**-מגישים : ליאל ביטון 315021923 , חי לוי 313589038**

**DNS over HTTPS - (DoH)**

* **הציגו יתרון אחד לשימוש ב-DoH והסבירו אותו:**

יש המון מידע בשאילתות DNS ובגלל שהפרוטוקול לא מוצפן, הן יכולות לראות את התעבורה,

את המידע הזה לאחר מכן הן יכולות למכור לצד ג'.

לכן שירות ה-DoH יודע את שאילתת ה-DNS ומצפין אותה וככה מונע את שוק המכירת המידע "החופשי" שהיה עד עכשיו לצד ג'.

* **הציגו והסבירו על שני חסרונות לשימוש בשיטת DoH לעומת DNS הרגיל.**

**⦁** בקשת ה-DoH היא מוצפנת ואינה נראית לעיני צד ג' ולכן היא מעכבת תוכנות אבטחת סייבר המסתמכות על פיקוח DNS כדי לחסום בקשות לתחומים זדוניים ידועים.

לכן תולעת DDoS 2019 Godlua השתמשה ב- DoH כדי להסוות חיבורים לשרת הפקודה והשליטה שלה. כלומר נוצרה לנו פרצת אבטחה חדשה בגלל ה-DoH שחברות אבטחת סייבר ממשלתיות שמנסות להגן על אזרחיה בעזרת ה-DNS הרגיל, לא יכולות להגן בגלל השירות החדש "אבטחה" החדש הזה - DoH.

⦁ כיום ברשת הביתית קיימת חסימת אינטרנט אשר מגבילה את הגישה אל אתרים אשר הפרו בעיקר זכויות יוצרים, או גישה לספק שירותי תוכן למבוגרים חסימה זאת מתבצעת בדרך כלל על-ידי DNS הרגיל. במשך הרבה שנים היה מאבק להלחם במניעת הורדות פיראטיות של מוזיקה ותכנים נוספים שפוגעים בפרנסתם של אנשי תרבות ויצירה. לכן שימוש ב-DoH יכול לעקוף את חסימת אתרי האינטרנט האלו וכל מי שמשתמש בשירות ה-DoH יכול לגשת למידע שהיה חסום ולפגוע בזכויות יוצרים ולהיפגע מרשת לא מאובטחת (ולא תמיד מודע לזה שיכול להיפגע).

* **בחרו אחד מהחסרונות משאלה (2), הציעו דרך למתן\לעקוף\לפתור חיסרון זה והסבירו אותה.**

אנחנו מציעים למתן את החיסרון הראשון שהצגנו (תולעת ה-Godlua) באופן הבא:

להקים ארגון בינלאומי שיהיה אחראי על הנושא הזה.

כלומר כל מדינה החתומה בארגון תציע את רמת האבטחה שלה לאזרחיה, על כל מדינה למנות גוף אמין, מרכזי, יחידי ומוכר שמתעסק באבטחת סייבר לאומית ולה תיהיה גישה לשימוש בשירות ה-DoH מבחינת יכולת קריאת השאילתות וזאת אך ורק למטרת הגנה מתקיפות סייבר בלבד. וכך אפשר למנוע את ההתעכבות הרבה שהגבילה את אבטחת הסייבר נגד תקיפת ה-DDOS ב-2019 ולהפיק לקח יקר לקראת התקיפה הבאה.

* **ישנן 4 דרכים בהן ניתן לשלב את שיטת ה-DoH באינטרנט שלנו:**

**⦁ מימוש DoH ברמת האפליקציות (לדוגמא: לעדכן את קוד הדפדפן כך שישלחו שאילתות דרך HTTPS).**

**⦁ מימוש DoH ברמת שרת proxy\* ברשת (מהמחשב לשרת נשלח לפורט 53 והלאה, כבר 443).**

**⦁ מימוש DoH ברמת שרת proxy מקומי (על המכונה רץ שרת proxy)**

**⦁ התקנת plugin המממש DoH ברמת הגדרות המחשב .**

**כתבו השוואה בין כל ארבעת השיטות, בהשוואתכם הראו יתרונות וחסרונות לכל שיטה והציגו מהי, לדעתכם, השיטה המועדפת מבין הארבעה.**

