XCTF高校战“疫”网络安全分享赛Misc wp

原创

z.volcano 于 2021-04-18 11:12:59 发布 972 收藏 1

分类专栏： ctf # 比赛&复现

本文链接：https://blog.csdn.net/weixin_45696568/article/details/115818172

ctf 同时被 2 个专栏收录

9 篇文章 0 订阅

订阅专栏

比赛&复现

4 篇文章 0 订阅

订阅专栏

题目地址

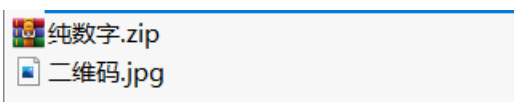
赛题

题目

2019-nCoV

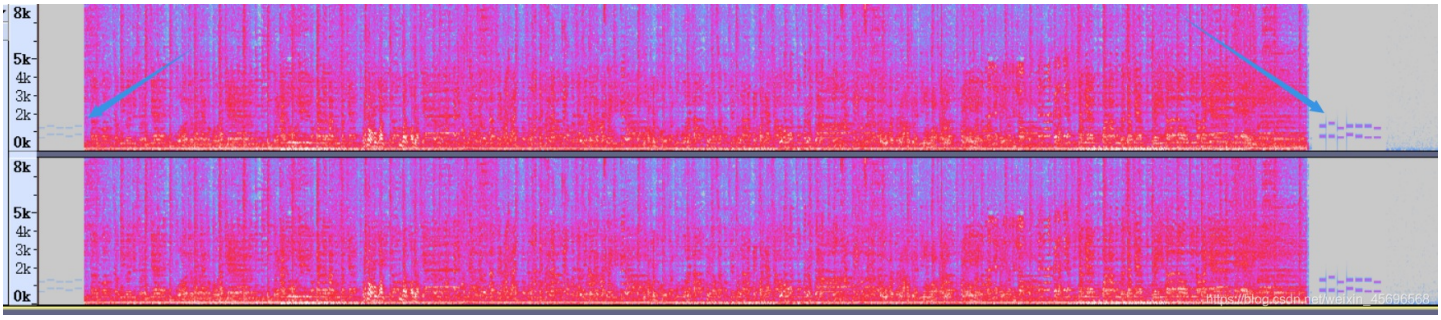
点开就给flag

隐藏的信息



我上来先爆破压缩包的密码，后面发现这个是`伪加密`，破解之后拿到一个wav文件，放到`Audacity`里分析，`查看频谱图`，有发现。

头部和尾部都有东西



这里是拨号音隐写，双音多频，用脚本撸或者对照表格看。

双音多频键盘

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

最后得到一串数字`187485618521`。

再回过头看那个二维码，进过反色处理后再补全定位符，会发现上当受骗了。
用winhex打开，发现两串字符

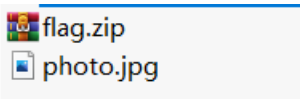
```
000091E0  7D FF 00 C9 34 7F C2 63  F1 BF FE 8E 07 E2 07 FE  }ÿ É4 Âcñ¿þŽ â þ
000091F0  0F 2F BF F9 26 8A 28 00  FF 00 84 C7 E3 7F FD 1C   /¿ù&Š( ÿ „Çã ý
00009200  0F C4 0F FC 1E 55 53 45  42 41 53 45 36 34 FC 6F   Ä ü USEBASE64üo
00009210  FF 00 A3 81 F8 81 FF 00  83 CB EF FE 49 A2 8A 00  ÿ £ ø ÿ ƒËïþI¢Š
00009220  3F E1 31 E8 DF FF 00 47  03 E1 03 FF 00 07 97 DF  ?á1èßÿ G á ÿ —ß
```

```
00009470  CF A7 FF 00 67 45 14 50  07 FF D9 54 4F 47 45 54  Ï§ÿ gE P ÿÙTOGET
00009480  59 4F 55 52 46 4C 41 47                          YOURFLAG
```

所以把得到的数字base64编码一次，再套上flag{}，即`flag{MTg3NDg1NjE4NTIx}`

简单MISC

也是两个文件



压缩包是加密的，先看`photo.jpg`，winhex分析发现有额外信息，foremost分离出一个压缩包，里面有一个txt文件，内容如下

```
./.–/…/-…/./–/…/-.-/…/…/-/…-/.-/-/…/—/-./—/…-./…-/-./…/…-/./.-/…/…/-/-.–/.–/.-/.-.
```

