NISRA Enlightened-Pwn

2024/08/28 betan@NISRA Enlightened

NISRA
ENLIGHTENED

Whoami

- 貝坦betan
  - FJU AIS 2
  - 2024 MFCTF 銀質獎
  - 2024 AIS3 最佳專題獎
  - 臺灣好厲駭第九屆學員
  - B33F 50uP
  - ~~AIS3專業蹭飯戶~~

NISRA
ENLIGHTENED

Today's Outline

- pwn簡介及相關應用
- Pwn2Own-Your printer is not your printer！
- Lab: Flag Shop
  - 小試身手
  - Integer Overflow
  - Overwrite Data
  - Buffer Overflow?

NISRA
ENLIGHTENED

資安倫理宣傳

本課程目的在提升學員對資訊安全之認識及資安實務能力，深刻體認到資安的重要性！所有課程學習內容不得從事非法攻擊或違法行為，所有非法行為將受法律規範，提醒學員不要以身試險。

NISRA
ENLIGHTENED

https://docs.google.com/presentation/d/1K4u-FwueFBGprh-m_kOQkM_wsp573Rv7LCnAvbz2e4w/edit#slide=id.g16608

大家都要當個好駭客喔

NISRA
ENLIGHTENED

簡報

沒有簡報 所以打開你們的共筆

PWN

講師：貝坦betan

時間：8/28 10:00~12:00

NISRA
ENLIGHTENED

Why CTF

- 知己知彼
- 審核能力
- 休閒娛樂
- $$
- ~~訓練自己的通靈能力~~

NISRA
ENLIGHTENED

pwn簡介及相關應用

NISRA
ENLIGHTENED

What's pwn?

- 對二進制可執行文件的 **漏洞利用**
- Binary Exploitation
- 特定輸入造成崩潰
- own⇒pwn
- get shell

What's pwn?

- 對二進制可執行文件的 漏洞利用
- Binary Exploitation
- 特定輸入造成崩潰
- own⇒pwn
- get shell

Pwned

NISRA
ENLIGHTENED

Why pwn

- 比賽
  - CTF
  - Pwn2Own
- 學習相關漏洞
- blue team
- red team
- $$

NISRA
ENLIGHTENED

還是沒有概念？

NISRA
ENLIGHTENED

來直接看別人打 pwn吧 (X

NISRA
ENLIGHTENED

pwnカ流程

pwnカ流程

1. 訊息蒐集

NISRA
ENLIGHTENED

pwn力流程

1. 訊息蒐集
2. 固體分析

NISRA
ENLIGHTENED

pwn力流程

1. 訊息蒐集
2. 固體分析
3. 漏洞挖掘

NISRA
ENLIGHTENED

pwnカ流程

1. 訊息蒐集
2. 固體分析
3. 漏洞挖掘
4. 漏洞利用

NISRA
ENLIGHTENED

pwn力流程

1. 訊息蒐集
2. 固體分析
3. 漏洞挖掘
4. 漏洞利用
5. 後門植入

NISRA
ENLIGHTENED

pwnの流程

1. 訊息蒐集
2. 固體分析
3. 漏洞挖掘
4. 漏洞利用
5. 後門植入

Pwned

NISRA
ENLIGHTENED

pwn力流程

1. 訊息蒐集
2. 固體分析
3. 漏洞挖掘
4. 漏洞利用
5. 後門植入

請選出存在
被駭風險的裝置









回報問題

驗證

NISRA
ENLIGHTENED

Your printer is not your printer !

- [2021-pwn2own]

Target	Cash Prize	Master of Pwn Points
HP Color LaserJet Pro MFP M283fdw	$20,000 (USD)	2
Lexmark MC3224i	$20,000 (USD)	2
Canon ImageCLASS MF644Cdw	$20,000 (USD)	2

Master of Pwn Standings

Contestant	Cash	Points
Synacktiv	$197,500	20
DEVCORE	$180,000	18
STARLabs	$112,500	12
Sam Thomas	$90,000	9
THEORI	$80,000	8
Bien Pham	$52,500	6.5
NCC Group	$60,000	6
trichimtrich	$40,000	5
Martin Rakhmanov	$40,000	4
Flashback	$33,750	3.75

Pwn2Own AUSTIN

ZERO DAY INITIATIVE

NISRA ENLIGHTENED

DEVCORE Hacking Printers

前年我們也在 Canon 及 HP 的印表機中發現了 Pre-auth RCE 的漏洞 (CVE-2022-24673 及 CVE-2022-3942) 及 Lexmark 發現漏洞 (CVE-2021-44734), 並在 Pwn2Own Austin 2021 中取得了 Canon ImageCLASS MF644Cdw、HP Color LaserJet Pro MFP M283fdw 及 Lexmark MC3224i 的控制權, 而成功獲得 Pwn2Own 中駭客大師 (Master of Pwn) 的點數。

