CTF——MISC习题讲解（GKCTF 2021系列）

原创

TJA小傲 已于 2022-03-04 10:34:42 修改 1852 收藏 2

分类专栏： CTF-Misc 文章标签： 安全

于 2022-03-04 10:20:46 首次发布

本文链接：https://blog.csdn.net/tlovejr/article/details/123245417

版权

CTF-Misc 专栏收录该内容

9 篇文章 0 订阅

订阅专栏

CTF——MISC习题讲解（GKCTF 2021系列）

前言

接下来陆续给大家复现一些赛事的杂项习题讲解，因为本人也是小白入门，有些题目做的不对还请各位大佬多多包涵。

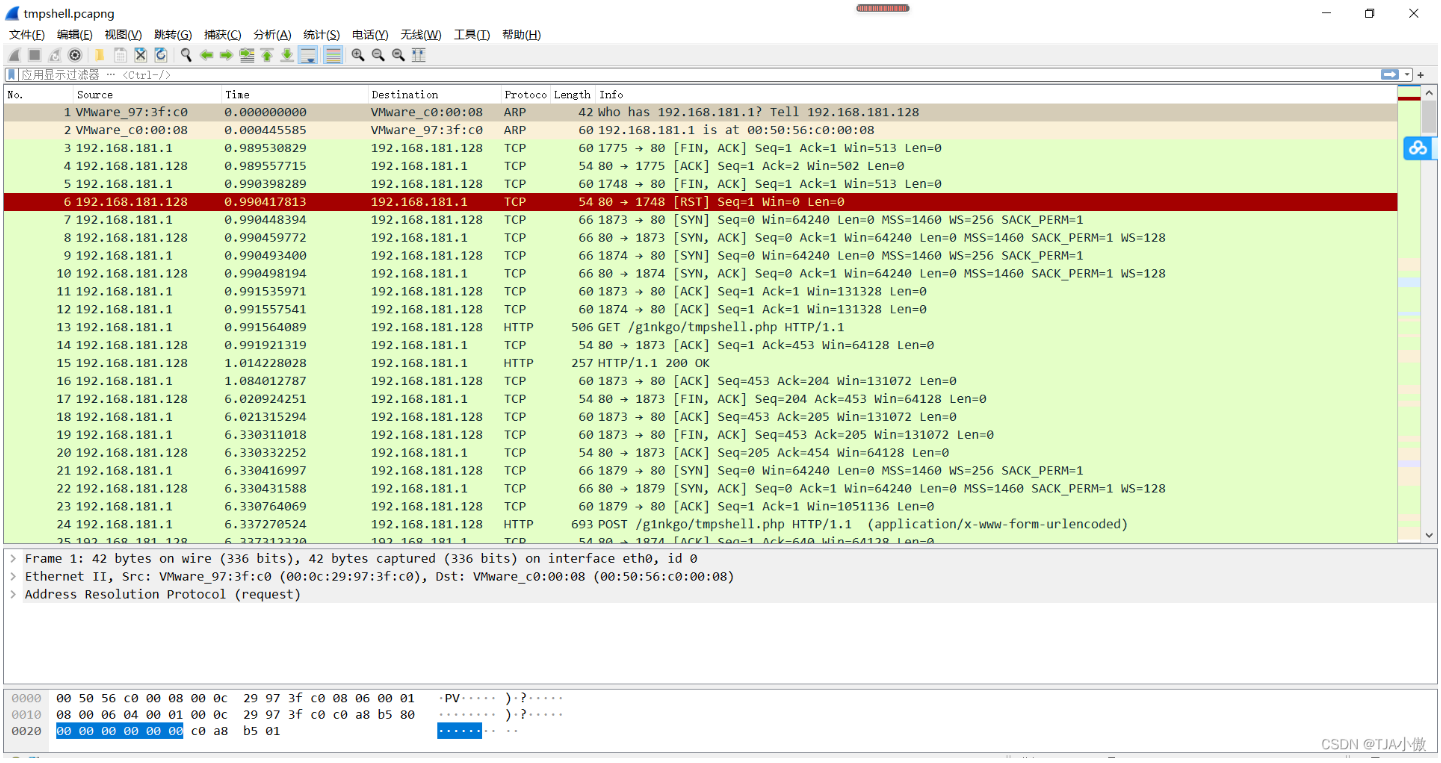
一、[GKCTF 2021]签到

题目链接如下：

链接：https://pan.baidu.com/s/1rlBOLJMn-nYCsuyCT3eOSg?pwd=lfuj

提取码：lfuj

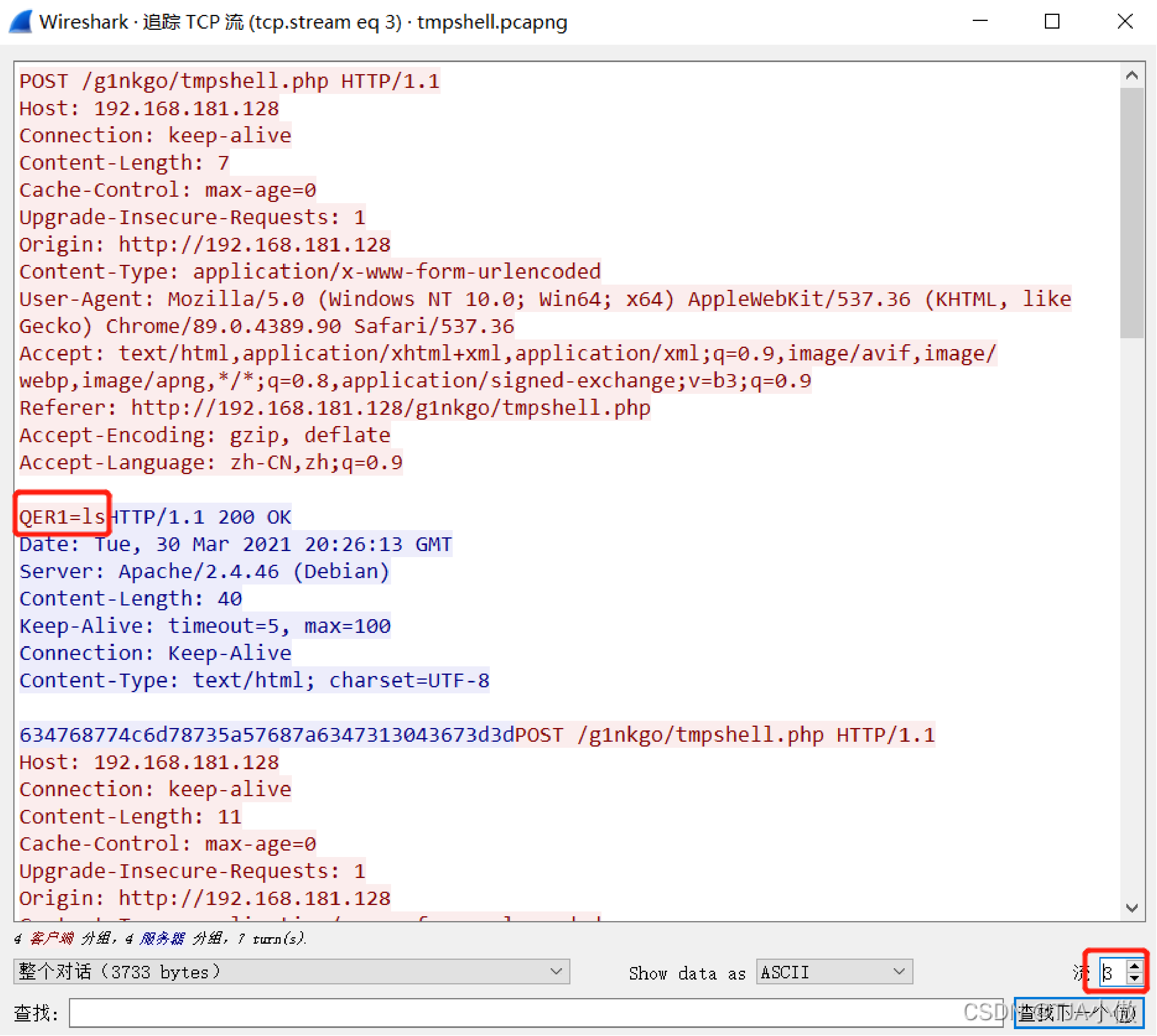
打开题目后是一个流量分析题目


tmpshell.pcapng
文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)
应用显示过滤器 … <Ctrl-/>
CSDN @TJA小傲

```
No.  Source            Time         Destination       Protoco Length Info
   1 VMware_97:3f:c0   0.000000000  VMware_c0:00:08   ARP      42 Who has 192.168.181.1? Tell 192.168.181.128
   2 VMware_c0:00:08   0.000445585  VMware_97:3f:c0   ARP      60 192.168.181.1 is at 00:50:56:c0:00:08
   3 192.168.181.1     0.989530829  192.168.181.128   TCP      60 1775 → 80 [FIN, ACK] Seq=1 Ack=1 Win=513 Len=0
   4 192.168.181.128   0.989557715  192.168.181.1     TCP      54 80 → 1775 [ACK] Seq=1 Ack=2 Win=502 Len=0
   5 192.168.181.1     0.990398289  192.168.181.128   TCP      60 1748 → 80 [FIN, ACK] Seq=1 Ack=1 Win=513 Len=0
   6 192.168.181.128   0.990417813  192.168.181.1     TCP      54 80 → 1748 [RST] Seq=1 Win=0 Len=0
   7 192.168.181.1     0.990448394  192.168.181.128   TCP      66 1873 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
   8 192.168.181.128   0.990459772  192.168.181.1     TCP      66 80 → 1873 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
   9 192.168.181.1     0.990493400  192.168.181.128   TCP      66 1874 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
  10 192.168.181.128   0.990498194  192.168.181.1     TCP      66 80 → 1874 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
  11 192.168.181.1     0.991535971  192.168.181.128   TCP      60 1873 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
  12 192.168.181.1     0.991557541  192.168.181.128   TCP      60 1874 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
  13 192.168.181.1     0.991564089  192.168.181.128   HTTP    506 GET /g1nkgo/tmpshell.php HTTP/1.1
  14 192.168.181.128   0.991921319  192.168.181.1     TCP      54 80 → 1873 [ACK] Seq=1 Ack=453 Win=64128 Len=0
  15 192.168.181.128   1.014228028  192.168.181.1     HTTP    257 HTTP/1.1 200 OK
  16 192.168.181.1     1.084012787  192.168.181.128   TCP      60 1873 → 80 [ACK] Seq=453 Ack=204 Win=131072 Len=0
  17 192.168.181.128   6.020924251  192.168.181.1     TCP      54 80 → 1873 [FIN, ACK] Seq=204 Ack=453 Win=64128 Len=0
  18 192.168.181.1     6.021315294  192.168.181.128   TCP      60 1873 → 80 [ACK] Seq=453 Ack=205 Win=131072 Len=0
  19 192.168.181.1     6.330311018  192.168.181.128   TCP      60 1873 → 80 [FIN, ACK] Seq=453 Ack=205 Win=131072 Len=0
  20 192.168.181.128   6.330332252  192.168.181.1     TCP      54 80 → 1873 [ACK] Seq=205 Ack=454 Win=64128 Len=0
  21 192.168.181.1     6.330416997  192.168.181.128   TCP      66 1879 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
  22 192.168.181.128   6.330431588  192.168.181.1     TCP      66 80 → 1879 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
  23 192.168.181.1     6.330764069  192.168.181.128   TCP      60 1879 → 80 [ACK] Seq=1 Ack=1 Win=1051136 Len=0
  24 192.168.181.1     6.337270524  192.168.181.128   HTTP    693 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  25 192.168.181.128   6.337312320  192.168.181.1     TCP      54 80 → 1874 [ACK] Seq=1 Ack=640 Win=64128 Len=0

> Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
> Ethernet II, Src: VMware_97:3f:c0 (00:0c:29:97:3f:c0), Dst: VMware_c0:00:08 (00:50:56:c0:00:08)
> Address Resolution Protocol (request)

0000  00 50 56 c0 00 08 00 0c  29 97 3f c0 08 06 00 01   ·PV····· )·?·····
0010  08 00 06 04 00 01 00 0c  29 97 3f c0 c0 a8 b5 80   ········ )·?·····
0020  00 00 00 00 00 00 c0 a8  b5 01                     ········ ··
```

然后我们看一下http协议，并追踪TCP流可以发现，在众多HTTP协议中，好像是执行Linux系统命令。

在这里进行了ls查看



```
POST /g1nkgo/tmpshell.php HTTP/1.1
Host: 192.168.181.128
Connection: keep-alive
Content-Length: 7
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.181.128
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/89.0.4389.90 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.181.128/g1nkgo/tmpshell.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9

QER1=lsHTTP/1.1 200 OK
Date: Tue, 30 Mar 2021 20:26:13 GMT
Server: Apache/2.4.46 (Debian)
Content-Length: 40
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

634768774c6d78735a57687a6347313043673d3dPOST /g1nkgo/tmpshell.php HTTP/1.1
Host: 192.168.181.128
Connection: keep-alive
Content-Length: 11
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.181.128
```

4 客户端 分组, 4 服务器 分组, 7 turn(s).