显然是摩斯密码，解密得到`EPIDEMICSITUATIONOFUNIVERSITYWAR`，这个是压缩包的密码

压缩包里面有个txt，内容

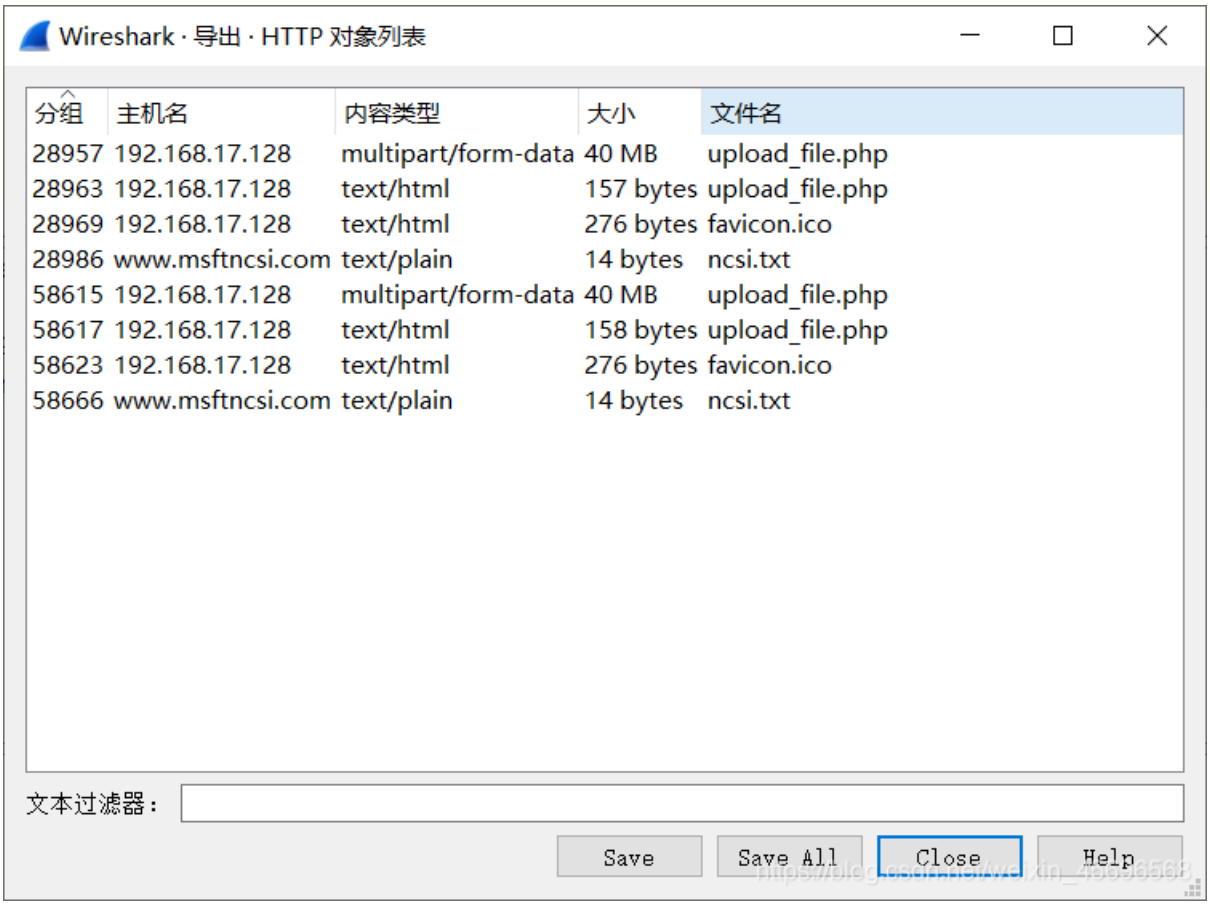
> VGgxc19pc19GbGFHX3lvdV9hUkVfcmlnSFQ=

base64解密得到flag

ez_mem&usb

> usb内存取证，最终得到的结果需要转换为小写字母

下载附件拿到一个流量包，扔进wireshark里，导出http流发现有这些东西



把文件提取出来后，把`upload_file.php`扔进010editor里，发现有压缩包，binwalk提取出来，最后拿到压缩包里面的`data.vmem`文件