NISRA
ENLIGHTENED

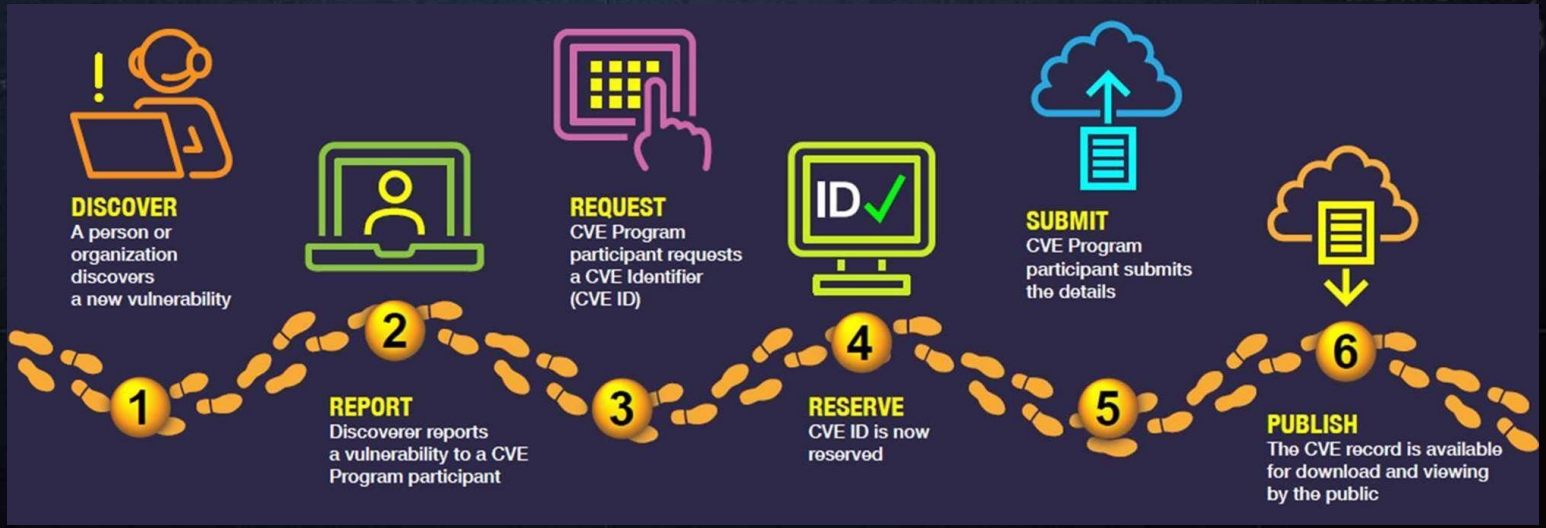
RCE

- Remote Code Execution
- 遠端程式碼執行

NISRA
ENLIGHTENED

CVE

- Common Vulnerabilities and Exposures
- 公共漏洞和暴露
- CVE-YYYY-NNNN



NISRA
ENLIGHTENED

為什麼是印表機

為何要研究印表機呢？

紅隊內網需求

過去我們團隊在執行紅隊演練過程中，**印表機普遍出現於現代企業 內網中**，幾乎都會有一台以上，但往往是 **被忽略**的一塊，也常常沒在更新。印表機本身也非常適合作為攻擊者的藏身處，通常很難被偵測出來。值得一提的是比較大型的企業也很有可能將其接上 AD，成為獲取機密資訊的入口。

' OR 1=1 --

TP/1.1 200 OK

SYN/ACK

";alert(1)//

--dbms=My
--level=3

> mysql -uroot -p

從何開始？

NISRA
ENLIGHTENED

pwn力流程

1. ~~訊息蒐集~~
2. 固體分析
3. 漏洞挖掘
4. 漏洞利用
5. 後門植入

NISRA
ENLIGHTENED

從Firmware開始分析

NISRA
ENLIGHTENED

下載舊版的Firmware

- 分析下載網址
- 根據規則下載到其他版本
  - V2.01
  - V4.02
  - V6.03
  - V9.03
  - V10.02

https://pdisp01.c-wss.com/gdl/WWUFORedirectTarget.do?id=MDQwMDAwNDc1MjA1&cmp=Z01&lang=EN

040000475205

Type	Ordinal Number	Version
Pdf,fimware …	Other model	Firmware version

分析Firmware

```
00000080: 3925 8607 7a14 a377 500b b031 0794 3f19  9%..z..wP..1..?.
00000090: b7ce 8750 461d 1107 728d 6700 0961 c324  ...PF...r.g..a.$
000000a0: 314c 7acf 0bab a5fa 16ad 076d 1ce7 b22a  1Lz........m...*
000000b0: acdb cbea 5a08 c2b6 f0fc 40d4 6f91 138a  ....Z.....@.o...
000000c0: ad06 fc85 3b9d 5eed 72ce 0351 c8c8 cde5  ....;.^.r..Q....
000000d0: 538f 830a d459 0952 ee5c e987 a56d 540a  S....Y.R.\...mT.
000000e0: 0b63 fad2 bac4 f2c4 3134 0c70 1ba3 aea9  .c......14.p....
000000f0: f032 65cf a260 b29a 0031 8a25 1a4d 783d  .2e..`...1.%.Mx=
00000100: 01b0 4199 b70f af96 ae9b ee62 ccb3 f390  ..A........b....
00000110: 429c 10c3 e704 1a21 e337 99f8 d2fa bca7  B......!.7......
00000120: d5b6 8060 a03d 1766 b179 0b28 04d9 191b  ...`.=.f.y.(....
```

