<https://www.freesion.com/article/77601403253/#CTFmisc_25>

**misc做题步骤**

**各类文件头信息**

JPEG(jpg) FFD8FF

pyc(反编译) 03F30D0A

PNG(png) 89504E47

GIF(gif) 47494638

Windows Bitmap(bmp) 424D

XML(xml) 3C3F786D6C

HTML(html) 68746D6C3E

Email(eml) CFAD12FEC5FD746F

7z的压缩包文件头为37 7A BC AF 27 1C

WEBP：52 49 46 46 ?? ?? ?? ?? 57 45 42 50

tif: 49 49 2A 00

优秀博主链接：[ctf misc 图片题知识点\_z.volcano的博客-CSDN博客\_ctf图片题](https://blog.csdn.net/weixin_45696568/article/details/116082336)

**工具**

**010工具**

可以改bmp,jpg,png,gif的高宽

**png同时计算宽和高**

[CRC 爆破 png 图片宽度和高度原理以及 python 代码\_png crc 计算 - CSDN 博客](https://blog.csdn.net/qq_47875210/article/details/126171502)

影响上图的 CRC 的有 49 48 44 52 00 00 01 FF 00 00 01 FF 08 06 00 00 00

49 48 44 52 ASCII 码为 IHDR，表明数据块为 IHDR

00 00 01 FF 图片的宽

00 00 01 FF 图片的高

08 06 00 00 00 五个字节依次为: Bit depth、ColorType、 Compression method、 Filter method、Interlace method

      **一般不会考虑最后 5 个字节的变化，所以可以根据 CRC 来爆破图片的宽度和高度。因为除了图片宽度和图片高度，再加上忽略的最后 5 字节，其他的都是固定的，所以只有图片宽度和高度会影响 CRC 数值。**

**脚本在编程语言文件夹里面。**

gif的高要改两个地方可以直接搜索宽就可以定位到高（把搜索改成搜索值）

**bmp计算宽**

bmp的文件格式

这张图片由900\*150个像素，文件头占用54字节，文件尾在0x76F50位置，换算成10进制为487248

因为每个像素点由3个字节（十六进制的6位）表示，每个字节负责控制一种颜色，分别为蓝，绿，红，所以文件真是像素个数为

(487248-54)/3=162398



题目提示图片高度是正确的，所以说要计算宽度、

bmp : 像素等于：文件尾总字节-文件头总字节/3字节

宽度即为像素数除以高度：192398/150=1082（如果有余数要舍去）

将图片宽度改为1082保存图片打开即为flag

公式 **size-offbits/3/height 求高**

**Crunch**

**@：** 代表小写字母

**,：** 代表大小字母

**^：** 代表特殊字符

**%：** 代表0-9

Crunch 生成字典 提示共八位前四为DBOQ后四位为数字

Crunch 8 8 -t DBOQ%%%%>> zippasswd.txt

**fcrackzip**

用fcrackzip进行爆破

Fcrackzip -D -p zippasswd.txt -u flag.zip

或者使用zip2john flag.zip > pass.txt

john --wordlist=pass.zippasswd.txt pass.txt 爆破

TweakPNG。

功能简述

删除不需要或不想要的块，通常以减小文件大小。

注释的图像文件添加或修改文本块。

更改背景颜色和透明度设置。

为了使测试用于测试应用程序读取PNG文件的图像。

如果有 30 个 IDAT 块，从第一个开始一个一个删除，删到第八个的时候，图片显示 flag

**steghide**

隐藏文件

steghide embed -cf [图片文件载体] -ef [待隐藏文件] 2. [回车] 3. 输入密码，提取文件时用到，如果不想设密码，直接按回车 4. [ENTER]

hide

提取文件

steghide extract -sf background.jpg -a 密码

当找不到任何方法后可以用010看看有没有关键字（要么在前面要么在后面）

软盘可以放到虚拟机打开

**convert工具**

convert glance.gif flag.png #将glance.gif分割成多个flag.png

montagee工具

montage flag\*.png -tile x1 -geometry +0+0 flag.png #合并图片

**zsteg命令**

安装命令：gem install zsteg

zsteg shan.bmp -a 尝试所有以知组合（最有用的命令 -a）

提取照片

zsteg -e b1,rgb,lsb,xy 1.png>99.png

**tshark**

tshark -r data.pcap -T fields -e usb.capdata 分离usb流量

**convert**

**如果照片被转了方向可以用Linux的convert**

//翻转

//上下翻转：

convert -flip foo.png bar.png

//左右翻转：

convert -flop foo.png bar.png

**strings**

**可以查找出字符串**

strings xxxx.pcapng | grep -i "flag"

outguess

**outguess算法 提取1.jpg的隐藏flag**

**outguess -k 123456 -r 1.jpg -t sss.txt #-k 提取密码，猜一下是123456**

outguess -r 1.jpg -t hidden.txt

**john**

**压缩包破解**

zip.rar

zip2john/rar2john xxxx.zip/rar>> pass.txt

john pass.txt

**提取分离文件中隐藏文件**

binwalk -e xxx.jpg

foremost -T xxx.jpg

**dd**

**dd分离 （有些不能分离的他可以）**

【2】binwalk分析：

显示里面有一个ZIP压缩包

【3】使用dd命令分离该压缩包：

dd命令格式：

dd if=图片名 of=ZIP压缩包的名称（自己命名） skip=偏移量（本题是68019） bs=1

JPHS for windows

功能：

对有损压缩JPEG文件进行信息的加密隐藏、探测提取的工具

JPHS包含2个功能（靠2个程序实现）：

JPHIDE：能够将信息文件加密隐藏到JPEG图像

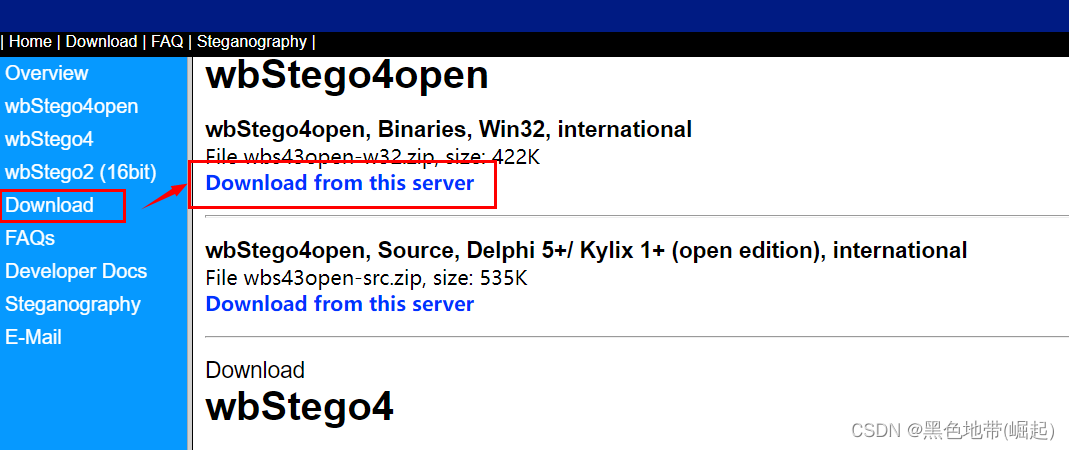
JPSEEK：从用JPHIDE程序加密隐藏得到的JPEG图像中探测并提取到信息文件

wbStego4open

wbStego4open  (<http://wbstego.wbailer.com)>我windows下的这个

这个是使用网站[【PDF隐写】wbStego4open，且还包括BMP、 TXT、 HTM文件隐写\_黑色地带(崛起)的博客-CSDN博客](https://blog.csdn.net/qq_53079406/article/details/123783006)