然后掏出`volatility`进行分析
先看一下系统版本信息:`python vol.py -f data.vmem imageinfo`

```
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
                     AS Layer1 : IA32PagedMemoryPae (Kernel AS)
                     AS Layer2 : FileAddressSpace (/home/volcano/桌面/volatility-master/data.vmem)
                      PAE type : PAE
                           DTB : 0xb18000L
                          KDBG : 0x80546ae0L
          Number of Processors : 1
     Image Type (Service Pack) : 3
                KPCR for CPU 0 : 0xffdff000L
             KUSER_SHARED_DATA : 0xffdf0000L
           Image date and time : 2020-02-24 07:56:47 UTC+0000
     Image local date and time : 2020-02-24 15:56:47 +0800
```

得到可能性最大的架构`WinXPSP2x86`

再看看进程：`python vol.py --profile=WinXPSP2x86 -f data.vmem pslist`

```
Offset(V)  Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start                          Exit
---------- -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ -----
-------------------------
0x80ea2660 System                    4      0     52      231 ------      0
0xff57fc28 smss.exe                372      4      3       19 ------      0 2020-02-23 13:17:13 UTC+0000
0xff432020 csrss.exe               464    372     12      317      0      0 2020-02-23 13:17:13 UTC+0000
0xff435020 winlogon.exe            492    372     20      501      0      0 2020-02-23 13:17:13 UTC+0000
0xff445020 services.exe            668    492     16      253      0      0 2020-02-23 13:17:14 UTC+0000
0xff46b020 lsass.exe               680    492     19      309      0      0 2020-02-23 13:17:14 UTC+0000
0xff510bf0 vmacthlp.exe            836    668      1       25      0      0 2020-02-23 13:17:14 UTC+0000
0xff493568 svchost.exe             848    668     14      189      0      0 2020-02-23 13:17:14 UTC+0000
0xff491a78 svchost.exe             932    668     11      230      0      0 2020-02-23 13:17:14 UTC+0000
0xff416b10 svchost.exe            1024    668     44      939      0      0 2020-02-23 13:17:14 UTC+0000
0x80dac020 svchost.exe            1072    668      4       57      0      0 2020-02-23 13:17:14 UTC+0000
0xff4ca4e0 svchost.exe            1132    668      7      118      0      0 2020-02-23 13:17:14 UTC+0000
0xff30d020 explorer.exe           1476   1400     13      481      0      0 2020-02-23 13:17:15 UTC+0000
0xff51d468 spoolsv.exe            1568    668     10      120      0      0 2020-02-23 13:17:15 UTC+0000
0xff5793d8 VGAuthService.e        1932    668      2       60      0      0 2020-02-23 13:17:33 UTC+0000
0xff576da0 vmtoolsd.exe           2008    668      7      265      0      0 2020-02-23 13:17:40 UTC+0000
0xff4afda0 wmiprvse.exe            540    848     13      242      0      0 2020-02-23 13:17:41 UTC+0000
0xff486da0 vmtoolsd.exe            588   1476      6      229      0      0 2020-02-23 13:17:41 UTC+0000
0xff47dda0 ctfmon.exe              596   1476      1       71      0      0 2020-02-23 13:17:41 UTC+0000
0xff5b57b8 cmd.exe                1396   1476      1       61      0      0 2020-02-23 13:24:09 UTC+0000
0xff4583c0 conime.exe              544   1396      1       38      0      0 2020-02-23 13:24:09 UTC+0000
```

比较可疑的就是这个cmd.exe了

所以看一下 cmd 命令使用情况：`python vol.py --profile=WinXPSP2x86 -f data.vmem cmdscan`

```
**************************************************
CommandProcess: csrss.exe Pid: 464
CommandHistory: 0x556bb8 Application: cmd.exe Flags: Allocated, Reset
CommandCount: 2 LastAdded: 1 LastDisplayed: 1
FirstCommand: 0 CommandCountMax: 50
ProcessHandle: 0x504
Cmd #0 @ 0x3609ea0: passwd:weak_auth_top100
Cmd #1 @ 0x5576d0: start wireshark
Cmd #13 @ 0x9f009f: ??
Cmd #41 @ 0x9f003f: ?\?????????
```

前两条命令是有用的，首先得到密码`weak_auth_top100`，然后启动了wireshark

再扫一波文件：`python vol.py --profile=WinXPSP2x86 -f data.vmem filescan`

```
0x00000000011556f0      1      0 R------ \Device\HarddiskVolume1\WINDOWS\system32\drivers\scsiport.sys
0x0000000001155990      1      0 R--r-d \Device\HarddiskVolume1\WINDOWS\system32\compatUI.dll
0x0000000001155f90      1      0 R--rwd \Device\HarddiskVolume1\Documents and Settings\Administrator\flag.i
mg
0x0000000001156768      3      0 RWD--- \Device\HarddiskVolume1\$Directory
0x00000000011569a8      1      0 R--rwd \Device\HarddiskVolume1\Documents and Settings\All Users\「开始」菜
单\程序\附件\desktop.ini
0x0000000001156ae8      1      1 ------ \Device\NamedPipe\srvsvc
```

