שיעור 08 – תרגיל מסכם סקאפי

**E.T. COME HOME**

זה היה עוד יום ה' שגרתי במחלקת "חיים תבוניים" של סוכנות החלל האמריקאית, נאס"א, כשלפתע התפרץ למשרד סוכן השטח גרי בצרחות.

"הפעם זה הדבר האמיתי. אני אומר לכם! קיבלנו דיווחים ממטיילים שראו יצורים חשודים במדבר נוואדה. כשהגענו למקום ראינו פיצוץ לבן מטורף. אני יודע שאתם לא תאמינו לי אבל הפעם יש לי הוכחות. משהו נשאר מאחור".

גרי שלף שקית ומתוכה הוציא הארדיסק שבור ומפוחם. הסוכנים החלו לחקור אותו ולנסות לשחזר את המידע שהיה בתוכו.

כעבור שבוע, מעבדת המידע בנאס"א הצליחה לשחזר את כל המידע, ולאחר סינון קפדני נמצאה רק תוכנה אחת שעל פי החוקרים עשויה ללמד אותנו משהו על מה שקרה במדבר נוואדה. על פי ההערכות, הישות שנצפתה ניסתה להשתמש בתוכנה כדי לשוב אל כוכב הלכת ממנו היא הגיעה, אך לא הצליחה.

הפנייה לחניכי מגשימים לא איחרה להגיע. האם תוכלו לעזור לנאס"א לפענח מה עושה התוכנה המסתורית, ויותר חשוב – **האם תוכלו לעזור לישות לחזור לכוכב שלה?**

**איך מתחילים?**

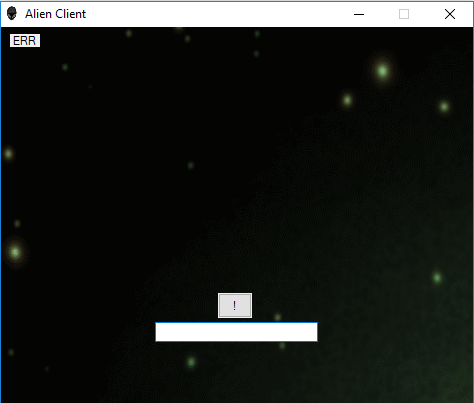
בתיקיית התרגיל בדרייב תוכלו למצוא את התוכנה שחולצה מההארדיסק המסתורי.

**הוראות?**

הוראות זה לחלשים. אתם עוד שניה בוגרי שנה א' של מגשימים! בהצלחה....

