第二届“祥云杯”网络安全大赛暨吉林省第四届大学生网络安全大赛 WriteUp 2021年祥云杯misc

原创

是Mumuzi 于 2021-08-23 12:01:48 发布 3039 收藏 6

分类专栏： ctf

本文链接：https://blog.csdn.net/qq_42880719/article/details/119859244

团队汇总WP:n03tack

6web+4misc+4crypto+2re+3pwn

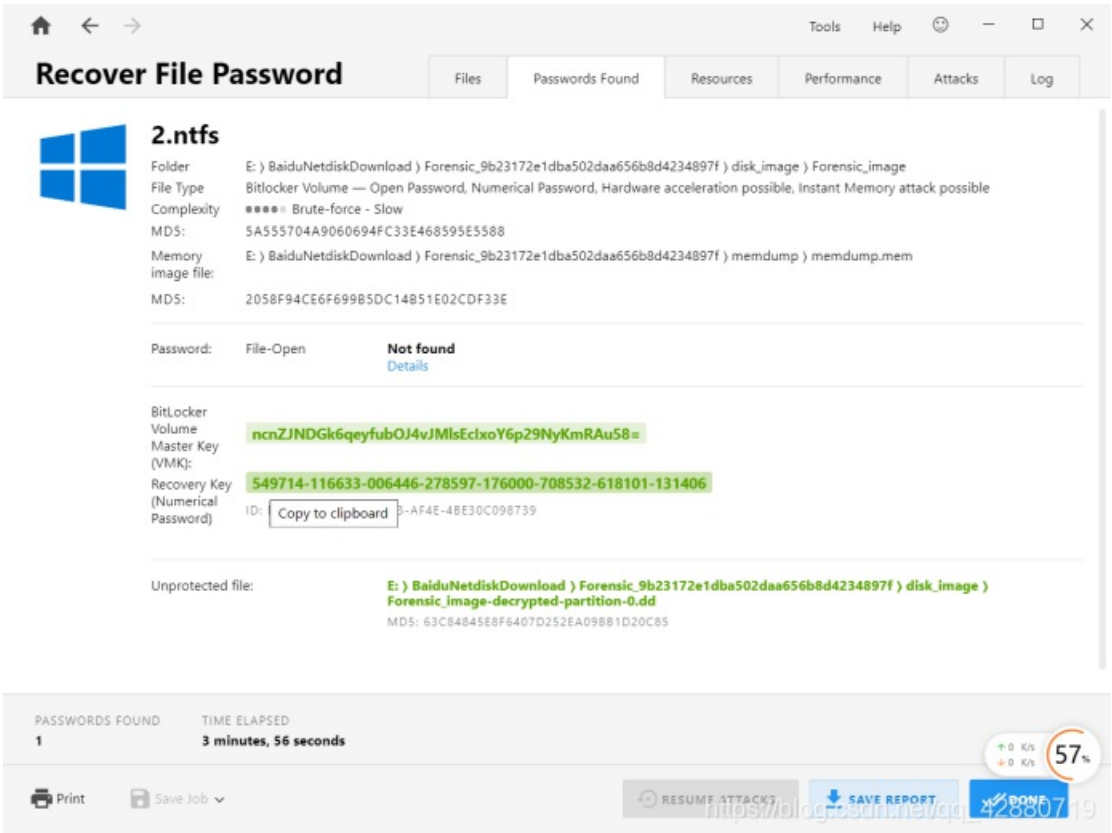
层层取证

层层套娃取证(确信)

给了一个内存和一个虚拟磁盘的取证，先看磁盘，取证大师打开提示存在bitlocker加密

用Passware Kit Forensic 2021 v1 (64-bit)能直接梭出来bitlocker的秘钥

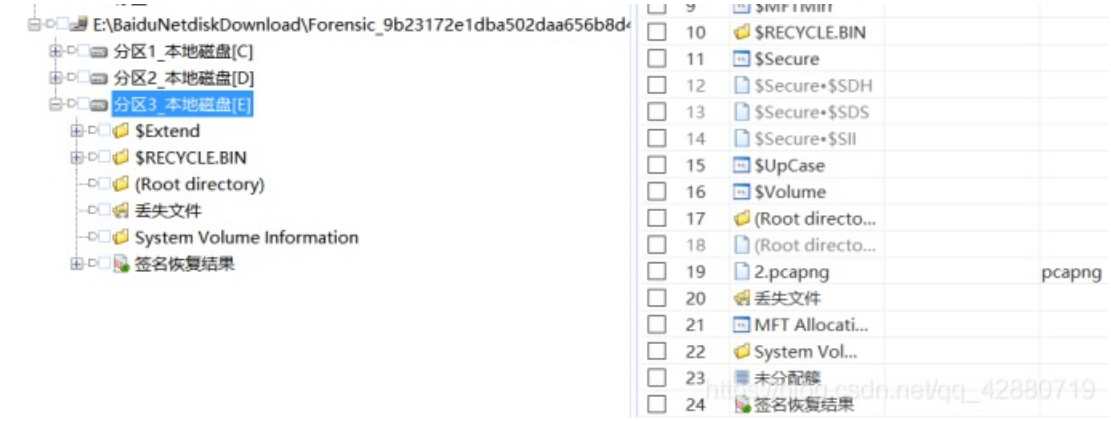
方法是把001解压出来，然后把2.ntfs放进去，再选择有内存镜像，导入这道题的内存镜像，然后等待……



