CTF中的无线电以及一些取证题目

原创

m0re 于 2021-01-11 01:09:45 发布 934 收藏 5

分类专栏： CTF 文章标签： CTF 无线电 取证 文件恢复 磁盘恢复

m0re

本文链接：https://blog.csdn.net/qq_45836474/article/details/112412259

前言

好久没发博客了，我的GitHub搭建的博客主题改好了，也就不改了， 以后慢慢转到我的个人博客上面了，CSDN很少发了。可能会不定时发一些。

之前打的比赛，看了眼还有附件，于是就找一下wp并复现了一遍。

这个比赛我记得当时好像就web签到成功了，然后就没有然后了。

无线电

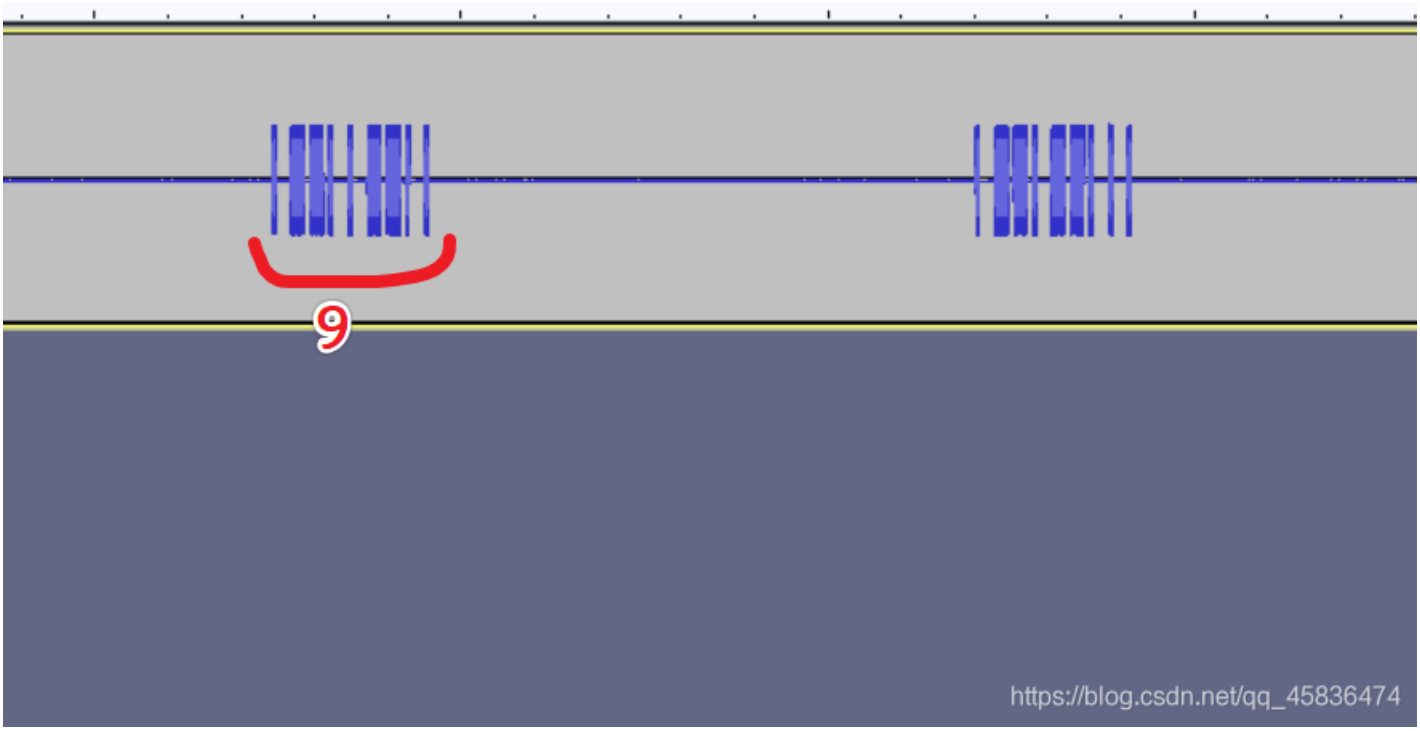
roarCTF

无线电的知识点——https://md.byr.moe/s/Bk6tsb2O8

Hi_433MHz

附件给的是扩展名为s8的文件，它是音频的原始数据。所以在给audacity中导入时，需要选择导入原始数据

打开在中间可以看到有数据模块，放大看是这样的。而且它每个部分有9个，不符合摩斯密码的规律(尝试转一下，转不出来)。



转换成01，细的为0，粗的为1。

```
01100110 01101100 01100001 01100111 01111011 00110010 00110101 01100011 00110010 00110001 01100010 00110000 0110
0100 00101101 00110110 01100001 00110001 00110001 00101101 00110100 00110011 00110001 00110010 00101101 00111001
 00110111 00110001 01100010 00101101 00110100 00110010 00111000 01100100 00110000 00110001 01100011 01100100 011
00011 00110101 00110011 00110100 01111101
```

将其转换为10进制数字，然后转成ASCII码就是flag
一个简单的Python脚本，进行转换

```python
#!/usr/bin/python
# -*- coding: utf-8 -*-
# author:  m0re
# blog:  https://m0re.top
test = ['01100110', '01101100', '01100001', '01100111', '01111011', '00110010', '00110101', '01100011', '00110010', '00110001', '01100010', '00110000', '01100100', '00101101', '00110110', '01100001', '00110001', '00110001', '00101101', '00110100', '00110011', '00110001', '00110010', '00101101', '00111001', '00110111', '00110001', '01100010', '00101101', '00110100', '00110010', '00111000', '01100100', '00110000', '00110001', '01100011', '01100100', '01100011', '00110101', '00110011', '00110100', '01111101']
for number in test:
    print(int(number, 2))
```

Encode Decode Encrypt Decrypt Binary About Others

Source Replace Replace Clear Copy

102, 108, 97, 103, 123, 50, 53, 99, 50, 49, 98, 48, 100, 45, 54, 97, 49, 49, 45, 52, 51, 49, 50, 45, , 98, 45, 52, 50, 56, 100, 48, 49, 99, 100, 99, 53, 51, 52, 125

