

מבוא לרשתות תקשורת

תרגיל מספר 1

חלק א:

א. צרו קבצי קליינט וסרבר בעזרת הקוד שבמצגת התרגול עבור פרוטוקול udp.

ב. שנו את קוד הלקוח שישלח לשרת את השמות של המגישים במקום hello world. הריצו את קוד הקליינט והסרבר והסניפו את התעבורה בעזרת wireshark. אתם יכולים להשתמש בשתי מכונות וירטואליות, מכונה וירטואלית אחת ומערכת ההפעלה המקורית שלכם או להשתמש במכונה אחת עם מספר טרמינלים, לבחירתכם. העיקר, שתוכלו לתפוס את התעבורה ב wireshark ולנתח אותה.

ג. הסבירו את הקוד ואת התעבורה ב wireshark בעזרת הסברים טקסטואליים וצילומי מסך מתאימים. הסבירו כיצד הקוד השפיע על התעבורה עצמה (בשכבות השונות ובשדות השונים), ובפרט, הסבירו את השכבות השונות של המידע בתעבורה והמידע בהן.

חלק ב:

כעת נממש מעין שירות 144 דיגיטלי.

המטרה שלנו היא לבנות שרת שיודע לקבל שאלות ולהשיב עליהן.

בשירות 144 הטלפוני, ניתן לשאול למשל: "מה מספר הטלפון של ישראל ישראלי?".

שירות 144 יסתכל ברשומות שלו ויענה: "מספר הטלפון של ישראל ישראלי הוא 03-1234567".

אנחנו רוצים לבנות משהו דומה, אבל במקום עם מספרי טלפון, עם כתובות IP, ובמקום שמות, דומיינים.

עליכם לבנות שרת בפרוטוקול UDP. לפני שהשרת מתחיל להאזין לסוקט, עליו לטעון קובץ בשם zone.txt אשר מכיל את המיפויים שהוא מכיר (מיפויים בין דומיינים לכתובות IP).

למשל, להלן דוגמה לתוכן אפשרי של קובץ zone.txt:

biu.ac.il,1.2.3.4,A

.co.il,1.2.3.5:777,NS

example.com,1.2.3.7,A

שימו לב שבכל שורה ישנו רק מיפוי אחד, ובכל שורה 3 "שדות": דומיין, כתובת IP (או כתובת ip נקודתיים פורט), ו"סיווג" A או NS, אשר מופרדים ביניהם בפסיק ללא רווחים. כמות השורות בקובץ אינה מוגבלת.

השרת מצפה לקבל הודעות אשר יכילו דומיין כלשהו, ועל כל הודעה כזאת, הוא מחזיר את כתובת ה IP המתאימה, על בסיס קובץ ה zone.txt שלו.

למשל, נתייחס לקובץ ה zone.txt הנ"ל.

נניח והשרת קיבל כהודעה "biu.ac.il" או "example.com" (ללא הגרשיים).

מכיוון שלכל אחד מהדומיינים הללו יש רשומה מתאימה בקובץ, וספציפית, רשומה מסוג A, התשובה צריכה להיות פשוט הרשומה המתאימה. כלומר, "biu.ac.il,1.2.3.4,A" במקרה של "biu.ac.il" ו"example.com,1.2.3.7,A" במקרה של "example.com" (בלי הגרשיים כמובן).

מעתה הלאה, תמיד הכוונה ללא הגרשיים, למרות שלא אכתוב זאת יותר ליד כל פעם שיהיו גרשיים.

במידה והשרת מקבל דומיין שלא מופיע בzone.txt שלו, הוא משיב: non-existent domain
נניח והשרת קיבל כהודעה "google.co.il".

אין רשומה מתאימה בקובץ.

עם זאת, יש רשומה מסוג NS, אשר רלוונטית עבור **כל** הכתובות המסתיימות ב co.il. (שורה מספר 2 בקובץ), ולכן, תופסת גם עבור google.co.il

לכן, במידה והשרת מקבל כהודעה "google.co.il", עליו להשיב "co.il,1.2.3.5:777,NS".

עליכם לממש שרת נוסף, אחר, שיקרא רזולבר.

