# 题目：实验1-1-3添加系统调用

# 一、实验目的

（1）建立对系统调用接口的深入认识；

（2）掌握系统调用的基本过程；

（3）能添加自定义系统调用，完成系统调用的全面控制；

# 二、实验内容

（1）添加自定义系统调用

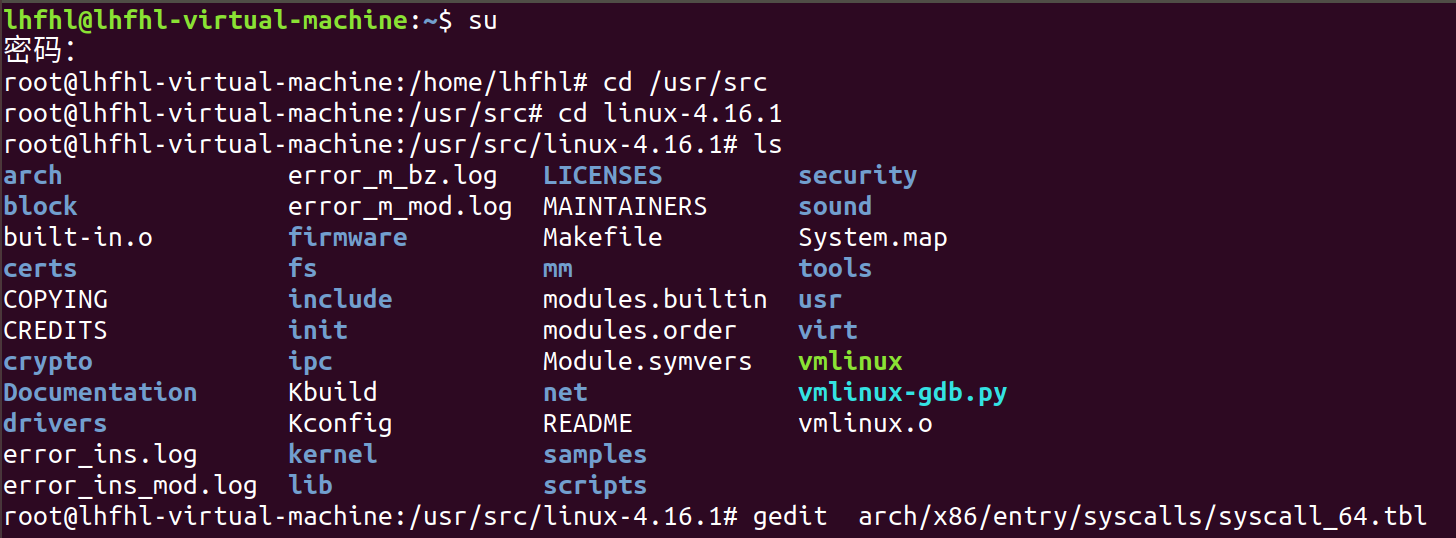