⦁ ההבדל בין a ל-b הוא שב-a המימוש עובד תחת הגדרות הדפדפן בלבד והשאילתות שלך בטוחות מספק האינטרנט שלך דרך הדפדפן ועבור b דיי מזכיר את a (כלומר בעת גישה לדפי אינטרנט בדפדפן האינטרנט שלך) אומנם לרוב, שרתים פרטיים אלה מוצעים על ידי הספקים המציעים לך שרתים ללא עלות אשר אין אמון עליהם בקלות. אם אינך משלם עבור שירותים בכסף, ייתכן שתשלם בדרך אחרת - כמו נתונים פרטיים משלך גם אם פועל DoH- כלומר פגיעה בפרטיות לכן a לדעתי עדיף על b במקרה זה.

⦁ ההבדל בין a ל-c הוא שב-a המימוש עובד תחת הגדרות הדפדפן בלבד, עם זאת, בעוד שהשאילתות שלך בטוחות מספק האינטרנט שלך, עדיין ספקי DoH יכולים לעקוב אחר מי שמחפש אילו תחומים - לא משנה עד כמה הם נוקשים לגבי פרטיות. ולכן ב-c שרת proxy שולח בקשות לרשימת ספקים שנבחרה באמצעות שיטות כמו למשל round-robin-selected, ומסווה ביעילות את התעבורה שלך מכל ספק זר לכן c מבטיחה יותר הגנה.

⦁ ההבדל בין a ל-d: עובד תחת הדפדפן בלבד בעוד ש-d מדבר רק על כל האפליקציות שנמצאות אצלנו במחשב כמו skype,zoom או כל תוכנה תקשורתית אחרת שנמצאת במחשב שלנו.

עבור ההבדלים האלו כל מקרה לגופו, a לא מכסה את d וגם להפך.

⦁ ההבדל בין b לבין c:עבור b שרתים פרטיים אלה מוצעים על ידי הספקים המציעים לך שרתים ללא עלות אשר אין אמון עליהם בקלות בנוסף b פועל על הדפדפן בלבד ולכן אם כבר לממש DoH אז עם c, הוא יותר אמין ומספק "חומת אש" לא רק עבור הדפדפן אלא עבור כל השער בינך לבין האינטרנט כלומר הוא מתרגם כל שאילתת DNS שיוצאת מהמחשב ל-HTTPS .

⦁ ההבדל בין b לבין d: d ממש DoH עבור אפליקציות במחשב כמו skype בעוד ש-b מספק DoH רק עבור הדפדפן, כלומר כל אחד מהם מכסה את ה-DoH עבור הגוף שלו לכן הם לא כל-כך ברי השוואה.

⦁ ההבדל בין c לבין d: d ממש DoH עבור אפליקציות במחשב בעוד ש-c הוא שרת פרוקסי מקומי שמתרגם כל שאילתת DNS שיוצאת מהמחשב ל-HTTPS עבור c יכול לכסות את d ולכן מן הסתם עדיף כבר להשתמש ב-c שגם יכול להסתיר את השאילתות עבור ספקי DoH "רעים" בשונה מ-d.

לכן היינו בוחרים את c להיות השיטה המועדפת כי היא יכולה לכסות את a ואת d איתה כלומר אם הוא מתרגם כל שאילתת DNS שיוצאת מהמחשב ל-HTTPS אז מן הסתם הוא מצפין גם עבור דפדפן וגם עבור אפליקציה במחשב וכמובן שיכול לתת יותר אמינות לעומת b שכן יכול לפגוע לנו בפרטיות ולמכור מידע אישי שלנו לספקי DoH רעים.

