

SDBCTF2021 writeup

原创

电子young 于 2022-01-15 19:17:09 发布 2140 收藏

分类专栏: [ctf比赛](#) 文章标签: [网络安全](#)

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_63157899/article/details/122513102

版权



[ctf比赛](#) 专栏收录该内容

1 篇文章 0 订阅

订阅专栏

WEB

Ez_math

请输入三个整数A、B、C,动动你的脑袋瓜,使得

$$A*19+B*23+C*67=207247$$

A:

B:

C:

Submit

$$()*19+()*23+()*67=$$

CSDN @电子young

下方有相应计算语句,那么构建

$$a=1, b=1, c=1)+207137+(1$$

语句为:

$$(1)*19+(1)*23+(1)+207137+(1)*67=207247$$

flag{Your_Math_1s_Very_G00D}

CSDN @电子young

得出flag

Ez_math2

草率了，貌似能爆破出来，我修改一下数据，看你怎么办！！！！ 😭😭😭😭😭😭😭😭

$$A*207247+B*207213+C*207212=2072$$

A:

B:

C:

(0)*207247+(0)*207213+(0)+2072+(0)*207212=2072
flag{1_am_A_j0k3r!!!}

CSDN @电子young

和上面的方法一样

Ez_login



尝试万能密码

```
admin' or '1'='1
```

提交得到flag

套娃

```

<?php
error_reporting(0);
$cr0fy = $_REQUEST['cr0fy'];
is_numeric($cr0fy)?die("no way"):NULL;
if($cr0fy>=2072)
{
    echo file_get_contents('../1.php');
    echo "How's that possible";
}
highlight_file(__FILE__);
//研究研究is_numeric()函数
?>

```

CSDN @电子young

is_numeric()检测变量是否为数字或数字字符串

阅读代码后构造payload

```
https://69ab9160.cpolar.io/?cr0fy=2072a
```

进入第二个页面

```

Just glve it a try. <?php
error_reporting(0);
if (isset($_GET['N0vice']) and isset($_GET['E1li0t']))
{
    if ($_GET['N0vice'] == $_GET['E1li0t'])
        echo 'no no no,you can not do this';

    else if (sha1($_GET['N0vice']) == sha1($_GET['E1li0t'])) {
        echo 'ok ok ok,I glve you fl4g!';
        echo file_get_contents('../2.php');
    }
    else
        echo 'not right,may be you need google';
}
else
    echo 'Just glve it a try.';
highlight_file(__FILE__);
?>

```

//嘿嘿啊哈哈哈哈哈哈哈哈哈， sha1强碰撞来咯 🐦🐦🐦😄😄

//嘿嘿啊哈哈哈哈哈哈哈哈哈， sha1强碰撞来咯 🐦🐦🐦😄😄

CSDN @电子young

可以用数组绕过sha1检测， payload:

```
https://69ab9160.cpolar.io/come_baby.php/?N0vice[]=1&&E1li0t[]=2
```

进入第三个页面

Just glve it a try. <?php

```
error_reporting(0);
if (isset($_GET['N0vice']) and isset($_GET['Ell10t']))
{
    if ($_GET['N0vice'] == $_GET['Ell10t'])
        echo 'no no no,no you can not do this';
    else if (is_array($_GET['N0vice']) || is_array($_GET['Ell10t']))
    {
        echo 'too young too simple';
    }
    else if (sha1($_GET['N0vice']) == sha1($_GET['Ell10t'])) {
        echo 'we1!!!,I will give you real flag:';
        echo "<br/>";
        echo file_get_contents('../flag.php');
    }
    else
        echo 'May be you need google or baidu?';
}
else
    echo 'Just glve it a try.';
highlight_file(__FILE__);
?>
```

//太可恶了，太可恶了，刚刚大意了，没有设置数组检测，让你给过来了，不讲武德，来偷袭!!!

//我这次检测了数组，看你怎么办，哼😏😏😏😏 //太可恶了，太可恶了，刚刚大意了，没有设置数组检测，让你给过来了，不讲武德，来偷袭!!! //我这次检测了数组，看你怎么办，哼😏😏😏😏

CSDN @电子young

发现存在is_array检测数组

只能用sha1碰撞，找到谷歌发布的sha1

```
N0vice=%25PDF-1.3%0A%25E2%E3%CF%D3%0A%0A%0A1%20%20obj%0A%3C%3C/Width%20%20%20R/Height%20%20%20R/Type%
Ell10t=%25PDF-1.3%0A%25E2%E3%CF%D3%0A%0A%0A1%20%20obj%0A%3C%3C/Width%20%20%20R/Height%20%20%20R/Type%
```

提交得到flag

召唤神龙



蝌蚪

击败了全球10%的玩家!