（2）写一个简单的应用程序测试（1）中添加的系统调用

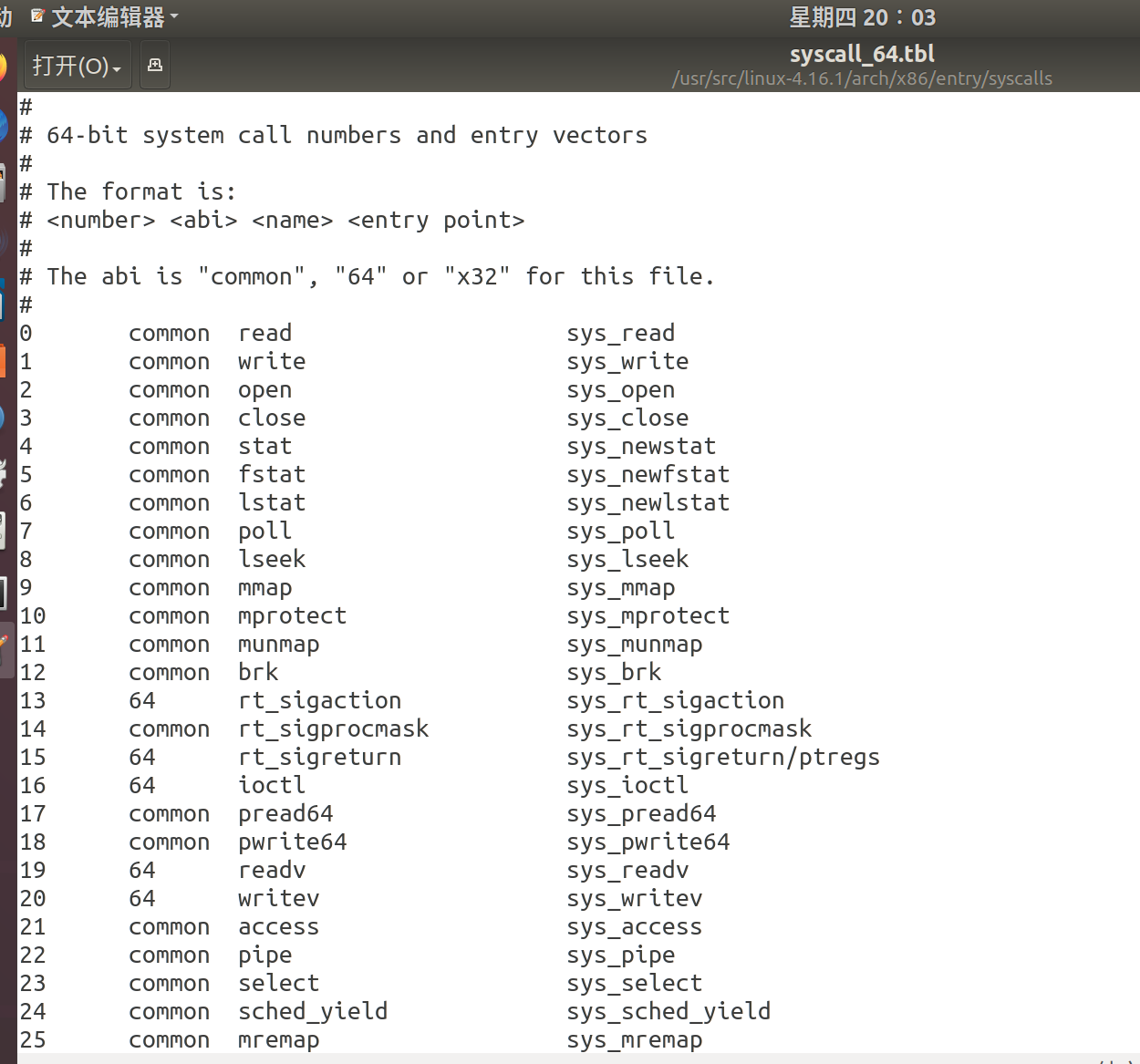
# 三、实验设计原理/步骤

在实验1-1-1的基础上，进行本次实验

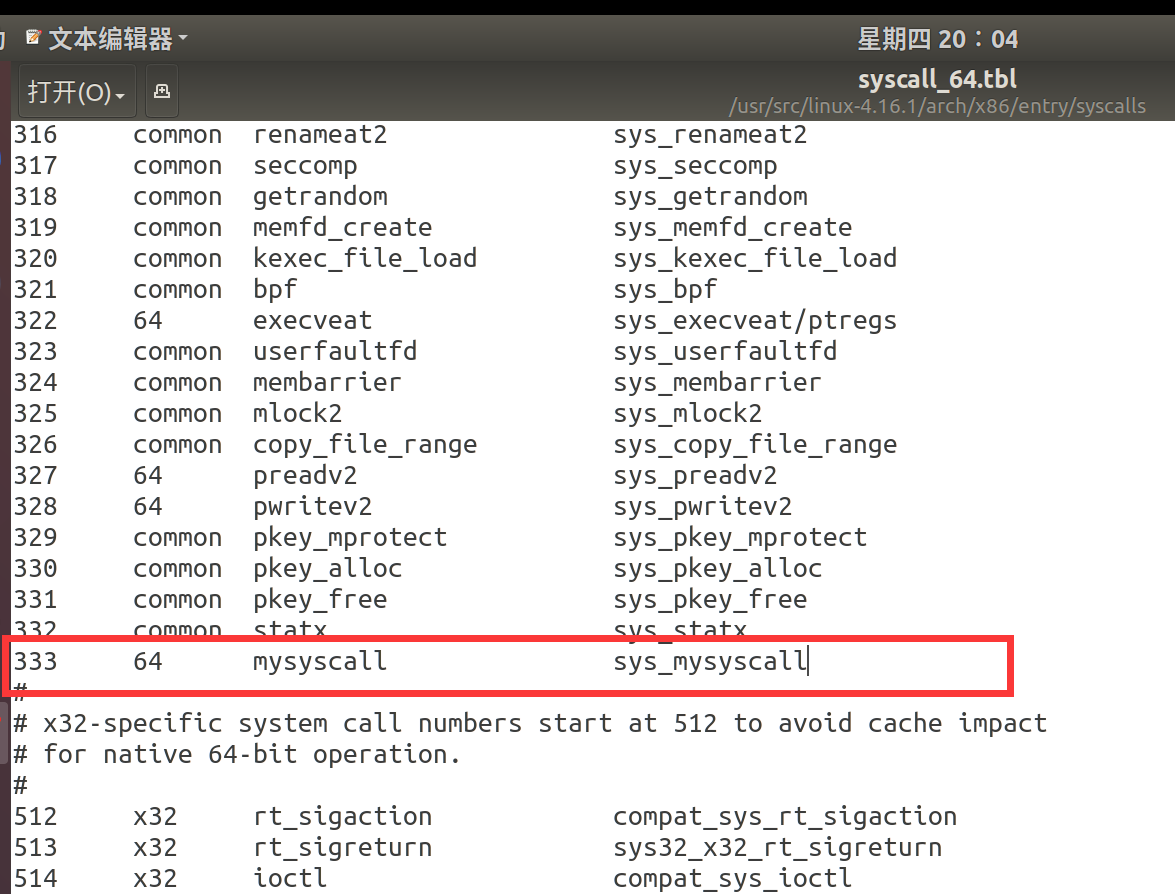