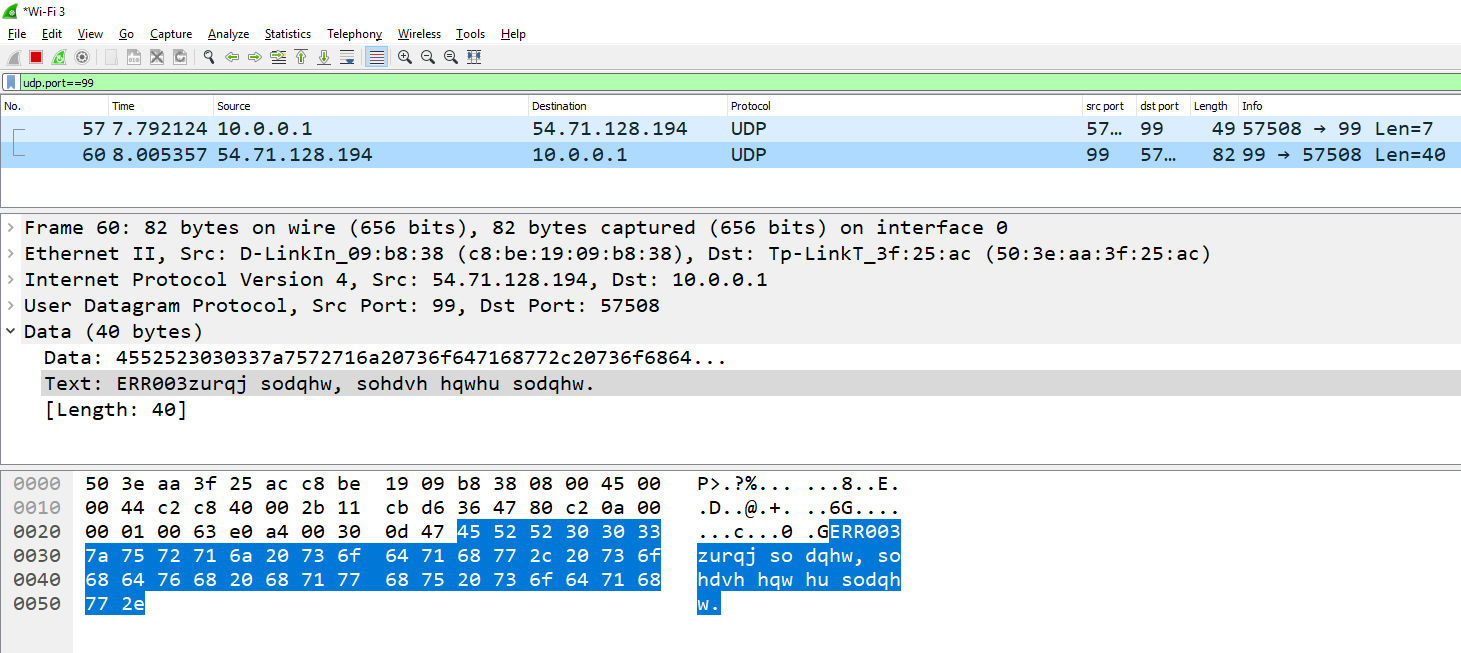
**פתרון מודרך:**

התוכנה נפתחת במסך הבא:



ניתן לראות שניתן לשלוח הודעות וכל מה שאנו כותבים מתקבלת תשובת ERR (שגיאה). בשלב הזה אנו יכולים רק להסניף ולראות מה קורה מאחורי הקלעים..

אם נסניף לפרק זמן קצר ונשלח הודעה, נגלה שהפורט של התוכנה הוא UDP 99. נוסיף פילטר udp.port==99 ונראה את כל התקשורת מול השרת.



אנו רואים הודעות שמגיעות מהשרת שיש להן מבנה די ברור:  
ERR003[text] אבל הטקסט הוא מג'וברש. אנו מקבלים את הודעת השגיאה הזאת כל הזמן:

**ERR003zrrnj sldnht! wo iqiwidth d fope bdcn, poedsh hnweu d yaoig vooau vyvthm poaqew qape.**

אז מתברר שיש פה הצפנה כלשהי, אבל לא ברור מה היא.

**פענוח ההצפנה**

הדרך הטובה ביותר שלנו לפענח את זה זה לראות דווקא את ההודעות היוצאות שלנו ולעשות איתן נסיונות. נשים פילטר על הודעות יוצאות לפורט 99 בלבד בווירשארק כדי להקל על המלאכה.

אם נקיש a נראה שיוצא בהסנפה d, מה שנראה כמו צופן הזזה פשוט (צופן קיסר) של 3 (אגב, שימו לב שהמספר 003 גם מופיע בהודעה וזה מפתח ההזזה). **אבל** אם נשלח abc אנחנו מקבלים dbf כך שזה לא צופן הזזה פשוט. אם נשלח aaa נקבל dad ואם נשלח aaaaa נקבל dadad.

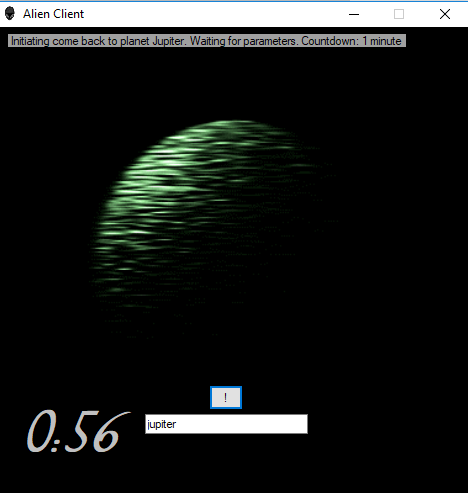
מכאן שאנחנו רואים צופן הזזה של 3 אבל רק של האיברים האי זוגיים, כלומר איבר במיקום הראשון, השלישי וכו'. כל שאר התווים נשארים בדיוק כפי שהיו.

אם כך עלינו לקחת את ההודעה שקיבלנו מהשרת ולהזיז את כל האותיות האי זוגיות 3 **אחורה**. אפשר לעשות את זה בצורה ידנית בשלב הזה, אבל זה יהיה מלא עבודה. לכן נכתוב סקריפט פייתון שעושה את המלאכה. שימו לב שתווים שאינם אותיות לא משתנים גם כן. אם נשתמש בסקריפט שלנו על הודעת השרת נקבל את ההודעה הבאה:

**ERR003wrong planet! to initiate a come back, please enter the name of one of the planets in the solar system.**

כעת אנו מבינים שעלינו לכתוב שם של כוכב לכת ממערכת השמש ואנחנו יכולים לנסות את כולם. אם נקיש Jupiter סיימנו את השלב הראשון בהצלחה.

