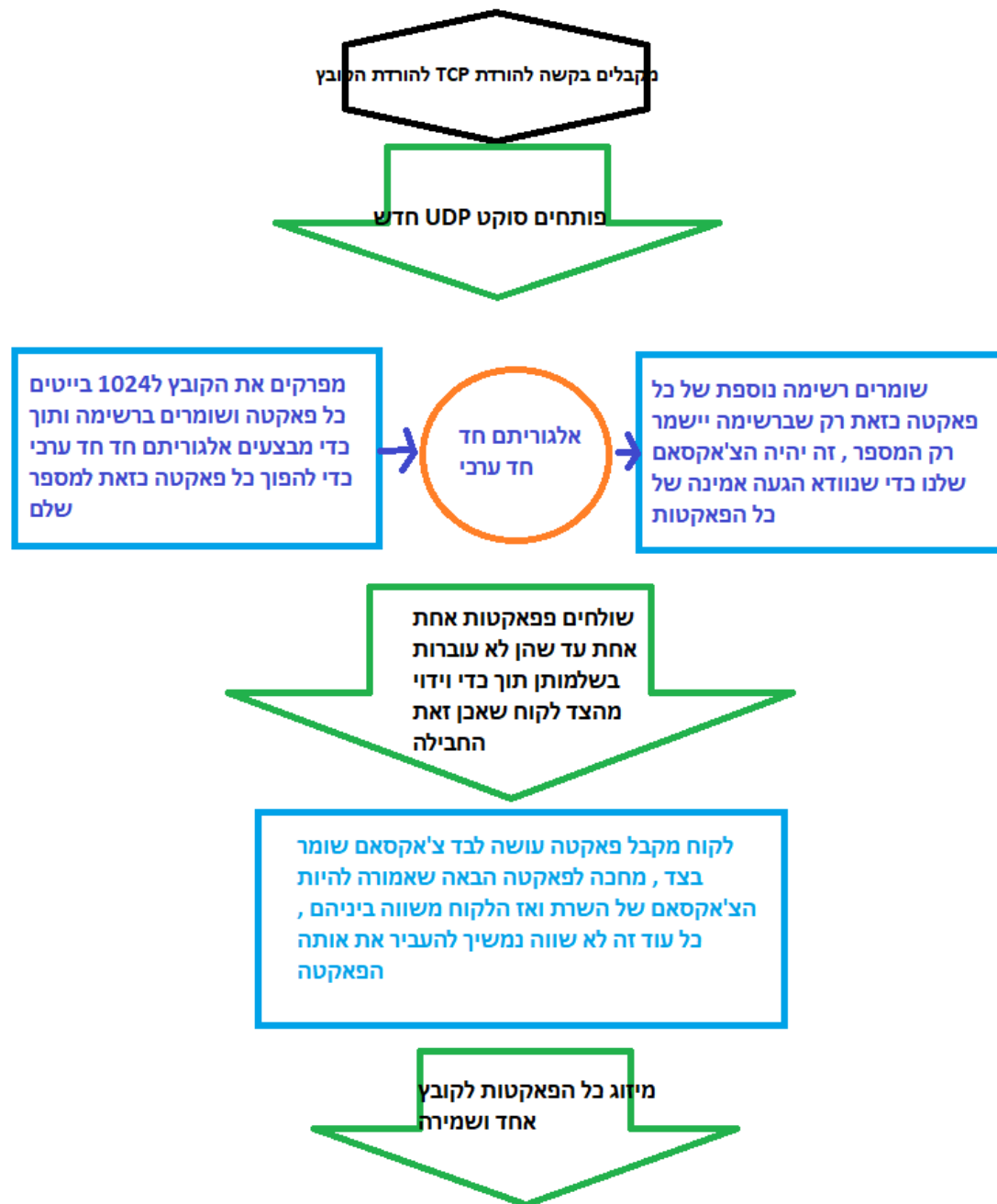


## כמו כן ניתן לראות תעבורת TCP של כניסה לשרת ותכתובת



חלק ב:





התרשים הזה פועל ב-THREAD משלו ולכן בהצטרבות של לקוחות נוספים הם מתפרשים באופן שווה על רוחב הפס הקיים, בנוסף לזה הבקשות של ה-TCP פועלות לפי עקרון CC קיוביק (שהוא הדיפולטיבי)

## חלק ג:

1. נניח שיש רק מחשב אחד ברשת הפנימית שלו

טבלת NAT

אחרי שהראוטר מקבל כתובת MAC של המכונה (LAN)

- סוג הודעה TCP כתובת מקור 10.0.0.3 כתובת יעד 130.57.299.8:55001

SA = 130.57.299.8

מודם

- 10.0.0.0\29

אינטרנט

הדלקת תוכנת ההודעות ללא הורדת קבצים

- TCP כתובת מקור 130.57.299.8:55001 כתובת יעד 56.13.25.15:55001  
(בהנחה שהתוכנה נמצאת על שרת לא ברשת הפנימית שלי)

שימוש בהורדת קבצים

- UDP כתובת מקור 130.57.299.8:55001 כתובת יעד 56.13.25.15:55001

2. CRC זה אלגוריתם שמוצא שגיאות בהעברת נתונים, CRC יהיה שארית החלוקה אשר נצמד לצד הימני של הביטים שאמורים לעבור, לדוגמא נמצא את CRC של 100100 בחלוקה ב1101

נחפש את האורך של המחלק (1101) הוא 4

נוסיף אורך מינוס 1 ביטים לצד הימני של הביטים שאנחנו מעבירים

100100000, כעת נבצע חלוקה והשארית תהיה CRC

1101/100100000 אחרי חילוק בינארי נקבל שארית 001 זהו CRC כעת נוסיף אותו לביטים המקוריים 100100001 ואת זה נשלח

3. http1.0 ההבדל העיקרי הוא בצורה בא הפרוטוקול מאפשר לדבר עם השרת בגרסא 1.0 ניתן להעביר בכל פעם בקשה אחת כלומר הלקוח יבקש דף HTML השרת יחזיר לו את מה שהוא מבקש, אחרי זה הלקוח יבקש תמונה שהוא צריך בשביל HTML השרת יחזיר לו אותה וכן הלאה ..

http1.1 בגרסא הזאת ניתן לבקש כמות של דברים אחרי לחיצת היד, הלקוח יבקש דף HTML

השרת יחזיר לו דף HTML, אחרי זה הלקוח יבקש את כל התמונות שהוא צריך עבור הדף HTML הזה והשרת יוכל להחזיר לו אותם אחד אחרי השני מבלי שהלקוח יצטרך לבקש אותם כל הזמן

http2.0 זה כבר פרוטוקול פחות תלתי, הוא יכול להבין מה הלקוח צריך ולשלוח לו את המידע מבלי שהוא יבקש, הלקוח יבקש דף HTML והשרת יחזיר לו דף HTML בנוסף לכל התמונות שהוא צריך עבור הקובץ הזה

http3.0 QUIC פיתוח של גוגל, בעוד שהפרוטוקולים הקודמים מתבססים על TCP לחיצות ידיים, quic עובד עם פרוטוקול UDP בנוסף גם עובד עם TCP, ומכיל גם לחיצות ידיים יותר בטוחות (הצפנה)

4. בעיקרון בהנחה שיש רק שני מחשבים בעולם שמעבירים מידע מסוג אחד בלבד לא צריך פורטים אבל בגלל שאנחנו משתמשים בהמון מוצרים מעל הרשת והמון מידע מידע מכל מיני סוגים בכל מיני

פרוטוקולים אנחנו חייבים דרך להבדיל בין סוגים שונים של תעבורה וזה גם מגן לנו אבטחתית על המכשיר והמידע שאנחנו מקבלים\שולחים , לדוגמא פורט 80 בדרך כלל מקבל HTTP ופורט 443 מקבל HTTPS , בלינוקס פורטים מתחת ל1000 הם בעלי הרשאת מנהל ולכן צריך להשתמש בפקודת SUDO כדי לקבל\לשלוח מידע בפורט הזה