用多了就熟练了

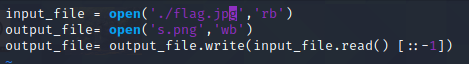


**杂**

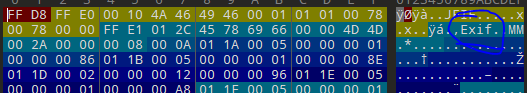
**python切片逆转**

s=123456

python(s[::-1])



文件头有这样的字符就说明有exif信息



**exif信息查看网站**

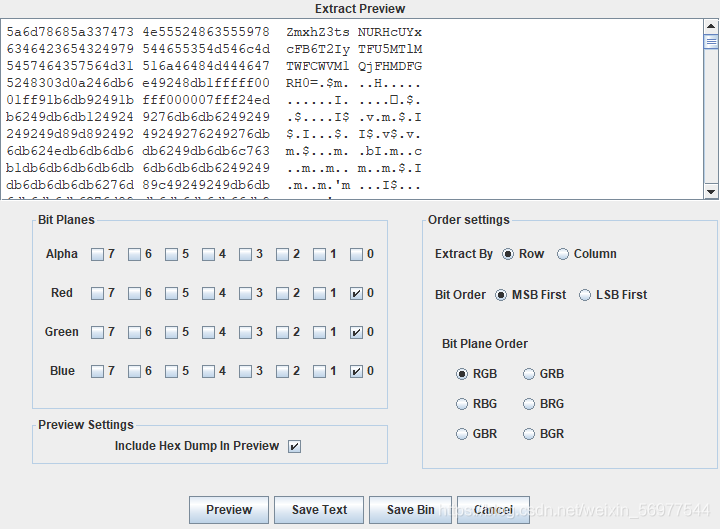
1、windows 下隐藏：copy /b 123.jpg +123.rar 1234.jpg将 123.rar 隐藏到 123.jpg 中，生成的新文件为 1234.jpg，修改后缀名还原，注意图片文件和压缩文件的顺序不能错，图片在前，否则图片不能正确显示。2、linux 下隐藏：cat 123.zip >> 123.jpg将 123.zip 文件隐藏到 123.jpg 中，重定向到 123.jpg 中，修改后缀名还原

**stegsolve.java**

**image combiner 这个选项如果是两个一样的照片就可以重叠成新照片，如果是后缀一样就没事，但是如果是一个png一个jpg那要先打开png然后重叠jpg不然不能成功**

数据解析不一定要所以颜色

当检查各个[通道](https://so.csdn.net/so/search?q=%E9%80%9A%E9%81%93&spm=1001.2101.3001.7020)均未发现异常，常看隐藏东西比较多的最低有效位



修改magic number

file xxxx 可以查看文件属性，获得文件类型

看相似的文件头改过来

在Alpha0、B0、G0通道均发现lsb隐写痕迹，而且和misc53不同的是，这里是竖向排列的，即按列(column) 勾选上对应选项

**py文件反编译**

首先在windows安装 pip install uncompyle

uncompyle6 xxx.pyc > py.py

**碰到ascii值被掉换位置**

shan = "复制保存的ascii码值"

print(''.join(reversed(shan)))

**如果在需要的值被空格打满就用，当然可以举一反三如果是"/"那就在" "改成"/".或者其他**

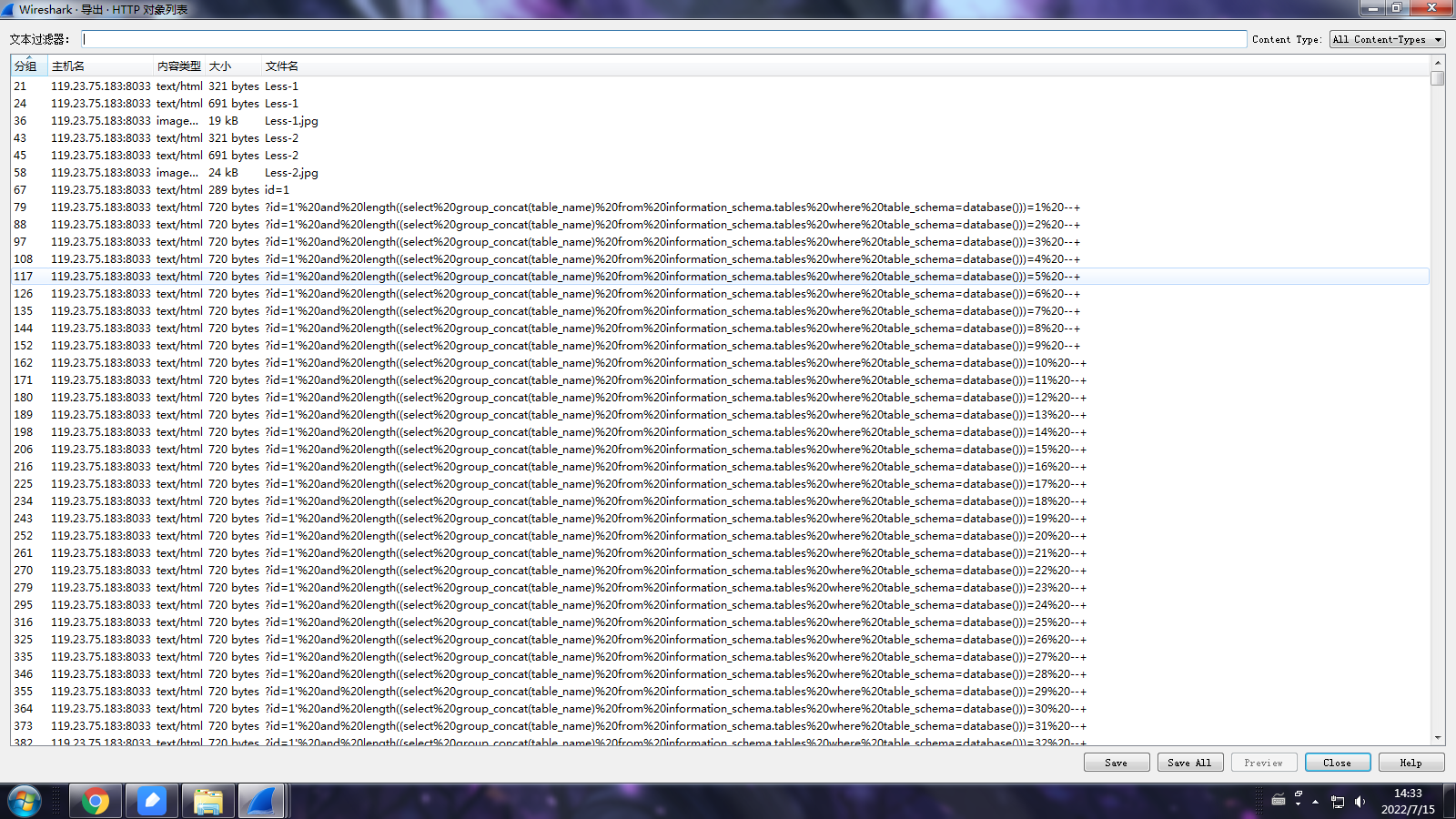
str = " Hello world "

print(str.replace(" ",""))

