catthequest 副本

Web:

Web File Viewer

```
/secret/flag.txt base64 encode
```

Reverse:

Wramup

.pyc文件，pycdc反编译得到源码，写脚本解密即可

```python
input = "j6emavj5arn5vj2t6a2ha5pet{rv32p"
flag = ""
for i in range(len(input)):
    flag += chr(ord(input[i]) - 2)
print(flag)
```

CAT{h4ck_th3_pl3th0r4_0f_3ncrypt10n}

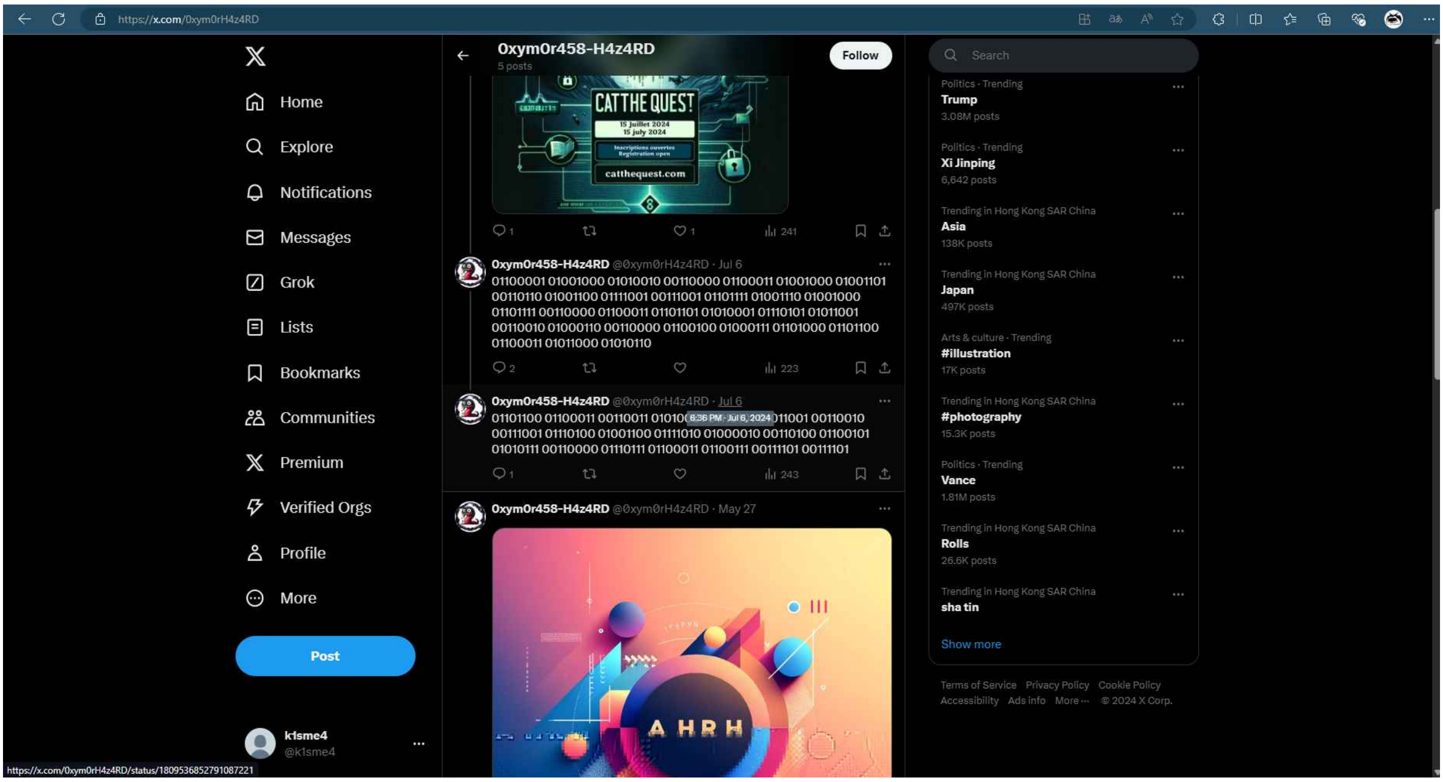
Jakshu

```c
#include <stdio.h>
int main()
{
  int v8[34];
  v8[0] = 46;
  v8[1] = 46;
  v8[2] = 58;
  v8[3] = 17;
  v8[4] = 88;
  v8[5] = 95;
  v8[6] = 86;
  v8[7] = 93;
  v8[8] = 90;
  v8[9] = 91;
  v8[10] = 92;
  v8[11] = 86;
  v8[12] = 94;
  v8[13] = 89;
  v8[14] = 80;
  v8[15] = 93;
  v8[16] = 94;
  v8[17] = 93;
  v8[18] = 11;
  v8[19] = 8;
  v8[20] = 12;
  v8[21] = 13;
  v8[22] = 11;
  v8[23] = 15;
  v8[24] = 80;
  v8[25] = 93;
  v8[26] = 86;
  v8[27] = 94;
  v8[28] = 15;
  v8[29] = 13;
  v8[30] = 8;
  v8[31] = 10;
  v8[32] = 11;
  v8[33] = 23;
  int input[40];
  char s[] = "monji";
  for (int i = 0; i < 34; i++)
  {
    v8[i] = v8[i] ^ s[i % 5];
    printf("%C", v8[i]);
  }
  return 0;
}
```

CAT{12930219039013aaabee9090edeee}

Misc:

0xym0r



https://x.com/0xym0rH4z4RD

0xym0r458-H4z4RD
5 posts

0xym0r458-H4z4RD @0xym0rH4z4RD · Jul 6

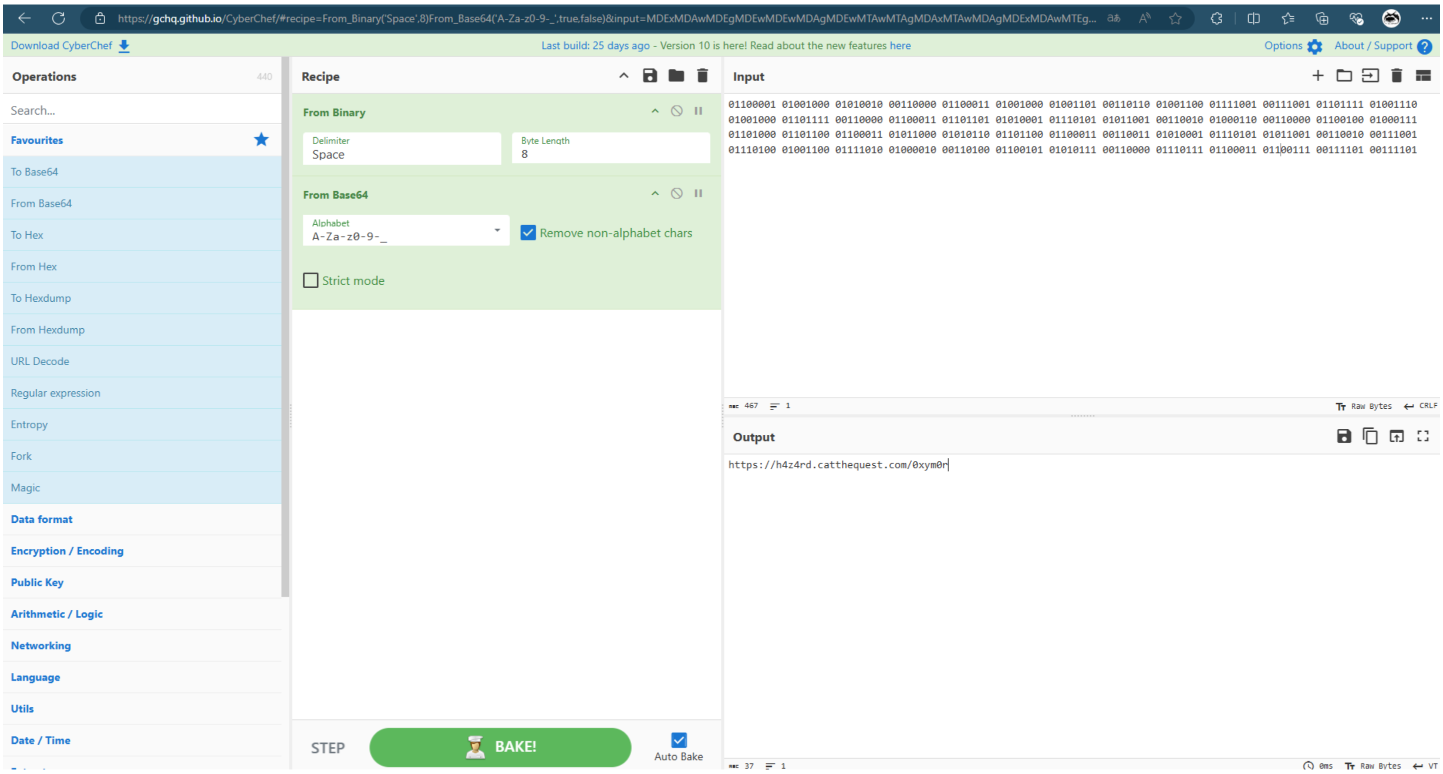
01100001 01001000 01010010 00110000 01100011 01001000 01001101 00110110 01001100 01111001 00111001 01101111 01001110 01001000 01101111 00110000 01100011 01101101 01010001 01110101 01011001 00110010 01000110 00110000 01100100 01000111 01101000 01101100 01100011 01011000 01010110

0xym0r458-H4z4RD @0xym0rH4z4RD · Jul 6

01101100 01100011 00110011 01010(6:36 PM · Jul 6, 2024)11001 00110010 00111001 01110100 01001100 01111010 01000010 00110100 01100101 01010111 00110000 01110111 01100011 01100111 00111101 00111101

0xym0r458-H4z4RD @0xym0rH4z4RD · May 27

https://x.com/0xym0rH4z4RD/status/1809536852791087221



https://gchq.github.io/CyberChef/#recipe=From_Binary('Space',8)From_Base64('A-Za-z0-9-_',true,false)&input=MDExMDAwMDEgMDEwMDEwMDAgMDEwMTAwMTAgMDAxMTAwMDAgMDExMDAwMTEg...

Recipe

From Binary
Delimiter: Space
Byte Length: 8

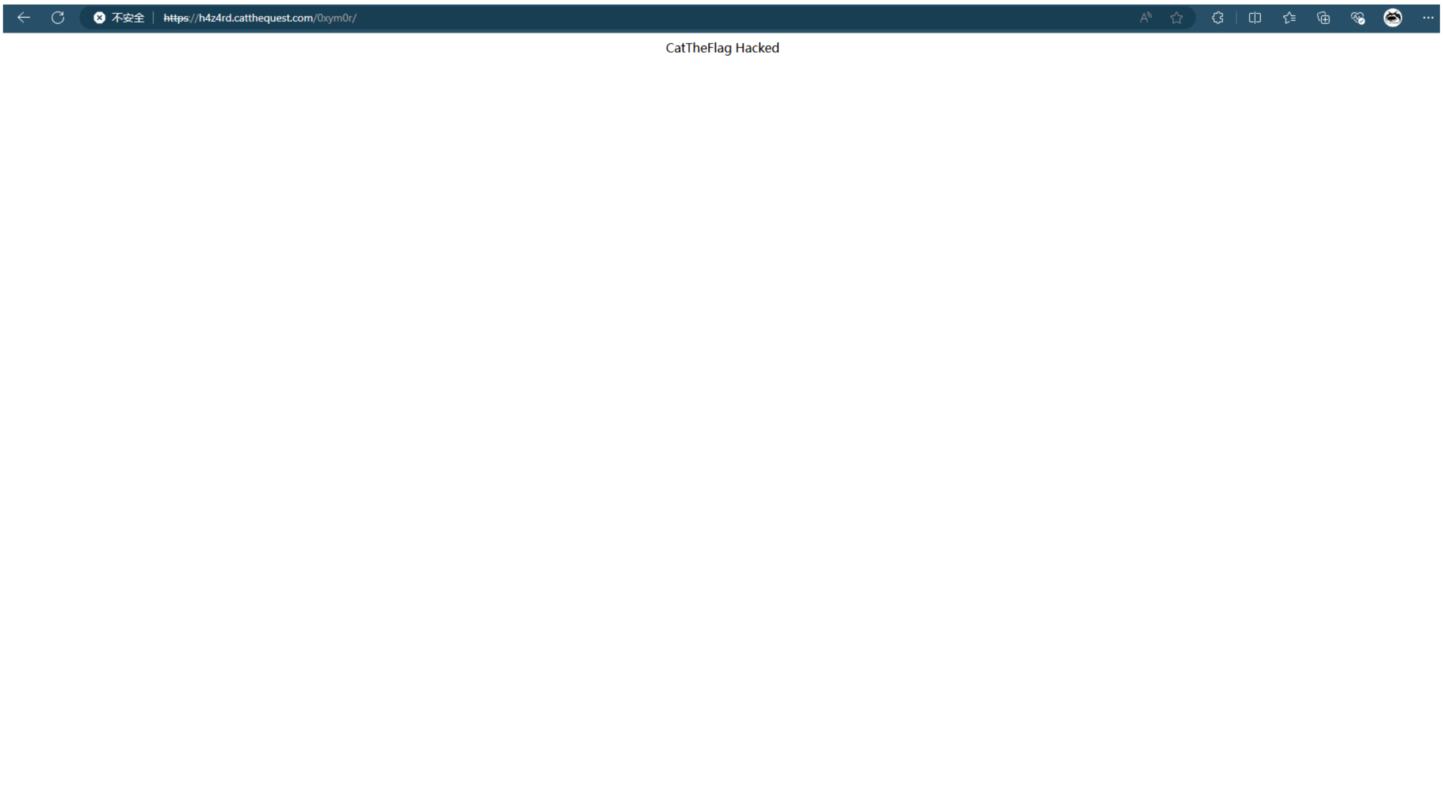
From Base64
Alphabet: A-Za-z0-9-_
☑ Remove non-alphabet chars
□ Strict mode