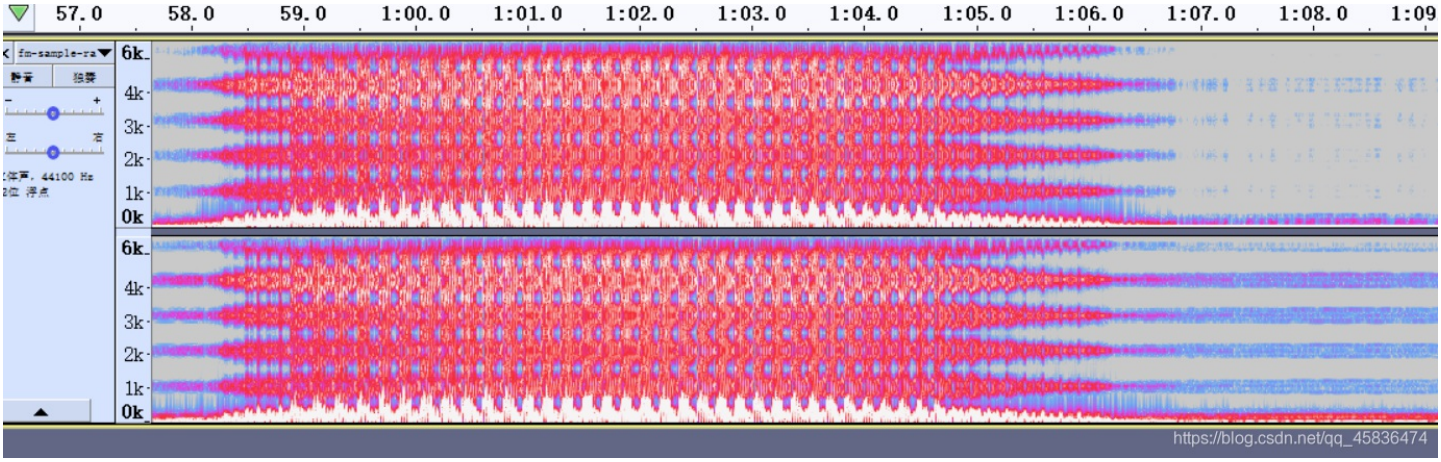
Result Replace Replace Clear Copy

flag{25c21b0d-6a11-4312-971b-428d01cdc534}

FM

这个题目是根据此篇博客进行复现。https://www.r00team.cc/writeup/2020-roarctf.html

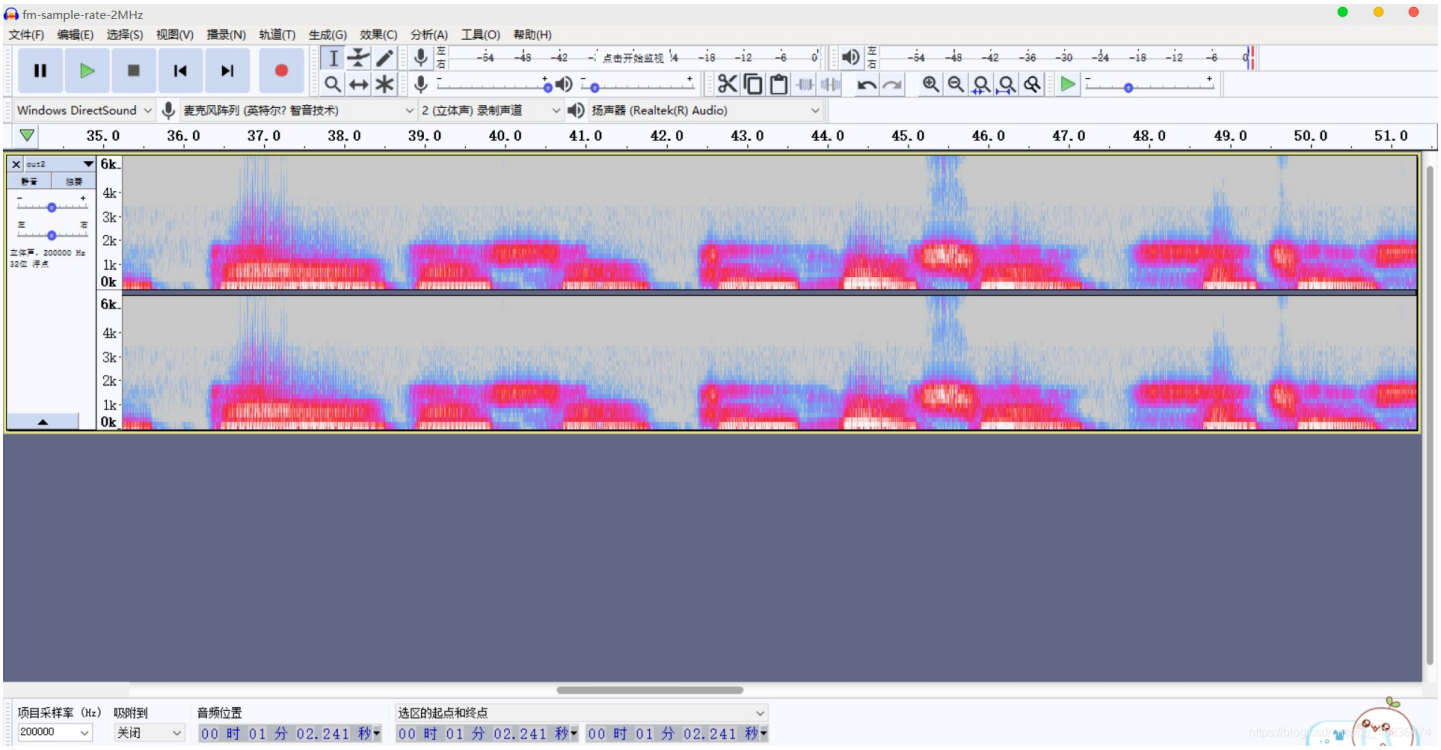
做频谱图查看，



使用脚本进行处理，这里贴一下大师傅的脚本

```
import numpy as np
y = np.fromfile('fm-sample-rate-2MHz.iq', dtype = np.complex64)
re1 = y.real[:-1]
re2 = y.real[1:]
im1 = y.imag[:-1]
im2 = y.imag[1:]
y_demod = im2 * re1 - im1 * re2
y_demod = y_demod / np.max(np.abs(y_demod))
y_demod.tofile('out2.bin')
```