分析Firmware

- 居..居然都被加密了

```
00000080: 3925 8607 7a14 a377 500b b031 0794 3f19  9%..z..wP..1..?.
00000090: b7ce 8750 461d 1107 728d 6700 0961 c324  ...PF...r.g..a.$
000000a0: 314c 7acf 0bab a5fa 16ad 076d 1ce7 b22a  1Lz........m...*
000000b0: acdb cbea 5a08 c2b6 f0fc 40d4 6f91 138a  ....Z.....@.o...
000000c0: ad06 fc85 3b9d 5eed 72ce 0351 c8c8 cde5  ....;.^.r..Q....
000000d0: 538f 830a d459 0952 ee5c e987 a56d 540a  S....Y.R.\...mT.
000000e0: 0b63 fad2 bac4 f2c4 3134 0c70 1ba3 aea9  .c......14.p....
000000f0: f032 65cf a260 b29a 0031 8a25 1a4d 783d  .2e..`...1.%.Mx=
00000100: 01b0 4199 b70f af96 ae9b ee62 ccb3 f390  ..A........b....
00000110: 429c 10c3 e704 1a21 e337 99f8 d2fa bca7  B......!.7......
00000120: d5b6 8060 a03d 1766 b179 0b28 04d9 191b  ...`.=.f.y.(....
```

逆向查看Firmware如何加密

- 找到疑似加密的函式
- 嘗試解密

```
1  char *__fastcall sub_41AB68A8(char *result, unsigned int size, char a3)
2  {
3    unsigned int i; // r3
4    unsigned int v4; // r4
5
6    for ( i = 0; i < size; ++i )
7    {
8      v4 = (unsigned __int8)(result[i] - (a3 + i) - 1);
9      result[i] = ~((2 * v4) | (v4 >> 7));
10   }
11   return result;
12 }
```

NISRA
ENLIGHTENED

解密Firmware

- 順利解密！

```
00000080: 3925 8607 7a14 a377 500b b031 0794 3f19  9%..z..wP..1..?.
00000090: b7ce 8750 461d 1107 728d 6700 0961 c324  ...PF...r.g..a.$
000000a0: 314c 7acf 0bab a5fa 16ad 076d 1ce7 b22a  1Lz........m...*
000000b0: acdb cbea 5a08 c2b6 f0fc 40d4 6f91 138a  ....Z.....@.o...
000000c0: ad06 fc85 3b9d 5eed 72ce 0351 c8c8 cde5  ....;.^.r..Q....
000000d0: 538f 830a d459 0952 ee5c e987 a56d 540a  S....Y.R.\...mT.
000000e0: 0b63 fad2 bac4 f2c4 3134 0c70 1ba3 aea9  .c......14.p....
000000f0: f032 65cf a260 b29a 0031 8a25 1a4d 783d  .2e..`...1.%.Mx=
00000100: 01b0 4199 b70f af96 ae9b ee62 ccb3 f390  ..A........b....
00000110: 429c 10c3 e704 1a21 e337 99f8 d2fa bca7  B......!.7......
00000120: d5b6 8060 a03d 1766 b179 0b28 04d9 191b  ...`.=.f.y.(....
```

```
00000520: 17f9 0120 10bd 0000 496e 7661 6c69 6420  ... ....Invalid
00000530: 4f70 6572 6174 696f 6e00 0000 4469 7669  Operation...Divi
00000540: 6465 2042 7920 5a65 726f 0000 4f76 6572  de By Zero..Over
00000550: 666c 6f77 0000 0000 556e 6465 7266 6c6f  flow....Underflo
00000560: 7700 0000 496e 6578 6163 7420 5265 7375  w...Inexact Resu
00000570: 6c74 0000 5349 4746 5045 3a20 4172 6974  lt..SIGFPE: Arit
00000580: 686d 6574 6963 2065 7863 6570 7469 6f6e  hmetic exception
00000590: 3a20 0000 10b5 0146 02a0 00f0 d9f8 0120  : .....F.......
000005a0: 10bd 0000 5349 4752 5452 4544 3a20 5265  ....SIGRTRED: Re
000005b0: 6469 7265 6374 3a20 6361 6e27 7420 6f70  direct: can't op
000005c0: 656e 3a20 0000 0000 10b5 0128 05d0 0021  en: .......(...!
000005d0: 03a0 00f0 bdf8 0120 10bd 09a1 f8e7 0000  ....... ........
000005e0: 5349 4752 544d 454d 3a20 4f75 7420 6f66  SIGRTMEM: Out of
000005f0: 2068 6561 7020 6d65 6d6f 7279 0000 0000   heap memory....
00000600: 3a20 4865 6170 206d 656d 6f72 7920 636f  : Heap memory co
00000610: 7272 7570 7465 6400 10b5 0021 02a0 00f0  rrupted....!....
00000620: 97f8 0120 10bd 0000 5349 4750 5646 4e3a  ... ....SIGPVFN:
00000630: 2050 7572 6520 7669 7274 7561 6c20 666e   Pure virtual fn
00000640: 2063 616c 6c65 6400 0b46 0146 1846 00f0   called..F.F.F..
```

Plaintext message

找Image base address

- 利用 rbasefind 工具找Image base
- 找到 base 為 0x40b0000

```
0x40b00000: 9800
0x40aff000: 8771
0x40b01000: 8176
0x40afe000: 8143
0x40afd000: 8026
0x40afa000: 7861
0x40afb000: 7796
0x40b02000: 7791
0x40afc000: 7756
0x40af9000: 7681
```

