

【CTF】记录一次CTF比赛的Writeup（附题目下载地址）

原创

TeamsSix 于 2019-09-25 17:16:15 发布 2122 收藏 12

分类专栏: [Writeup](#) 文章标签: [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/qq_37683287/article/details/101374506

版权



[Writeup](#) 专栏收录该内容

2 篇文章 0 订阅

订阅专栏

0x00 前言

最近因为省赛快来了, 因此为实验室的小伙伴准备了这次比赛, 总共10道题目, 考虑到大多数小伙伴都刚从大一升到大二, 因此整体难度不高, 当然有几道难度还是有的。

题目大多数都是从网上东找西找的, 毕竟我也是个菜鸟呀, 还要给他们出题, 我太难了。

废话不多说, 直接上Writeup吧, 以下题目的文件下载地址可以在我的公众号 (TeamsSix) 回复CTF获取。

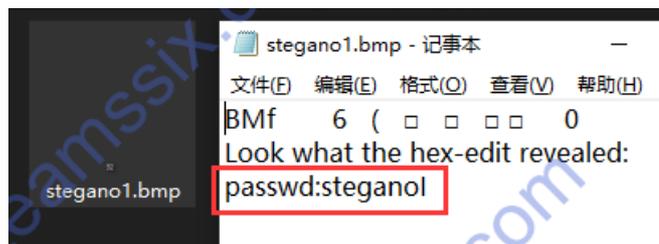
0x01 隐写 1

```
flag: steganoI
```

```
flag格式: passwd:
```

```
题目来源: http://www.wechall.net/challenge/training/stegano1/index.php
```

签到题, 下载题目图片, 利用记事本打开即可看到flag

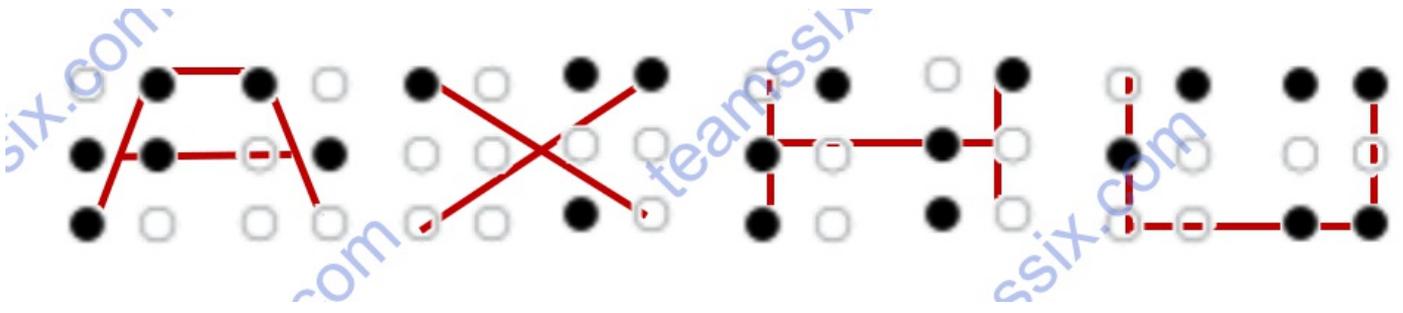


0x02 隐写 2

```
flag: teamssix
```

```
Hint:一般在公共场合才能看的见
```

```
题型参考: http://www.wechall.net/challenge/connect_the_dots/index.php
```



打开图片，参考题目提示说一般在公共场合才能看见，因此通过盲文对照表可以得出flag是teamssix，图片中的AXHU只是用来干扰的，这道题也是我参考wechall里面的一道题型。

a	b	c	d	e	f	g	h	i	j
● ○ ○ ○ ○ ○	● ● ● ○ ○ ○	● ● ○ ○ ○ ○	● ● ○ ● ○ ○	● ○ ○ ● ○ ○	● ● ● ○ ○ ○	● ● ● ● ○ ○	● ○ ● ● ○ ○	○ ● ● ○ ○ ○	○ ● ● ● ○ ○
k	l	m	n	o	p	q	r	s	t
● ○ ○ ○ ● ○	● ○ ○ ○ ● ○	● ● ○ ○ ● ○	● ● ○ ○ ● ○	● ○ ○ ● ○ ○	● ● ● ○ ○ ○	● ● ● ○ ○ ○	● ○ ● ● ○ ○	○ ● ● ○ ○ ○	● ● ○ ○ ● ○
u	v	x	y	z					
● ○ ○ ○ ● ●	● ○ ○ ○ ● ●	● ● ○ ○ ● ●	● ● ○ ○ ● ●	● ○ ○ ● ○ ○					
									w
									○ ● ● ● ○ ●

0x03 Web 1

flag:iamflagsafsfskdf11223

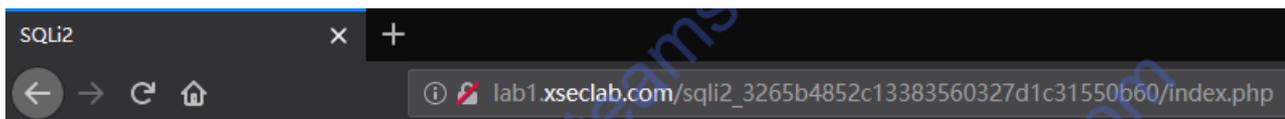
Hint: 站内有提示

题目地址:

http://lab1.xseclab.com/sqli2_3265b4852c13383560327d1c31550b60/index.php

参考来源: <http://hackinglab.cn/ShowQues.php?type=sqlinject>

1、打开题目地址

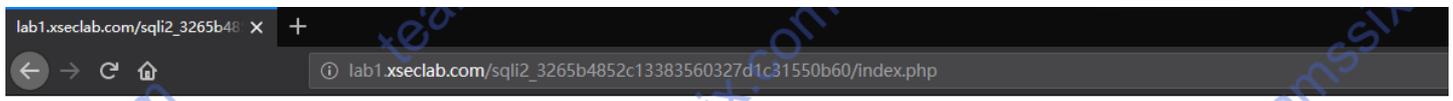


用户名:

密码:

验证码:

验证码:

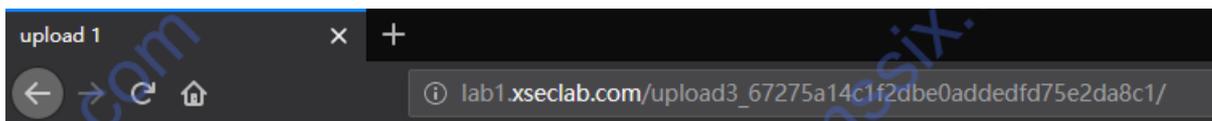


登录成功! 我的座右铭(flag)是 iamflagsafsfskdf11223

0x04 Web 2

```
flag:76tyuh120Kkytig#$$^&  
  
题目地址: http://lab1.xseclab.com/upload3_67275a14c1f2dbe0addedfd75e2da8c1/  
  
flag格式: key is :  
  
题目来源: http://hackinglab.cn/ShowQues.php?type=upload
```