3.1编辑系统调用入口表

先切换目录，再打开syscall\_64.tbl文件





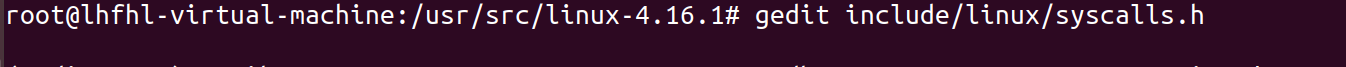
然后定位到文件中间，添加我们的系统调用号和名字：

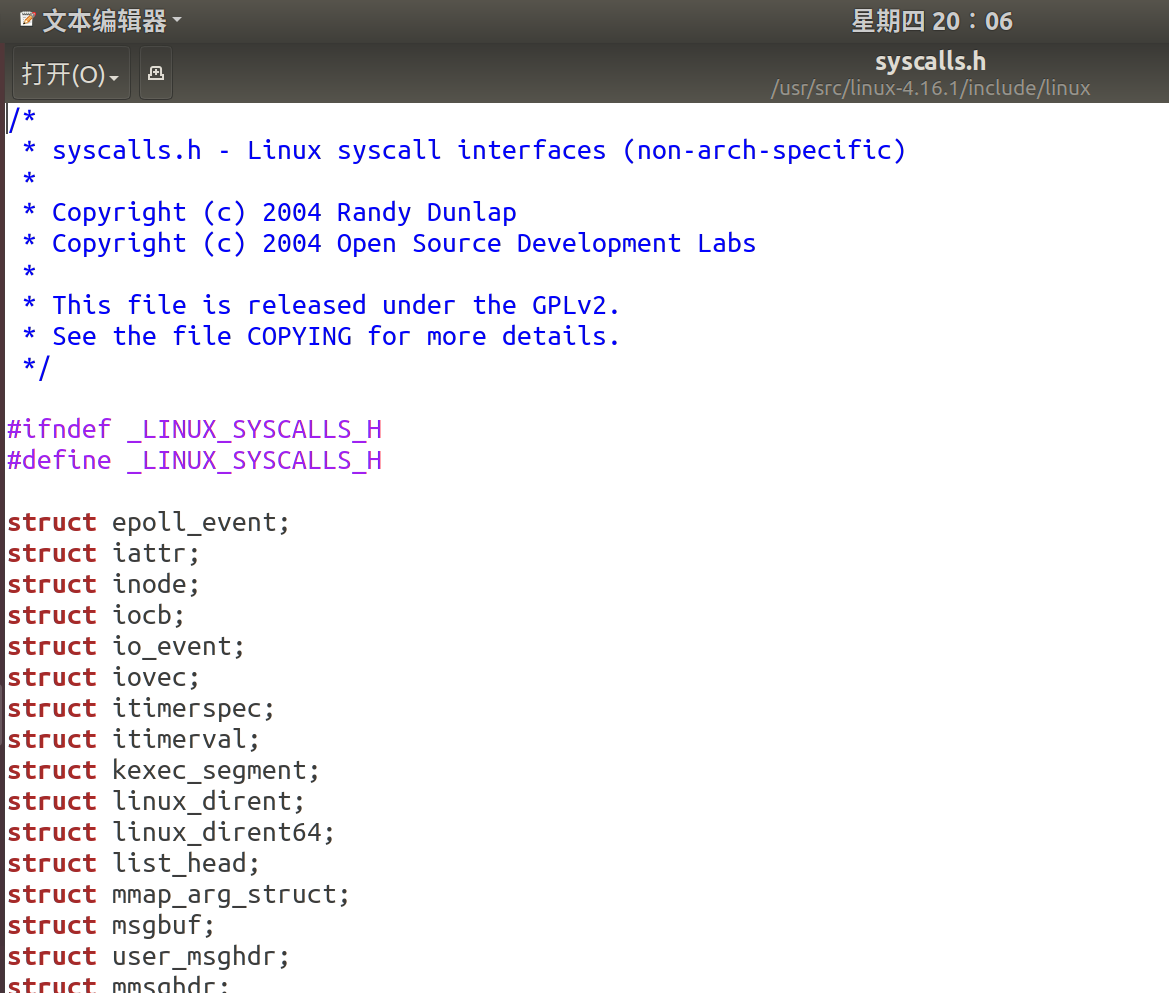