再次导入，数据格式为单通道32位浮点数，采样率2MHz。作频谱图



进行降采样，将上述信号的采样率降低10倍（200kHz），转换为可以处理和收听的音频信号

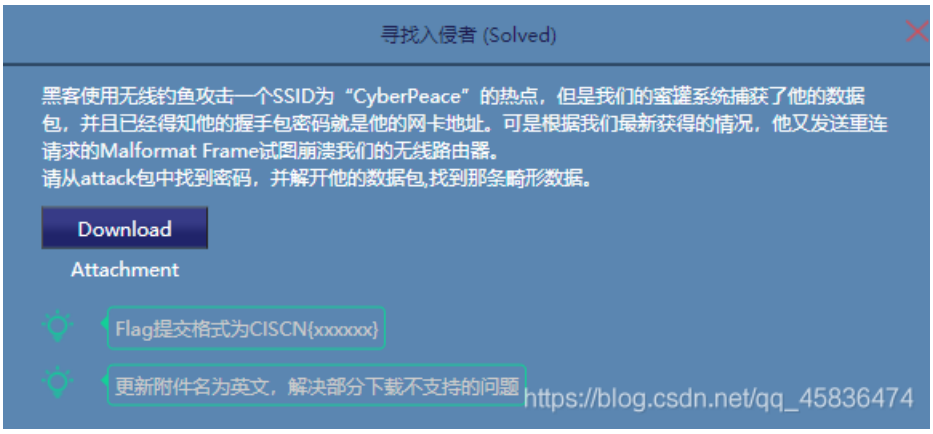
```
import numpy as np
import scipy.signal
y = np.fromfile('out2.bin', dtype = np.float32)
y_dec = scipy.signal.decimate(y, 10, ftype = 'fir')
y_dec.tofile('out2-dec.bin')
```

导入，按照原格式导入即可听到正常的音频。读的flag内容。

`flag{82c83416-dadc-4947-80df-b84852b8f35d}`

2018ciscn-misc-寻找入侵者

这个题目是在查找资料过程中，发现的一个不了解的知识。所以就也进行复现一下，了解一下这个知识。原文地址——https://www.anquanke.com/post/id/162546



打开wireshark，导入该数据包，可以看到很多的数据，
wireshark中无线帧的类型和过滤规则对照表

帧类型	过滤器语法

Management frame	wlan.fc.type == 0
Control frame	wlan.fc.type == 1
Data frame	wlan.fc.type == 2
Association request	wlan.fc.type_subtype == 0x00
Association response	wlan.fc.type_subtype == 0x01
Reassociation request	wlan.fc.type_subtype == 0x02
Reassociation response	wlan.fc.type_subtype == 0x03
Probe request	wlan.fc.type_subtype == 0x04
Probe response	wlan.fc.type_subtype == 0x05
Beacon	wlan.fc.type_subtype == 0x08
Disassociate	wlan.fc.type_subtype == 0x0A
Authentication	wlan.fc.type_subtype == 0x0B
Deauthentication	wlan.fc.type_subtype == 0x0C
Action frame	wlan.fc.type_subtype == 0x0D
Block ACK requests	wlan.fc.type_subtype == 0x18
Block ACK	wlan.fc.type_subtype == 0x19
Power save poll	wlan.fc.type_subtype == 0x1A
Request to send	wlan.fc.type_subtype == 0x1B
Clear to send	wlan.fc.type_subtype == 0x1C
ACK	wlan.fc.type_subtype == 0x1D
Contention free period end	wlan.fc.type_subtype == 0x1E
NULL data	wlan.fc.type_subtype == 0x24
QoS data	wlan.fc.type_subtype == 0x28
Null QoS data	wlan.fc.type_subtype == 0x2C

题目描述：“黑客使用无线钓鱼攻击一个SSID为‘CyberPeace’的热点”，我们可以先过滤SSID名为CyberPeace的Beacon的信标分组(一般通过Beacon就可以分析出AP的基本信息)，过滤语句`wlan.fc.type_subtype == 0x0008&&wlan.ssid==CyberPeace`

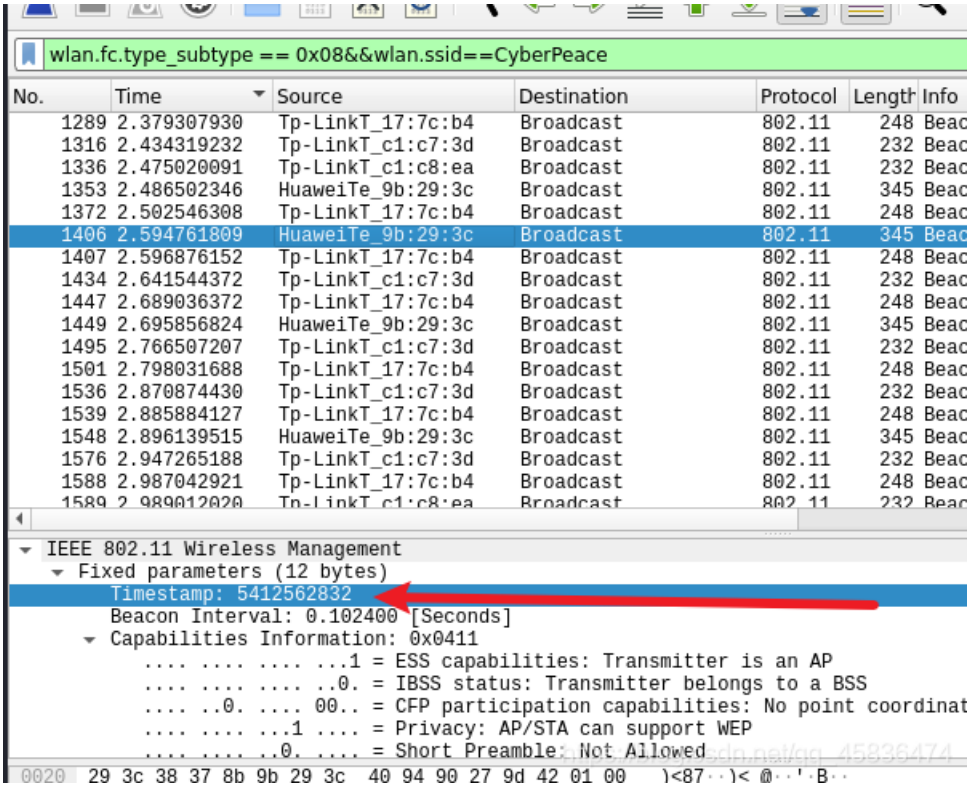
详情：Beacon的说明如下

序号	SubType	说明
1	1000	Beacon，STA接受AP信标帧，感知到AP，获取SSID及AP参数
2	0100	STA主动发送Probe探测请求
3	0101	AP应答STA Prob Response
4	1011	STA发送Authentication请求认证
5		AP应答Authentication请求，指示STA认证成功或失败
6	0000	STA发送Association请求
7	0001	AP应答Association Response