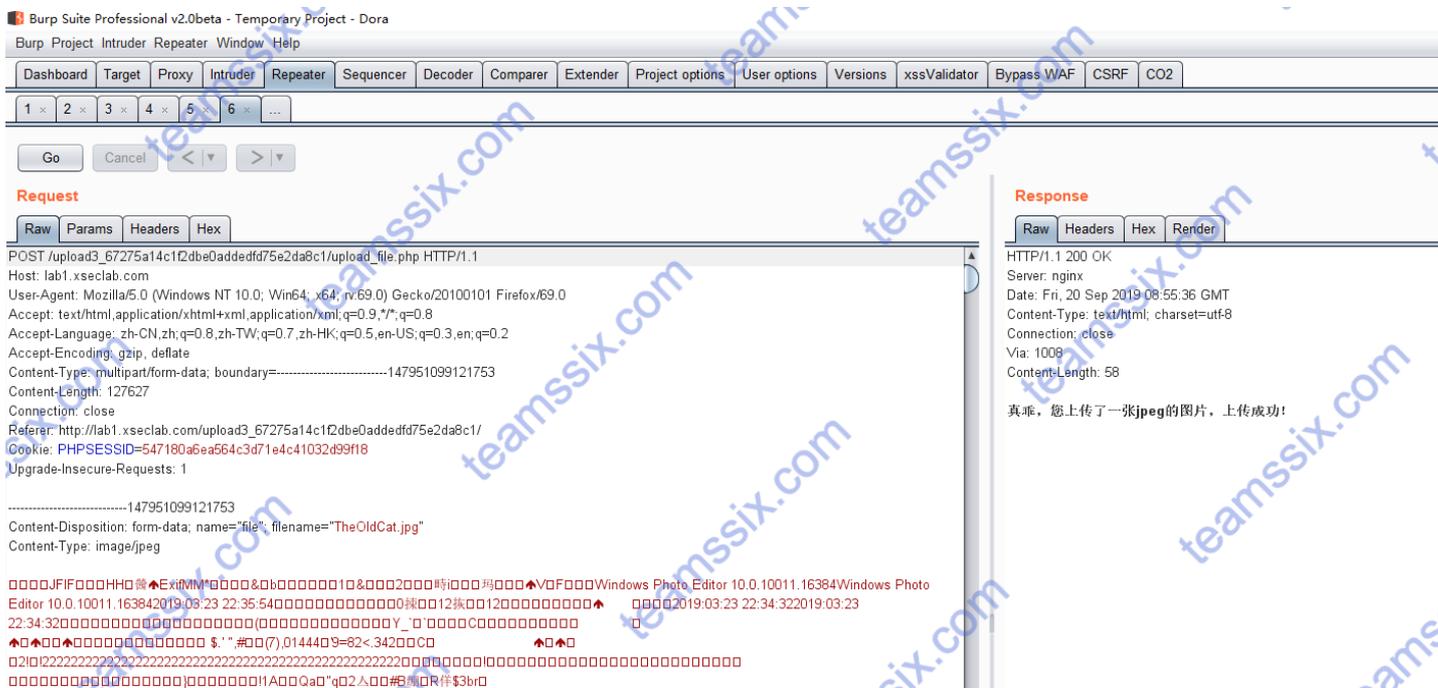
1、打开题目地址，发现是一个文件上传界面



请上传一张JPG格式的图片!

文件名 未选择文件。

2、先把Burp挂上，随便上传一个JPG图片试试，并来到Burp重发这个包



3、在Burp中对文件名进行修改，比如在jpg后加上.png或者其他东西，成功看到flag



0x05 soeasy

flag:HackingLabHdd1b7c2fb3ff3288bff

Hint:在这个文件中找到key就可以通关

flag格式:key:

题目来源: <http://hackinglab.cn/ShowQues.php?type=pentest>

解法一:

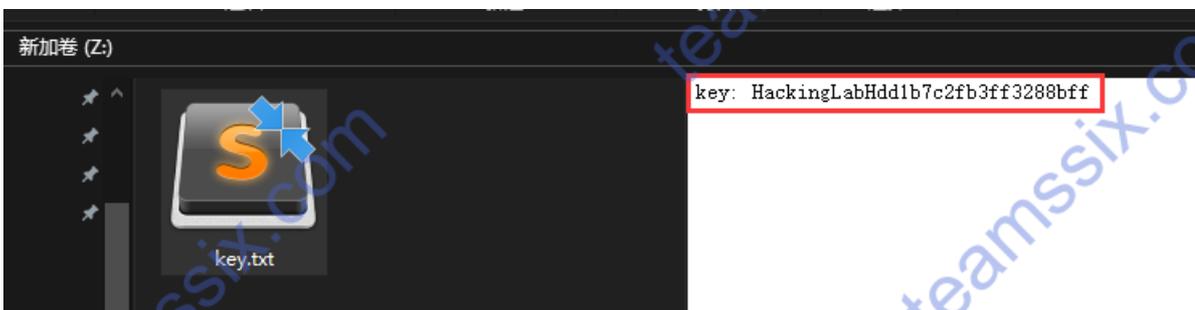
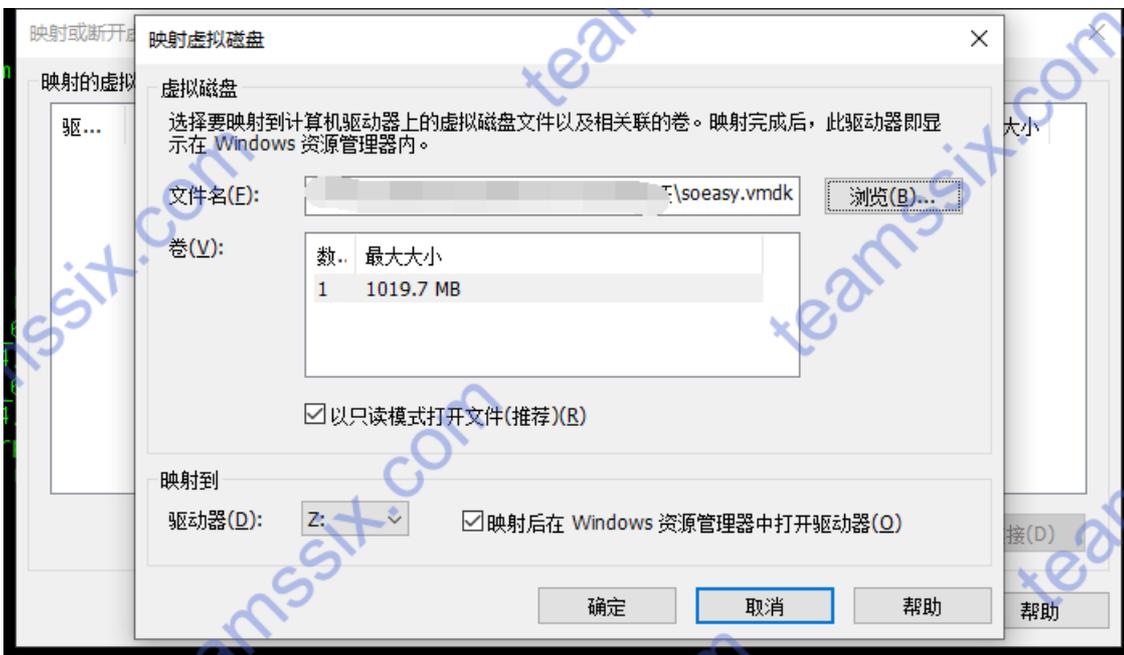
1、下载文件后,发现是vmdk文件,利用DeskGenius打开后,发现Key,此为正确答案





解法二:

1、利用Vmware映射虚拟硬盘同样可以打开



0x06 Crack

flag:19940808

Hint:flag就是密码

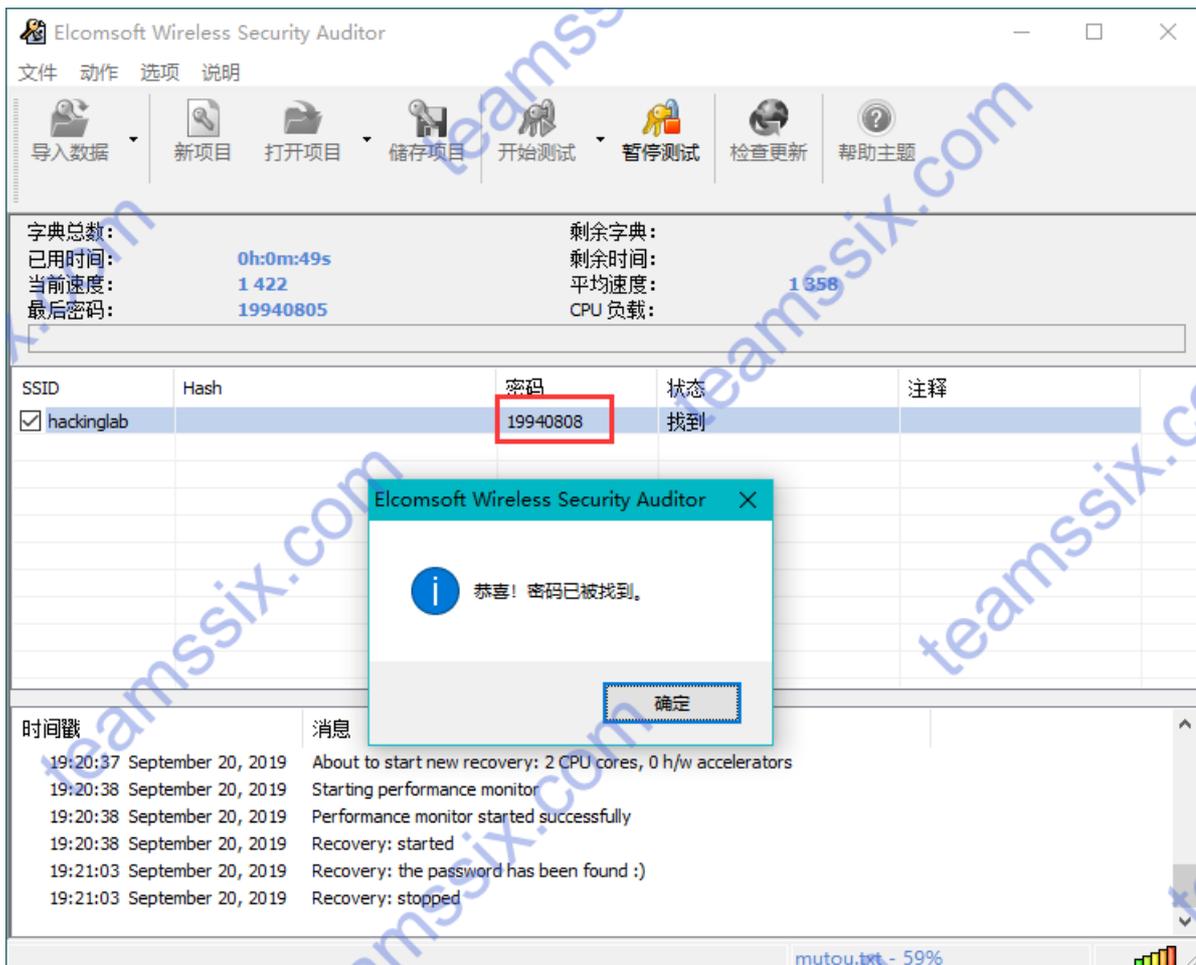
题目：邻居悄悄把密码改了，你只知道邻居1994年出生的，能找到她的密码吗？

题目来源：<http://hackinglab.cn/ShowQues.php?type=decrypt>

1、下载题目文件，根据题意，需要对WiFi密码破解，而且密码很有可能是邻居的生日，因此我们利用工具生成字典。



2、接下来利用ewsa进行破解，可以看到破解后的密码



这道题目当时实验室有人用kali做的，kali下的工具感觉破解速度更快。

0x07 Bilibili

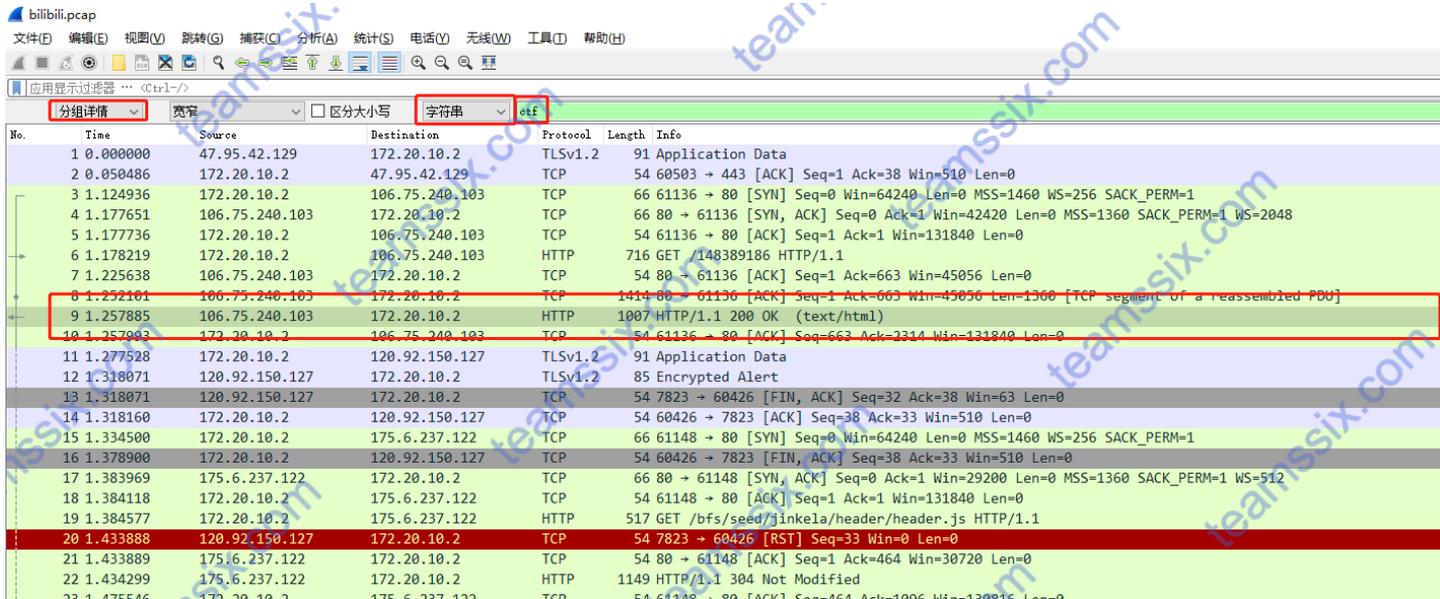
flag:Congratulations_you_got_it

题目: bilibili

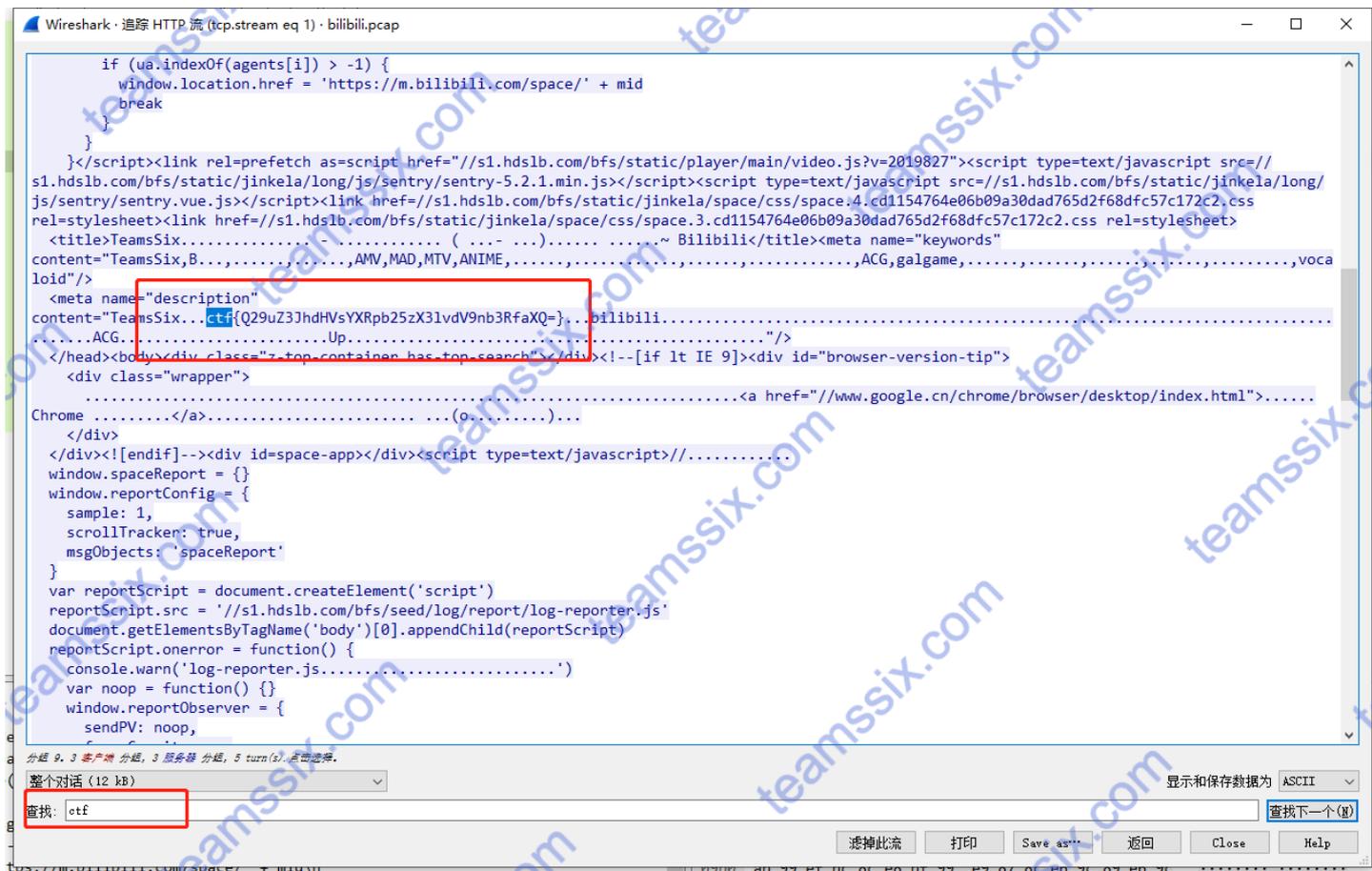
flag格式: ctf{}

解法一:

1、使用Wireshark打开数据包, 直接搜索ctf



2、找到标识的那一行右击进行追踪对应的协议, 比如这条是http协议就追踪http协议, 之后再次查找ctf



3、发现ctf括号后的内容为base64加密, 解密即可得到flag

加密/解密 散列/哈希 BASE64 图片/BASE64转换

明文:
Congratulations_you_got_it

BASE64:
Q29uZ3JhdHVzYXRpb25zX3lvdV9nb3RfaXQ=

BASE64编码 >

< BASE64解码

解法二:

1、和解法一一样，对数据包进行追踪http流，不难看出这是访问space.bilibili.com/148389186的一个数据包

Wireshark · 追踪 HTTP 流 (tcp.stream eq 1) · bilibili.pcap

```
GET /148389186 HTTP/1.1
Host: space.bilibili.com
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) Appl
