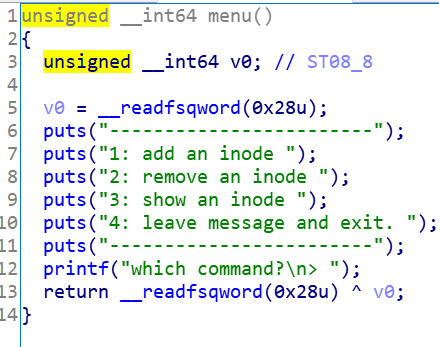
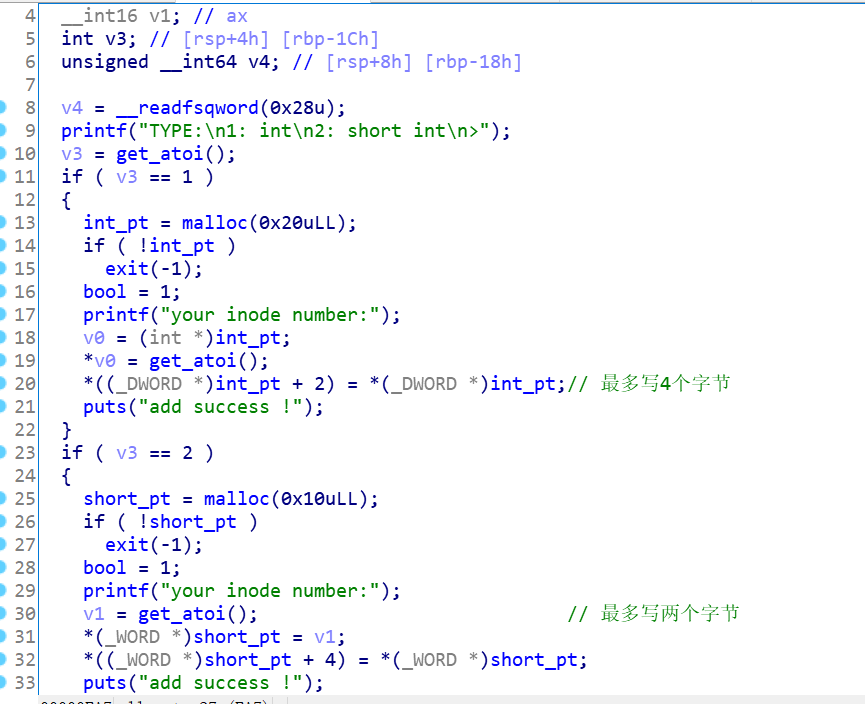


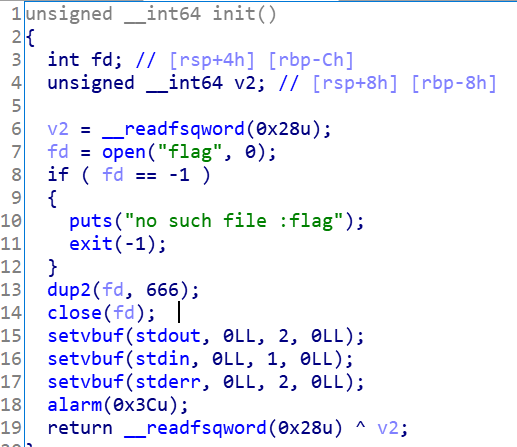
禁止了很多系统调用：execve

只能orw了



Allocate





文件句柄为666

利用思路为：

1、 利用tcache dup修改size为0x91，然后释放8次，得到libc的基地址的低4个字节

2、 获取到stdin的地址，并且获取到fileno的地址，可以用print \*(\_IO\_FILE\*)（&stdin）来打印

3、 再次利用tcache dup将该地址处从0改为666

4、 触发leave函数，读取flag。

这里的坑：

1. 不知道可以修改fileno这个东西

2. tcache dup用得不是很熟练

需要注意，如果free(1)又free(1)，那么在tcache bin中的状态是

Chunk1 fd -🡪 chunck1 fd

第一次malloc的时候，会把chunk1给我们，同时指针指向chunk1 fd，然后我们再给fd赋值。

第二次，由于tcache entry指向的还是chunk1 fd，所以还是会把chunk1给我们。然后这个时候，tcache entry的next指针才会指向刚刚编辑的fd的值。这个时候，tcache bin对应size的count已经是0了。再分配一次，就会变成0xff，永远都别想从tacahe分配了。

第三次，分配，才会分配到我们指定的地址去。

坑有点多，顶不住，做了一下午才做出来。