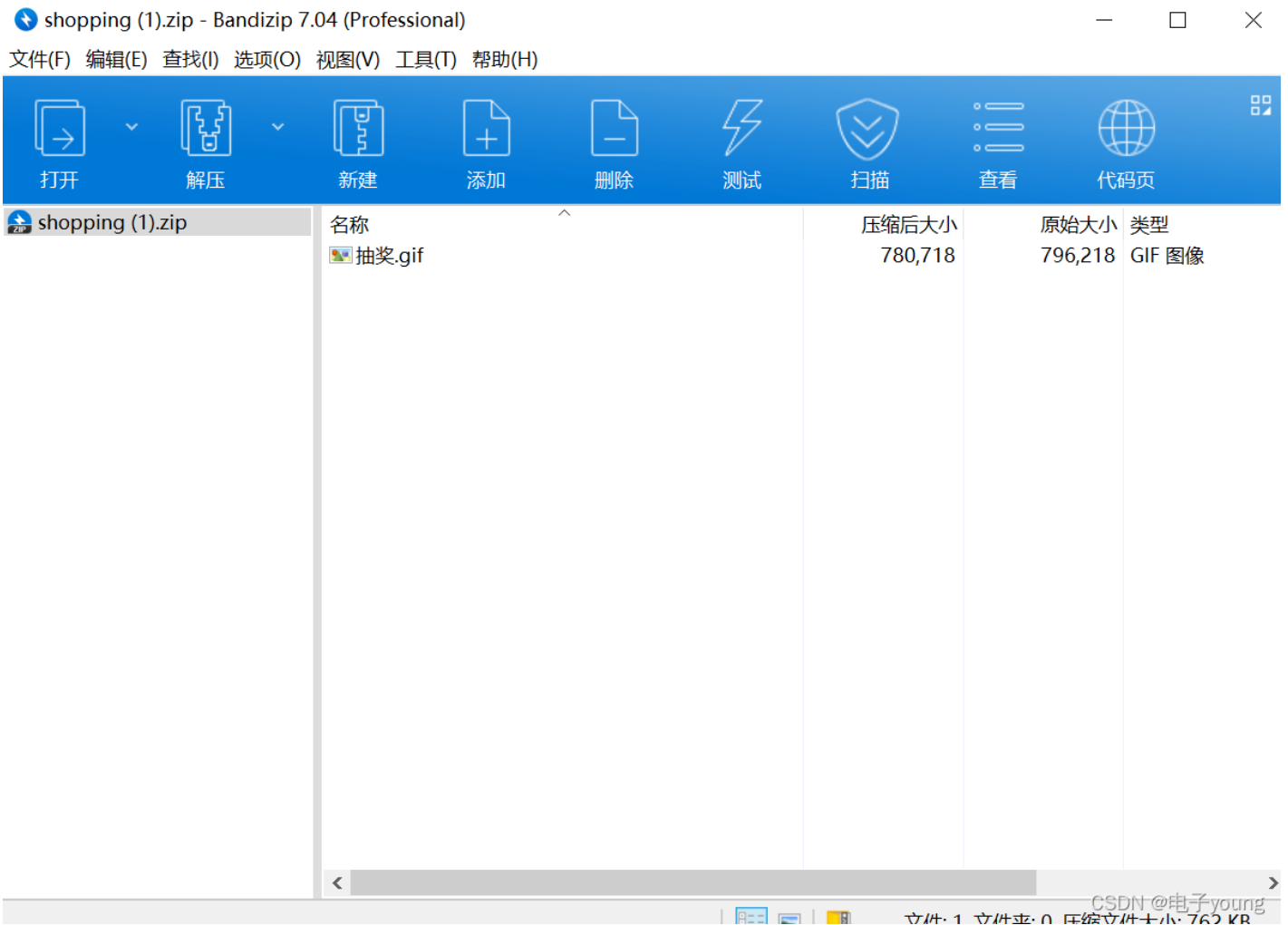
CSDN @电子young

进入检查----猜测flag在网页某个代码中----搜索flag得到

The screenshot shows the Chrome DevTools Network tab. The left pane displays the request and response details for a resource named 'flag'. The response is a long string of characters, including 'flag{zhaohu@mshenlon9_...}'. The right pane shows the response body, which is a JSON array containing a single element: 'flag{zhaohu@mshenlon9_is_funny!}'. The status bar at the bottom indicates '75 个请求 | 已传输 4.0 MB | 6.4'.

