Forensics in CTF

oalieno

What is Forensics

- 在 CTF 比賽中，Forensics 可能會是以下幾種類型
  - File Format Analysis
  - Steganography
  - Memory Dump Analysis
  - Network Packet Analysis
  - …

What is Forensics

chtsecurity.com/service/m401

- Forensics 這個字的意思就是**鑑識**、**取證**，原本是用在法律案件中的犯罪鑑識、取證
- 如今，新型態的資安案件的出現，Forensics 也可以指資安事件的鑑識、取證
- 主要目的是在協助客戶緊急應變、入侵管道定位、受影響範圍評估及回復受駭系統

File Format Analysis

file

https://en.wikipedia.org/wiki/List_of_file_signatures

```
FILE(1)                    BSD General Commands Manual                    FILE(1)

NAME
     file -- determine file type
```

- 這個指令可以用來推測檔案類型
- 很多檔案都有固定的檔頭或叫 magic numbers
- 檔名和副檔名只是一個名字，可以隨便取，只能當參考

file

```
~ file hello                                                    ✔ 10007 08:30:50
hello: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamica
lly linked, interpreter /lib64/l, for GNU/Linux 3.2.0, BuildID[sha1]=eb
276cd9a7df84a611cecb2f8401a55b080f32b7, not stripped

~ file files.zip                                                ✔ 10013 08:31:47
files.zip: Zip archive data, at least v2.0 to extract

~ file test.py                                                  ✔ 10018 08:33:03
test.py: Python script, ASCII text executable

~ file icons-icomoon.png                                        ✔ 10024 08:34:31
icons-icomoon.png: PNG image data, 960 x 672, 8-bit colormap, non-interlaced
```

binwalk

- 是一個 firmware analysis tool
- 可以幫你挖出在檔案裡面的檔案
- CTF 比賽中常有通靈題會在檔案裡面藏檔案

```
oalieno@macbook ~ binwalk encrypt.zip                                    ✔ 10029 19:38:37

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, compressed size: 495, uncompressed size: 1284, name: solve.py
561           0x231           Zip archive data, at least v2.0 to extract, compressed size: 542, uncompressed size: 1287, name: encrypt.py
1171          0x493           Zip archive data, at least v1.0 to extract, compressed size: 16, uncompressed size: 16, name: cipher
1251          0x4E3           Zip archive data, at least v1.0 to extract, compressed size: 16, uncompressed size: 16, name: flag
1329          0x531           Zip archive data, at least v1.0 to extract, compressed size: 1, uncompressed size: 1, name: key
1772          0x6EC           End of Zip archive
```

binwalk

```
binwalk --dd=".*" picture.png
```

- dd 這參數可以指定要把哪種檔案格式的檔案抓出來
- .* 就是我全都要
- 這樣可以把所有 binwalk 認得的檔案抓出來

strings

- 幫你把檔案裡面的 printable 字元抓出來
- 可能可以看到一些有趣的東西

```
STRINGS(1)



NAME
       strings - find the printable strings in a object, or other binary, file
```

