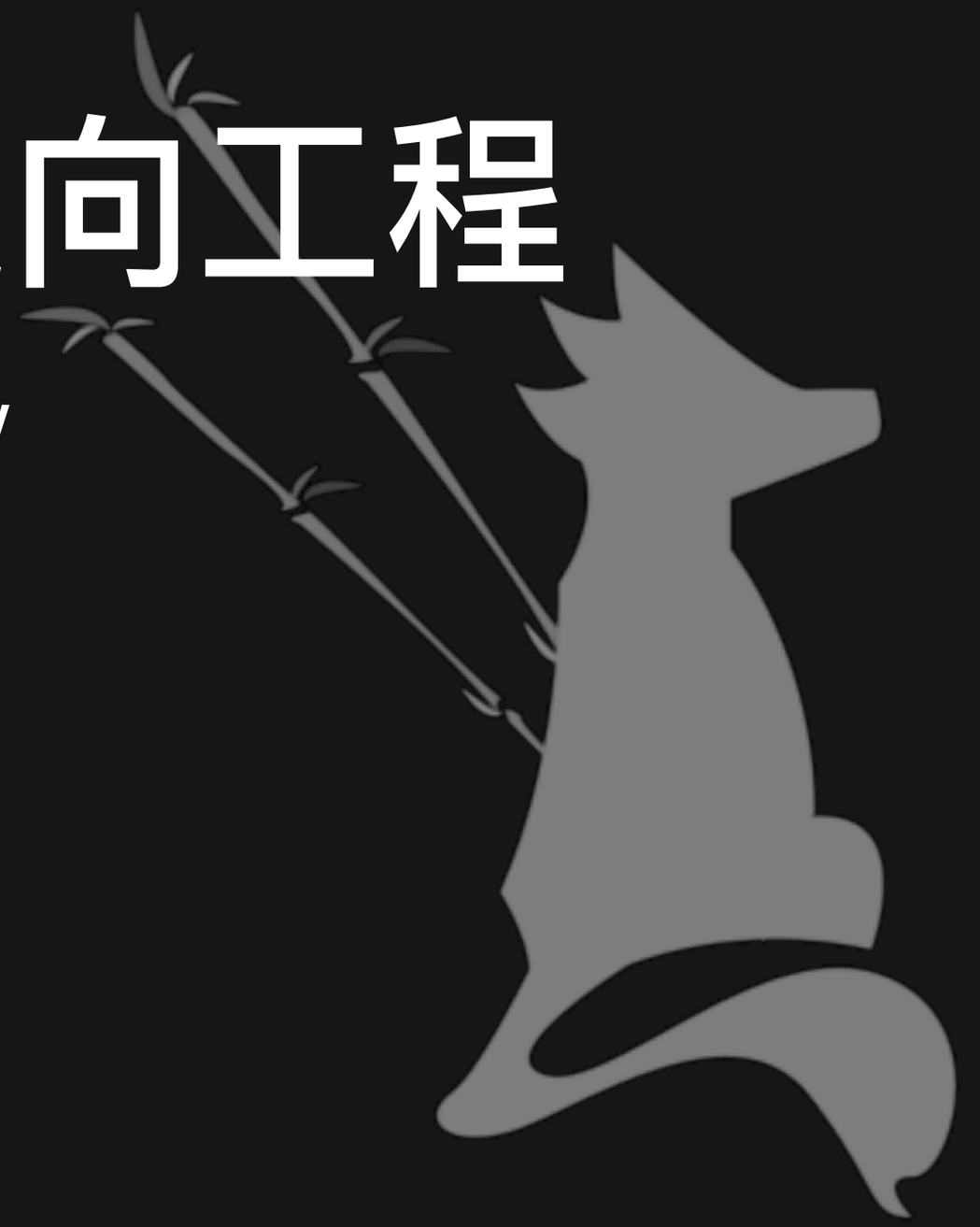
初探 CTF 逆向工程

ss8651twtw



Day2

- Reverse CTF 實戰
- Python 逆向分析
- Java 逆向分析

Day2

- Reverse CTF 實戰
- Python 逆向分析
- Java 逆向分析

Reverse CTF 實戰

❖ 工具

- Ghidra
  - https://ghidra-sre.org/
- IDA pro (if you have)