MISC

抽奖



发现是gif文件，进行分离，之前的工具不知道为啥分离不了，去在线工具找gif分离



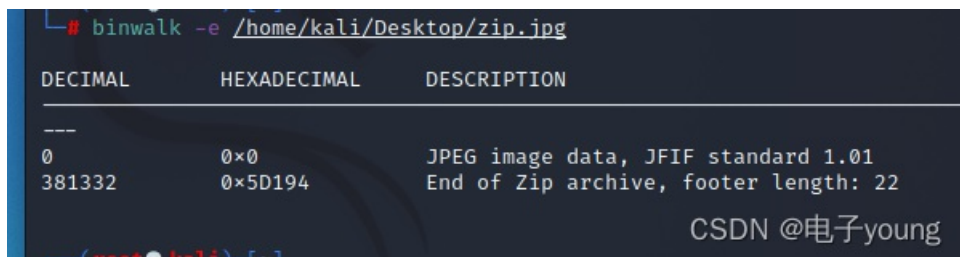
藏猫猫



使用工具找

倘来之物

打开发现是一个图片，丢入binwalk分析



发现有zip存在，把它提取出来

The screenshot shows the WinHex application with a file list on the left and a hex dump on the right.

File list:

1. _dhdffjwdujihchicler (1).png
- 2 C:\Users\young\checkin.png
- 3 _dhdffjwdujihchicler (1) - 副本.png
- 4 C:\Users\young\Desktop\2D4.zlib
- 5 C:\Users\young\mianju.jpg
- 6 C:\Users\young\消失的flag.png
- 7 C:\Users\young\Do_!flag.zip
- 8 C:\Users\young\Desktop\pic.jpg
- 9 C:\Users\young\Desktop\music.zip

Hex dump (ANSI ASCII):

```

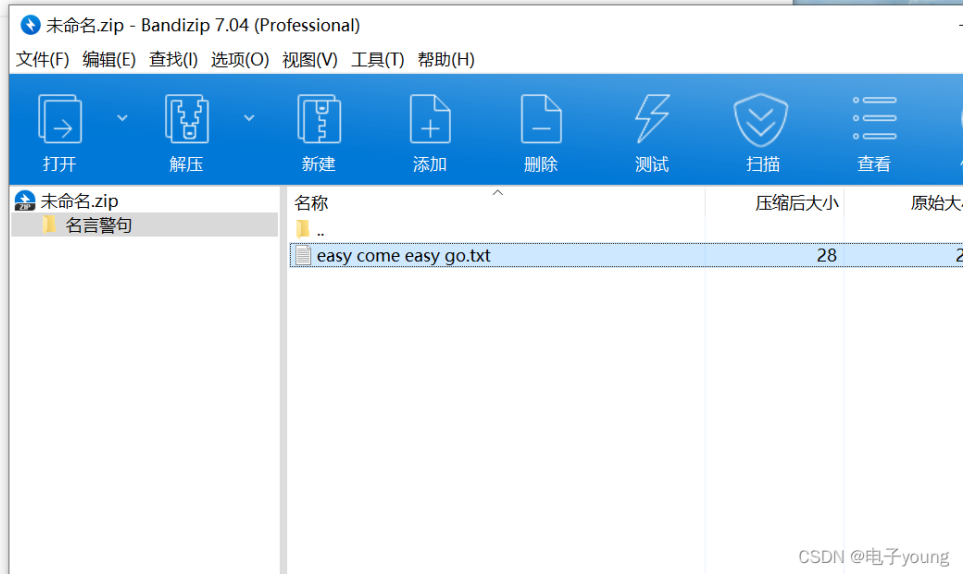
00381152 08 02 00 80 20 08 02 00 80 20 08 02 00 FF D9      ly0
00381159 58 48 03 04 14 00 00 00 00 A9 80 26 54 F7 11 FK  0*GT-
00381164 EB 62 10 00 00 1C 00 00 00 1E 00 00 C3 FB 68    8
00381200 D1 D4 BE AF BE E4 2F 65 61 73 79 20 63 6F 6D 65  00*ks/easy.com
00381216 20 65 61 73 79 20 67 6F 2E 74 78 74 5A 6D 78 68  easy.go.txtZmxh
00381232 5A 33 74 5A 61 56 39 4D 59 57 6C 67 57 6C 65 2F  2FC2AVWMTFwIc
00381248 55 58 56 66 66 51 3D 3D 50 4B 01 02 14 00 14 00  UCVeFfo-PK
00381264 00 00 00 00 A9 80 26 54 F7 11 EB 6E 3C 00 00 00  0*GT-en
00381280 10 00 00 00 1E 00 00 00 00 00 00 01 00 20 08    8
00381296 08 00 00 00 00 C3 FB D1 D4 BE AF BE E4 2F 65    say.com/easy.go
00381312 81 73 79 20 63 6F 6D 65 20 65 61 73 79 20 67 6F  say.com/easy.go
00381328 2E 74 78 74 50 4B 05 06 01 00 01 00 01 00 01 00  x-zlib
00381344 40 00 00 00 58 00 00 00 00 00
  