Accept: text/html,application/xhtml+xml,application/xml;q=
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: LIVE_BUVID=AUT07615641143128169; sid=8y4vuzga; Ded
SESSDATA=4c80f47c%2C1571577236%2C57daee91; bili_jct=5b823d
```

2、打开这个网址，同样可以看到被base64加密的flag

space.bilibili.com/148389186

主站 音频 游戏中心 直播 会员购 漫画 70年 下载APP 预言家日报上B站啦!

Teams Six LU4 ctf{Q29uZ3JhdHVzYXRpb25zX3lvdV9nb3RfaXQ=}

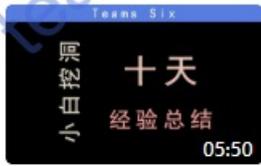
主页 动态 投稿 5 频道 0 收藏 1 订阅 设置 搜索视频

我的粉丝会看到 新访客会看到

设置置顶视频

置顶视频是粉丝们看到的第一个视频。请选择你最喜欢的作品，让粉丝们一饱眼福吧~

我的视频 4 最新发布 最多播放 最多收藏



【经验总结】小白挖洞十天经验分享



【Python实例】今天才知道原来B站电影中播放、弹幕



【漏洞复现】CVE 2019-0708 | 17年的勒索病毒又双叒卷土



【Python实例】批量下载斗罗大陆高清视频

另外打个小广告，上面这个是我的bilibili号（TeamsSix），欢迎大家关注，嘿嘿

0x08 Check

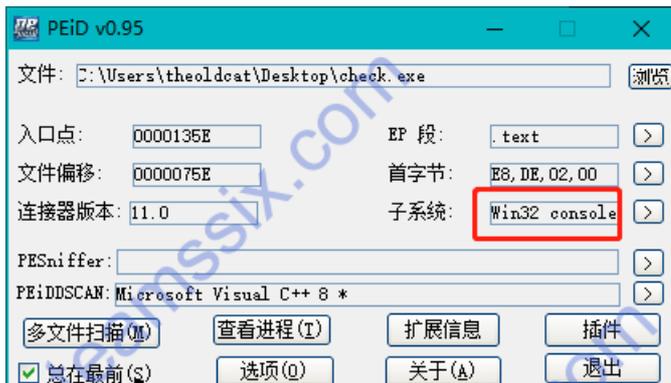
flag:sAdf_fDfk1_Fdf

题目：简单的逆向

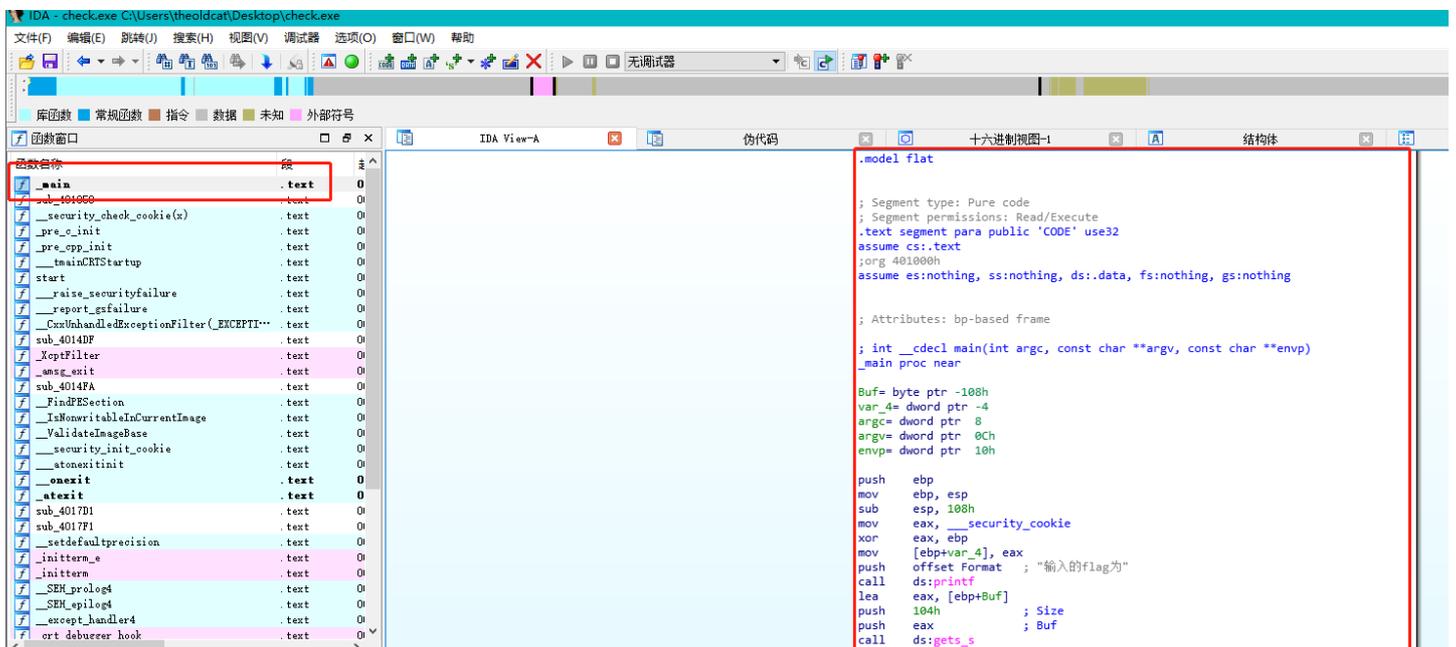
flag格式：flag{}

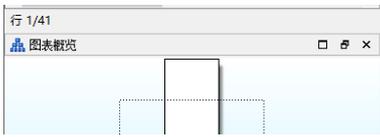
题目来源：<https://www.cnblogs.com/QKsword/p/9095242.html>