Reverse CTF 實戰

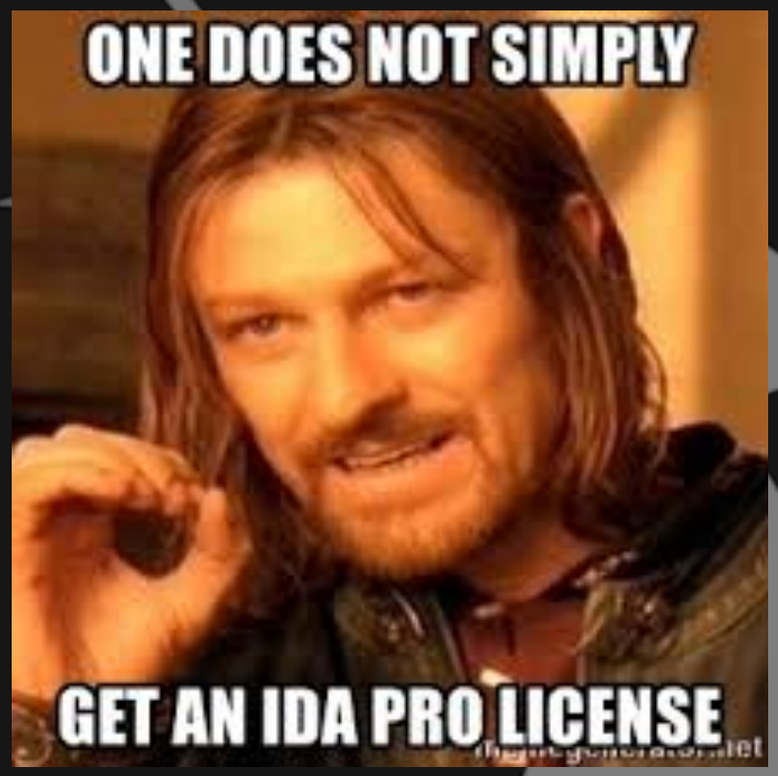
❖ Ghidra

- NSA 的逆向工程研究工具
- 免費開源

Reverse CTF 實戰

❖ IDA pro

- 強大好用的逆向分析工具
- 需付費使用



Reverse CTF 實戰

- IDA pro
  - 反編譯大法
    1. 在 functions window 點選想看的 function
    2. 按下 F5
    3. 完成!!!

Reverse CTF 實戰

- IDA pro
  - 字串表
    - 列出可視字串
    - View => Open subviews => String
    - Shift + F12

Reverse CTF 實戰

❖ IDA pro

- 交叉引用 Cross references (xrefs)