```


发现

easy come easy go.txt - 记事本

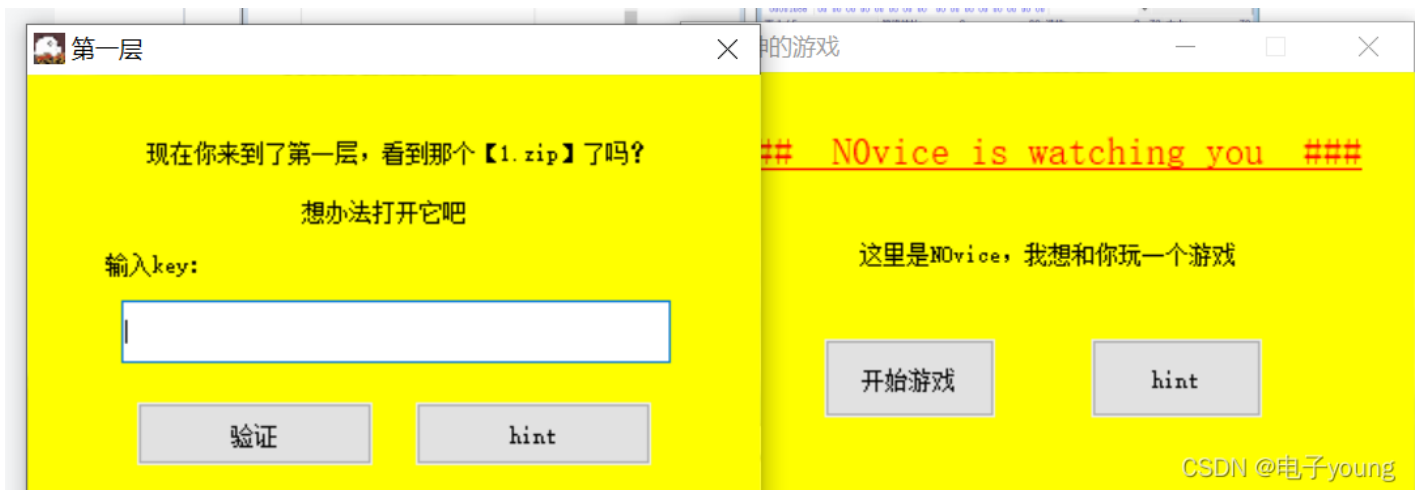
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

ZmxhZ3tZaV9MYWlfWWlfUXVffQ==



一串base64编码，解码得到flag

N0vice的游戏



一个压缩包，爆破密码为123456，得到二维码，扫码得到key

第二个为一张图片，放入binwalk分离

```
root@kali: ~
File Actions Edit View Help

(root@kali)-[~]
# binwalk /home/kali/Desktop/2.png

DECIMAL      HEXADECIMAL  DESCRIPTION
---
0            0x0         PNG image, 1209 x 1920, 8-bit/color RGBA, non-i
nterlaced
41          0x29       Zlib compressed data, default compression
1551846     0x17ADE6   Zip archive data, at least v2.0 to extract, com
pressed size: 49, uncompressed size: 44, name: key_2.txt
1552025     0x17AE99   End of Zip archive, footer length: 22

(root@kali)-[~]
# binwalk -e /home/kali/Desktop/2.png

DECIMAL      HEXADECIMAL  DESCRIPTION
---
0            0x0         PNG image, 1209 x 1920, 8-bit/color RGBA, non-i
nterlaced
41          0x29       Zlib compressed data, default compression
1551846     0x17ADE6   Zip archive data, at least v2.0 to extract, com
pressed size: 49, uncompressed size: 44, name: key_2.txt
1552025     0x17AE99   End of Zip archive, footer length: 22
```