Input

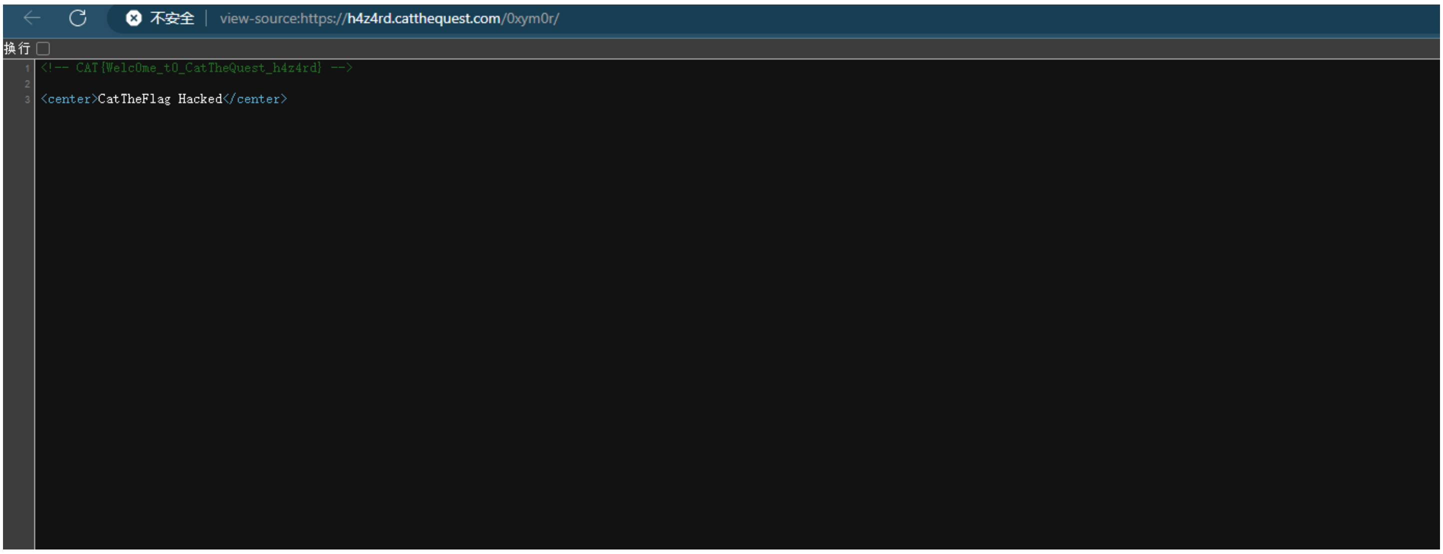
```
01100001 01001000 01010010 00110000 01100011 01001000 01001101 00110110 01001100 01111001 00111001 01101111 01001110
01001000 01101111 00110000 01100011 01101101 01010001 01110101 01011001 00110010 01000110 00110000 01100100 01000111
01101000 01101100 01100011 01011000 01010110 01101100 01100011 00110011 01010001 01110101 01011001 00110010 00111001
01110100 01001100 01111010 01000010 00110100 01100101 01010111 00110000 01110111 01100011 01100111 00111101 00111101
```

Output

```
https://h4z4rd.catthequest.com/0xym0r
```



CatTheFlag Hacked

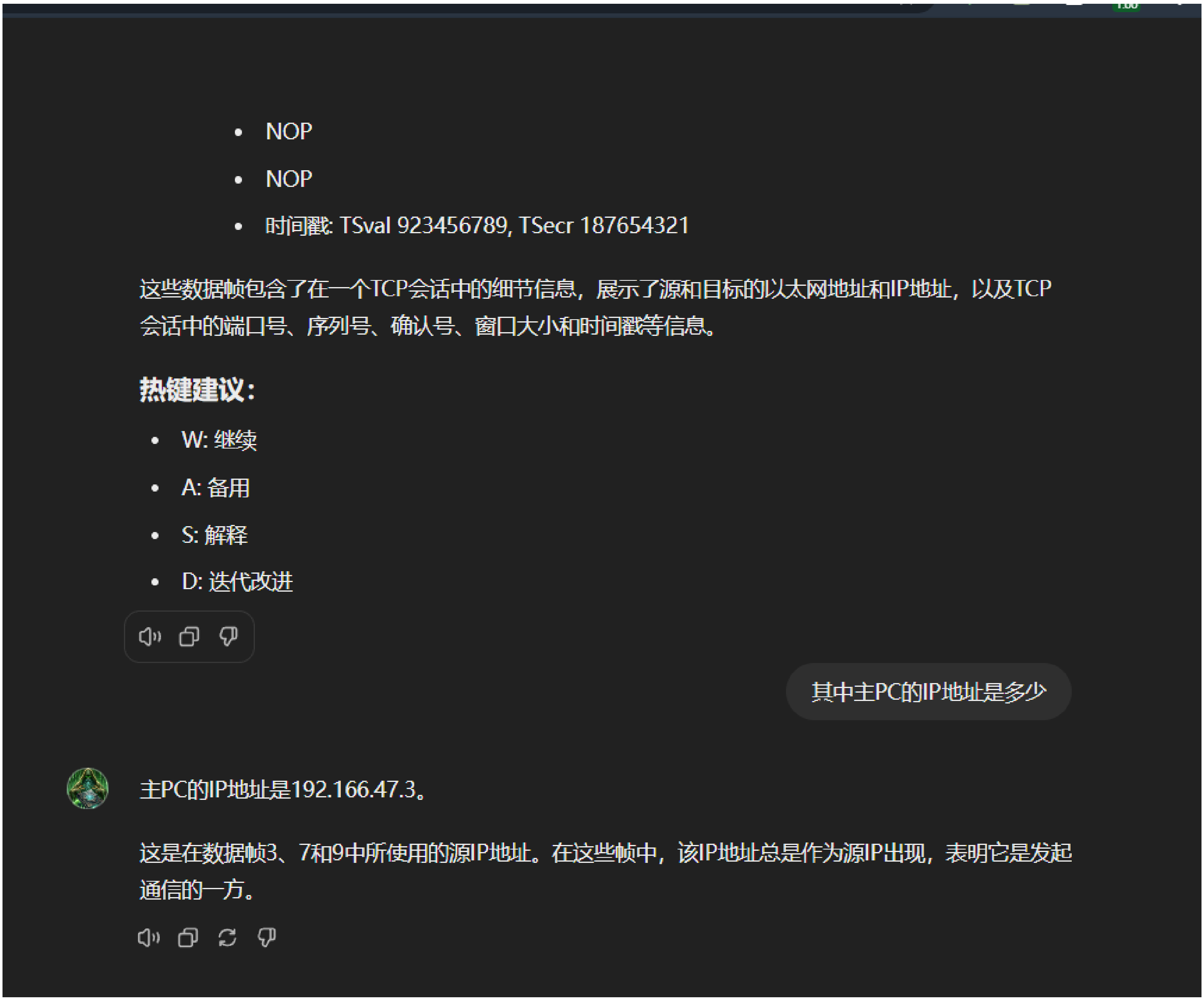


```
<!-- CAT{Welc0me_t0_CatTheQuest_h4z4rd} -->

<center>CatTheFlag Hacked</center>
```

```
CAT{Welc0me_t0_CatTheQuest_h4z4rd}
```