גם שרת הרזולבר מצפה לקבל הודעות אשר יכילו דומיין כלשהו, אך לשרת הרזולבר אין קובץ zone.txt

כלומר, כאשר שרת הרזולבר מקבל שאלה על דומיין שהוא לא נשאל עליו בעבר, שרת הרזולבר מפנה את השאלה לשרת ה"אב" שלו. כתובת ה IP של שרת האב תועבר כפרמטר לשרת הרזולבר. הרזולבר מפנה את השאלה בדיוק באותו הפורמט לשרת האב, מקבל את התשובה ממנו, "לומד" אותה, ועונה ללקוח. (תכף אדייק את שני המשפטים הקודמים)

המשמעות של "למידה" היא, שמעתה והלאה, לקוחות שישאלו את השרת את אותה השאלה יקבלו מיד את התשובה - כי השרת כבר יודע אותה. אין לשרת צורך שוב ללכת לשאול את שרת "האב".

למשל, נניח ושרת הרזולבר קיבל בתור שרת ה"אב" את השרת הראשון שהגדרנו בתרגיל.

ונניח שלקוח שואל את הרזולבר "biu.ac.il".

ה cache של הרזולבר כרגע ריק, ולכן, הרזולבר נאלץ לשאול את שרת ה"אב". שרת ה"אב" יענה לו "biu.ac.il,1.2.3.4,A", והרזולבר שומר את זה ב cache וגם שולח את זה ללקוח.

מעתה והלאה, ולמשך x שניות, כל לקוח שישאל את הרזולבר "biu.ac.il", הרזולבר מיידית יענה לו "biu.ac.il,1.2.3.4,A". אם יחלפו x שניות מאז שהרזולבר למד את התשובה משרת ה"אב", המידע יימחק מה cache של הרזולבר, והרזולבר יילמד אותו שוב (באותו האופן, שוב למשך x שניות, וחוזר חלילה). הערך x יינתן כקלט לרזולבר.

נניח והרזולבר קיבל שאלה "google.co.il". אין לו אותה ב cache, אז יפנה לשרת האב. שרת האב יענה לו "co.il,1.2.3.5:777,NS". הרזולבר שומר את זה ב cache כמו תשובות אחרות למשך x שניות, אבל, מיד פונה לכתובת השרת הרשומה בתשובה, כלומר, פונה לכתובת 1.2.3.5 לפורט 777 במקרה הזה, ושולח לשרת בכתובת זו את אותה השאלה "google.co.il".

כאשר הרזולבר יקבל תשובה לשאלה מהשרת ב 1.2.3.5, הוא מטפל בה לפי אותה הלוגיקה שתוארה עד כה. כלומר, שומר ב cache ל x שניות, ואם התשובה היא מסוג A מחזיר אותה ללקוח אבל אם התשובה מסוג NS, הרזולבר פונה לשרת שמופיע בתשובה וחוזר חלילה.

שימו לב: אם הרזולבר יקבל שאלה "foo.co.il" במהלך ה x שניות קרובות, הוא אינו פונה לשרת האב שלו, כי התשובה "co.il,1.2.3.5:777,NS". אמורה להיות לו ב cache עדיין, ולכן יפנה ישירות ל 1.2.3.5 לפורט 777 עם השאלה. אם היה מקבל שוב את השאלה "google.co.il" במהלך ה x שניות הקרובות, אמור להיות לו תשובה עדיין ב cache - לכן הוא אינו אמור לפנות לאף אחד ויכול לענות ישר מה cache.

שימו לב 2: יכול להיווצר מצבים של "שרשראות" ארוכות. למשל, שרת הרזולבר פונה לשרת אב, שמפנה אותו לשרת אחר, שמפנה אותו לשרת אחר, שמפנה אותו לשרת אחר (וכו'), אבל בסוף, כל שרשרת תסתיים. כלומר, בסוף השרשרת תמיד יהיה שרת שיענה בעזרת רשומה מסוג A או שהדומיין לא קיים (כפי שהוגדר לעיל).

בנוסף, כתבו לקוח אשר מקבל בלולאה קלט מהמשתמש (דומיין), ומדפיס למסך את כתובת ה IP שלה (בלבד) בעזרת השרתים הנ"ל.

השרת (אב) מקבל 2 ארגומנטים כקלט לתוכנית:

[myPort] [zoneFileName]

כאשר [myPort] זה הפורט שהשרת עושה לו bind, ו [zoneFileName] זה הנתיב והשם של קובץ ה zone.txt

השרת רזולבר מקבל 4 ארגומנטים כקלט לתוכנית:

[myPort] [parentIP] [parentPort] [x]

כאשר [myPort] זה הפורט שהרזולבר עושה לו bind, ושני הפרמטרים הבאים זה ה ip והפורט של שרת האב שאליו הוא פונה במידת הצורך, והפרמטר [x] הינו משך הזמן שרשומה נשמרת ב cache.