保存，退出

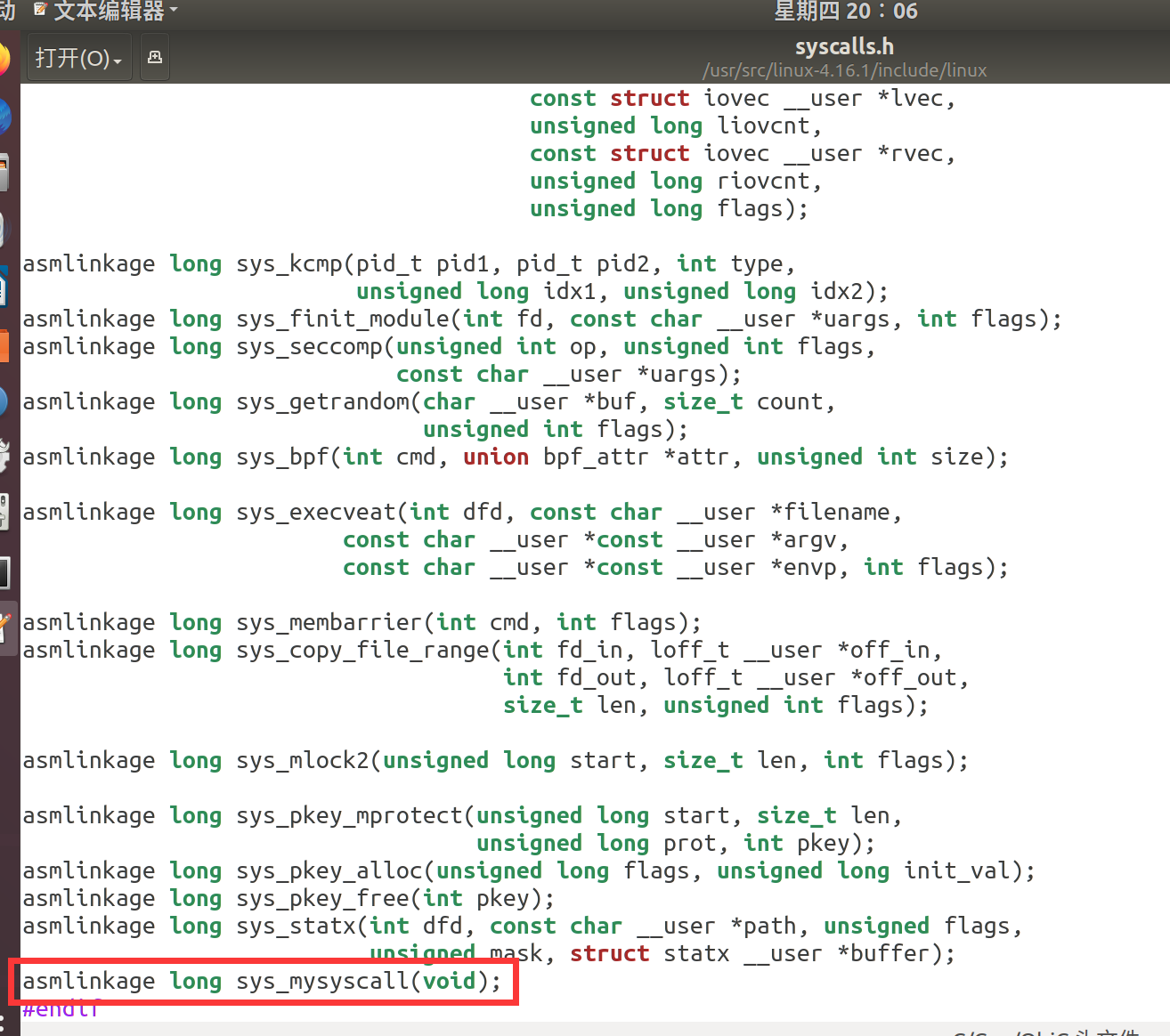
3.2添加系统调用函数声明

打开syscalls.h文件，如下命令





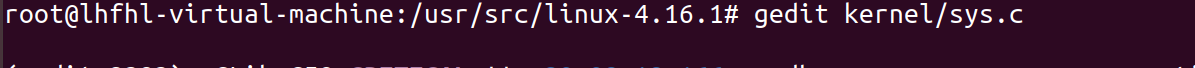
定位到最后一行，与上文一致，添加我们的函数声明，如下

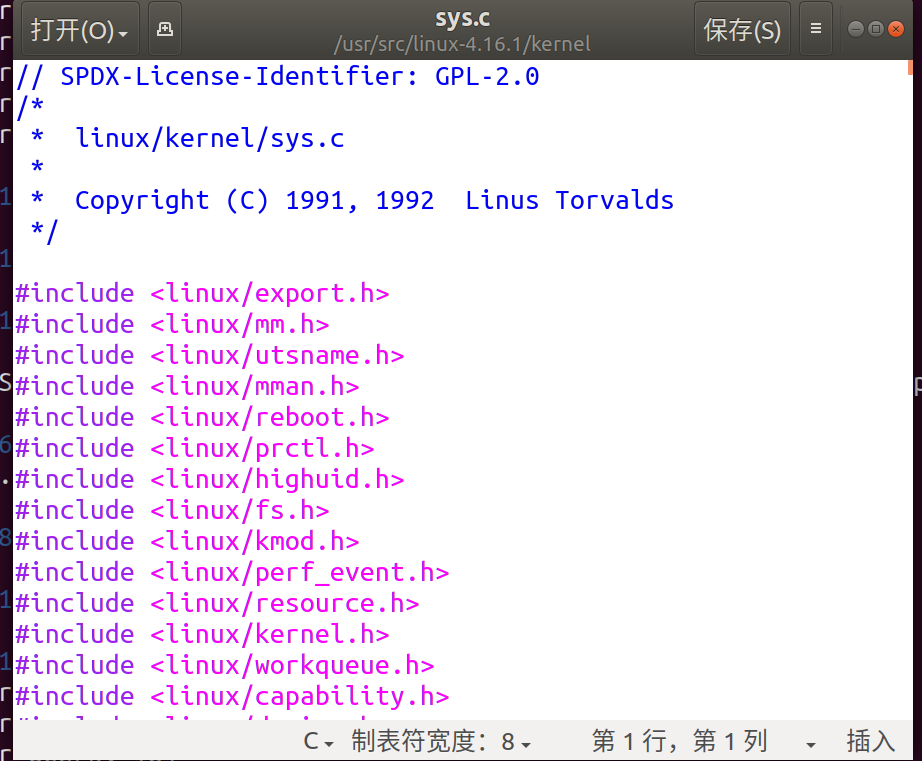


保存，退出

3.3添加系统调用的函数定义

打开sys.c文件，如下命令：





在文件末尾添加如下函数：



函数在日志文件打印一行语句，并且，如果调用成功，返回0