得到key

第三个是修改图片高度



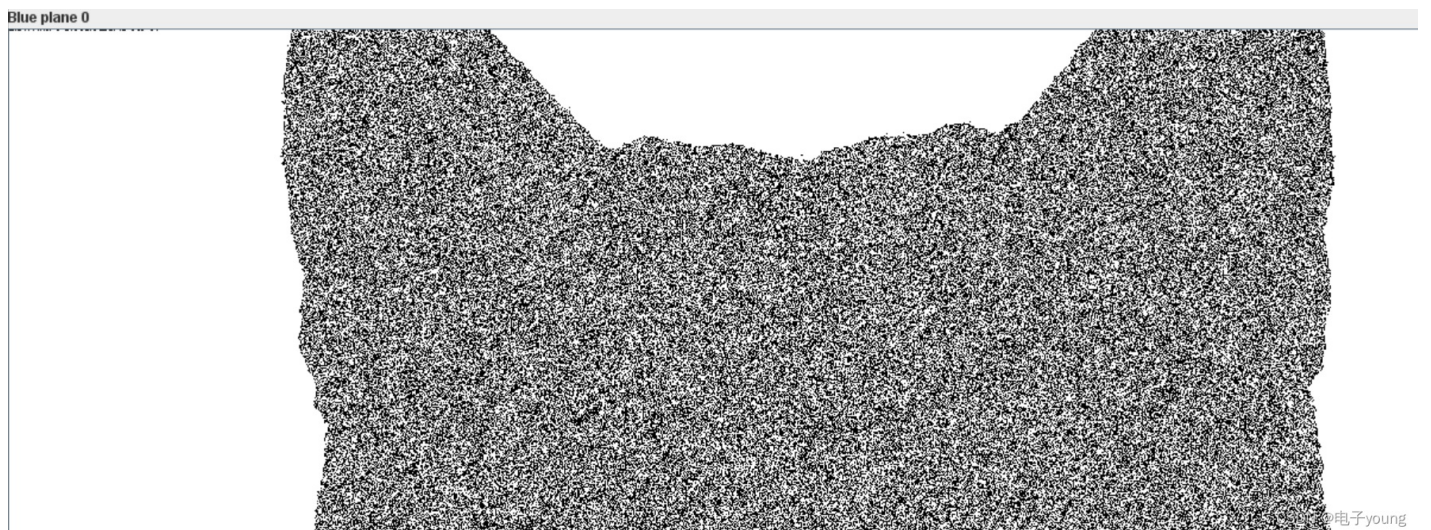
CSDN @电子young

可以看出下方缺少，winhex修改高度得到key

b1ndog

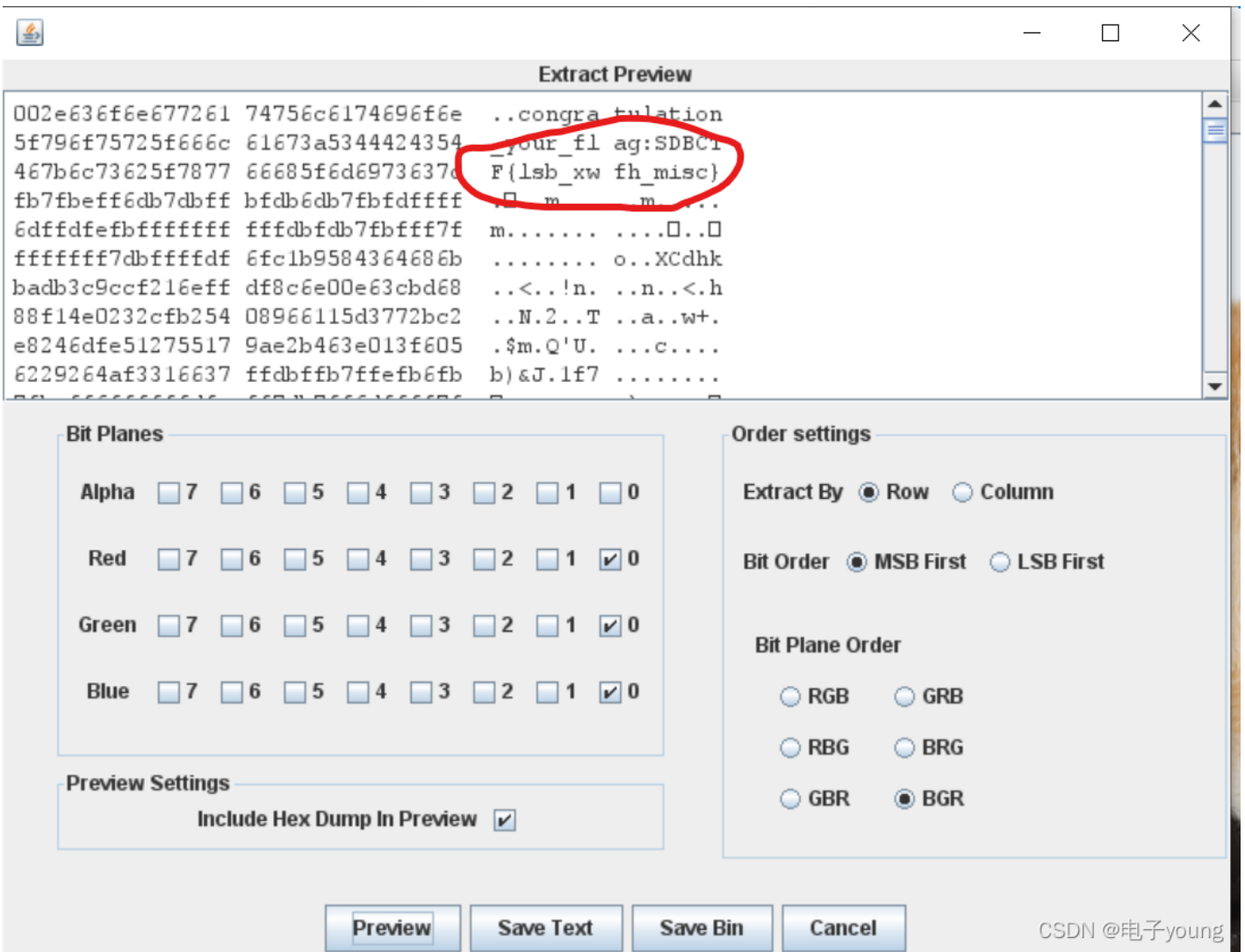


lsb隐写



上方明显有东西

一开始RGB发现没有啥东西，然后看到反向的{ }猜测是BGR，得到flag



CRYPTO

affine

提示是affine的话就试试仿射

yopf{kj_bjz_hcjin_pyytcr_dtqmre?}

仿射密码

Affine Cipher

vopf{ki_biz_hcin_pvyter_dtamre?}

7

15

加密

解密

flag{do_you_know_affine_cipher?}

CSDN @电子young

Signin

$b \lambda \alpha \gamma \{ \forall \uplus \nu _ \Lambda \alpha T \epsilon \Xi _ M \approx \triangleleft \hbar \}$

CSDN @电子young

一开始懵了，然后看到图片名字symbols，一直找，找到了Latex symbol这个东西，对了很久很久很久，总算对照出来了

±	\pm	∩	\cap	◊	\diamond	⊕	\oplus
∓	\mp	∪	\cup	△	\bigtriangleup	⊖	\ominus
×	\times	⊕	\uplus	▽	\bigtriangledown	⊗	\otimes
÷	\div	∏	\sqcap	◀	\triangleleft	⊘	\oslash
*	\ast	∪	\sqcup	▶	\triangleright	⊙	\odot
*	\star	∨	\vee	◁	\lhd	○	\bigcirc
◦	\circ	∧	\wedge	▷	\rhd	†	\dagger
•	\bullet	\setminus	\setminus	◁	\unlhd	‡	\ddagger
·	\cdot	ℓ	\wr	▷	\unrhd	ℙ	\P