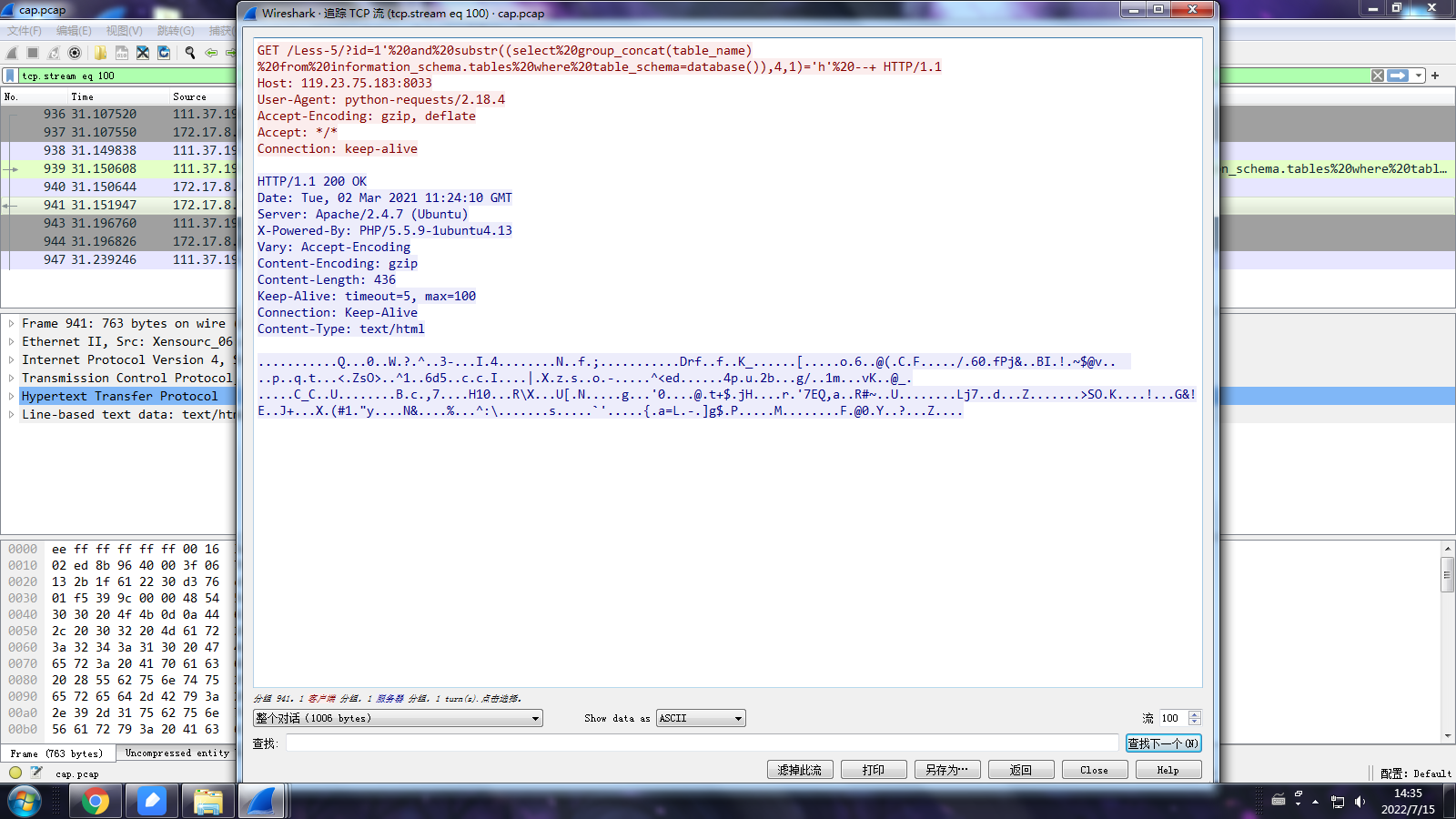
输出：

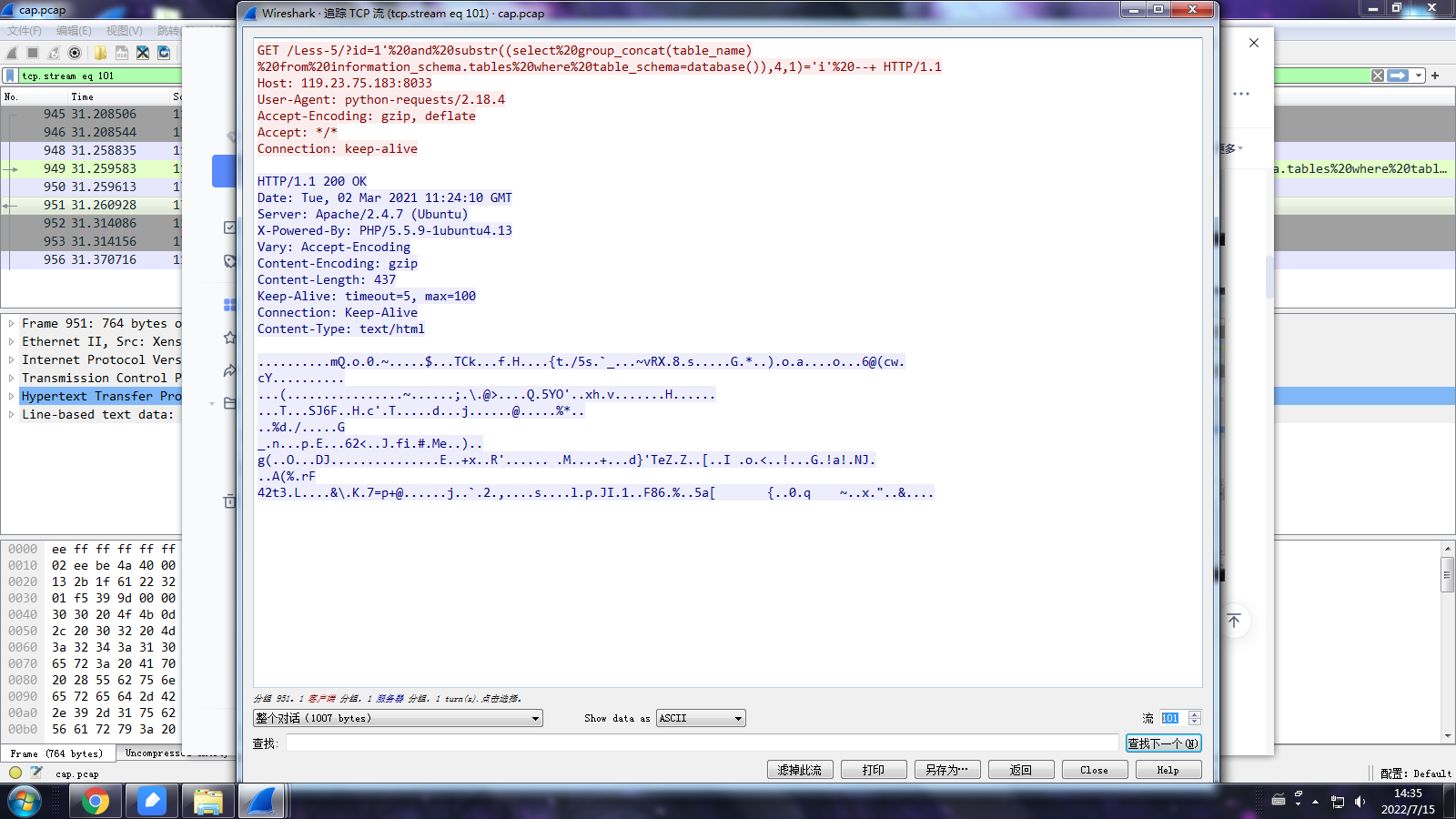
"Helloworld"

pcap文件内容是sql手，盲注



首先导出到文件夹然后调大小（如下因为正确与错误是大小不一样的）





**pcap文件提权usb首先用命令将它分离**

tshark -r data.pcap -T fields -e usb.capdata

然后用py脚本跑

**hexedito**

rhexeditorxxxxxx 可以代替Windows的WinHex改16进制

**伪加密**

● 无加密：压缩源文件数据区的全局加密应当为 00 00 （504B0304 两个 bytes 之后） 且压缩源文件目录区的全局方式位标记应当为 00 00（504B0304 四个 bytes 之后）

