**aתקשורת ומחשוב – תשפ"ב - סמס' א' - מטלה רביעי**

**יוסף סוקולוב וסלי שרפמן**

את מטלה זו יש להגיש בזוגות כקובץ ZIP עם מספרי ת"ז של הסטודנטים/ות. לתיבת ההגשה במודל. הגשות באיחור יתאפשרו עד 4 ימים כאשר לכל יום איחור ירדו 5 נקודות. שימו לב, יש להגיש קובץ Pdf המכיל צילומי מסך (בכל מקום שאתם עושים משהו תצלמו מסך ותסבירו איך הגעתם למסקנה). **אין להגיש צילומי מסך של מרוכזים בקובץ אחד עם הסברים, קובץ ללא הסברים או צילומי מסך בלבד לא יבדקו**

1. את המטלה יש להגיש עד התאריך המצויין בתיבת ההגשה
2. כל הקבצי המטלה (קוד, פלט תעבורה, הסבר) כולל הסברים שלכם והקלטות **wireshark** דחוסים לקובץ ZIP ששמו הוא מס' ת.ז. של המגישים עם קו תחתון בינהם ID\_ID.
3. הגשה בזוגות אפשרית.
4. מותר לכם להשתמש בכל החומר שנמצא במודל כולל קוד בתרגולים. חומרים אחרים אין אפשרות. כמובן שאפשר להעזר באינטרנט להבנה של תהליכים וקוד אבל בשום פנים ואופן לא להעתיק קוד
5. אין איחורים ללא אישור מיוחד של רכז הקורס (עמית), איחור ללא אישור יגרור אפס אוטומטי
6. הגשת העבודות תתבצע דרך מערכת ה Moodle של הקורס (לא דרך האימייל).
7. יש להקפיד על כללי עיצוב הקוד שנלמדו בתואר (נא להקפיד על פלט ברור, הערות קוד במידה ושמות משתנים בעלי משמעות). קוד רץ בלבד יכול לקבל לכל היותר ציון 60, שאר 40 הנקודות זה הסברים שלכם, ידע, קוד קריא וכו.
8. ניתן להגיש תרגילים למערכת מספר בלתי מוגבל של פעמים כאשר כל הגשה דורסת את הקודמת.
9. העבודה הינה אישית של הזוג ואסור לקבל עזרה מאנשים מחוץ לאוניברסיטה או בתוכה לה. אנשים המתקשים ורוצים עזרה יכולים לפנות לצוות הקורס בשעות הקבלה או להעלות שאלה לאתר הקורס.
10. אסור להעביר קטעי קוד בין סטודנטים, להעלות פתרונות או חלקי פתרונות לאתרים ברשת האינטרנט, פורומים או בקבוצות תקשורת שונות.
11. סטודנטים שיעתיקו פתרון, יקבלו 0 בכל המטלות בקורס ונעלה דיווח לוועדת המשמעת המוסדית.

במטלה זו תעמיקו ב-DNS וב-TCP, תרחיבו על העקרונות שלהם ומימושם. כמו כן, תשתמשו בסוקטים למימוש תוכנת מחקר משלכם שתחקור עקרון של TCP (אולי החשוב ביותר שלו), Congestion Control.

**חלק א' - DoH**

DNS over HTTPS

הינה שיטה חדשה יותר המוצעת לשימוש. כמו שמרמז השם, זוהי שיטת תשאול DNS בחיבור HTTPS מאובטח.

1. הציגו יתרון אחד לשימוש ב-DoH והסבירו אותו (כמובן, מעבר לעובדה שהוא מאובטח ומוצפן)

**תשובה:** DoH שולח את החבילה על פי פרוטוקול TCP שמבטיח לנו העברת פאקטות לאיבוד יותר מהיר מאשר היינו משתמשים בפרוטוקול DNS רגיל שמשתמש בUDP בשביל העברה.

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

1. בקשות DNS מוצפנות ולכן קשה לעקוב אחרי תעבורה ה-DNS ולבחון אותה כי קשה להבדיל בין חבילות שהם בעצם בקשות וכאלה שלא (חבילות מוצפנות נשלחות דרך פורט אחר ולא פורט רגיל) ואם נרצה לחסום למשתמש אתרים מסויימים לא נוכל לעשות זאת כי DoH מסתיר את המידע הזאת וזה לא טוב כי אם ההורים ירצו לחסום לילדים שלהם אתרים מסויימים הם לא יוכלו לעשות זאת.

2. מכיוון שפרוטוקול זה די חדש יש צורך להתקין תוכנת צד שלישי ותוספים למערת ההפעלה. המחשב לא פותח אותו כערך מחדל.