关于`802.11帧格式`—https://www.cnblogs.com/shwang/p/12162692.html

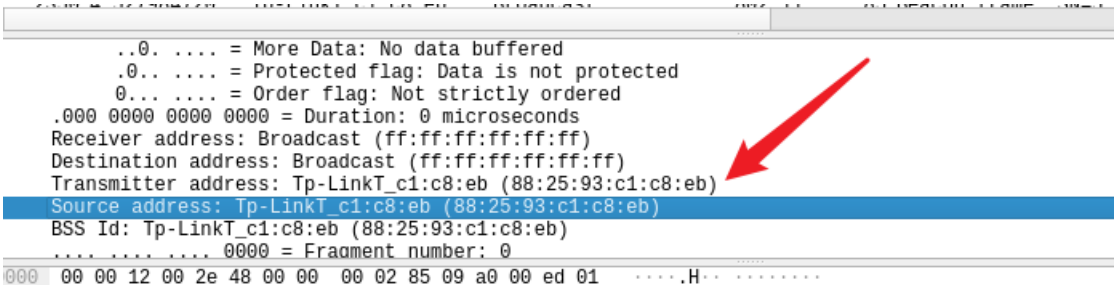
这篇文章写得比较详细，比较容易懂。

看时间戳，普通的数据包的时间戳是



而看到不同的时间戳是00

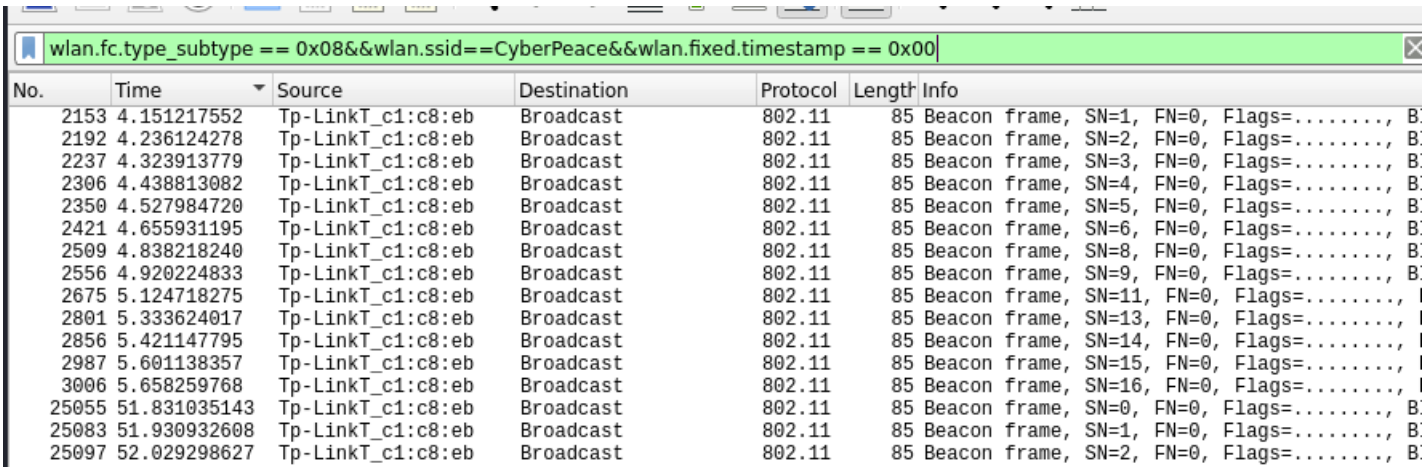
就是到2153条数据时，是00，看到的mac地址是(88:25:93:c1:c8:eb)



可以筛选一下全部时间戳为00的数据。

过滤语句是`wlan.fc.type_subtype == 0x0008&&wlan.ssid==CyberPeace&&wlan.fixed.timestamp == 0x00`

时间戳全部都是00，而且Mac地址是一样的。

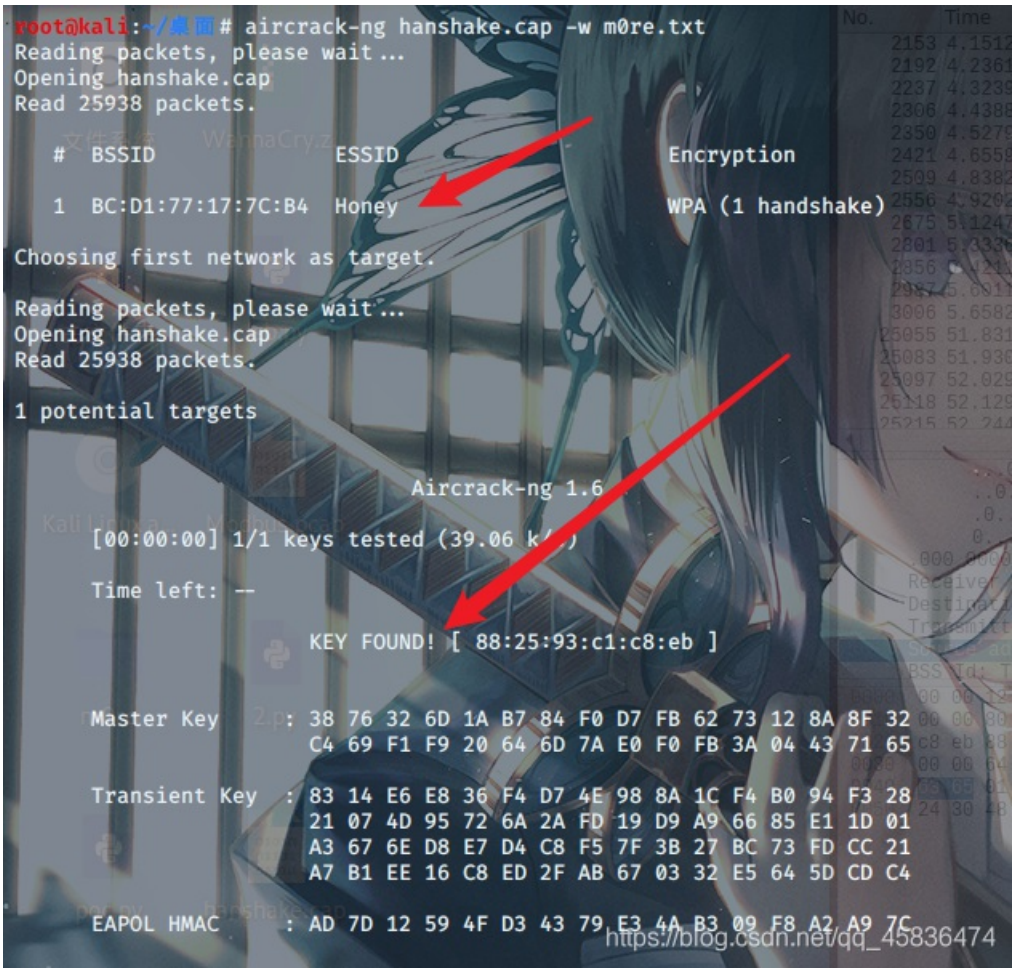


```
25118 52.129634919  Tp-LinkT_c1:c8:eb    Broadcast           802.11      85 Beacon frame, SN=3, FN=0, Flags=........, B
25215 52.244937605  Tp-LinkT_c1:c8:eb    Broadcast           802.11      85 Beacon frame, SN=4, FN=0, Flags=........, B

        ...0 .... = PWR MGT: STA will stay up
        ..0. .... = More Data: No data buffered
        .0.. .... = Protected flag: Data is not protected
        0... .... = Order flag: Not strictly ordered
    .000 0000 0000 0000 = Duration: 0 microseconds
    Receiver address: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Transmitter address: Tp-LinkT_c1:c8:eb (88:25:93:c1:c8:eb)
    Source address: Tp-LinkT_c1:c8:eb (88:25:93:c1:c8:eb)
    BSS Id: Tp-LinkT_c1:c8:eb (88:25:93:c1:c8:eb)

0000  00 00 12 00 2e 48 00 00  00 02 85 09 a0 00 ed 01   .....H.. ........
0010  00 00 80 00 00 00 ff ff  ff ff ff ff 88 25 93 c1   ........ .....%..
0020  c8 eb 88 25 93 c1 c8 eb  10 00 00 00 00 00 00 00   ...%.... ........
0030  00 00 64 00 01 04 00 0a  43 79 62 65 72 50 65 61   ..d..... CyberPea
0040  63 65 01 04 02 04 0b 16  03 01 06 32 08 0c 12 18   ce...... ...2....
0050  24 30 48 60 6c                                     $0H`l
```