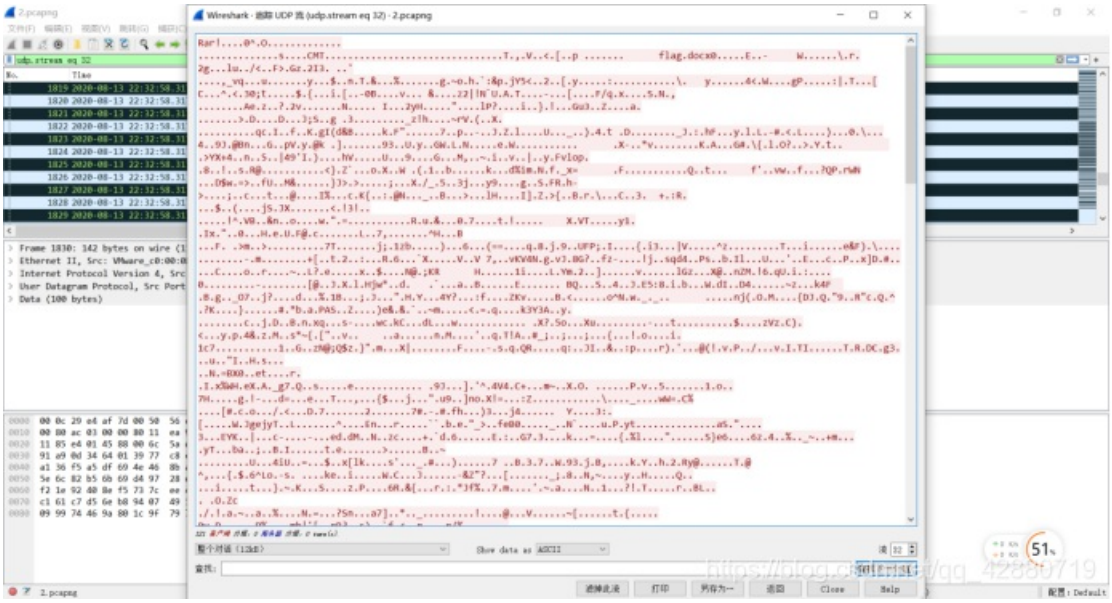
当然之后也发现内存中也可以找到，但是已经不重要

得到549714-116633-006446-278597-176000-708532-618101-131406

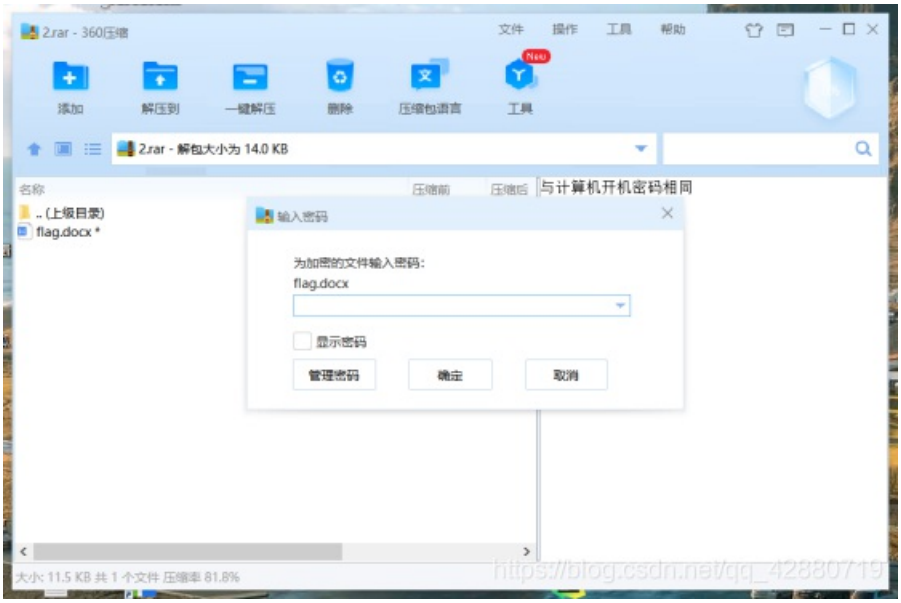
解开发现E盘存在一个流量包



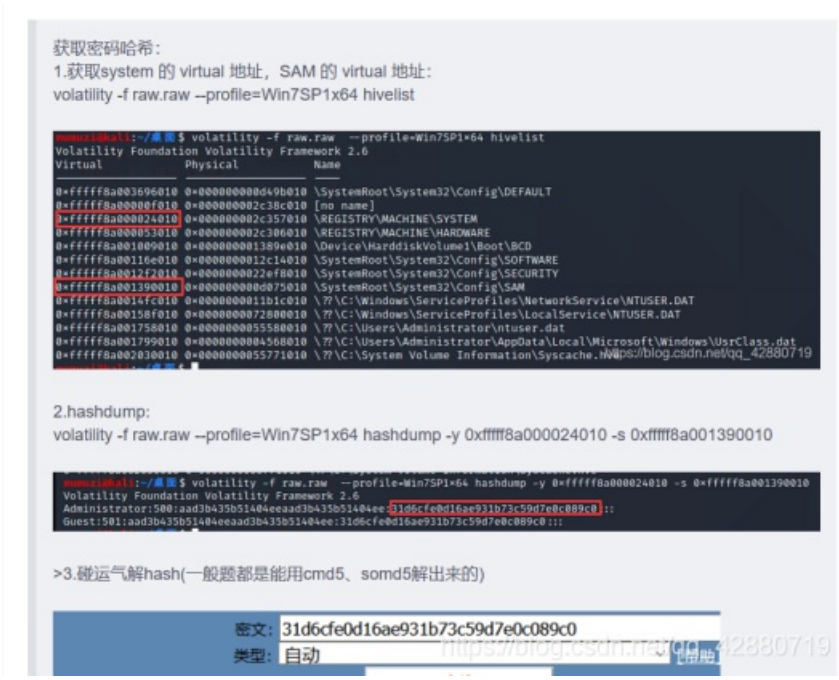
在udp里面找到一个rar，里面包含了一个flag.docx



然后导出，说压缩包密码是开机密码

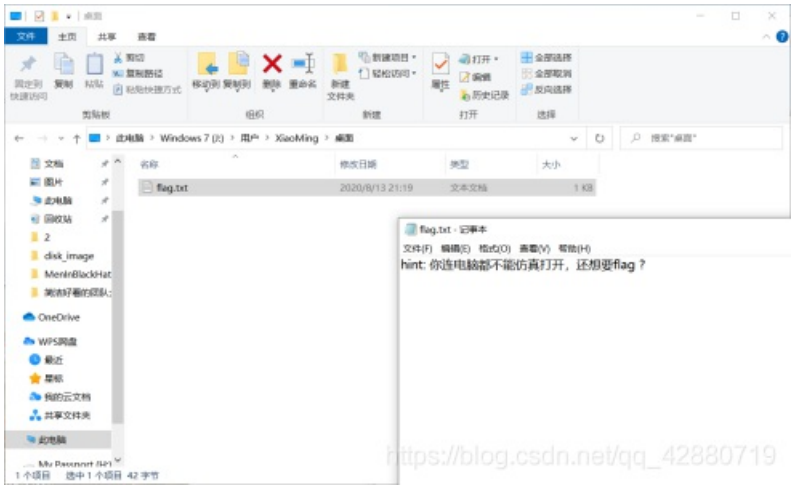


然后去看俺的另一篇博客



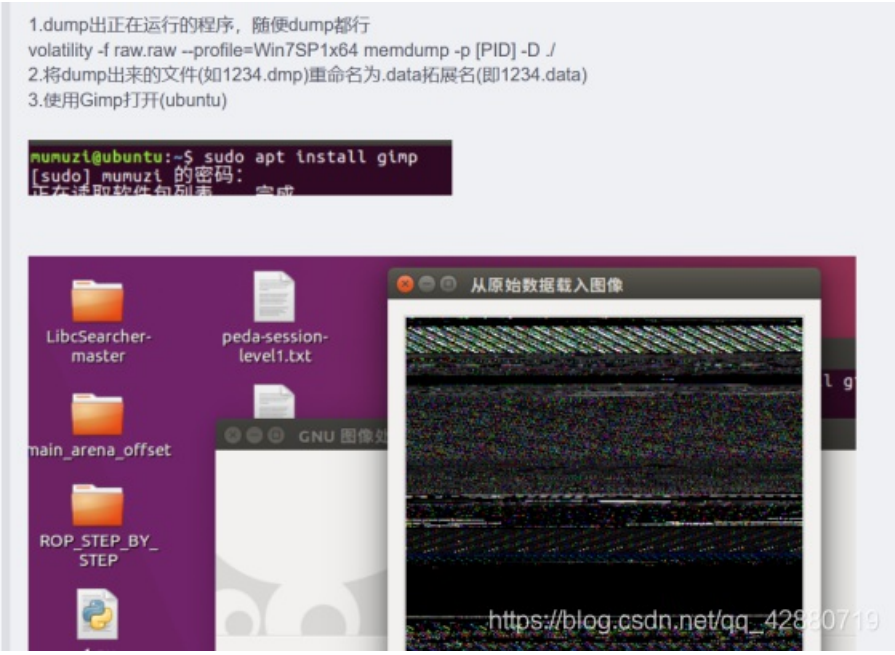
这里提取的是92EFA7F9F2740956D51157F46521F941

密码xiaoming_handsome，cmd5解一下就行了(不会有人没开会员吧)
解压之后发现docx还有密码

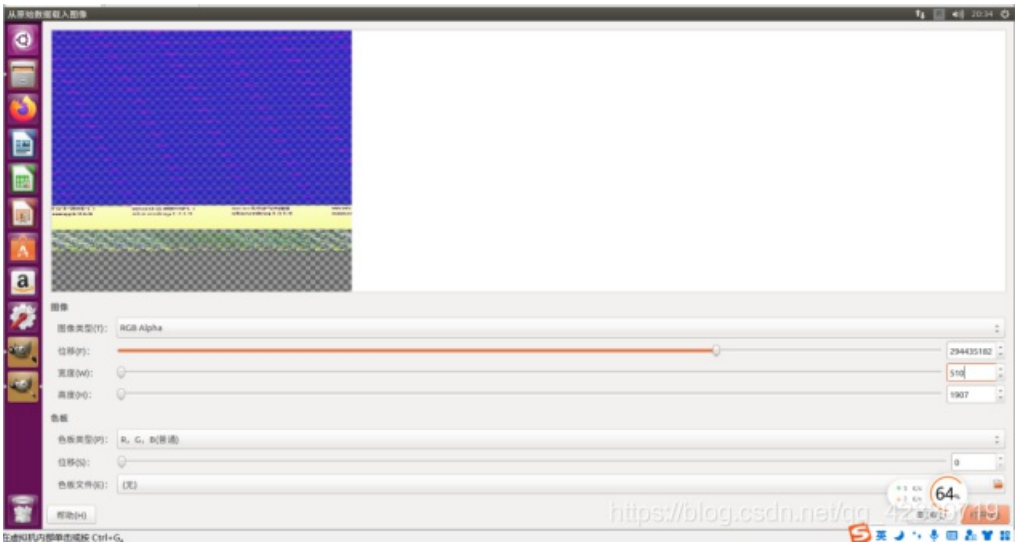


实在找不到了的时候，挂载一下，发现hint

然后去仿真。。。。。。嗯仿真不出来
照着https://www.freesion.com/article/51701409948/#DDE01_8我也不行
然后就去导出内存，还是看□的博客



找到然后密码



可以自己去操作放大看，偏移294435180，宽510
结合用户名，得到xiaoming1314，解压成功
得到flag

> flag{9ca871b668f2-b668-097c-cbm8-9op404c891e2}

当然对于赛后复盘，其实Passware Kit能直接把xiaoming的开机密码梭出来，如果看内存的图像知道了是在便笺上但是看不清，可以直接取证的时候将C:\Users\帐户文件夹\AppData\Roaming\Microsoft\Sticky Notes\StickyNotes.snt打开，然后再找到里面的密码。顺带还发现，其实不用解bitlocker也能找到含有rar的流量包，在XiaoMing\AppData\Local\Temp下。总之这种内存取证解法挺多的，自己喜欢用啥方法就用啥方法。

鸣雏恋

改zip解压，在_rels下发现key和love，key是妥妥的零宽