1. בחרו אחד מהחסרונות משאלה (2), הציעו דרך למתן\לעקוף\לפתור חיסרון זה והסבירו אותה.  
   ״מכיוון שפרוטוקול זה די חדש יש צורך להתקין תוכנת צד שלישי ותוספים למערת ההפעלה. המחשב לא פותח אותו כערך מחדל.״ פתרון לזה זה להתקין את הפרוטוקול יחד עם גרסה חדשה של דפדפן וכך לכולם יהיה אותו כשהם יורידו דפדפן אם לא היה להם או יעדכנו גרסה.
2. ישנן 4 דרכים בהן ניתן לשלב את שיטת ה-DoH באינטרנט שלנו:
   1. מימוש DoH ברמת האפליקציות (לדוגמא: לעדכן את קוד הדפדפן כך שישלחו שאילתות דרך HTTPS)
   2. מימוש DoH ברמת שרת proxy\* ברשת (מהמחשב לשרת נשלח לפורט 53 והלאה, כבר 443)
   3. מימוש DoH ברמת שרת proxy מקומי (על המכונה רץ שרת proxy)
   4. התקנת plugin המממש DoH ברמת הגדרות המחשב ("מעכשיו, אתה שולח רק DoH")

כתבו השוואה בין כל ארבעת השיטות, בהשוואתכם הראו יתרונות וחסרונות לכל שיטה והציגו מהי, לדעתכם, השיטה המועדפת מבין הארבעה. כלומר, הציגו את השיטה בה, לדעתכם, היתרונות הגדולים ביותר לעומת החסרונות הקטנים ביותר.

|  |  |  |
| --- | --- | --- |
|  | יתרון | חסרון |
| a | ״פיזור״ העדכון ברמת הלקוח, זאת אומרת כל לקוח פותר את הבעיה בעצמו ולא צריך למצוא פיתרון רחב. | עדכון ברמת האפליקציה יגרום לכך שהלקוח יצטרך לעדכן עבור כל דפדפן/ אפליקציה על מנת להשתמש ב DOH במקום לעדכן פעם אחת. |
| b | לא מעמיס על המחשב ולוקח משאבים יקרים ממנו | שרת proxy ברשת הוא שרת מרוחק מבחינה פיזית, החבילות יצטרכו לעבור דרך ולכן במצב הזה המידע פגיע יותר.  למשל במקרה של תקיפת man in the middle, על ידי גילוי של מה היא הכתובת של שרת הproxy המרוחק. וכך אנו מאבדים מיתרון ה DOH מבחינת ההצפנה והאבטחה. |
| c | -שרת ״אישי״ לכל מעונה ולכן מונע עומס למשל שלא כמו על שרת אחד שאחראי על כמה מחשבים במקביל  - יותר בטוח מproxy ברשת כי החבילות לא יוצאות החוצה מהרשת הפנימית ולכן יותר בטוח נגד תקיפות חיצוניות. | מימוש שרת proxy על כל מכונה דורש משאבים רבים מאותה מכונה. |
| d | -מעדכן את המימוש עבור כל האפליקציות/דפדפנים בבת אחת במקום לכל אחד בנפרד.  -מונע בזבוז משאבים שלא כמו הקמת שרת. | כאשר ״נכריח״ את המחשב לשלוח רק DOH המצב יכול לגרום לבעיה בכך שאולי הservers שהם היעד לא תומכים בפרוטוקול זה, ולכן לא יוכלו לתקשר! |

1. 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)

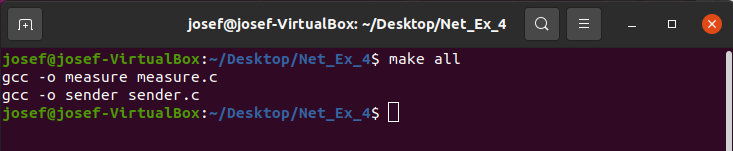
בעזרת מנגנון שקיים ב-TCP, בעזרת fast retrainsmit לא נצטרך לחכות ל-timeout עבור כל פאקטה אלה לפי הASK נוכל לדעת האם החבילה אבדה לפני שמגיעים לtimeout על ידי שמנגנון שולח אישור של החבילה כמה פעמים וכך ניתן לדעת לפני timeout ולחסוך זמן. מנגנון זה לא קיים ב53Do שעובד על פי UDP ולכן DoH ביתרון עליו.

\* שרת פרוקסי(proxy), במילים פשוטות, הינו שרת שאנו בוטחים בו שתפקידו לטפל, במקומנו, באינטראקציות עם שרתים חיצוניים.