● 假加密：压缩源文件数据区的全局加密应当为 00 00 且压缩源文件目录区的全局方式位标记应当为09 00

● 真加密：压缩源文件数据区的全局加密应当为 09 00 且压缩源文件目录区的全局方式位标记应当为09 00。

**一般是09**

若把未加密的 zip 压缩源文件目录区的全局方式位标记改为 01 00 (或者 09 00),就会被压缩软件认为是已加密,即所谓的伪加密

压缩源文件数据区的全局加密应当为00 00  （504B0304两个bytes之后）

且压缩源文件目录区的全局方式位标记应当为00 00（504B0102四个bytes之后）

**私钥使用方法**

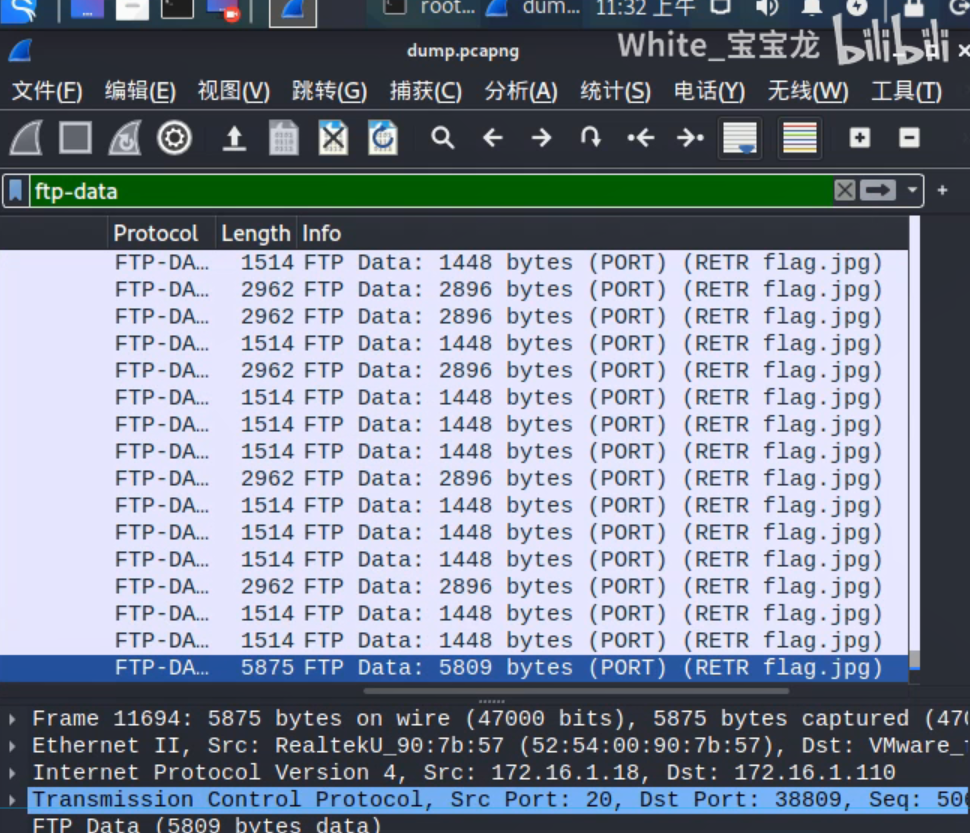
openssl rsautl -decrypt -in key.txt -inkey flag.key -out flag.txt