解压之后是两张图，其实这里可以不用解压直接读CRC，可惜写的脚本有点问题，为了不浪费时间还是解压了(其实是解压途中写的zip读CRC脚本写的有问题就还是用读图片了)
然后写个脚本，2进制的ASCII

```
from PIL import Image
from tqdm import tqdm
path = 'C:\\Users\\mumuzi\\AppData\\Local\\Temp\\鸣雏恋_2dad763070b79f50c4635a906359909a\\鸣雏恋\\_rels\\love\\out\\'
flag = ''

for i in tqdm(range(129488)):
    img = Image.open(path+str(i)+'.png')
    s = img.getpixel((10,10))
    if(str(s) == '1'):
        flag += '0'
    elif(str(s) == '3'):
        flag += '1'
    else:
        print('wrong!')
        exit()
s = ''
rflag = ''
for i in flag:
    s+=i
    if len(s)==8:
        rflag += chr(int(s,2))
        s=''
print(rflag)
```

得到base64的png图，图片最下面就是flag

> flag{57dd74fb21bb1aee50f19421bf836f23}

考古

最后的xor我是真的服气这一来怎么直接想得到太浪费时间了
先imageinfo，发现是XP，然后pslist列出表，发现cmd正在运行，cmdscan看一下cmd

```
mumuzi@kali:~/桌面$ volatility -f memory.mem --profile=WinXPSP2x86 cmdscan
Volatility Foundation Volatility Framework 2.6
**************************************************
CommandProcess: csrss.exe Pid: 620
CommandHistory: 0x3833938 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 16 LastAdded: 15 LastDisplayed: 15