* **נניח שאנו ברשת שקיים בה איבוד פקטות (packet loss) באחוז לא ידוע ואנו רוצים לטעון דף שצריך 25 שאילתות כדי לבקש את כל המשאבים שבו. הציגו יתרון ברור שיש ל-DoH לעומת Do53. (רמז: מנגנון הקיים ב-TCP)**

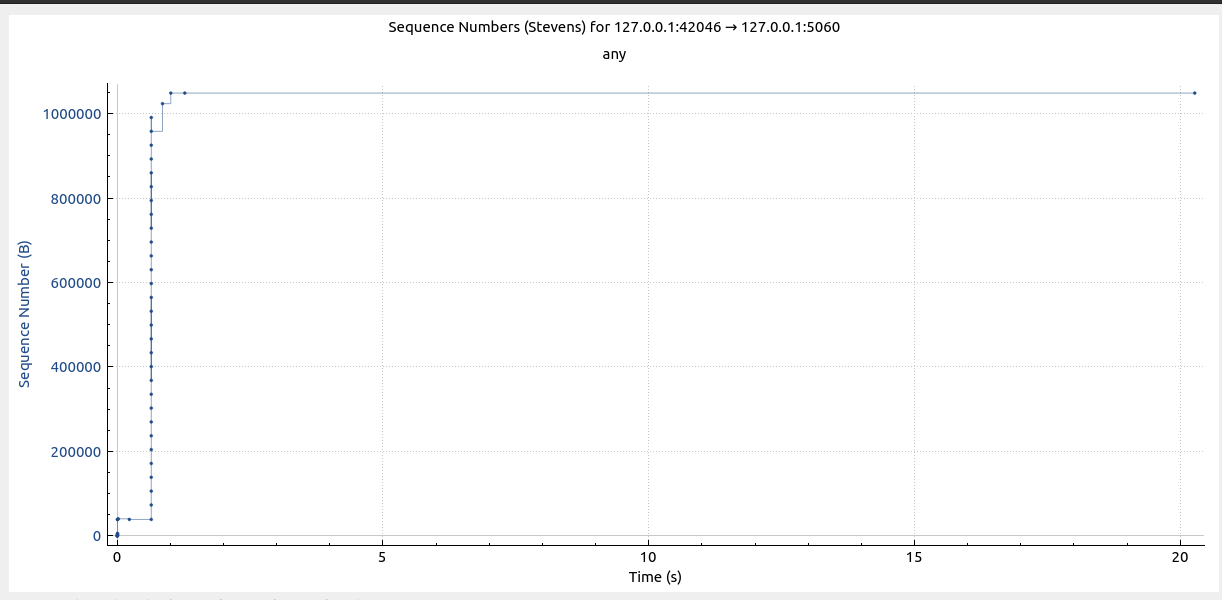
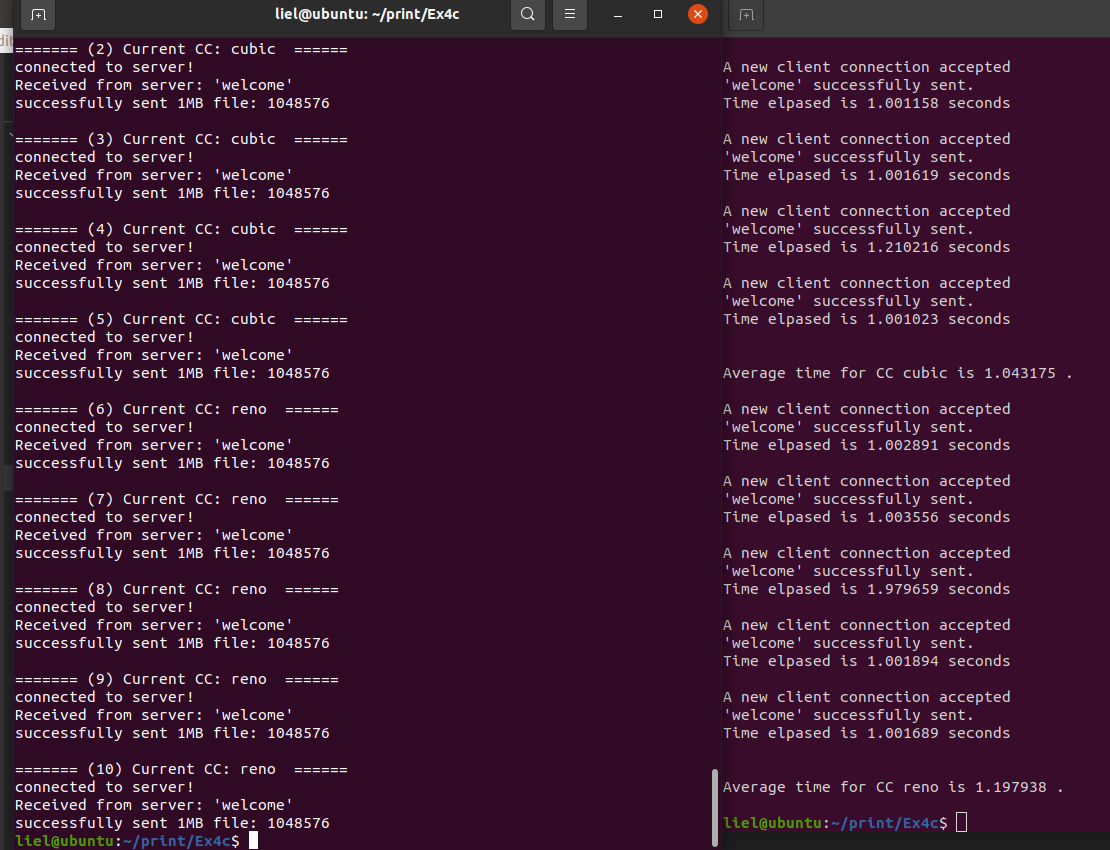
תעבורת ה-DNS הרגילה (Do53) כפי שנלמד בהרצאה משתמש בפרוטוקול ה-UDP.

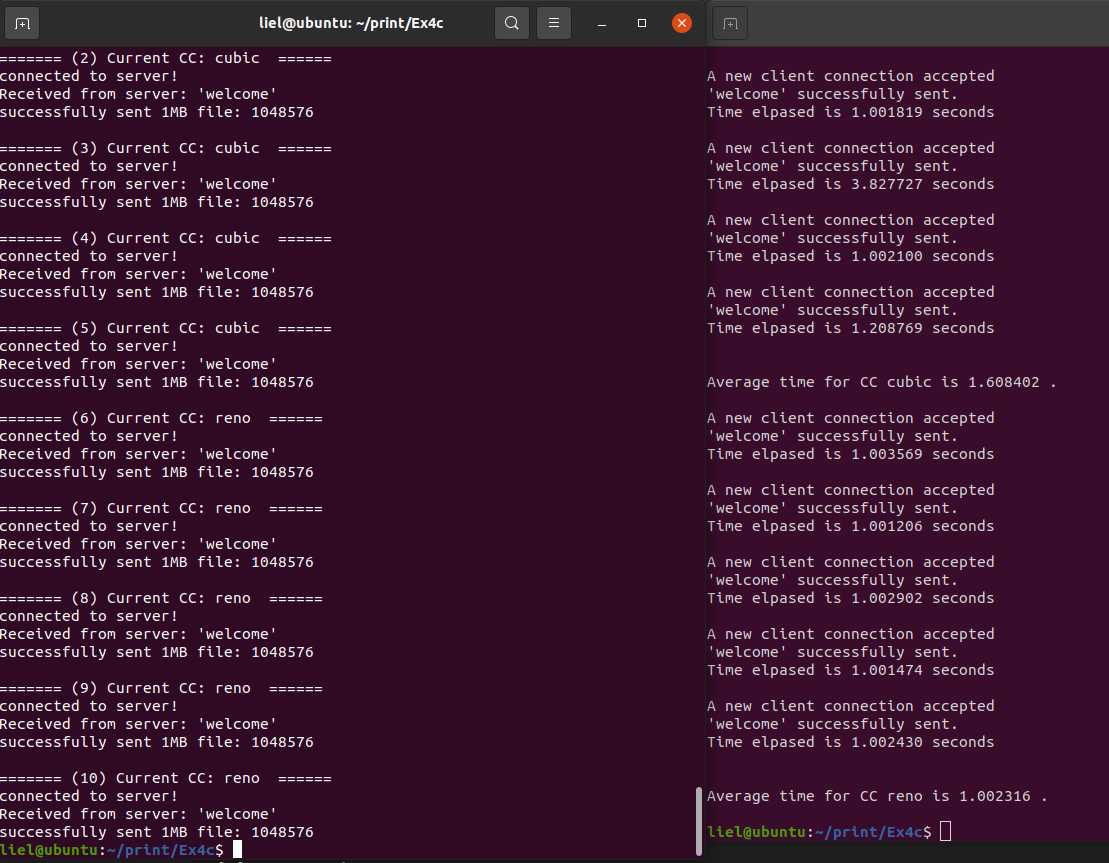
היתרון המובהק ששאילתות DoH שאבדו מסתמכות עליהן מדיניות השידור מחדש של פרוטוקול TCP הבסיסי, ולא טיימר קבוע. DoH בהשוואה ל-Do53 לגבי הבעיה המוצגת בשאלה היא שמאחר וה-DoH עובד עם TCP אז מופעל בו מנגנון TCP Fast Retransmit, לפיכך DoH יוכלו לשחזר במהירות רבה יותר שאילתות DNS שאבדו בטעינת הדף מאשר Do53 שאין בו את האפשרות הזאת.