Image base address

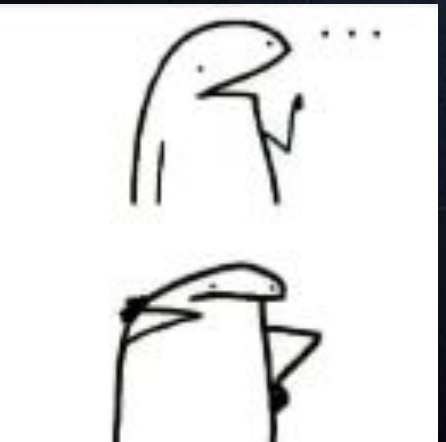


Image base address

- 地址
- 通常預設0x400000
- 可能會被占用？

A公司

你這次出差去住在 401套房

NISRA
ENLIGHTENED

B公司

你這次出差去住在 401套房


401套房
401

發生錯誤？

- loc_4489AC08 應為字串，但卻是被當成 code 區段
- 推測有些偏移

```
sub_41AE51A0(6, "%s: called, len=%d.\n", (const char *)&loc_4489AC08, *a3);
if ( a1[46] < 0x20u )
{
  sub_41AE51A0(5, "%s: auth header creation.\n", (const char *)&loc_4489AC08);
```

```
ROM:4489AC08 loc_4489AC08                    ; DATA XREF: sub_41C24A3
ROM:4489AC08                                 ; sub_41C24A38+40↑o ...
ROM:4489AC08                 ANDEQ           R0, R0, R0
ROM:4489AC0C                 ANDEQ           R0, R0, R0
ROM:4489AC10                 ANDEQ           R0, R0, R4
ROM:4489AC14                 ANDEQ           R0, R0, R2
ROM:4489AC18                 ANDEQ           R0, R0, R4
ROM:4489AC1C                 ANDEQ           R0, R0, R4
ROM:4489AC20                 ANDEQ           R0, R0, R1
ROM:4489AC24                 ANDEQ           R0, R0, R3
ROM:4489AC28                 ANDEQ           R0, R0, R1
ROM:4489AC2C                 ANDEQ           R0, R0, R4
ROM:4489AC30                 ANDEQ           R0, R0, R8
ROM:4489AC34                 ANDEQ           R0, R0, R4
ROM:4489AC38                 ANDEQ           R0, R0, R1
ROM:4489AC3C                 ANDEQ           R0, R0, R7
```

Should be strings

B公司

這次出差去住在 8620套房

NISRA
ENLIGHTENED


8620
8620

調整偏移量

- 透過已知名稱的函式和找到屬於他的名稱字串來做調整
- 最終 image base 為 0x40affde0

```
18  debugprintf(6, "%s: called, len=%d.\n", aReadsigneddata, *a3);
19  if ( a1[46] < 0x20u )
20  {
21    debugprintf(5, "%s: auth header creation.\n", aReadsigneddata);
22    v8 = (*(int (**)(void))(*(_DWORD *)(a1[2] + 76) + 68))();
23    debugprintf(6, "%s: lenRead %d.\n", aReadsigneddata, v8);
```

NISRA
ENLIGHTENED

新北市新莊區中正路 510號

你住哪

NISRA
ENLIGHTENED

徹底分析Firmware

Canon ImageCLASS MF644Cdw 架構如下 ：

- OS - DryOSV2
  - Customized RTOS by Canon
- ARMv7 32bit little-endian
- Linked with application code into single image
  - Kernel
  - Service

pwnカ流程

1. ~~訊息蒐集~~
2. ~~固體分析~~
3. 漏洞挖掘
4. 漏洞利用
5. 後門植入

NISRA
ENLIGHTENED

分析是否有保護

Canon 印表機本身並沒有任何記憶體相關的保護。

Protection

- No Stack Guard 防止不合理的輸入攻擊
- No DEP 防止存取非法的記憶體片段
- No ASLR 防止攻擊者精準定位系統或漏洞記憶體位址

沒有 Stack Guard、沒有 DEP 也沒有 ASLR，可以說是 hacker friendly！

NISRA
ENLIGHTENED

難度要開始加速了

你相信資安嗎

NISRA
ENLIGHTENED

Exploitation Step

1. 使用 BJNP 將我們的 shellcode 放到固定的已知位置。

2. 觸發 SLP 的 stack overflow 並覆蓋 return address

3. 跳到我們的 shellcode 上執行程式碼。

1. BJNP

- 印表機與其他設備溝通協定
- 由 Canon 自己所設計
- 過去也曾經有過 漏洞 💀
- 沒甚麼限制所以打 shellcode丟這裡就可以

NISRA
ENLIGHTENED

2. Service Location Protocol

- 一種服務發現協定
- 用於讓電腦快速找到印表機
- 其中存在stack overflow的漏洞

```
int parse_scope_list(...){
    char destbuf[36];
    unsigned int max = 34;
    parse_escape_char(...,destbuf,max);
}
```

NISRA
ENLIGHTENED

3. 順利Pwn掉

NISRA
ENLIGHTENED

NISRA
ENLIGHTENED

Lab Time:
Flag Shop

NISRA
ENLIGHTENED

FlagShop

你在一家有販賣 flag的商店中，flag的售價為 $10000，但你身上只有 $500，嘗試各種輸入以此來順利取得 flag吧！

NISRA
ENLIGHTENED

nc

nc(Netcat)能夠在網路通訊中執行各種任務。

- 建立簡單的TCP/UDP連接。
- 格式:nc [網址] [port]