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x404
Cmd #0 @ 0x3832110: It's useless to find so many things
Cmd #1 @ 0x3832ed0: ......................
Cmd #2 @ 0x52c778: what can i do about it
Cmd #3 @ 0x3833360: Heard that there is a one-click cleaning that is very useful
Cmd #4 @ 0x52b3c8: try it
Cmd #5 @ 0x52b7e8: "C:\Documents and Settings\Administrator\??\Oneclickcleanup.exe"
Cmd #6 @ 0x5224a0: what???
Cmd #7 @ 0x52d5c0: what happened??
Cmd #8 @ 0x52d410: who is 1cepeak?
Cmd #9 @ 0x3832de0: what's the meaning of hack?
Cmd #10 @ 0x3830e50: oh,no
Cmd #11 @ 0x52af40: holy shit
Cmd #12 @ 0x3830cf8: aaaaaa
Cmd #13 @ 0x522d28: Nonononononononononononono!!!!!!!!!!!!!!!
Cmd #14 @ 0x522d88: "C:\Documents and Settings\Administrator\??\Oneclickcleanup.exe"
Cmd #15 @ 0x5224b8: fuc
**************************************************
CommandProcess: csrss.exe Pid: 620
CommandHistory: 0x386b900 Application: DumpIt.exe Flags: Allocated
CommandCount: 0 LastAdded: -1 LastDisplayed: -1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x520
```