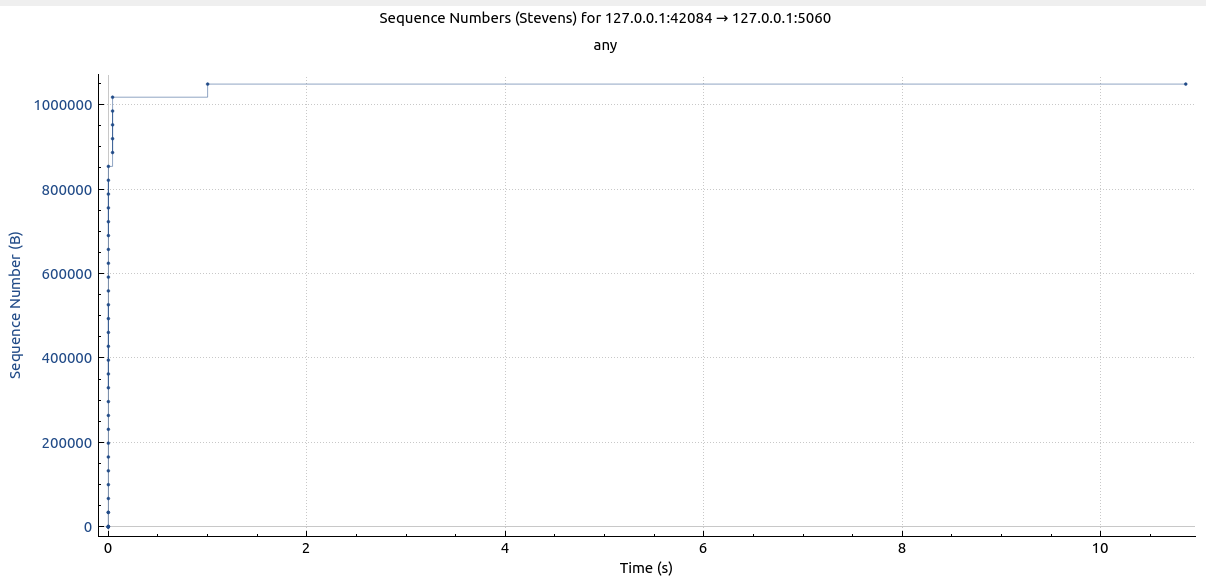
| **Packet Loss** | **Reno Algorithm** | **Cubic Algorithm** |
| --- | --- | --- |
| 10%-1mb | 1.197938 | 1.043175 |
| 15%-1mb | 1.002316 | 1.608402 |
| 20%-1mb | 2.316772 | 4.734238 |
| 25%-1mb | 13.800836 | 23.008833 |
| 30%-0.5mb | 4.955840 | 2.326557 |