1、下载文件，发现是exe文件，放到PEiD里看看有没有壳以及是什么语言编写的，如果有壳需要先脱壳。



2、可以看到使用的C语言写的，同时是32位，因此使用IDA32位打开，之后找到main函数





```
add esp, 4
lea ecx, [ebp+Buf]
call sub_401050
mov ecx, [ebp+var_4]
xor eax, eax
xor ecx, ebp
call @_security_check_cookie@4 ; __security_check_cookie(x)
mov esp, ebp
pop ebp
```

3、按F5查看伪代码，并点击sub_401050子函数

```
IDA View-A 伪代码
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3   char Buf; // [esp+0h] [ebp-108h]
4   printf(Format);
5   gets_s(&Buf, 0x104u);
6   sub_401050(&Buf);
7   return 0;
8 }
9 }
```

4、不难看出下列是一个10进制到ASCII码的转换

```
1 int __thiscall sub_401050(_BYTE *this)
2 {
3   int result; // eax
4   if ( *this != 102 )
5     goto LABEL_22;
6   if ( this[1] == 108
7     && this[2] == 97
8     && this[3] == 103
9     && this[4] == 123
10    && this[5] == 115
11    && this[6] == 65
12    && this[7] == 100
13    && this[8] == 102
14    && this[9] == 95
15    && this[10] == 102
16    && this[11] == 68
17    && this[12] == 102
18    && this[13] == 107
19    && this[14] == 108
20    && this[15] == 95
21    && this[16] == 70
22    && this[17] == 100
23    && this[18] == 102
24    && this[19] == 125 )
25   {
26     printf("yes,this is a flag");
27     getchar();
28   }
29 LABEL_22:
30   result = 0;
31 }
32 return result;
33 }
```

5、利用在线网站转换即可获得flag，网站地址：<http://ctf.ssleye.com/jinzi.html>

文本

清空

二进制 01100110 01101100 01100001 01100111 01111011 01110011 01000001 01100100 01100110 01011111 01100110 01000100 01100110 01101011 01101100
01011111 01000110 01100100 01100110 01111101