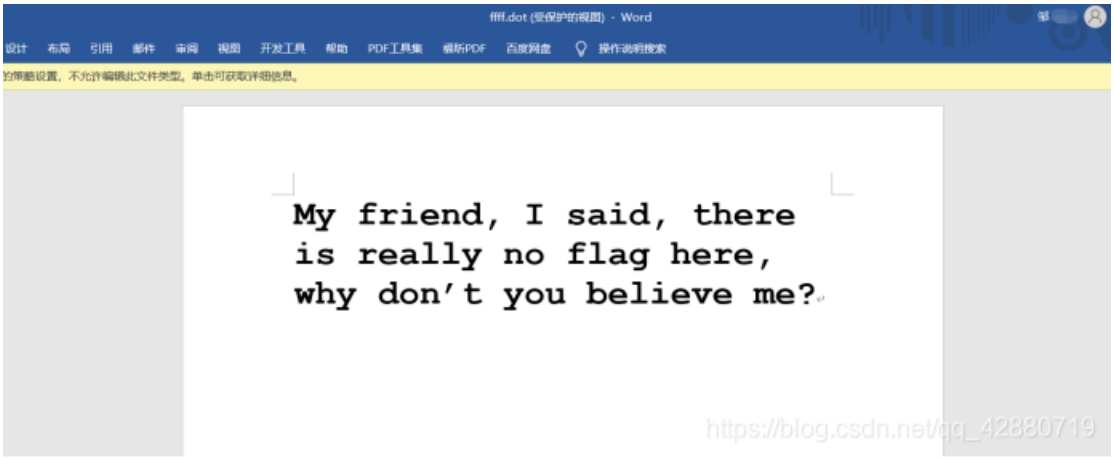

然后Filescan |grep “Oneclickcleanup.exe”，然后dumpfiles -Q 0x00000000017bcbc0 -D ./
一共是得到一个dat和一个iso，dat直接拖IDA32
发现是两个异或，第一个异或是存放地址，第二个异或是异或的数据

```
__main();
for ( i = 0; i <= 44; ++i )
  _data_start__[i] ^= key[i % 10];
for ( j = 0; j < (int)size; ++j )
  data[j] ^= key[j % 10];
for ( k = 0; k <= 9; ++k )
  puts("Hacked by 1cePack!!!!!!!");
v4 = fopen(_data_start__, "wb+");
fwrite(data, size, 1u, v4);
return 0;
```

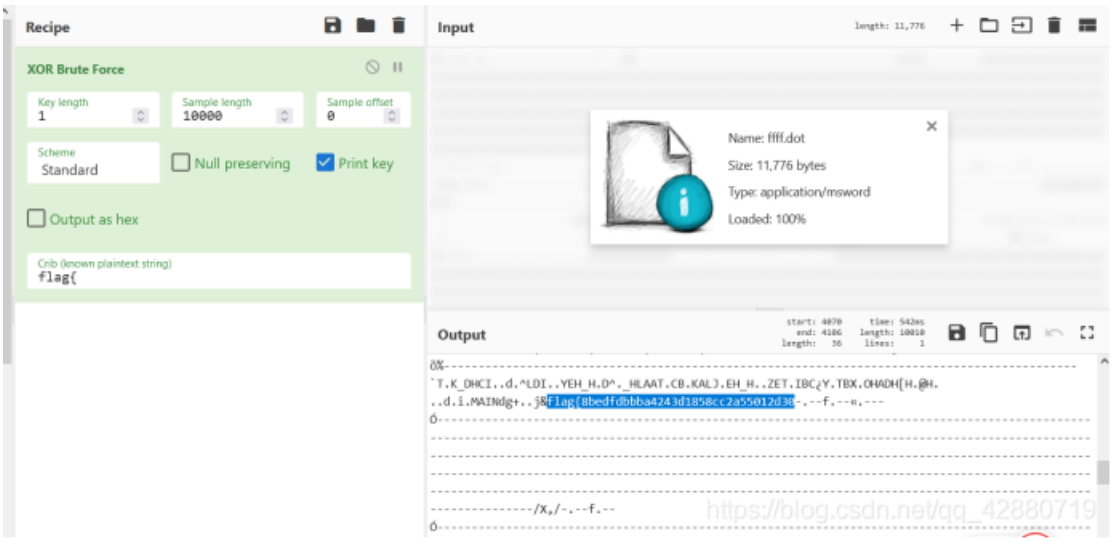

key是this_a_key，data就是数据，把data导出，写个脚本

```
s = 'this_a_key'
flag = ''
#flags = [0x37,0x52,0x35,0x37,0x30,0x02,0x2A,0x06,0x00,0x17,0x00,0x1B,0x49,0x12,0x31,0x05,0x7F,0x38,0x00,0x0D,0x
00,0x01,0x07,0x14,0x2C,0x3D,0x1E,0x07,0x09,0x59,0x21,0x1B,0x0C,0x01,0x2C,0x3D,0x0B,0x0E,0x08,0x09,0x18,0x09,0x1D
,0x16,0x2C]
flags = [0xA4,0xA7,0x78,0x93,0xFE,0xD0,0x45,……中间略,0x79,0x74,0x68,0x69,0x73,0x5F,0x61]
for i in range(len(flags)):
    flag += str(hex(flags[i] ^ ord(s[i%10])))[2:].zfill(2)
print(flag)
f = open('ffff.dot','wb')
f.write(flag.encode())
```

