資安禁術 - 逆向工程地獄試煉

葉東逸（**Kazma**）
Speaker @ B33F 50μP

Whoami

- Common Nickname: Kazma
- CTF @ B33F 50UP
- TSCCTF 2024: Chief Organizer
- Taiwan Security Club: Founder
- NCKUCTF: Founder/President
- INTERN @ Telecom Technology Center

@kazma.tw

HITCON CTF 2023

UTC 09/08 14:00 ~ ~~09/10 14:00~~ 09/10 15:00

Due to the instability at the start of the competition, we've decided that we'll extend the competition by 1 hour.

Discord: https://discord.gg/ypqCsNxHmc

Awards in Final

1st place	2nd place	3rd place	Taiwan Star
$10,000 USD	$5,000 USD	$2,000 USD	$1,000 USD
+ Pre-qualification for DEF CON 32 CTF Final	-	-	Special award for Taiwan team

競賽說明

預賽

- 競賽為 48 小時的線上 Jeopardy 形式 CTF
- 競賽採取積分累計制，依分數高低進行排名
  同分者，依最後一次正確提交的時間判定
- 每道題目的分數將會根據解題隊伍數即時進行動態調整
- Flag 形式為: `hitcon{printable ascii+}`

決賽

Contest Information

Quals

- Contest is online jeopardy style CTF for 48 hours.
- Solving each challenge will get certain points. Teams will be ranked by total points.
  If the score are the same, the ranking will be determined by the time of last correct submission.
- The points of each challenge will be dynamically adjusted based on the number of solved teams.

Full Chain - The Blade

- Introduction
- Warm up
- Description
- Hell

Introduction


老師教的內容
我得到的

Warm up: flag_checker0

Warm up

Challenge | 0 Solves

Flag_checker0

50

Let's do some warm up before the fight. Please encapsulate the flag within the `NCKUCTF{}` format.

flag_checker0

Flag

Submit

Description

Description - link

Full Chain - The Blade [234pts]

A Rust tool for executing shellcode in a seccomp environment. Your goal is to pass the hidden flag checker concealed in the binary.

https://github.com/hitconctf/ctf2023.hitcon.org/releases/download/v1.0.0/blade-4c2ff1b60902623f702f0245a6a9ea0e71eeb385

(Hey, this is my first Rust project. Feel free to give me any advice and criticism 😃)

Author: wxrdnx

40 Teams solved.

#	Team	Submit Time
1	Super Guesser	2023-09-08 16:16:27 UTC
2	organizers	2023-09-08 16:20:17 UTC
3	mode13h	2023-09-08 18:42:57 UTC

Contest is over.

Description

> A Rust tool for executing shellcode in a seccomp environment. Your goal is to pass the hidden flag checker concealed in the binary.
> (Hey, this is my first Rust project. Feel free to give me any advice and criticism 😃)

Rust

Rust

- 安全性：自動管理記憶體安全，預防懸掛指針和緩衝區溢出等問題。
- 速度：與 **C**和**C++**相當的執行速度，沒有運行時垃圾收集的開銷。
- 並行性：設計上保證了資料競爭的安全性，使多線程程式設計更安全、更容易。
- 系統編程：適合開發系統軟件、嵌入式系統、操作系統等底層應用。
- 跨平台：支持多平台開發，包括 **Linux**、**macOS**、**Windows** 等。
- 生態系統：擁有強大的包管理工具 **Cargo** 和豐富的庫支持，便於開發和共享代碼。
- 學習資源：提供豐富的學習資料和活躍的社群支持，幫助新手上手。

Shellcode

Shellcode

- 直接執行：以機器碼形式存在，能 夠被操作系統直接執行。
- 目的：用於在軟件漏洞被利用後獲取系統控制，如命令行殼。
- 高效緊湊：設計得小巧精緻，以適應有限的空間和快速執行。
- 繞過安全：能 夠繞過安全機制，如防病毒軟件和入侵檢測系統。
- 功能多樣：不僅限於獲取殼，也能執行其他操作，如修改系統設置、竊取數據。
- 技術要求：開發和使用需要深入的技術知識，包括對操作系統和處理器架構的了解。
- 合法與非法用途：雖常用於安全測試，但也可能被用於惡意活動。

Shellcode


DEVCORE CONFERENCE 2024
ATTENDEE
DEVCORE CONFERENCE
Shell me if you can_
const char shellcode[] = "\x48\xbf\x2f\x62\x69\x6e"
int main(int argc, char *argv[], char *envp[]) {
/* What does DEVCORE say? */
((void(*)(void))&shellcode)();