Find da target

给了一个网络数据帧文件，说是让找一个主PC端的IP地址，懒得看，大模型直接出了



    - NOP
    - NOP
    - 时间戳: TSval 923456789, TSecr 187654321

这些数据帧包含了在一个TCP会话中的细节信息，展示了源和目标的以太网地址和IP地址，以及TCP会话中的端口号、序列号、确认号、窗口大小和时间戳等信息。

热键建议:

- W: 继续
- A: 备用
- S: 解释
- D: 迭代改进

其中主PC的IP地址是多少

主PC的IP地址是192.166.47.3。

这是在数据帧3、7和9中所使用的源IP地址。在这些帧中，该IP地址总是作为源IP出现，表明它是发起通信的一方。

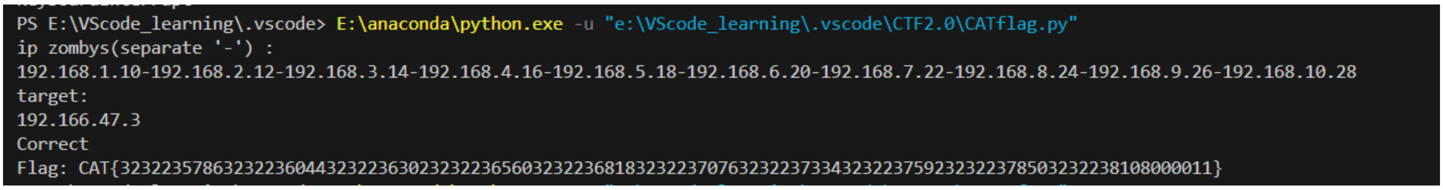
Research and Destroy

给了一段模拟将流量从僵尸计算机重定向到目标计算机来模拟攻击的脚本，说白了就是解密，按照它的逻辑将列表中的僵尸ip和目标ip转化出来，把原本的代码修修，按照原本逻辑跑一下，对应ip输进去直接出了

```
zombys_ip:
192.168.1.10-192.168.2.12-192.168.3.14-192.168.4.16-192.168.5.18-192.168.6.20-192.168.7.22-192.168.8.24-192.168.9.26-192.168.10.28
target_ip:
192.166.47.3
```

代码给修修

```python
import struct

def ip_to_binary(ip):
    return struct.unpack("!I", struct.pack("!BBBB", *map(int, ip.split('.'))))[0]

def binary_to_ip(binary):
    return '.'.join(map(str, struct.unpack('!BBBB', struct.pack('!I', binary))))

def ip_to_hex(ip):
    return f"{ip_to_binary(ip):08x}"

def ip_to_base64(ip):
    return struct.pack('!I', ip_to_binary(ip)).hex()

def hex_to_ip(hex_str):
    return binary_to_ip(int(hex_str, 16))

def bin_to_ip(bin_str):
    return binary_to_ip(int(bin_str, 2))

zombys = [
    'c0a8010a',
    'c0a8020c',
    'c0a8030e',
    'c0a80410',
    'c0a80512',
    'c0a80614',
    'c0a80716',
    'c0a80818',
    'c0a8091a',
    'c0a80a1c'
]

target = 'c0a62f03'

zombie_ips_binary = [bytes.fromhex(ip) for ip in zombys]
target_ip_binary = bytes.fromhex(target)

def ip_to_binary_str(ip):
    return f"{ip_to_binary(ip):032b}"

def binary_str_to_ip(bin_str):
    return binary_to_ip(int(bin_str, 2))

def check_zombie_ips(zombie_ips):
    try:
        zombie_ips_input_binary = [ip_to_binary(ip) for ip in zombie_ips]
        return set(zombie_ips_input_binary) == set([int(ip.hex(), 16) for ip in zombie_ips_binary])
    except ValueError as e:
        print(f"Error: {e}")
        return False

def check_target_ip(target_ip):
    return ip_to_binary(target_ip) == struct.unpack("!I", target_ip_binary)[0]

print("ip zombys(separate '-') :")
zombie_ips_input = input().split('-')
print("target:")
target_ip_input = input().strip()

if check_zombie_ips(zombie_ips_input) and check_target_ip(target_ip_input):
    print("Correct")
    last_octets = [ip.split('.')[-1] for ip in zombie_ips_input]
    last_octets.append(target_ip_input.split('.')[-1])
    bin_strs = [ip_to_binary_str(ip) for ip in zombie_ips_input]
    target_ip_bin_str = ip_to_binary_str(target_ip_input)

    flag_parts = [
        "CAT{",
        ''.join([str(int(bin_str, 2)) for bin_str in bin_strs]),
        target_ip_bin_str[-6:],
        "} "
    ]

    flag = ''.join(flag_parts)
    print("Flag:", flag)
else:
    print("No No No")

```

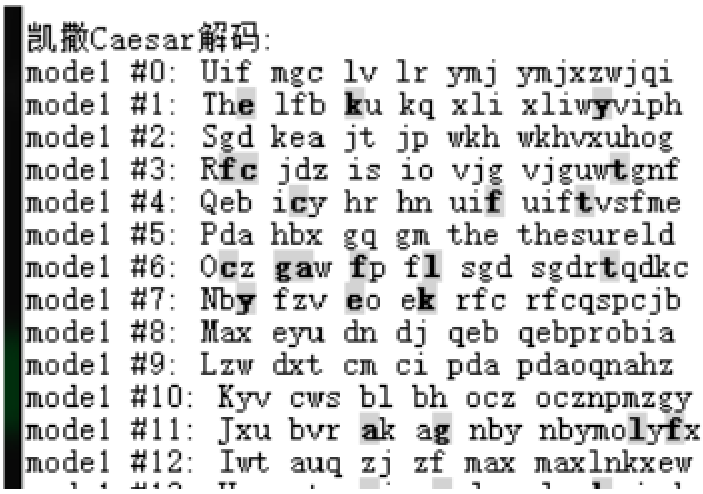


```
PS E:\VScode_learning\.vscode> E:\anaconda\python.exe -u "e:\VScode_learning\.vscode\CTF2.0\CATflag.py"
ip zombys(separate '-') :
192.168.1.10-192.168.2.12-192.168.3.14-192.168.4.16-192.168.5.18-192.168.6.20-192.168.7.22-192.168.8.24-192.168.9.26-192.168.10.28
target:
192.166.47.3
Correct
Flag: CAT{3232235786323223604432322363023232236560323223681832322370763232237334323223759232322378503232238108000011}
```

