BUUCTF-MISC-[GXYCTF2019]SXMgdGhpcyBiYXNlPw==~zip

原创

七堇墨年 于 2021-12-25 17:22:06 发布 604 收藏 1

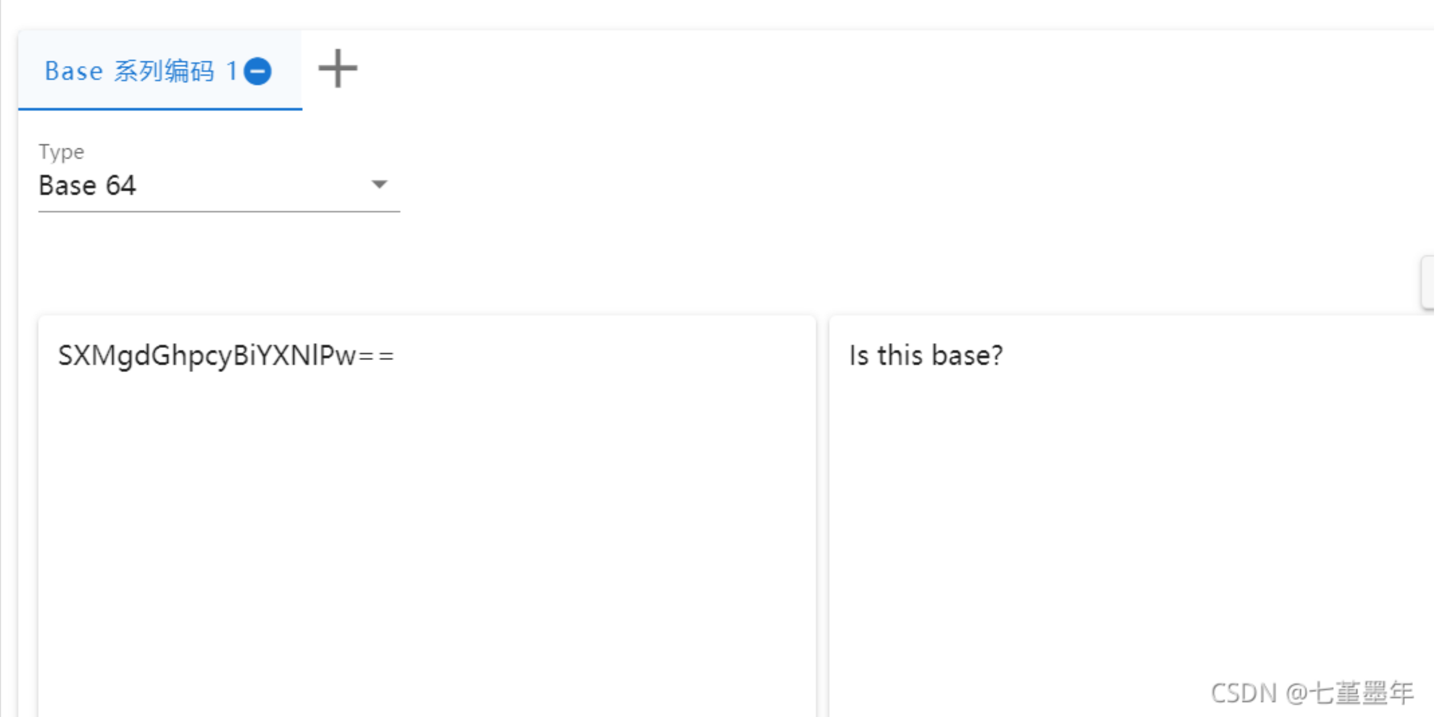
文章标签： 安全

文章目录

1.[GXYCTF2019]SXMgdGhpcyBiYXNlPw==

题目描述：得到的 flag 请包上 flag{} 提交。

解题步骤：发现题目为base64编码，解密得到Is this base?



SXMgdGhpcyBiYXNlPw==

Is this base?

打开flag.txt,发现base64隐写

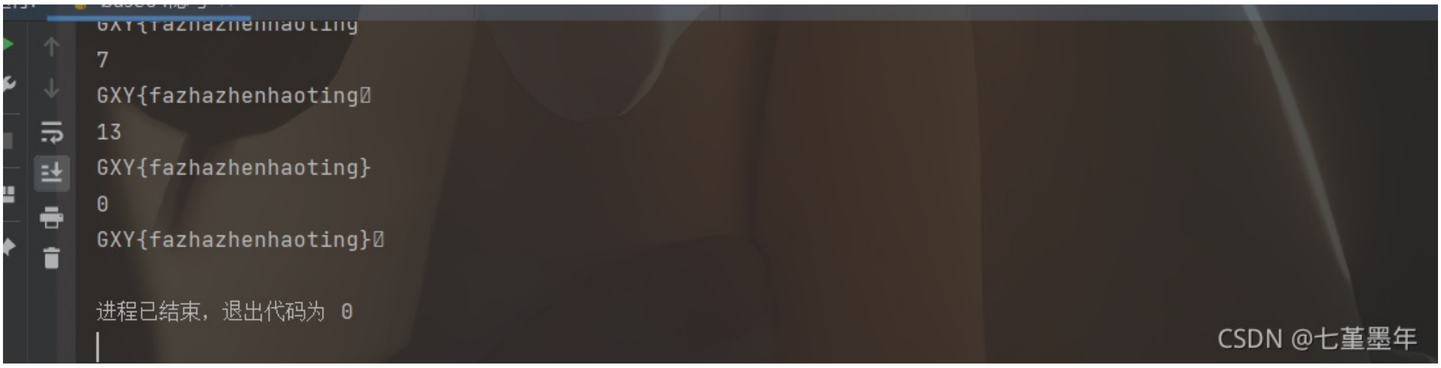
```
Q2V0dGUgbnVpdCwK
SW50ZW5hYmxlIGluc29tbmllLAp=
TGEgZm9saWUgbWUgZ3VldHRlLAo=
SmUgc3VpcyBjZSBxdWUgamUgZnVpcwp=
SmUgc3ViaXMsCt==
Q2V0dGUgY2Fjb3Bob25pZSwK
UXVpIG1lIHNjaWUgbGEgdOmUmnRlLAp=
QXNzb21tYW50ZSBoYXJtb25pZSwK
RWxsZSBtZSBkaXQsCo==
VHUgcGFpZXJhcyB0ZXMgZGVsaXRzLAp=
UXVvaSBxdSdpbCBhZHZpZW5uZSwK
T24gdHJh5Y2vbmUgc2VzIGNoYWNr25lcywK
U2VzIHBlaW5lcywK
SmUgdm91ZSBtZXMgbnVpdHMsCm==
QSBsJ2Fzc2FzeW1waG9uaWUsCl==
QXV4IHJlcXVpZW1zLAr=
VHVhbnQgcGFyIGRlcGl0LAq=
Q2UgcXVlIGplIHNlbWUsCt==
SmUgdm91ZSBtZXMgbnVpdHMsCp==
QSBsJ2Fzc2FzeW1waG9uaWUsCp==
RXQgYXV4IGJsYXNwaGVtZXMsCo==
Sidhdm91ZSBqZSBtYXVkaXMsCl==
VG91cyBjZXV4IHF1aSBzJ2FpbWVudCwK
TCdlbm5lbWksCu==
VGFwaSBkYW5zIG1vbiBlc3ByaXQsCp==
RumUmnRlIG1lcyBkZWZhaXRlcywK
U2FucyByZXBpdCBtZSBkZWZpZSwK
SmUgcmVuaWUsCq==
TGEgZmF0YWxlIGhlcmVzaWUsCh==
UXVpIHJvbmdlIG1vbiDplJp0cmUsCo==
SmUgdmV1eCByZW5h5Y2vdHJlLAp=
UmVuYeWNr3RyZSwK
SmUgdm91ZSBtZXMgbnVpdHMsCn==
QSBsJ2Fzc2FzeW1waG9uaWUsCq==
QXV4IHJlcXVpZW1zLAp=
VHVhbnQgcGFyIGRlcGl0LAq=
Q2UgcXVlIGplIHNlbWUsCo==
SmUgdm91ZSBtZXMgbnVpdHMsCm==
QSBsJ2Fzc2FzeW1waG9uaWUsCl==
RXQgYXV4IGJsYXNwaGVtZXMsCm==
Sidhdm91ZSBqZSBtYXVkaXMsCu==
VG91cyBjZXV4IHF1aSBzJ2FpbWVudCwK
UGxldXJlbnQgbGVzIHZpb2xvbnMgZGUgbWEgdmllLAp=
TGEgdmlvbGVuY2UgZGUgbWVzIGVudmllcywK
U2lwaG9ubmVlIHN5bXBob25pZSwK
RGVjb25jZXJ0YW50IGNvbmNlcnRvLAq=
SmUgam91ZSBzYW5zIHRvdWNoZXIgbGUgRG8sCo==
TW9uIHRhbGVudCBzb25uZSBmYXV4LAp=
SmUgbm9pZSBtb24gZW5udWksCo==
RGFucyBsYSBtZWxvbWFuaWUsCl==
SmUgdHVlIG1lcyBwaG9iaWVzLAq=
RGFucyBsYSBkZXNoYXJtb25pZSwK
SmUgdm91ZSBtZXMgbnVpdHMsCv==
QSBsJ2Fzc2FzeW1waG9uaWUsCn==
QXV4IHJlcXVpZW1zLAp=
VHVhbnQgcGFyIGRlcGl0LAo=
Q2UgcXVlIGplIHNlbWUsCm==
SmUgdm91ZSBtZXMgbnVpdHMsCp==
QSBsJ2Fzc2FzeW1waG9uaWUsCm==
RXQgYXV4IGJsYXNwaGVtZXMsCu==
Sidhdm91ZSBqZSBtYXVkaXMsCm==
VG91cyBjZXV4IHF1aSBzJ2FpbWVudCwK
```

```
VG91cyBjZXV4IHF1aSBzJ2FpbWVudCwK
SmUgdm91ZSBtZXMgbnVpdHMsCn==
QSBsJ2Fzc2FzeW1waG9uaWUgKGwnYXNzYXN5bXBob25pZSksCn==
Sidhdm91ZSBqZSBtYXVkaXMsCt==
VG91cyBjZXV4IHF1aSBzJ2FpbWVudA==
```