+	+	-	-				

CSDN @电子young

。其中一些，例如“和”，是通过键入相应的键盘字符产生的。其他通过下表中的命令获取：

♣	#club	↔	#Leftrightarrow	←	#leftarrow	∇	#nabla	â	#aa
∅	#voidn		#void8	⊗	#otimes	↙	#downleftarrow	/	#/
≤	#leq	h	#hbar	←	#Leftarrow	—	#topbar	\	#backslash
≈	#approx	◆	#diamond	∏	#prod		#arcbar	·	#upoint
∈	#in	ℵ	#aleph	□	#Box	↑	#uparrow	∂	#partial
⊃	#supset	≥	#geq	∥	#parallel	⊕	#oplus	⌋	#corner
∩	#cap	≠	#neq	♥	#heart	↑	#Uparrow	}	#lbar
©	#ocopyright	∉	#notin	ℵ	#Jgothic	∑	#sum	⏟	#bottombar
™	#trademark	⊆	#subsetq	<	#LT	⊥	#perp	→	#rightarrow
×	#times	∪	#cup	≡	#equiv	∀	#forall	√	#surd
•	#bullet	©	#copyright	⊂	#subset	♠	#spade	⇒	#Rightarrow
f	#voidb	™	#void3	⊇	#supseteq	℞	#Rgothic	∫	#int
”	#doublequote	÷	#divide	^	#wedge	>	#GT	⊙	#odot
	#lbar	°	#circ	®	#oright	∞	#propto	∃	#exists
\	#arcbottom	∞	#infty	Å	#AA	⊄	#notsubset	+	#plus
↓	#downarrow	∠	#angle	±	#pm	∅	#oslash	-	#minus
↔	#leftrightarrow		#cbar	∓	#mp	∨	#vee		
↓	#Downarrow	(#arctop	⋯	#3dots	⊕	#void1		

CSDN @电子young

```

δ \delta
ε \epsilon

```

flag{fun_LaTeX_Math}

Ez_classical

```
Opkvr eii va seac geopy xb hzkdbgp gvrrnnuvzekmjv gw glvvz ixi bvxeiqfegmfrn xavfyzrb bni plrpgmtr - eeh z
```

古典密码的话应该是维吉尼亚密码

但是我也不知道密钥是啥，最后找到个网站 [Vigenere Solver - www.guballa.de](http://www.guballa.de) 在线爆破

Input

Cipher Text:

```
Opkvr eii va seac geopy xb hzkdbgp gvrrnnumvzekmjv gw glvvz ixi
bvxeiqfegmfrn xavfyzrb bni plrpgmtrk - eeh zdkvl trxc xuwrv
umankvrrk vdaqw.Lslv atgk vw
QqspF3xUhNpaEslyfc8bx19ID1F3op9bCRfnfliYjL==. ZYE vw yimm zs
uict twa qnrrkz guye hzkdbgp emjo rqzl n vrrbm uj pegewqrmgmvw
vvj iktvvoqyi vrtpplort mexzoxegiu vdaq qnrrkzukrg, xyvziz
hrxvgoqur nru vzavsawv, mymtxvxp eil ggpijw hitetidiib, grq
jiepl vvrzvroqur.
```

Cipher Variant:

Classical Vigenere ▾

Language:

German ▾

Key Length:

3-30

(e.g. 8 or a range e.g. 6-10)

Break Cipher

Clear Cipher Text

Result

Clear text [\[hide\]](#)

Clear text using key "vigenere":

```
There are as many paths to digital transformation as there are
organizations pursuing the challenge - and every path poses different
risks.Your flag is ZmxhZ3tHdWlfWmh1bl8xc19AX1B3bl9kYWxhbyEhfQ==. RSA
is here to help you manage your digital risk with a range of
capabilities and expertise including integrated risk management,
threat detection and response, identity and access management, and
fraud prevention.
```

CSDN @电子yoying

看到个base64，解码得到flag

Basic_RSA

```
n = 7168710045719540786046981944478960617029015738768368491690610753094240667686070925471276988585806662404
c = 6867458099093215876610304261510609489636746022985210210835338885537647480727039161144038016078969507780
e = 0x1001
```

在线网站分解n无效，可能是接近，调用yufu进行分解，得到p, q

```
P77 = 84668235163605133373102334532238432455270507565499112735876274741500617207761
```

```
P77 = 84668235163605133373102334532238432455270507565499112735876274741500617207713
```

```
import gmpy2 as gp
import binascii
p = gp.mpz()
q = gp.mpz()
e = gp.mpz()
c = gp.mpz()
n = p*q
phi = (p-1) * (q-1)
d = gp.invert(e, phi)
m = pow(c, d, n)
print(m)
```

转字符得到flag{Th1s_1s_B@sic_R54!}

RSA's door

```
n = 8346017033278954335944506041149709170580434736481428124547059237113366310649353060852940946910295493718
c = 2625075742310856779272450137803154469847061654258322294979102873405936370009240616237581144135014732776
e = 3
```

e特别小，那就不用低加密指数攻击

```
#python3
import RSAwienerHacker
n=
e=
d = RSAwienerHacker.hack_RSA(e,n)
if d:
    print(d)
import hashlib
flag = "flag{" + hashlib.md5(hex(d)).hexdigest() + "}"
print flag
```

或者用一些工具更方便，最后得出

```
1433763655511617498648135287891321364534462577872359854642976836020073844911403039243128
```