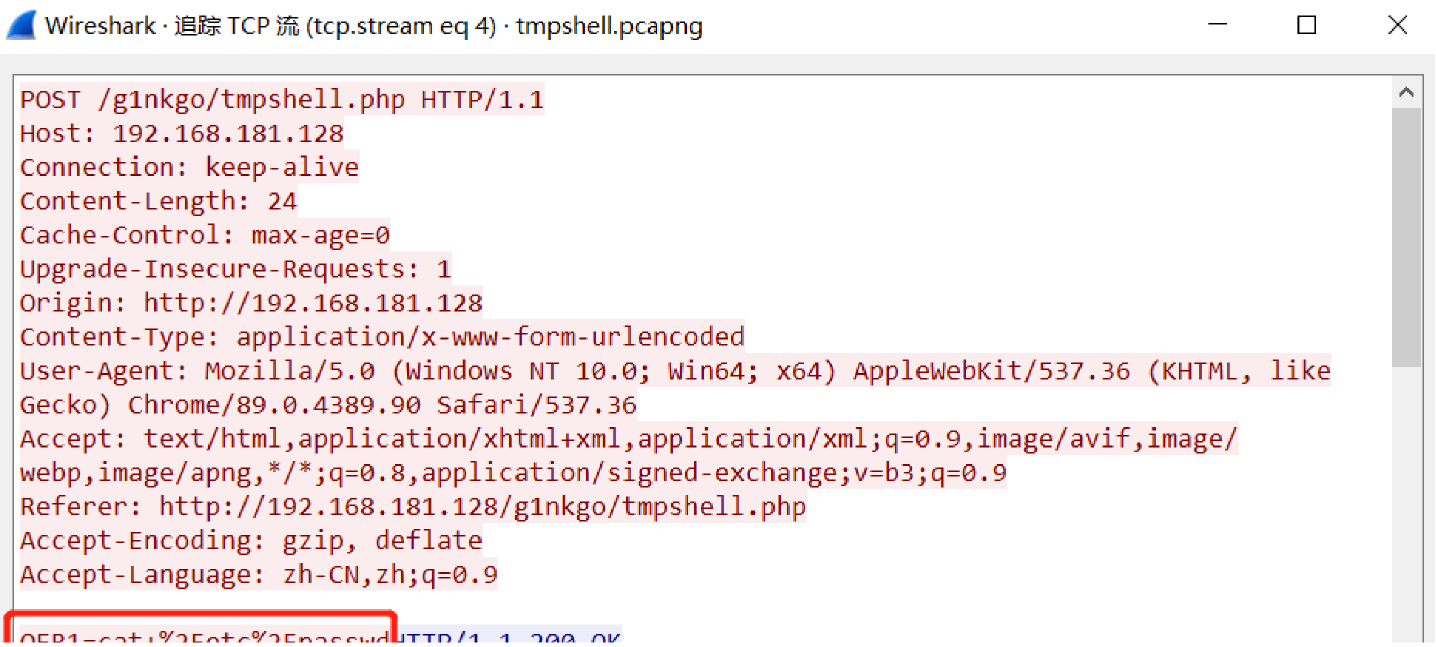
整个对话（3733 bytes）

Show data as ASCII

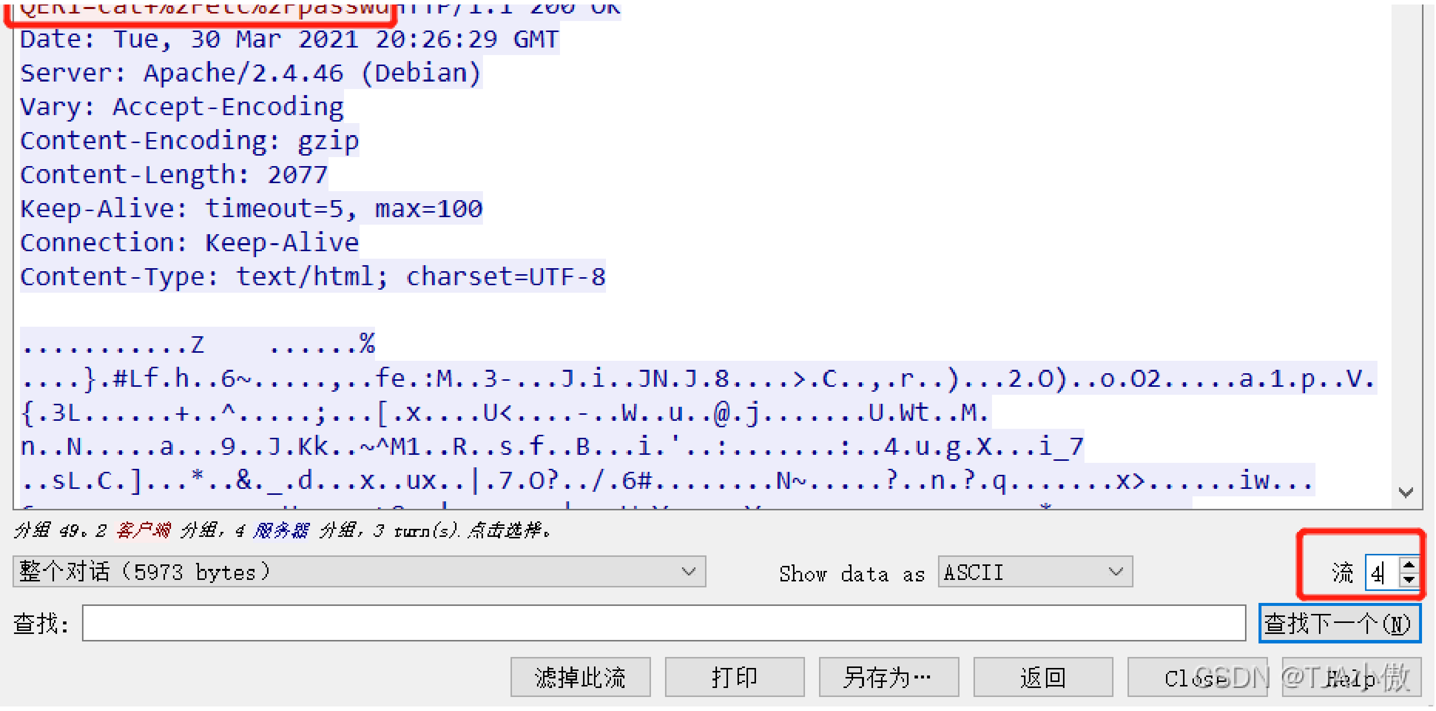
流 3

查找:

在这里进行了cat /etc/passwd命令，在这里，+代表的是空格，%2f代表的是/



```
POST /g1nkgo/tmpshell.php HTTP/1.1
Host: 192.168.181.128
Connection: keep-alive
Content-Length: 24
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.181.128
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/89.0.4389.90 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.181.128/g1nkgo/tmpshell.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9

QER1=cat+%2Fetc%2FpasswdHTTP/1.1 200 OK
```

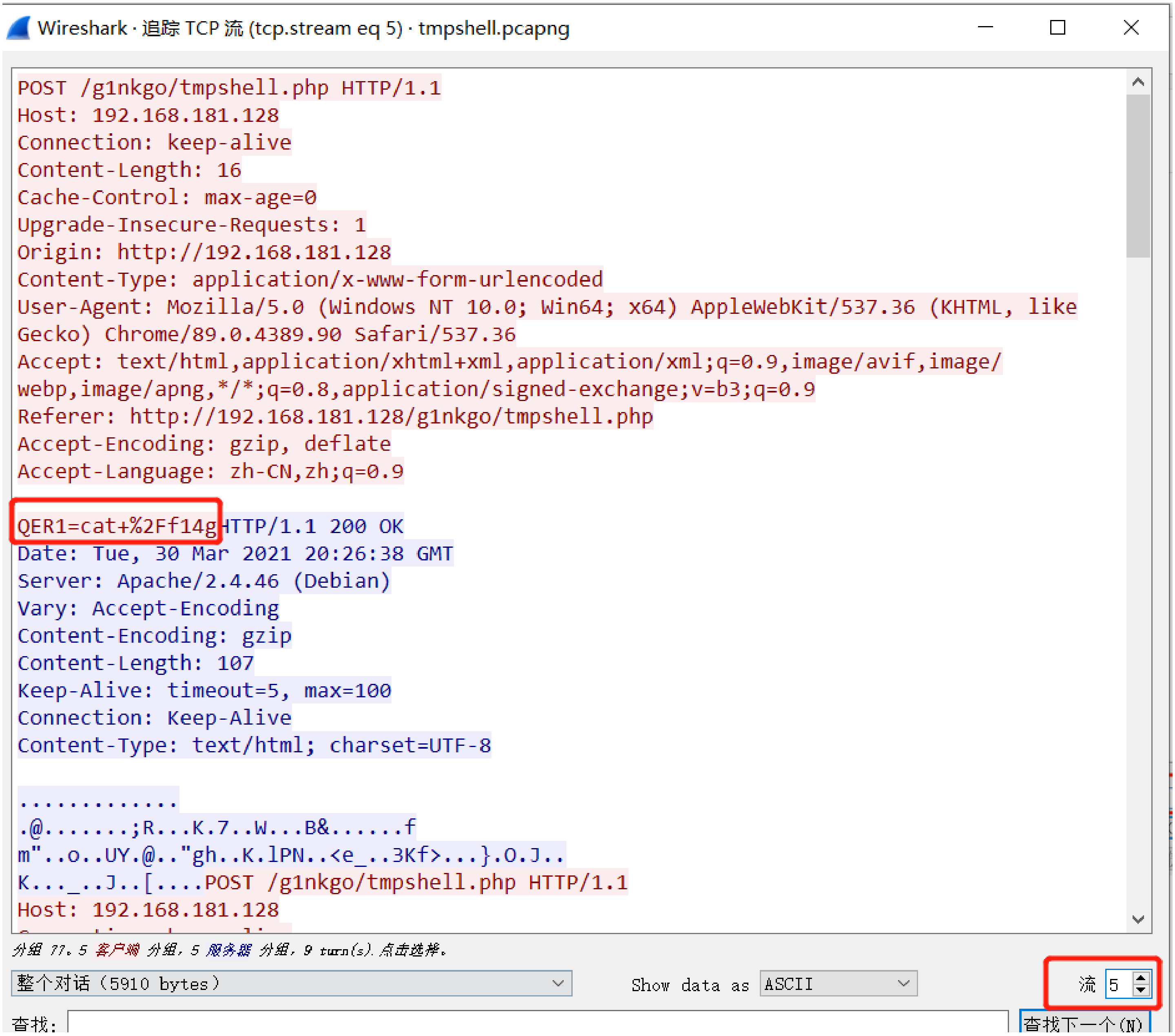


```
QER1=cat+%2Fetc%2FpasswdHTTP/1.1 200 OK
Date: Tue, 30 Mar 2021 20:26:29 GMT
Server: Apache/2.4.46 (Debian)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 2077
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

...........Z    ......%
....}.#Lf.h..6~.....,..fe.:M..3-...J.i..JN.J.8....>.C..,.r..)...2.O)..o.O2.....a.1.p..V.
{.3L......+..^.....;...[.x....U<....-..W..u..@.j.......U.Wt..M.
n..N.....a...9..J.Kk..~^M1..R..s.f..B...i.'..:.......:..4.u.g.X...i_7
..sL.C.]...*..&._.d...x..ux..|.7.O?../.6#........N~.....?..n.?.q.......x>......iw...
```