base64隐写脚本

```python
from urllib3.connectionpool import xrange


def get_base64_diff_value(s1, s2):
    base64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    res = 0
    for i in xrange(len(s2)):
        if s1[i] != s2[i]:
            return abs(base64chars.index(s1[i]) - base64chars.index(s2[i]))
    return res


def solve_stego():
    with open('flag.txt', 'rb') as f:
        file_lines = f.readlines()
        bin_str = ''
        for line in file_lines:
            steg_line = line.replace('\n', '')
            norm_line = line.replace('\n', '').decode('base64').encode('base64').replace('\n', '')
            diff = get_base64_diff_value(steg_line, norm_line)
            print diff
            pads_num = steg_line.count('=')
            if diff:
                bin_str += bin(diff)[2:].zfill(pads_num * 2)
            else:
                bin_str += '0' * pads_num * 2
            print goflag(bin_str)


def goflag(bin_str):
    res_str = ''
    for i in xrange(0, len(bin_str), 8):
        res_str += chr(int(bin_str[i:i + 8], 2))
    return res_str


if __name__ == '__main__':
    solve_stego()
```

运行得到结果GXY{fazhazhenhaoting}

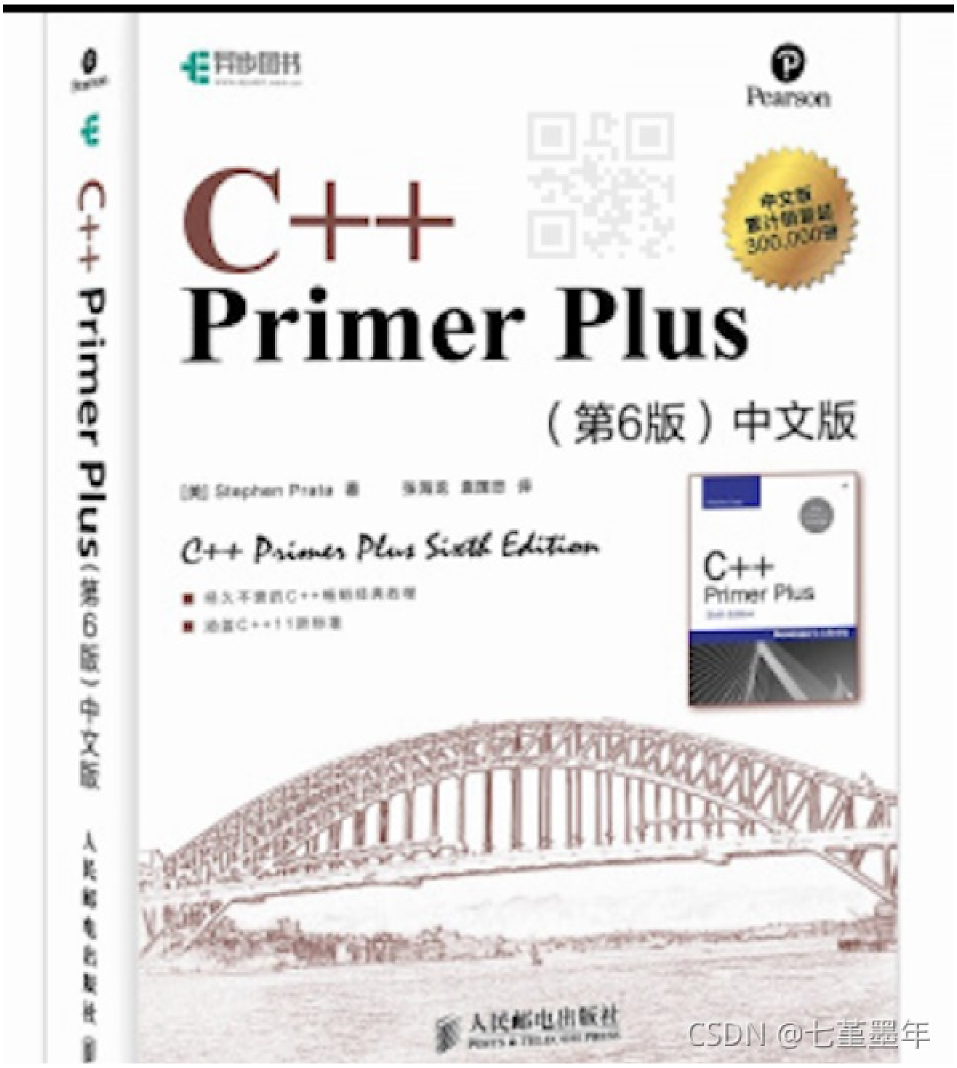


```
GXY{fazhazhenhaoting
7
GXY{fazhazhenhaoting⍰
13
GXY{fazhazhenhaoting}
0
GXY{fazhazhenhaoting}⍰

进程已结束，退出代码为 0
```