V@mos

```
the key is in the thesureld
题干就是flag
```

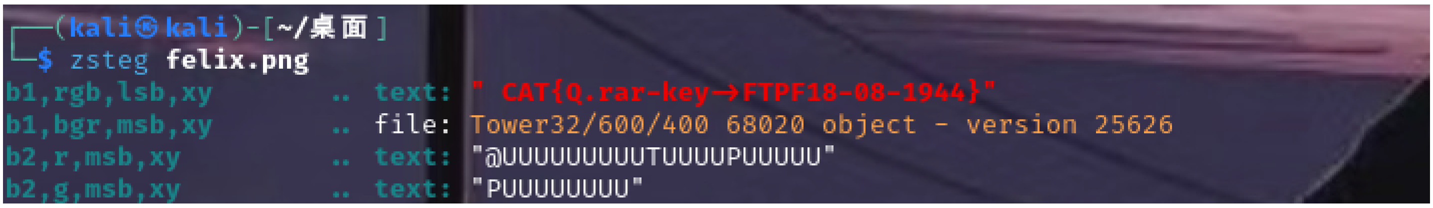
```
CAT{V@m0s_12}
```



```
凯撒Caesar解码:
model #0:  Uif mgc lv lr ymj ymjxzwjqi
model #1:  The lfb ku kq xli xliwyviph
model #2:  Sgd kea jt jp wkh wkhvxuhog
model #3:  Rfc jdz is io vjg vjguwtgnf
model #4:  Qeb icy hr hn uif uiftvsfme
model #5:  Pda hbx gq gm the thesureld
model #6:  Ocz gaw fp fl sgd sgdrtqdkc
model #7:  Nby fzv eo ek rfc rfcqspcjb
model #8:  Max eyu dn dj qeb qebprobia
model #9:  Lzw dxt cm ci pda pdaoqnahz
model #10:  Kyv cws bl bh ocz ocznpmzgy
model #11:  Jxu bvr ak ag nby nbymolyfx
model #12:  Iwt auq zj zf max maxlnkxew
```

[2]Affiche Cybernétique

签到题，图片保存下来，zsteg拿flag：



```
┌──(kali㉿kali)-[~/桌面]
└─$ zsteg felix.png
b1,rgb,lsb,xy       .. text: " CAT{Q.rar-key→FTPF18-08-1944}"
b1,bgr,msb,xy       .. file: Tower32/600/400 68020 object - version 25626
b2,r,msb,xy         .. text: "@UUUUUUUUUTUUUUPUUUUU"
b2,g,msb,xy         .. text: "PUUUUUUUU"
```

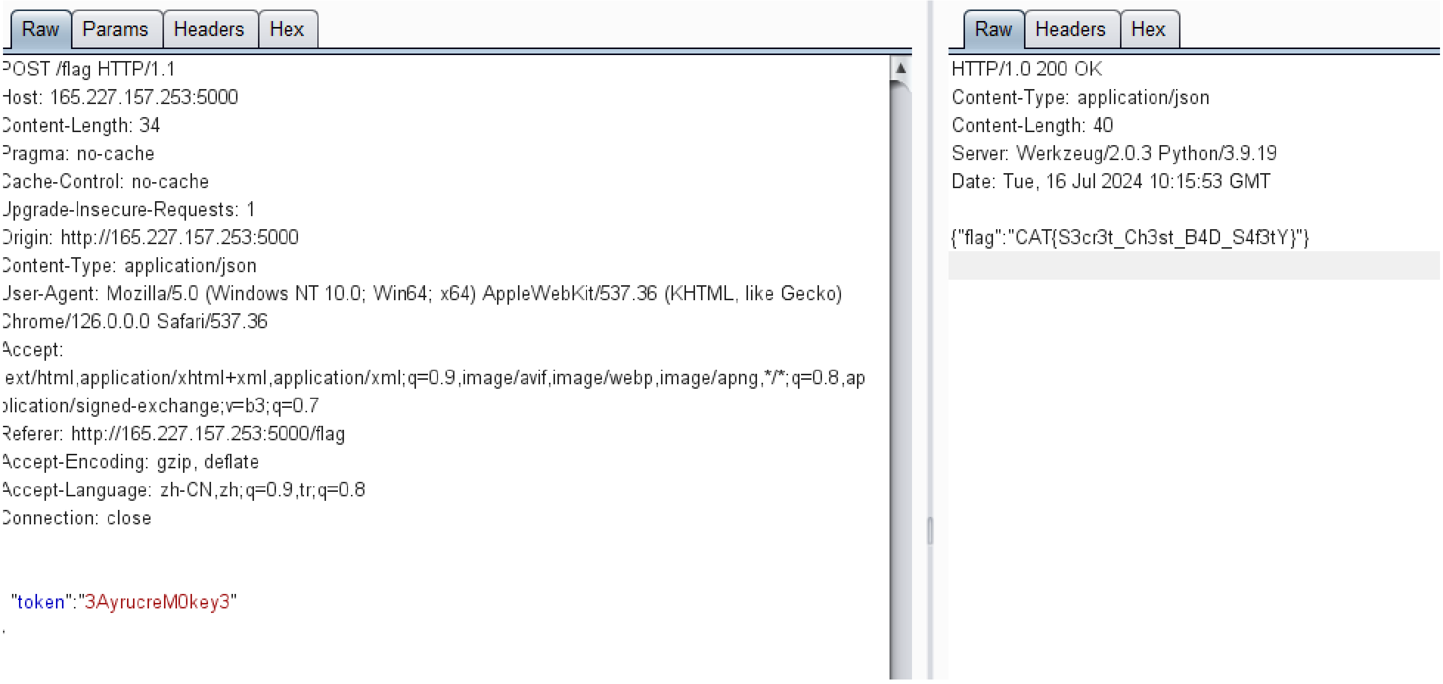
根据题目要求改一下即可

```
CAT{Q.rar-key->FTPF18/08/1944}
```

Secret Chest

```
POST /flag HTTP/1.1
Host: 165.227.157.253:5000
Content-Length: 35
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
Origin: http://165.227.157.253:5000
Content-Type: application/json
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://165.227.157.253:5000/flag
```

```
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,tr;q=0.8
Connection: close

{
  "token":"3AyrucreM0key3"
}