题目：握手包密码就是他的网卡地址

将密码写入TXT文件中，然后通过利用aircrack-ng工具(kali自带)进行验证密码的正确性。

```
aircrack-ng hanshake.cap -w m0re.txt
```



然后就是对cap包进行解密。

第一种方法

利用wireshark官方在线网站——https://www.wireshark.org/tools/wpa-psk.html输入密码和SSID就可以得到一串PSK

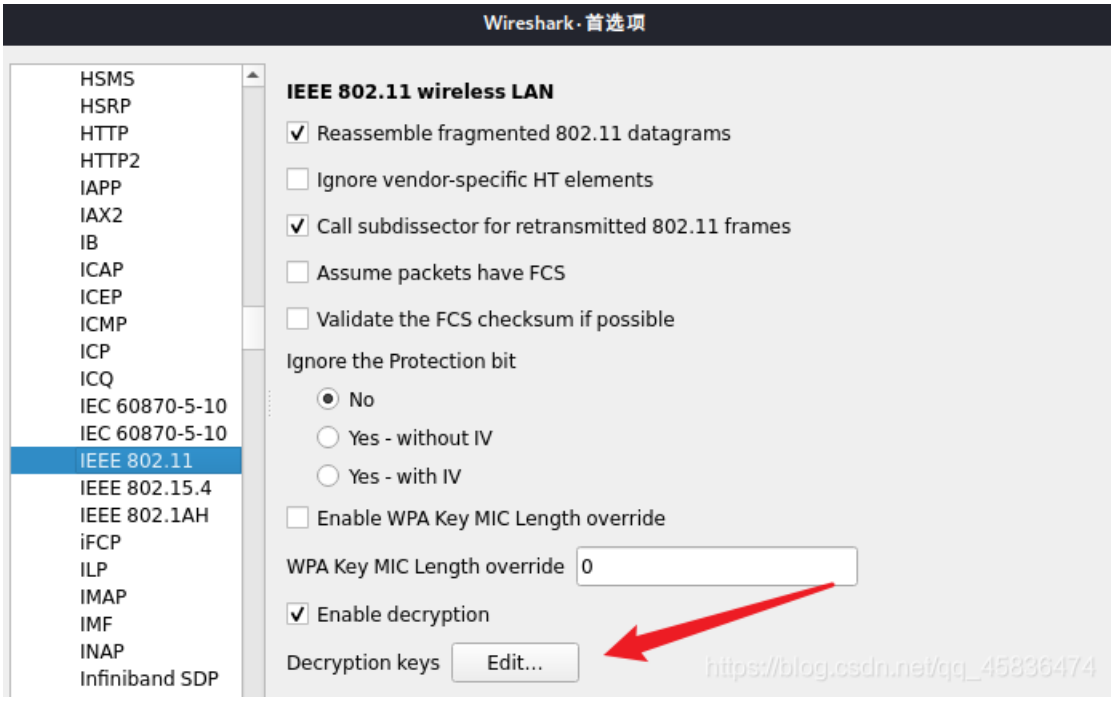
WPA PSK（原始密钥）生成器

Wireshark WPA预共享密钥生成器提供了一种简单的方法，可以将WPA密码和SSID转换为用于密钥派生的256位预共享（“原始”）密钥。

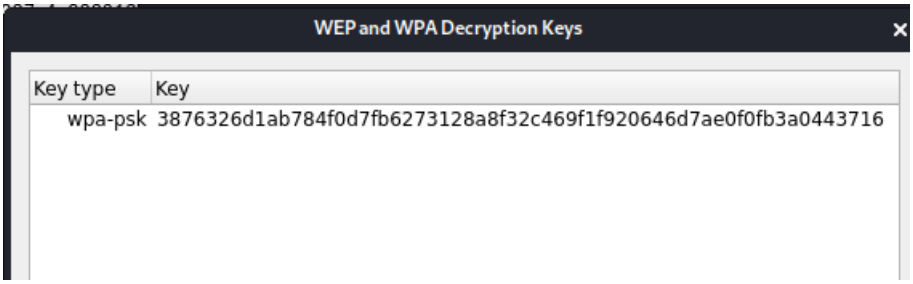
说明:
在下面输入或粘贴您的WPA密码和SSID。**稍等片刻**。PSK将由您的浏览器计算。Javascript以其惊人的加密速度而闻名。**没有**这些信息将通过网络发送。如果您不相信我们，请与Wireshark进行跟踪。