-in 为要解密的加密文档 -inkey 为密钥 -out 为输出文档

**数据包分析**

**查看ftp下载了什么就用**

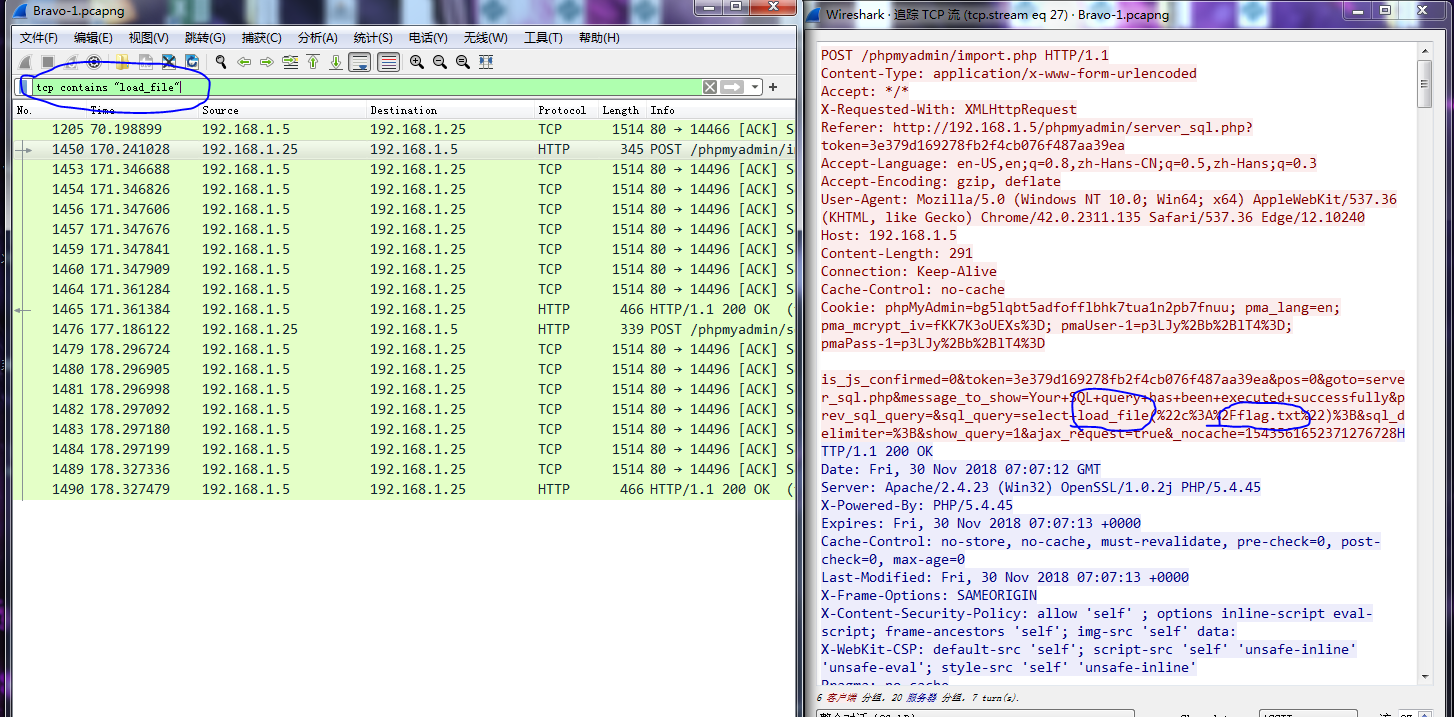
**ftp-data**

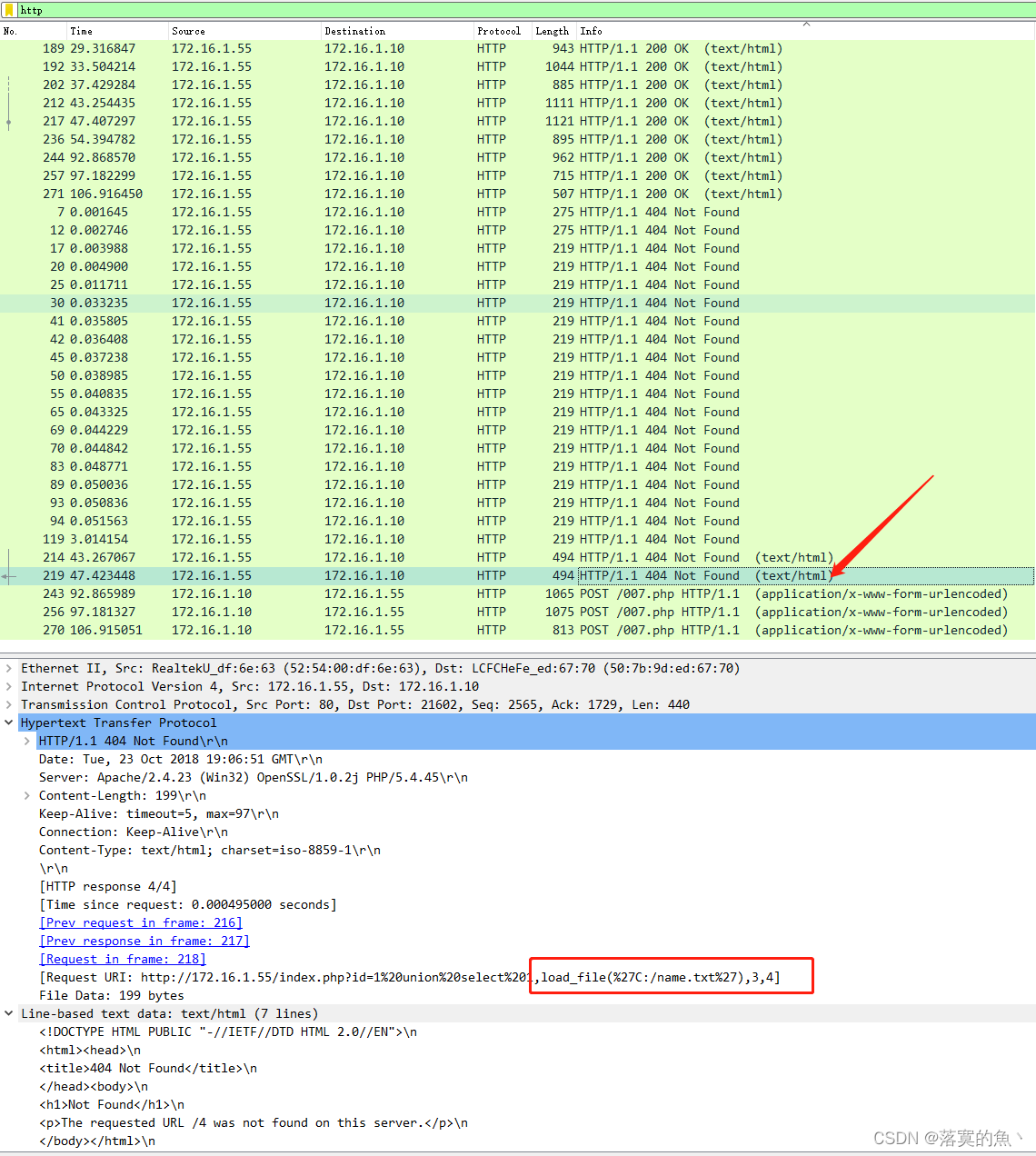


3.继续查看数据包文件分析出恶意用户读取服务器的文件名是什么，并将该文件名作为FLAG（形式：[robots.txt]）提交；

(一般是load\_file)

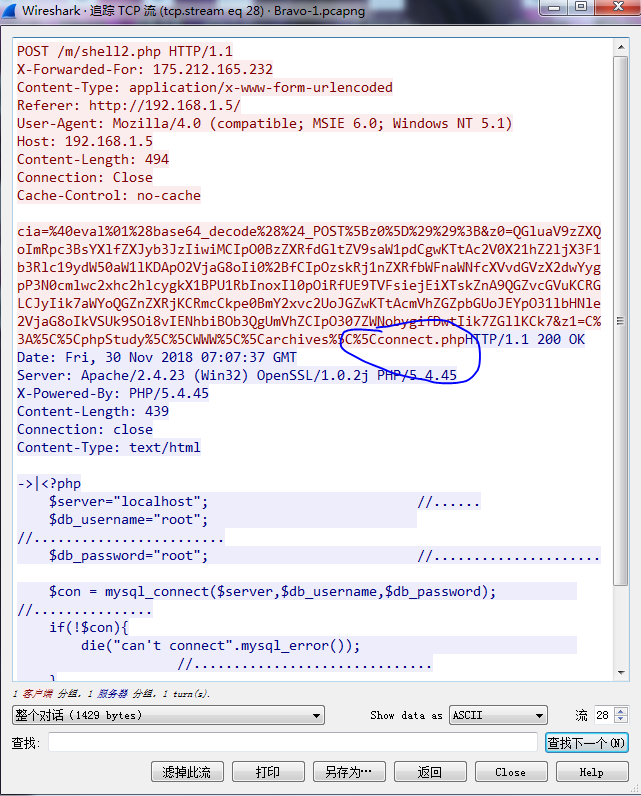
tcp contains "load\_file"





7.继续查看数据包文件将恶意用户下载的文件里面的内容作为Flag值（形式：[文件内容]）提交。

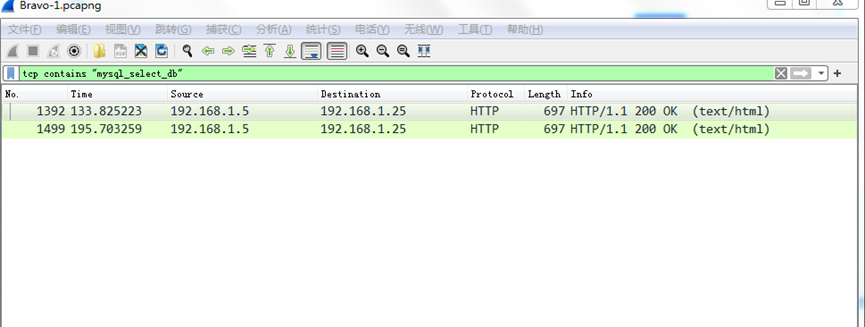
（一般是跟在一句话木马的密码后面）

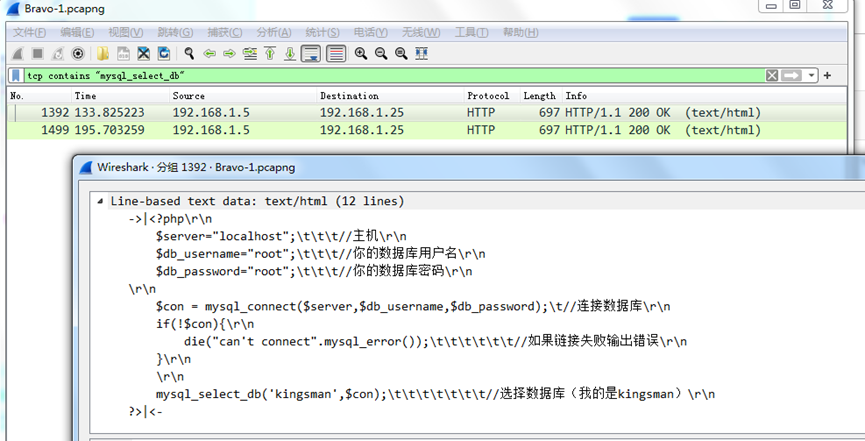




7.继续查看数据包文件Bravo-1.pcapng将Web服务器连接的数据库名作为Flag值提交。

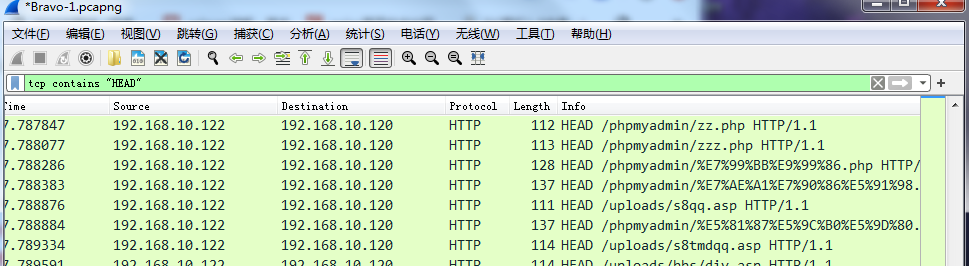
Tcp contains “mysql\_elect\_db”