分组 49。2 客户端 分组，4 服务器 分组，3 turn(s). 点击选择。

整个对话（5973 bytes） Show data as ASCII 流 4

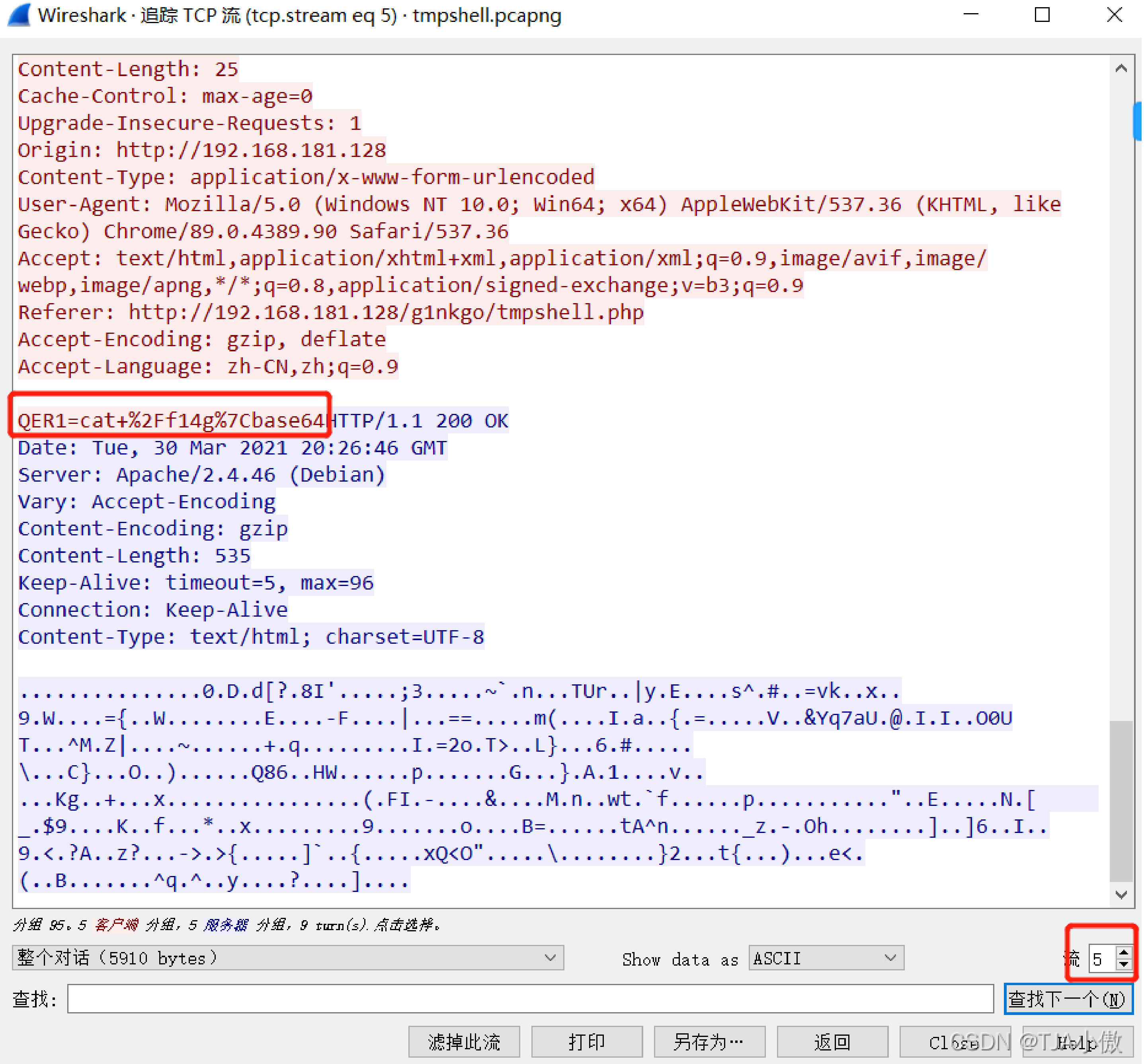
在这里输入命令cat /f14g



```
POST /g1nkgo/tmpshell.php HTTP/1.1
Host: 192.168.181.128
Connection: keep-alive
Content-Length: 16
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.181.128
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/89.0.4389.90 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.181.128/g1nkgo/tmpshell.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9

QER1=cat+%2Ff14gHTTP/1.1 200 OK
Date: Tue, 30 Mar 2021 20:26:38 GMT
Server: Apache/2.4.46 (Debian)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 107
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

.............
.@.......;R...K.7..W...B&......f
m"..o..UY.@.."gh..K.1PN..<e_..3Kf>...}.O.J..
K..._..J..[....POST /g1nkgo/tmpshell.php HTTP/1.1
Host: 192.168.181.128
```

分组 77。5 客户端 分组，5 服务器 分组，9 turn(s). 点击选择。

整个对话（5910 bytes） Show data as ASCII 流 5



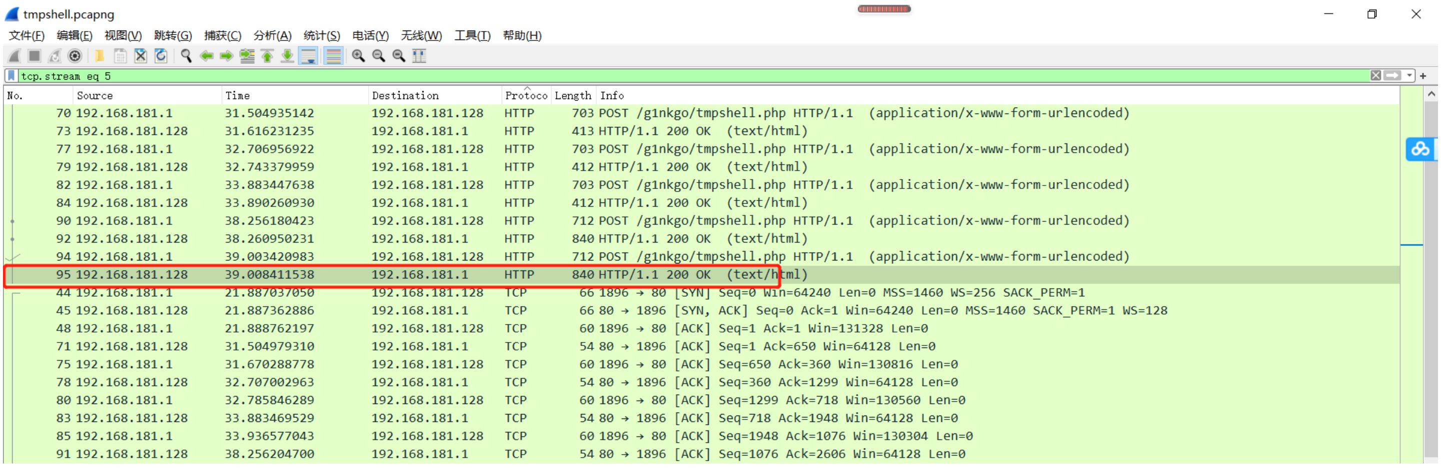
```
Content-Length: 25
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.181.128
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/89.0.4389.90 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.181.128/g1nkgo/tmpshell.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9

QER1=cat+%2Ff14g%7Cbase64HTTP/1.1 200 OK
Date: Tue, 30 Mar 2021 20:26:46 GMT
Server: Apache/2.4.46 (Debian)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 535
Keep-Alive: timeout=5, max=96
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

...............0.D.d[?.8I'.....;3.....~`.n...TUr..|y.E....s^.#..=vk..x..
9.W....={..W........E....-F....|...==.....m(....I.a..{.=.....V..&Yq7aU.@.I.I..O0U
T...^M.Z|....~......+.q.........I.=2o.T>..L}...6.#.....
\...C}...O..)......Q86..HW......p.......G...}.A.1....v..
...Kg..+...x...............(.FI.-....&....M.n..wt.`f......p..........."..E.....N.[
_.$9....K..f...*..x.........9.......o....B=......tA^n......_z.-.Oh........]..]6..I..
9.<.?A..z?...->.>{.....]`..{.....xQ<O".....\........}2...t{...)...e<.
(..B.......^q.^..y....?....]....
```

分组 95。5 客户端 分组，5 服务器 分组，9 turn(s).点击选择。

整个对话（5910 bytes） Show data as ASCII 流 5

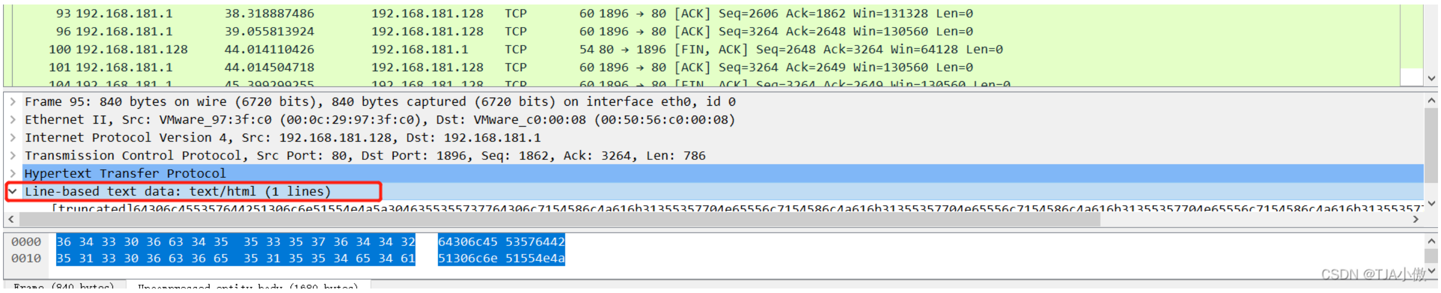
查找：

在tcp.stream eq 5找到cat flag相关信息，然后查看分组字节流


tmpshell.pcapng
文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)