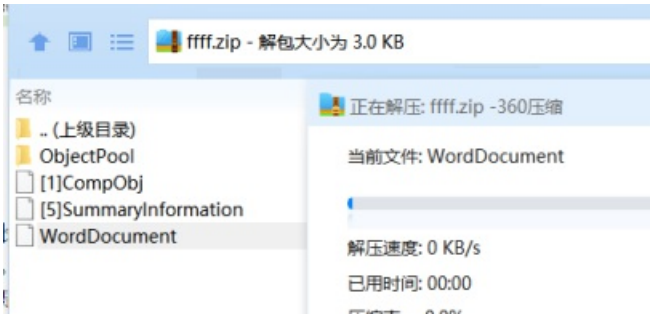
得到的ffff.dot用notepad++打开，将16进制转ascii
为什么要保存为.dot，因为我在复现我知道是dot我乐意
然后WPS打开发现提示版本过低，用office打开

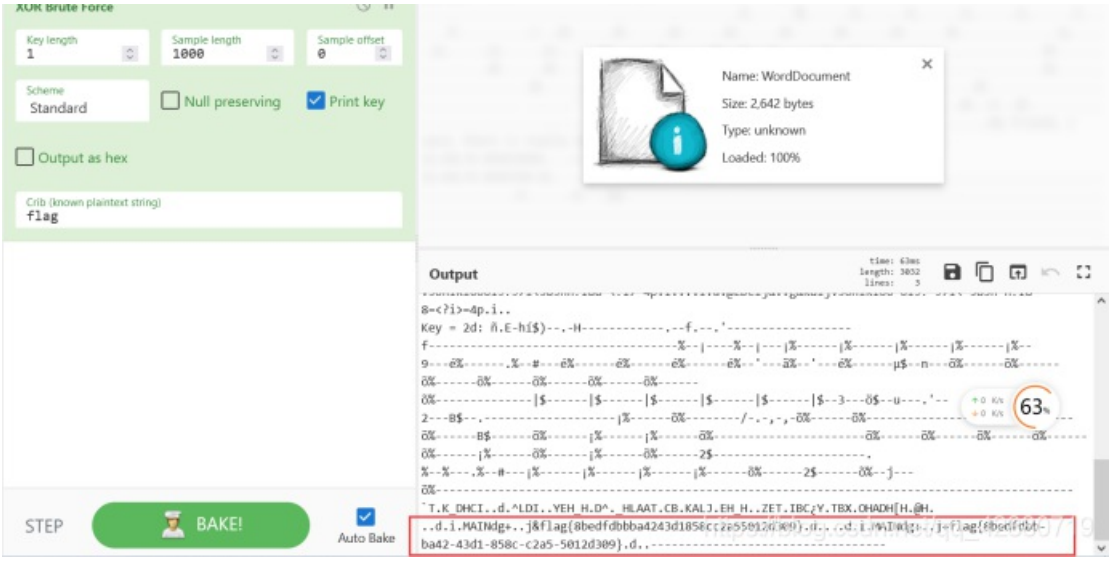


然后就瞎试，比如看16进制，从200h-1800h和1800h到最后是一样的，两者相比较什么的，都试过了，没出继续看内存文件。
最后。。。想着可打印ascii的词频和异或，结果异或出了