  例如:nc 192.168.1.1 12345

NISRA
ENLIGHTENED

FlagShop 01

歡迎來到FlagShop！

嘿嘿你有辦法湊到足 夠的錢來買 flag嗎？

NISRA
ENLIGHTENED

FlagShop 01

提示：

前面有偷偷提到喔

FlagShop 01

Input:-10000

NISRA
ENLIGHTENED

Avoid

- 限制輸入大小

NISRA
ENLIGHTENED

Integer Overflow

NISRA
ENLIGHTENED

FlagShop 02

嘿嘿嘿 我這次限制了不能輸入負數

看看你還怎麼湊到足 夠的錢

NISRA
ENLIGHTENED

FlagShop 02

提示：

還記得int、long之類的是不是有個上限，如果超過上限的話會怎樣呢？

NISRA
ENLIGHTENED

What's Integer Overflow

- 超出暫存器最大數 值
- 數值發生預期外的錯誤

最大值

最小值

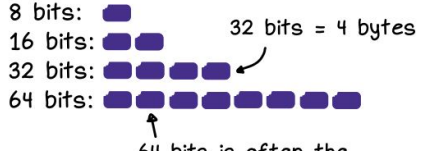
NISRA
ENLIGHTENED

Julia Evans
@b0rk

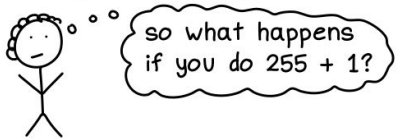
integer overflow

integers have a limited amount of space

The usual sizes for integers are:



the biggest 8-bit unsigned integer is 255



Going above/below the limits is called overflow.

ways overflow is handled

① wrap around

255 + 1 = 0

255 + 3 = 2

② raise an error

③ saturate ← this one is unusual

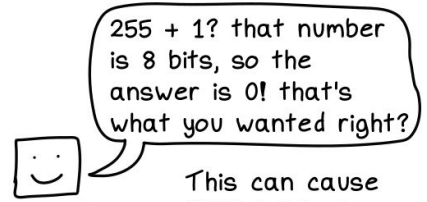
255 + 1 = 255

255 + 3 = 255

maximum numbers for different sizes

bits	signed	unsigned
8	127	255
16	32767	65535
32	~2 billion	~4 billion
64	~9 quintillion	~18 quintillion

overflows often don't throw errors



This can cause VERY tricky bugs.

some languages where integer overflow happens

Java/Kotlin C/C++ Rust SQL C# Swift Go R Dart

Some throw errors on overflow, some don't, for some it depends on various factors. Look up how it works in your language!

https://wizardzines.com/comics/integer-overflow/

FlagShop 02

Input:(2^31 - 500)/50

Avoid Integer Overflow

- 使用不同程式語言
- 限制輸入數字大小即可
- 檢查運算後的值

來實際玩看看吧

nc chall2.nisra.net 44010

NISRA
ENLIGHTENED

Overwrite Data

NISRA
ENLIGHTENED

FlagShop 03

嘿嘿嘿 現在flag變成贈品了

但這個資格可不是人人都有的喔

NISRA
ENLIGHTENED

FlagShop 03

提示:如果超過了會怎樣？

NISRA
ENLIGHTENED

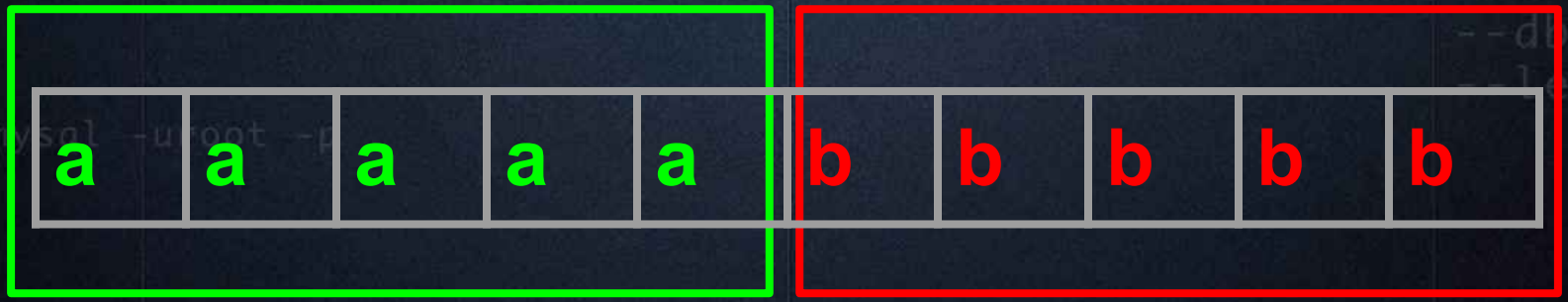
FlagShop 03

Input: ABCDEABCDEY

Why Overwrite

char a[5]

char b[5]