tcp.stream eq 5

```
No.  Source           Time          Destination      Protoco Length Info
  70 192.168.181.1    31.504935142  192.168.181.128  HTTP    703 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  73 192.168.181.128  31.616231235  192.168.181.1    HTTP    413 HTTP/1.1 200 OK  (text/html)
  77 192.168.181.1    32.706956922  192.168.181.128  HTTP    703 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  79 192.168.181.128  32.743379959  192.168.181.1    HTTP    412 HTTP/1.1 200 OK  (text/html)
  82 192.168.181.1    33.883447638  192.168.181.128  HTTP    703 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  84 192.168.181.128  33.890260930  192.168.181.1    HTTP    412 HTTP/1.1 200 OK  (text/html)
  90 192.168.181.1    38.256180423  192.168.181.128  HTTP    712 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  92 192.168.181.128  38.260950231  192.168.181.1    HTTP    840 HTTP/1.1 200 OK  (text/html)
  94 192.168.181.1    39.003420983  192.168.181.128  HTTP    712 POST /g1nkgo/tmpshell.php HTTP/1.1  (application/x-www-form-urlencoded)
  95 192.168.181.128  39.008411538  192.168.181.1    HTTP    840 HTTP/1.1 200 OK  (text/html)
  44 192.168.181.1    21.887037050  192.168.181.128  TCP      66 1896 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
  45 192.168.181.128  21.887362886  192.168.181.1    TCP      66 80 → 1896 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
  48 192.168.181.1    21.888762197  192.168.181.128  TCP      60 1896 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
  71 192.168.181.128  31.504979310  192.168.181.1    TCP      54 80 → 1896 [ACK] Seq=1 Ack=650 Win=64128 Len=0
  75 192.168.181.1    31.670288778  192.168.181.128  TCP      60 1896 → 80 [ACK] Seq=650 Ack=360 Win=130816 Len=0
  78 192.168.181.128  32.707002963  192.168.181.1    TCP      54 80 → 1896 [ACK] Seq=360 Ack=1299 Win=64128 Len=0
  80 192.168.181.1    32.785846289  192.168.181.128  TCP      60 1896 → 80 [ACK] Seq=1299 Ack=718 Win=130560 Len=0
  83 192.168.181.128  33.883469529  192.168.181.1    TCP      54 80 → 1896 [ACK] Seq=718 Ack=1948 Win=64128 Len=0
  85 192.168.181.1    33.936577043  192.168.181.128  TCP      60 1896 → 80 [ACK] Seq=1948 Ack=1076 Win=130304 Len=0
  91 192.168.181.128  38.256204700  192.168.181.1    TCP      54 80 → 1896 [ACK] Seq=1076 Ack=2606 Win=64128 Len=0
```

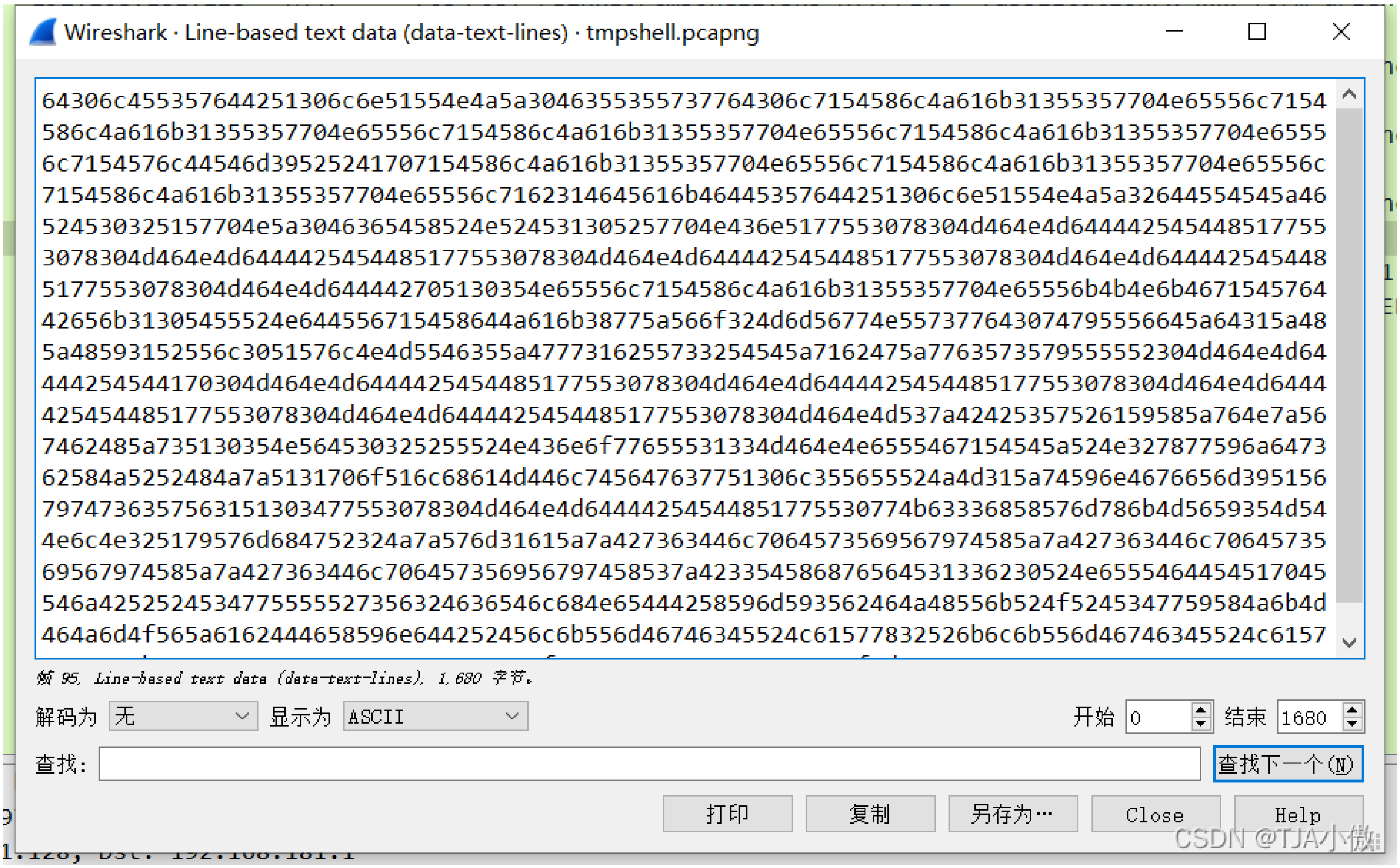


```
 93 192.168.181.1     38.318887486   192.168.181.128  TCP   60 1896 → 80 [ACK] Seq=2606 Ack=1862 Win=131328 Len=0
 96 192.168.181.1     39.055813924   192.168.181.128  TCP   60 1896 → 80 [ACK] Seq=3264 Ack=2648 Win=130560 Len=0
100 192.168.181.128   44.014110426   192.168.181.1    TCP   54 80 → 1896 [FIN, ACK] Seq=2648 Ack=3264 Win=64128 Len=0
101 192.168.181.1     44.014504718   192.168.181.128  TCP   60 1896 → 80 [ACK] Seq=3264 Ack=2649 Win=130560 Len=0
104 192.168.181.1     45.399299255   192.168.181.128  TCP   60 1896 → 80 [FIN, ACK] Seq=3264 Ack=2649 Win=130560 Len=0
```

```
> Frame 95: 840 bytes on wire (6720 bits), 840 bytes captured (6720 bits) on interface eth0, id 0
> Ethernet II, Src: VMware_97:3f:c0 (00:0c:29:97:3f:c0), Dst: VMware_c0:00:08 (00:50:56:c0:00:08)
> Internet Protocol Version 4, Src: 192.168.181.128, Dst: 192.168.181.1
> Transmission Control Protocol, Src Port: 80, Dst Port: 1896, Seq: 1862, Ack: 3264, Len: 786
> Hypertext Transfer Protocol
v Line-based text data: text/html (1 lines)
    [truncated]64306c455357644251306c6e51554e4a5a304635535573776430 6c7154586c4a616b31355357704e65556c7154586c4a616b31355357704e65556c7154586c4a616b31355357
```

```
0000  36 34 33 30 36 63 34 35  35 33 35 37 36 34 34 32   64306c45 53576442
0010  35 31 33 30 36 63 36 65  35 31 35 35 34 65 34 61   51306c6e 51554e4a
```

Frame (840 bytes) Uncompressed entity body (1680 bytes)



Wireshark · Line-based text data (data-text-lines) · tmpshell.pcapng

```
64306c455357644251306c6e51554e4a5a304635535573776430 6c7154586c4a616b31355357704e65556c7154
586c4a616b31355357704e65556c7154586c4a616b31355357704e65556c7154586c4a616b31355357704e6555
6c7154576c44546d39525241707154586c4a616b31355357704e65556c7154586c4a616b31355357704e65556c
7154586c4a616b31355357704e65556c7162314645616b46445357644251306c6e51554e4a5a32644554545a46
524530325157704e5a304636545852 4e524531305257704e436e5177553078304d464e4d6444425454485177 55
3078304d464e4d644442545448517755 3078304d464e4d6444425454485177553078304d464e4d644442545448
5177553078304d464e4d6444427051303 54e65556c7154586c4a616b31355357704e65556b4b4e6b4671545764
42656b31305455524e6445567154586 44a616b38775a566f324d6d56774e557377643074795556645a64315a48
5a48593152556c3051576c4e4d5546355a47773162557332545 45a7162475a77635735795555 52304d464e4d64
444254544170304d464e4d6444425454485177553078304d464e4d6444425454485177553078304d464e4d6444
4254544851775530 78304d464e4d6444425454485177553078304d464e4d537a42425357526159585a764e7a56
7462485a735130354e5645303252555 24e436e6f77655531334d464e4e6555467154545a524e327877596a6473
62584a5252484a7a5131706f516c68614d446c745647637751306c355655524a4d315a74596e4676656d395156
797473635756315130347755307830 4d464e4d6444425454485177553077 4b63336858576d786b4d5659354d54
4e6c4e325179576d684752324a7a5 76d31615a7a427363446c70645735 69567974585a7a427363446c70645735
69567974585a7a427363446c706457356956 7974585 37a42335458687656453133623052 4e6555464454517045
546a42525245347755555273563246 36546c684e65444258596d593562464a48556b524f5245347759584a6b4d
464a6d4f565a6162444658596e644252456c6b556d4674634552 4c61577832526b6c6b556d46746345524c6157
```

帧 95, Line-based text data (data-text-lines), 1,680 字节。

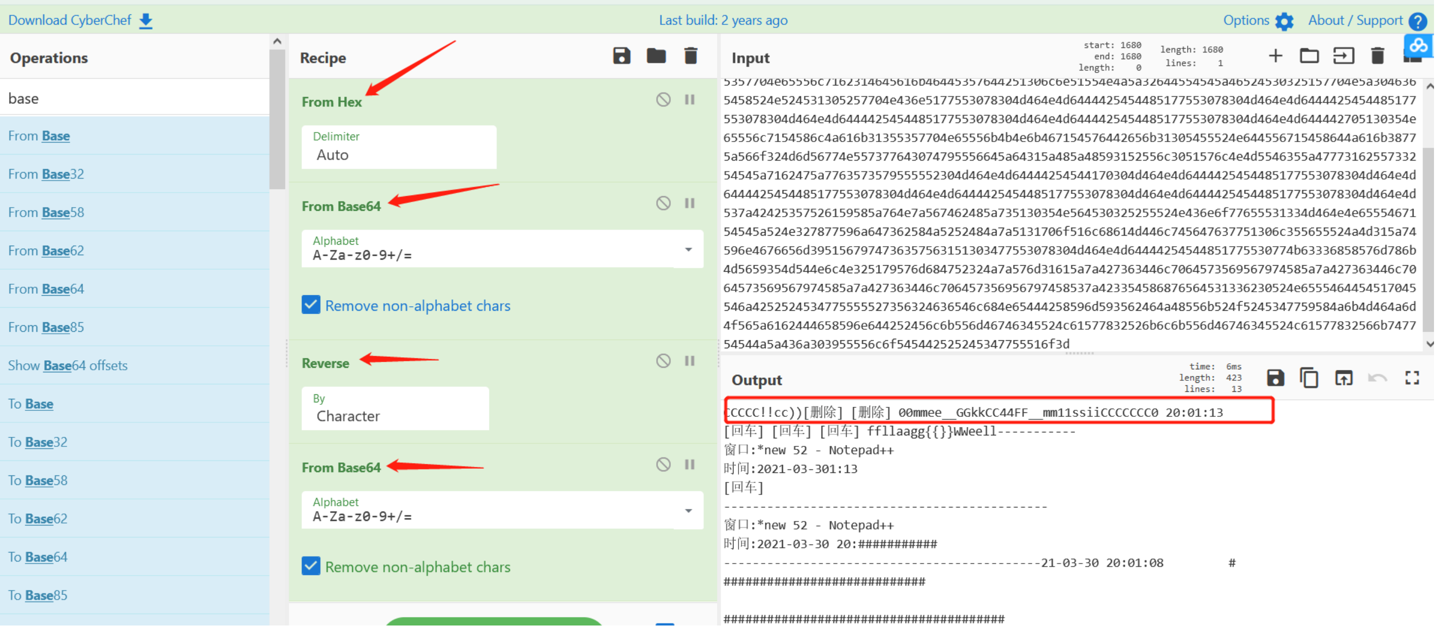
解码为 无 显示为 ASCII 开始 0 结束 1680

查找:

然后开始进行破解密码，破解的顺序为下

hex decode->base64 decode->rev

利用cyberchef解密



From Hex
Delimiter: Auto

From Base64
Alphabet: A-Za-z0-9+/=
☑ Remove non-alphabet chars

Reverse
By: Character

From Base64
Alphabet: A-Za-z0-9+/=
☑ Remove non-alphabet chars

Input start: 1680 end: 1680 length: 0 length: 1680 lines: 1

```
5357704e65556c7162314645616b46445357644251306c6e51554e4a5a32644554545a46524530325157704e5a304636
5458524e524531305257704e436e5177553078304d464e4d6444425454485177553078304d464e4d6444425454485177
553078304d464e4d6444425454485177553078304d464e4d6444425454485177553078304d464e4d64444270513035 4e
65556c7154586c4a616b31355357704e65556b4b4e6b4671545764 42656b31305455524e6445567154586 44a616b3877
5a566f324d6d56774e557377643074795556645a64315a485a48593152556c3051576c4e4d5546355a477731625573 32
54545a7162475a77635735795555 52304d464e4d644442545441 70304d464e4d6444425454485177553078304d464e4d
6444425454485177553078304d464e4d6444425454485177553078304d464e4d6444425454485177553078304d464e4d
537a42425357526159585a764e7a567462485a735130354e56453032525552 4e436e6f77655531334d464e4e65554671
54545a524e327877596a647362584a5252484a7a5131706f516c68614d446c745647637751306c355655524a4d315a74
596e4676656d39515679747363575675131303477553078304d464e4d64444254544851775530774b63336858576d786b
4d5659354d544e6c4e325179576d684752324a7a576d31615a7a427363446c706457356956797458 5a7a427363446c70
6457356956797458537a42335458687656453133623052 4e6555464454517045
546a42525245347755555273563246 36546c684e65444258596d593562464a48556b524f5245347759584a6b4d464a6d
4f565a6162444658596e644252456c6b556d46746345524c61577832526b6c6b556d46746345524c61577832566b7477
54544a5a436a303955556c6f54544252524 5347755516f3d
```

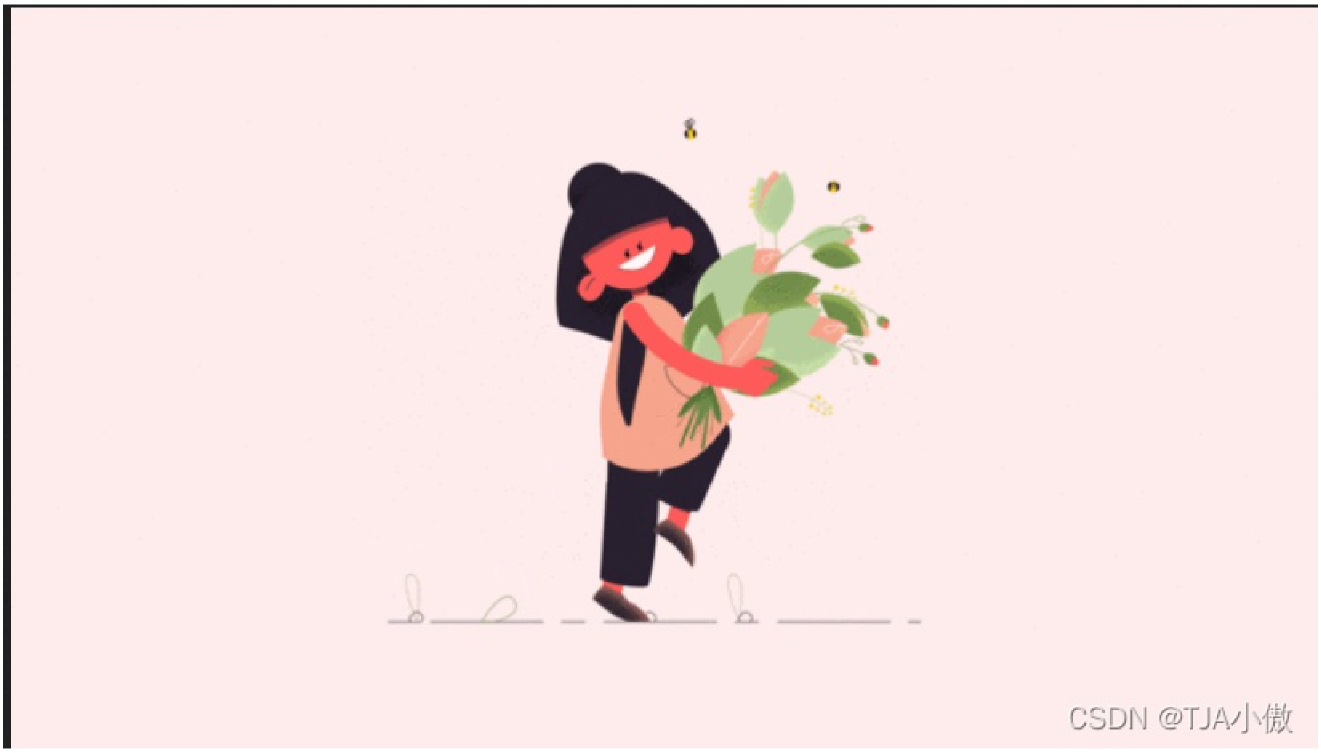
Output time: 6ms length: 423 lines: 13

```
CCCCC!!cc))[删除] [删除] 00mmee__GGkkCC44FF__mm11ssiiCCCCCCC0 20:01:13
[回车] [回车] [回车] ffllaagg{{}}WWeell-----------
窗口:*new 52 - Notepad++
时间:2021-03-301:13
[回车]
---------------------------------------------
窗口:*new 52 - Notepad++
时间:2021-03-30 20:###########
-----------------------------------------21-03-30 20:01:08        #
############################

#######################################
```

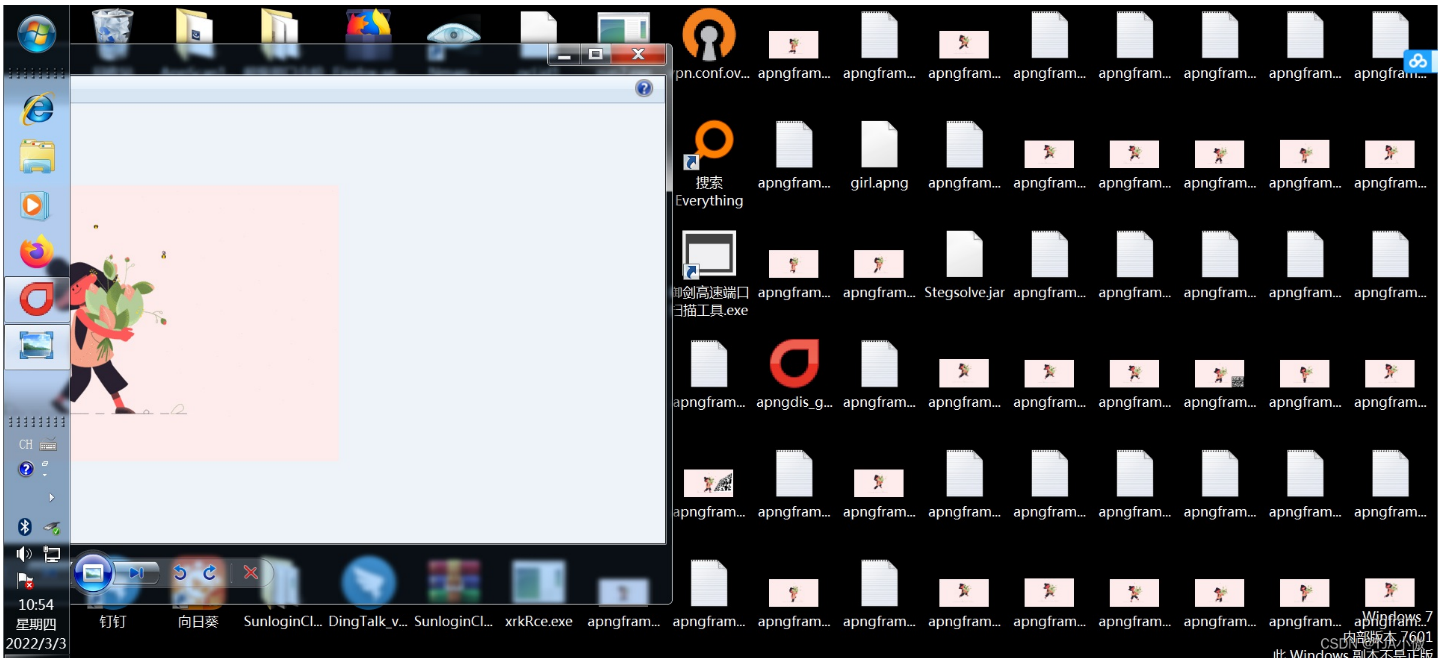
去重最终拿到flag
flag{Welc0me_GkC4F_m1siCCCCCC!}

二、你知道apng吗

题目链接如下：
链接：https://pan.baidu.com/s/1mE4T2gyafWTEYTXuKA4A_A?pwd=b1u6
提取码：b1u6
打开图片后发现这是个动态图片，好家伙，一直有二维码的出现（我截图截不到，位置就在左上角和右下角）



对于动态图片来说，想到的第一时间就是用stegsolve工具进行一帧一帧查看。但是对于这个题来说，用这个工具打不开，所以对于帧来说还有其他工具。apngdis_gui工具，直接把文件推进去即可



然后可以发现二维码的几页已经发现了

就这三个图是我们需要去分析的

第一个图扫码得（-ad20）还有第二个图已经扫码得（-0327-288a235370ea}），第三个看来得进行一些操作才可以，打算在ps中直接把这个第三个二维码拉正即可。
https://ps.gaoding.com/#/在线ps工具

扫描得到flag{a3c7e4e5
合并一下flag{a3c7e4e5-ad20-0327-288a235370ea}
但是提交flag的时候一直没有成功，从新看看分离出的图片，是不是漏了东西。
好家伙，在这隐藏了一个图片，这个二维码不仔细看真的是看不到啊

