תרגיל בית 3 בהנדסה לאחור

Hooking

236496

סמסטר חורף תשפ"א

הגשה בזוגות עד 4.1.2020 בשעה 23:59

לשאלות על התרגיל: idan.raz@campus.technion.ac.il

נא לרשום בנושא ההודעה "REW20-EX3" (ללא המרכאות).

הקפידו על הגשת קבצים עם השמות הנכונים כפי שמוגדר במסמך זה.

טעות בשם הקובץ תגרור ציון 0 ע״י הבודק האוטומטי בחלק הרלוונטי.

לפני ההגשה, ודאו שההגשה שלכם תואמת את הנחיות ההגשה בסוף התרגיל.

הנכם מוזמנים לצרף memes להגשה. סוג זה של בידור אהוד במיוחד על בודק התרגילים.

חלק יבש – איך תבצעו Hook כאשר...

לפניכם שישה מקרים בהם אנו מעוניינים לבצע hooking. עבור כל אחד מהתרחישים המתוארים הסבירו כיצד ניתן לבצע את ה hook. בכל אחד מהסעיפים יש לבצע hook ע"י מיקום jmp אחד לכל היותר **בתחילת** פונקציה קיימת. אם לא נאמר אחרת, ניתן להניח כי לפונקציות נקודת כניסה יחידה. בנוסף, אם לא נאמר אחרת, ניתן להניח כי לא קיים מצב בו ישנן שתי מסגרות של הפונקציות המדוברות על המחסנית. (בפרט הן אינן רקורסיביות ולא מכילות רקורסיות הדדיות).

1. נתונה הפונקציה poll המבצעת בדיקה מסוימת אל מול שרת. ידוע כי הפונקציה כותבת לזכרון גלובאלי מספר הקובע את מספר המילישניות שמחכה התכנית בין בקשה לבקשה. נרצה לבצע hook שישנה את התדירות בה הפונקציה ניגשת לשרת. בנוסף לפונקציה poll יש מספר רב של נקודות יציאה (ret).

2. נתונה הפונקציה mine המקבלת פרמטר יחיד ומבצעת חישוב מסובך במיוחד. נרצה לבצע hook כך שערך החזרה של mine עם הארגומנט x יהיה כערך החזרה שלה אם היינו מעבירים כאגומנט את $\bar{x}$ (היפוך הביטים של x). ידוע כי החישוב שמבצעת הפונקציה משפיע רק על המשתנים הלוקאליים שעל המחסנית והרגיסטרים. כלומר, אף תא אחר בזיכרון אינו מושפע מהחישוב. לפונקציה mine יותר מנקודת כניסה אחת. כלומר ישנן מספר נקודות בקוד המבצעות call לכתובות שונות בגוף הפונקציה mine. הנקודה בה מתחילה הפונקציה mine יכולה להשפיע על החישוב שמתבצע. ל mine ישנה נקודת יציאה יחידה. בסעיף זה בלבד מותר לבצע דריסה **נוספת** בסוף הפונקציה mine. לא ניתן להניח כי לאחר ה ret אין פקודות חשובות של פונקציה אחרת.

3. נתונה הפונקציה sendf השולחת הודעה על גבי socket. נרצה לבצע hook המצפין את ההודעה טרם תשלח. הפונקציה sendf מקבלת handle ל-socket, מחרוזת פורמט ומספר לא ידוע של ארגומנטים (בדומה ל printf).

4. נתונות שתי פונקציות connect, parse המבצעות חישובים בלתי תלויים אחד בשני. נרצה לבצע hook כך ששני החישובים יתבצעו במקביל. הפונקציות משפיעות על אזורי זיכרון שונים בתכנית ובפרט, אינן משפיעות האחת על השנייה. הפונקציות נקראות בזו אחר זו תמיד. כלומר, קריאה ל parse תמיד תופיע מיד לאחר קריאה ל connect. אם אחת הפונקציות מסיימת לפני האחרת אין חובה לחכות לשנייה (אך מותר) לפני המשך הביצוע של שאר התכנית. שימו לב כי שאר התכנית תלויה בשינויים שביצעו connect, parse לזיכרון.

