הטכניון מכון טכנולוגי לישראל
הפקולטה להנדסת חשמל ומחשבים
מבוא למערכות תכנה 044101

תרגיל בית 9 –"חדר בריחה – הכספת"

תרגיל הבית הזה הינו חדר בריחה וירטואלי. חדר הבריחה הזה הוא כספת, המכילה 6 שלבים – דלתות ממולכדות. כל סטודנט יקבל קובץ binary **אישי** להרצה, עם 6 שלבים. בכל שלב מצופה מהסטודנט להקליד מחרוזת קלט. אם הקלדתם את המחרוזת הנכונה, פתחתם את הדלת ועברתם שלב. אחרת תתפוצץ פצצה ויודפס "BOOM!!!" והתוכנית תסתיים. כשתענו נכונה על כל 6 השלבים תקבלו את המפתח ליציאה מהחדר.

שלב 1

הורידו את הכספת האישית שלכם. ישנם 2 קבצים:

1. **packer.exe**. זהו קובץ הרצה אותו אתם מתבקשים להריץ עם תעודת הזהות של אחד משני הסטודנטים שמגישים. חשוב מאוד לוודא שאין טעות בתעודת הזהות אותה אתם מכניסים! עלים להריץ התוכנית תחת לינוקס באופן הבא:

```
./packer.exe 123456789
```

כאשר 123456789 במקרה זה זו תעודת הזהות של אחד מהמגישים. התוכנית תייצר קובץ הרצה חדש ב-CWD בשם **vault.** זהו הקובץ הבינארי **האישי** אותו אתם צריכים לפרוץ בתרגיל בית זה.

2. **vault.c**. קובץ המקור עם פונקציית ה-main של הכספת.

שלב 2

המשימה שלכם היא לגלות את 6 המחרוזות לצורך פתיחת הכספת. היזהרו מהמלכודות. תוכלו להשתמש במגוון כלים על מנת לגלות כיצד פותחים את הדלתות. תוכלו להיעזר *בטיפים* בסוף המסמך. הדרך הטובה ביותר היא להשתמש ב-debugger: GDB, וללכת צעד-צעד על ה- disassembly של קובץ ההרצה.

השלבים שווים 23 נקודות כל אחד. מה שאומר שהציון המירבי האפשרי הוא 138. כמו בחדר בריחה, ניתן לבקש רמז תמורת 15 נקודות. בפורום "בקשת רמזים" כתבו את השלב עליו אתם מבקשים רמז ואיזה רמז מבוקש. תתקבל תשובה בהודעה פרטית. כל שלב נהיה קשה יותר בהדרגה, אז אל תחכו לרגע האחרון...

<u>כדי שלא תצטרכו להקליד מחדש כל סיסמא לדלת שדרכה כבר עברתם</u>, אתם יכולים להפעיל את הכספת עם ארגומנט של קובץ txt המכיל בכל שורה את הסיסמא לשלב שכבר גיליתם.

```
./vault part_solution.txt
```

במידה ואתם מצליחים, מומלץ לבצע static analysis כפי שנלמד בתרגול. זוהי הכנה מעולה לחלק מהשאלות במבחן. העזרו ב-single-step וב-break-points. הערך הלימודי כאן הוא שתלמדו היטב להשתמש ב-debugger, דבר שיהיה שימושי גם לשאר הלימודים והקריירה שלכם בכלל. זוהי מיומנות קריטית.

בסיום התרגיל עליכם להגיש את הקובץ solution.txt לכספת שלכם.

הוראות הגשה

1. עברו היטב על הוראות ההגשה של תרגילי הבית המופיעים באתר טרם ההגשה!
2. יש להגיש את הקובץ solution.txt (שימו לב לשם הקבצים עם lower case).

סיכום מפרט התרגיל:

סעיף	תיאור
נושא התרגיל	Machine Basics - debugging
הקבצי הנתונים	vault.c packer.exe
הקבצים שיש להגיש	solution.txt

בהצלחה!

רמזים

ישנם דרכים רבות "לפצח" את הכספת: אפשר לנסות לפצח ב-brute force בעזרת סקריפט, אך אינכם יודעים מה הם אורכי המחרוזות ולא כמה זמן ייקח לעבור על כל האפשרויות.. לכן, זו לא הדרך עליה אנחנו ממליצים. תוכלו למשל לנתח את הקובץ מבלי להריץ אותו, בדומה ל-static analysis אותו למדנו בתרגול.

היעזרו בכלים מההרצאות, התרגולים, והסדנאות –

- strings – כלי המדפיס את כל המחרוזות שישנם בקובץ בינארי
- objdump -t

ידפיס לכם את ה- symbol table של הקובץ הרצה. ה-symbol table מכיל את כל הפונקציות והמשתנים הגלובאלים בקובץ. את שמות הפונקציות והכתובות שלהם. *ניתן ללמוד רמז כלשהו משמות הפונקציות*

- objdump -d

יבצע disassembly לקובץ. תוכלו להסתכל על פונקציות ספציפיות. אבל הקריאה הזאת לא תתן לכם את כל התמונה: קריאות לפונקציית מערכת מופיעות בצורה מעט קריפטית (תמיד, זה לא חלק מהסוד של הכספת). לדוגמא, קריאת ל- sscanf יכולה להופיע כך:

```
8048c36:        e8 99 fc ff ff    call      80488d4 <_init+0x1a0>
```

כדי לגלות שמדובר ב-sscanf תצטרכו לעבוד עם gdb.

העזרו ב-man או ב-google כדי להבין מה הארגומנטים של פונקציות מערכת, לדוגמא: man sscanf.

תקציר פקודות שימושיות ב-gdb:

- disassemble – מציג assembly (disas != disa שימו לב)
- break (שם פונקציה או כתובת) – עוצר את התוכנית כשמגיעים לנקודה
- info b - מידע על כל ה-breakpoints המוגדרים
- disable – מבטל breakpoint עם המספר הנתון
- stepi – התקדם פקודה תוך כניסה לפונקציות
- nexti – התקדם פקודה ללא כניסה לפונקציה
- c – המשך עד לנקודת עצירה הבאה
- print [c ביטוי בשפת]

שימושי למשל כדי לבדוק משתנה מקומי או אזור זיכרון, שימו לב לבצע casting:

לדוגמא, print *(long*)pointer_to_long_var

- x – ניתוח הזכרון
- info registers – מציג את ערכי הרגיסטרים
- set disassemble-next-line on
- show disassemble-next-line

להצגת פקודת ה-assembly הבאה אחרי כל step.

<u>שאלות נפוצות</u>

- לא מבין מה עושה קטע קוד גדול ב-assembly? GDB
- צריך לדעת מה יש בכתובת מסוימת בזכרון? GDB
- רוצה לדעת איך כמה רגיסטרים משתנים לאורך הזמן? GDB
- לא יודע איך בכלל להתחיל? ראו רמזים
- פקודות GDB שימושיות? Google: GDB cheat sheet
- מה פקודת Assembly מסוימת עושה? שקפי הרצאות
- שימוש ב-GDB? סדנא או וידאו ללמידה עצמית
- מבנה הרצה של תוכנית? שקפי הרצאות
- פקודות וכלים שימושיים? תרגולים וסדנאות