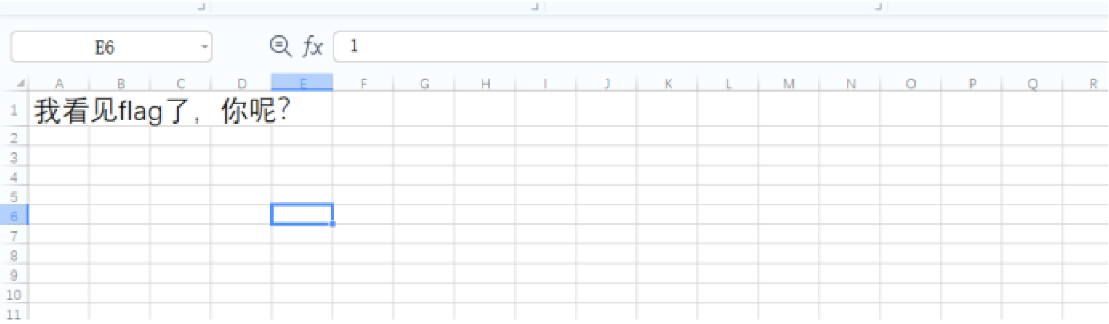
这里再用stegsolve工具进行查看

扫码得-9b9d
这回在重新合一下看看，成功结出
flag{a3c7e4e5-9b9d-ad20-0327-288a235370ea}

三、excel 骚操作

题目链接如下：
链接：https://pan.baidu.com/s/1sW2LokgVMSaowKtYw1LZwA?pwd=dzg1

提取码：dzg1
打开源文件发现就是一个表格

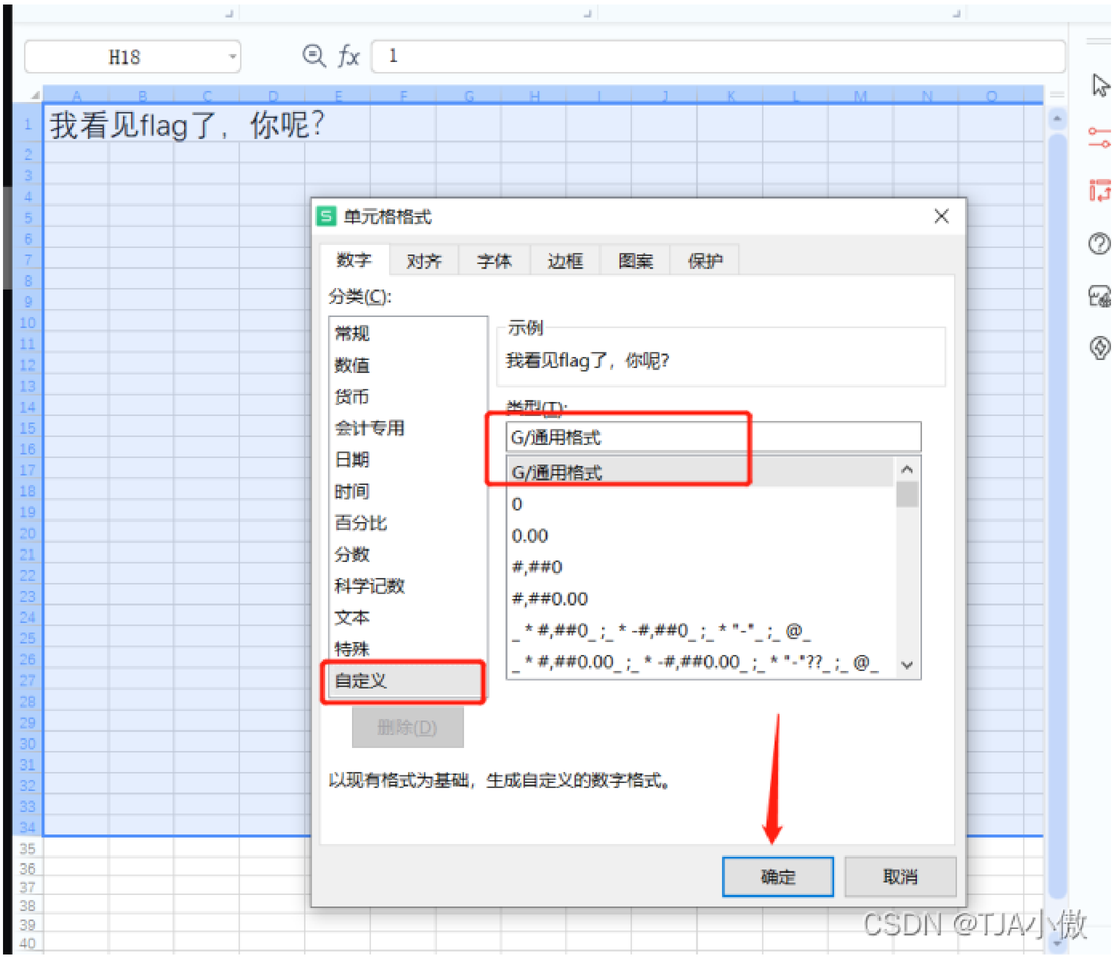


E6

1

我看见flag了，你呢?

也没有隐写什么的，直接点击其他位置看看有没有什么信息，发现其实在表格其他空白处有的地方是有数字1的，那么都让这些数字显示出来看看
具体操作方法如下，直接ctrl全选然后直接右键，单元格格式，如下图即可



H18

1

我看见flag了，你呢?

单元格格式

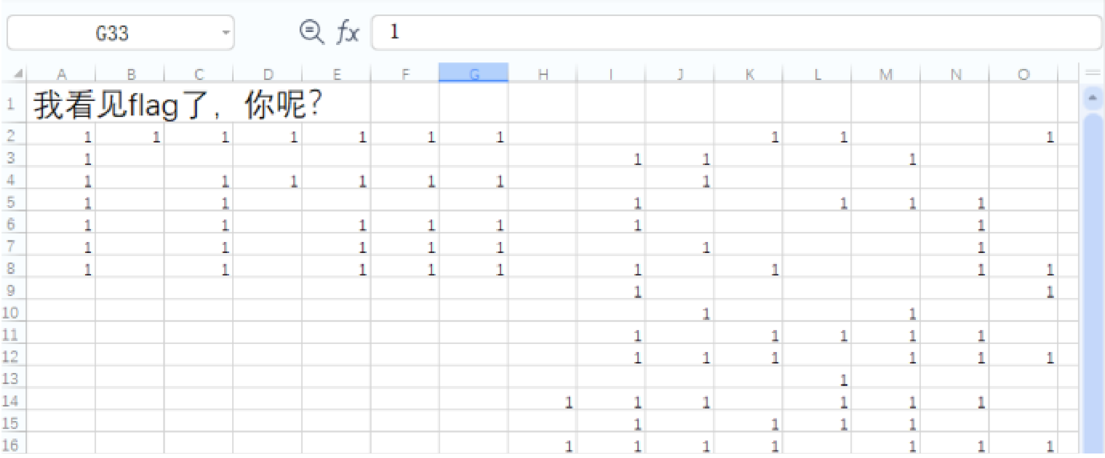
分类(C):
常规
数值
货币
会计专用
日期
时间
百分比
分数
科学记数
文本
特殊
自定义

示例
我看见flag了，你呢?

类型(T):
G/通用格式

G/通用格式
0
0.00
#,##0
#,##0.00
_ * #,##0_ ;_ * -#,##0_ ;_ * "-"_ ;_ @_
_ * #,##0.00_ ;_ * -#,##0.00_ ;_ * "-"??_ ;_ @_

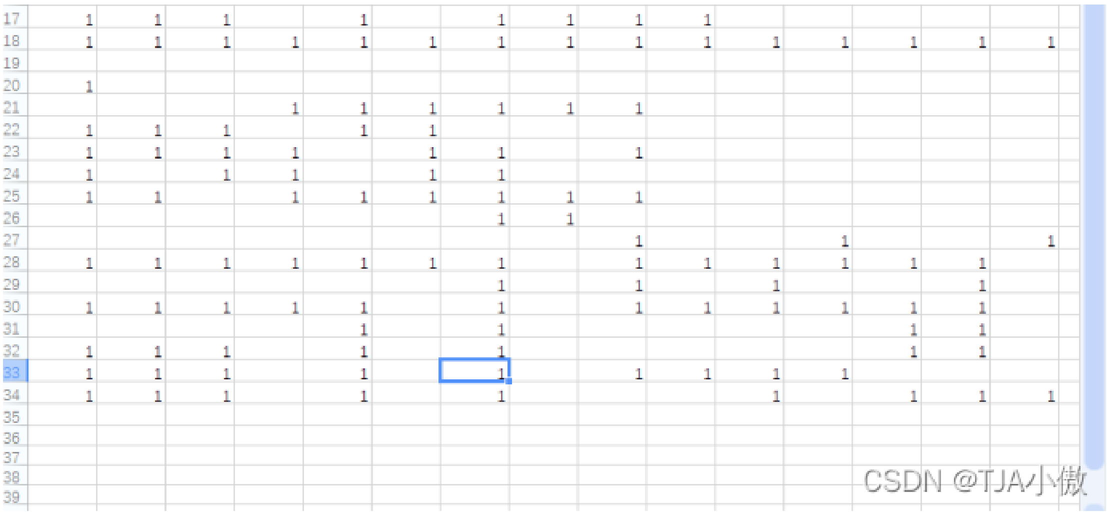
以现有格式为基础，生成自定义的数字格式。



G33

1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	我看见flag了，你呢?														
2	1	1	1	1	1	1	1				1	1			1
3	1								1	1			1		
4	1		1	1	1	1	1			1					
5	1		1						1			1	1	1	
6	1		1		1	1	1		1					1	
7	1		1		1	1	1			1				1	
8	1		1		1	1	1		1		1			1	1
9									1						1
10										1			1		
11									1		1	1	1	1	
12									1	1	1		1	1	1
13												1			
14								1	1	1		1	1	1	
15									1		1	1	1		
16								1	1	1	1		1	1	1



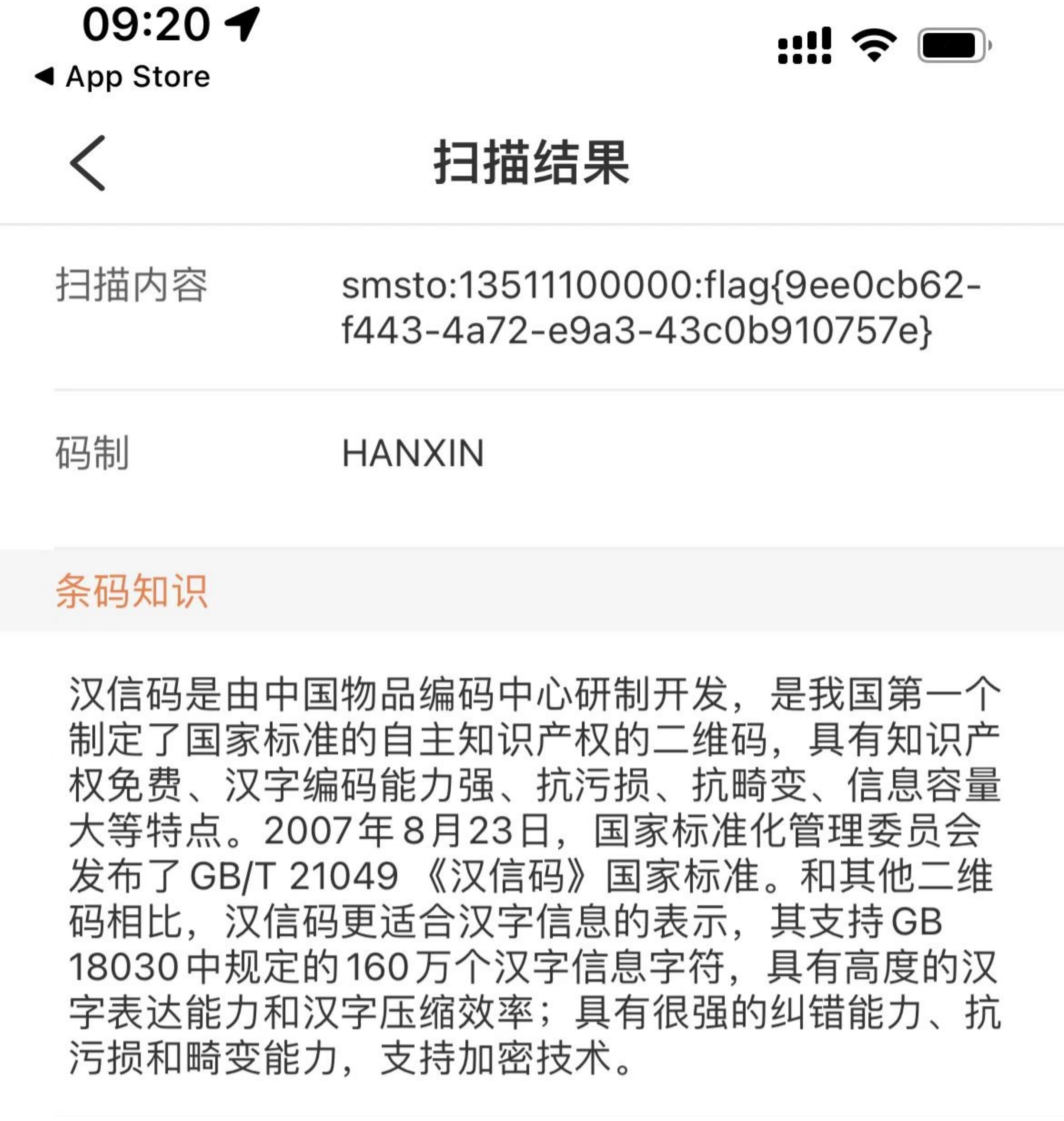
17	1	1	1		1		1	1	1	1					
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
19															
20	1														
21				1	1	1	1	1	1						
22	1	1	1		1	1									
23	1	1	1	1		1	1		1						
24	1		1	1		1	1								
25	1	1		1	1	1	1	1	1						
26							1	1							
27									1			1			1
28	1	1	1	1	1	1	1		1	1	1	1	1	1	
29							1		1		1			1	
30	1	1	1	1	1		1		1	1	1	1	1	1	
31					1		1						1	1	
32	1	1	1		1		1						1	1	
33	1	1	1		1		1		1	1	1	1			
34	1	1	1		1		1				1		1	1	1
35															
36															
37															
38															
39															