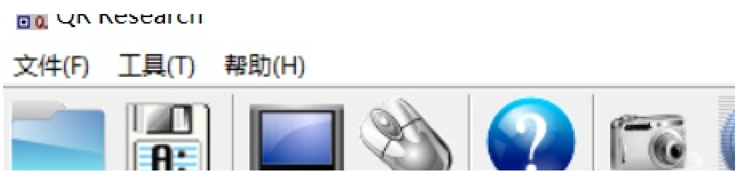
flag{fazhazhenhaoting}

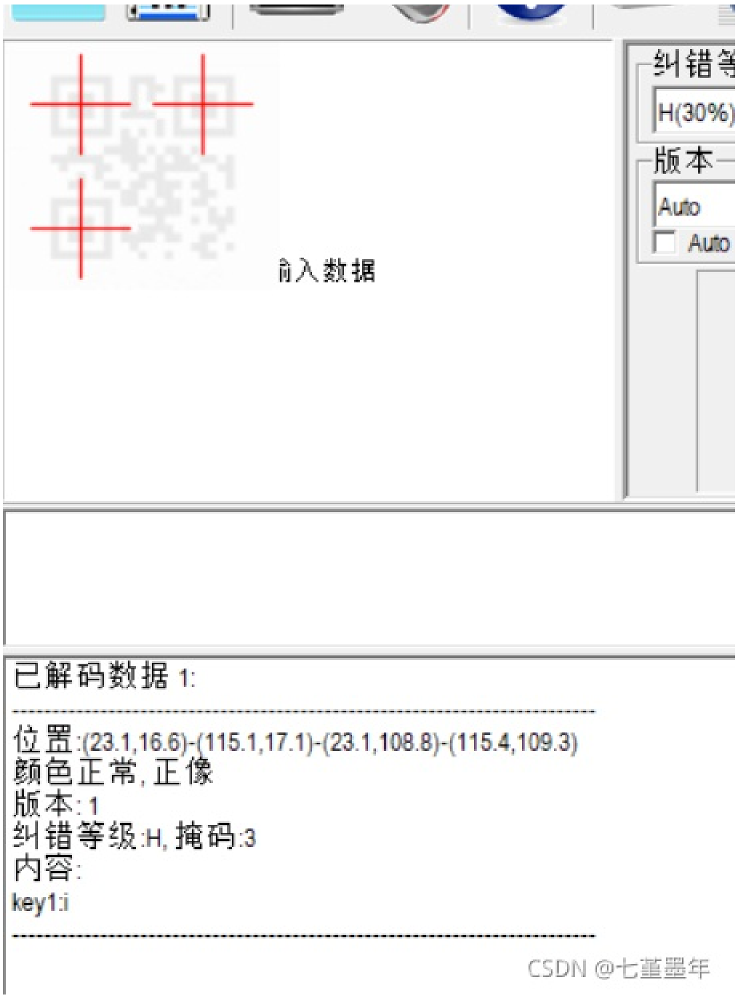
2.[RoarCTF2019]黄金6年

题目描述：得到的 flag 请包上 flag{} 提交。
解题步骤：Kinovea打开附件，在2.46s时发现一个二维码



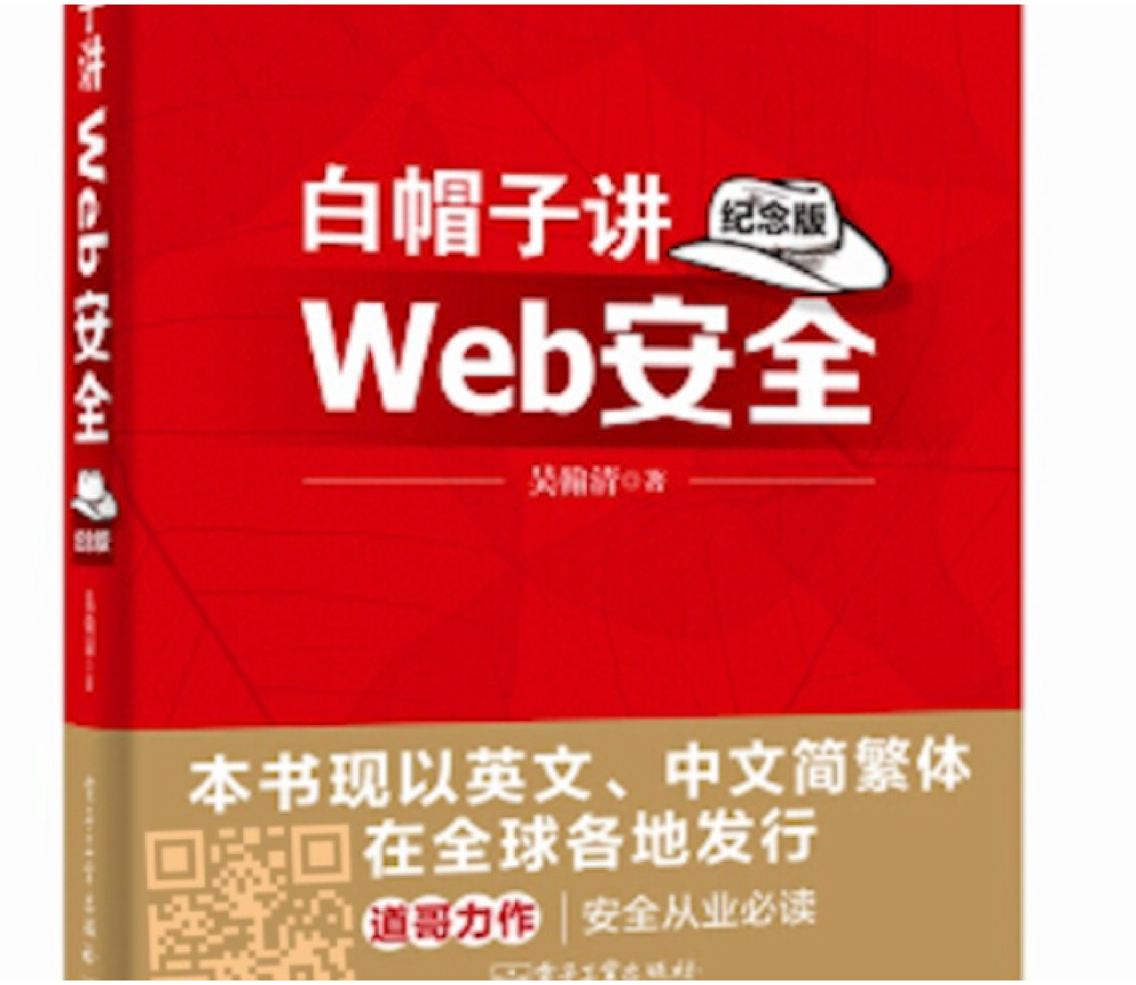
QR打开发现：key1:i



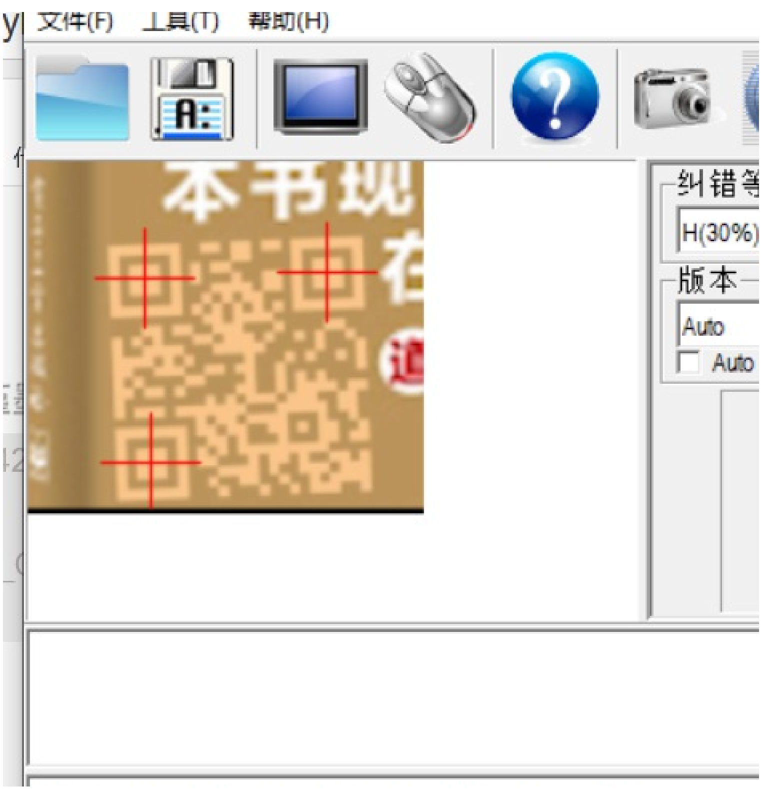


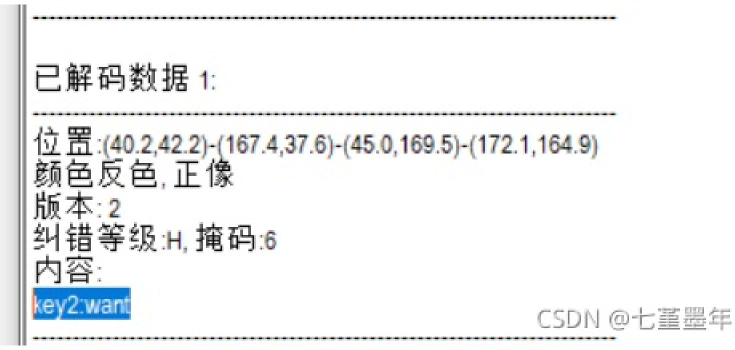
```
已解码数据 1:
--------------------------------------------------------------------------------
位置:(23.1,16.6)-(115.1,17.1)-(23.1,108.8)-(115.4,109.3)
颜色正常, 正像
版本: 1
纠错等级:H, 掩码:3
内容:
key1:i
--------------------------------------------------------------------------------
```

继续观察视频，分别在4.90s，8.16s

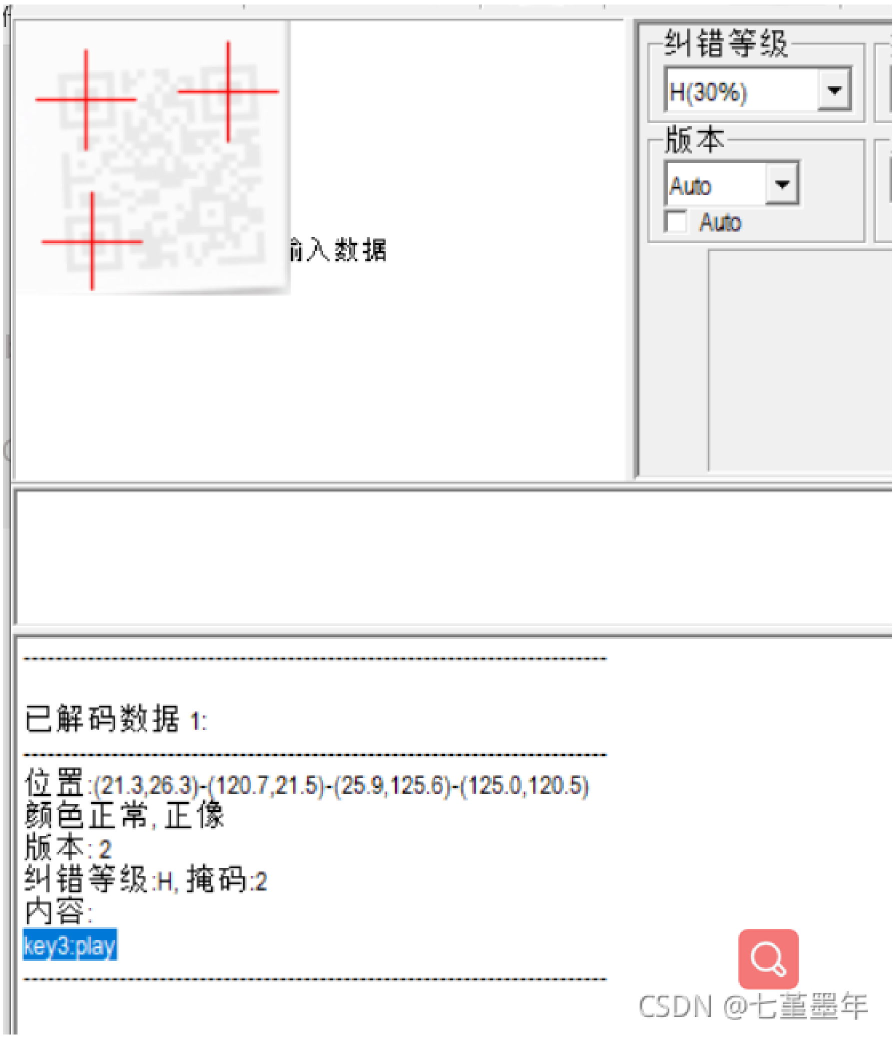


QR打开分别发现：key2:want，key3:play



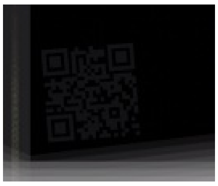


```
--------------------------------------------------------------------
已解码数据 1:
--------------------------------------------------------------------
位置:(40.2,42.2)-(167.4,37.6)-(45.0,169.5)-(172.1,164.9)
颜色反色, 正像
版本: 2
纠错等级:H, 掩码:6
内容:
key2:want
--------------------------------------------------------------------
```



```
--------------------------------------------------------------------
已解码数据 1:
--------------------------------------------------------------------
位置:(21.3,26.3)-(120.7,21.5)-(25.9,125.6)-(125.0,120.5)
颜色正常, 正像
版本: 2
纠错等级:H, 掩码:2
内容:
key3:play
--------------------------------------------------------------------
```

连接为iwantplay,感觉不像一句话，应该差了一个二维码，眼看瞎了也没找到，看别人wp才知道，还有一个，QR扫码为：

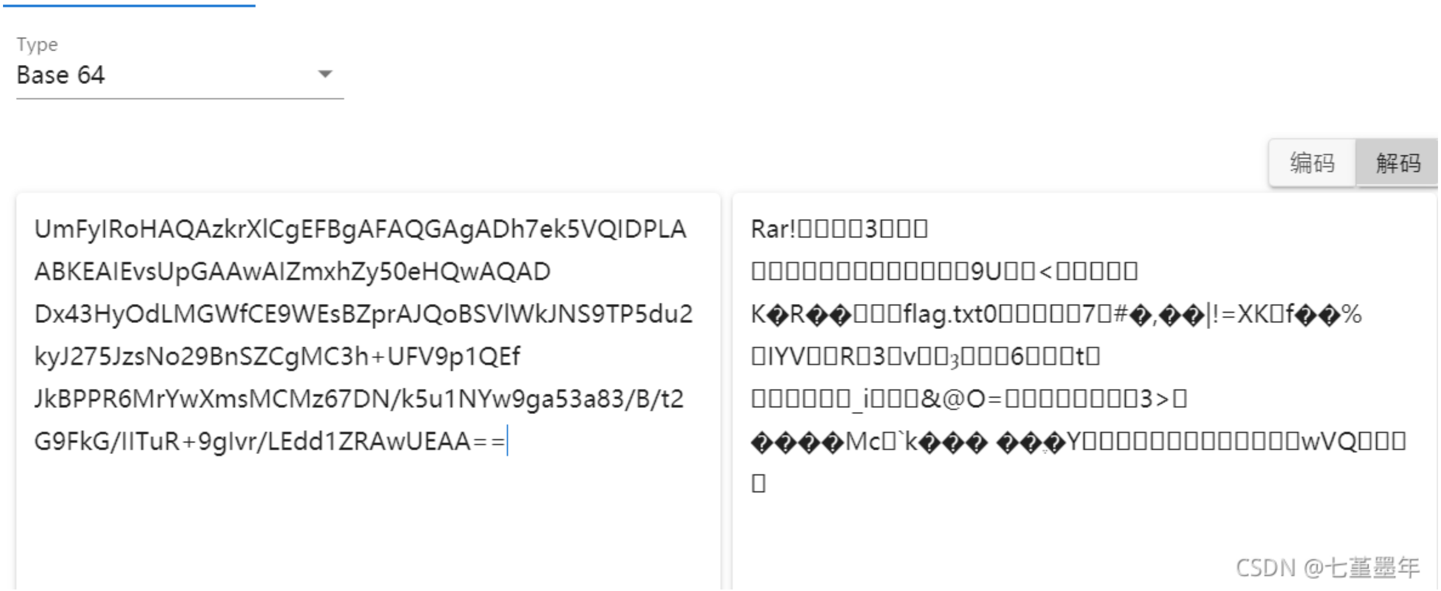


key4:ctf

连接起来为：iwantplayctf，试了下不是flag，010打开.mp4文件，在末尾发现了一段base64编码：

UmFyIRoHAQAzkrXlCgEFBgAFAQGAgADh7ek5VQIDPLAABKEAIEvsUpGAAwAIZmxhZy50eHQwAQAD
Dx43HyOdLMGWfCE9WEsBZprAJQoBSVlWkJNS9TP5du2kyJ275JzsNo29BnSZCgMC3h+UFV9p1QEf
JkBPPR6MrYwXmsMCMz67DN/k5u1NYw9ga53a83/B/t2G9FkG/IITuR+9gIvr/LEdd1ZRAwUEAA==

解码发现为rar文件，里面包含flag.txt



```
UmFyIRoHAQAzkrXlCgEFBgAFAQGAgADh7ek5VQIDPLA
ABKEAIEvsUpGAAwAIZmxhZy50eHQwAQAD
Dx43HyOdLMGWfCE9WEsBZprAJQoBSVlWkJNS9TP5du2
kyJ275JzsNo29BnSZCgMC3h+UFV9p1QEf
JkBPPR6MrYwXmsMCMz67DN/k5u1NYw9ga53a83/B/t2
G9FkG/IITuR+9gIvr/LEdd1ZRAwUEAA==
```

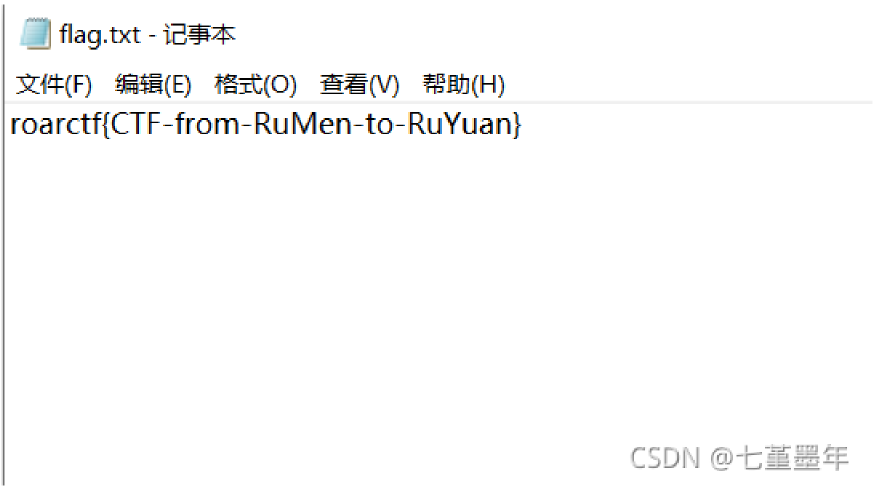
```
Rar!□□□□3□□□
□□□□□□□□□□□□□9U□□<□□□□□
K�R��□□□flag.txt0□□□□□7□#�,��|!=XK□f��%
□IYV□□R□3□v□□₃□□□6□□□t□
□□□□□□_i□□□&@O=□□□□□□□□3>□
����Mc□`k��� ���Y□□□□□□□□□□□□wVQ□□□
□
```

CSDN @七堇墨年

python代码转换为rar文件

```python
import base64
code="UmFyIRoHAQAzkrXlCgEFBgAFAQGAgADh7ek5VQIDPLAABKEAIEvsUpGAAwAIZmxhZy50eHQwAQADDx43HyOdLMGWfCE9WEsBZprAJQoBSVlWkJNS9TP5du2kyJ275JzsNo29BnSZCgMC3h+UFV9p1QEfJkBPPR6MrYwXmsMCMz67DN/k5u1NYw9ga53a83/B/t2G9FkG/IITuR+9gIvr/LEdd1ZRAwUEAA=="
r=base64.b64decode(code)
test_file=open("flag.rar","wb")
test_file.write(r)
test_file.close()
```

运行得到压缩包，用iwantplayctf为密码进行解压，发现flag.txt，打开发现flag：roarctf{CTF-from-RuMen-to-RuYuan}



roarctf{CTF-from-RuMen-to-RuYuan}

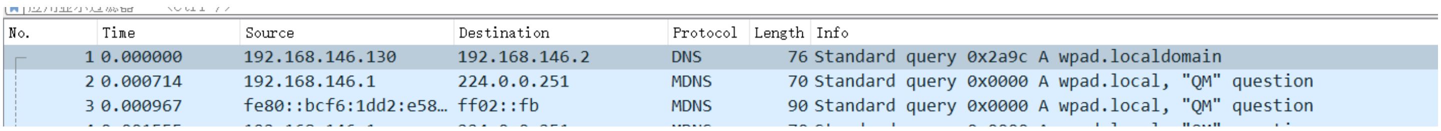
CSDN @七堇墨年

flag{CTF-from-RuMen-to-RuYuan}

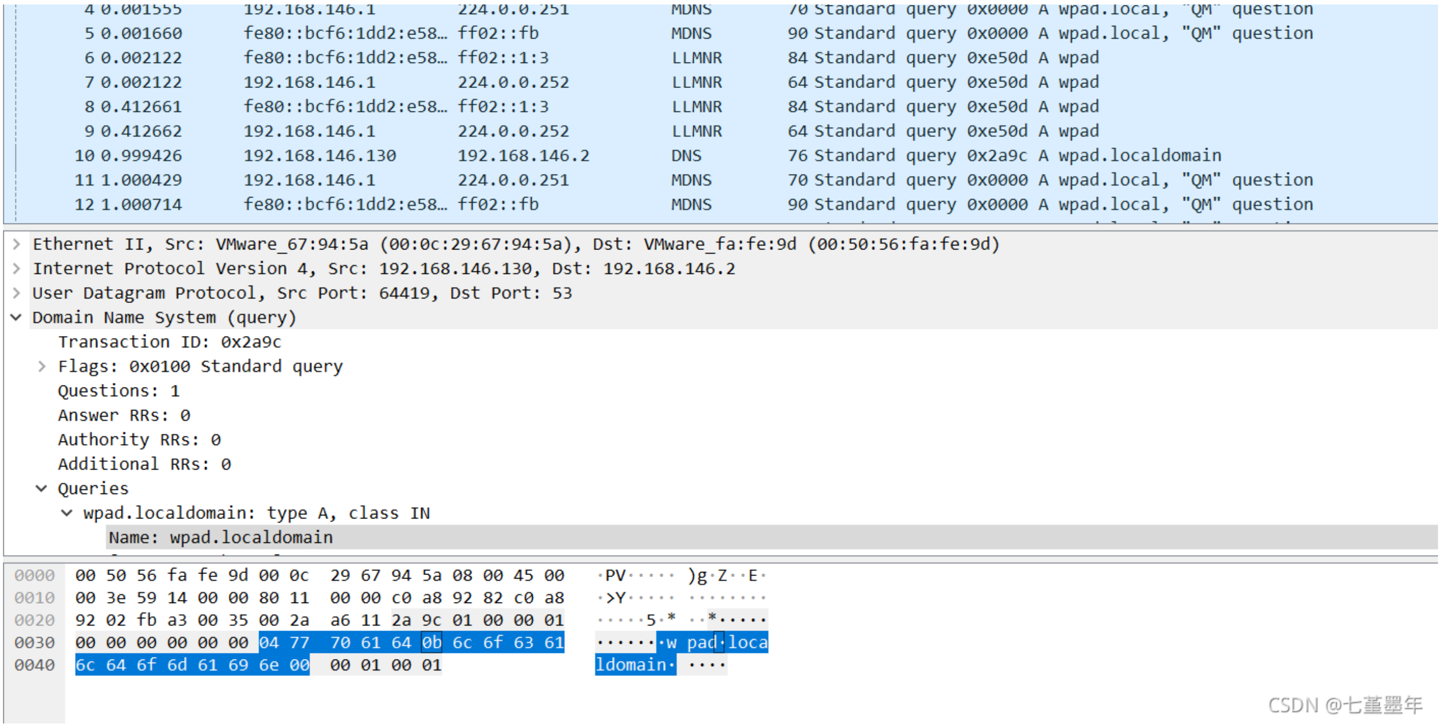
3.[ACTF新生赛2020]swp

题目描述：得到的 flag 请包上 flag{} 提交。

解题步骤：打开wget.pcapng



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.146.130	192.168.146.2	DNS	76	Standard query 0x2a9c A wpad.localdomain
2	0.000714	192.168.146.1	224.0.0.251	MDNS	70	Standard query 0x0000 A wpad.local, "QM" question
3	0.000967	fe80::bcf6:1dd2:e58…	ff02::fb	MDNS	90	Standard query 0x0000 A wpad.local, "QM" question



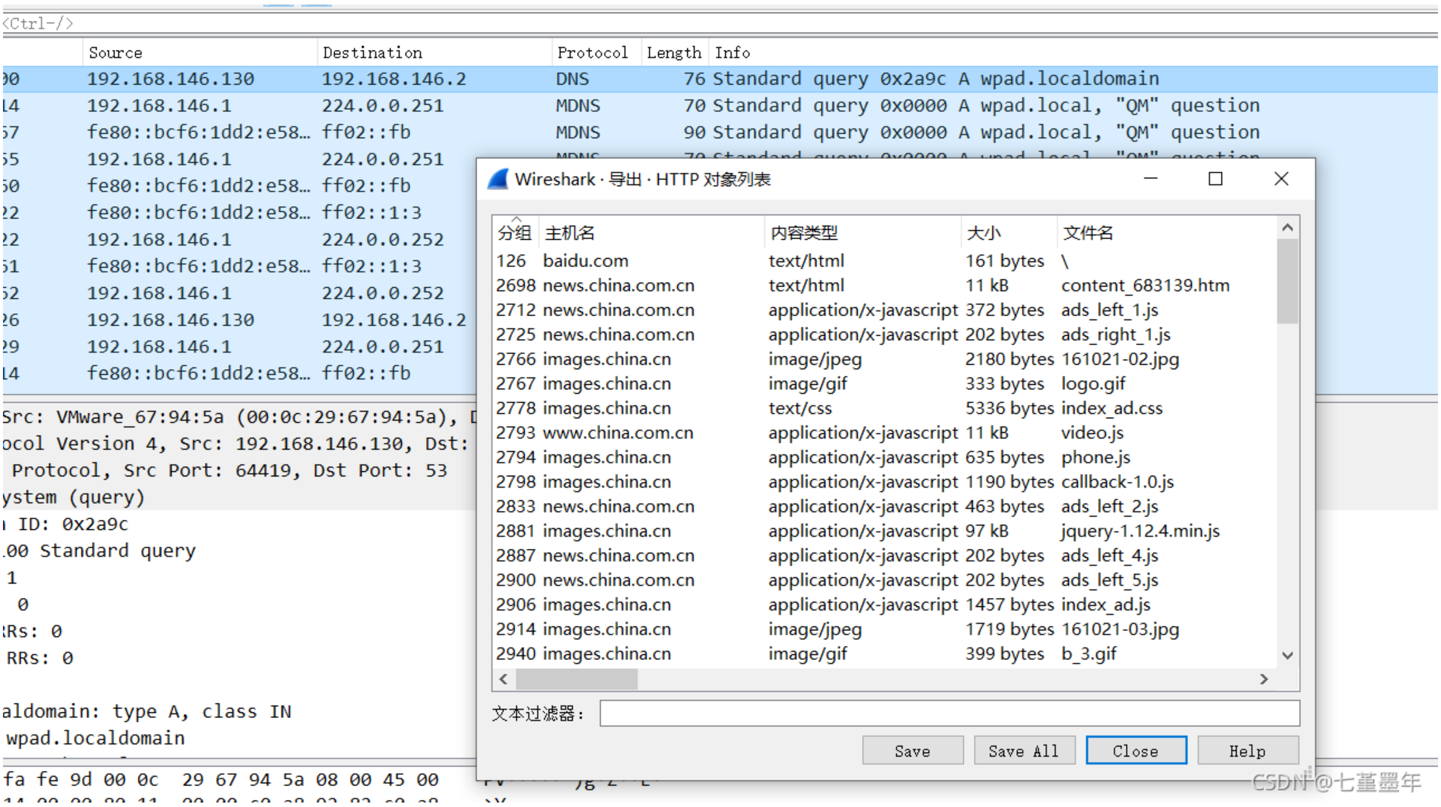
```
    4 0.001555     192.168.146.1          224.0.0.251          MDNS         70 Standard query 0x0000 A wpad.local, "QM" question
    5 0.001660     fe80::bcf6:1dd2:e58… ff02::fb             MDNS         90 Standard query 0x0000 A wpad.local, "QM" question
    6 0.002122     fe80::bcf6:1dd2:e58… ff02::1:3            LLMNR        84 Standard query 0xe50d A wpad
    7 0.002122     192.168.146.1          224.0.0.252          LLMNR        64 Standard query 0xe50d A wpad
    8 0.412661     fe80::bcf6:1dd2:e58… ff02::1:3            LLMNR        84 Standard query 0xe50d A wpad
    9 0.412662     192.168.146.1          224.0.0.252          LLMNR        64 Standard query 0xe50d A wpad
   10 0.999426     192.168.146.130        192.168.146.2        DNS          76 Standard query 0x2a9c A wpad.localdomain
   11 1.000429     192.168.146.1          224.0.0.251          MDNS         70 Standard query 0x0000 A wpad.local, "QM" question
   12 1.000714     fe80::bcf6:1dd2:e58… ff02::fb             MDNS         90 Standard query 0x0000 A wpad.local, "QM" question
```

```
> Ethernet II, Src: VMware_67:94:5a (00:0c:29:67:94:5a), Dst: VMware_fa:fe:9d (00:50:56:fa:fe:9d)
> Internet Protocol Version 4, Src: 192.168.146.130, Dst: 192.168.146.2
> User Datagram Protocol, Src Port: 64419, Dst Port: 53
v Domain Name System (query)
    Transaction ID: 0x2a9c
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  v Queries
    v wpad.localdomain: type A, class IN
        Name: wpad.localdomain
```

```
0000  00 50 56 fa fe 9d 00 0c  29 67 94 5a 08 00 45 00   ·PV·····)g·Z··E·
0010  00 3e 59 14 00 00 80 11  00 00 c0 a8 92 82 c0 a8   ·>Y·············
0020  92 02 fb a3 00 35 00 2a  a6 11 2a 9c 01 00 00 01   ·····5·*··*·····
0030  00 00 00 00 00 00 04 77  70 61 64 0b 6c 6f 63 61   ·······wpad·loca
0040  6c 64 6f 6d 61 69 6e 00  00 01 00 01               ldomain·····
```

CSDN @七董墨年

发现应该是http，导出http

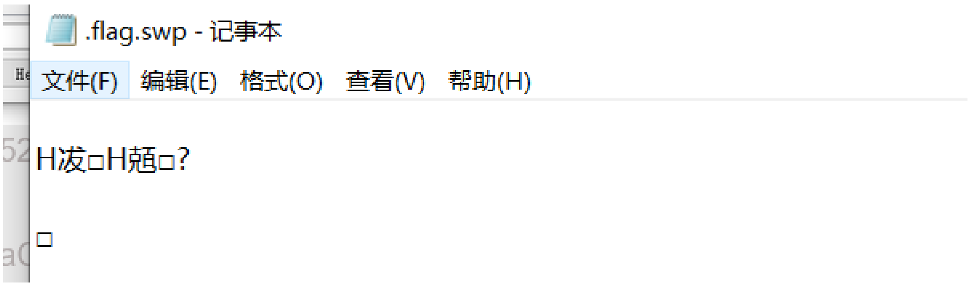


```
<Ctrl-/>
      Source                 Destination          Protocol  Length  Info
)0    192.168.146.130        192.168.146.2        DNS          76 Standard query 0x2a9c A wpad.localdomain
l4    192.168.146.1          224.0.0.251          MDNS         70 Standard query 0x0000 A wpad.local, "QM" question
57    fe80::bcf6:1dd2:e58…   ff02::fb             MDNS         90 Standard query 0x0000 A wpad.local, "QM" question
55    192.168.146.1          224.0.0.251
50    fe80::bcf6:1dd2:e58…   ff02::fb
22    fe80::bcf6:1dd2:e58…   ff02::1:3
22    192.168.146.1          224.0.0.252
51    fe80::bcf6:1dd2:e58…   ff02::1:3
52    192.168.146.1          224.0.0.252
26    192.168.146.130        192.168.146.2
29    192.168.146.1          224.0.0.251
l4    fe80::bcf6:1dd2:e58…   ff02::fb

Src: VMware_67:94:5a (00:0c:29:67:94:5a), D
ocol Version 4, Src: 192.168.146.130, Dst:
 Protocol, Src Port: 64419, Dst Port: 53
ystem (query)
 ID: 0x2a9c
.00 Standard query
 1
 0
RRs: 0
 RRs: 0

aldomain: type A, class IN
 wpad.localdomain

fa fe 9d 00 0c  29 67 94 5a 08 00 45 00
```

分组	主机名	内容类型	大小	文件名
126	baidu.com	text/html	161 bytes	\
2698	news.china.com.cn	text/html	11 kB	content_683139.htm
2712	news.china.com.cn	application/x-javascript	372 bytes	ads_left_1.js
2725	news.china.com.cn	application/x-javascript	202 bytes	ads_right_1.js
2766	images.china.cn	image/jpeg	2180 bytes	161021-02.jpg
2767	images.china.cn	image/gif	333 bytes	logo.gif
2778	images.china.cn	text/css	5336 bytes	index_ad.css
2793	www.china.com.cn	application/x-javascript	11 kB	video.js
2794	images.china.cn	application/x-javascript	635 bytes	phone.js
2798	images.china.cn	application/x-javascript	1190 bytes	callback-1.0.js
2833	news.china.com.cn	application/x-javascript	463 bytes	ads_left_2.js
2881	images.china.cn	application/x-javascript	97 kB	jquery-1.12.4.min.js
2887	news.china.com.cn	application/x-javascript	202 bytes	ads_left_4.js
2900	news.china.com.cn	application/x-javascript	202 bytes	ads_left_5.js
2906	images.china.cn	application/x-javascript	1457 bytes	index_ad.js
2914	images.china.cn	image/jpeg	1719 bytes	161021-03.jpg
2940	images.china.cn	image/gif	399 bytes	b_3.gif

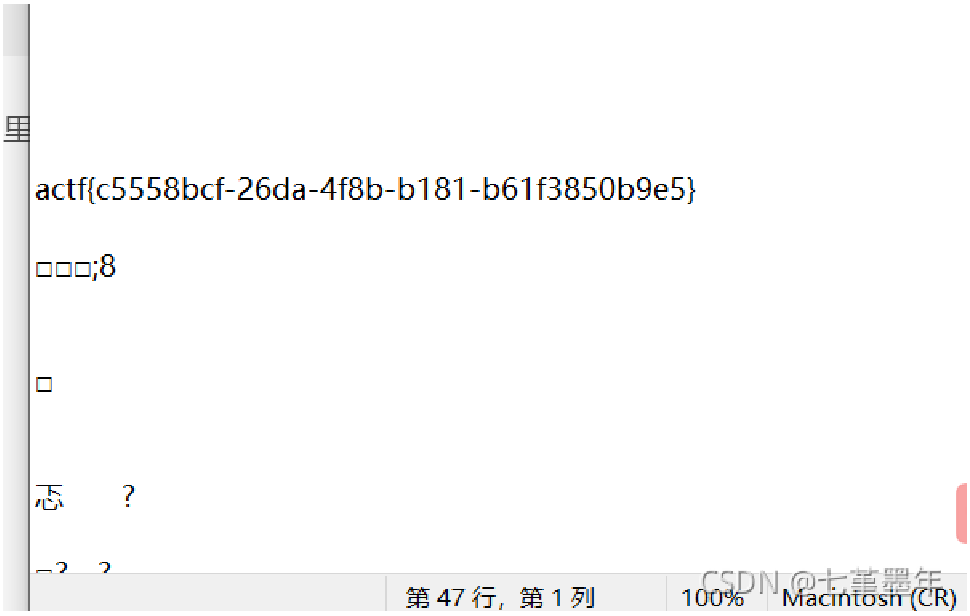
CSDN @七董墨年

发现secret.zip，解压得到.flag.swp和flag文件，看见.swp文件，隐藏文件，flag应该在里面，打开发现flag：actf{c5558bcf-26da-4f8b-b181-b61f3850b9e5}



H泼□H颉□?

□

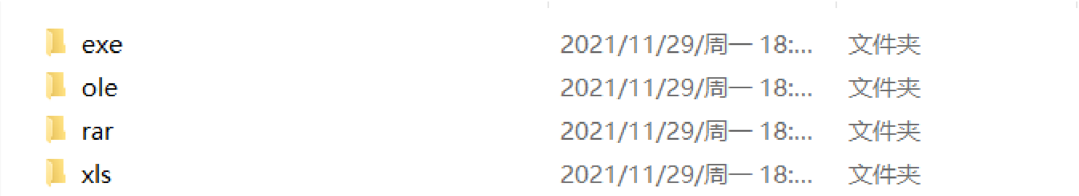


```
里
actf{c5558bcf-26da-4f8b-b181-b61f3850b9e5}

□□□;8

□

忑      ?
```

flag{c5558bcf-26da-4f8b-b181-b61f3850b9e5}

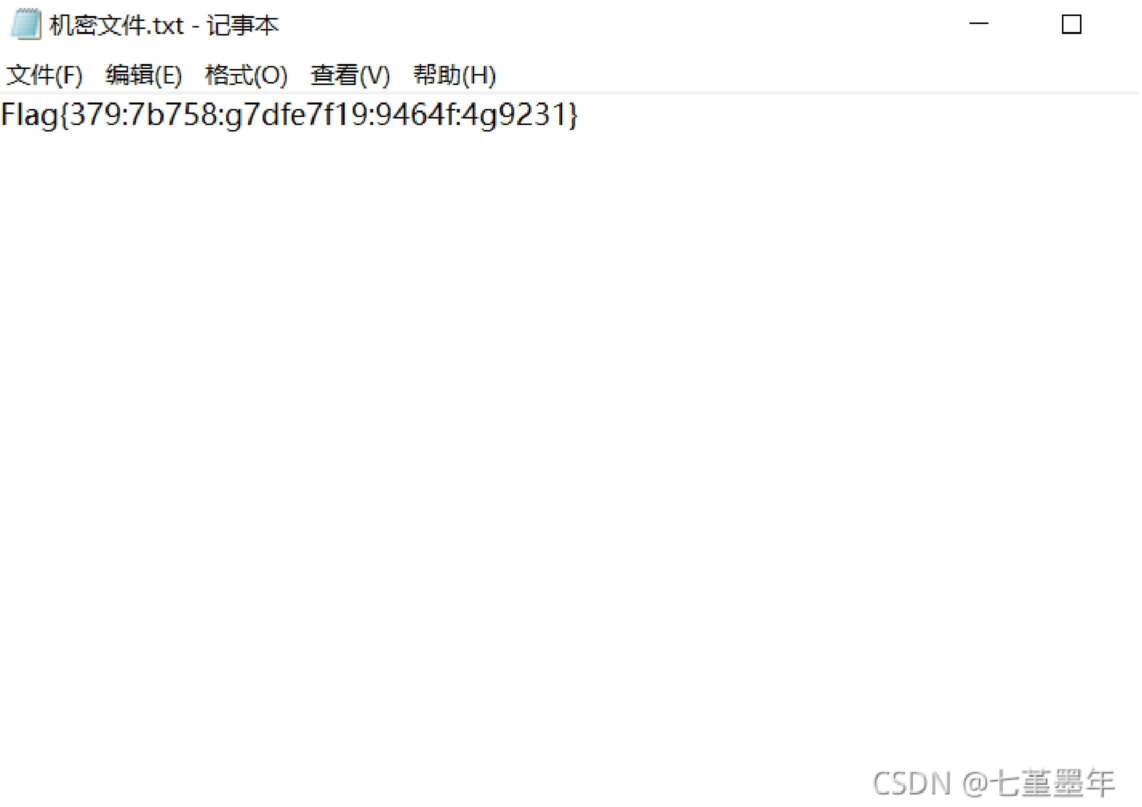
4.间谍启示录

题目描述：在城际公路的小道上，罪犯G正在被警方追赶。警官X眼看他正要逃脱，于是不得已开枪击中了罪犯G。罪犯G情急之下将一个物体抛到了前方湍急的河流中，便头一歪突然倒地。警官X接近一看，目标服毒身亡。数分钟后，警方找到了罪犯遗失物体，是一个U盘，可惜警方只来得及复制镜像，U盘便报废了。警方现在拜托你在这个镜像中找到罪犯似乎想隐藏的秘密。 注意：得到的 flag 请包上 flag{} 提交
解题步骤：发现是.iso文件，foremost文件分离下发现四个文件夹



exe	2021/11/29/周一 18:...	文件夹
ole	2021/11/29/周一 18:...	文件夹
rar	2021/11/29/周一 18:...	文件夹
xls	2021/11/29/周一 18:...	文件夹

在rar文件夹中发现压缩包，解压后发信flag.exe，勾选隐藏项目的复选框，运行flag.exe，出现机密文件.txt，打开文件，发现flag：Flag{379:7b758:g7dfe7f19:9464f:4g9231}



Flag{379:7b758:g7dfe7f19:9464f:4g9231}

flag{379:7b758:g7dfe7f19:9464f:4g9231}

5.zip

题目描述：拼在一起解下base64就有flag 注意：得到的 flag 请包上 flag{} 提交
解题步骤：打开附件发现68个压缩包，但是每个压缩包里只有4bytes大小，于是可以想到是crc爆破，代码

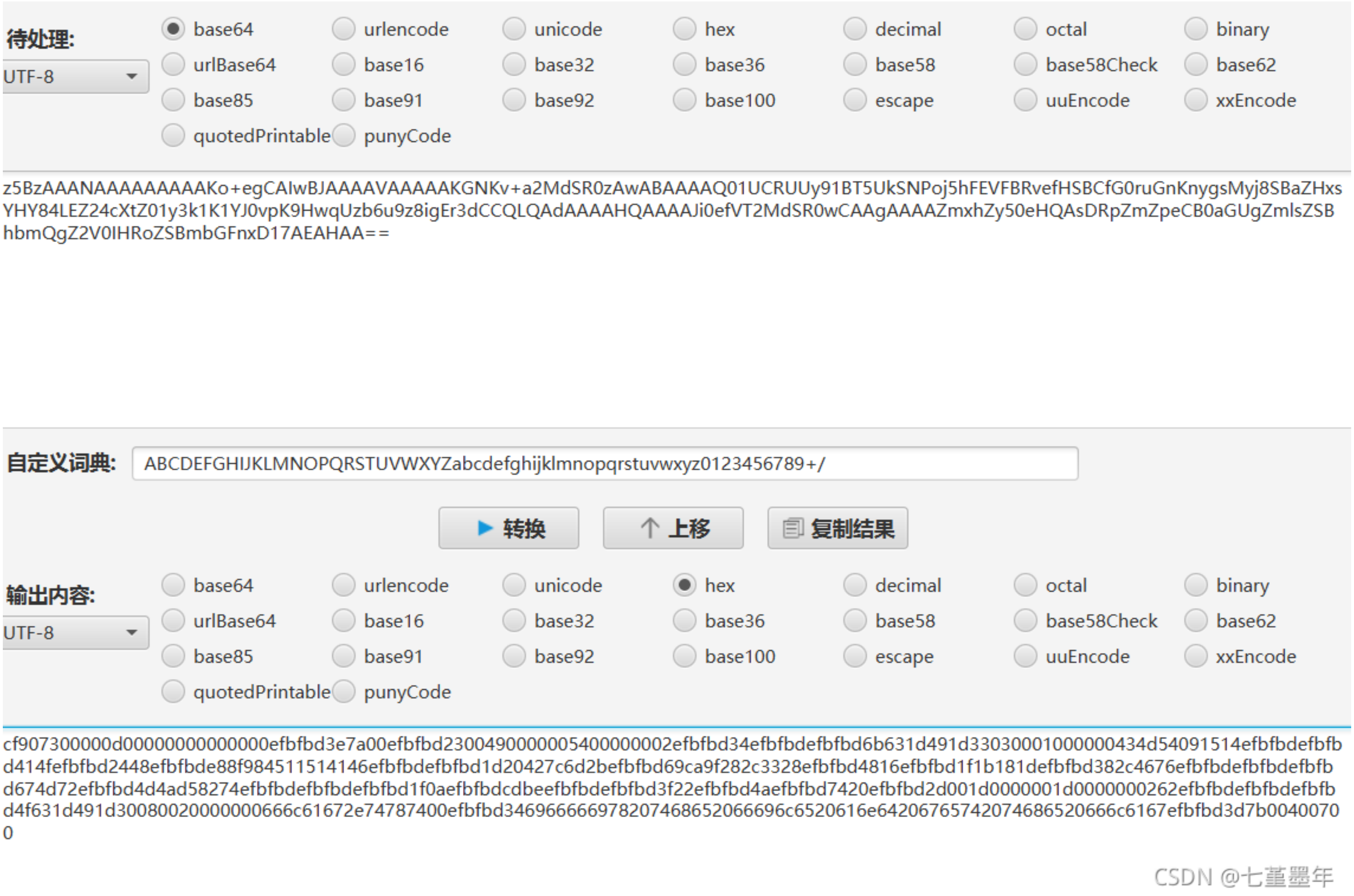
```python
import zipfile
import string
import binascii

def CrackCrc(crc):
    for i in dic:
        for j in dic:
            for k in dic:
                for h in dic:
                    s = i + j + k + h
                    if crc == (binascii.crc32(s.encode())):
                        f.write(s)
                        return

def CrackZip():
        for i in range(0,68):
            file = 'out'+str(i)+'.zip'
            crc = zipfile.ZipFile(file,'r').getinfo('data.txt').CRC
            CrackCrc(crc)

dic = string.ascii_letters + string.digits + '+/='

f = open('out.txt','w')
CrackZip()
print("CRC32碰撞完成")
f.close
```

解出base64编码：

```
`z5BzAAANAAAAAAAAAAKo+egCAIwBJAAAAVAAAAAKGNKv+a2MdSR0zAwABAAAAQ01UCRUUy91BT5UkSNPoj5hFEVFBRvefHSBCfG0ruGnKnygsMyj
8SBaZHxsYHY84LEZ24cXtZ01y3k1K1YJ0vpK9HwqUzb6u9z8igEr3dCCQLQAdAAAAHQAAAAJi0efVT2MdSR0wCAAgAAAAZmxhZy50eHQAsDRpZmZ
peCB0aGUgZmlsZSBhbmQgZ2V0IHRoZSBmbGFnxD17AEAHAA==
```

解码为十六进制：

```
cf907300000d00000000000000efbfbd3e7a00efbfbd2300490000005400000002efbfbd34efbfbdefbfbd6b631d491d3303000100000043
4d54091514efbfbdefbfbd414fefbfbd2448efbfbde88f984511514146efbfbdefbfbd1d20427c6d2befbfbd69ca9f282c3328efbfbd4816
efbfbd1f1b181defbfbd382c4676efbfbdefbfbdefbfbd674d72efbfbd4d4ad58274efbfbdefbfbdefbfbd1f0aefbfbdcdbeefbfbdefbfbd
3f22efbfbd4aefbfbd7420efbfbd2d001d0000001d0000000262efbfbdefbfbdefbfbd4f631d491d30080020000000666c61672e74787400
efbfbd3469666669782074686520666696c6520616e6420676574207468652066c6167efbfbd3d7b00400700
```



待处理: UTF-8

● base64 ○ urlencode ○ unicode ○ hex ○ decimal ○ octal ○ binary
○ urlBase64 ○ base16 ○ base32 ○ base36 ○ base58 ○ base58Check ○ base62
○ base85 ○ base91 ○ base92 ○ base100 ○ escape ○ uuEncode ○ xxEncode
○ quotedPrintable ○ punyCode

```
z5BzAAANAAAAAAAAAKo+egCAIwBJAAAAVAAAAAKGNKv+a2MdSR0zAwABAAAAQ01UCRUUy91BT5UkSNPoj5hFEVFBRvefHSBCfG0ruGnKnygsMyj8SBaZHxs
YHY84LEZ24cXtZ01y3k1K1YJ0vpK9HwqUzb6u9z8igEr3dCCQLQAdAAAAHQAAAAJi0efVT2MdSR0wCAAgAAAAZmxhZy50eHQAsDRpZmZpeCB0aGUgZmlsZSB
hbmQgZ2V0IHRoZSBmbGFnxD17AEAHAA==
```

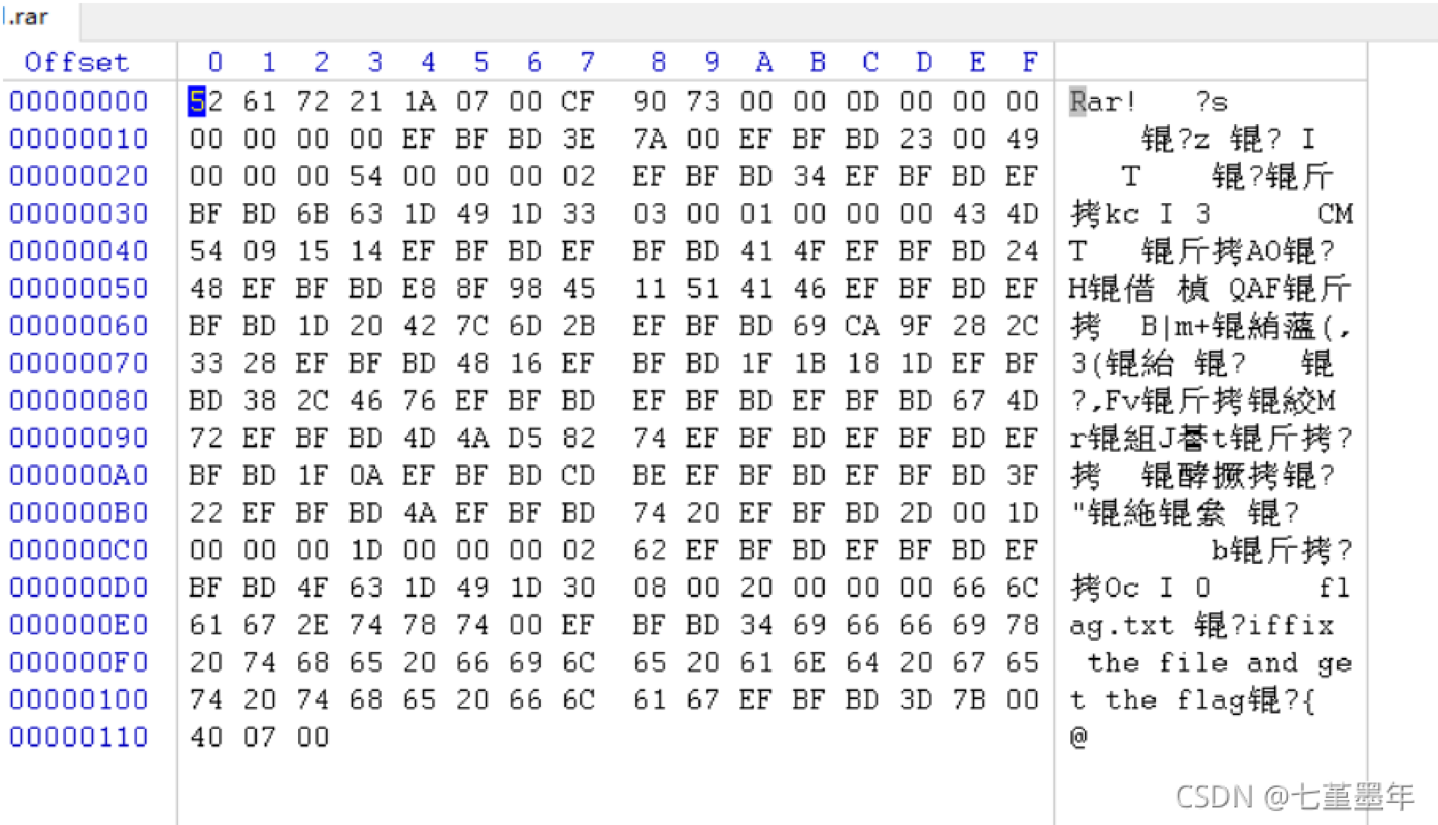
自定义词典: ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/

输出内容: UTF-8

○ base64 ○ urlencode ○ unicode ● hex ○ decimal ○ octal ○ binary
○ urlBase64 ○ base16 ○ base32 ○ base36 ○ base58 ○ base58Check ○ base62
○ base85 ○ base91 ○ base92 ○ base100 ○ escape ○ uuEncode ○ xxEncode
○ quotedPrintable ○ punyCode

```
cf907300000d00000000000000efbfbd3e7a00efbfbd2300490000005400000002efbfbd34efbfbdefbfbd6b631d491d33030001000000434d54091514efbfbdefbfb
d414fefbfbd2448efbfbde88f984511514146efbfbdefbfbd1d20427c6d2befbfbd69ca9f282c3328efbfbd4816efbfbd1f1b181defbfbd382c4676efbfbdefbfbdefbfb
d674d72efbfbd4d4ad58274efbfbdefbfbdefbfbd1f0aefbfbdcdbeefbfbdefbfbd3f22efbfbd4aefbfbd7420efbfbd2d001d0000001d0000000262efbfbdefbfbdefbfb
d4f631d491d3008002000000666c61672e74787400efbfbd346966666978207468652066696c6520616e64206765742074686520666c6167efbfbd3d7b0040070
0
```

放入winhex中，看到CF 90 73我们知道这是一个残缺的rar，补上52 61 72 21 1A 07 00 保存后解压



.rar

```
Offset     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
00000000  52 61 72 21 1A 07 00 CF  90 73 00 00 0D 00 00 00   Rar!    ?s
00000010  00 00 00 00 EF BF BD 3E  7A 00 EF BF BD 23 00 49       锟?z 锟? I
00000020  00 00 00 54 00 00 00 02  EF BF BD 34 EF BF BD EF      T      锟?锟斤
00000030  BF BD 6B 63 1D 49 1D 33  03 00 01 00 00 00 43 4D   拷kc I 3      CM
00000040  54 09 15 14 EF BF BD EF  BF BD 41 4F EF BF BD 24   T    锟斤拷AO锟?
00000050  48 EF BF BD E8 8F 98 45  11 51 41 46 EF BF BD EF   H锟借 楨 QAF锟斤
00000060  BF BD 1D 20 42 7C 6D 2B  EF BF BD 69 CA 9F 28 2C   拷  B|m+锟絠蕽(,
00000070  33 28 EF BF BD 48 16 EF  BF BD 1F 1B 18 1D EF BF   3(锟紿 锟?    锟
00000080  BD 38 2C 46 76 EF BF BD  EF BF BD EF BF BD 67 4D   ?,Fv锟斤拷锟紾M
00000090  72 EF BF BD 4D 4A D5 82  74 EF BF BD EF BF BD EF   r锟組J諅t锟斤拷?
000000A0  BF BD 1F 0A EF BF BD CD  BE EF BF BD EF BF BD 3F   拷  锟酵撅拷锟?
000000B0  22 EF BF BD 4A EF BF BD  74 20 EF BF BD 2D 00 1D   "锟絁锟絫 锟?
000000C0  00 00 00 1D 00 00 00 02  62 EF BF BD EF BF BD EF             b锟斤拷?
000000D0  BF BD 4F 63 1D 49 1D 30  08 00 20 00 00 00 66 6C   拷Oc I 0       fl
000000E0  61 67 2E 74 78 74 00 EF  BF BD 34 69 66 66 69 78   ag.txt 锟?iffix
000000F0  20 74 68 65 20 66 69 6C  65 20 61 6E 64 20 67 65    the file and ge
00000100  74 20 74 68 65 20 66 6C  61 67 EF BF BD 3D 7B 00   t the flag锟?{
00000110  40 07 00                                           @
```

保存，修改后缀为rar，在压缩文件注释中看到了flag：flag{nev3r_enc0de_t00_sm4ll_fil3_w1th_zip}