Flag: kingsman

1.分析Server8桌面下的Bravo-1.pcapng数据包文件，通过分析数据包Bravo-1.pcapng找出恶意用户目录扫描的第2个目录名，并将该目录名作为Flag值提交;

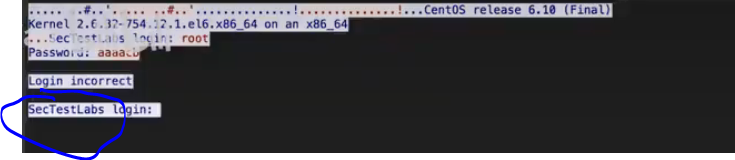


分析扫描流量可以看到，扫目录的请求头一般为HEAD，所以

flag{/uploads}

**telnet找到登录的主机名字**

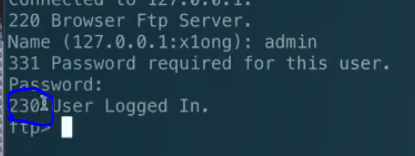
**telnet登录时候会回显**



ftp找到黑客登录的账号密码（root,toor）(ftp登录成功都会返回230状态)

过滤规则

ftp|telnet|http contains "230" 或 ftp contains "success"



黑客执行命令LIST，一般是dir，ls

分析A.pcapng数据包，找到黑客的一句话木马的密码，将该密码作为flag提交：

在显示过滤器中输入:ip.addr==192.168.10.12and http contains “@eval”

或有可能是 ip.addr==192.168.10.12and http contains “$\_GET”

Flag{oo}

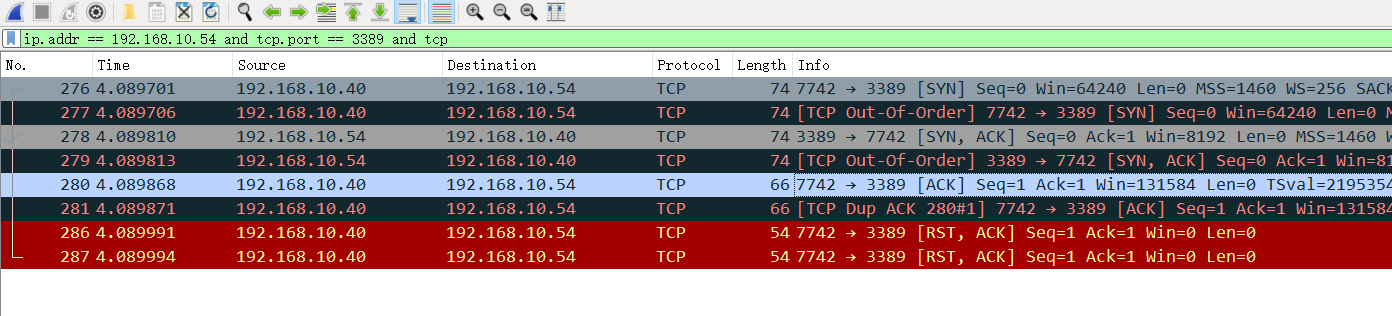
通过分析数据包，找到服务器安装的第一个修补程序

http contains “KB”

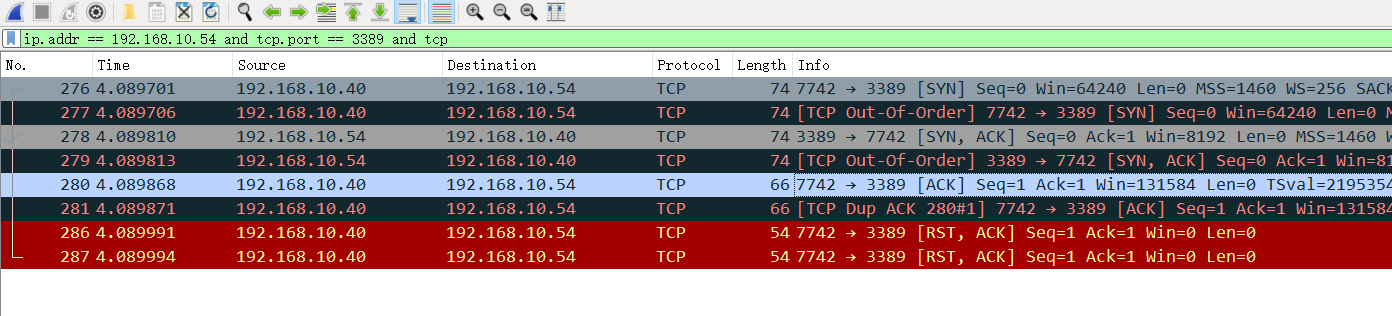
找到黑客扫描的IP地址网段，从上一题可以得到黑客的IP地址为192.168.10.12.看一下arp协议的数据

