```c
#include <fcntl.h>
#include <unistd.h>
#include <sys/types.h>
#include <stdio.h>
#include <stdlib.h>

int lock_set(int fd, int type)
{
    struct flock lock;
    lock.l_type = type;
    lock.l_len = 0;
    lock.l_start = 0;
    lock.l_whence = SEEK_SET;
    fcntl(fd, F_GETLK, &lock); // 获取文件的加锁信息
    if (lock.l_type != F_UNLCK)
    {
        if (lock.l_type == F_RDLCK)
        {
            printf("文件已经被加上了读锁 PID=%d", lock.l_pid);
        }
        if (lock.l_type == F_WRLCK)
        {
            printf("文件已经加上了写锁 PID=%d", lock.l_pid);
        }
    }
    lock.l_type = type;
    if ((fcntl(fd, F_SETLKW, &lock)) < 0)
    {
        printf("加锁失败\n");
        return -1;
    }
    switch (lock.l_type)
    {
    case F_RDLCK:
    {
        printf("Read lock set by %d\n", getpid());
        break;
    }
    case F_WRLCK:
    {
        printf("write lock set by %d\n", getpid());
```

```c
            break;
        }
        case F_UNLCK:
        {
            printf("unlock by %d\n", getpid());
            break;
        }
        default:
            break;
        }
        return 0;
}
```