Seccomp

Seccomp

- 安全計算模式：**Linux** 內核特性，允許進程限制可用的系統調用。
- 減少攻擊面：通過限制系統調用，降低被攻擊的風險。
- 兩種模式：
- **SECCOMP_MODE_STRICT** ：極限制性，僅允許極少數系統調用。
- **SECCOMP_MODE_FILTER** ：靈活模式，使用 **BPF** 定義過濾規則。
- 廣泛應用：用於沙箱環境、容器技術（如 **Docker**）、瀏覽器沙箱等。
- 行為控制：不允許的系統調用可以導致進程終止或生成信號。
- 增強安全：為瀏覽器、容器和服務進程提供額外的安全層。

Reconnaissance

File

File

```
└─$ file count
count: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, inter
preter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=77418302fcb6b1854dbd493345fa49f2af01160
d, for GNU/Linux 3.2.0, not stripped
```

File

```
└─❯ file count
count: Mach-O 64-bit executable arm64
❯
```

Practice Time

Reverse Engineering

Practice Time

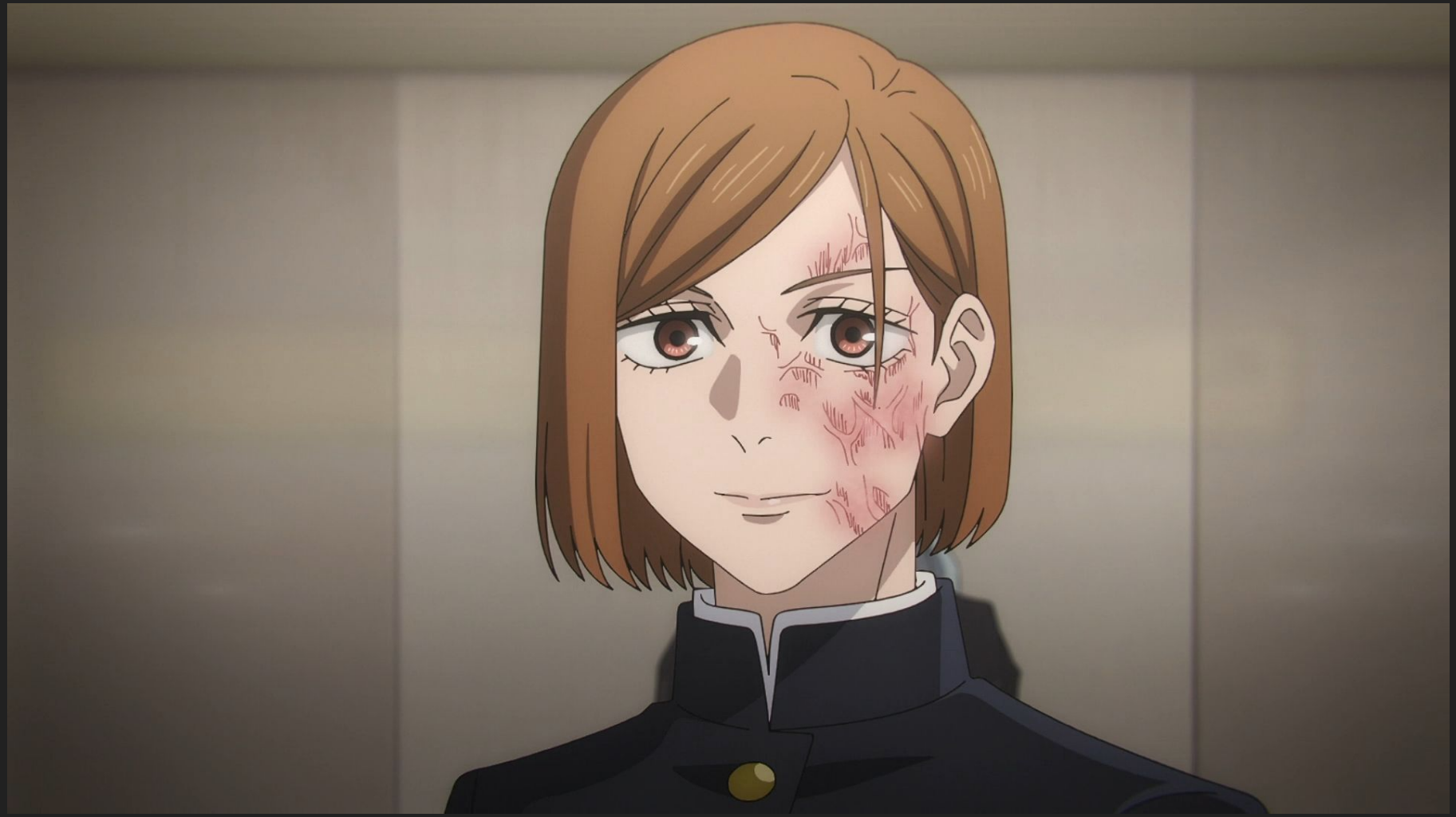
Exploitation

Practice Time

Exploit

Practice Time

Summary



References

- kazma.tw - blade