הסבר הקלטות RUDP

בחלק זה נתייחס להקלטה “RUDP segmentation and retransmission”, הקלטה זו נוצרה באמצעות תוכנות rudp\_sender\_test ו-rudp\_receiver\_test שבנינו על מנת לבדוק את הפרוטוקול, התוכנות מבצעות שליחה פשוטה של מחרוזת. לצורך ההדגמה הורדנו את MAX\_SEGMENT\_SIZE ל-10 (על מנת שתתבצע סגמנטציה) והגדרנו packet loss של 10% בעזרת tc על מנת להדגים retransmission.

ראשית נראה את כל התקשורת:

A screen shot of a computer

Description automatically generated

שימו לב שכל הודעה באורך 8 היא הודעה של header בלבד (SYN/FIN/ACK).

ראשית שתי ההודעות הראשונות הן לחיצת הידיים:

A screen shot of a computer

Description automatically generatedA black screen with white text

Description automatically generated

בהודעה הימנית (SYN) ניתן לראות שלאחר 4 ספרות ראשונות (אורך החבילה – 0 – בשני בתים) נמצאות שתי הספרות של ה-flag שהן 01 בהתאמה ל-flag SYN בלבד. ניתן לראות מהפורטים שההודעה נשלחה מה-sender ל-receiver (פורט ה-receiver הוגדר להיות 1025).

בהודעה השמאלית (SYN-ACK) ניתן לראות באותן ספרות של ה-flags 03 בהתאמה ל- SYN | ACK (00000001 | 0000010 = 00000011 = 0x03).

בנוסף ניתן לראות בכל אחת מההודעות checksum (לדוגמא 0xfeff) ו-segment\_number (0 בשניהם).

לאחר לחיצת הידיים ההודעה הבאה היא הסגמנט הראשון:

A black screen with white text

Description automatically generated

ניתן לראות שבשדה len קיים 0010 (0xa00 בהקסה ב-little-endian) מכיוון שזה האורך המקסימלי, שדה flags הוא 0, ה-checksum הוא איזשהו מספר שתלוי בכל ההודעה, ו-segment\_number הוא 0 (זה הסגמנט הראשון בהודעה). את שאר המידע נראה מצד ימין של החלון:



ניתן לראות את החלק הראשון של ההודעה ששלחנו “Hello thro”, נראה את שאר ההודעה בסגמנט הבא.

לאחר מכן קיימת הודעת ACK מה-receiver:

A screen shot of a computer

Description automatically generated

ניתן לראות שהיא דומה להודעת SYN-ACK רק שהשדה flags מכיל רק 2 (רק דגל ACK), שדה segment\_number מכיל 0 מכיוון שזה ACK לסגמנט הראשון.

לאחר שסיימנו עם ההודעה הראשונה מנגנון stop-and-wait עובר להודעה השנייה.

רואים שההודעה השנייה נשלחה 4 פעמיים, ניתן להסיק שזאת מכיוון שבשלושת הפעמים הראשונות נאבד ה-ACK, נראה את אחת מהן:

A screen shot of a computer

Description automatically generated

ניתן לראות כאן דברים דומים להודעה הראשונה – אין דגלים והאורך הוא 10, אך בניגוד ניתן לראות ששני הבתים האחרונים של ה-header (0100) מכילים segment\_number 1 (ב-little endian) מכיוון שזה הסגמנט השני.

נראה את תוכן ההודעה בצד ימין של החלון:



ניתן לראות כאן את סוף ההודעה: “ugh RUDP!” ובסוף null-terminator (00 בבית האחרון).

כמו שאמרנו, ניתן לראות שאותה הודעה נשלחה 4 פעמים, עד שלבסוף התקבלה הודעת ACK:

A black screen with white text

Description automatically generated

ניתן לראות שבדומה ל-ACK הראשון ה-len הוא 0 ושדה flags הוא 2, אך כאן בדומה להודעה שזה ה-ACK שלה ב-segment\_number יש 1 (0x0100 ב-little-endian).

לבסוף קיימות שתי הודעות סגירה:

A black screen with white text

Description automatically generated A black screen with white text

Description automatically generated

ניתן לראות שהן דומות מאוד להודעות הפתיחה מלבד ששדה flags הוא 4 ב-FIN (צד ימין) ו-6 ב-FIN-ACK בהתאמה למה שהיינו מצפים מאיך שהשדות מוגדרים – שדה FIN הוא 4 ו-bitwise-or שלו עם שדה ACK שהוא 2 ייתן 6.

לאחר שראינו את התקשורת נגלה את פלט התכנות:

A screenshot of a computer

Description automatically generated

ניתן לראות שלאחר פתיחת הקשר ה-sender ביקש הודעה ונתנו לו את ההודעה “Hello through RUDP!”, הוא ניסה לשלוח אותה, כולל נסיונות חוזרים לאחר שהוא לא קיבל ACK 3 פעמים (כמו שראינו בהקלטה) ולבסוף ה-receiver קיבל את ההודעה והציג אותה בשלמותה (לאחר שפרוטוקול RUDP חיבר בחזרה את הסגמנטים).

לאחר מכן נתנו ל-sender את הפקודה “close” שגרמה לו לסגור את את התקשורת ושתי התוכנות דיווחו על סגירה.