十进制

十六进制 66 6c 61 67 7b 73 41 64 66 5f 66 44 66 6b 6c 5f 46 64 66 7d

0x09 Android RE

flag:DDCTF-397a90a3267641658bbc975326700f4b@didichuxing.com

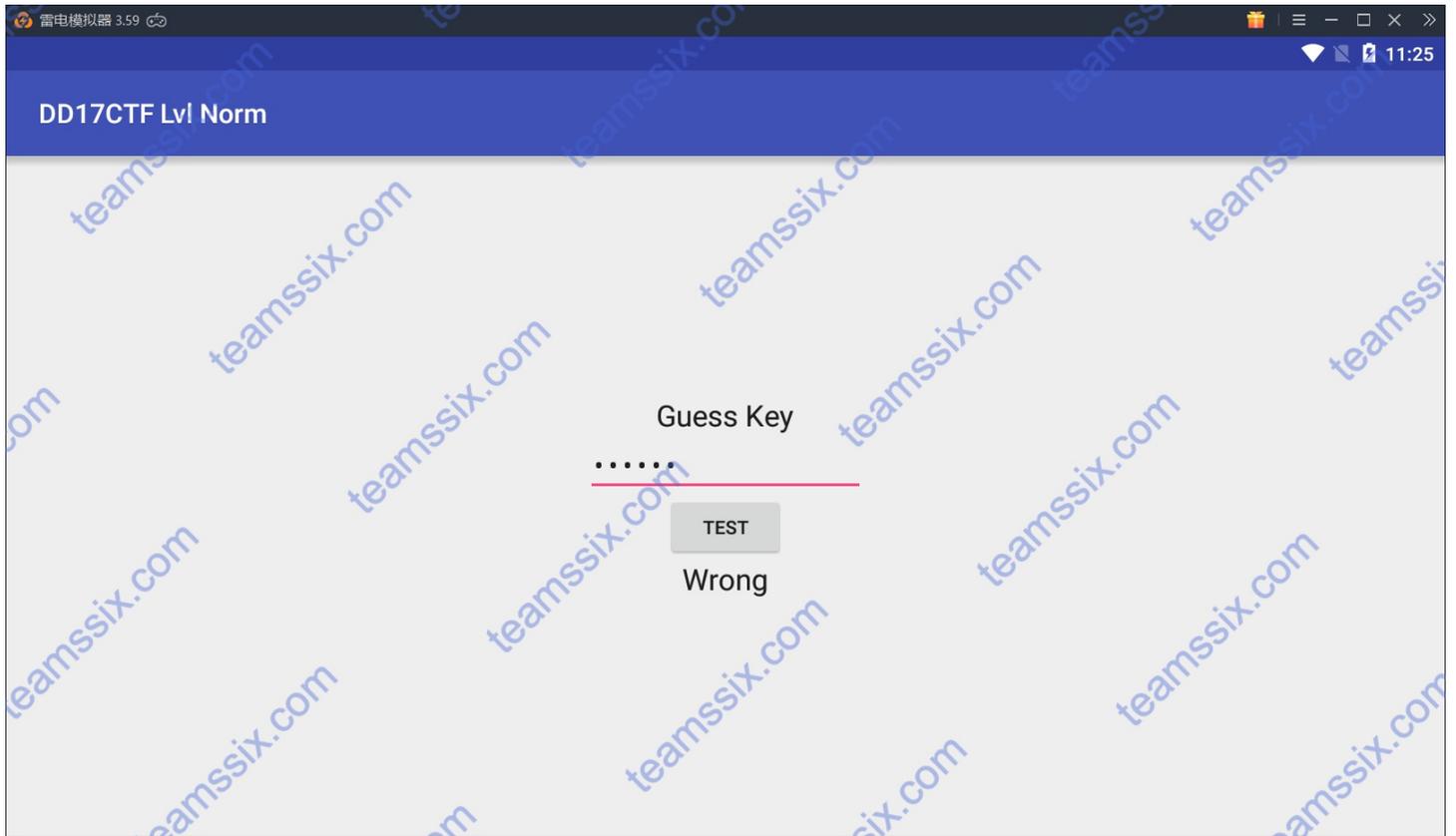
题目: 安卓逆向

flag格式: DDCTF-

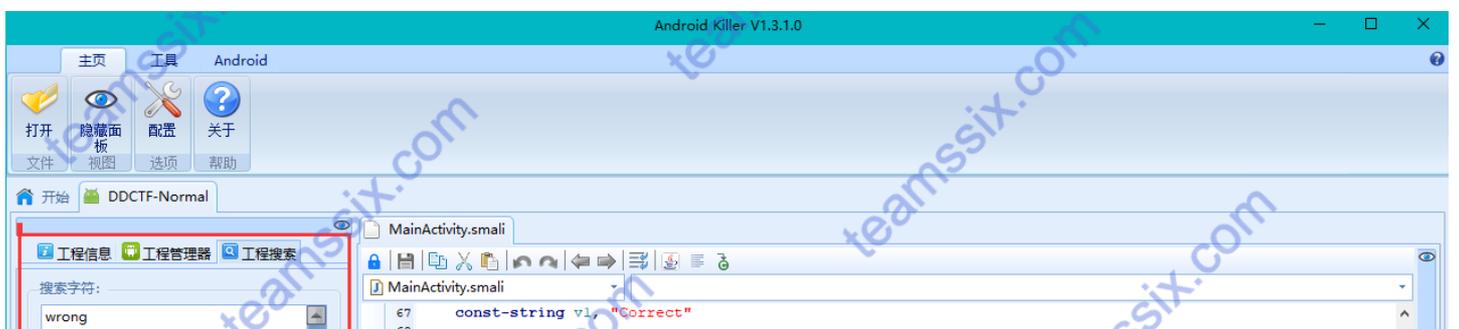
Hint: flag中包含chuxing

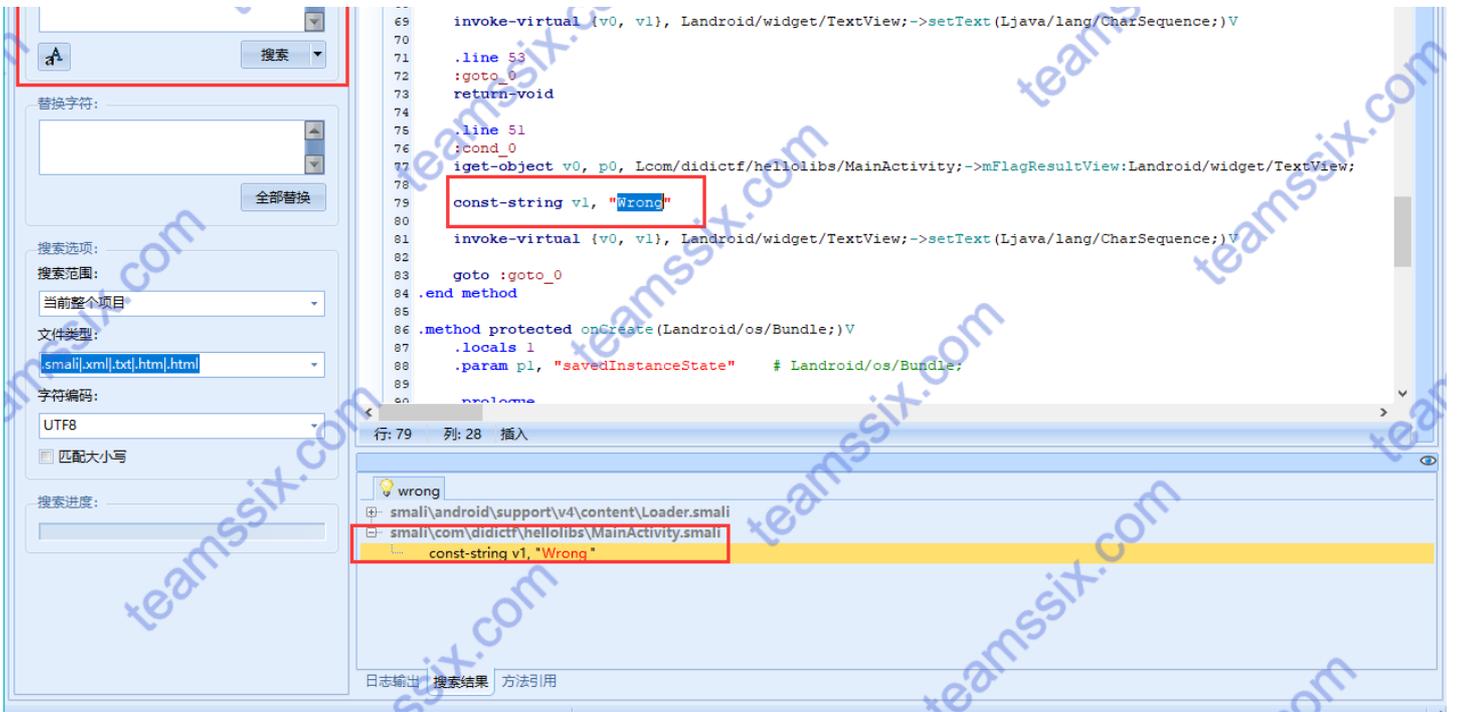
题目来源: <https://xz.aliyun.com/t/1103>

1、这道题是滴滴出行的一道CTF，下载题目可以看到一个apk文件，先在模拟器中运行看看是个什么东西

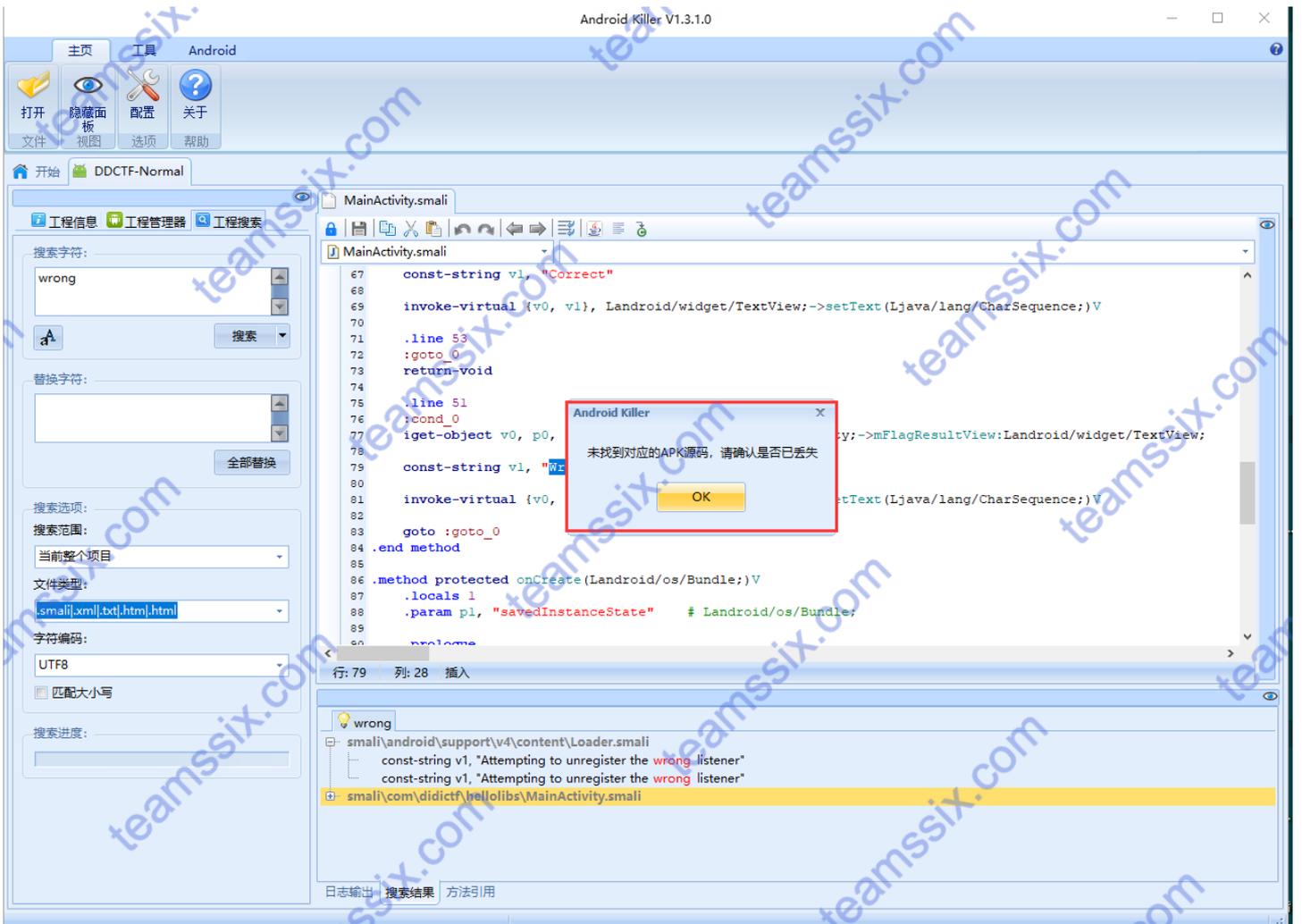


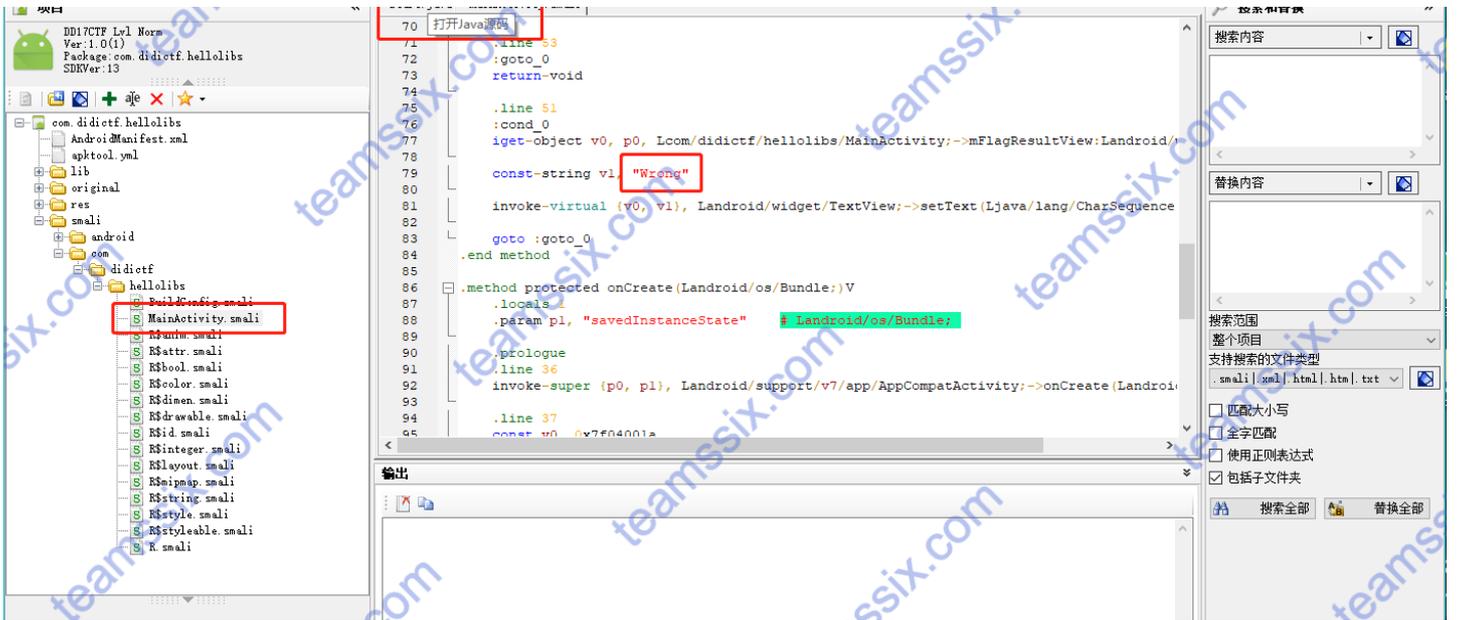
