

תז: 214968240

תז: 208731091

קישור לגיט: https://github.com/matanklein/communication_networking_ex3.git

חלק 2 + 3

בחלק 2 השתמשתי ב ipv4.

0% איבוד פאקטות:

No.	Time	Source	Destination	Protocol	Length	Info
79	1.234788367	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
80	1.234815890	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
81	1.234838765	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
82	1.234844164	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
83	1.234850840	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
84	1.234855714	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
85	1.234861696	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
86	1.234866995	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
87	1.234871351	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
88	1.234876211	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
89	1.234881578	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
90	1.234886268	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
91	1.234891525	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
92	1.234896257	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
93	1.234901411	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
94	1.234905994	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
95	1.234911164	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
96	1.234915750	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
97	1.234921142	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
98	1.234925778	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
99	1.234930896	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
100	1.234935730	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
101	1.2349402346	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
102	1.234947098	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
103	1.234952496	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
104	1.234957379	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
105	1.234962767	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
106	1.234967804	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
107	1.234973235	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036
108	1.234978136	127.0.0.1	127.0.0.1	UDP	1078	54321 → 42433 Len=1036
109	1.234983126	127.0.0.1	127.0.0.1	UDP	1078	42433 → 54321 Len=1036

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Server finished!
*****
Run #1: Time = 36ms; Speed = 55.556MB/s
Run #2: Time = 104ms; Speed = 19.231MB/s
Run #3: Time = 34ms; Speed = 58.824MB/s
Run #4: Time = 38ms; Speed = 52.632MB/s
Run #5: Time = 45ms; Speed = 44.444MB/s

Average time: 51.648ms
Average bandwidth: 46.137MB/s
*****
matan@matan:~/Desktop/RT_EX3/part_b$
```

2% איבוד פאקטות:

```
Server finished!
*****
Run #1: Time = 38ms; Speed = 52.632MB/s
Run #2: Time = 49ms; Speed = 40.816MB/s
Run #3: Time = 133ms; Speed = 15.038MB/s
Run #4: Time = 91ms; Speed = 21.978MB/s
Run #5: Time = 149ms; Speed = 13.423MB/s

Average time: 92.570ms
Average bandwidth: 28.777MB/s
*****
matan@matan:~/Desktop/RT_EX3/part_b$
```

5% איבוד פאקטות:

```
Server finished!

*****
Run #1: Time = 261ms; Speed = 7.663MB/s
Run #2: Time = 64ms; Speed = 31.250MB/s
Run #3: Time = 131ms; Speed = 15.267MB/s
Run #4: Time = 67ms; Speed = 29.851MB/s
Run #5: Time = 220ms; Speed = 9.091MB/s

Average time: 149.110ms
Average bandwidth: 18.624MB/s
*****
```

10% איבוד פאקטות:

```
Server finished!

*****
Run #1: Time = 306ms; Speed = 6.536MB/s
Run #2: Time = 68ms; Speed = 29.412MB/s
Run #3: Time = 59ms; Speed = 33.898MB/s
Run #4: Time = 46ms; Speed = 43.478MB/s
Run #5: Time = 53ms; Speed = 37.736MB/s

Average time: 106.799ms
Average bandwidth: 30.212MB/s
*****
```

חלק שלישי

- (1) נשים לב ש reno היה יותר טוב שהאחוז איבוד פאקטות היה נמוך יותר אבל ככל שהעלו את האחוז איבוד פאקטות אז ה cubic נתן תוצאות יותר טובות. נראה זאת שע"י העלאת האחוז איבוד אז ה reno קפץ לזמן גבוה יותר ולא יחסי לקודמיו לעומת ה cubic שעלה באופן עקבי (ולא חד).
- (2) נראה ש RUDP נתן ביצועים טובים יותר מאשר ה TCP וגם באיבוד גבוהה. אפשר להסביר זאת ע"י שימוש של header קטן יותר מאשר ה TCP למשל.
- (3) לפי הנתונים נעדיף תמיד להשתמש ב RUDP. (למרות שהנתונים לא הכי מדויקים)

השאלות שבע"מ 8:

שאלה ראשונה-

1. בקשר ארוך על גבי רשת אמינה עם RTT גדול.

נשים לב:

- (*) אם הקשר קצר אז יכול להיות שלא נגיע בכלל ל SStreshold וכתוצאה מכך זה לא יועיל.
- (*) אם הקשר לא אמין אז ה SStreshold יכול להתקצר עוד לפני שהגענו אליו.
- (*) אם ה RTT קטן אז כן תהיה השפעה על הגדלת ה SStreshold אבל לא כמו RTT גדול שההשפעה תהיה גדולה יותר (כיוון שב RTT גדול ייקח יותר זמן לשלוח פאקטות ולכן ע"י הגדלת ה SStreshold נשלח בפחות זמן את הפאקטות).

שאלה שנייה-

$$א. \text{ בערך } 2S \frac{MSS}{\lg S \cdot RTT}$$

מכיוון שלא אובדות חבילות אז לפי ה slow start יהיו לוג חבילות (כי כל פעם נשלח פי 2 חבילות יותר מפעם קודמת) וכל חבילה לוקח RTT ולכן סך הזמן ייקח $\lg S \cdot RTT$. כמות הפאקטות יהיו סכום כך שבפעם הראשונה שולחים 1 ואז 2 ואז 4 וכו' עד שמגיעים ללוג $s2$ וזה יוצא בערך $s2$, כל פאקטה בגודל MSS ולכן כך הכמות תהיה $MSS \cdot s2$. ולכן התשובה תהיה א.

שאלה שלישית-

ה x המינימלי הוא 0 אבל מכיוון שזה מנוון את התרגיל לקחנו את השני שזה 1.
 $X = 1$. (הסימן '/' בא לסמן חילוק)
גודל כל חבילה הוא KBps.

זמן ההשהיה יהיה קצב התפשטות/המרחק לכן זה יהיה
 $1000/2 \cdot 10^8 = 0.0005$

זמן העברת חבילה תהיה קצב התקשורת/גודל החבילה כלומר $10^3 \cdot 8/8 \cdot 10^9 = 0.000001$

לכן ה RTT יהיה פעמיים זמן ההשהיה ועוד זמן העברת חבילה ולכן זה 0.001001 שניות.

ולכן גודל החלון יהיה גודל חבילה / (RTT * קצב תקשורת) שזה:

$$8 \cdot 10^9 \cdot 0.001001 / 8 \cdot 10^3 = \mathbf{1001}$$