

# CTF-pwn 2014-stkof writeup

原创

[Vic1fe](#) 于 2019-09-18 20:14:06 发布 213 收藏

分类专栏: [pwn](#) 文章标签: [pwn](#) [unlink](#) [exp](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_41918771/article/details/100985377](https://blog.csdn.net/qq_41918771/article/details/100985377)

版权



[pwn](#) 专栏收录该内容

19 篇文章 1 订阅 ¥9.90 ¥99.00

订阅专栏  超级会员免费看

题目链接: [Github](#)

参考链接: [传送门](#)

堆的一些基础这里就不再介绍了, 网上有很多, 也可以加qq群一起讨论: **946220807**

准备开始正文

## 读懂题目

拿到题目，开启我们的IDA查看伪代码。运行程序并没有使用帮助，只能自己慢慢琢磨了。

```
2{
3  int choice; // eax
4  signed int v5; // [rsp+Ch] [rbp-74h]
5  char nptr; // [rsp+10h] [rbp-70h]
6  unsigned __int64 v7; // [rsp+78h] [rbp-8h]
7
8  v7 = __readfsqword(0x28u);
9  alarm(0x78u);
10 while ( fgets(&nptr, 10, stdin) )
11 {
12     choice = atoi(&nptr);
13     if ( choice == 2 )
14     {
15         v5 = fill();
16         goto LABEL_14;
17     }
18     if ( choice > 2 )
19     {
20         if ( choice == 3 )
21         {
22             v5 = free_chunk();
23             goto LABEL_14;
24         }
25         if ( choice == 4 )
26         {
27             v5 = print();
28             goto LABEL_14;
29         }
30     }
31     else if ( choice == 1 )
32     {
33         v5 = alloc();
34         goto LABEL_14;
35     }
36     v5 = -1;
37 LABEL_14:
38     if ( v5 )
39         puts("FAIL");
40     else
41         puts("OK");
42     fflush(stdout);
43 }
```

可以看到输入不同的数字对应不同的函数(ida不同函数名可能也不同)，共4个函数：

1. `fill()`：这个函数用来向分配的空间填充数据。
2. `free_chunk()`：用来释放malloc分配的空间
3. `print()`：没什么卵用，本以为是打印用户的数据的，啥也不输出。
4. `alloc()`：这个函数用malloc分配空间供用户使用，并且把返回的指针放到全局数组global中。

函数具体实现方法点进去看看就