2、功能很简单，一个输入框，输错会提示Wrong，那么利用Android killer给它反编译一下，查找字符“Wrong”



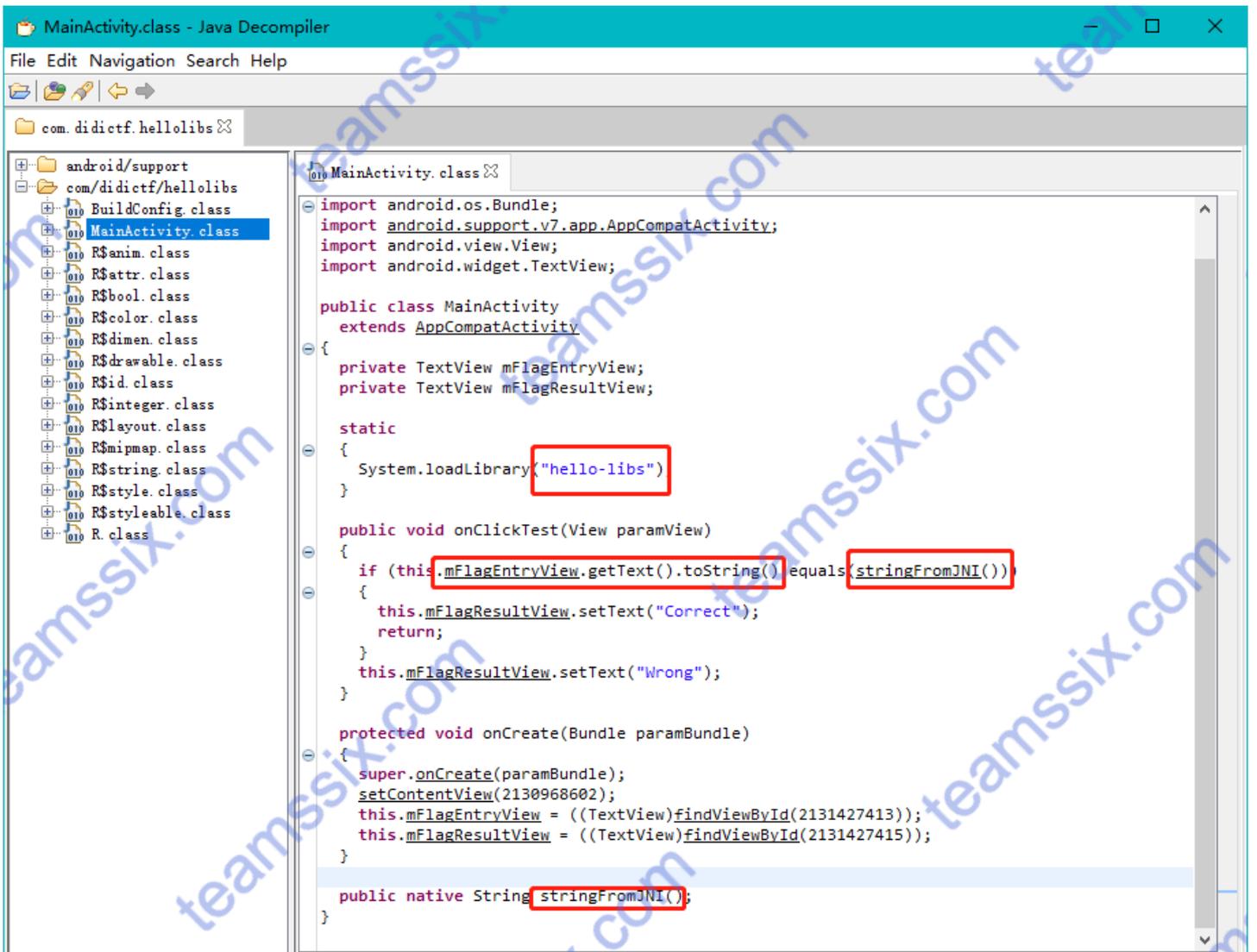


3、可以看到Wrong字符的路径，接下来进行反编译，不过可能由于本身软件的文件，反编译提示未找到对应的APK源码，没关系，换ApkIDE对其进行编译





4、等待一段时间后，可以看到对应源码，简单分析就可以知道该代码从hello-libs.so文件加载，并且对 mFlagEntryView.getText().toString()函数的内容即我们输入的内容和stringFromJNI()函数的内容做判断，如果一致就Correct，即正确，不一致就返回Wrong，即错误，那么接下来只需要分析stringFromJNI()的内容就行了，因此我们需要知道系统从hello-libs.so文件加载了什么



5、将APK解压，找到hello-libs.so文件，由于现在手机都是用arm64位的CPU（我也不知道是不是的啊，听别人说的），因此我们找到arm64-v8a文件夹下的libhello-libs.so文件。用IDA打开