xxd & hexdump

- 把檔案用 hex string 的方式印出來

```
~ hexdump binary.dll | head
0000000 5a4d 0090 0003 0000 0004 0000 ffff 0000
0000010 00b8 0000 0000 0000 0040 0000 0000 0000
0000020 0000 0000 0000 0000 0000 0000 0000 0000
0000030 0000 0000 0000 0000 0000 0000 00f8 0000
0000040 1f0e 0eba b400 cd09 b821 4c01 21cd 6854
0000050 7369 7020 6f72 7267 6d61 6320 6e61 6f6e
0000060 2074 6562 7220 6e75 6920 206e 4f44 2053
0000070 6f6d 6564 0d2e 0a0d 0024 0000 0000 0000
0000080 b4e3 c0b0 d5a7 93de d5a7 93de d5a7 93de
0000090 c9dc 93d2 d5a6 93de c924 93d0 d5a5 93de
```

```
~ xxd binary.dll | head
00000000: 4d5a 9000 0300 0000 0400 0000 ffff 0000  MZ..............
00000010: b800 0000 0000 0000 4000 0000 0000 0000  ........@.......
00000020: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000030: 0000 0000 0000 0000 0000 0000 f800 0000  ................
00000040: 0e1f ba0e 00b4 09cd 21b8 014c cd21 5468  ........!..L.!Th
00000050: 6973 2070 726f 6772 616d 2063 616e 6e6f  is program canno
00000060: 7420 6265 2072 756e 2069 6e20 444f 5320  t be run in DOS
00000070: 6d6f 6465 2e0d 0d0a 2400 0000 0000 0000  mode....$.......
00000080: e3b4 b0c0 a7d5 de93 a7d5 de93 a7d5 de93  ................
00000090: dcc9 d293 a6d5 de93 24c9 d093 a5d5 de93  ........$.......
```

hexedit

- 在 hex string 格式下編輯檔案
- 做 binary patch 會用到

```
00000000   4D 5A 90 00  03 00 00 00  04 00 00 00  FF FF 00 00  B8 00 00 00  00 00 00 00  40 00 00 00  MZ..............@...
0000001C   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ....................
00000038   00 00 00 00  F8 00 00 00  0E 1F BA 0E  00 B4 09 CD  21 B8 01 4C  CD 21 54 68  69 73 20 70  ............!..L.!This p
00000054   72 6F 67 72  61 6D 20 63  61 6E 6E 6F  74 20 62 65  20 72 75 6E  20 69 6E 20  44 4F 53 20  rogram cannot be run in DOS
00000070   6D 6F 64 65  2E 0D 0D 0A  24 00 00 00  00 00 00 00  E3 B4 B0 C0  A7 D5 DE 93  A7 D5 DE 93  mode....$.......
0000008C   A7 D5 DE 93  DC C9 D2 93  A6 D5 DE 93  24 C9 D0 93  A5 D5 DE 93  C8 CA D4 93  A3 D5 DE 93  ............$.....
000000A8   C8 CA DA 93  A5 D5 DE 93  A7 D5 DF 93  8C D5 DE 93  C5 CA CD 93  A2 D5 DE 93  A1 F6 D4 93  ................
000000C4   A6 D5 DE 93  60 D3 D8 93  A6 D5 DE 93  58 F5 DA 93  A6 D5 DE 93  52 69 63 68  A7 D5 DE 93  ....`.......X.......Rich....
000000E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  50 45 00 00  ....................PE..
000000FC   4C 01 03 00  13 19 C2 40  00 00 00 00  00 00 00 00  E0 00 0E 21  0B 01 06 00  00 72 00 00  L......@.........!.....r..
00000118   00 0E 00 00  00 00 00 00  81 7A 00 00  00 10 00 00  00 90 00 00  00 00 80 18  00 10 00 00  .........z......................
00000134   00 02 00 00  04 00 00 00  00 00 00 00  04 00 00 00  00 00 00 00  00 B0 00 00  00 10 00 00  ................................
00000150   00 00 00 00  02 00 00 00  00 00 10 00  00 10 00 00  00 00 10 00  00 10 00 00  00 00 00 00  ................................
0000016C   10 00 00 00  E0 80 00 00  56 00 00 00  18 7D 00 00  50 00 00 00  00 90 00 00  10 06 00 00  ........V....}..P...........
00000188   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 A0 00 00  1C 05 00 00  00 00 00 00  ................................
000001A4   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................................
000001C0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................................
000001DC   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  2E 74 65 78  74 00 00 00  .....................text...
000001F8   36 71 00 00  00 10 00 00  00 72 00 00  00 04 00 00  00 00 00 00  00 00 00 00  00 00 00 00  6q.......r......................
00000214   20 00 00 E0  2E 72 73 72  63 00 00 00  10 06 00 00  00 90 00 00  00 08 00 00  00 76 00 00  .....rsrc.............v..
00000230   00 00 00 00  00 00 00 00  00 00 00 00  40 00 00 40  2E 72 65 6C  6F 63 00 00  8E 05 00 00  ............@..@.reloc......
```