如下图若端口是打开的，返回了SYN,ACK则表明端口开放，然后扫描主机箱目标机发送ACK/RST包断开连接。

分析黑客nmap扫过的端口



如果端口是关闭的，则会直接返回RST/ACK报文断开连接，说明端口是关闭的。



该种方法进行扫描会留下扫描纪录，而且速度较慢

先找靶机IP

ip.src==靶机ip 一般到最下面它会有个总结 开启颜色放便一些

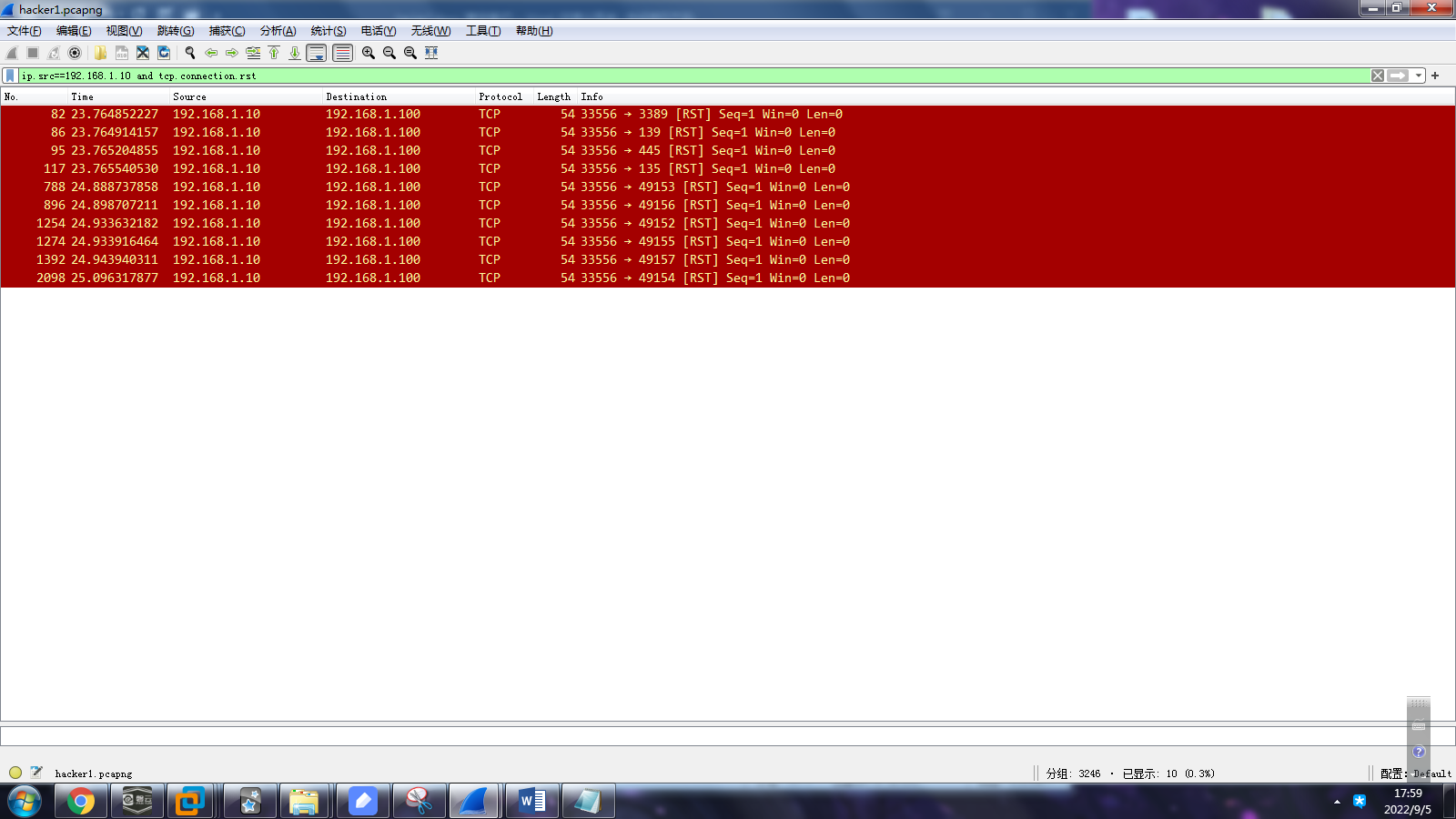
ip.src==攻击机ip and tcp.flags.reset==0

可以用下面这个命令测试如果开了端口就会返回【 SYN,ACK】

ip.addr==172.16.123.115 and tcp.port == 88

还有这种方法可以查看开放端口虽然是红的

ip.src==黑客ip and tcp.connection.rst



使用Wireshark查看并分析PYsystem20191桌面下的capturepcap数据包文件，通过设置过滤规则，要求只显示三次握手协议过程中的RST包以及实施攻击的源IP地址，将该过滤规则作为Flag值（存在两个过滤规则时，使用and连接，提交的答案中不包含空格，例d如tcp.ack and ip.dst == 1716.1.1

则Flag为tcp.ackandip.dst==172.16.1.1）提交；

过滤参数是：tcp.connection.rst and ip.src==172.16.123.112

flag:tcp.connection.rstandip.src==172.16.123.112

过滤为POST

http.request.method == POST

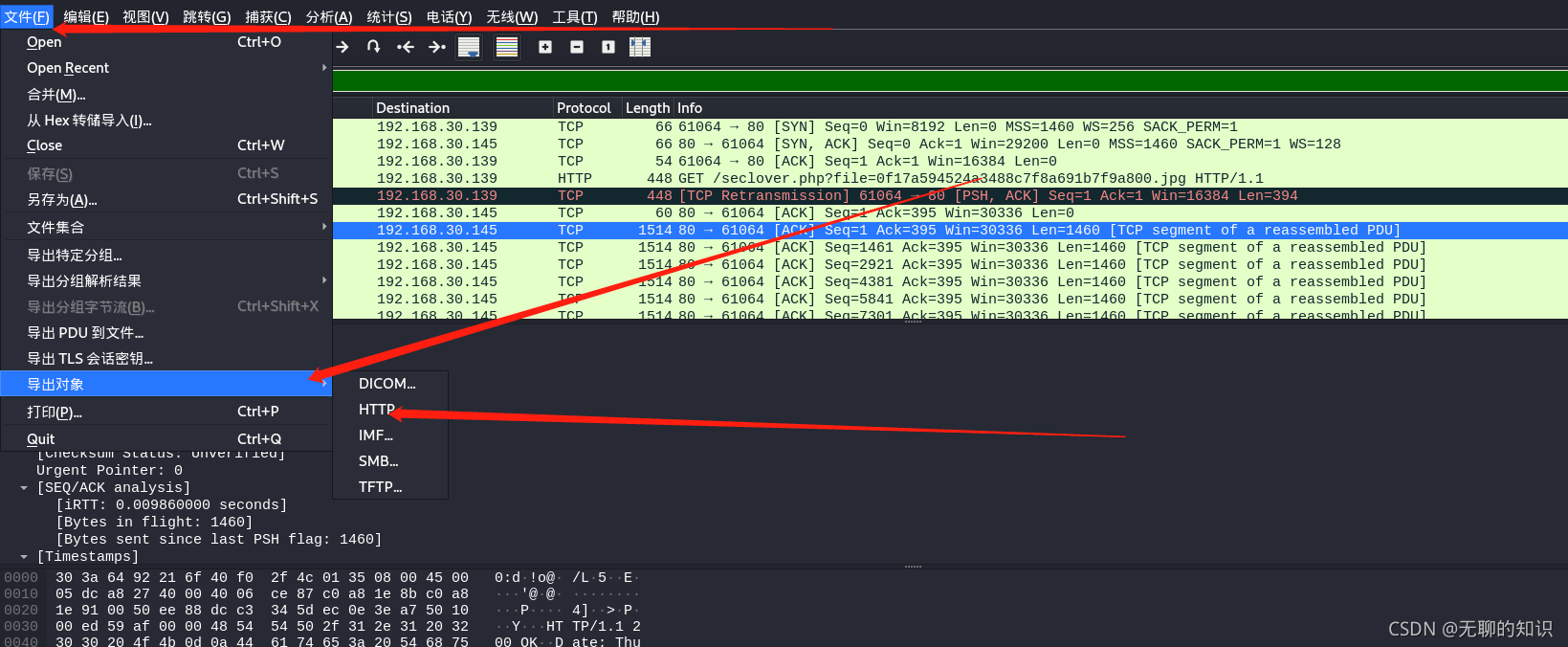
**tcpxtract 命令从pcap提取照片**

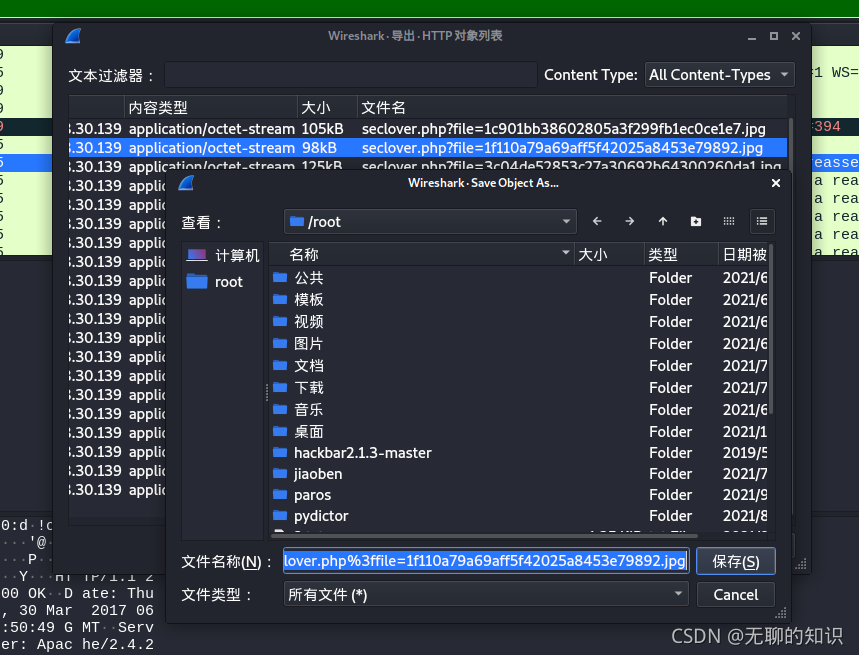
tcpxtract -f 434c8c0ba659476caa9635b97f95600c.pcap

