

## RE: Header File

### Challenge:

[flag.c](#)

[P1.h](#)

[P2.h](#)

[P3.h](#)

### Solution:

Sau khi tải các file đề bài cho, chạy file flag.c nhận được đoạn văn bản

```
n0w_H0w_T0_Wr1t3_Ur_Own_L1Br4ry}miniCTF{N0w_U_K
```

Nhưng dựa vào form flag ta thấy có vẻ flag đã bị thay đổi thứ tự. Đọc source của file flag.c ta thấy hàm sub1(), sub2(), sub3() được gọi theo đúng thứ tự đó. Ta để ý ở phần include có 3 thư viện “P1.h”, “P2.h” và “P3.h”. 3 file này gọi là header file, khá giống với thư viện trong C.

Khi đọc source của 3 file đó, ta đoán hàm sub3() trong file “P1.h” sẽ phải được gọi trước, sau đó là đến hàm sub1() trong file “P2.h” và cuối cùng là hàm sub2() trong file “P3.h”. Vì thế, để tìm được flag, ta chỉ cần gọi 3 hàm này theo đúng thứ tự trên sẽ ra được flag của bài

```
1  #include<stdio.h>
2  #include"P1.h"
3  #include"P2.h"
4  #include"P3.h"
5
6  int main(){
7      int a1[] = {333, 225, 119, 297, 360, 39, 378, 490, 48,
8      int a2[] = {8322, 4067, 9280, 3417, 7220, 7225, 7524,
9      int a3[] = {108, 101, 101, 89, 42, 48, 21, 59, -3, -52
10     char key[] = "ISPCLUB";
11     sub3(a3);
12     sub1(a1);
13     sub2(a2, key);
14 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\BCTF_2024\10000\5000\5000\ISP\miniCTF\Reverse\Header f
($?) { gcc decrypt.c -o decrypt } ; if ($?) { .\decrypt }
miniCTF{N0w_U_Kn0w_H0w_T0_Wr1t3_Ur_0wn_L1Br4ry}
PS D:\BCTF_2024\10000\5000\5000\ISP\miniCTF\Reverse\Header f
```

Flag: miniCTF{N0w\_U\_Kn0w\_H0w\_T0\_Wr1t3\_Ur\_0wn\_L1Br4ry}