fcrackzip

https://sc8log.blogspot.com/2017/05/blog-post_15.html

- 好用的工具，用來爆破 zip 密碼

用字典檔爆破

```
fcrackzip -u -D -p wordlist.txt file.zip
```

用小寫字母的組合爆破

```
fcrackzip -b -u -c a wordlist.txt file.zip
```

Steganography

Steganography

- Steganography 的中文叫圖像隱寫術
- 主要應用在惡意軟體為了規避資安人員的追查
- 比如惡意程式把設定檔藏在圖片裡，有機會藉此騙過資安人員

Stegsolve

http://www.caesum.com/handbook/Stegsolve.jar

- 好用的工具，包含了各種常見圖片隱藏的破解方式
- 是 GUI 的介面

```
java -jar Stegsolve.jar
```

exiftool

https://exiftool.org/

- 查看 metadata 的工具
- 比如照片大小、拍攝時間、拍攝地點等

```
~ exiftool lena.jpeg
ExifTool Version Number         : 11.70
File Name                       : lena.jpeg
Directory                       : .
File Size                       : 8.0 kB
File Modification Date/Time     : 2019:12:08 21:11:58+08:00
File Access Date/Time           : 2019:12:08 22:48:02+08:00
File Inode Change Date/Time     : 2019:12:08 21:12:02+08:00
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
```

LSB 隱寫術

- 每個像素都有 RGB 三個顏色，分別都用 1 bytes 來儲存總共 3 bytes
- 1 bytes 有 8 個 bit，改動最低位的 bit 對顏色改動不大
- 比如說把 181 改成 180 只有最後一個 bit 改變
- 人眼辨識不出這麼細微的改動
- 那我們就可以把資料藏在這些 bit 裡面

zsteg

- 專門解 LSB 隱寫術的工具
- Command Line Tool

```
zsteg -a test.png
```

```
~ zsteg -a out.png
b1,r,msb,xy             .. text: "+qRSt.e "
b2,r,lsb,xy             .. file: PGP         Secret Key -
b2,rgb,msb,xy           .. file: PGP         Secret Key -
b3,abgr,msb,xy          .. file: PGP         Secret Sub-key -
b4,g,msb,xy             .. file: PGP         Secret Key -
b4,rgb,msb,xy           .. text: " bDdVw rU13"
b4,bgr,msb,xy           .. text: "1\"@dftW\"Pu3"
b6,g,msb,xy             .. file: PGP         Secret Sub-key -
b8,rgb,lsb,xy           .. text: "-9Y}i\t{o"
b8,bgr,msb,xy           .. text: "BDl'|\t-+"
```

Python PIL

Python PIL

- Python Imaging Library (PIL)
- 圖像處理的函式庫，功能強大，且簡單易用
- 不過年久失修，後來有心人士創建 Pillow 並繼承了 PIL

```
pip install pillow
```

Example Usage

```
#!/usr/bin/env python3
from PIL import Image

im = Image.open("image.jpg")

print(im.format, im.size, im.mode)

pixel = im.getpixel((1, 1))
im.putpixel((1, 1), (255, 0, 0))
```

把 QRCOde 插進 LSB

```python
#!/usr/bin/env python3
from PIL import Image, ImageColor
import qrcode

qr_img = qrcode.make('Secret')
qr_pixels = qr_img.load()

img = Image.open('image.png')
for i in range(qr_img.size[0]):
    for j in range(qr_img.size[1]):
        pixel = img.getpixel((i, j))
        pixel = (pixel[0], (pixel[1] & 0b11111110) | int(qr_pixels[i, j] == 255), pixel[2])
        img.putpixel((i, j), pixel)

img.save('qr-hidden.png')
```