因为长度只有31位，爆破一下0-f，竟然不对，然后想着一共出现了两次，再去试试，还是不对。麻了
因为是dot文档，又想着zip解压直接看源文件，然后再来异或





???我可去你的吧居然还有一段
最终得到flag

> flag{8bedfdbb-ba42-43d1-858c-c2a5-5012d309}

麻了，太打脑壳了，别问，问就是4:07交的

Spoil_mu 2021/8/22 4:07:16
考古13解了

ChieftainsSecret

首先给了一个文档和一张图片，文档就是题目描述，图片的话是一个古时候的电话

搜了一下是怎么用的

https://haokan.baidu.com/v?pd=wisenatural&vid=15222023905414500076

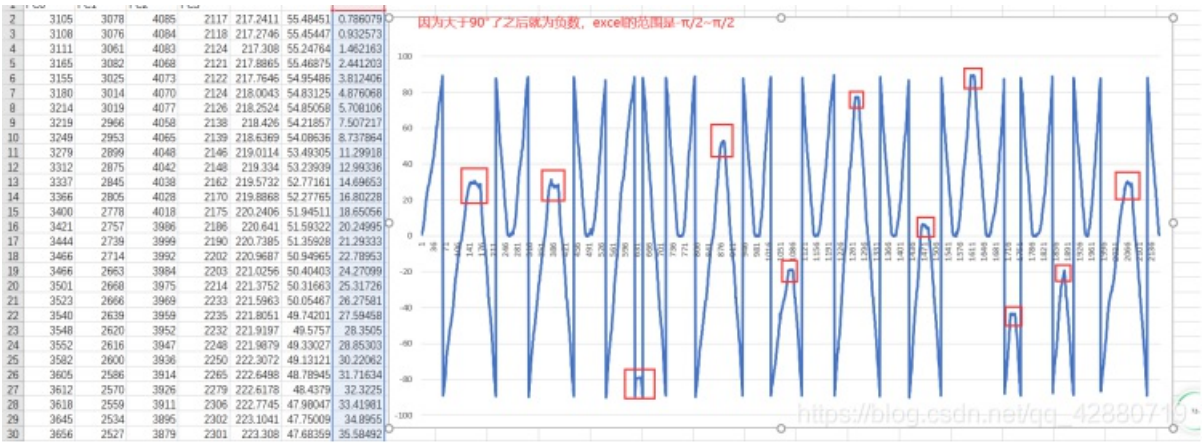
Jpg图片后面还有rar，foremost提取出来，发现是一堆TLE5501生成的数据，搜了一下他是一个角度的传感器，给了sinP,cosP,sinN,cosN

本着不会做就乱试+搜公式的原则，发现想要计算出角度，就得找到tan，于是萌生了sinP/cosP,sinN/cosN,(sinP-sinN)/(cosP-cosN)的想法，想知道角度，就再加上一个ATAN函数，除出来是弧度，就再乘以一个57.3°

顺便看了油管的视频，研究了一下四个值的变换

https://www.youtube.com/watch?v=y68ldqZs4PM

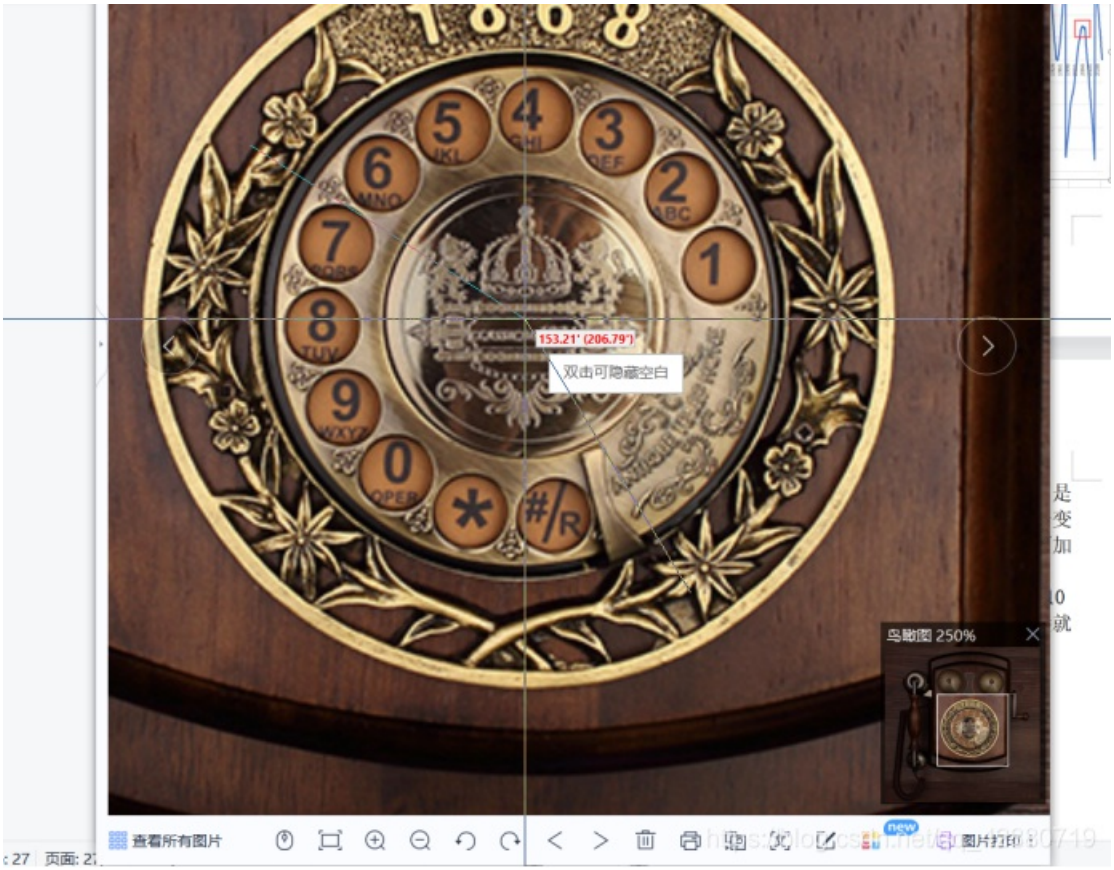
最后在看第三个想法的时候恍然大悟，搞出来也正好是11个峰



上图的意思就是ATAN((sinP-sinN)/(cosP-cosN))*57.3,然后因为ATAN的范围是-90°~90°，所以才会出现到90°的时候马上反转成负的，然后负的又接着变为正的，总之，第一个90°之后，后面的值要加180.第2个90°之后，后面加270，妈的简直找惨了。