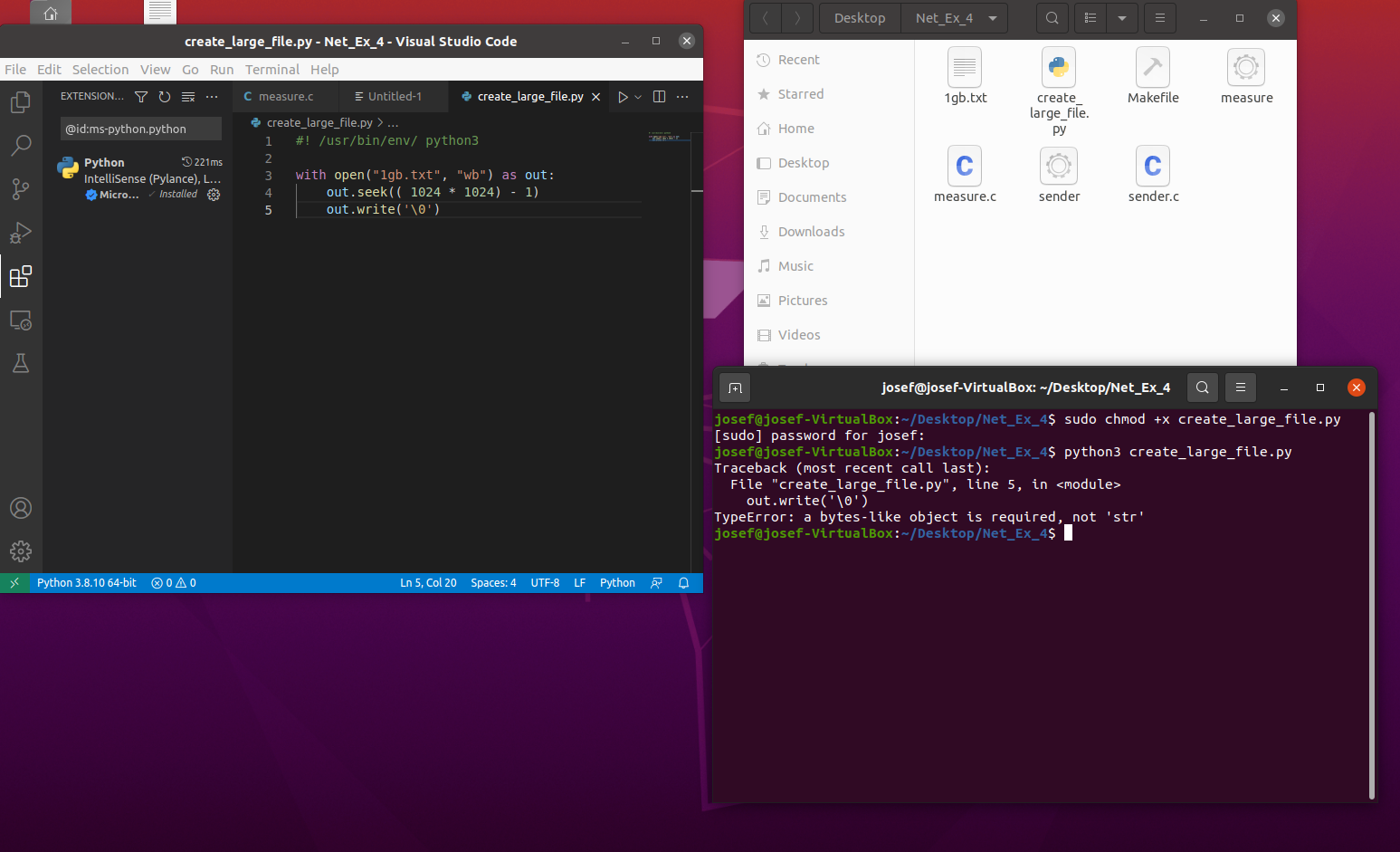
**חלק ב' - Congestion Control**

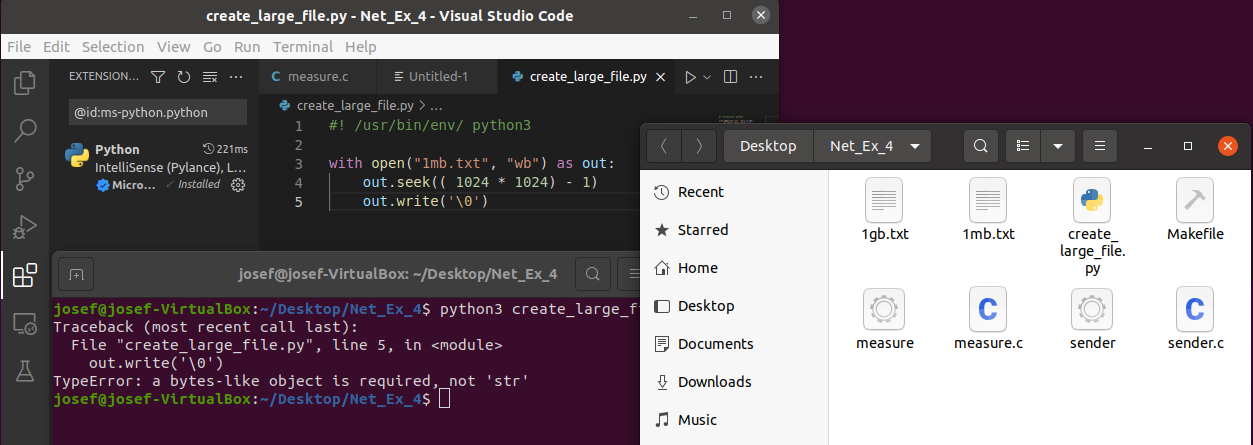
Our screenshots:

Make all:

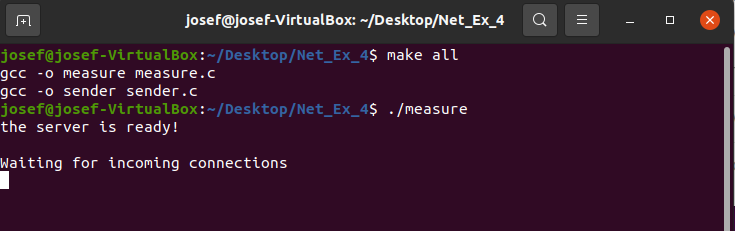
****

Create large file:

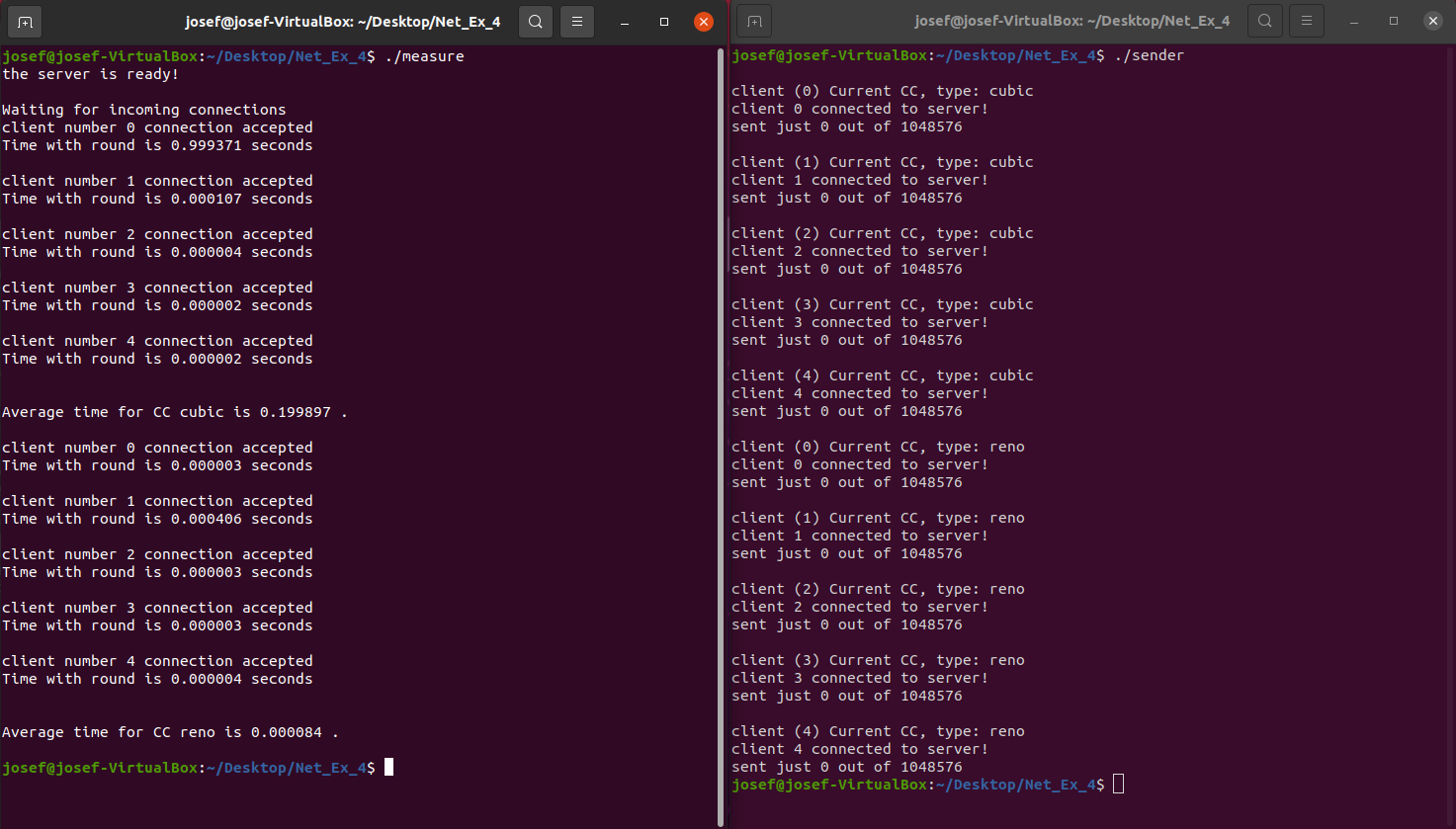




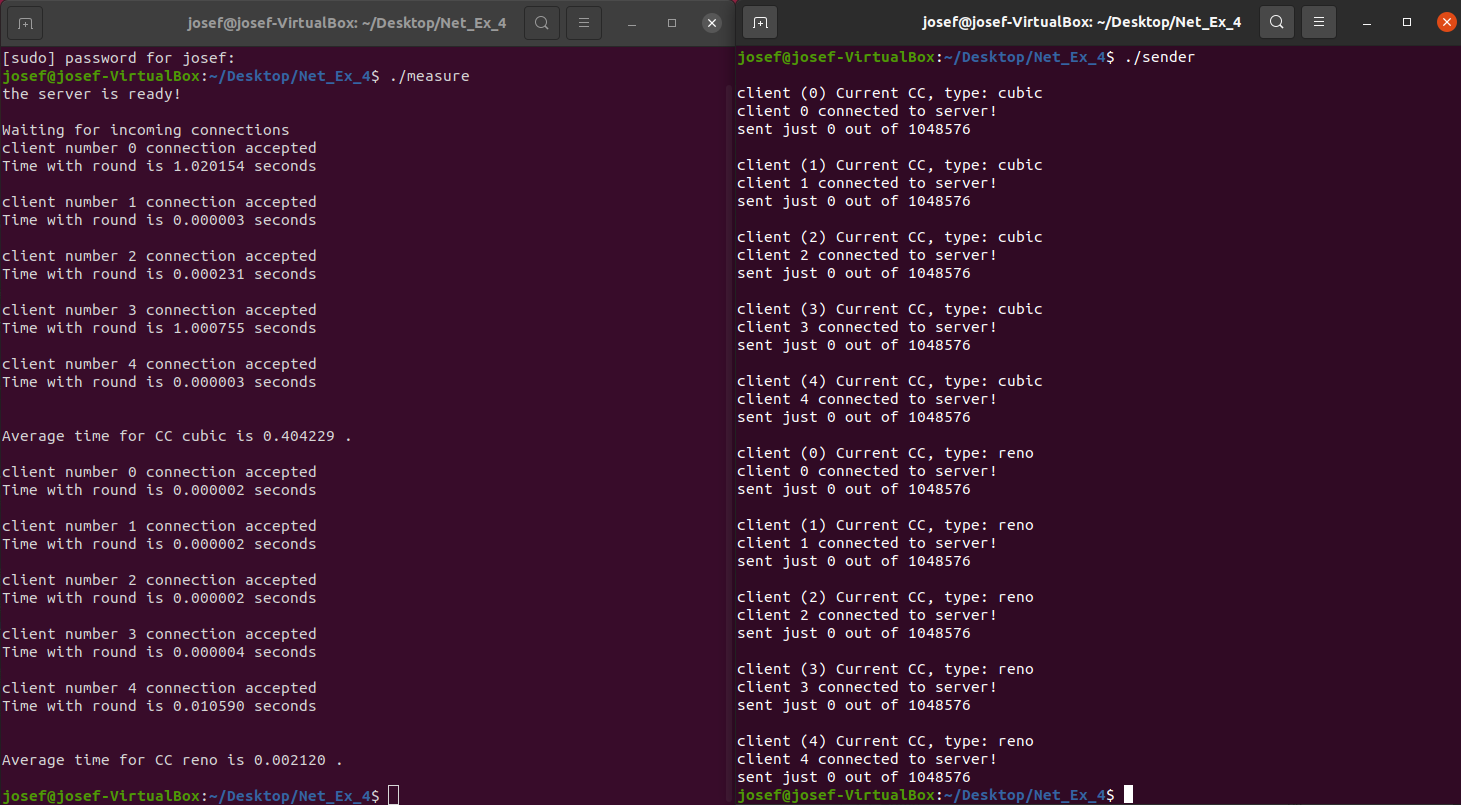
Run measure:



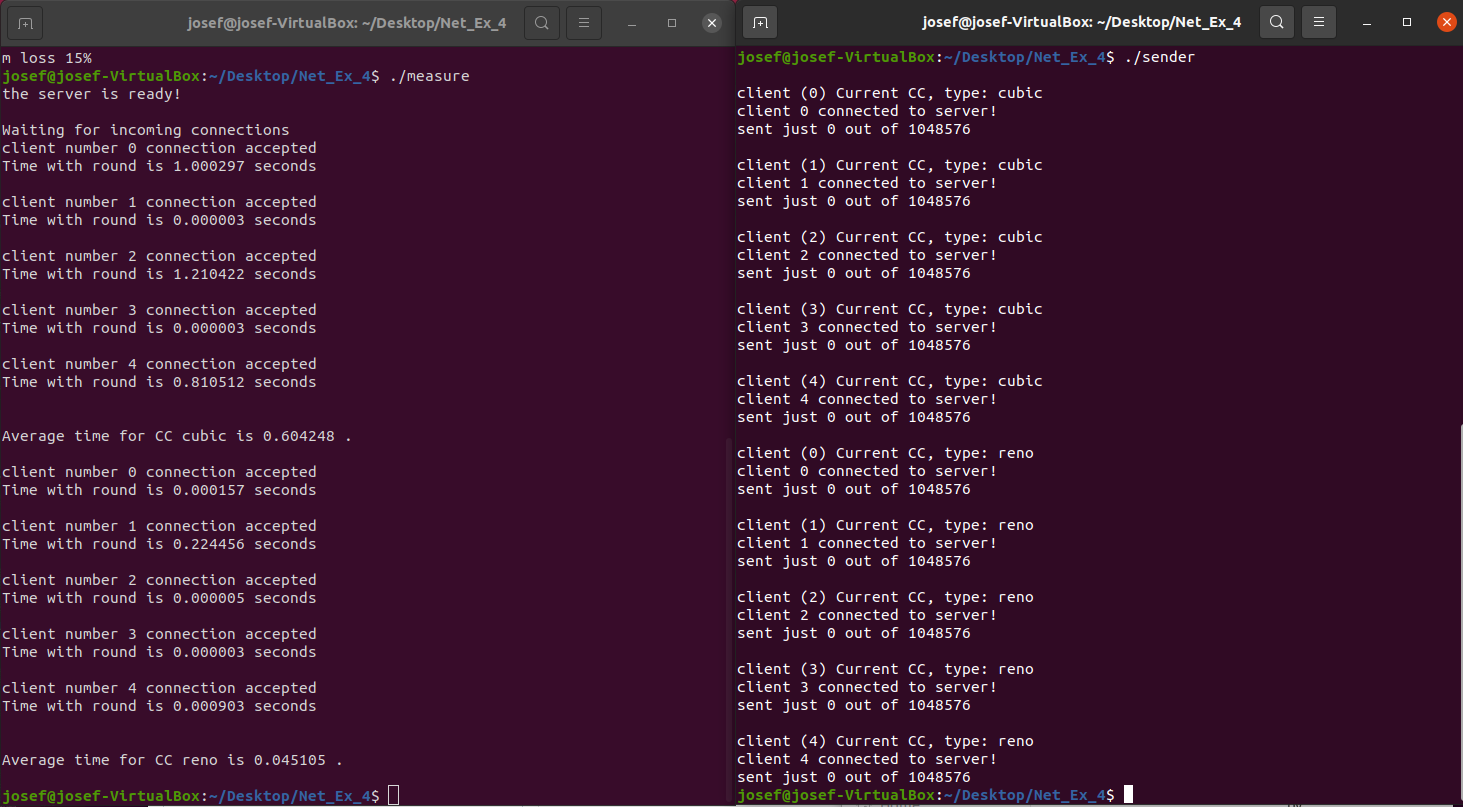
Running measure and sender for the first time with 0% loss:



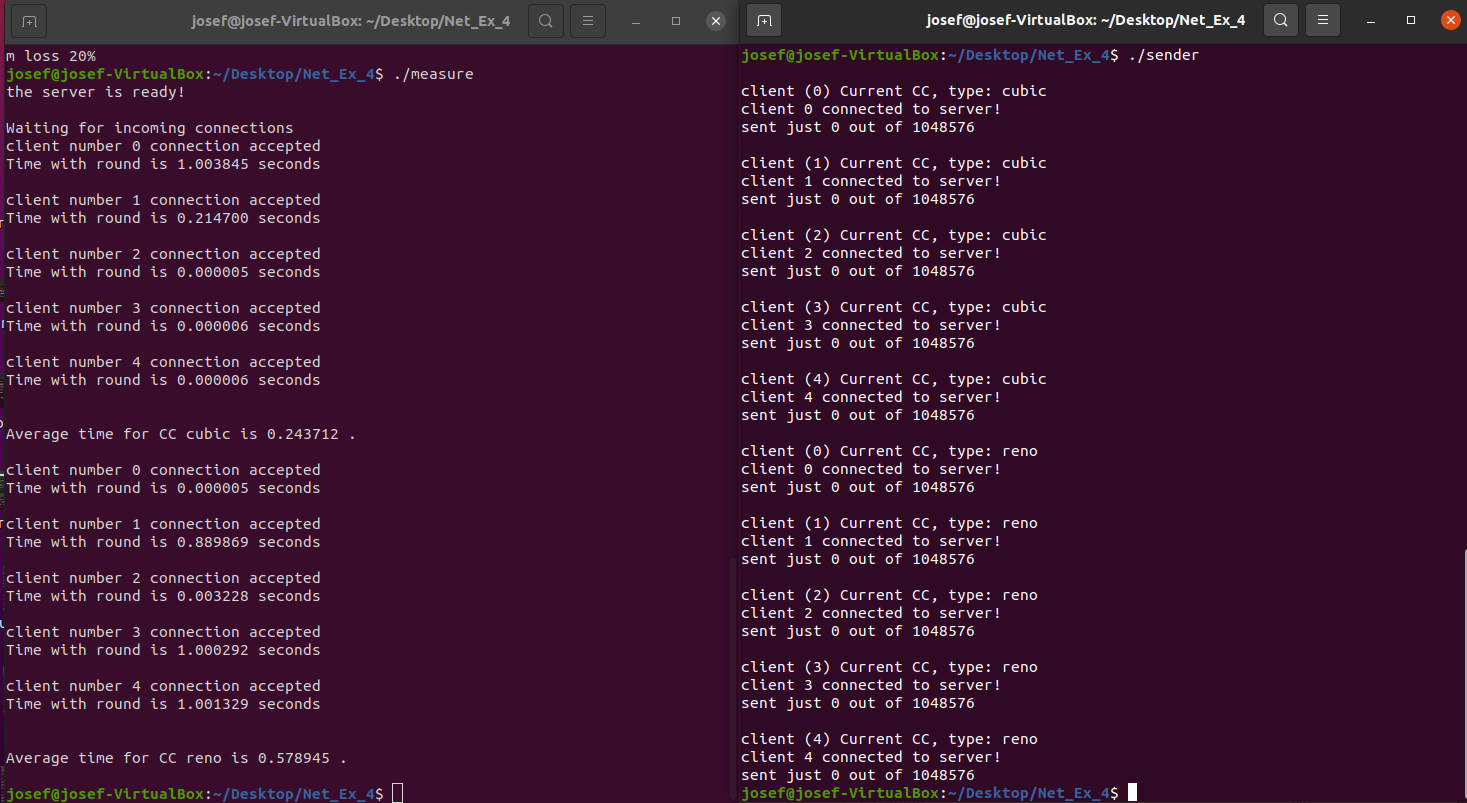
Running measure and sender with 10% loss:



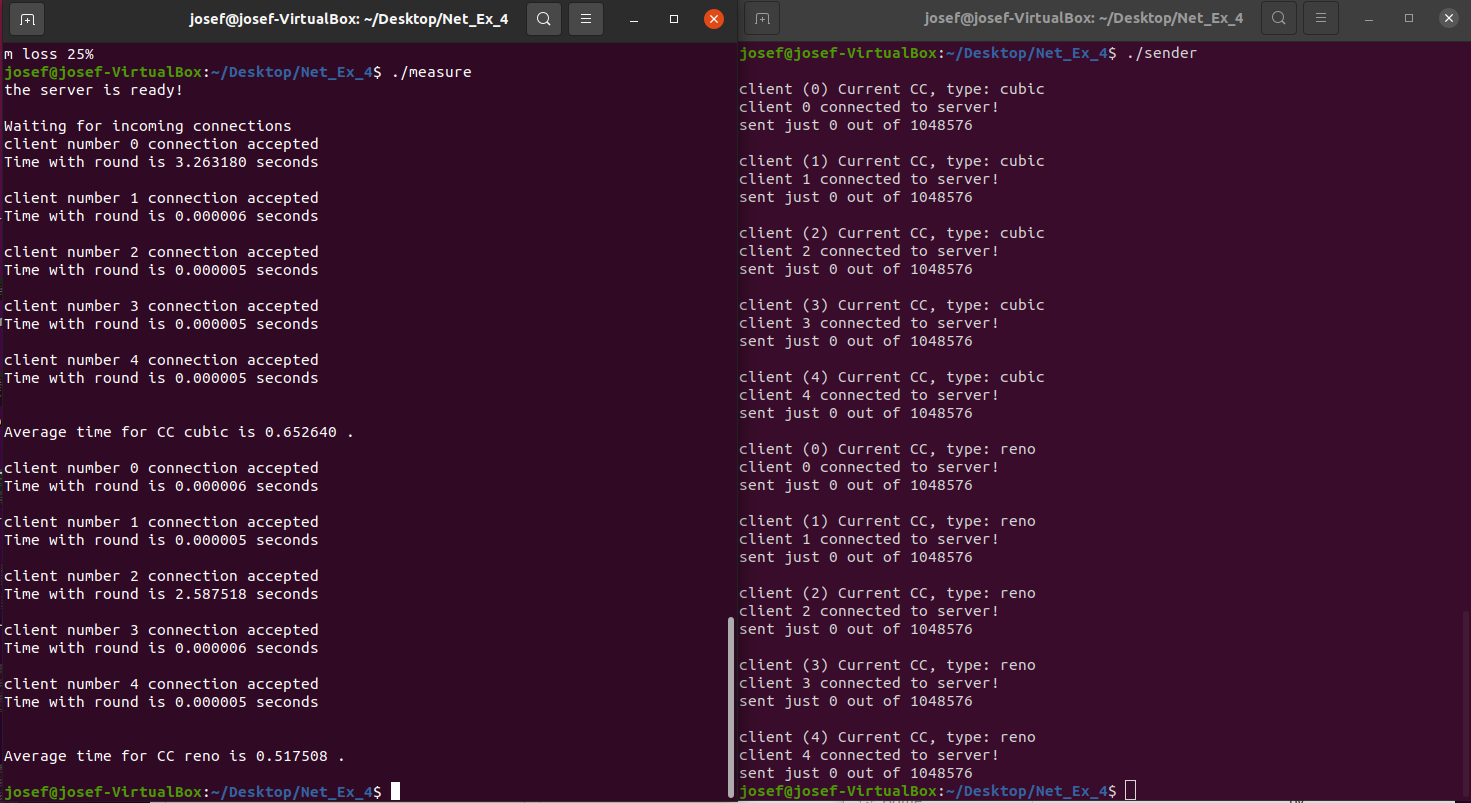
Running measure and sender with 15% loss:



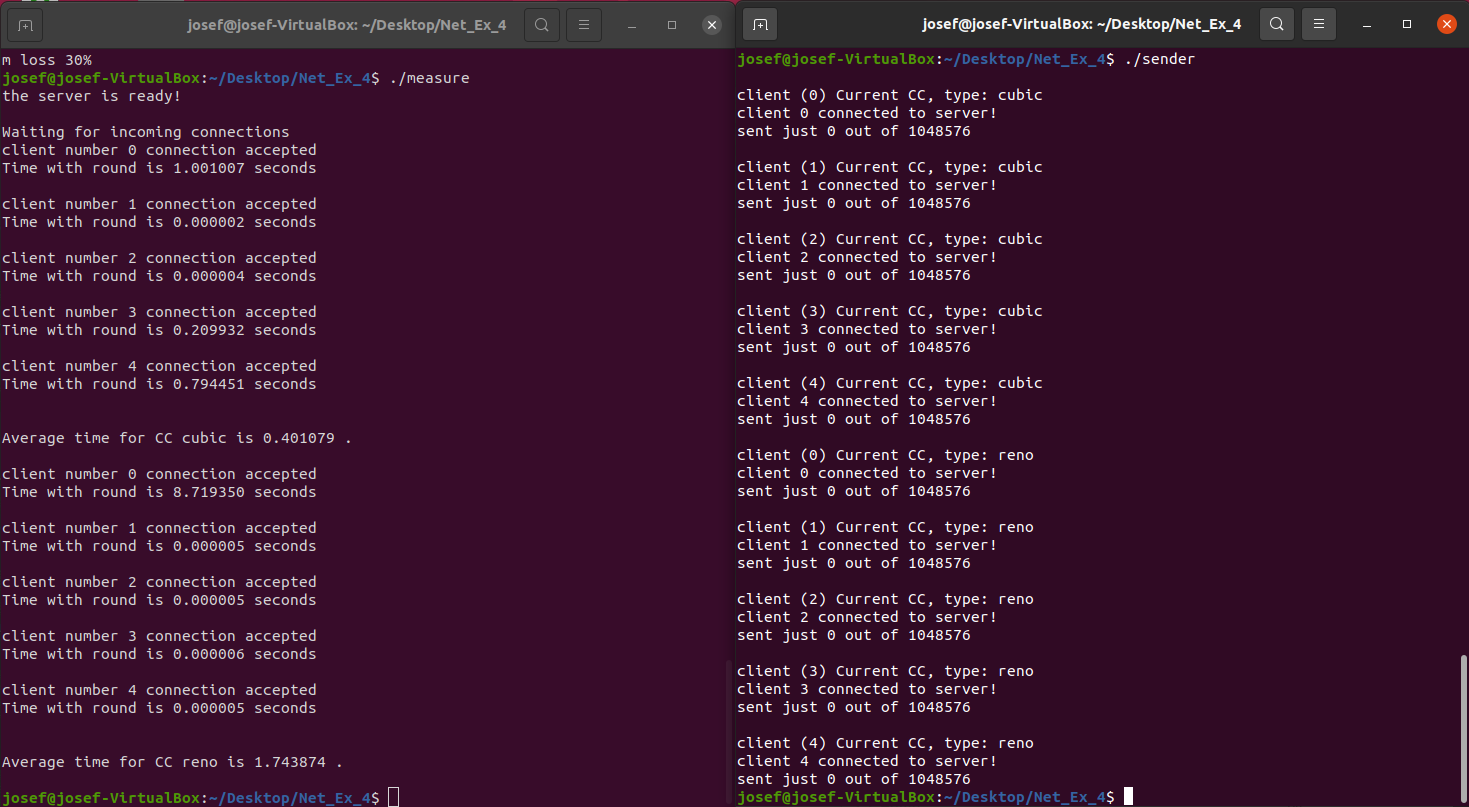
Running measure and sender with 20% loss:



Running measure and sender with 25% loss:



Running measure and sender with 30% loss:



Our graph:

