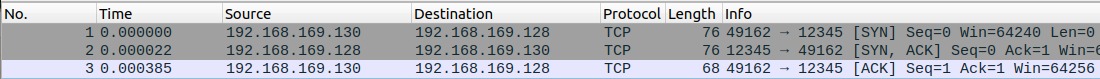
משימה 2- חלק 1

הדגל המסמן על SYNCHRONIZE

בקשה לתחילת התקשרות

אם כן, נתאר את "טקס" הקמת החיבור בין הלקוח לשרת.



בתמונה שלעיל ניתן לראות את שליחת ההודעה הראשונית של הלקוח אל השרת. הודעה עם דגל דולק "SYN".

רצף הבתים שנשלחו בהתחלה

0, כיוון שזוהי אינה הודעה עם DATA אלא הודעה לבקשת התקשרות

Source-IP של הלקוח המבקש להתקשר לשרת.

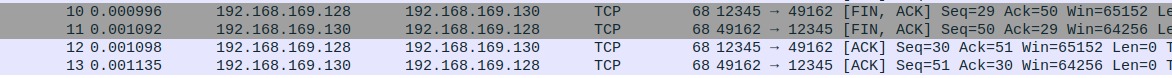
Destination-IP – IP של השרת.

הודעת החזרה של השרת על "אישור החיבור"

לאחר בקשת החיבור הראשונית של הלקוח, השרת שולח ללקוח גם כן בקשה לסנכרון בעזרת הרמת הדגלים "SYN" ובנוסף אישור על בקשת ההתקשרות של הלקוח. לכן 2 הדגלים מורמים: ACK+SYNC.

כמו כן, לאחר מכן הלקוח מחזיר ACK לשרת על אותה בקשה חוזרת להתקשרות ומייד לאחר מכן שולח את הDATA אותו הוא מעוניין להעביר לשרת.

כעת נראה את קיום החיבור:



כפי שניתן לראות בתמונה שלעיל, מתקיימת תקשורת בין הלקוח לשרת.

בשורה הראשונה בצד ימין, אנו רואים את receive buffer(ack)&send buffer(Seq). נשים לב כעת כי בתיבת ה- seq אצל השרת, גודלו 30, אך בתמונה הראשונה שלעיל גודל הsend buffer היה 0.

המשמעות היא שבהתחלה הלקוח לא שלח שום דבר ולכן לשרת לא היה שום דבר לשלוח לו חזרה. שכן, הגדרנו בקוד כי השרת יחזיר את אותה הודעה של הלקוח אך בupper case.

כעת אנו רואים שהשרת כבר החזיר הודעת ACK וכעת מספר הבתים שהוא מוכן לקבל הם מ51 ומספר הבתים שהוא שולח הם החל מ30.

תמונה שמכילה טקסט, עיתון, קבלה

התיאור נוצר באופן אוטומטי

כאן בתמונה אנו רואים את הודעת השרת. ניתן לראות כי השרת מעוניין לשלוח הודעה על ידי הרמת הדגל PSH+ACK

הרמת דגלים אלו מודיעים כי השרת כעת שולח גם ACK על ההודעה שהלקוח שלח לו וגם כעת מוסיף DATA משלו על שליחת הודעת הACK ללקוח.

אנו רואים בתיבת ה Len כי אורך ההודעה הוא 28 בתים. בנוסף, אנו כעת רואים את המספר האמיתי של הבתים שהתקבלו ונשלח עד כה בreceive buffer &send buffer (ללא סינון של Wireshark).

אנו יודעים שההודעה הגיעה מהשרת ללקוח על ידי התיבה הנ"ל : תמונה שמכילה טקסט, עיתון, קבלה

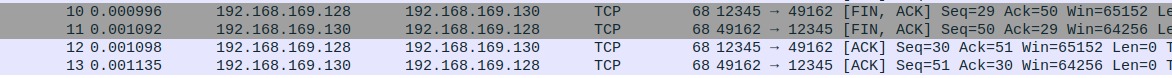
התיאור נוצר באופן אוטומטי

ניתן לראות כי ההודעה נשלחת מפורט 12345 (הפורט שהקצאנו לשרת) אל פורט 49162, הפורט שהוקצה אקראית ללקוח.