הלקוח מקבל 2 ארגומנטים כקלט לתוכנית:

[serverIP] [serverPort]

כאשר שני הפרמטרים זה ה ip והפורט של שרת הרזולבר שהלקוח מפנה אליו שאלות.

שימו לב: כל הארגומנטים, בכל התוכניות, הם ארגומנטים ל main בלבד. **שימו לב שכל הארגומנטים ל main שהוגדרו לעיל מועברים בשורת הפקודה בלבד.**

חובה להקפיד על דוגמאות הקלט/פלט במדויק המופיעות בעמוד הבא.

אין לסטות מהן בשום צורה, גם לא סטייה של תו אחד.

דוגמת קלט פלט:

תוכן קובץ zone.txt:

biu.ac.il,1.2.3.4,A

.co.il,1.2.3.5:777,NS

example.com,1.2.3.7,A

תוכן קובץ zone2.txt:

www.google.co.il,1.2.3.8,A

mail.google.co.il,1.2.3.9,A

דוגמא להרצת שרת אב:

```
server.py 55555 zone.txt
```

דוגמא להרצת שרת אב נוסף:

```
server.py 777 zone2.txt
```

דוגמא להרצת שרת רזולבר:

```
resolver.py 12345 127.0.0.1 55555 60
```

דוגמא להרצת לקוח:

```
client.py 127.0.0.1 12345
```

להלן המסך של הלקוח בהרצה לדוגמא (השרתים אינם מדפיסים פלט כלל):

הטקסט בכתום הוא הערה שלי כדי להסביר ואינו חלק מהמסך

קלט שהמשתמש הזין לתוכנית // biu.ac.il

1.2.3.4

קלט שהמשתמש הזין לתוכנית // www.biu.ac.il

non-existent domain

קלט שהמשתמש הזין לתוכנית // mail.google.co.il

1.2.3.9

מה יש להגיש?

1. דו"ח המכיל מענה לחלק הראשון של התרגיל. דו"ח זה יהיה קובץ PDF בלבד. כל פורמט אחר לא יתקבל (גם לא word).
- יש להשתמש בחצים, קווים ומסגרות (למשל בעזרת "צייר") כדי להדגיש את המידע הרלוונטי מתוך צילום המסך.
- אין צורך לעשות מאות צילומי מסך, אבל עליכם להבין שהבודק חייב לראות שהבנתם ושאתם יודעים להסביר את השימוש ב-wireshark ואת השכבות השונות, ולכן עליכם לספק מספיק צילומי מסך כדי שאכן הוא יוכל להיווכח בזאת.
2. קבצי השרת, הרזולבר והלקוח שכתבתם בחלק השני, עם תיעוד בסיסי (כל כמה שורות). חובה להקפיד על שמות הקבצים במדויק, כולל case-sensitive. **אי הקפדה על זה - תגרור הורדה אוטו' של 10 נק'.**
3. המימוש הוא ב-Python גרסה 3 ומעלה בלבד. חובה לבדוק את תקינות התרגיל על שרתי האוני'. אין להגיש שום קובץ קוד נוסף למעט השלושה שצויינו בסעיף 2. מותר להשתמש אך ורק בספריות סטנדרטיות של Python, ולא ניתן להשתמש בשום ספרייה אחרת (למשל, כזו שדורשת התקנה).
4. חובה להקפיד על הקלט/פלט במדויק. **אי הקפדה על זה - תגרור ציון 0 אוטו'.**
5. חובה להגיש בנוסף קובץ טקסט בשם details.txt (כך עליו להקרא במדויק) עם שמות ות.ז. של המגישים. כלומר:

Israel Israeli 012345678

Israela Israeli 123456789

- אם מגישים לבד, אזי רק השורה הראשונה רלוונטית.
- חובה להקפיד על תוכן הקובץ במדויק כפי שהוגדר לעיל. (וכמובן באנגלית)
- אי הקפדה על זה - תגרור הורדה אוטו' של 20 נק'.**
6. הגשה ביחידים או בזוגות, לבחירתכם. (**רק אחד** מבני הזוג מגיש את התרגיל בפועל)

בהצלחה