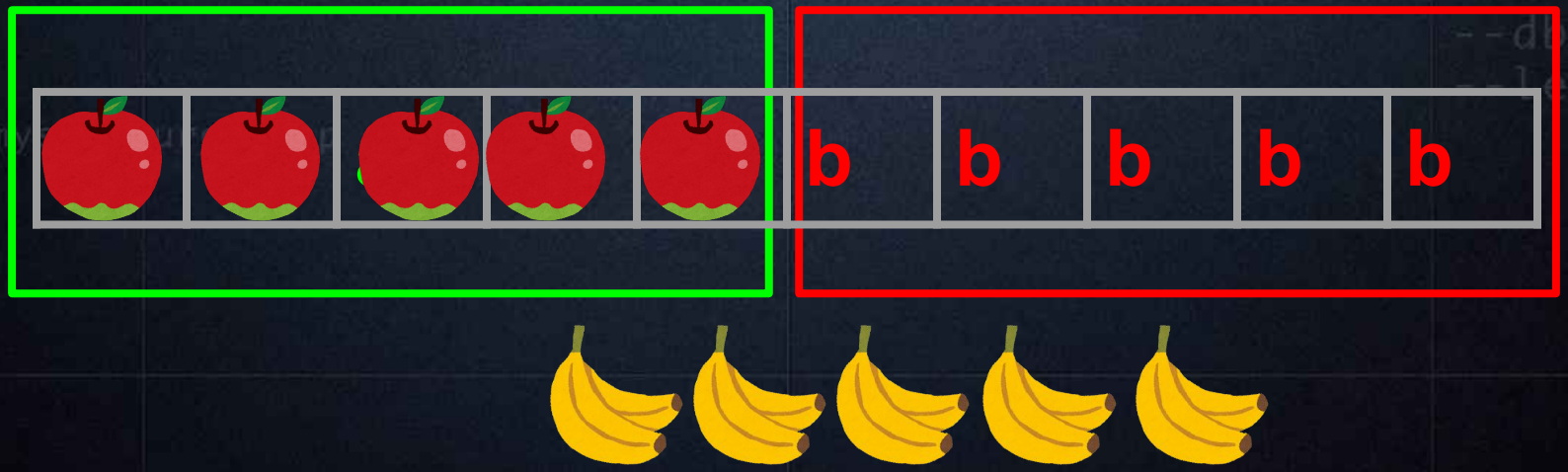


NISRA
ENLIGHTENED

Why Overwrite

char a[5]

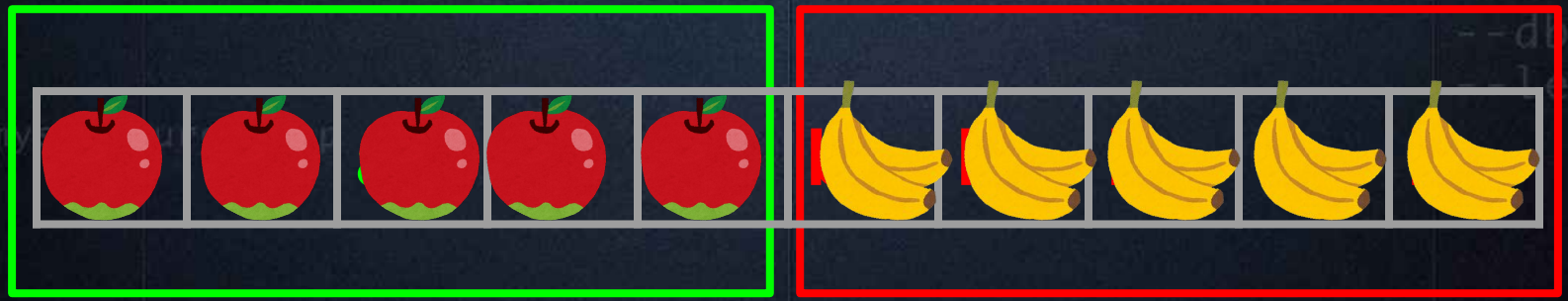
char b[5]



Why Overwrite

char a[5]

char b[5]

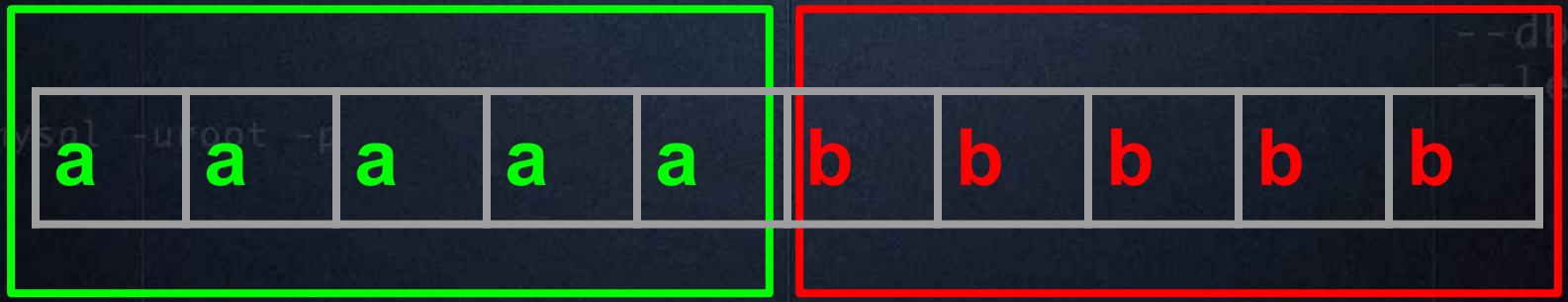


NISRA
ENLIGHTENED

Why Overwrite

char a[5]

char b[5]