密码短语 88:25:93:c1:c8:eb
SSID Honey
PSK 3876326d1ab784f0d7fb6273128a8f32c469f1f920646d7ae0f0fb3a04437165
产生PSK

把`hanshake.cap`在wireshark中打开。
在编辑——首选项——Protocols——IEEE 802.11

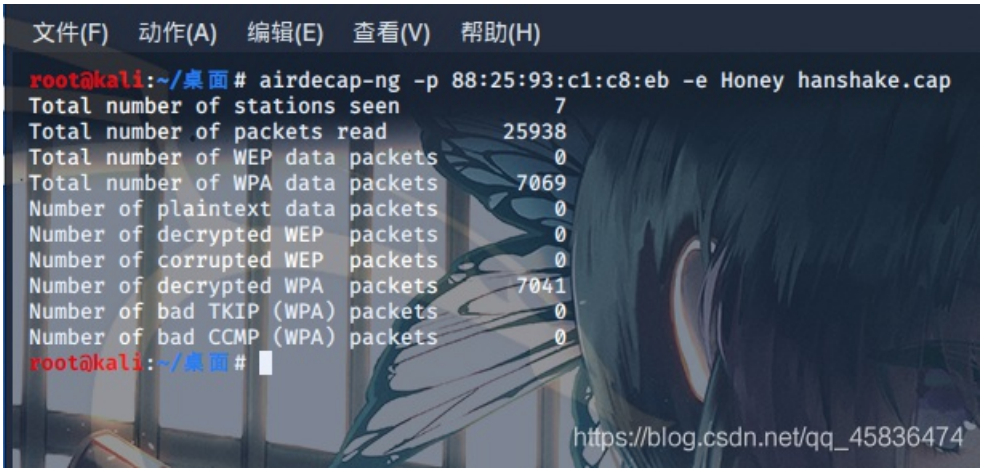


然后添加key-type为wpa-psk，添加key为刚才生成的PSK



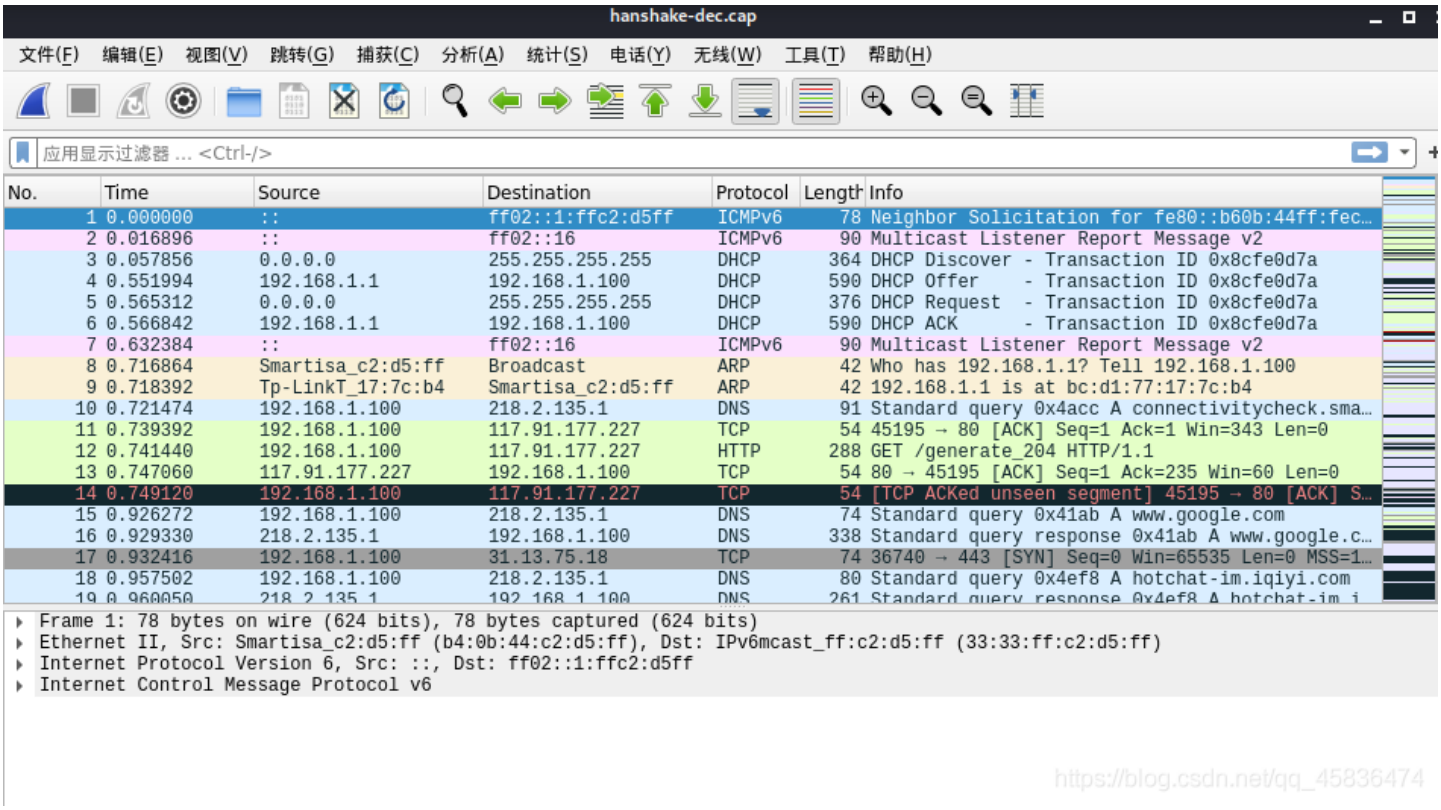
点击OK就解密成功了。
第二种方法
使用airdecap-ng工具解密

```
#命令
airdecap-ng -p 88:25:93:c1:c8:eb -e Honey hanshake.cap
```