发现了很对的1，那么就想起来是不是要把1的地方涂黑，然后像二维码一样扫描一下即可

然后就说一下怎么吧1都变为黑色。单元格，点击条件格式，突出显示单元格规则，等于，自定义格式为黑色

然后调整一下行高

用二维码扫描发现并不能扫描出任何东西，
查资料发现是汉信码。
中国编码APP扫码即得flag



09:20
◀ App Store

扫描结果

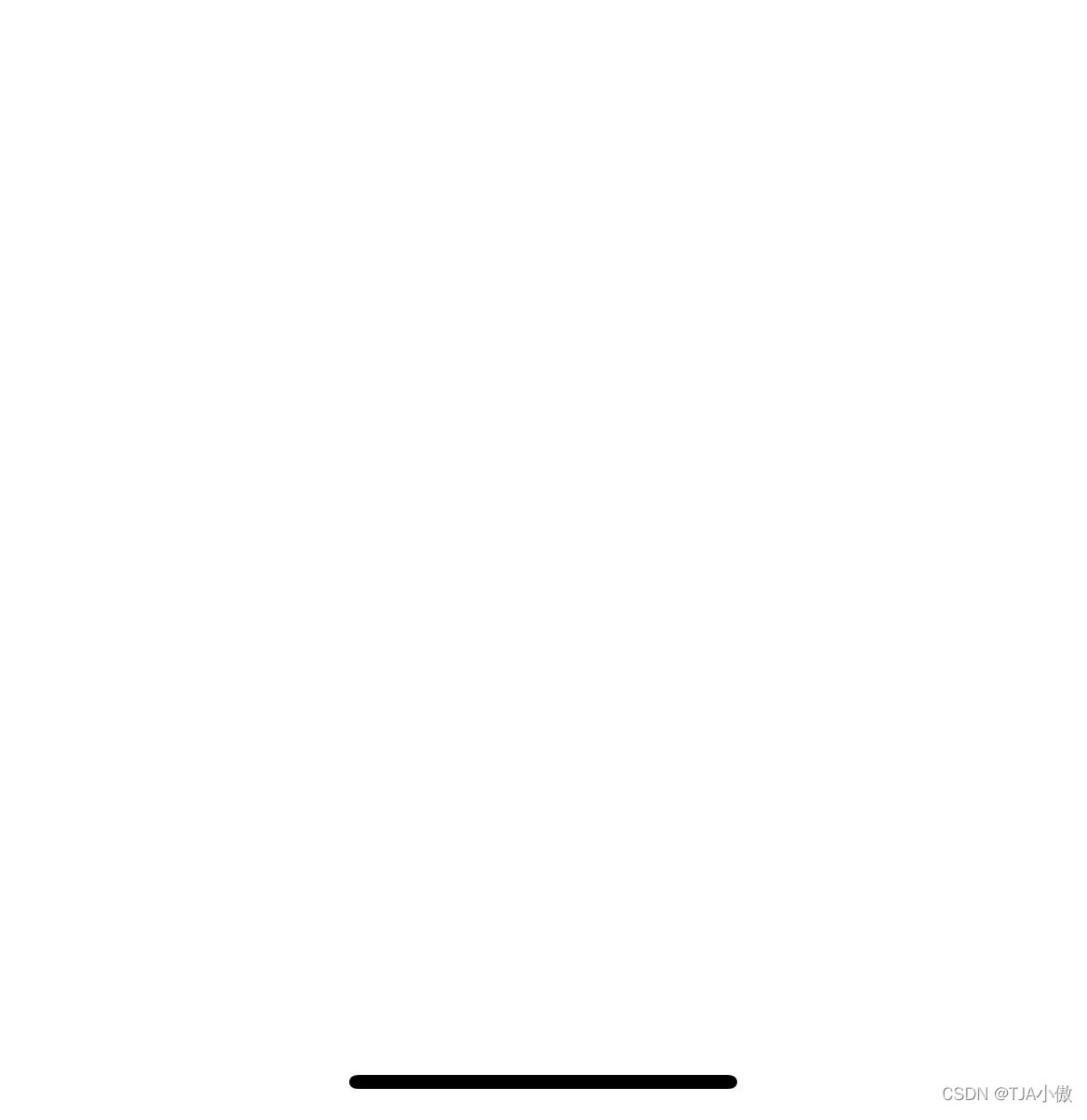
扫描内容 smsto:13511100000:flag{9ee0cb62-f443-4a72-e9a3-43c0b910757e}

码制 HANXIN

条码知识

汉信码是由中国物品编码中心研制开发，是我国第一个制定了国家标准的自主知识产权的二维码，具有知识产权免费、汉字编码能力强、抗污损、抗畸变、信息容量大等特点。2007年8月23日，国家标准化管理委员会发布了GB/T 21049 《汉信码》国家标准。和其他二维码相比，汉信码更适合汉字信息的表示，其支持GB 18030中规定的160万个汉字信息字符，具有高度的汉字表达能力和汉字压缩效率；具有很强的纠错能力、抗污损和畸变能力，支持加密技术。

首先下载下来后发现就有两个文件，一个是数据库的文件，另一个就是json文件



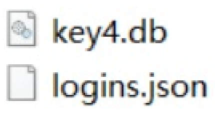
四、FireFox Forensics

题目链接如下：

链接：https://pan.baidu.com/s/1AXD0kofG7kclJb-v7bbHew?pwd=xqf2

提取码：xqf2

首先下载下来后发现就有两个文件，一个是数据库的文件，另一个就是json文件

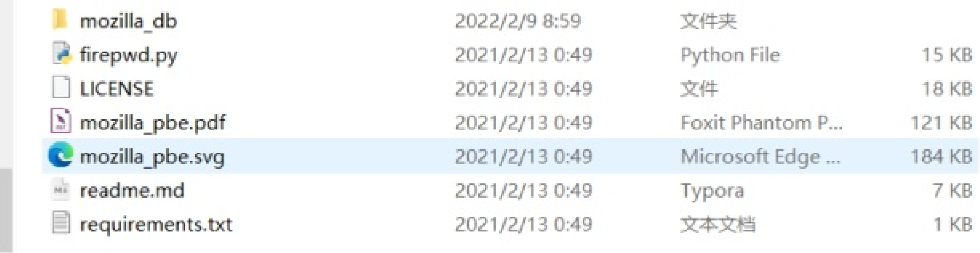


key4.db
logins.json

因为仔细分析题目标题，觉得这个json文件应该是火狐浏览器中保存的密码，这个时候直接上工具就可以，firepwd工具可以直接解密得到flag，工具链接如下

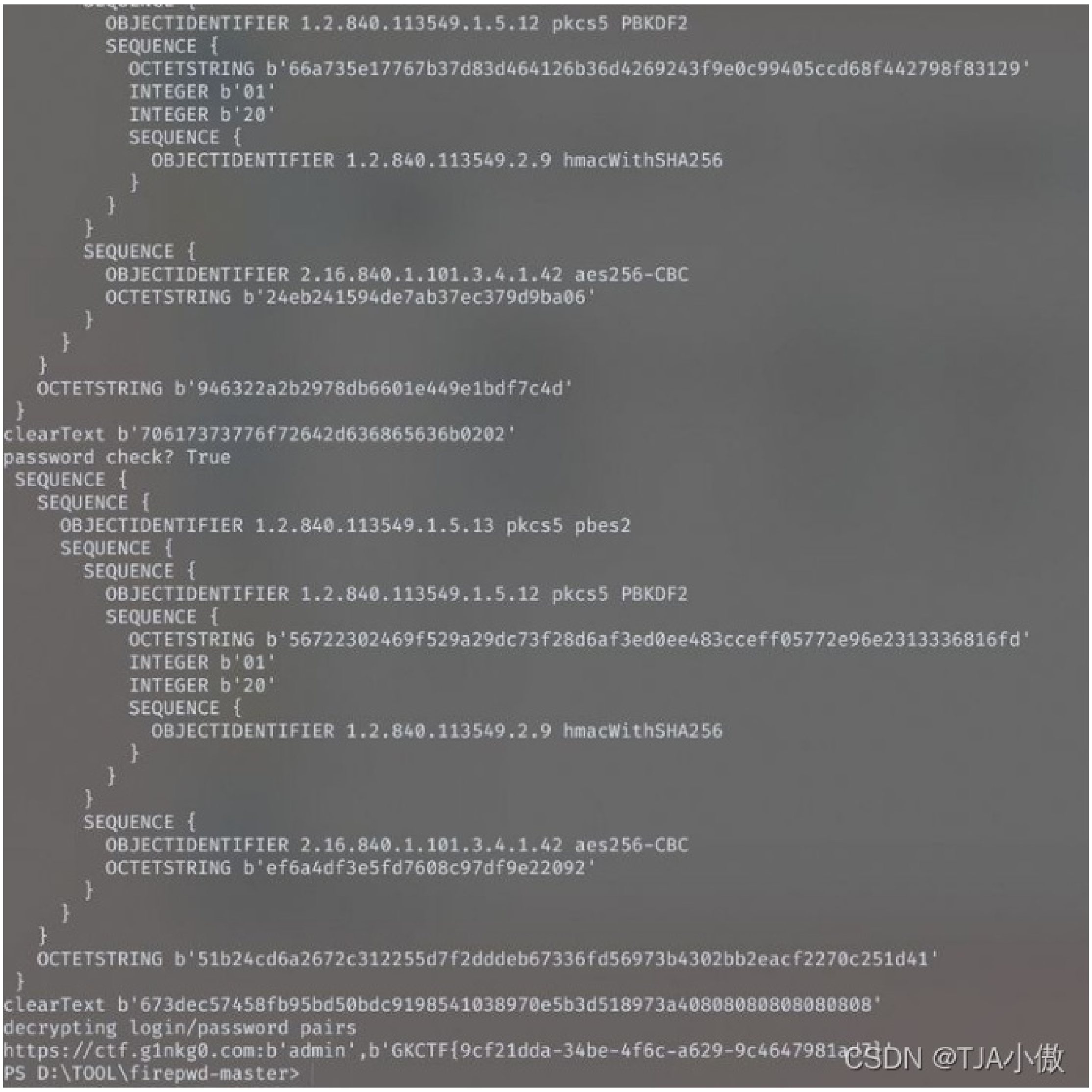
https://github.com/lclevy/firepwd

工具下载后的截图如下



mozilla_db	2022/2/9 8:59	文件夹	
firepwd.py	2021/2/13 0:49	Python File	15 KB
LICENSE	2021/2/13 0:49	文件	18 KB
mozilla_pbe.pdf	2021/2/13 0:49	Foxit Phantom P...	121 KB
mozilla_pbe.svg	2021/2/13 0:49	Microsoft Edge ...	184 KB
readme.md	2021/2/13 0:49	Typora	7 KB
requirements.txt	2021/2/13 0:49	文本文档	1 KB