发现可疑文件`flag.img`，把它导出来：`python vol.py --profile=WinXPSP2x86 -f data.vmem dumpfiles -Q 0x0000000001155f90 -n --dump-dir=./`，得到文件

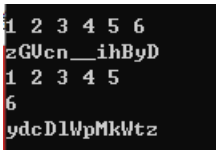
`foremost`分离出一个加密的压缩包（这里用binwalk可能会出问题），用前面得到的密码解密。
拿到`usbdata.txt`，里面是usb流量数据，很明显是键盘流量，用网上找来的脚本，运行之后得到flag

```
mappings = { 0x04:"A",  0x05:"B",  0x06:"C", 0x07:"D", 0x08:"E", 0x09:"F", 0x0A:"G",  0x0B:"H", 0x0C:"I",  0x0D:
"J", 0x0E:"K", 0x0F:"L", 0x10:"M", 0x11:"N",0x12:"O",  0x13:"P", 0x14:"Q", 0x15:"R", 0x16:"S", 0x17:"T", 0x18:"U
",0x19:"V", 0x1A:"W", 0x1B:"X", 0x1C:"Y", 0x1D:"Z", 0x1E:"1", 0x1F:"2", 0x20:"3", 0x21:"4", 0x22:"5",  0x23:"6",
 0x24:"7", 0x25:"8", 0x26:"9", 0x27:"0", 0x28:"\n", 0x2a:"[DEL]",  0X2B:"    ", 0x2C:" ",  0x2D:"-", 0x2E:"=", 0
x2F:"[",  0x30:"]",  0x31:"\\", 0x32:"~", 0x33:";",  0x34:"'", 0x36:",",  0x37:"." }
nums = []
keys = open('usbdata.txt')
for line in keys:
    if line[0]!='0' or line[1]!='0' or line[3]!='0' or line[4]!='0' or line[9]!='0' or line[10]!='0' or line[12]
!='0' or line[13]!='0' or line[15]!='0' or line[16]!='0' or line[18]!='0' or line[19]!='0' or line[21]!='0' or l
ine[22]!='0':
        continue
    nums.append(int(line[6:8],16))
keys.close()
output = ""
for n in nums:
    if n == 0 :
        continue
    if n in mappings:
        output += mappings[n]
    else:
        output += '[unknown]'
print ('output :\n' + output)
```

武汉加油

binwalk分离出一个压缩包，里面是flag.exe文件

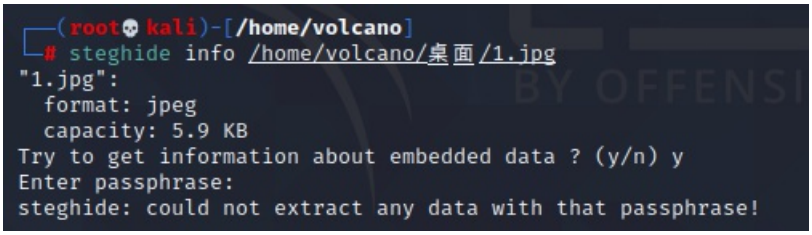
打开测试了一波，发现`每输入6个字符`，它会返回一个字符串



应该是输入特定字符，返回flag，可惜我不会逆向...

应该不是靠硬猜，换个方向看有没有提示，因为是jpg图片，所以使用`outguess`和`steghide`查看一下

发现是需要密码的`steghide加密`



steghide本身不支持爆破，这里借助https://github.com/Va5c0/Steghide-Brute-Force-Tool

```
具体用法：
python steg_brute.py -b -d [字典] -f [jpg_file]
这里需要安装一个库：
pip install progressbar2
```

最后得到密码是`ctf`，提取`steghide extract -p 'ctf' -sf 1.jpg`

得到flag.txt

> 新型冠状病毒感染的肺炎疫情牵动全国人心，大家守望相助、众志成城、共克时艰，一起驰援武汉。相信在党中央、国务院的领导下，一定能打赢这场没有硝烟的疫情防控战役。
> ' 武 汉 加 油 ! --HEUctfer

很明显，flag.exe需要的6个字符就是`' 武 汉 加 油 !`