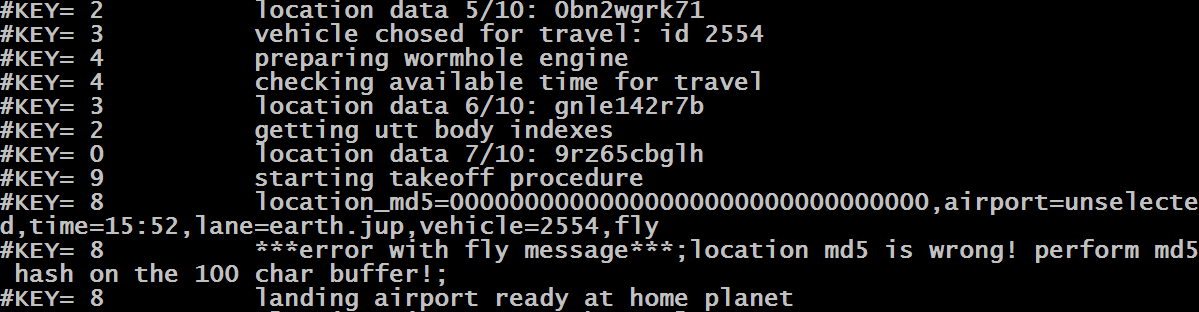
**הספירה לאחור**

התוכנה תיכנס למצב שמתחיל תהליך של החזרה לכוכב הנמשך בדיוק דקה, ונראה כך:

כעת אם נסניף נראה שיש מלא מלא הודעות שמגיעות אלינו. ההודעות מוצפנות באותו צופן הזזה אבל הפעם עם מפתחות שונים. המפתח מפורט בתחילת ההודעה. כלומר

ERR003 אומר שמפתח הזזה הוא 3 אותיות אחורה  
DAT005 אומר שמפתח ההזזה הוא 5 אותיות אחורה.

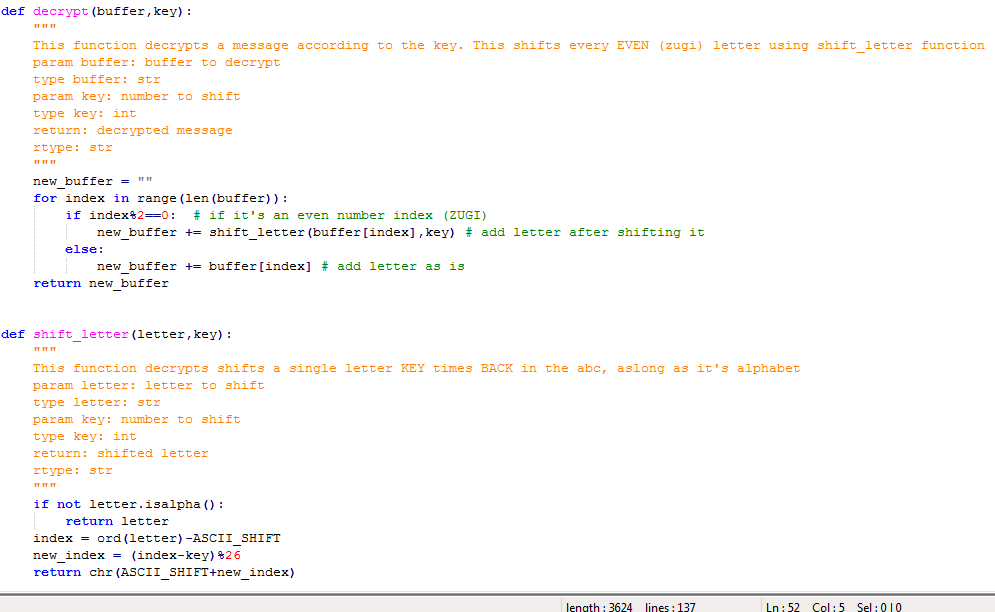
כעת מכיוון שיש לנו הצפה של הודעות, אנחנו לא יכולים לעבוד ידנית ונרצה לכתוב איזשהו סקריפט שמבצע את הפענוח בלייב. משהו שנראה כך:



הדרך הנוחה ביותר היא לכתוב סניפר עם סקאפי אשר מקבל את ההודעות ומפענח אותן בעזרת פונקציית הפענוח שכתבו קודם. ההבדל העיקרי מהשלב הקודם הוא שפונקציית הפענוח שלנו צריכה לתמוך בכל מפתח הזזה ולא רק ב-3 (זה שינוי קטן).

