שיעור 7 – TCP

**תרגיל: מתיידדים עם TCP**

בתרגיל זה נכיר לעומק את מנגנון הפעולה של TCP. שימו לב – מנגנון הפעולה הוא ברמה תיאורטית. עדיין לא דיברנו כיצד הוא ממומש במחשבים במציאות – זאת נעשה בשבוע הבא.

**1. נכון או לא נכון?**על כל היגד, כתבו האם הוא נכון או לא נכון, **והסבירו את עצמכם.**

1. TCP ו-UDP הם פרוטוקולים בשכבת הרשת.

| נכון / לא נכון: לא נכון  נימוק: TCP ו-UDP הם פרוטוקולים בשכבת התעבורה ולא בשכבת הרשת. |
| --- |
| *תשובה* |

1. שיחת TCP מתחילה בלחיצת יד משולשת

| נכון / לא נכון: נכון  נימוק: שיחת TCP מתחילה בתהליך של שלוש הודעות הנקרא "לחיצת יד משולשת". מטרתה להקים חיבור אמין בין שני הצדדים ולסנכרן פרמטרים. |
| --- |
| *תשובה* |

1. בפרוטוקול TCP כל חבילה נשלחת פעם אחת בלבד, גם אם לא התקבל עבורה אישור קבלה (ACK)

| נכון / לא נכון: לא נכון  נימוק: אם לא מתקבל אישור קבלה (ACK) על חבילה, TCP ישלח את החבילה שוב עד שתתקבל תגובה או שהחיבור יפוג. |
| --- |
| *תשובה* |

1. מטרת לחיצת היד המשולשת היא סנכרון מספרי ה-Seq איתם כל צד יתחיל, ווידוא כי החיבור הוא תקין ואכן השרת האזין בפורט בו פנו אליו.

| נכון / לא נכון: נכון  נימוק: לחיצת היד המשולשת משמשת לסנכרון מספרי הרצף בין הלקוח לשרת ולאימות שהשרת מאזין בפורט היעד. בנוסף, התהליך מוודא ששני הצדדים מוכנים לתקשורת. |
| --- |
| *תשובה* |

1. בחבילה בודדת שנשלחת ב-TCP, ה-seq הוא המספר של הבית הראשון בחבילה, ואילו ה-len הוא מספר הבתים בחבילה

| נכון / לא נכון: נכון  נימוק: מספר ה-seq בחבילת TCP מציין את המיקום של הבית הראשון בחבילה בזרם הנתונים. ה-len מציין את מספר הבתים בחבילה. |
| --- |
| *תשובה* |

1. בפרוטוקול UDP אפשר לקיים שיחה דו-כיוונית, כלומר ששני הצדדים שולחים מידע, בעוד ב-TCP המצב אינו אפשרי

| נכון / לא נכון: לא נכון  נימוק: שני הפרוטוקולים תומכים בתקשורת דו-כיוונית. ב-TCP זה קורה באמצעות חיבור מבוסס-חיבור, בעוד שב-UDP זה מבוצע ללא יצירת חיבור. ההבדל הוא באופי התקשורת, לא באפשרות לדו-כיווניות. |
| --- |
| *תשובה* |

1. ב-TCP נעשה שימוש ב-ISN כדי לשפר את קצב העברת החבילות.

| נכון / לא נכון: לא נכון  נימוק: ISN אינו קשור לשיפור קצב ההעברה אלא לסנכרון מספרי רצף בתחילת השיחה. זה מונע התנגשות בין חיבורים קודמים וחדשים. |
| --- |
| *תשובה* |