```

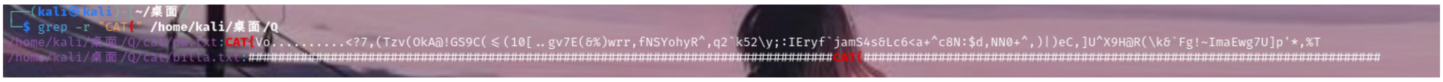


```
POST /flag HTTP/1.1
Host: 165.227.157.253:5000
Content-Length: 34
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
Origin: http://165.227.157.253:5000
Content-Type: application/json
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/126.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,ap
plication/signed-exchange;v=b3;q=0.7
Referer: http://165.227.157.253:5000/flag
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,tr;q=0.8
Connection: close


 "token":"3AyrucreM0key3"
```

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length: 40
Server: Werkzeug/2.0.3 Python/3.9.19
Date: Tue, 16 Jul 2024 10:15:53 GMT

{"flag":"CAT{S3cr3t_Ch3st_B4D_S4f3tY}"}
```

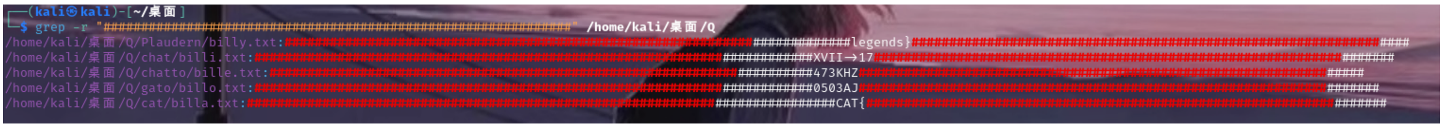
[3]Mise en Abymes

测试一下`CAT{`:



```
┌──(kali㉿kali)-[~/桌面]
└─$ grep -r "CAT{" /home/kali/桌面/Q
/home/kali/桌面/Q/cat/dw.txt:CAT{Vo..........<?7,(Tzv(OkA@!GS9C(≤(10[..gv7E(&%)wrr,fNSYohyR^,q2`k52\y;:IEryf`jamS4s&Lc6<a+^c8N:$d,NN0+^,)|)eC,]U^X9H@R(\k&`Fg!~ImaEwg7U]p'*,%T
/home/kali/桌面/Q/cat/billa.txt:######################################################################################CAT{##########################################################################################
```

了解了，标志为这一大串#号，继续搜：



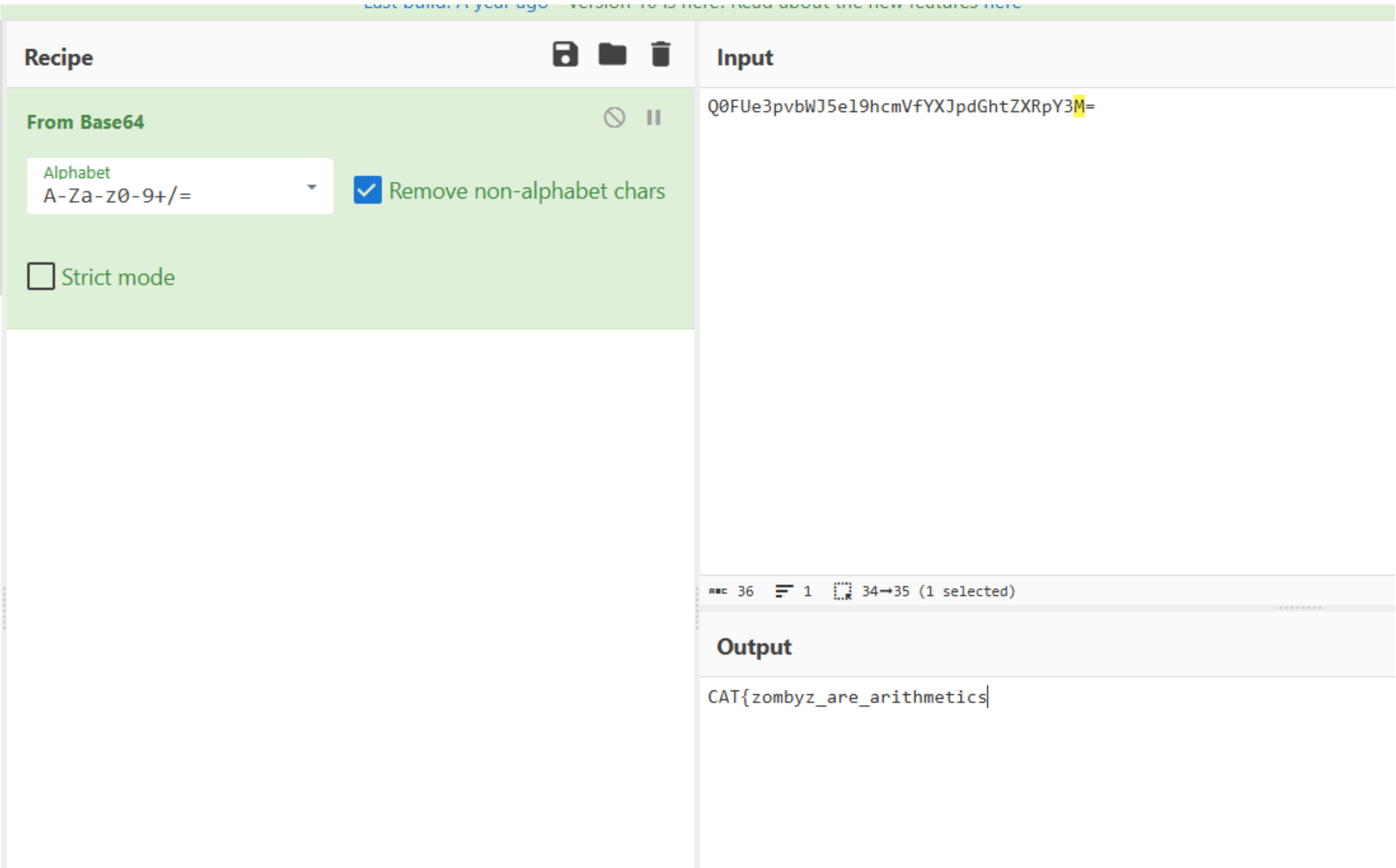
```
┌──(kali㉿kali)-[~/桌面]
└─$ grep -r "##################################################################" /home/kali/桌面/Q
/home/kali/桌面/Q/Plaudern/billy.txt:############################################################################legends}######################################################################
/home/kali/桌面/Q/chat/billi.txt:##########################################################################XVII→17##########################################################################
/home/kali/桌面/Q/chatto/bille.txt:######################################################################473KHZ######################################################################
/home/kali/桌面/Q/gato/billo.txt:########################################################################0503AJ#######################################################################
/home/kali/桌面/Q/cat/billa.txt:############################################################################CAT{#######################################################################
```

按顺序拼一下即可：