5. Subnet היא תת רשת , רשת בתוך הרשת, היא מייעלת את הרשת , באמצעותה ניתן להקטין את המרחק בה התעבורה מטיילת ברשת מבלי להגיע לראוטרם מיותרים כדי להגיע ליעד , אפשר להסתכל על דוגמא פשוטה שבה לחברה גדולה יש 200 מחשבים שכולם מחוברים לאותה התת רשת , כאשר מחשב כלשהוא בעולם מחפש כתובת ושואל ראוטרם בדרך איפה ולכן ללכת הוא יפנה לIP ברירת מחדל אשר יחזיר לו תשובה מהירה או שיגיד לו שלא יחפש כאן בכלל וכך המחשב הזה חסר את הטרחה בלעבור ראוטר ראוטר בחברה

6. הראוטר יודע לאן לשלוח את הפאקטות הנכנסות בעזרת כתובת הMAC ולא בעזרת מידע UDP\TCP

כאשר TCP\UDP לא יודע לזהות מכונות, אנו צריכים לזהות חיבור ברמת הרשת , כתובת הMAC מזהה את מתאם הLAN בשכבה מספר 2 (מלמטה למעלה) לעומת כתובת IP שרק עובד בשכבת הIP שכבה 3

7. נתחיל מהכי יוצא דופן NAT , זהו שירות החלפת IP שראוטרם משתמשים בו המטרה שלו היא לתרגם קבוצה של כתובות IP ולקבוצה אחרת של כתובות IP , והסיבה לזה היא כדי לשמור על החד חד ערכיות של המספר המוגבל של כתובות IP הציבוריים אשר קיימים בעולם בערך ארבעה מליארד כתובות IP שזמינות בגרסא 4 של IP ,

SWITCH הוא מכשיר בעל כמה פורטים אשר מקבל חיבור ETHERNET ממכשירים ברשת , הסוויץ' הוא מכשיר אינטליגנטי והוא יכול ללמוד את הכתובות הפיזיות למכשירים שהוא מחובר אליהם ושומר את הכתובות MAC אצלו בטבלה, כשפאקטה מגיעה לסוויץ' היא מיועדת אך ורק לפורט היעד

Router הוא מכשיר שמעביר מידע מרשת אחת לשנייה בהתבסס על כתובת הIP שלהם , כשפקטת מידע מגיעה לראוטר , הראוטר בודק את כתובת הIP של הפאקטה ובודק האם היא מיועדת לרשת שלו או לרשת אחרת , כשהפקטה לא מיועדת לרשת של הראוטר הוא שולח אותה לרשת אחרת

8. יש לנו פיתרון של להשתמש בכתובות IPv6 אשר יכול להכיל כמות הרבה יותר גדולה של כתובות עוד פיתרון זה לשחרר ולמחזר כתובות IPv4 ראינו איך הNAT מקצה כתובות למכשירים הוא לוקח כתובות פנויות שסביר להניח שמישהו אחר השתמש בהם בעבר השירות הזה יכול גם להקצות לחברות גדולות כתובות IP פרטיות

פיתרון נוסף זה CIDR שאומר לחלק את כתובות הIP ל5 מחלקות A E וקח לייצר עדיפויות וחוסר עדיפויות לכתובות IP ברשת

9.

e. RIP כי הדרך היחידה שלו להגיע לא היא דרך 4AS אשר משתמש בRIP

f. RIP כמו סעיף קודם כי התת עץ שהוא מקבל מOSPF זה בעצם קודקוד A3 שמחובר לB3 שמחובר לC3 שמחובר לC4 וכן הלאה , לא קיבלנו פה עץ קיבלנו פה רשימת קפיצות

g. RIP אותו דבר אותה סיבה

h. OSPF כיוון שיש לו כמה אפשרויות להגיע 2AS משתמש בOSPF ולכן הוא בונה מסלול הכי קצר