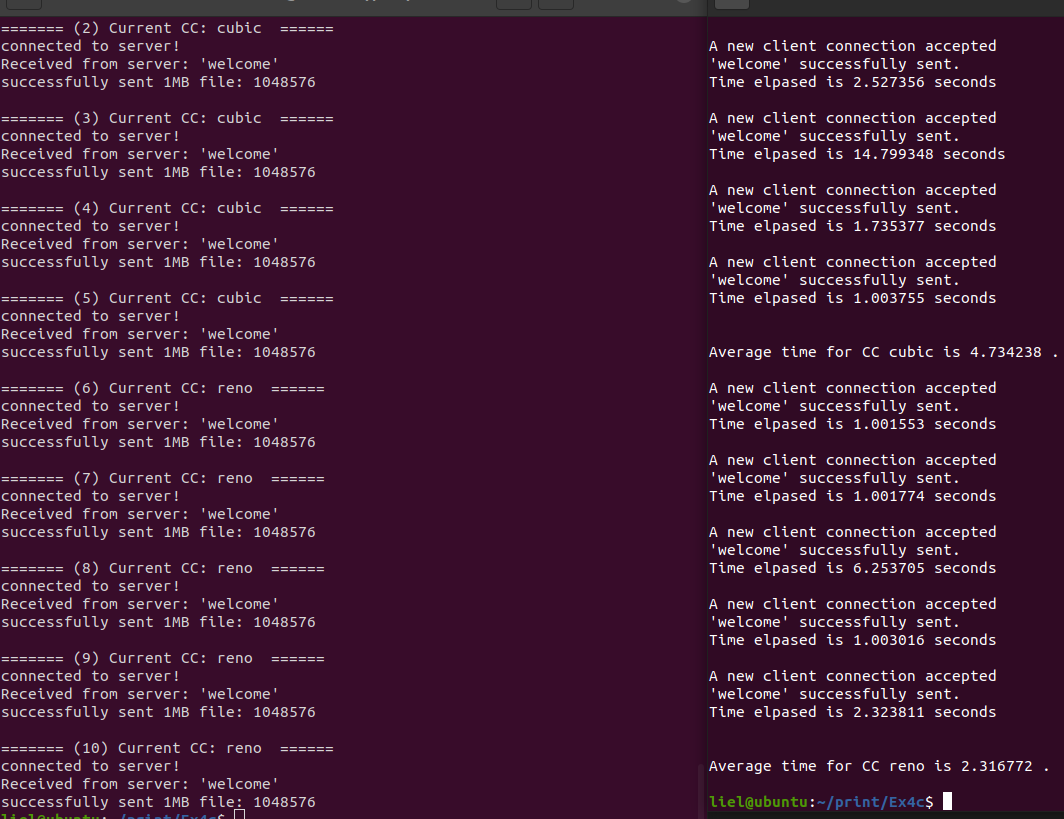
Reno uses **packet loss to detect network congestion** [1]. TCP-BIC. ... CUBIC spends a lot of time at a plateau between the concave and convex growth region which allows the network to stabilize before CUBIC begins looking for more bandwidth