```
CAT{473KHZXVII->170503AJlegends}
```

Door and machines

Q0FUe3pvbWJ5el9hcmVfYXJpdGhtZXRpY3M=



From Base64

Alphabet: A-Za-z0-9+/=

☑ Remove non-alphabet chars

□ Strict mode

Input

```
Q0FUe3pvbWJ5el9hcmVfYXJpdGhtZXRpY3M=
```

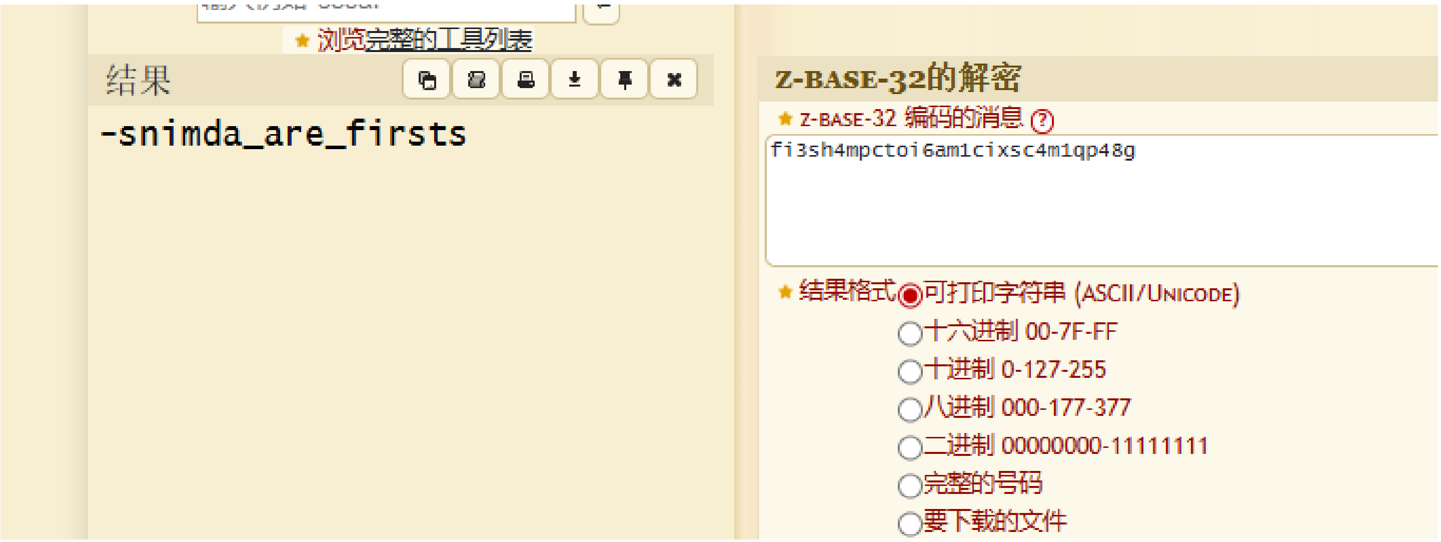
36 1 34→35 (1 selected)

Output

```
CAT{zombyz_are_arithmetics
```

fi3sh4mpctoi6am1cixsc4m1qp48g

`Z-Base-32` 解密：



结果

```
-snimda_are_firsts
```

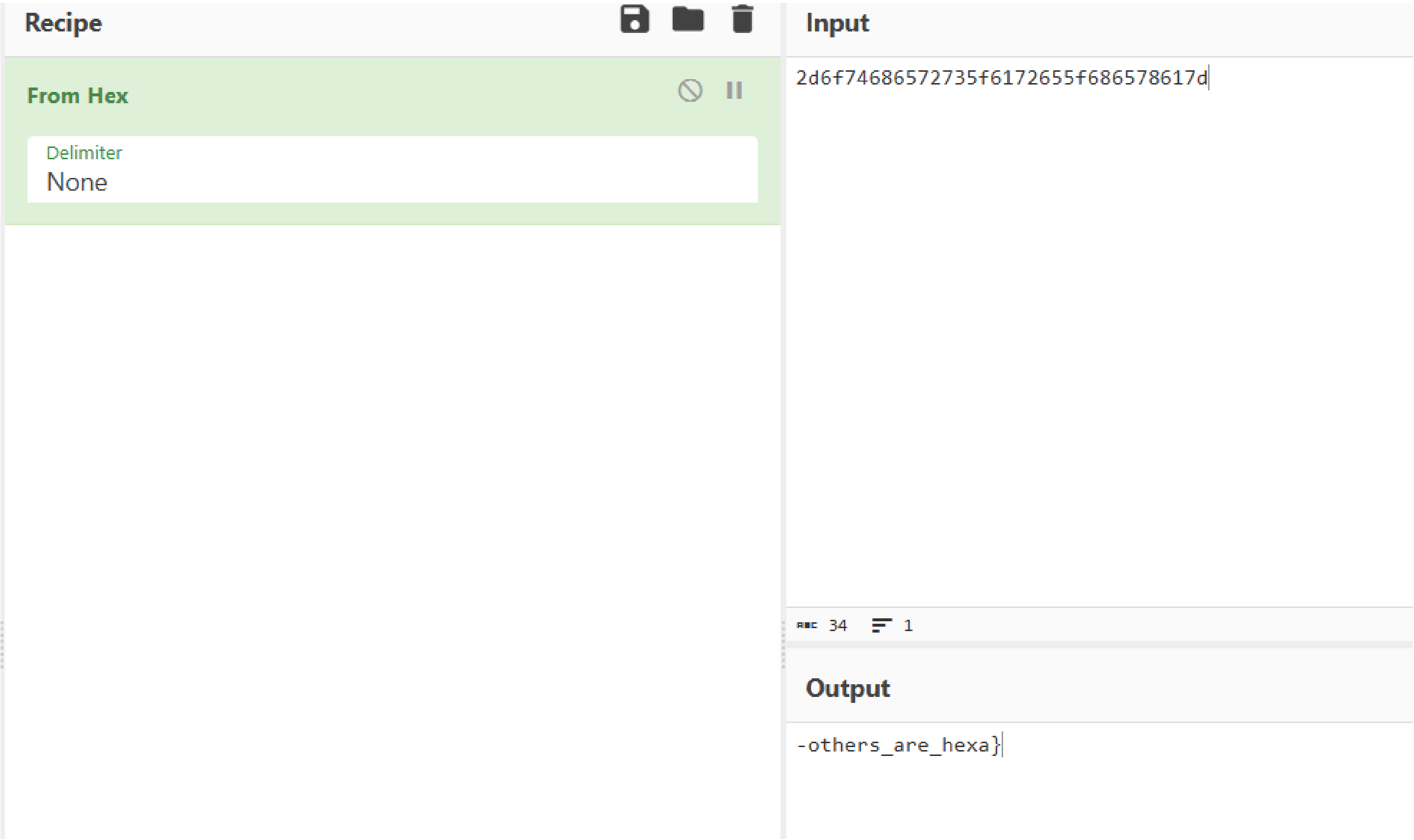
Z-BASE-32的解密

★ Z-BASE-32 编码的消息 ?

```
fi3sh4mpctoi6am1cixsc4m1qp48g
```

★ 结果格式 ●可打印字符串 (ASCII/UNICODE)
○十六进制 00-7F-FF
○十进制 0-127-255
○八进制 000-177-377
○二进制 00000000-11111111
○完整的号码
○要下载的文件

2d6f74686572735f6172655f686578617d



Delimiter
None

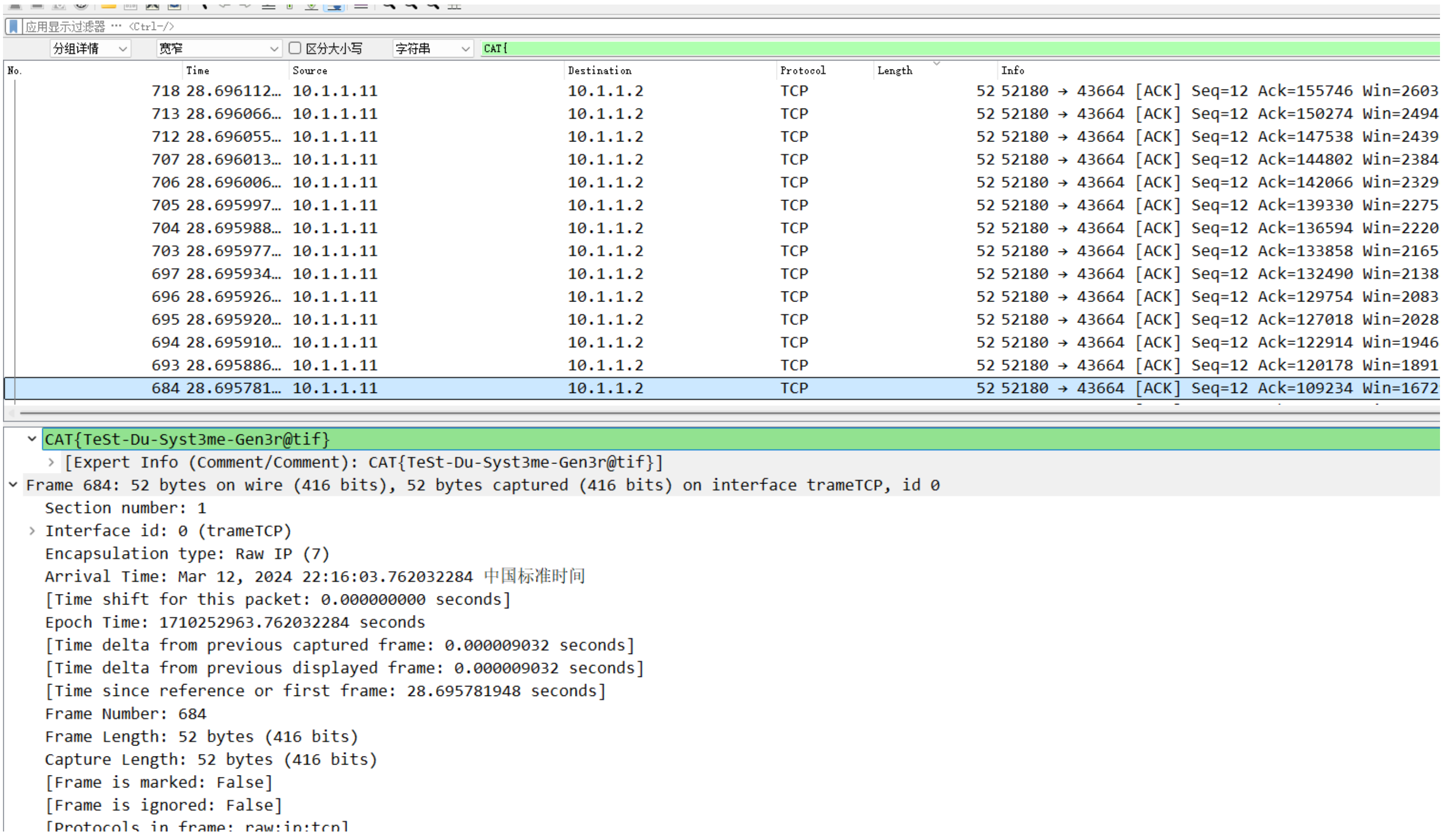
```
2d6f74686572735f6172655f686578617d
```

```
-others_are_hexa}
```

`CAT{zombyz_are_arithmetics-snimda_are_firsts-others_are_hexa}`

A network trace

看了一眼 全是tcp 大小都差不多 直接搜就有了



CAT{

No.	Time	Source	Destination	Protocol	Length	Info
718	28.696112…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=155746 Win=2603
713	28.696066…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=150274 Win=2494
712	28.696055…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=147538 Win=2439
707	28.696013…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=144802 Win=2384
706	28.696006…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=142066 Win=2329
705	28.695997…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=139330 Win=2275
704	28.695988…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=136594 Win=2220
703	28.695977…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=133858 Win=2165
697	28.695934…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=132490 Win=2138
696	28.695926…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=129754 Win=2083
695	28.695920…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=127018 Win=2028
694	28.695910…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=122914 Win=1946
693	28.695886…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=120178 Win=1891
684	28.695781…	10.1.1.11	10.1.1.2	TCP	52	52180 → 43664 [ACK] Seq=12 Ack=109234 Win=1672

```
  ˅ CAT{TeSt-Du-Syst3me-Gen3r@tif}
    › [Expert Info (Comment/Comment): CAT{TeSt-Du-Syst3me-Gen3r@tif}]
˅ Frame 684: 52 bytes on wire (416 bits), 52 bytes captured (416 bits) on interface trameTCP, id 0
    Section number: 1
  › Interface id: 0 (trameTCP)
    Encapsulation type: Raw IP (7)
    Arrival Time: Mar 12, 2024 22:16:03.762032284 中国标准时间
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1710252963.762032284 seconds
    [Time delta from previous captured frame: 0.000009032 seconds]
    [Time delta from previous displayed frame: 0.000009032 seconds]
    [Time since reference or first frame: 28.695781948 seconds]
    Frame Number: 684
    Frame Length: 52 bytes (416 bits)
    Capture Length: 52 bytes (416 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: raw:ip:tcp]
```

`CAT{TeSt-Du-Syst3me-Gen3r@tif}`