5. נתונה הפונקציה calc המבצעת חישוב קשה להבנה. ידוע כי הפונקציה נקראת מספר רב של פעמים וניתן ללמוד עליה רבות מסדרת הערכים שמחזירה. נרצה לכתוב hook שידפיס את הערך שהפונקציה מחזירה בכל קריאה (logging). ידוע כי הפונקציה בודקת **במהלך ריצתה** את שלמותה, כלומר שתוכנה בזיכרון בזמן ריצה זהה למצופה. במידה וזה אינו המצב, הפונקציה מסיימת את ריצה מיד עם ערך חזרהלא מוגדר.

6. נתונה הפונקציה הרקורסיבית solve. נרצה לבצע hook כך שערך החזרה של הפונקציה הוא כפול מזה שהיית אמורה להחזיר. נניח כי לפונקציה ישנן נקודות כניסה ויציאה יחידות. מספר הקריאות הרקורסיביות בכל רמה בעץ אינו קבוע. שימו לב כי לא ניתן להניח שהפונקציה כתובה בצורה רציפה בזיכרון. התייחסו לשני המקרים הבאים: נרצה שערך החזרה שישתנה הוא של (א) כל אחת מן הקריאות הרקורסיביות (ב) של הקריאה החיצונית ביותר בלבד.

חלק רטוב – לשחרר את השודד

קבוצת המורדים שלנו ממשיכה לחולל טרור ב Catan Universe. לאחר שהחזרתם את הכבשים ללוח, השודד נכלא, ונעלם. עליכם לשחרר את השודד ולהשיבו אל המשחק. תחילה נבין את תפקידו של השודד במשחק.

השודד

השודד מופעל כאשר סכום הקוביות שהוטלו הוא 7. ראשית, שחקנים המחזיקים בנקודה זו ביותר מ שבעה קלפי משאב בידם, צריכים לוותר על מחצית מהם. לאחר מכן, מזיז השחקן שהטיל את הקוביות אל אריח כלשהו בלוח המשחק. כל עוד השודד נשאר על אריח זה, אריח זה לא יפיק משאבים, גם כאשר סכום הקוביות שהוטלו זהה למספר הרשום על האריח. בנוסף, לאחר מיקום השודד, יכול השחקן לבחור יריב אחד שלו התיישבות או עיר על אחד מקודקודי אריח זה ולשדוד אותו – לבחור קלף משאב באקראי מיד היריב ולהעבירו לידו שלו.

נראה כי היעלמותו של השודד אינה בהכרח דבר רע. אך הוא חלק בלתי נפרד מהמשחק, ותפקידכם הוא להשיב את הסדר על כנו. נוסף על כך, חיפוש קצר אודות השודד ברשת, מעלה כי השודד עבר הרבה, ואין הצדקה לקח שסיבלו יגדל:

“The truth is that the robber is a harmless fellow who is merely taken advantage of by the players, for the sake of their own benefit. He is pushed around from terrain hex to terrain hex, unable to escape his sometimes quite harsh destiny. What happens is that the robber doesn't always end up safely in the box after a game. Many reported a bitter end for the fellow in black.

For example, it seems that he was swallowed by dogs, which surely happened because he was touched with hands still sticky from clasping burgers and fries. Children, not knowing anything about his importance, apparently threw him into the toilet and sent him on a long journey through the sewer pipes. Some robbers experienced a hot finale in the fireplace, and others were simply lost”.

השבת השודד – חלק ראשון

על מנת להשיב את השודד עליכם לגשת לדף ה Tools באתר. דף זה נעול עם סיסמא. שמנו ידינו על תוכנה בשם **keygen.exe** אשר סופקה לכם. תוכנה זו מקבלת סיסמא לדף זה ומייצרת סיסמת גישה לאתר (כמו זו שהשתמשתם בה כשהתחברתם לאתר לראשונה). הפכו את התהליך על מנת להשיג את הסיסמא לדף זה. אם אתם מנסים להריץ את keygen.exe על המחשב האישי שלכם ולא על VM ייתכן ותצטרכו לכבות את DEP על מנת שהוא יוכל לרוץ.

השבת השודד – חלק שני

דף ה Tools מכיל מספר כלים. חלקם משמשים את משתמשי האתר בלבד. אחרים הם כלים בהם משתמש המשחק עצמו על מנת ליישם את חוקיו. נראה כי פלטפורמה זו משמשת את משתמשי האתר על מנת לשנות את הקבצים הללו ובכך להשפיע על המשחק. אחד הכלים שנמצא בדף זה הוא Client שנכתב כהוכחת התכנות להרצת פקודות על שרת המשחק. אחת הפקודות שהוא מאפשר היא DMSG. פקודה זו מאפשרת גישה להודעה האחרונה שהושארה ע"י משתמש אחר בשרת. ההודעה עוברת בצורה מוצפנת באמצעות הכלי Secure Pipe שגם הוא נמצא בדף זה.