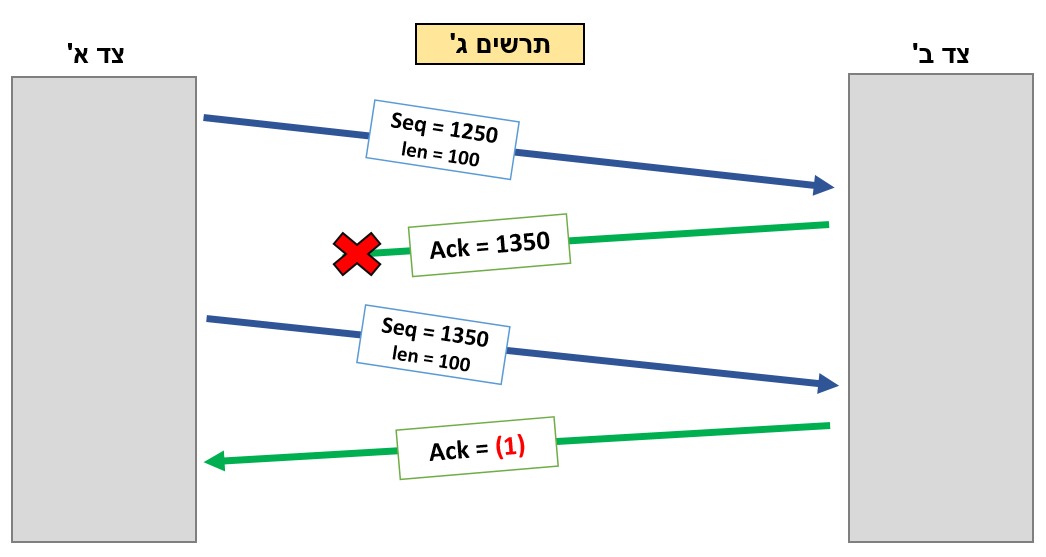
1. ב-TCP אם הסדר של החבילות שנשלחו השתבש (לדוגמה, חבילה מס' 2 הגיעה ליעד לפני חבילה מס' 1), הצד המקבל ידאג לסדר אותן.

| נכון / לא נכון: נכון  נימוק:TCP מבטיח סדר נכון של החבילות. אם חבילות מגיעות בסדר שגוי, הצד המקבל יסדר אותן לפי מספרי ה-seq לפני העברתן לאפליקציה. |
| --- |
| *תשובה* |

**2. השלם את החסר**מלאו את הפרטים החסרים (צבועים באדום) בתרשימים הבאים, והסבירו את תשובתכם:

| **(1) 5641**  **(2) 3914**  **(3) 5661**  **הסבר: השרת מחזיר את הack של הצד b באמצעות seq והseq הזה יהיה שווה לack של הצד ב, מתוך ההקשרים הקודמים אפשר לדעת שlen של צד א הוא 20 ושל צד ב הוא 3, אז בתשובה שנייה seq יהיה את הack של צד א והack שלו יהיה את הseq של צד א + 20.** |
| --- |
| *תשובה* |

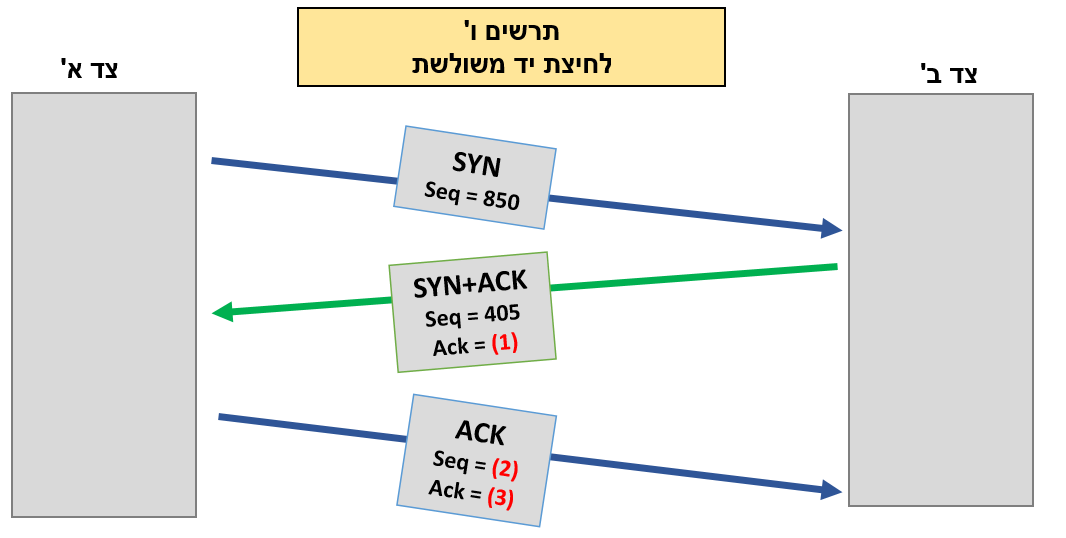
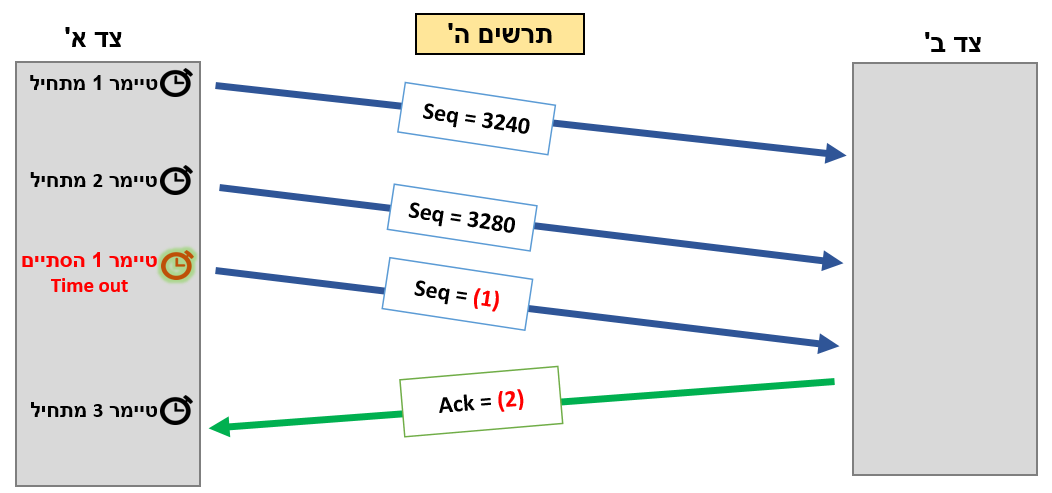
| **(1) 810**  **(2)825**  **(3)825**  **הסבר: התשובה הראשונה של צד ב יהיה הקלט ראשון של צד א, אך בגלל שצד א לא הצליח לשלוח את הפקטה השנייה הוא שולח אותה שוב פעם, והתשובה יהיה זה שהוא קיבל אותה.** |
| --- |
| *תשובה* |