其中, mips、armeabi、armeabi-v7a和x86都表示CPU的类型。一般的手机或平板都是用arm的cpu。

armeabi 是针对普通的或旧的arm v5 cpu, 32位

armeabi-v7a 是针对有浮点运算或高级扩展功能的arm v7 cpu, 32位

arm64-v8a 针对64位的

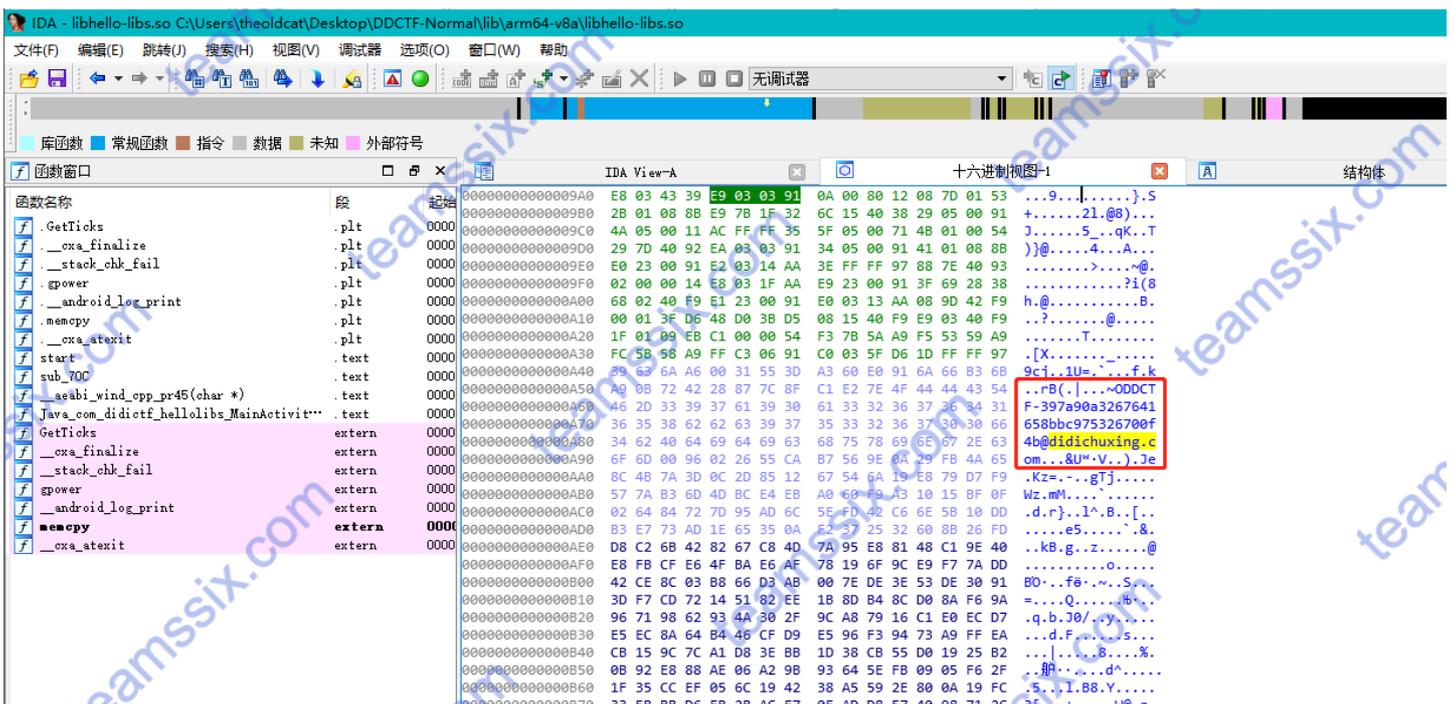
mips 是一种采取精简指令集 (RISC) 的处理器架构, 32位

mips64 64位

x86 IA-32位指令集

x86_64 64位

6、打开IDA后, 根据题目提示, Alt + T 查找chuxing



7、成功找到flag (DDCTF-397a90a3267641658bbc975326700f4b@didichuxing.com)

) 输入到模拟器中看到提示Correct, 说明flag正确。



0x10 Easy_dump

```
flag: F0rens1cs_St2rt
```

```
题目: Easy_dump
```

```
flag格式: LCTF{}
```

```
Hint: volatilty了解一下
```

```
题目来源: https://www.tr0y.wang/2016/12/16/MiniLCTF/index.html
```

解法一:

1、下载题目文件, 提示利用volatility工具, 同时结合文件后缀为vmem (VMWare的虚拟内存文件), 因此判断是一个内存取证
的题目, 关于volatility的使用可以参考官方手册: <https://github.com/volatilityfoundation/volatility/wiki/Command-Reference>, 废话不
多说, 先看看镜像信息

```
# volatility -f xp.vmem imageinfo
```

```
root@kali:~/Desktop# volatility -f xp.vmem imageinfo
Volatility Foundation Volatility Framework 2.6
INFO : volatility.debug : Determining profile based on KDBG search...
Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
AS Layer1 : IA32PageMemoryPae (Kernel AS)
AS Layer2 : FileAddressSpace (/root/Desktop/xp.vmem) 文件 此电脑.ln
PAE type : PAE
DTB : 0xb2a000L
KDBG : 0x8054e2e0L
Number of Processors : 2
Image Type (Service Pack) : 3
KPCR for CPU 0 : 0xffdf000L
KPCR for CPU 1 : 0xfc72b000L
KUSER_SHARED_DATA : 0xffdf0000L
Image date and time : 2016-12-02 11:17:55 UTC+0000
Image local date and time : 2016-12-02 19:17:55 +0800
```

2、可以看到该镜像信息的为WinXPSP2x86，接下来直接扫描查看一些系统文件中有没有flag文件

```
# volatility -f xp.vmem --profile=WinXPSP2x86 filescan | grep flag
```

```
root@kali:~/Desktop# volatility -f xp.vmem --profile=WinXPSP2x86 filescan | grep flag
Volatility Foundation Volatility Framework 2.6
0x0000000005ab74c8 1 0 RW-r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\flag.txt
0x0000000007782ef8 1 0 RW-rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\Recent\flag.lnk
```

3、将该flag.txt文件dump下来

```
# volatility -f xp.vmem --profile=WinXPSP2x86 dumpfiles -Q 0x0000000005ab74c8 -D ./ -u
```

```
root@kali:~/Desktop# volatility -f xp.vmem --profile=WinXPSP2x86 dumpfiles -Q 0x0000000005ab74c8 -D ./ -u
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x05ab74c8 None \Device\HarddiskVolume1\Documents and Settings\Administrator\桌面\flag.txt
```

4、直接cat flag文件即可看到flag

```
root@kali:~/Desktop# ls
file.None.0xff876ac0.dat xp.vmem
root@kali:~/Desktop# cat file.None.0xff876ac0.dat
LCTF{F0renslcs_St2rt}root@kali:~/Desktop#
```

解法二：

因为该题作者将flag复制到了自己电脑的粘贴板里的，所以直接获取粘贴板的内容也是可以看到flag的，不过谁能想到这种操作[笑哭]

```
# volatility -f xp.vmem clipboard
```

```
root@kali:~/Desktop# volatility -f xp.vmem clipboard
Volatility Foundation Volatility Framework 2.6
Session WindowStation Format Handle Object Data
-----
0 WinSta0 0xc009L 0x7e009d 0xe1577c30
0 WinSta0 CF_UNICODETEXT 0x1c00d1 0xe10ca7c0 LCTF{F0renslcs_St2rt}
0 WinSta0 0xc013L 0x1000b3 0xe1260b38
0 WinSta0 CF_LOCALE 0x16008b 0xe15a08e8
0 WinSta0 CF_TEXT 0x1 -----
0 WinSta0 CF_OEMTEXT 0x1 -----
```

以上就是本次我为他们准备的CTF的全部内容，大多数都是很基础的题目，平时拿来练练手还是不错的，拓宽一下自己的了解面，发现一些自己以前不知道的东西，如果你也想拿上面的题目来玩玩，在公众号（TeamsSix）回复CTF就可以获取下载地址哦。

更多信息欢迎关注微信公众号：TeamsSix

原文链接：<https://www.teamssix/year/190925-114420.html>