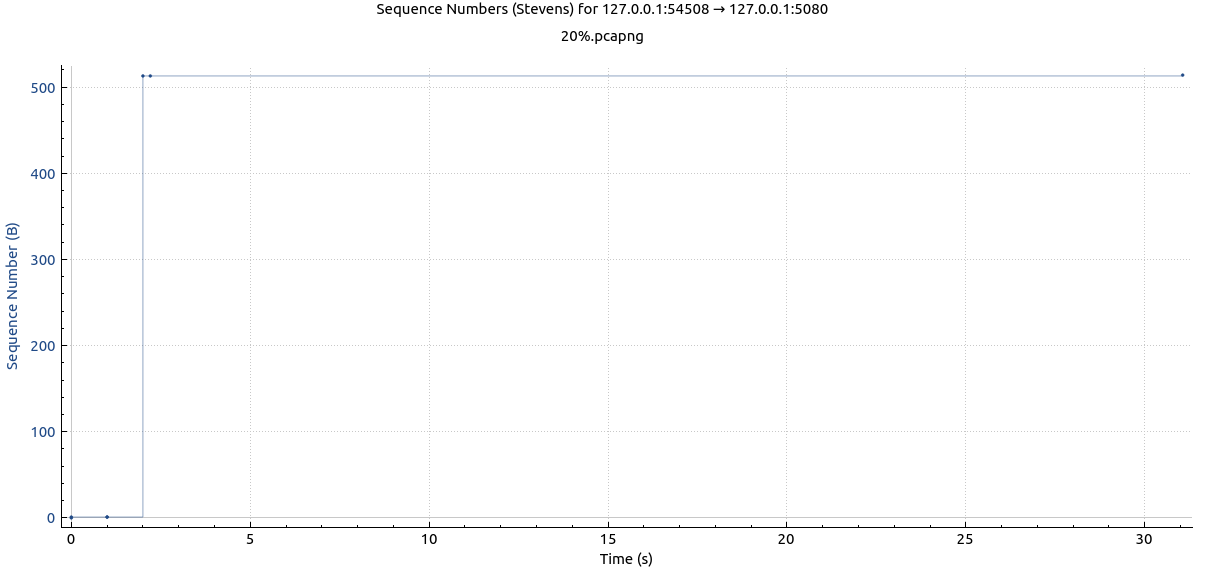
losing 10%-1mb



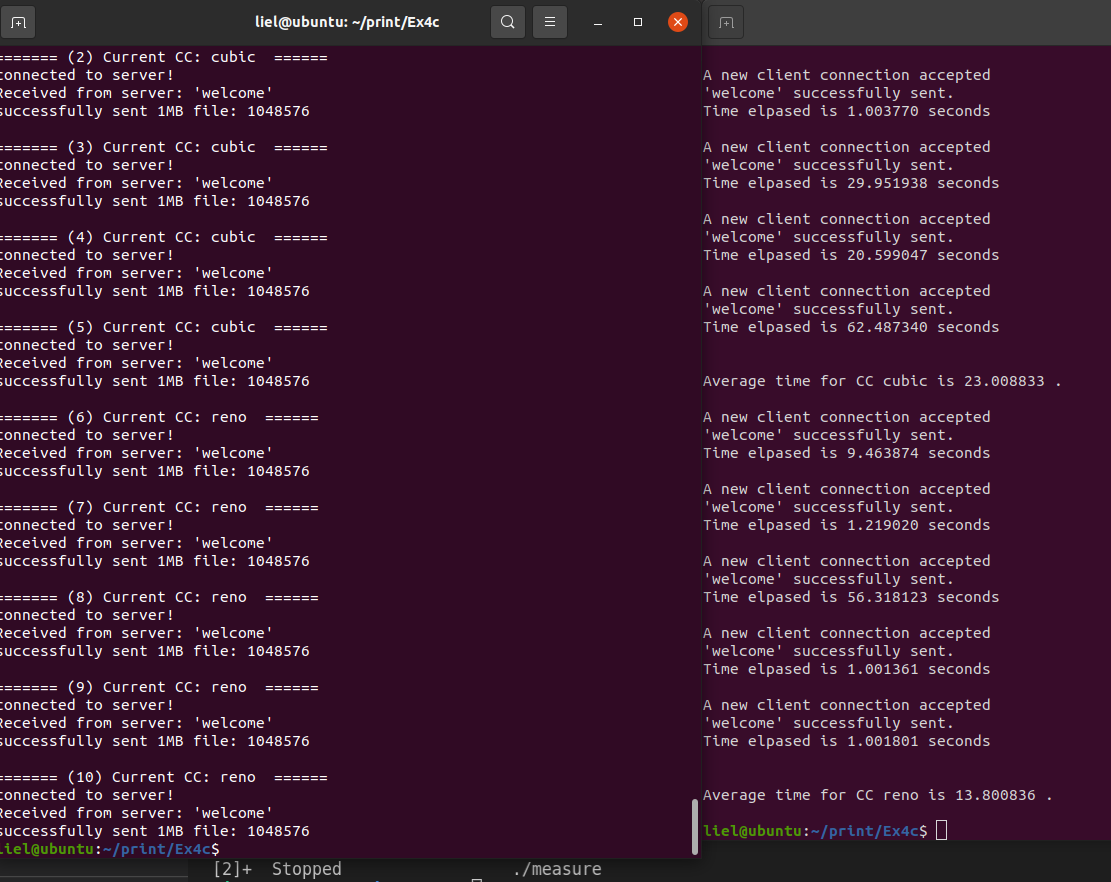
losing 15%-1mb