\* ניתן גם לכתוב סוקט UDP ששולח את הודעת Jupiter ואז מקבל תשובות. זה פתרון לגיטימי שיקבל ציון מלא, אם כי אנחנו מעדיפים שהם יעבדו עם סקאפי בתרגיל זה.

פונקציית הפענוח יכולה להיראות כך:



כעת אם נכתוב סניפר כזה נוכל לקרוא את כל ההודעות שנשלחות מהשרת. ההודעות חוזרות על עצמן בלופ. ההודעות המשעומותיות הן הודעות ה-Location בהמשכים המסומנים בירוק, והודעת השיגור המסומנת כחול. הודעת השגיאה של השרת מופיעה באדום.

ההודעות שנראה הן:

#KEY= 10 initializing parameters for take off procedure

#KEY= 3 location data 2/10: igtp28pk9u

#KEY= 10 selecting airport for takeoff

#KEY= 3 location data 3/10: 437pmp8qpy

#KEY= 2 requesting to clear air field

#KEY= 5 scanning air field

#KEY= 5 scanning air field (infra-red)

#KEY= 8 location data 4/10: 5p7jic6vne

#KEY= 8 synchronizing threads

#KEY= 8 location data 5/10: kmsmru6em6

#KEY= 1 vehicle chosed for travel: id 2554

#KEY= 8 preparing wormhole engine

#KEY= 4 checking available time for travel

#KEY= 8 location data 6/10: 7plv85lmal

#KEY= 10 getting utt body indexes

#KEY= 9 location data 7/10: ud74t2236c

#KEY= 8 location\_md5=00000000000000000000000000000000,airport=unselecte

d,time=15:52,lane=earth.jup,vehicle=2554,fly

#KEY= 8 \*\*\*error with fly message\*\*\*;location md5 is wrong! perform md5

hash on the 100 char buffer!;

בעצם הלקוח מנסה להתחיל תהליך של שיגור ולשם כך הוא צריך לקבל Location מהשרת ולשלוח אותו בבקשה השיגור. הלקוח שולח את ההודעה **בכחול**, אבל כל הזמן מקבל עליה שגיאה מהשרת.

כפי שאומרת הודעת השגיאה, הודעת השיגור יוצאת עם location md5 לא נכון. השרת רומז לנו שעלינו לחבר את כל הודעות ה-location (יש 10) לרצף אחד ולעשות לו תהליך של md5.

ברגע שיש לנו את זה ,עלינו ליצור הודעת FLY זהה לזו של השרת, ורק להחליף את השדה של location\_md5. וכמובן להצפין את ההודעה (לא משנה איזה מפתח. אפשר אפילו מפתח 0 ☺)

למשל כך:

**FLY008tokabiwn\_ml5=a6c1040009dk82645n5n6451bmfnf2d5,iizpwrb=vedala25.84,biue=15:52,linm=maztp.rux,depiklm=2554,fty**

כעת על שעלינו לעשות הוא לשלוח את ההודעה הזאת בזמן ה-Countdown בעצמנו. עלינו להקפיד שאנחנו משתמשים באותם פורטים כמו שהלקוח משתמש בהם, לבנות הודעת UDP ולשגר אותה.  
ניתן לעשות זאת באמצעות סקאפי או סוקט UDP – מה שיותר נוח לנו.

בסקאפי הקוד יראה כך כאשר MSG זו ההודעה המתוקנת (בצהוב למעלה). הפורט מקור משתנה בכל הרצה של התוכנה, כך שעלינו לקחת אותו מההסנפה **בלייב.**

HOST = "54.71.128.194"

SRCPORT = 2322

DSTPORT = 99

msg = Ether() / IP(dst=HOST) / UDP(sport=SRCPORT,dport=DSTPORT) / **MSG**

answer = sendp(msg)

אם כל הפרטים ששמנו יהיו נכונים, נראה שהצלחנו לתקן את השיגור. השרת יחזיר לנו הודעה שהשיגור הצליח והתוכנה תהיה שמחה כך:

