

BUUCTF MISC刷题笔记(三)

原创

[z.volcano](#) 于 2021-05-07 18:05:21 发布 2476 收藏 6

分类专栏: [# buuoj # 刷题](#)

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/weixin_45696568/article/details/116421340

版权



[buuoj](#) 同时被 2 个专栏收录

7 篇文章 1 订阅

订阅专栏



[刷题](#)

8 篇文章 0 订阅

订阅专栏

BUUOJ

Misc

[\[MRCTF2020\]pyFlag](#)

[Business Planning Group](#)

[\[ACTF新生赛2020\]剑龙](#)

[\[GWCTF2019\]huyao](#)

[\[UTCTF2020\]File Carving](#)

[\[GUET-CTF2019\]soul sipse](#)

[\[watevrCTF 2019\]Evil Cuteness](#)

[\[UTCTF2020\]sstv](#)

[\[UTCTF2020\]spectogram](#)

[我爱Linux](#)

Misc

[\[MRCTF2020\]pyFlag](#)

三个图片尾部都有额外数据

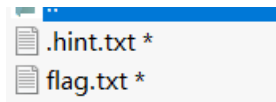
2:4750h:	53 65 63 72 65 74 20 46 69 6C 65 20 50 61 72 74	Secret File Part
2:4760h:	20 31 3A 5D 50 4B 03 04 14 00 00 00 00 00 54 88	1:]PK.....T^
2:4770h:	6C 50 00 00 00 00 00 00 00 00 00 00 00 00 0B 00	lP.....
2:4780h:	00 00 53 65 63 72 65 74 46 69 6C 65 2F 50 4B 03	..SecretFile/PK.
2:4790h:	04 14 00 09 00 08 00 74 86 6C 50 F9 1A 23 EB DFt+lPù.#èß
2:47A0h:	00 00 00 E3 00 00 00 14 00 00 00 53 65 63 72 65	...ã.....Secre
2:47B0h:	74 46 69 6C 65 2F 2E 68 69 6E 74 2E 74 78 74 D2	tFile/.hint.txtò
2:47C0h:	D0 98 8C 59 64 4C 2A 21 4D 96 C0 5F 09 FE 93 67	Ð~(YdI*!M-À .p`g
2:47D0h:	3A 9D 5D FE 1C C9 8E 6A 97 D9 F3 48 D5 FD 22 F0	:.]p.Éžj-ùóHóY"ð
2:47E0h:	36 F6 9F 89 C1 F9 3E A2 00 DC 69 B0 FD 3E 58 3D	6öY%Aù>ç.ùì°ý>X=
2:47F0h:	20 A3 6F 6B 0A 66 63 EC B8 D9 22 93 63 A4 55 35	foK.fcì.ù"``cαU5
2:4800h:	28 4C 51 2A A9 BD A8 86 09 B1 70 E5 52 D0 78 29	(LQ*@%`±.±páRÐx)
2:4810h:	3C 95 FD AB 42 97 9A DA E1 63 A9 6F FA 86 CD C9	<·ý«B-šÚác@ou+íÉ
2:4820h:	0B 34 F2 D3 68 1E A8 0F 67 4E 77 9D C6 BC 98 03	.4òÓh.``.gNw.Æ%`.
2:4830h:	22 8D E5 24 F6 3B 3E 93 11 0B 6E 2E 2E FB 38 9A	"..ãşö;>".n..û8š
2:4840h:	1F 40 47 A3 D8 63 FD 32 9F AE C9 6A 42 E2 60 A7	.@G£Øcý2ÿ@ÉjBâ`\$
2:4850h:	5D 78 44 88 1D 21 F4 AC 20 88 2C 51 FD 99 8A 22]xD^!.ô¬ ^,QY"Š"
2:4860h:	31 51 A5 DD A3 52 4E CD 82 FE 1D 0E 68 D7 B1 2D	lQ¥Y£RNÍ,p..h×±-
2:4870h:	6B 10 C5 8B 29 C4 E3 D2 5C AE 86 54 C7 44 65 23	k.Å<)ÃãÒ\@+TçDe#
2:4880h:	75 3D 42 5A FD E8 89 3D 70 B8 FB 07 A7 22 1F 1A	u=BZýè% =p.ú.\$" ..
2:4890h:	EC 11 91 48 D2 E0 6C FD EF 09 3D F2 A6 27 50 4B	ì.'Hòàlví.=ò!'PK

模板结果 - JPG.bt

名称	值	开始	大小	颜色
struct JPGFILE jpgfile		0h	248ECh	Fg:
enum M_ID SOIMarker	M_SOI (FFD8h)	0h	2h	Fg:
> struct APP0 app0		2h	12h	Fg:
> struct APP1 app1		14h	1Ah	Fg:
> struct DQT dqt[0]		2Eh	45h	Fg:
> struct DQT dqt[1]		73h	45h	Fg:
> struct SOF0 sof0		B8h	13h	Fg:
> struct DHT dht[0]		CBh	1Eh	Fg:
> struct DHT dht[1]		E9h	5Eh	Fg:
> struct DHT dht[2]		144h	1Dh	Fg:
> struct DHT dht[3]		161h	3Fh	Fg:
> struct SOS scanStart		1A0h	Eh	Fg:
> char scanData[148895]		1AEh	2459Fh	Fg:
enum M_ID EOIMarker	M_EOI (FFD9h)	2474Dh	2h	Fg:
> char unknownPadding[413]	[Secret File Part 1:]PK	2474Fh	19Dh	Fg:

https://blog.csdn.net/weixin_45696568

按顺序拼在一起，另存为zip文件，里面有俩txt，爆破得到密码 1234



hint.txt:

我用各种baseXX编码把flag套娃加密了，你应该也有看出来。
但我只用了一些常用的base编码哦，毕竟我的智力水平你也知道...像什么base36base58听都没听过
提示：0x10,0x20,0x30,0x55

flag.txt:

G&eOhGcq(ZG(t2*H8M3dG&wXIGcq(ZG&wXyG(jtG&eOdGcq+aG(t5oGjqG&eleGcq+aG)6Q<G(jrG&eOdH9<5qG&eLvGjsG&nRdH9<8rG%++qG%
_eG&eleGc+|cG(t5oG(jsG&eOH9<8rH3C_qH9<8oG&eOhGc+_bG&eLvH9<8sG&eLgCcz?cG&3sH8M3cG&eOIG%?
aG(t5oGjtG&wXxGcq+aH8V6sH9<8rG&eOhH9<5qG(<E-H8M3eG&wXIGcq(ZG)6Q<G(jtG&eOtg%+
<aG&wagG%_cG&eGcq+aG&M9uH8V6cG&eOH9<8rG(<HrGjqG&eLcH9<8sG&wUwGek2)

根据hint可以知道可能有base16、32、48、85

先用Python进行base85解码一次

```
>>> import base64
>>> n='G&e0hGcq(ZG(t2*H8M3dG&wXiGcq(ZG&wXyG(j~tG&e0dGcq+aG(t5oG(j~qG&eIeGcq+aG)6Q<G(j~rG&e0dH9<5qG&eLvG(j~sG&rnRdH9<8rG%++qG%_eG&eIeGc+|cG(t5oG(j~sG&e01H9<8rH8C_qH9<8oG&e0hGc+_bG&eLvH9<8sG&eLgGcz?cG&3|sH8M3cG&e0tG%_?aG(t5oG(j~tG&wXxGcq+aH8V6sH9<8rG&e0hH9<5qG(<E-H8M3eG&wXiGcq(ZG)6Q<G(j~tG&e0tG%+<aG&wagG%_cG&eIeGcq+aG&M9uH8V6cG&e01H9<8rG(<HrG(j~qG&eLcH9<8sG&wUwGek2)''
>>> base64.b85decode(n)
b'4755324444B4E5255494532444494E4A574751325444B514A544734325444F4E4A5547515A444474D4A5648415A54414E42574734345444B514A5647595A54514D5A5147553444474D5A5547453355434E5254475A42444B514A57494D3254534D5A5447555A4444D4E52564945324444F4E4A57475A41544952425547343254454E534447595A5444D524A5447415A55493D3D3D'
>>> |
```

https://blog.csdn.net/weixin_45696568

再base16解码一次

```
GU2DKNRUIE2DINJWGQ2TKQJTG42TONJUGQZDGMJVHAZTANBWG44TKQJVGYZTQMZQGU4DGMZUGE3UCNRTGZBDKQJWIM2TS
MZTGUZDMNRVIE2DONJWGZATIRBUG42TENS DGYZTMRJTGAZUI===
```

然后base32

```
54564A4456455A3757544231583046795A5638305833417A636B5A6C593352665A47566A4D47526C636E303D
```

再16

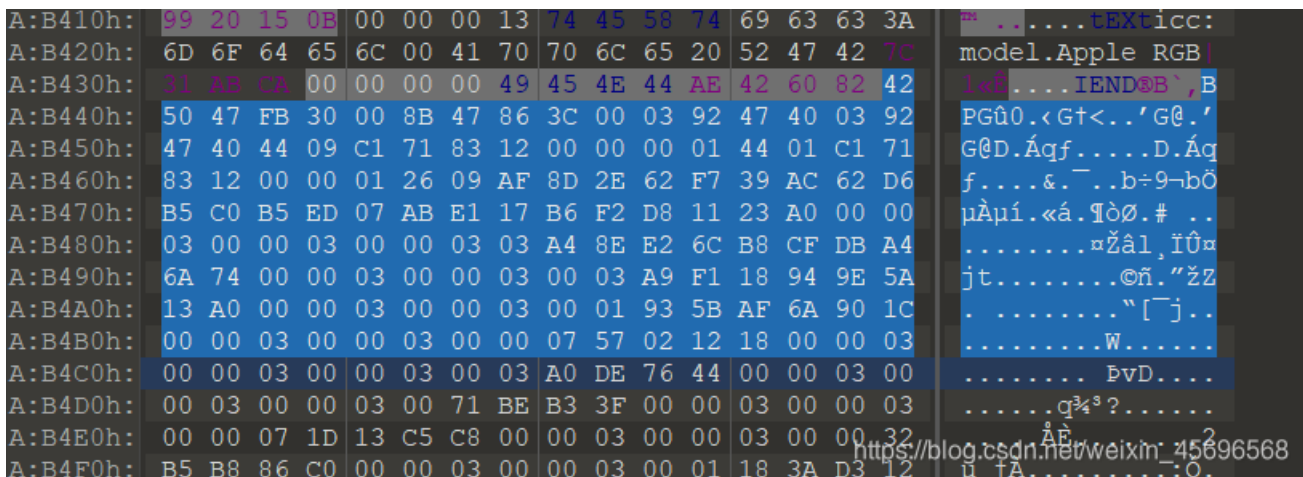
```
TVJDVEZ7WTB1X0FyZV80X3AzckZIY3RfZGVjMGRlcn0=
```

最后来一下base64

```
MRCTF{Y0u_Are_4_p3rFect_dec0der}
```

Business Planning Group

图片结尾有点东西，这个其实是bpg文件，misc入门的misc3考过



https://blog.csdn.net/weixin_45696568

提取出来保存为bpg文件，不过windows下不能直接查看，参考上面那篇博客，下载工具查看

图片里有字符串





```
YnNpZGVzX2RibGhpe0JQR19pNV9iM3R0M3JfN2g0bl9KUEd9Cg==
```

base64解码得到 `bsides_delhi{BPG_i5_b3tt3r_7h4n_JPG}`

[ACTF新生赛2020]剑龙

 hint.zip	65,503	65,478	WinRAR ZIP 压缩...	2020/1/28 0:51	76C77770
 O_O	2,206	1,252	文件	2019/12/2 19:...	79B2BD1D

先看hint.zip的内容，里面有两个文件

 hh.jpg	74,380	64,781	JPG 文件	2020/1/16 22:...	4CCB99BE
 pwd.txt	1,974	450	文本文档	2020/1/28 0:18	B74E1613

看一下txt

```
w' /= / 'm') / ~|_ // '* \n `*/ [ '_']; o=(^ ) =_3; c=(^ )=(^ )-(^ ); (^ )=(^ )=(^ )/(^ )/(^ ); (^ )
)={^ : ' ', w' / : ((w' /=3) +'_') [^ ] , ^ / : (w' / +'_')[o^_o -(^ )] , ^ / : ((^ ==3) +'_')[^ ] }; (^ ) [^ ]
] =(w' /=3) +'_') [c^_o]; (^ ) [ 'c' ] = ((^ )+'_') [ (^ )+(^ )-(^ ) ]; (^ ) [ 'o' ] = ((^ )+'_') [^ ]; (^ )=
(^ ) [ 'c' ]+(^ ) [ 'o' ]+(w' / +'_')[^ ]+ ((w' /=3) +'_') [^ ] + ((^ ) +'_') [ (^ )+(^ )]+ ((^ ==3) +'_') [^ ]
]+((^ ==3) +'_') [ (^ ) - (^ )]+(^ ) [ 'c' ]+( (^ )+'_') [ (^ )+(^ )]+ (^ ) [ 'o' ]+( (^ ==3) +'_') [^ ]; (^ ) [ '
'_ ] =(o^_o) [^ ] [^ ]; (^ )=( (^ ==3) +'_') [^ ]+ (^ ) . ^ /+( (^ )+'_') [ (^ ) + (^ )]+((^ ==3) +'_') [o^_
o -^ ]+( (^ ==3) +'_') [^ ]+ (w' / +'_') [^ ]; (^ )+(^ ); (^ ) [ ^ ]='\\'; (^ ) . ^ /=( (^ + ^ ))[o^_o -(^ )]
; (o^_o)=(w' / +'_')[c^_o]; (^ ) [^ ]='\\'; (^ ) [ '_'] ( (^ ) [ '_'] ( ^ + (^ ) [^ ]+ (^ ) [ ^ ]+(^ )+ ((o^_o)
+(o^_o))+ ((^ ) + (o^_o))+ (^ ) [ ^ ]+(^ )+ (^ )+ ((^ ) + (^ ))+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^ ))+ (^ )+ (^
^ ) [ ^ ]+(^ )+ (^ )+ (o^_o)+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^ ))+ ((^ ) + (o^_o))+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^
^ ))+ ((^ ) + (^ ))+ (^ ) [ ^ ]+(o^_o) +(o^_o))+ (o^_o)+ (^ ) [ ^ ]+(^ )+ (^ )+ (^ ) [^ ] ( ^ ) ( ' ' );
```

到在线网址解密一手，得到 **welcom3!**

AAEncode加密/解密

```
w' /= / 'm') / ~|_ // '* \n `*/ [ '_']; o=(^ ) =_3; c=(^ )=(^ )-(^ ); (^ )=(^ )=(^ )/(^ )/(^ ); (^ )
)={^ : ' ', w' / : ((w' /=3) +'_') [^ ] , ^ / : (w' / +'_')[o^_o -(^ )] , ^ / : ((^ ==3) +'_')[^ ] }; (^ ) [^ ]
] =(w' /=3) +'_') [c^_o]; (^ ) [ 'c' ] = ((^ )+'_') [ (^ )+(^ )-(^ ) ]; (^ ) [ 'o' ] = ((^ )+'_') [^ ]; (^ )=
(^ ) [ 'c' ]+(^ ) [ 'o' ]+(w' / +'_')[^ ]+ ((w' /=3) +'_') [^ ] + ((^ ) +'_') [ (^ )+(^ )]+ ((^ ==3) +'_') [^ ]
]+((^ ==3) +'_') [ (^ ) - (^ )]+(^ ) [ 'c' ]+( (^ )+'_') [ (^ )+(^ )]+ (^ ) [ 'o' ]+( (^ ==3) +'_') [^ ]; (^ ) [ '
'_ ] =(o^_o) [^ ] [^ ]; (^ )=( (^ ==3) +'_') [^ ]+ (^ ) . ^ /+( (^ )+'_') [ (^ ) + (^ )]+((^ ==3) +'_') [o^_
o -^ ]+( (^ ==3) +'_') [^ ]+ (w' / +'_') [^ ]; (^ )+(^ ); (^ ) [ ^ ]='\\'; (^ ) . ^ /=( (^ + ^ ))[o^_o -(^ )]
; (o^_o)=(w' / +'_')[c^_o]; (^ ) [^ ]='\\'; (^ ) [ '_'] ( (^ ) [ '_'] ( ^ + (^ ) [^ ]+ (^ ) [ ^ ]+(^ )+ ((o^_o)
+(o^_o))+ ((^ ) + (o^_o))+ (^ ) [ ^ ]+(^ )+ (^ )+ ((^ ) + (^ ))+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^ ))+ (^ )+ (^
^ ) [ ^ ]+(^ )+ (^ )+ (o^_o)+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^ ))+ ((^ ) + (o^_o))+ (^ ) [ ^ ]+(^ )+ ((^ ) + (^
^ ))+ ((^ ) + (^ ))+ (^ ) [ ^ ]+(o^_o) +(o^_o))+ (o^_o)+ (^ ) [ ^ ]+(^ )+ (^ )+ (^ ) [^ ] ( ^ ) ( ' ' );
```

加密 解密

welcom3! https://blog.csdn.net/welxin_45696568

得到的这个应该是密码，说明hh.jpg应该是某种隐写，测试发现是 **steghide**

```
(root@kali)~/桌面
# steghide extract -sf hh.jpg -p welcom3!
wrote extracted data to "secret.txt".
```

解出的内容如下：

想要flag吗？解出我的密文吧~
U2FsdGVkX1/7KeHVI5984OsGUVSanPfpPednHpK9IKvp0kdrxO4Tj/Q==

U2F开头，应该是 AES 或者 DES 这种，不过一般需要密码，再去找一下密码的线索，在图片的exif信息中

属性	值
说明	
标题	这里有密钥
主题	@#\$\$%^&%%\$)
分级	★☆☆☆☆
标记	
备注	你瞧这个柚子它又大又圆
来源	

DES Encode Decode

Key: @#\$\$%^&%%\$) Output: Base64

U2FsdGVkX1/7KeHVI59840sGUVSanPfPednHpK9IKvp0kdrxO4Tj/Q== think about stegosaurus

https://blog.csdn.net/weixin_45696568

这里解出来让找一下 stegosaurus，在github上找到<https://github.com/AngelKitty/stegosaurus>

这个工具我以前做题的时候用过，详细介绍可以看[这篇博客](#)

题目中给的O_O其实是一个 pyc 文件，改回后缀后使用这个工具解一下就行

```
C:\Users\...\Desktop\stegosaurus隐写>python stegosaurus.py -x 0_0.pyc
Extracted payload: flag{3teg0Sauru3_!1}
```

[GWCTF2019]huyao

给了两张一样的图片，应该是盲水印



盲水印的话我遇到过三种，具体在另一篇博客介绍过

这里试了一下发现是 频域盲水印，上脚本

```

# coding=utf-8
import cv2
import numpy as np
import random
import os
from argparse import ArgumentParser
ALPHA = 5
def build_parser():
    parser = ArgumentParser()
    parser.add_argument('--original', dest='ori', required=True)
    parser.add_argument('--image', dest='img', required=True)
    parser.add_argument('--result', dest='res', required=True)
    parser.add_argument('--alpha', dest='alpha', default=ALPHA)
    return parser
def main():
    parser = build_parser()
    options = parser.parse_args()
    ori = options.ori
    img = options.img
    res = options.res
    alpha = options.alpha
    if not os.path.isfile(ori):
        parser.error("original image %s does not exist." % ori)
    if not os.path.isfile(img):
        parser.error("image %s does not exist." % img)
    decode(ori, img, res, alpha)
def decode(ori_path, img_path, res_path, alpha):
    ori = cv2.imread(ori_path)
    img = cv2.imread(img_path)
    ori_f = np.fft.fft2(ori)
    img_f = np.fft.fft2(img)
    height, width = ori.shape[0], ori.shape[1]
    watermark = (ori_f - img_f) / alpha
    watermark = np.real(watermark)
    res = np.zeros(watermark.shape)
    random.seed(height + width)
    x = range(height / 2)
    y = range(width)
    random.shuffle(x)
    random.shuffle(y)
    for i in range(height / 2):
        for j in range(width):
            res[x[i]][y[j]] = watermark[i][j]
    cv2.imwrite(res_path, res, [int(cv2.IMWRITE_JPEG_QUALITY), 100])
if __name__ == '__main__':
    main()

```

命令: `python BlindWaterMarkplus.py --original 1.png --image 2.png --result res.png`

拿到flag{BWM_1s_c00l}



[UTCTF2020]File Carving

尾部有额外数据，提取出来另存为zip文件

```
1:F2F0h: D5 9F AB 91 FA 82 A5 D2 E6 FF 05 58 DB A2 C5 34 0ÿ«'ú,ÿ0æÿ.XÛçÄ4
1:F300h: DD 72 07 00 00 00 00 49 45 4E 44 AE 42 60 82 50 Ýr.....TEND@B`,P
1:F310h: 4B 03 04 14 00 00 00 08 00 45 81 5B 50 B3 10 F1 K.....E.[P³.ñ
1:F320h: 08 1B 0A 00 00 18 41 00 00 0D 00 1C 00 68 69 64 .....A.....hid
1:F330h: 64 65 6E 5F 62 69 6E 61 72 79 55 54 09 00 03 42 den_binaryUT...B
1:F340h: 3E 58 5E 46 3E 58 5E 75 78 0B 00 01 04 E8 03 00 >X^F>X^ux....è..
```

解压后拿到文件 `hidden_binary`，看了一下发现是ELF文件，果断放进kali里运行一下

```
(root@kali)~/桌面
└─# ./hidden_binary
zsh: 权限不够: ./hidden_binary

(root@kali)~/桌面
└─# chmod u+x hidden_binary

(root@kali)~/桌面
└─# ./hidden_binary
Ah, you found me!
utf1ag{2f9e9adc2ad89c71da48cabe90a121c0}
```

这里遇到一个小问题，但是顺利解决了，同时学到了一个命令

`chmod u+x somefile` 只授予这个文件的所属者执行的权限

[GUET-CTF2019]soul sipse

下载得到out.wav，拿Audacity分析下，没啥东西

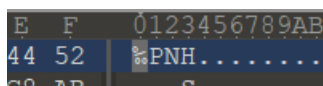
那应该就是隐写了，一般情况下是 `mp3stego`，不过这题居然是 `steghide`，属实少见

这里没有密码，解出一个txt

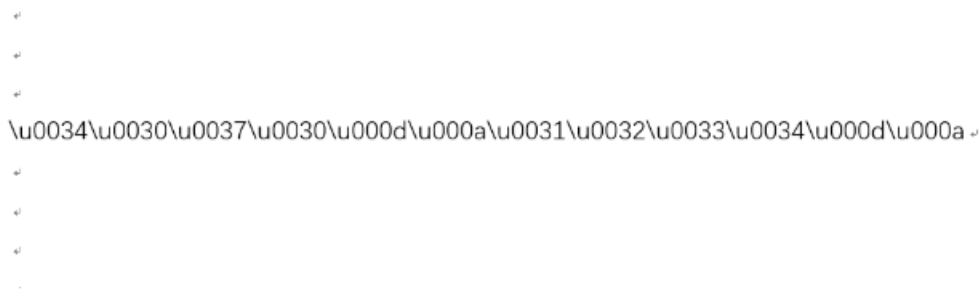
```
(volcano@kali)~/桌面
└─$ steghide extract -sf out.wav
Enter passphrase:
wrote extracted data to "download.txt".
```

<https://share.weiyun.com/5wWTIN3>

把文件下载下来，显示异常，原来是文件头有问题



修复好文件头，图片内容如下



unicode解码一下

两个数加起来就是flag: `flag{5304}`

[watevrCTF 2019]Evil Cuteness

尾部有额外数据，另存为zip

```
5460h: 9E E6 BD 5E A9 C6 72 4B 4F B1 E5 08 B7 B5 D0 71 (žæ*^@ErKO±å.µĐq
5470h: E2 46 61 94 5C 8F 5C 57 09 54 2A 39 F5 00 D7 AB âFa"\".\W.T*9đ.×«
5480h: D5 CF E6 1F A1 FF D9 50 4B 03 04 14 00 00 00 08 Ōİæ.ĵÛPK.....
5490h: 00 66 94 8D 4F 74 DF E4 E6 28 00 00 00 2A 00 00 .f".Otßæ(...*..
54A0h: 00 03 00 1C 00 61 62 63 55 54 09 00 03 E0 D9 F3 .....abcUT...àÛó
54B0h: 5D A3 D8 F3 5D 75 78 0B 00 01 04 D0 07 00 00 04 ]fðó]ux...Đ....
54C0h: D0 07 00 00 2B 4F 2C 49 2D 2B AA 36 CF 30 31 37 Đ...+O,I-+ª6İ017
54D0h: 8D 37 49 36 2F 35 C9 C9 A9 8C 2F 32 06 53 C9 A5 .7I6/5ÉÉ@E/2.SÉ¥
54E0h: E6 C6 F1 E6 19 06 A5 66 19 B5 5C 00 50 4B 01 02 æEñæ..¥f.µ\PK..
```

解压得到无后缀文件abc

```
abc
1 watevr{7h475_4c7u4lly_r34lly_cu73_7h0u6h}
2
```

就这...

[UTCTF2020]sstv

又是一个wav，百度一下题目名sstv，发现这玩意有点神奇啊

+ | ★ 收藏 | 13 | 1

慢扫描电视

编辑 | 讨论 | 上传视频

同义词 sstv一般指慢扫描电视

本词条由“科普中国”科学百科词条编写与应用工作项目 审核。

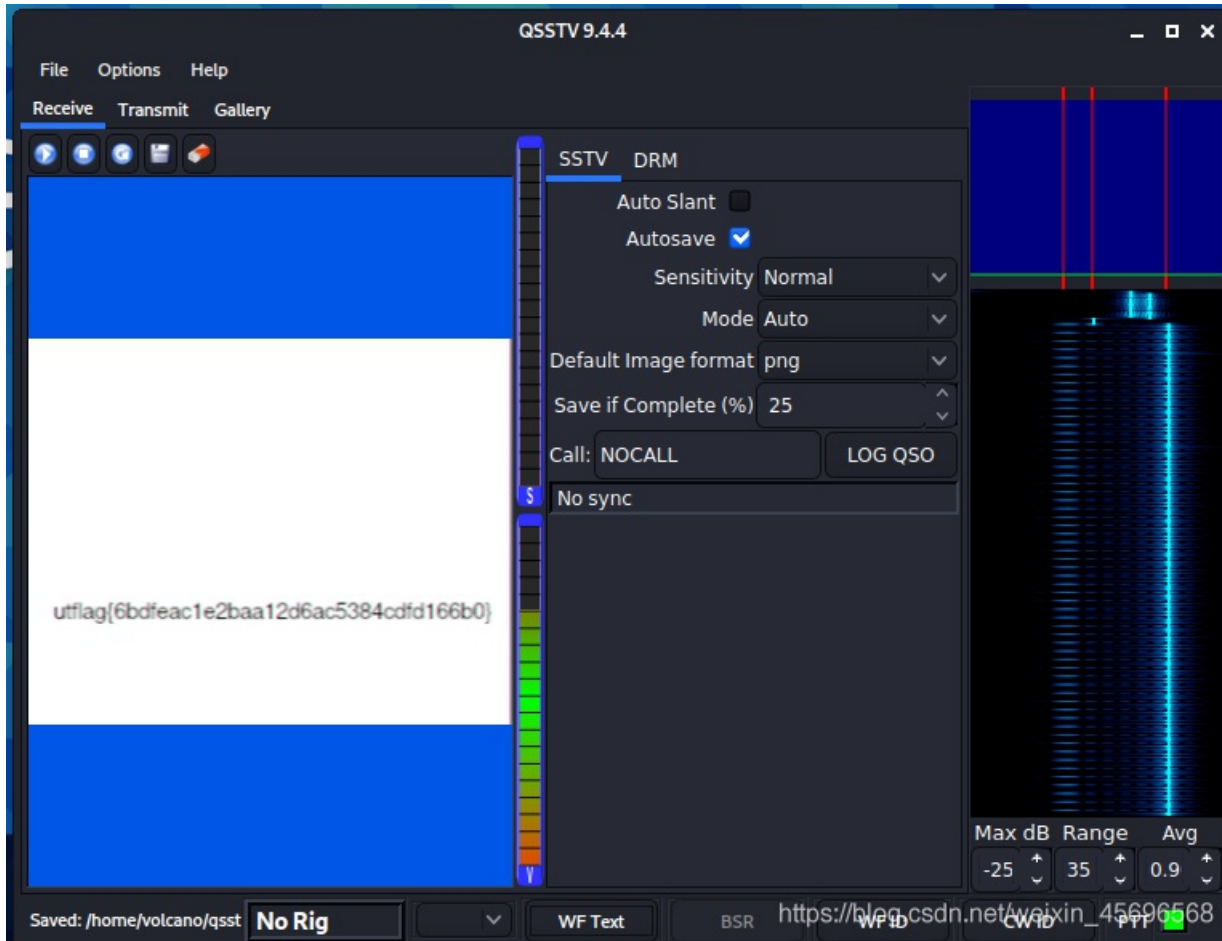
慢扫描电视（Slow-scan television）是**业余无线电**爱好者的一种主要图片传输方法，慢扫描电视通过**无线电**传输和接收单色或彩色静态图片。

中文名	慢扫描电视	提出	1958年
外文名	Slow Scan Television，SSTV	通过	普通的3KHz语音通道传输画面
		回扫	Fly back https://blog.csdn.net/weixin_45696568

这里用到工具 `qsstv`

在linux下安装：`apt install qsstv`

安装完打开 `qsstv`，操作一波即可，关于qsstv的具体使用参考这篇文章



[UTCTF2020]spectrogram

直接看频谱图...



我爱Linux

下载的图片打不开，发现是png的文件头，jpg的文件尾

```

F 0123456789ABCDEF
01 %PNG..JFIF.....
06 .....ÿÛ.G
0E .....
    struct PNG_Sk
    uint16 htPnc

```

定位到jpg尾部之后，发现后面的数据比较奇怪，看了大佬的wp得知这种数据格式是 [Python的序列化文件](#)

```

2990h: BF E0 6D 0F 4D D4 7C 23 A5 DE 5E D9 C1 35 CC F0 ;am.MO|#¥P^0A510
29A0h: 89 24 91 D0 65 98 92 4D 14 50 07 FF D9 80 03 5D %$ `ðe~`M.P.ÿÛ€.]
29B0h: 71 00 28 5D 71 01 28 4B 03 58 01 00 00 00 6D 71 q. (]q. (K.X...mq
29C0h: 02 86 71 03 4B 04 58 01 00 00 00 22 71 04 86 71 .tq.K.X..."q.tq
29D0h: 05 4B 05 68 04 86 71 06 4B 08 68 04 86 71 07 4B .K.h.tq.K.h.tq.K
29E0h: 09 68 04 86 71 08 4B 0A 58 01 00 00 00 23 71 09 .h.tq.K.X...#q.
29F0h: 86 71 0A 4B 1F 68 02 86 71 0B 4B 20 68 04 86 71 tq.K.h.tq.K h.tq
2A00h: 0C 4B 21 68 04 86 71 0D 4B 2C 68 02 86 71 0E 4B .K!h.tq.K,h.tq.K
2A10h: 2D 68 02 86 71 0F 4B 2E 68 02 86 71 10 4B 2F 68 -h.tq.K.h.tq.K/h
2A20h: 02 86 71 11 4B 32 68 02 86 71 12 4B 33 68 02 86 .tq.K2h.tq.K3h.t
2A30h: 71 13 4B 34 68 02 86 71 14 4B 35 68 02 86 71 15 q.K4h.tq.K5h.tq.
2A40h: 4B 36 68 02 86 71 16 4B 37 68 02 86 71 17 4B 3A K6h.tq.K7h.tq.K:
2A50h: 68 02 86 71 18 4B 3B 68 02 86 71 19 4B 3C 68 02 h.tq.K;h.tq.K<h.
2A60h: 86 71 1A 4B 3D 68 02 86 71 1B 4B 42 68 02 86 71 tq.K=h.tq.KBh.tq
2A70h: 1C 4B 43 68 04 86 71 1D 4B 44 68 04 86 71 1E 4B .KCh.tq.KDh.tq.K
2A80h: 4B 68 09 86 71 1F 65 5D 71 20 28 4B 01 68 02 86 Kh.tq.e]q (K.h.t
2A90h: 71 21 4B 02 68 02 86 71 22 4B 03 68 09 86 71 23 q!K.h.tq"K.h.tq#
2AA0h: 4B 04 68 02 86 71 24 4B 05 68 02 86 71 25 4B 0A K.h.tq$K.h.tq%K.
2AB0h: 68 09 86 71 26 4B 10 68 02 86 71 27 4B 11 68 02 h.tq&K.h.tq'K.h.
2AC0h: 86 71 28 4B 12 68 02 86 71 29 4B 17 68 02 86 71 tq(K.h.tq)K.h.tq
2AD0h: 2A 4B 18 68 02 86 71 2B 4B 19 68 02 86 71 2C 4B *K.h.tq+K.h.tq,K
2AE0h: 1A 68 02 86 71 2D 4B 1F 68 09 86 71 2E 4B 25 68 .h.tq-K.h.tq.K%h
2AF0h: 02 86 71 2F 4B 26 68 02 86 71 30 4B 27 68 02 86 .tq/K&h.tq0K'h.t
2B00h: 71 31 4B 2B 68 04 86 71 32 4B 2F 68 04 86 71 33 q1K+h.tq2K/h.tq3
2B10h: 4B 30 68 09 86 71 34 4B 36 68 09 86 71 35 4B 37 K0h.tq4K6h.tq5K7
2B20h: 68 04 86 71 36 4B 39 68 04 86 71 37 4B 3D 68 04 h.tq6K9h.tq7K=h.
2B30h: 86 71 38 4B 3E 68 09 86 71 39 4B 40 68 02 86 71 tq8K>h.tq9K@h.tq
2B40h: 3A 4B 41 68 02 86 71 3B 4B 42 68 09 86 71 3C 4B :KAh.tq;KBh.tq<K
2B50h: 43 68 02 86 71 3D 4B 44 68 02 86 71 3E 4B 49 68 Ch.tq=KKh.45096568
2B60h: 02 86 71 3F 4B 40 68 02 86 71 40 4B 43 68 02 86 tq0KTh.tq0KTh.t

```

把这些数据提取出来，保存在1.txt中

```

1.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
€]q (]q( (K)X) mq 嗶(K)X) "q嗶嗶(K)h嗶嗶(K)h嗶嗶(K) h嗶嗶(K
X) #q 嗶
Kh 嗶(K h嗶嗶K!h嗶嗶
K,h 嗶(K-h 嗶(K.h 嗶(K/h 嗶(K2h 嗶(K3h 嗶(K4h 嗶(K5h 嗶(K6h 嗶(K7h 嗶(K:h
嗶(K;h 嗶(K<h 嗶(K=h 嗶(KBh 嗶(KCh嗶(KDh嗶(KKh 嗶(e]q (K)h 嗶(K h 嗶(K)h
嗶(K)h 嗶($K)h 嗶(%K
h 嗶&K)h 嗶('K)h 嗶(K)h 嗶(K)h 嗶(*K)h 嗶(+K)h 嗶,K)h 嗶-K)h 嗶(K%)h
嗶/K&h 嗶0K'h 嗶1K+h嗶2K/h嗶3K0h 嗶4K6h 嗶5K7h嗶6K9h嗶7K=h嗶8K>h
嗶9K@h 嗶:KAh 嗶;KBh 嗶<KCh 嗶=KDh 嗶>KHh 嗶?K)h 嗶@K)h 嗶AKKh
嗶Be]qC(K)h 嗶DK
h 嗶EK)h嗶FK)h 嗶GK)h 嗶HK)h嗶IK)h嗶JK)h 嗶KKh 嗶LKh 嗶
MKh嗶NK$h嗶OK(h 嗶PK/h 嗶QK0h嗶RK5h 嗶SK6h嗶TK;h 嗶UK<h 嗶
VK=h 嗶WK>h嗶XKBh 嗶YKGh 嗶ZKHh嗶[K)h嗶\KKh 嗶]e]q^(K)h
嗶_K
h 嗶`K)h 嗶aK)h嗶bK)h嗶cK)h嗶dK)h 嗶eK)h 嗶fK)h 嗶gKh 嗶hK

```

```
$h 啱iK%h啱啱jK&h啱啱kK'h啱啱lK(h 啱mK-h 啱nK.h啱啱oK4h 啱pK5h啱啱qK=h啱啱
rK>h 啱sKBh 啱tKGh 啱uKKh 啱ve]qw(K啱h 啱xK
h啱啱yK啱h 啱zK▲h 啱{K啱h啱啱|K啱h 啱}K啱h 啱~K啱h啱啱 K啱h 啱€K啱h啱啱並啱h 啱
佻啱h 啱侵啱h啱啱夙啱h 啱囚h 啱佬$h啱啱嘖%h 啱鬩&h 啱塊'h啱啱亥(h 啱耍+h 啱享,h
啱啱站-h 啱巛.h 啱席/h 啱低0h 啱慘3h 啱扱4h啱啱摠9h啱啱摩:h 啱肺;h 啱駿<h 啱悖=h
啱楔>h啱啱橈Bh 啱啱欲Gh啱啱沘Hh 啱湖lh 啱瀟Jh 啱潜Kh 啱烟]q?K啱h 啱
啱h 啱 h啱啱 !h啱啱 ]q?K啱h啱啱 啱h啱啱 ]q?K▲h 啱啱犹啱h 啱啱猿啱h啱啱琯啱h啱啱
第 1 行, 第 1 列 100% Unix (LF) log.csdn.net ANSi (in_45696568)
```

找到一篇Python反序列化的文章：<https://jingyan.baidu.com/article/0bc808fcb7f91e5bd585b97d.html>，跟着步骤写脚本

```
import pickle
f=open("1.txt","rb")
result=pickle.load(f)
f.close()
f1=open("out.txt","w")
f1.write(str(result))
f1.close()
```

得到一些坐标：

```
out.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
[[ (3, 'm'), (4, ''), (5, ''), (8, ''), (9, ''), (10, '#'), (31, 'm'), (32, ''), (33, ''), (44, 'm'), (45,
'm'), (46, 'm'), (47, 'm'), (50, 'm'), (51, 'm'), (52, 'm'), (53, 'm'), (54, 'm'), (55, 'm'), (58,
'm'), (59, 'm'), (60, 'm'), (61, 'm'), (66, 'm'), (67, ''), (68, ''), (75, '#')], [(1, 'm'), (2, 'm'),
(3, '#'), (4, 'm'), (5, 'm'), (10, '#'), (16, 'm'), (17, 'm'), (18, 'm'), (23, 'm'), (24, 'm'), (25,
'm'), (26, 'm'), (31, '#'), (37, 'm'), (38, 'm'), (39, 'm'), (43, ''), (47, ''), (48, '#'), (54, '#'),
(55, ''), (57, ''), (61, ''), (62, '#'), (64, 'm'), (65, 'm'), (66, '#'), (67, 'm'), (68, 'm'), (72,
'm'), (73, 'm'), (74, 'm'), (75, '#')], [(3, '#'), (10, '#'), (15, ''), (19, '#'), (22, '#'), (23, ''),
(25, ''), (26, '#'), (29, 'm'), (30, 'm'), (31, ''), (36, ''), (40, '#'), (47, 'm'), (48, ''), (53,
'm'), (54, ''), (59, 'm'), (60, 'm'), (61, 'm'), (62, ''), (66, '#'), (71, '#'), (72, ''), (74, ''),
(75, '#')], [(3, '#'), (10, '#'), (15, 'm'), (16, ''), (17, ''), (18, ''), (19, '#'), (22, '#'), (26, '#'),
(31, '#'), (36, 'm'), (37, ''), (38, ''), (39, ''), (40, '#'), (45, 'm'), (46, ''), (52, 'm'), (53,
'') (61, '') (62, '#') (66, '#') (71, '#') (75, '#')] [(3, '#') (10, '') (11, 'm') (12, 'm') (15
```

写一手脚本给它画出来：


```

f=open("out.txt","r")
txt=eval(f.read())
for i in range(len(txt)):
    s=""
    for j in range(1,77):
        n=0
        for k in txt[i]:
            if k[0] == j:
                n=1
                break
        if n:
            s+=k[1]
        else:
            s+=" "
    print(s)
f.close()

```



https://blog.csdn.net/weixin_45696568

flag{a273fdedf3d746e97db9086ebbb195d6}