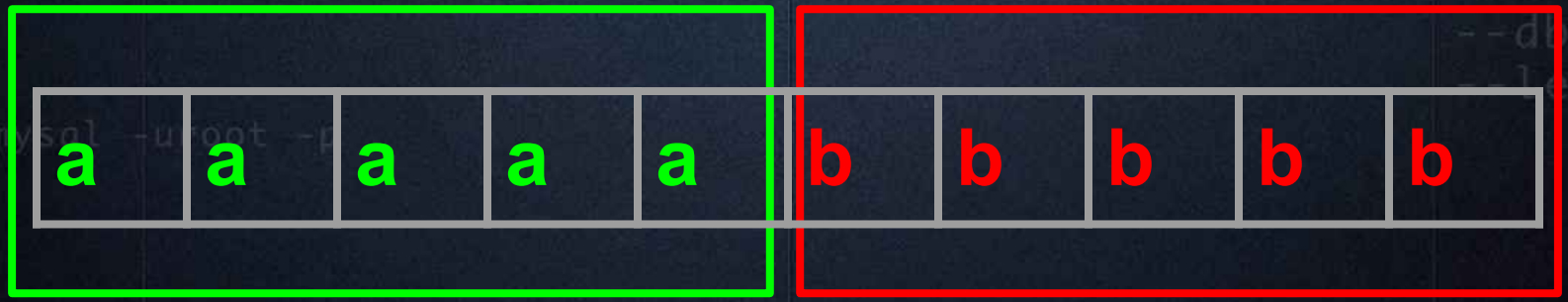


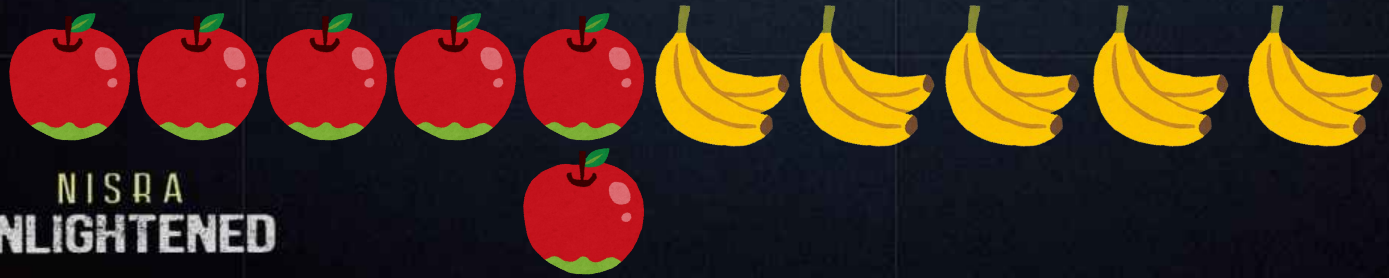
NISRA
ENLIGHTENED

Why Overwrite

char a[5]

char b[5]



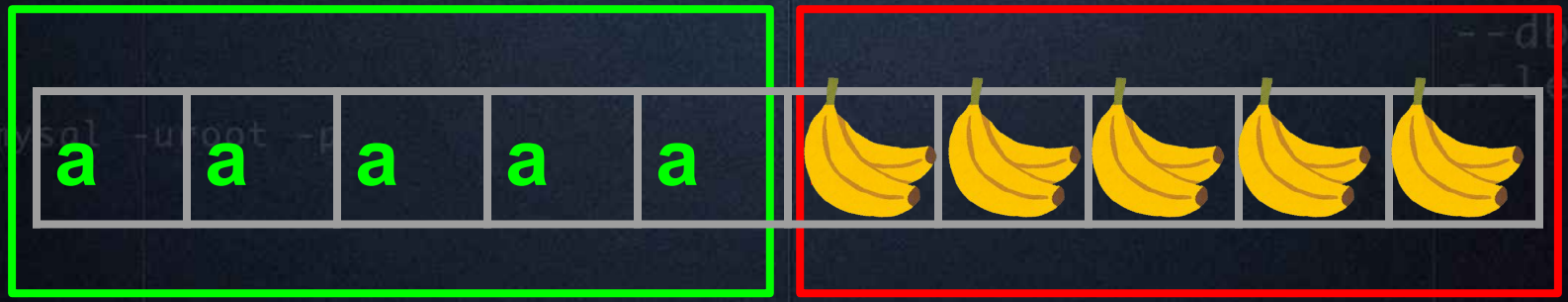


NISRA
ENLIGHTENED

Why Overwrite

char a[5]

char b[5]

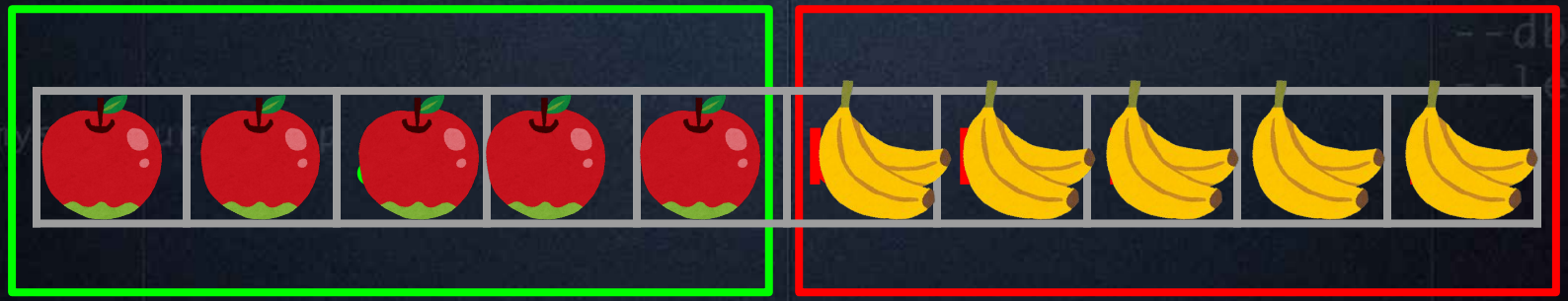


NISRA
ENLIGHTENED

Why Overwrite

char a[5]

char b[5]