כמו כן, אנו רואים את הודעת הACK שהלקוח שולח חזרה כפידבק להודעת השרת. בשורה מספר 68 הלקוח מרים את דגל הACK .

השרת שולח ללקוח הודעה עם הדגלים FIN+ACK שמורמים המצביעים על בקשה לניתוק החיבור

נראה כעת את "טקס סיום החיבור".



גודל החבילה שנשלחת

"קיבלתי עד בית 30"

הלקוח מודיע לשרת: שולח לו מידע החל מבית 51

ניתן לראות כי ברגע שהשרת קיבל את הודעות הלקוח, הוא מודיע ללקוח על סיום החיבור (כך הגדרנו בקוד, כי השרת הוא זה שמבקש להפסיק את החיבור).

נשים לב כי לשרת וללקוח יש לכל אחד buffer שונה משלהם. ללקוח יש receive & send buffers וגם לשרת.

Receiver buffer- מתואר כACK , כלומר כמה הלקוח/שרת קיבל עד כה ועד איזה בית הוא מוכן לקבל.

ניתן לראות בדוגמא שבה השרת מודיע על סיום ההתקשרות כי הוא שולח ACK ללקוח עם המספר 50. כלומר, השרת קיבל עד 50 והוא מוכן לקבל כעת מבית זה והלאה.

לעומת זאת, השרת שולח ללקוח Seq=29.

Send Buffer- מתואר כSeq. בדוגמא שלנו, השרת מודיע ללקוח כי הוא שולח לו החל מבית 29 + גודל החבילה שהוא שולח.

גודל החבילה מופיע בLen.

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

ניתן לראות כאן את ההודעה שנשלחה מאת הלקוח, גודל הDATA הינו 28 בתים, השמות המלאים שלנו.

תשובת השרת ללקוח.

השרת מחזיר ACK על הודעת הלקוח + אותה הודעה בUPPER CASE

ניתן ראות זאת על ידי החזרת ה ACK

ומשבצת הLEN נקבל 28 בתים שנשלחים

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

הלקוח שולח הודעת ת"ז לשרת.

שוב ניתן לראות את seq שולח החל מבית 29

גודל הDATA הוא 21 . כמו כן הלקוח מודיע כי הוא קיבל עד 29 והוא מוכן לקבל משם והלאה (ACK).

עד כה הצגנו את המספרים המדומים (רלטיביים) שהציג לנו הwireshark לשם הנוחות.

כעת נציג את המספרים האמיתיים של החבילות לפי המספור שלהן החל הראשית ההתקשרות בינהם.

תמונה שמכילה טקסט, עיתון, קבלה

התיאור נוצר באופן אוטומטי

המספר האמיתי של השרת ב SEND BUFFER

זהו המספר האמיתי של תחילת ההתקשרות

בין הלקוח לשרת. (SEND BUFFER) של הלקוח.

המספר האמיתי של הלקוח בעת תהליך סיום ההתקשרות

המספר האמיתי של השרת בעת תהליך סיום ההתקשרות

ניתן לראות בתמונה שלעיל את הקמת החיבור עם המספר האמיתי של תחילת ההתקשרות עבור השרת בsend buffer , וכן בrecieve buffer.

כמו כן, ניתן לראות את המספר האמיתי המופיע לעיל תחת "ACK" וSeq עבור הלקוח והשרת בהתאמה.

תמונה שמכילה טקסט, עיתון, קבלה

התיאור נוצר באופן אוטומטי

בתמונה זו אנו רואים את ה MSS – גודל ההודעה המקסימלית שיכולה להישלח (ולהתקבל) מהשרת ללקוח ומהלקוח לשרת בהתאמה. זהו הsegment המקסימלי שיכול להישלח בהודעה אחת.