תרגיל שני

חולשות זיכרון

מועד הגשה –03.05.22

תכנות בטוח – 89509

הנחיות כלליות

- עבודה ביחידים בלבד.
- ניתן להיעזר במקורות אינטרנטיים, אך יש לציין באילו מקורות השתמשתם. בכל מקרה, אל הפתרון הסופי יש להגיע לבד.
- כל סקריפט אותו כתבתם לצורך פתרון התרגיל חייב להיות מצורף. בראש הסקריפט יש לכתוב בהערה מדוע כתבתם את הסקריפט ואיך הוא עזר לכם לפתרון התרגיל. הסקריפט צריך להיות מותאים לפייתון גרסה 3.8 ללא תלויות שאינן טריוויאליות.
- הפעילו שיקול דעת.

סעיפי התרגיל

1. נתונה תכנה מקומפלת – ex1.out, הקוד שלה – ex1.c, ושורת הקימפול שלה – ex1.sh
תנו לתכנה הרשאות root (מומלץ על ידי suid), על מנת שהפקודה setuid(0) תעבוד.
מצאו חולשה בתכנה, וגרמו לה לתת הרשאות לקובץ /etc/shadow והדפיסו אות תוכנו.
שימו לב –
   א. ניתן להניח שמנגנון ASLR כבוי.
   ב. כל קוד אסמבלי שמסייע לכם בפתרון התרגיל חייב להיות מצורף בנפרד בתור קובץ טקסט, עם תיעוד המסביר בדיוק מה הוא עושה.
   ג. ההשמשה שלכם יכולה להיות סטטיסטית, אבל במידה והיא לא עובדת תמיד, עליכם להסביר באילו מקרים היא לא עובדת וכיצד ניתן לשפר את ההסתברות. פתרונות שמניחים הנחות לא טריוויאליות עלולים להוביל להורדת נקודות.
2. נתונה תכנה מקומפלת בשם rop.exe, המיועדת למערכת הפעלה ווינדווז
   א. גרמו לה להדפיס את תעודת הזהות שלכם למסך.
3. (רשות) – גרמו לתכנה מסעיף 2 לפתוח מחשבון (calc.exe).

רמזים לתרגיל

1. רמזים לסעיף 1 – Buffer overflow
   א. על מנת לקמפל תכנות עם 32 ביט, יש להוריד gcc-multilib
   ב. התכנה צריכה לרוץ ב-suid או ב-sudo. על מנת להריץ עם suid הריצו
   sudo chown root a.out && sudo chmod +s a.out
   ג. צורף שלקוד המריץ את התכנה /bin/sh על מנת לתת לכם תחושה על מה זה שלקודים.
   השלקוד מצורף בקובץ shell.asm, לצד הייצוג ההקסה-דצימלי שלו. מומלץ להתחיל לפתור את התרגיל עם /bin/sh ורק אחרי זה להתקדם לכתיבת קובץ של ממש (על ידי open/write).
   ד. ה-system call של exit הוא 1, chmod הוא 15.
   ה. על מנת לבטל את ASLR, יש לכתוב את הערך 0 לקובץ
   /proc/sys/kernel/randomize_va_space
   ו. הדיבאגר של לינוקס, gdb, יכול לעזור להתמצא בקוד בהתחלה. הפתרון צריך לעבוד גם מחוצה לו.
2. רמזים לסעיף 2 – ROP
   א. חיפוש גאדג'טים הוא לא חלק מהתרגיל, ולכן הגאדג'טים בפונקציה a1 אמורים להספיק לפתרון מלא. מותר להשתמש בגאדג'טים אחרים אם רוצים.
   ב. הקוד שלכם אמור להיות פשוט –
      i. אתם צריכים לדרוס את ה-buffer והמשתנים הלוקאליים שאחריו, מומלץ באמצעות הערך 0x41.

ii. לכתוב בעזרת הגאדג'טים מהפונקציה a1 את המחרוזת שאתם רוצים לאיזור זיכרון כתיב.
iii. להדפיס בעזרת printf את איזור הזיכרון אליו כתבתם בסעיף הקודם.
iv. לצאת מהתכנה בעזרת exit.

ג. אם תקמפלו את התרגיל בעצמכם, הכתובות כנראה ישתנו מעט. ולכן, עדיף לא לקמפל את התרגיל בעצמכם אלא להשתמש בגרסה המקומפלת. Visual studio ו-WinDbg יודעים לדבג תכנות שורה-שורה אם יש להם את ה-pdb וה-source. עם זאת, כאשר מדבגים את ה-ROP עצמו, כדאי לדבג אופקוד אופקוד. גם בזה ויזואל סטודיו ו-WinDbg תומכים. יש גם ל-IDA דיבאגר, אבל הוא לא יודע לדבג sourceים.

ד. מאחר ו-ASLR כבוי, אתם יכולים לפתוח את התכנה ב-IDA ולחפש סימבולים. הכתובות שלהם לא ישתנו.

ה. אם אתם בכל זאת מקמפלים בעצמכם, שימו לב שאתם מכבים aslr ו-stack cookie. אל תכבו DEP. וזכרו שהפתרון הסופי צריך לעבוד על התכנה המקורית. מומלץ לקמפל ב-debug כדי שהפונקציה a1 לא תעלם.

נספח

1. How to debug and profile any EXE with Visual Studio - Visual Studio Blog (microsoft.com)
2. How to view the assembly behind the code using Visual C++? - Stack Overflow

הנחיות להגשת התרגיל

את התרגיל יש להגיש בסבמיט לקבוצה 01. עיקר ההגשה הוא קובץ PDF המסביר עבור כל אחד מסעיפי התרגיל כיצד פתרתם אותו באופן מפורט – דהיינו, עם צירופי סקריפטים, שלקודים, רופים וכו'.

בנוסף, כל שמות הקבצים שאתם מגישים צריכים להיות הגיוניים, ועם תווים אנגליים וסימנים בקידוד ASCII בלבד (בלי אותיות בעברית או בשפות שהן לא אנגלית)

לסעיף ב יש להגיש קובץ 2.bin ובו התוכן אותו אתם מעוניינים להעביר ל-argv[1]

לכל תרגיל צריך להיות מצורף קובץ בשם details.txt, המכיל את תעודת הזהות של המגיש, ושם המשתמש שלו בסבמיט.

שימו לב שהקובץ הוא קובץ טקסט פשוט מקודד ב-ASCII על מנת שיוכלו לקרוא אותו בקלות.
פורמט מבוקש –

name
id

במידה ויש שאלות בנוגע לתוכן התרגיל, נא להפנות לסגל הקורס בהקדם האפשרי.

הערות

טעות לעולם חוזר.

בהצלחה!