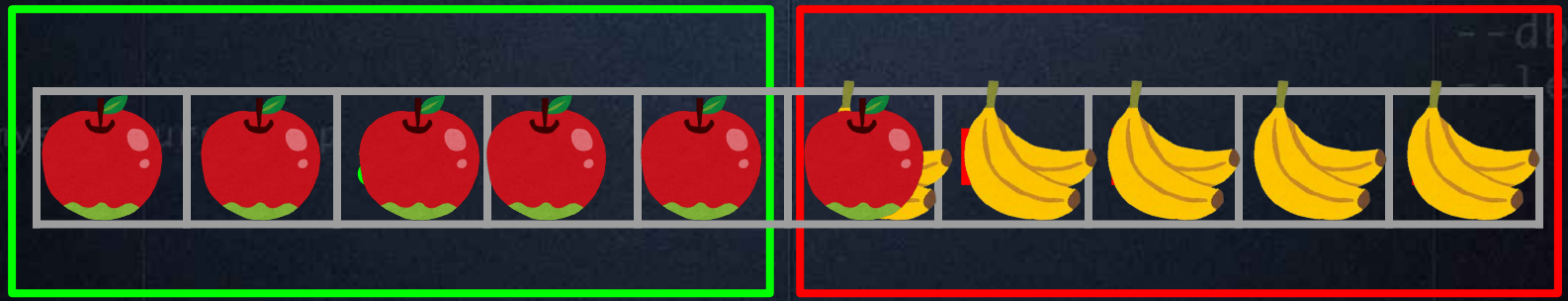


NISRA
ENLIGHTENED

Why Overwrite

char a[5]

char b[5]



Avoid Overwrite

- 避免使用危險的輸入函數 (ex:gets / scanf / strcpy / sprintf…)
- 使用有限制長度的輸入函數 (ex:"%4s")
- 使用相對安全的輸入函數 (ex:strncpy/strncat/snprintf)

NISRA
ENLIGHTENED

Buffer Overflow

- 與Overwrite Data概念相似
- 覆蓋資料 → 覆蓋地址

Buffer Overflow

- 執行的時候突然報錯？
- 直接退出、卡死？

```
Enter what you want to say:
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
You said: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa, relaying to the admin.
zsh: segmentation fault  ./lab3
```

NISRA
ENLIGHTENED

怎麼又開始抽象了

NISRA
ENLIGHTENED

Buffer Overflow

NISRA
ENLIGHTENED

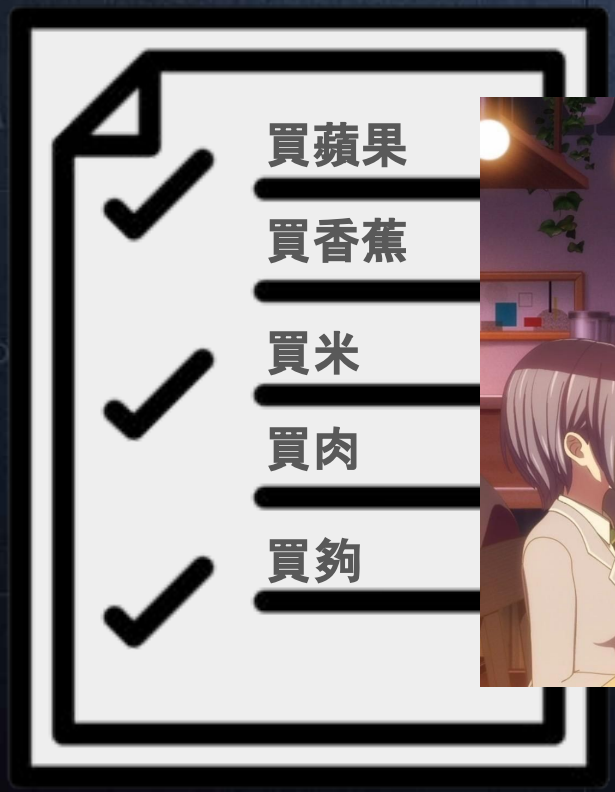
Buffer Overflow

NISRA
ENLIGHTENED

Buffer Overflow

NISRA
ENLIGHTENED

Buffer Overflow



NISRA
ENLIGHTENED

Buffer Overflow

NISRA
ENLIGHTENED

Buffer Overflow

NISRA
ENLIGHTENED

Buffer Overflow

NISRA
ENLIGHTENED

Buffer Overflow

- [x] 買蘋果
- [x] 買香蕉
- [ ] 買米
- [ ] 買肉
- [ ] 買夠 買夠 買夠 買夠 買夠 買夠 回到買夠

- [x] 從家中出門
- [x] 去到市場
- [x] 買清單內容
- [ ] 付錢
- [ ] 買夠 買夠 買夠 買夠 買夠 買夠 回到買夠

買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠
買夠 買夠 買夠 買夠 買夠 買夠

Buffer Overflow

Buffer Ov

Q&A

NISRA
ENLIGHTENED

Thank you!

NISRA
ENLIGHTENED