得到的数字大概为210 210 280 230 160 260 190 90 140 160 210

然后去下载一个量角器(PicPickPortable)，发现2正好是90°，所以我直接就排除1了，然后一个个去量，下图举个例

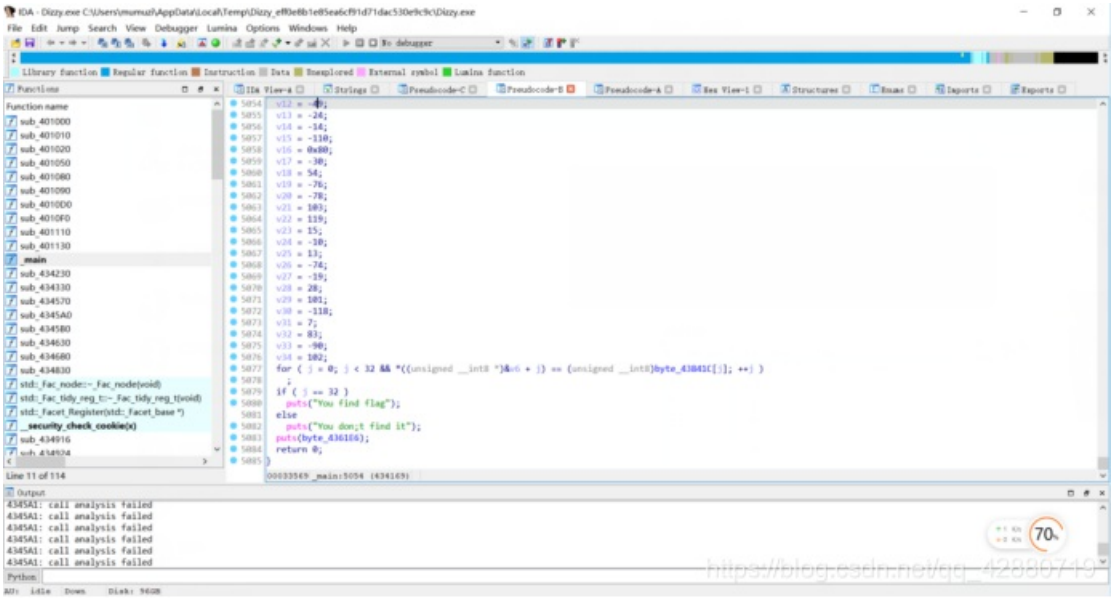


对应7，然后其他的也依次如图
得到77085962457

> 所以flag为flag{77085962457}

Rev_Dizzy

一串简单的数学计算，首先是IDA F5，挺慢的，多等等，如果提示要逆的太大了，就自己百度一下，改一下



给了运算、给了最后比较的值，逆一下就行了，将所有复制出来，只留下运算的部分，然后写个脚本

```
s = [0x27, 0x3C, 0xE3, 0xFC ,46 ,65 ,7 ,94 ,98 ,-49 ,-24 ,-14 ,-110 ,128 ,-30 ,54 ,-76 ,-78 ,103 ,119 ,15 ,-10 ,
13 ,-74 ,-19 ,28 ,101 ,-118 ,7 ,83 ,-90 ,102]

f = open('1.txt','r').readlines()
for i in range(len(f)):
    if(i != len(f)-1):
        f[i] = f[i][:-2]
    else:
        f[i] = f[i][:-1]
f1 = ['']*len(f)
for i in range(len(f)):
    f1[i] = f[len(f)-1-i]

for i in range(len(f1)):
    tmp = f1[i].replace('byte_43841C', 's')
    if('+=' in tmp):
        tmp = tmp.replace('+=','-=')
    elif('-=' in tmp):
        tmp = tmp.replace('-=','+=')
    else:
        pass
    exec(tmp)
for i in range(len(s)):
    print(chr(s[i]%256),end='')
```

> flag{Try_R3vers1ng_W1th_ScR!pt!}

shuffle_code的话提一下，后面就数织(补一句：我单推沃玛，沃玛8月13的视频就用了数织所以才知道这是数织的。见：沃玛的生活/第六期)来反推二维码。然后行数据是对的，但是每行所处的顺序错了，可以根据二维码的固定位置来反推出数织，然后会出现中间一部分无法确定的情况，用脚本爆破然后去try扫描二维码即可。

那有可能没有听懂是什么意思，这里再说明白一点。
就是首先把数织给的数，放到数织里面去解，会解出一张看起来毫无关系的图。但是你仔细看就会发现，行的数据是正确的，意思是二维码已经在图上了，但是这29行，被打乱了。
然后就需要根据二维码的性质，把开头的几行，最后的几行给手动恢复一下，中间的十几行，只需要上下移动即可，但是并不知道正确的图是哪样的，所以需要写脚本来爆破，然后每爆一张，就自动扫描一张，能扫出来的图就是flag。