然后会生成解密后的流量包

再次打开就会发现跟之前不一样了。

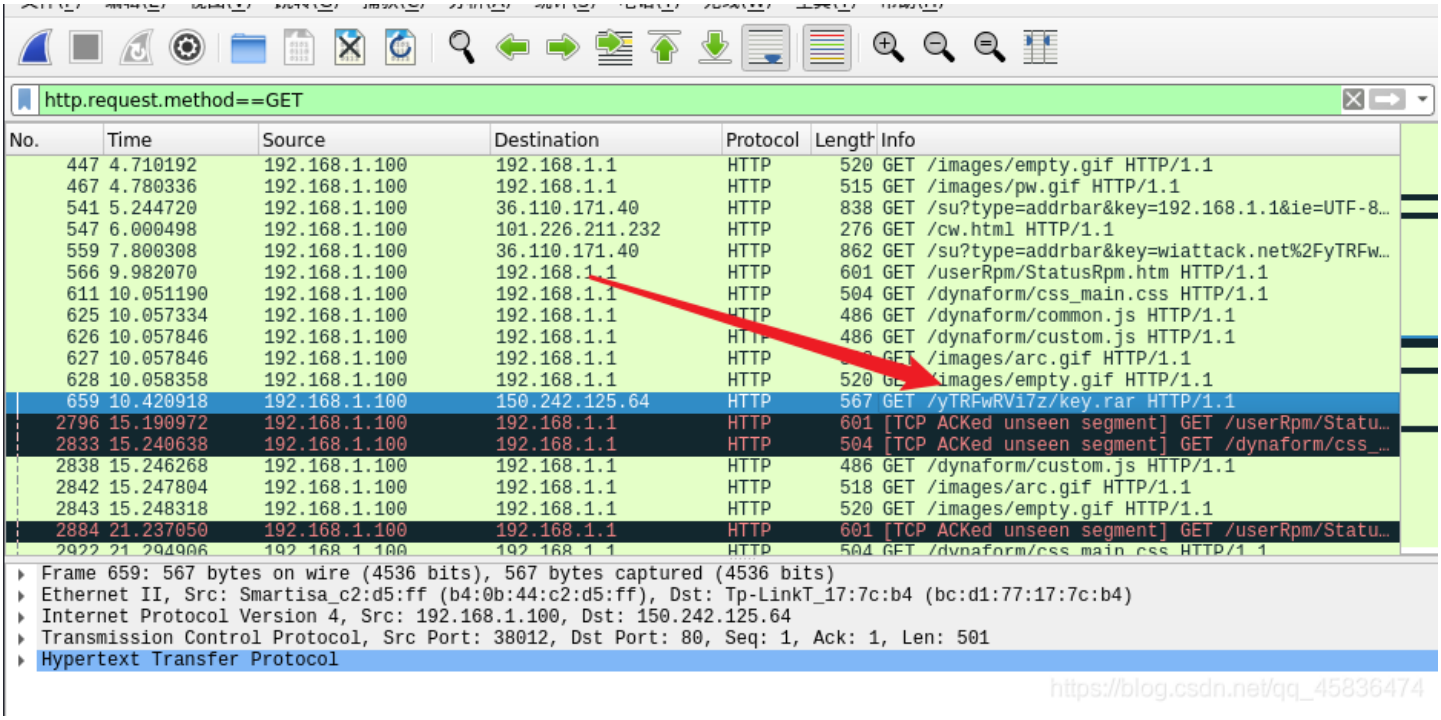


有很多协议的数据，所以首先查看了一下http的。尝试导出http数据流。

导出数据收集信息。

再执行一下过滤语句，将GET请求方式筛选出来。

```
http.request.method==GET
```



很奇怪的是，导出http对象的时候，没有发现这个key压缩包，但是在筛选get请求方式的时候，发现了这个压缩包。

正常流程是访问网址，下载压缩包。

题目描述：他又发送重连请求的Malformat Frame试图崩溃我们的无线路由器

故攻击者发送了大量请求。所以筛选一下发送者的Mac地址发送的请求。

过滤语句：`wlan.addr==88:25:93:c1:c8:eb`或者`wlan.sa == 88:25:93:c1:c8:eb`

然后emmmm，不是很懂最后这个

可以看看这篇博客

https://www.jianshu.com/p/04ef45f4b243

最后畸形数据`Z9DY20jZIyjzY5vs0UQZVUg86eMYlNMzH`

flag就是这个。

看了其他的wp，是strings命令解出来的

```
strings key.pcap
...
...
CyberPeace
0H`l-
ADSL Modem/Router
CyberPeace
0H`l-
!Z9DY20jZIyjzY5vs0UQZVUg86eMYlNMzH
```