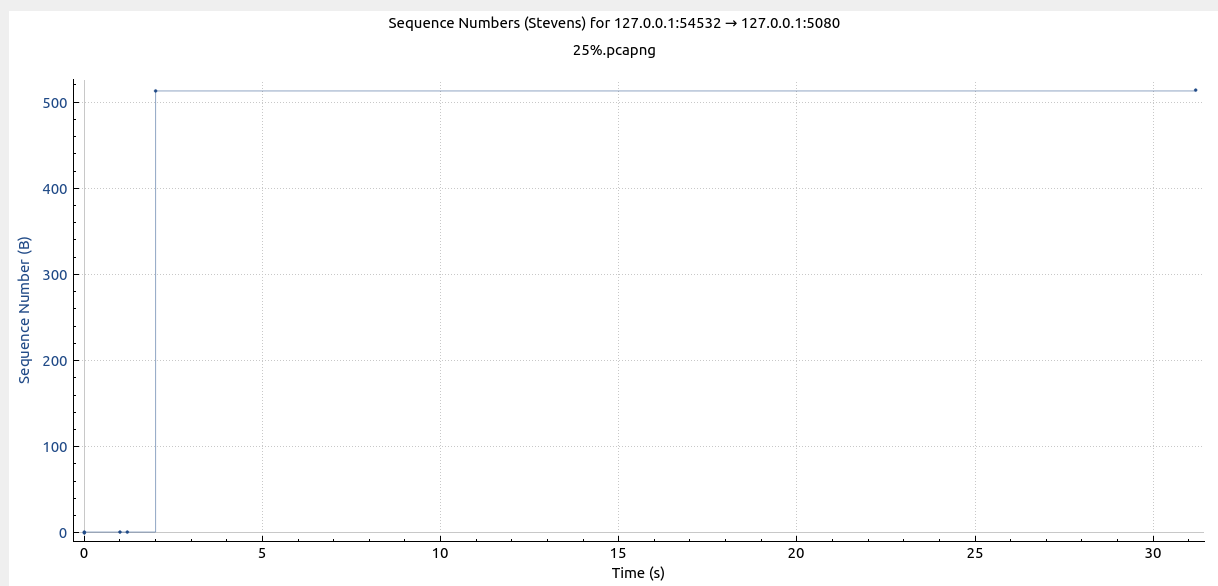


losing 20%-1mb



losing 25%-1mb





losining 30% - 0.5mb