| **(1) 1350**  **הסבר: הצד א לא קיבל תשובה של ack = 1350 ובגלל זה הוא שולח את התשובה שוב פעם כדי לוודא שהיא יגיע.** |
| --- |
| *תשובה* |

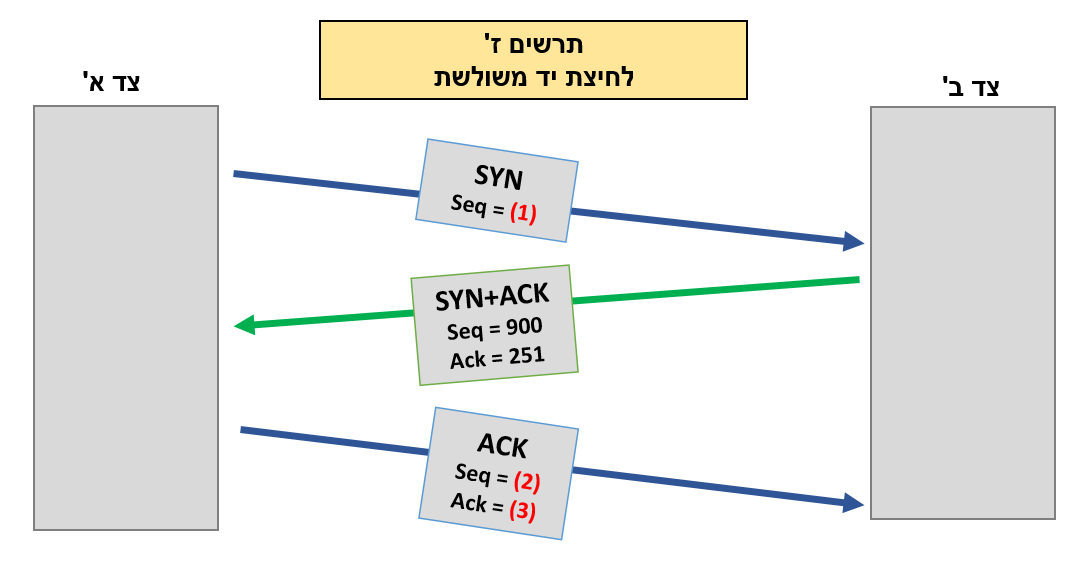
**s**

| **(1) 650**  **(2) 30**  **(3) 650**  **הסבר: צד א קיבל את התשובה שack הוא 650 אז הseq של צד א יהיה 650, אך היא לא הגיע לצד ב אז צד ב שוב פעם שולח את ההודעה שהוא שלח מקודם.** |
| --- |
| *תשובה* |



| **(1) 3240**  **(2) 3280**  **הסבר: צד ב זהה את seq 3280 ועשה כבר זיהוי (3320) אך לא שלח בחזרה תשובה מיידית, בגלל שטיימר ראשון כבר נגמר ולא נשלח אליו תשובה בזמן היה retransmission והוא שוב שלח 3240, לאחר מכן השרת שני שלח תשובה של 3240 שהיא 3280.** |
| --- |
| *תשובה* |