**当tcpxtract或foremost提照片有问题就用这个办法**

打开菜单栏中的导出文件

选择导出的位置



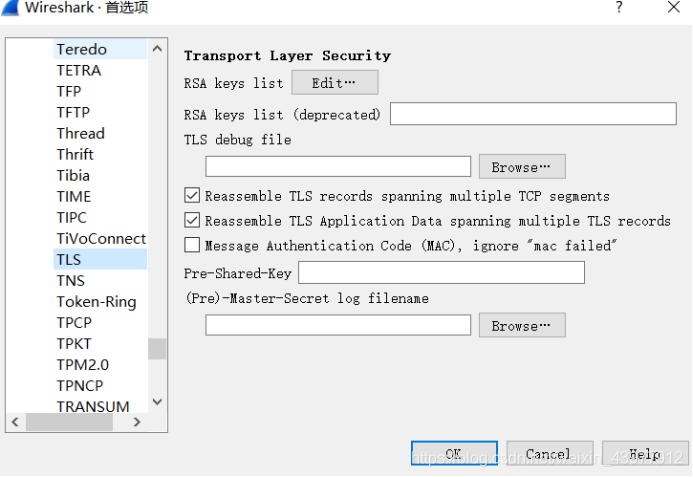


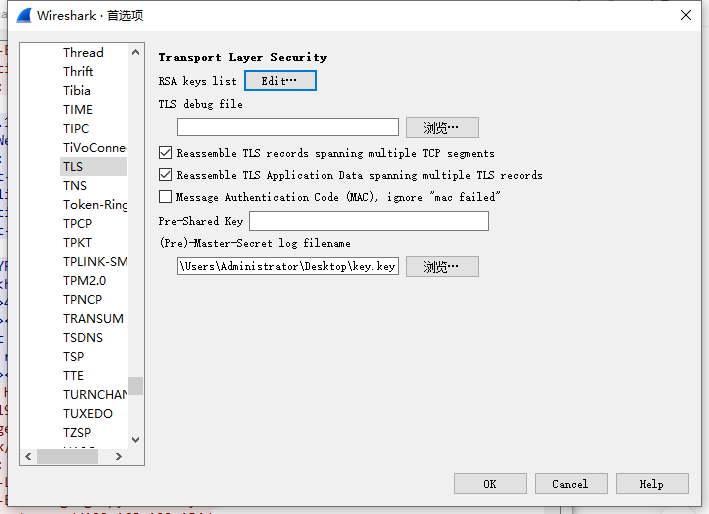
最近需要解析HTTPS流量，所以对[wireshark](https://so.csdn.net/so/search?q=wireshark&spm=1001.2101.3001.7020)的HTTPS解密进行了实测。

使用wireshark解密https的方法

方法一：

1、在wireshark的首选项中的protocols的tls选项里添加服务器[私钥](https://so.csdn.net/so/search?q=%E7%A7%81%E9%92%A5&spm=1001.2101.3001.7020)文件。p12文件需要填写密码。





**git的相关指令操作**

git log查看历史记录

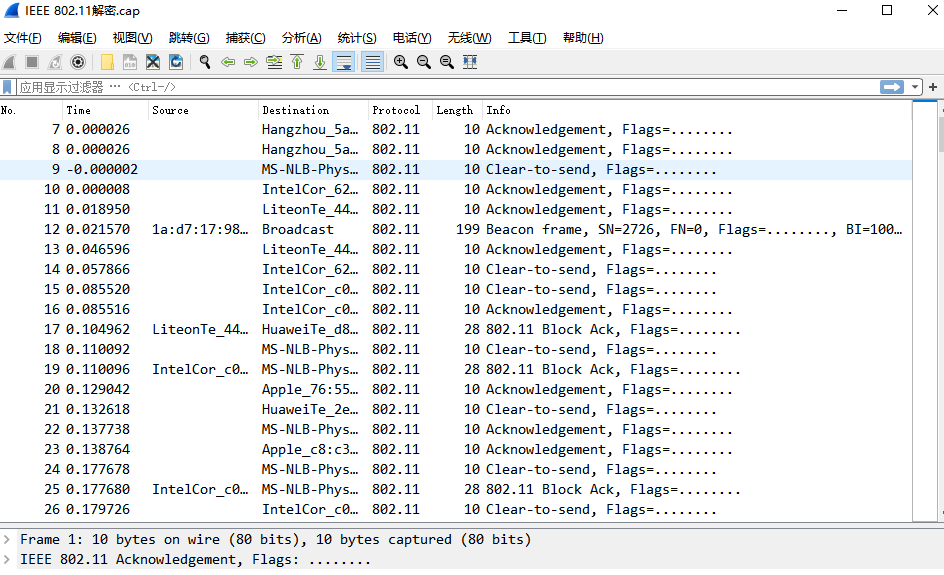
git stash list 查看隐藏的内容

git stash show 查看修改了什么文件

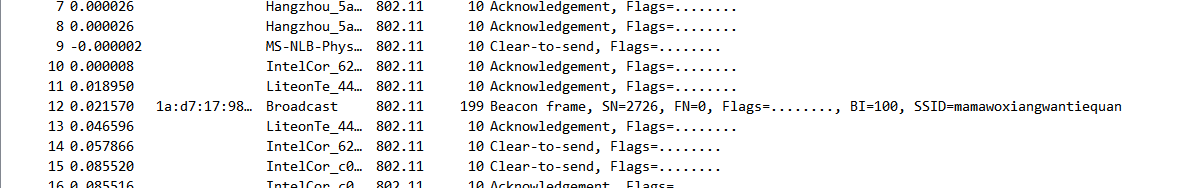
git stash apply 复原文件

**airdecap-ng工具**

**.cap数据包 IEEE 802.11**



在.cap文件中找到 SSID

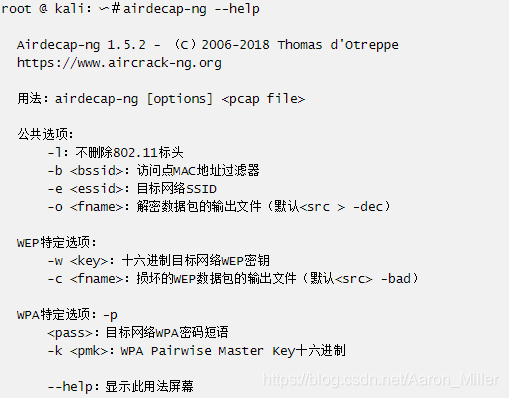


用crunch生成提示相关的密码比如说网络密码一般是八位数，或者使用kali自带字典

 aircrack-ng cacosmia.cap -w /usr/share/wordlists/fern-wifi/common.txt 破解密码

得到密码 12345678 就用aircrack-ng解密流量包

airdecap-ng cacosmia.cap -e mamawoxiangwantiequan -p 12345678



（如果是题目说找网站域名那就试试dns因为它解析域名）

(扫描的协议是arp协议)

有的题目中会出现缺少 png文件头的问题首先复制缺少部分然后在文件头粘贴AS-HEN

照片高度不是固定位置可以求

公式 966x700 求出十六进制 700=2bc

搜索02BC就可以找到高（真理）

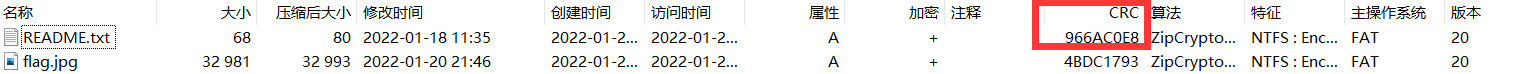
**明文攻击条件**

对于zip文件来说，进行明文攻击的条件是：有一个单独的文件已知且进行压缩之后的[CRC](https://so.csdn.net/so/search?q=CRC&spm=1001.2101.3001.7020)值与某个包含此文件的压缩包的CRC值相等

已知文件：

stickPicture.png

未知密钥压缩包：



只要两者CRC值相等，那么就可以进行明文攻击

攻击问题

可能在有时候已知文件的压缩包已经生成，但是不能进行攻击，报错如下

在选定的档案中没有匹配的文件。如果您想要仅使用文件的一部分执行明文攻击,请修改档案，使每个档案中只包含一个文件。

那么这个时候多半是已知文件再进行的压缩方式与待解密文件的压缩方式不同，使用winRAR可以进行调整压缩方式

**word选项**



