מבחן בהנדסה לאחור - 236496

סמסטר חורף תש"ף

מועד א׳, 20.02.2020

מרצה: עומר קדמיאל

מתרגל: טל שנקר

חלק שני

משך חלק זה של הבחינה: 3 שעות.

בחלק זה של הבחינה ישנן 2 שאלות. ענו על כולן.

בחלק זה 3 עמודים, כולל עמוד זה. וודאו שכל הדפים נמצאים.

מותר להשתמש בכל חומר עזר, אך אין לשוחח, להתייעץ, לקבל רמז, לתת רמז, או להיעזר בשום אדם אחר על המבחן, על תוכנו, והתשובות לשאלות. <u>יש לפתור את השאלות לבד.</u>

נא לכבות מכשירים סלולריים ושעונים חכמים.

בכל מקרה של שימוש בחומר חיצוני יש לתת הפניה אליו.

תוכנות מותרות: התוכנות אשר נלמדו בקורס. בפרט, אין להשתמש בכלי Decompilation!

ענו תשובות ברורות ככל האפשר. **<u>נמקו את כל תשובותיכם</u>**.

במידה שניתן לענות על שאלה במספר דרכים, **<u>ניקוד מלא יינתן לפתרונות קצרים ופשוטים</u>**.

<u>השאלות והקבצים המצורפים דומים אך אינם זהים לכל הסטודנטים</u>.

הגשה אלקטרונית באתר המקצוע עד השעה 17:00. יש להגיש קובץ ZIP (ואך ורק ZIP) ובו כל קבצי ההגשה. ההגשה כוללת בדיוק את הקבצים הבאים:

1. קובץ בשם part2.pdf עם התשובות בכתב <u>לכל</u> השאלות.
2. הקבצים אשר תצרו בשאלה 2.

<u>הערה לכל המבחן: בתרגילים הרטובים, אם אינכם מצליחים להגיע לפתרון שרץ בפועל, תארו במילים מה ניסיתם לעשות - זה עשוי לזכות אתכם בניקוד חלקי!</u>

בהצלחה!

שאלה 1 (25 נקודות) - יבש:

א. השוו בין hardware breakpoints לבין software breakpoints. הסבירו מהם היתרונות והחסרונות של כל אחת מהשיטות ותחת אילו תנאים כדי להשתמש בכל אחת מהן.

ב. נתונה תוכנית בה קיימת חולשת חריגה מחוצץ, אשר משתמשת DEP. ברצוננו לכתוב שרשרת ROP שתאפשר לנו להריץ shellcode בהקשר של תכנית זו. נמצאו בתוכנית ה gadgets מטה.
הסבירו מה מבצע Gadget1.

```
Gadget0:  mov  eax, [eax]
          ret
Gadget1:  rep  movsb
          xor  eax, eax
          ret
Gadget2:  mov  [ebx], eax
          ret
Gadget3:  add  reg1, reg2
          ret
Gadget4:  pop  reg
          ret
Gadget5:  mov  reg1, reg2
          ret
Gadget6:  jmp ebx
          ret
```

- הרגיסטרים reg, reg1, reg2 ניתנים לבחירתכם, וניתן להשתמש בהם מספר פעמים עם רגיסטרים שונים זה מזה.

ג. למרבה המזל, על אף שתוכנית זו לא מייבאת את פונקציות ניהול הזיכרון, הצלחנו למצוא את הפונקציה **GetJITMem** המקבלת כקלט גודל n, ומחזירה כתובת לאזור זיכרון בעל הרשאות כתיבה והרצה שגודלו n בתים. כתבו שרשרת ROP המנצלת את החולשה הקיימת בתוכנית ומשתמשת ב gadgets מהסעיף הקודם.

- ניתן להניח כי ה **shellcode שלכם קיים בכתובת** $[[ecx]]$ **וגודלו לכל היותר דף בודד**.
- על הפתרון להיות נתון בצורת מחסנית מיד לאחר ששרשרת ה ROP נקלטה ע"י התוכנית.
- פתרון שיכיל null bytes יקבל ניקוד חלקי בלבד.
- עליכם להתחשב ב Volatile Registers:

https://en.wikibooks.org/wiki/X86_Assembly/High-Level_Languages#Compilers

שאלה 2 (25 נקודות) – Hooking:

למבחן מצורף הקובץ secret.exe. התוכנה secret.exe היא תוכנה מסובכת, שלא ידוע לנו מה היא מבצעת. הצלחנו להבין כי התוכנה secret.exe מתנהגת בצורה שונה כאשר היא מורצת מתוך מנהל המשימות של Windows.

בשאלה זו תבצעו **Hook על ה-IAT** של מנהל המשימות. השתמשו בקובץ מנהל המשימות (TaskMgr.exe) הנמצא על מחשבכם בנתיב C:\Windows\System32. עליכם לבצע hook שיגרום להרצה של secret.exe ברגע שנפתחת התוכנית cmd.exe. **לצורך הפתרון עליכם לבצע Hook על ה IAT ע"י DLL Injection**. הפונקציה שעליה תבצעו Hook היא **NTQuerySystemInformation** שנמצאת ב ntdll.dll.

על התוכנית secret.exe לרוץ כאשר TaskMgr.exe הוא תהליך האב שלה. על התוכנית secret.exe לרוץ רק כאשר נפתח חלון חדש של cmd.exe, גם אם זהו לא החלון הראשון שנפתח. שימו לב שהתוכנה secret לא תרוץ עבור חלונות של cmd.exe שנפתחו לפני הרצת ה Hook שכתבתם. עליכם ליצור Injector.exe שיקבל כארגומנט בשורת הפקודה **רק** את הנתיב אל מנהל המשימות.

עבור שאלה זו עליכם להגיש את הקבצים הבאים:

- injector.cpp - קוד המקור של ה-injector.
- injector.exe - ה-injector המקומפל.
- hook.cpp - מכיל את הקוד עבור האתחול של ה-hook, ואת פונקציית ה-hook עצמה.
- hook.dll - ה-dll המוזרק.
- צרפו צילום מסך של המתרחש לאחר הרצה מוצלחת של ההוק ב-part2.pdf.

על שמות הקבצים ב-ZIP להיות זהים לאלו המופיעים ברשימה לעיל.