עליכם להשיג את ההודעה האחרונה שהושארה **כשהיא מפוענחת** . עליכם לגרום לכך שclient.exe יפלוט את ההודעה האחרונה כאשר היא מפוענחת.

חלק מהכלים בדף זה ניתנים לעדכון בשיטות שונות, לחלקם ניתן להעלות גרסה חדשה של ממש. אחרים ניתן לעדכן באמצעות העלאת תוכנה חיצונית **המקבלת נתיב** לתוכנה המתעדכנת. כאשר אתם מעדכנים באמצעות תוכנה חיצונית יוצג לכם הפלט של תוכנה זו ושל כל תכנית שהיא מריצה באמצעות CreateProcessA.

השבת השודד – חלק שלישי

ההודעה שמצאתם מכילה מידע אודות הליך השחרור של השודד ומיקומו. סיימו את המשימה והחזירו את השודד למקומו בלוח המשחק. רוב הכלים בדף Tools בנויים ממעטפת לפקודות בשפת Python. תוכלו לראות קוד שחוזר בכולן ומטרתו להעביר פלט מפקודות Python אל התכנית. **שימו לב שכל הקבצים שאתם מעלים מבצעים את פעולתם בצורה שקופה.** אין ליצור תהליכים חדשים שלא היו אמורים להיווצר, ואין להשפיע על תכניות לטווח הארוך. למשל, הפיכת הפלט של תכנית רנדומלית לקבוע הוא אסור בהחלט!

הוראות הגשה

עליכם לתאר **במפורט** את הדרך שבה פעלתם על מנת להשלים את המשימה. בנוסף הגישו כל קובץ שרלוונטי לפתרון התרגיל שלכם. בקטגוריה זו נמנים בין היתר קבצי קלט עבור תכניות, ארגומנטים בשורת הפקודה, וקבצי קוד שכתבתם לכל מטרה שהיא.

Hooking

כאשר אתם מבצעים Hook עליכם לבצע אותו באחת מהשיטות שנלמדה בכיתה. עליכם להסביר בפתרונכם מדוע בחרתם בשיטה שבחרתם, איך בחרתם על איזו פונקציה לבצע את ההוק, וכיצד ביצעתם אותו בפועל. אל תחסירו דבר. מותר לדרוס אך ורק בתים הנמצאים בתחילת פונקציה כפי שביצענו בכיתה. כאשר אתם מבצעים Injection הגישו את קבצי המקור, וכן את ה-dll ו/או ה-injector המקומפלים **ודאו שאתם יוצרים Console Application עבור ה Injector שלכם.**

שמות קבצים

- קובץ dll לקובץ הרצה **file.exe** יקרא **File.dll**. קובץ המקור של ה-dll ייקרא **FileHook.cpp.** שם ה-Injector יהיה **FileInjector.exe**. את שלוש הקבצים האלו יחד עם כל קובץ אחר שנחוץ להרצה שימו בקובץ zip וקראו לו **File.zip**. **שימו לב לאותיות קטנות וגדולות!**
- קובץ הרצה המכיל Hook פיסי עבור קובץ הרצה **file.exe** ייקרא **file_hooked.exe.**
- לכל קובץ שלא נופל לאחת הקטגוריות הנ"ל ניתן לתת שם כרצונכם ולציין אותו בהסבר.
- ניתן להניח כי כל הקבצים שבהם נתקלתם בתרגיל נמצאים בתיקייה הנוכחית בעת ההרצה.

קבצים להגשה

- קובץ **dry.pdf** המכיל את הפתרונות שלכם לחלק היבש ואת תיאור הפתרון של החלק הרטוב.
- קובץ **keygen_rev.exe** וקוד מקור **keygen_rev.c/cpp** עבור השלב הראשון בחלק הרטוב.
- כל הקבצים הרלוונטיים לשחזור ה Hooks שביצעתם.

שומעים חופשיים

שומעים חופשיים בקורס אשר אין להם גישה לתרגיל מוזמנים לפנות אל עידן במייל, ולקבל עותק.

בהצלחה!