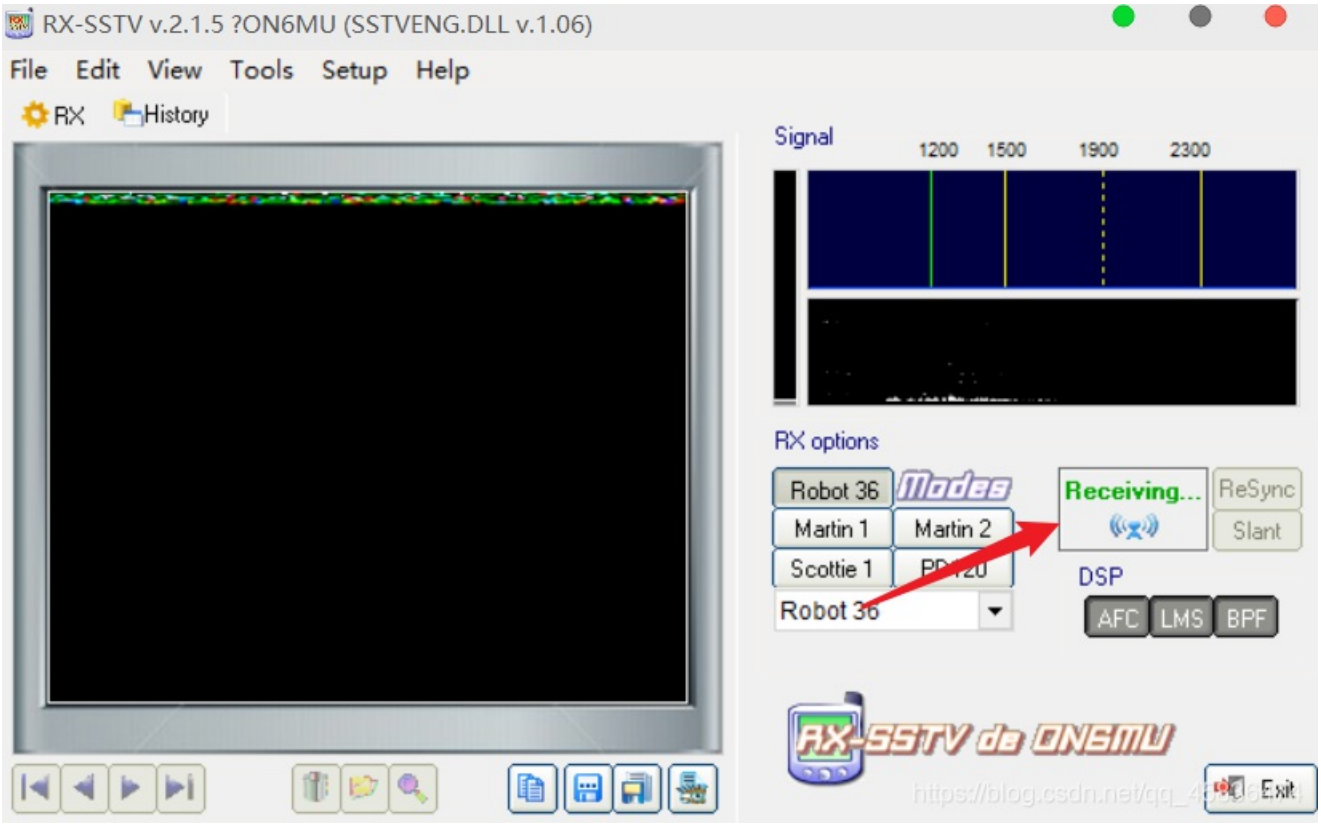
也能得到最后这个答案。

未知信号

题目来源——点击传送ctfshow

下载下来直接使用SSTV软件打开就可以了。软件安装包链接——https://m0re.lanzoux.com/ibgwKk96pyj

简单介绍一下使用方法吧。



这边是接收和停止。

使用的时候先点接收，然后直接放音频，由于音频太长，设备不足，我没有继续进行(主要是5分钟的无线电波听着贼难受)

参考——https://wp.ctf.show/d/101

取证

以下题目来源——CTFhub

文件恢复

emmmm 工具一把梭。
一条命令解决

```
extundelete DR --restore-all
```

恢复出来的文件有Flag.txt

磁盘恢复

预期结吧

挂载到空文件夹上

```
mount 86b265d37d1fc10b721a2accae04a60d /root/桌面/m0re/
```

查看当前挂载的文件系统

`df -lh`

当前挂载在/dev/loop0系统下。

重新挂载分区，以只读方式

`mount -r -o remount /dev/sdb2`

`-r`表示以只读方式，注意必须设置为只读方式

验证是否设置成功

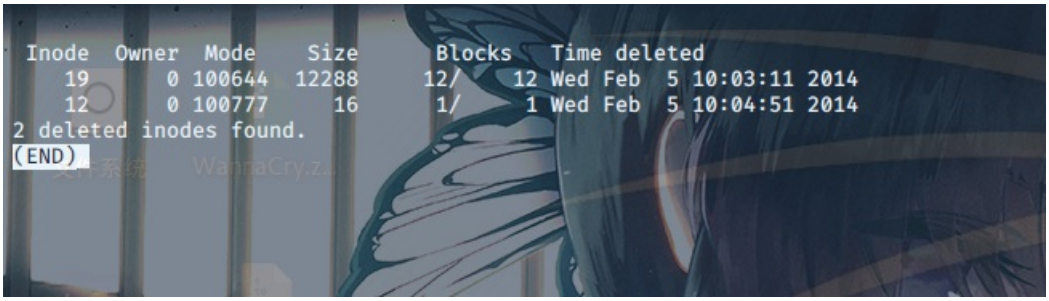
```
cd /guazai
mkdir aa
mkdir: cannot create directory `aa': Read-only file system
```

表示设置只读方式成功。

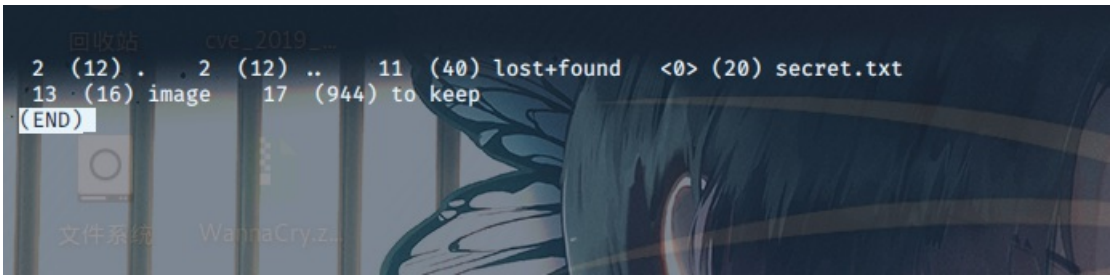
执行`debugfs`

```
debugfs /dev/loop0
```

然后在debugfs提示符后面输入命令。
使用`lsdel`命令，查看inode。



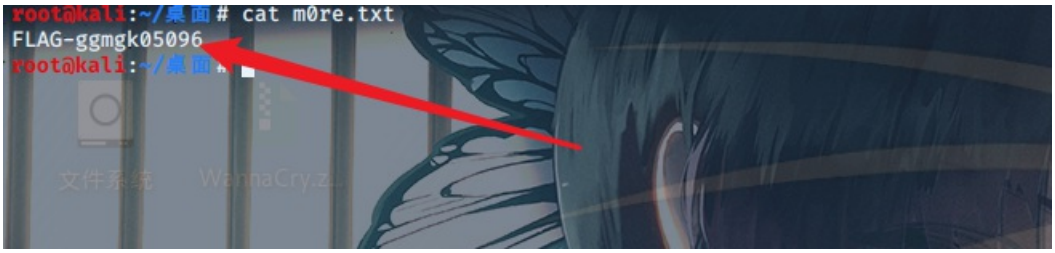
可以看到有两个inode，恢复文件时，不知道是哪个就挨个尝试一下。
使用`ls -d`查看被删除的文件。这里发现有个`secret.txt`文件，猜测应该就是flag所在了吧。



使用`dump`命令恢复文件内容。

```
#使用 dump <inode> /路径
dump <12> /root/桌面/m0re.txt
```

然后查看文件内容。发现flag。



非预期解(大多是这样的解法)

直接strings即可。

```
strings 86b265d37d1fc10b721a2accae04a60d | grep FLAG
```

又或者是直接记事本打开搜索flag字段。拿到flag。

小结

此次学习无线电的基础知识，和一些文件恢复的知识点。
嗯，总结就这样水了吧，没了。