| **(1) 851**  **(2) 851**  **(3)406**  **הסבר: שני הצדדים מתחילים סינכרון, צד א שולח seq של 850 להתחלה, בחזרה צד ב מזהה את המספר (850) ואז שולח את seq משל עצמו (405) בחזרה לכך צד א מזהה אותו ומוסיף אחד (406) ושולח גם את ה – seq הבא 851.** |
| --- |
| *תשובה* |



| **1) 250**  **(2) 251**  **(3) 901**  **הסבר: צד א שולח seq של 250 של סנכרון, בתגובה לכך צד ב גם שולח את המספר שלו (900), ומזהה את צד א (דרך ack = 251) לאחר מכן צד א מזהה את צד ב ושולח את seq = 901 וגם את סימן הזיהוי של הצד השני ack = 251.** |
| --- |
| *תשובה* |

**3. שאלות פתוחות**כתבו תשובה מפורטת לכל שאלה.

1. השלימו, כיצד נראית לחיצת יד משולשת בפרוטוקול TCP?

| **1. צד א' שולח לצד ב': את isn שלו**  **2. צד ב' שולח לצד א': את הisn של משתמש א + 1 וisn של עצמו**  **3. צד א' שולח לצד ב': את isn של משתמש א + 1** |
| --- |
| *תשובה* |

1. באותו אופן, כתבו כיצד נראה סיום תקשורת מסודר בפרוטוקול TCP (כתבו בעצמכם מה כל צד שולח לצד השני):

| 1. **צד א' שולח לצד ב': הודעת FIN, מסמן שסיים לשלוח נתונים.** 2. **צד ב' שולח לצד א': הודעת ACK עבור ה-FIN של צד א + הודעת FIN, מסמן שגם הוא סיים לשלוח נתונים.** 3. **צד א' שולח לצד ב': הודעת ACK עבור ה-FIN של צד ב.** |
| --- |
| *תשובה* |

1. סהר שינה את ההגדרות בפרוטוקול ה-TCP שלו, כך שהוא לא מגריל ISN אקראי. מה הבעיה שעלולה לצוץ בעקבות זאת?

| **אם ISN אינו אקראי, התוקפים יכולים לנחש אותו ולבצע התקפות כמו spoofing או לזייף חיבורים.** |
| --- |
| *תשובה* |

1. הדר משתמשת בגרסת TCP שעובדת ללא טיימר על החבילות שנשלחות. מה עלול להתרחש ומדוע?

| **ללא טיימר, TCP לא יזהה פקטות אבודות אם אין תגובת ACK בזמן. זה עלול לגרום לאובדן נתונים או לעיכוב בלתי מוגבל בהעברה.** |
| --- |
| *תשובה* |

1. נגה איחרה לשיעור, והחמיצה את ההסבר על Sequence Number. הסבירו לה מה זה, ולמה הוא משמש?

| **Sequence Number הוא מספר ייחודי לכל חלקי נתונים, המסמן את מיקומן במידע. הוא משמש לוודא שהנתונים יגיעו בסדר הנכון ולטפל בחלקי נתונים שאבדו או הגיעו כפולות.** |
| --- |
| *תשובה* |

1. ציין שתי דוגמאות למקרים בהם נעדיף להשתמש בפרוטוקול TCP ושתי דוגמאות למקרים בהם נעדיף UDP (לדוגמה, במה נעדיף להשתמש כאשר נשלח מייל?)

| **TCP:שליחת מייל,הורדת קבצים.**  **UDP:סטרימינג וידאו,משחקי רשת.** |
| --- |
| *תשובה* |

1. מדוע TCP נחשב פרוטוקול אמין יותר מ-UDP? שלב בתשובתך דוגמאות מהפרוטוקולים.

| **TCP נחשב אמין יותר מ-UDP בגלל הסיבות אלו:**  **אישור קבלה, דוגמה: HTTP מבטיח טעינת דפים תקינה.**  **בקרת סדר ותיקון שגיאות: TCP מסדר נתונים מחדש ומתקן בעיות. דוגמה: FTP מבטיח העברת קבצים שלמה.**  **UDP לא מספק מנגנונים אלה ולכן מתאים לשידורים מהירים כמו וידאו או משחקים.** |
| --- |
| *תשובה* |

1. מדוע UDP נחשב פרוטוקול מהיר יותר מ-TCP? מה ב-UDP יותר מהיר?

| **UDP מהיר יותר מ-TCP כי הוא אינו מבצע אימות קבלה, תיקון שגיאות, או סידור מחדש של נתונים. זה חוסך זמן ומקטין עיכובים.** |
| --- |
| *תשובה* |