  1. 先點擊要標記型態的變數或 function

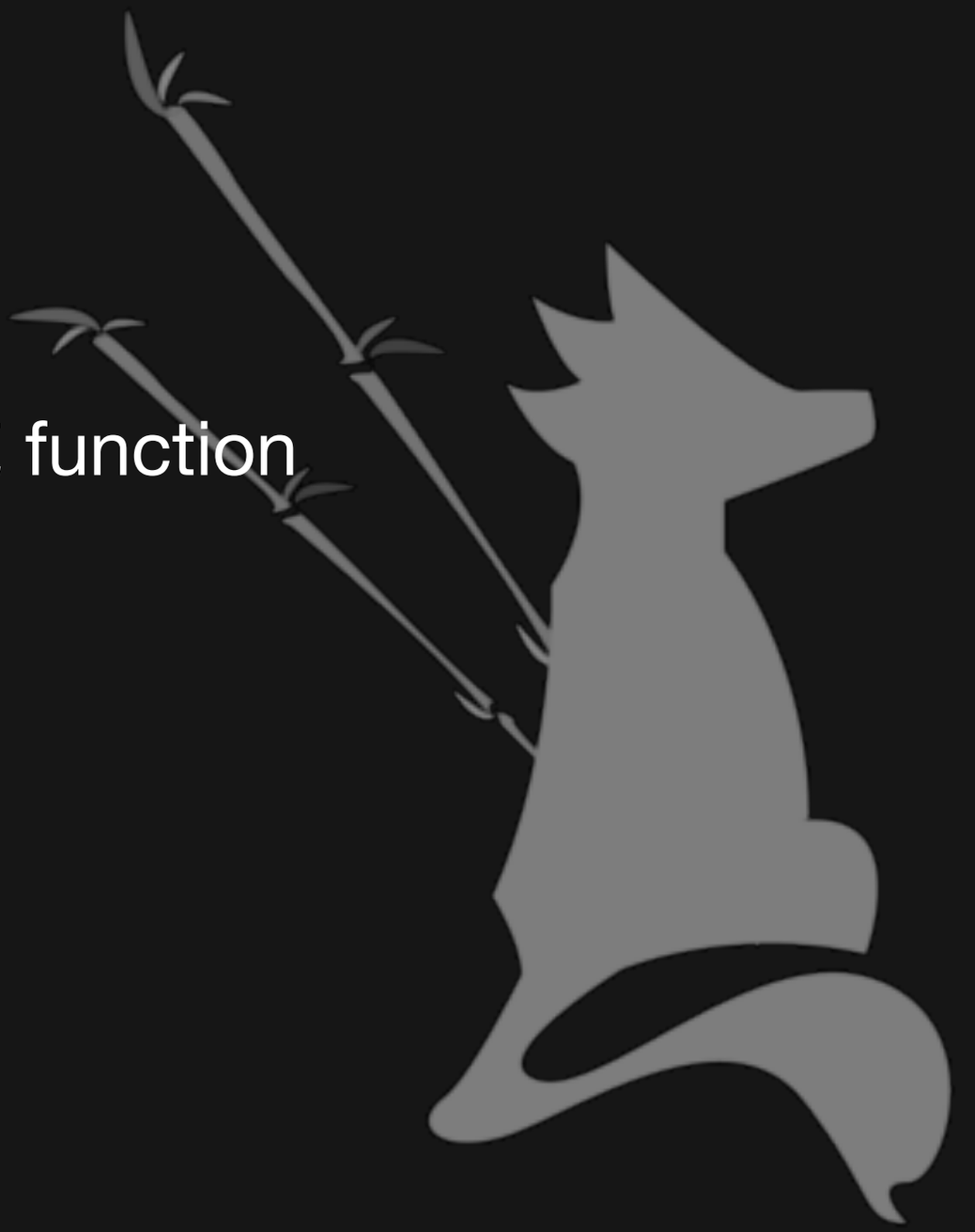
  2. 按下 x (Ctrl + j、Ctrl + x)

Reverse CTF 實戰

- IDA pro
  - 標記變數名
    1. 先點擊要命名的變數
    2. 按下 n
    3. 輸入新的變數名

Reverse CTF 實戰

- IDA pro
  - 標記型態
    1. 先點擊要標記型態的變數或 function
    2. 按下 y
    3. 輸入正確的型態



Reverse CTF 實戰

❖ IDA pro

- 標記 struct 結構

1. 切到 Structures 頁面

2. 新增 struct 並標記裡面的內容

```
; Ins/Del : create/delete structure
; D/A/*   : create structure member (data/ascii/array)
; N       : rename structure or structure member
; U       : delete structure member
```

Reverse CTF 實戰

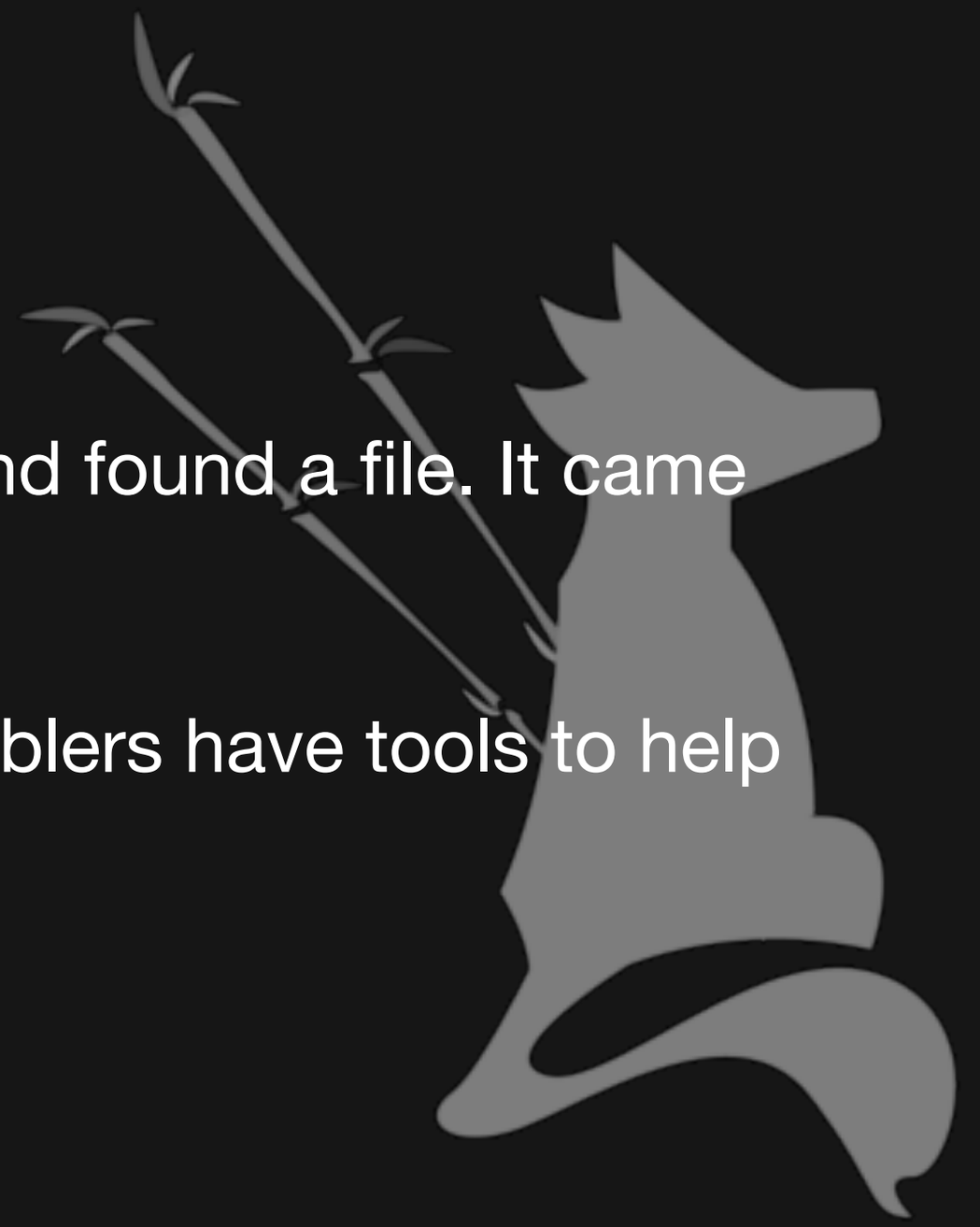
❖ IDA pro

- 常用快捷鍵們

https://www.hex-rays.com/products/ida/support/freefiles/IDA_Pro_Shortcuts.pdf

Reverse CTF 實戰

- IDA pro 🌰
  - picoCTF 2017 - forest
    - I was wandering the forest and found a file. It came with some string
    - Hints: A number of disassemblers have tools to help view structs



Reverse CTF 實戰

❖ IDA pro 🌰

- picoCTF 2017 - forest
  - https://github.com/0e85dc6eaf/CTF-Writeups/raw/master/PicoCTF%202017/Level%204/Reverse%20Engineering/Forest/forest
  - https://raw.githubusercontent.com/0e85dc6eaf/CTF-Writeups/master/PicoCTF%202017/Level%204/Reverse%20Engineering/Forest/string.txt

Reverse CTF 實戰

❖ IDA pro 🌰

- DefCamp CTF Qual 2019 - crack-me-username
  - Mach-O 64-bit executable x86_64
  - https://github.com/shinh/maloader

Reverse CTF 實戰

❖ IDA pro 🌰

- DefCamp CTF Qual 2019 - crack-me-username
  - https://trello-attachments.s3.amazonaws.com/5d6f5de4c0af3304440c5820/5d737557c11e880580ed7683/285a6b55f014f5602f10cb06aa45b486/crackme_username.out

Reverse CTF 實戰

❖ 實用小技巧

- LD_PRELOAD
  - 預先載入所指定的 shared object
  - 可以用來 hook function

Reverse CTF 實戰

❖ 實用小技巧

- LD_PRELOAD
  - 找出 function prototype
  - 撰寫一個 prototype 相同的 function
  - 編譯成 shared object
  - $ gcc -fPIC -shared hook.c -ldl -o hook.so

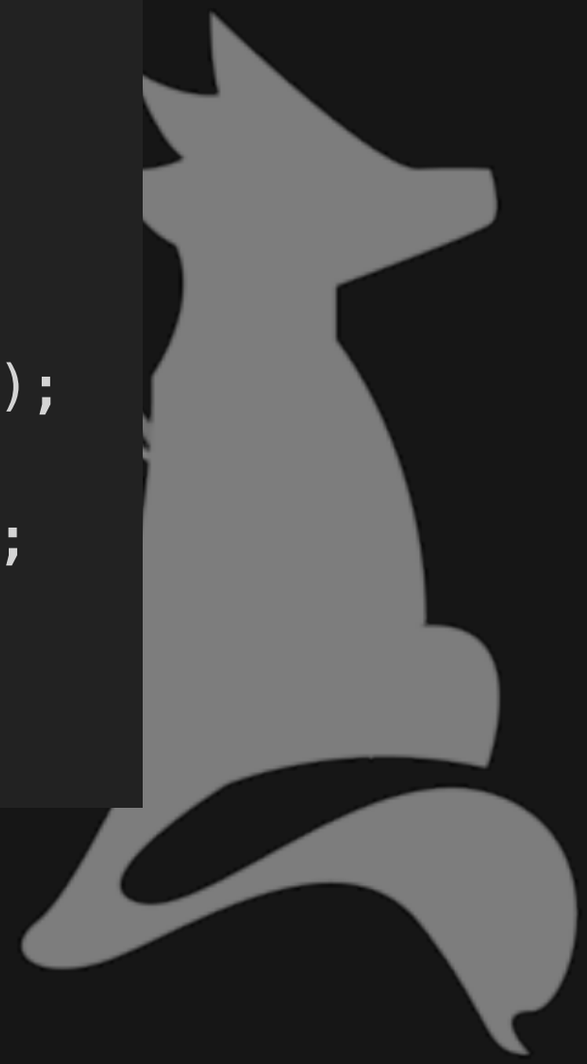
Reverse CTF 實戰

C

```c
#define _GNU_SOURCE

#include <dlfcn.h>
#include <stdio.h>

int rand(void) {
    int (*original_rand)(void);
    original_rand = dlsym(RTLD_NEXT, “rand");
    int out = original_rand();
    printf("=== %d ===\n", out % 0x4000000);
    return out;
}
```



Reverse CTF 實戰

❖ 實用小技巧

- LD_PRELOAD
  - 查看所使用的 shared object
  - $ ldd <binary>

Reverse CTF 實戰

❖ 實用小技巧

- LD_PRELOAD
  - 好用的 preload library 們
  - https://github.com/zardus/preeny

Reverse CTF 實戰

❖ 實用小技巧

- patch
  - 直接修改程式
  - IDA pro
  - hexedit

Reverse CTF 實戰

❖ 實用小技巧

- patch
  - nop
  - jmp <address>
  - push <address>; jmp [rsp];

Reverse CTF 實戰

❖ 刷題 🌰

- Reverse CTF
- AIS3 Pre-exam challenges

Day2

- ❖ Reverse CTF 實戰
- ❖ Python 逆向分析
- ❖ Java 逆向分析

Python 逆向分析

- 原始碼混淆
- Byte code 和 pyc 逆向

Python 逆向分析

- ❖ 原始碼混淆
  - ▸ 代換變數名稱
  - ▸ 加入無意義的程式碼
  - ▸ 用其他方式包裝真正執行的程式
  - ▸ 用一些看起來很難很複雜的操作完成一個簡單的功能

Python 逆向分析

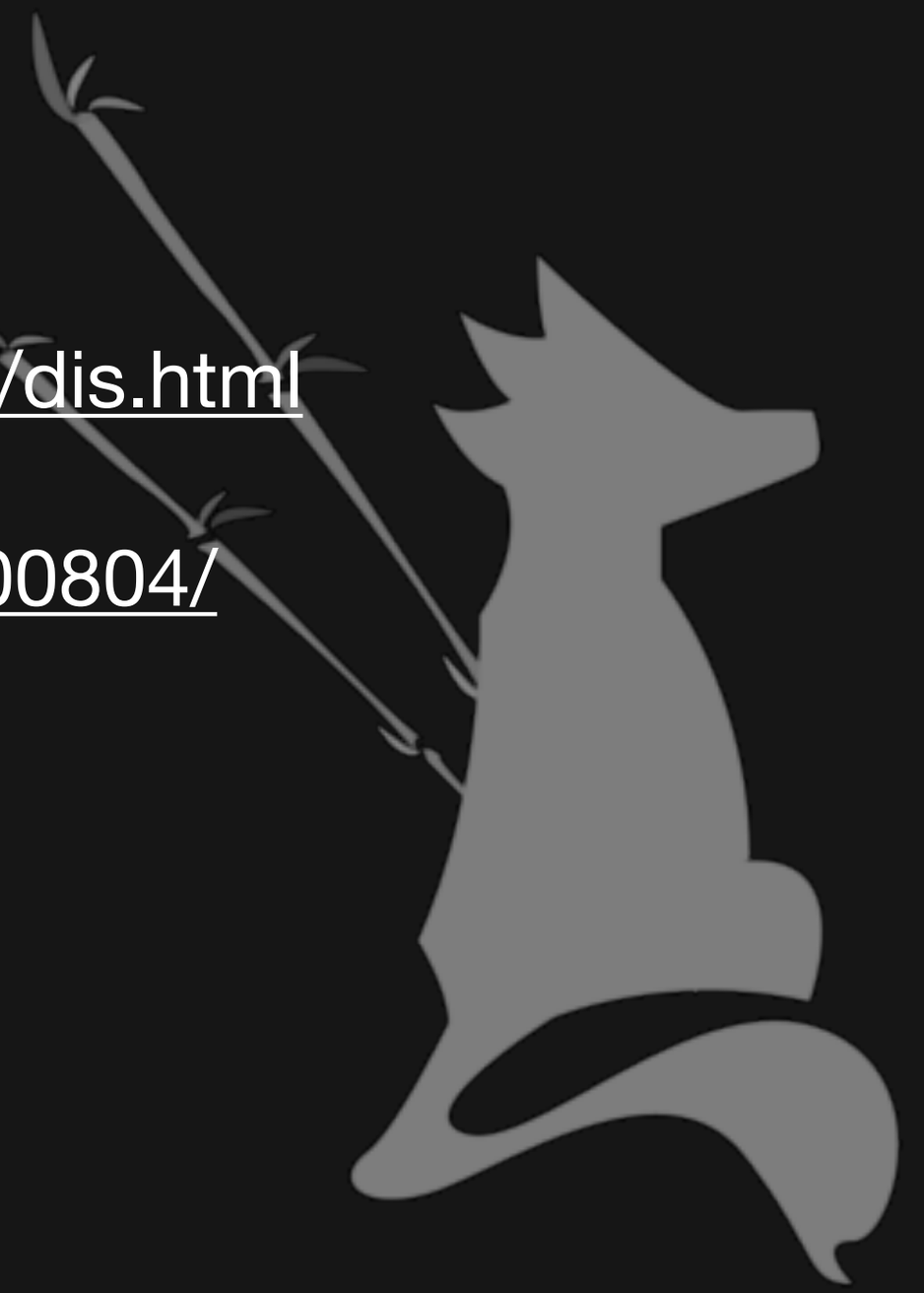
❖ 原始碼混淆 🌰

- EasyCTF IV - soupstitution_cipher
- EasyCTF 2017 - useless-python

Python 逆向分析

❖ Byte code 和 pyc 逆向

- https://docs.python.org/3.5/library/dis.html
- https://nedbatchelder.com/blog/200804/the_structure_of_pyc_files.html
- https://tool.lu/pyc/



Day2

- ❖ Reverse CTF 實戰
- ❖ Python 逆向分析
- ❖ Java 逆向分析

Java 逆向分析

- 原始碼混淆
- Byte code 逆向

Java 逆向分析

❖ Byte code 逆向

- https://docs.oracle.com/javase/specs/index.html
- http://www.javadecompilers.com/

Java 逆向分析

- Byte code 逆向 🌰
  - picoCTF 2017 - Coffee

Thanks for listening!