转字符串flag{We1c0me_To_C0ooooo0la_Gouliang_Cuppp!}

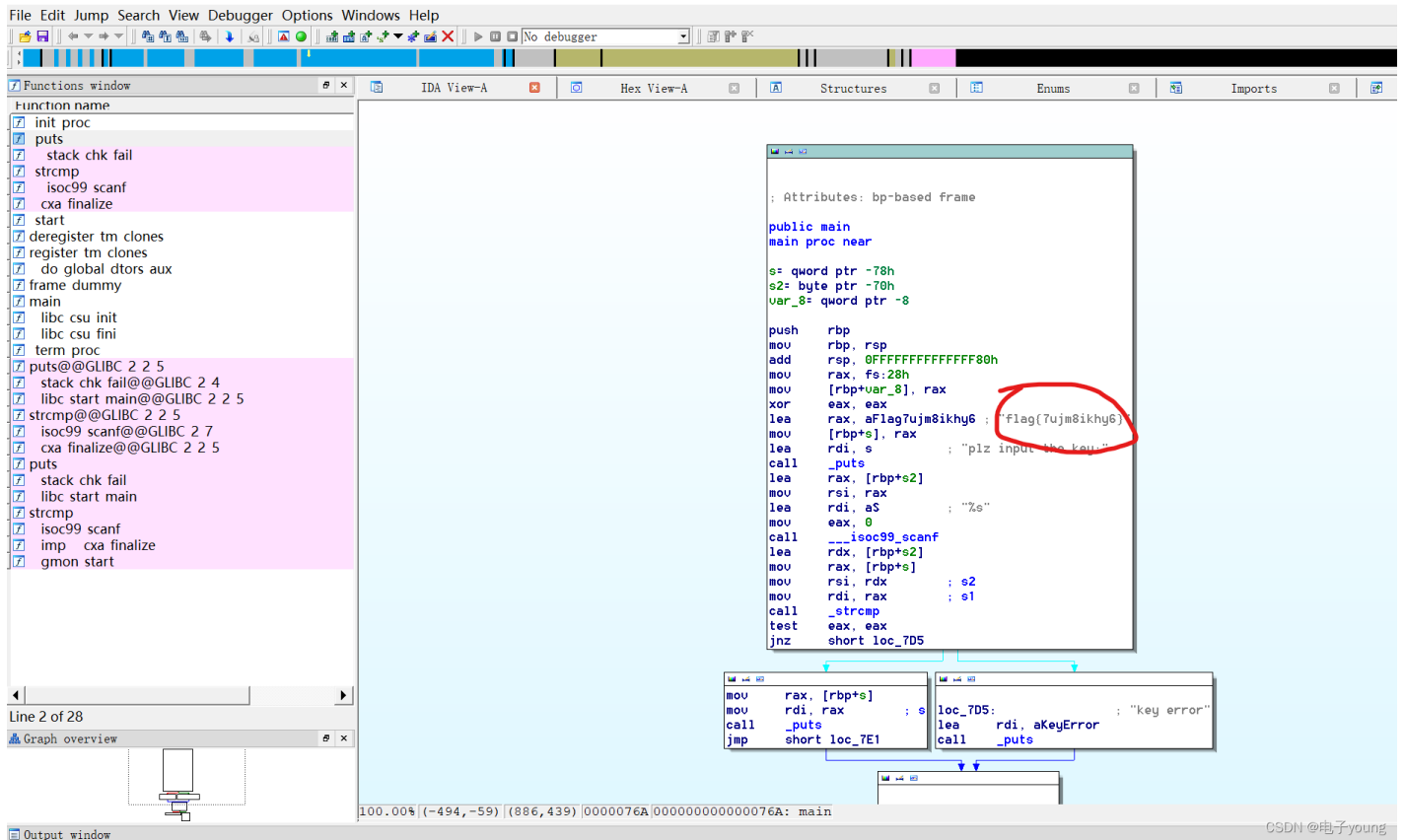


Reverse

逆向签到处

打开压缩包发现有密码，直接破解密码得出123456

ida打开文件找到flag



PWN

给出了个端口，kali中用nc进入

121.89.236.144 8889

```
root@kali: ~  
File Actions Edit View Help  
  
(root@kali)-[~]  
# nc 121.89.236.144 8889  
Welcome to Shidibei,hackers!  
ls  
Swiss_Army_knife  
bin  
dev  
flag  
lib  
lib32  
lib64  
cat flag  
flag{fcc93007-ee3a-4e0f-bb13-2465ca0d72ac}
```

CSDN @电子young

ls看看目录，cat flag得到flag

flag{fcc93007-ee3a-4e0f-bb13-2465ca0d72ac}



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)