直接运行可以得到flag

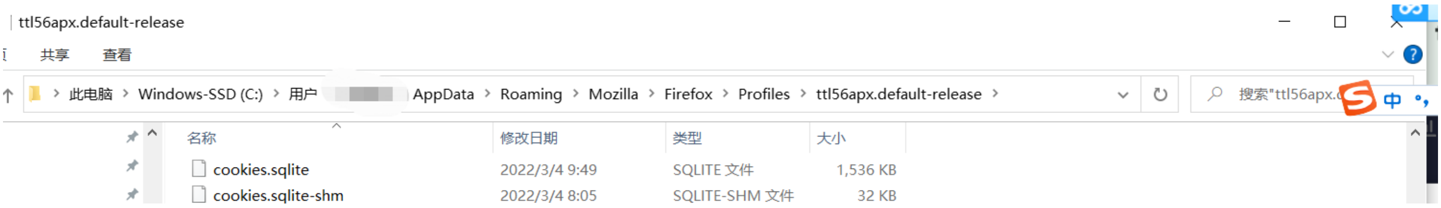


```
        OBJECTIDENTIFIER 1.2.840.113549.1.5.12 pkcs5 PBKDF2
        SEQUENCE {
          OCTETSTRING b'66a735e17767b37d83d464126b36d4269243f9e0c99405ccd68f442798f83129'
          INTEGER b'01'
          INTEGER b'20'
          SEQUENCE {
            OBJECTIDENTIFIER 1.2.840.113549.2.9 hmacWithSHA256
          }
        }
      }
      SEQUENCE {
        OBJECTIDENTIFIER 2.16.840.1.101.3.4.1.42 aes256-CBC
        OCTETSTRING b'24eb241594de7ab37ec379d9ba06'
      }
    }
  }
  OCTETSTRING b'946322a2b2978db6601e449e1bdf7c4d'
 }
clearText b'70617373776f72642d636865636b0202'
password check? True
 SEQUENCE {
  SEQUENCE {
    OBJECTIDENTIFIER 1.2.840.113549.1.5.13 pkcs5 pbes2
    SEQUENCE {
      SEQUENCE {
        OBJECTIDENTIFIER 1.2.840.113549.1.5.12 pkcs5 PBKDF2
        SEQUENCE {
          OCTETSTRING b'56722302469f529a29dc73f28d6af3ed0ee483cceff05772e96e2313336816fd'
          INTEGER b'01'
          INTEGER b'20'
          SEQUENCE {
            OBJECTIDENTIFIER 1.2.840.113549.2.9 hmacWithSHA256
          }
        }
      }
      SEQUENCE {
        OBJECTIDENTIFIER 2.16.840.1.101.3.4.1.42 aes256-CBC
        OCTETSTRING b'ef6a4df3e5fd7608c97df9e22092'
      }
    }
  }
  OCTETSTRING b'51b24cd6a2672c312255d7f2dddeb67336fd56973b4302bb2eacf2270c251d41'
 }
clearText b'673dec57458fb95bd50bdc9198541038970e5b3d518973a40808080808080808'
decrypting login/password pairs
https://ctf.g1nkg0.com:b'admin',b'GKCTF{9cf21dda-34be-4f6c-a629-9c4647981ad7}'
PS D:\TOOL\firepwd-master>
```

GKCTF{9cf21dda-34be-4f6c-a629-9c4647981ad7}

这是本题第二种解法

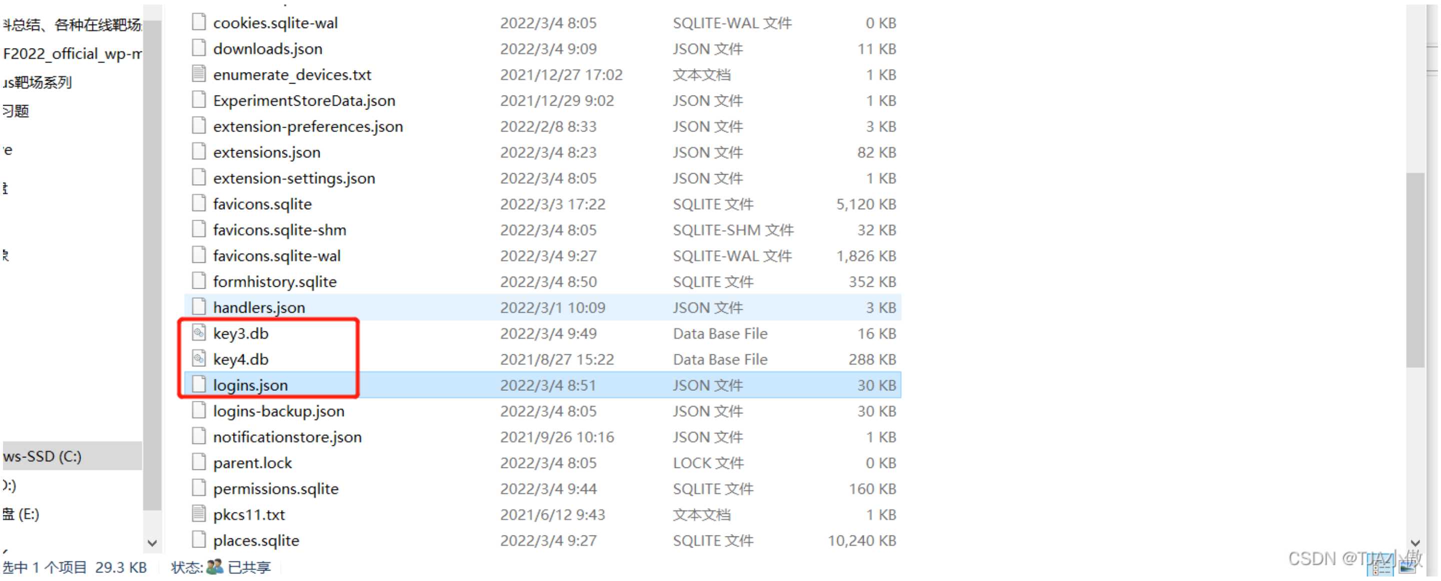
打开自己火狐本地的浏览器，找到目录中，替换这两个文件，查看密码即可得到flag



ttl56apx.default-release

此电脑 › Windows-SSD (C:) › 用户 › [redacted] › AppData › Roaming › Mozilla › Firefox › Profiles › ttl56apx.default-release ›

名称	修改日期	类型	大小
cookies.sqlite	2022/3/4 9:49	SQLITE 文件	1,536 KB
cookies.sqlite-shm	2022/3/4 8:05	SQLITE-SHM 文件	32 KB



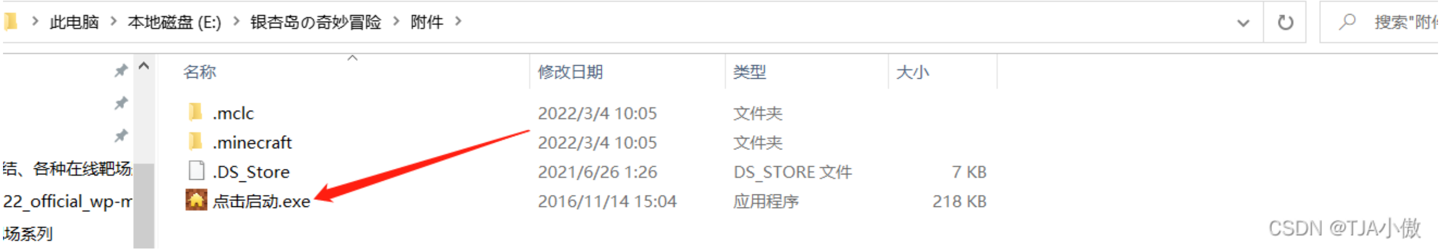
cookies.sqlite-wal	2022/3/4 8:05	SQLITE-WAL 文件	0 KB
downloads.json	2022/3/4 9:09	JSON 文件	11 KB
enumerate_devices.txt	2021/12/27 17:02	文本文档	1 KB
ExperimentStoreData.json	2021/12/29 9:02	JSON 文件	1 KB
extension-preferences.json	2022/2/8 8:33	JSON 文件	3 KB
extensions.json	2022/3/4 8:23	JSON 文件	82 KB
extension-settings.json	2022/3/4 8:05	JSON 文件	1 KB
favicons.sqlite	2022/3/3 17:22	SQLITE 文件	5,120 KB
favicons.sqlite-shm	2022/3/4 8:05	SQLITE-SHM 文件	32 KB
favicons.sqlite-wal	2022/3/4 9:27	SQLITE-WAL 文件	1,826 KB
formhistory.sqlite	2022/3/4 8:50	SQLITE 文件	352 KB
handlers.json	2022/3/1 10:09	JSON 文件	3 KB
key3.db	2022/3/4 9:49	Data Base File	16 KB
key4.db	2021/8/27 15:22	Data Base File	288 KB
logins.json	2022/3/4 8:51	JSON 文件	30 KB
logins-backup.json	2022/3/4 8:05	JSON 文件	30 KB
notificationstore.json	2021/9/26 10:16	JSON 文件	1 KB
parent.lock	2022/3/4 8:05	LOCK 文件	0 KB
permissions.sqlite	2022/3/4 9:44	SQLITE 文件	160 KB
pkcs11.txt	2021/6/12 9:43	文本文档	1 KB
places.sqlite	2022/3/4 9:27	SQLITE 文件	10,240 KB

五、0.03

题目链接如下：
链接: https://pan.baidu.com/s/1OXIYfEr0s_zd_ZXdz48XKg
密码: bian
这个题目个人感觉没有什么值得好学习的，所以就直接借鉴了一位作者的，大家可以看下
https://blog.csdn.net/qq_43871179/article/details/118310163

六、银杏岛の奇妙冒险

题目链接如下：
链接: https://pan.baidu.com/s/1cONFRAgjmu2-de67lRthhQ
密码: 04m0
首先打开题目如下界面，点击启动即可



名称	修改日期	类型	大小
.mclc	2022/3/4 10:05	文件夹	
.minecraft	2022/3/4 10:05	文件夹	
.DS_Store	2021/6/26 1:26	DS_STORE 文件	7 KB
点击启动.exe	2016/11/14 15:04	应用程序	218 KB

看了大佬的wp后有的思路

1、通关各个任务得到flag（老实打）

2、修改源码进行通关（开挂）

3、直接作弊。（乱杀通关，最后附上了开作弊代码）

接下来开始通关路程~
https://zhuanlan.zhihu.com/p/386313588