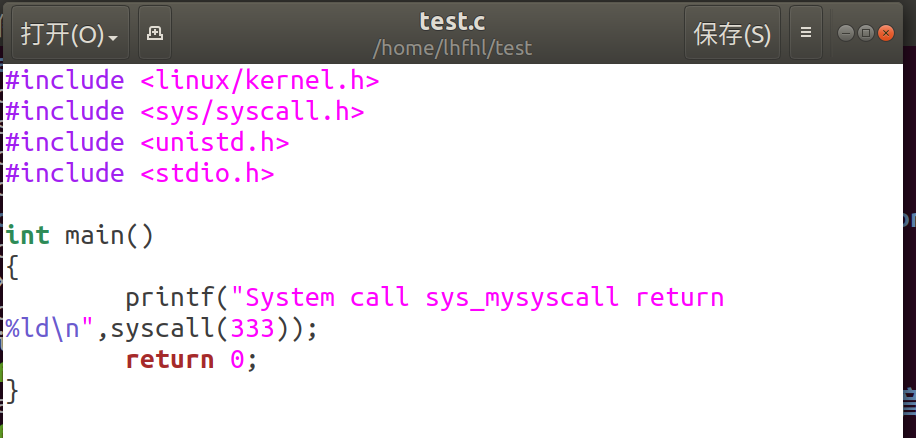
保存，退出。

3.4按照实验1-1-1重新编译安装内核，

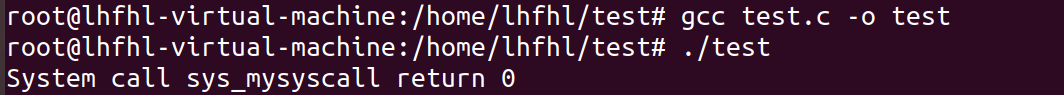
然后以重新安装的内核打开系统

3.5测试自己的系统调用

编写如下c文件：

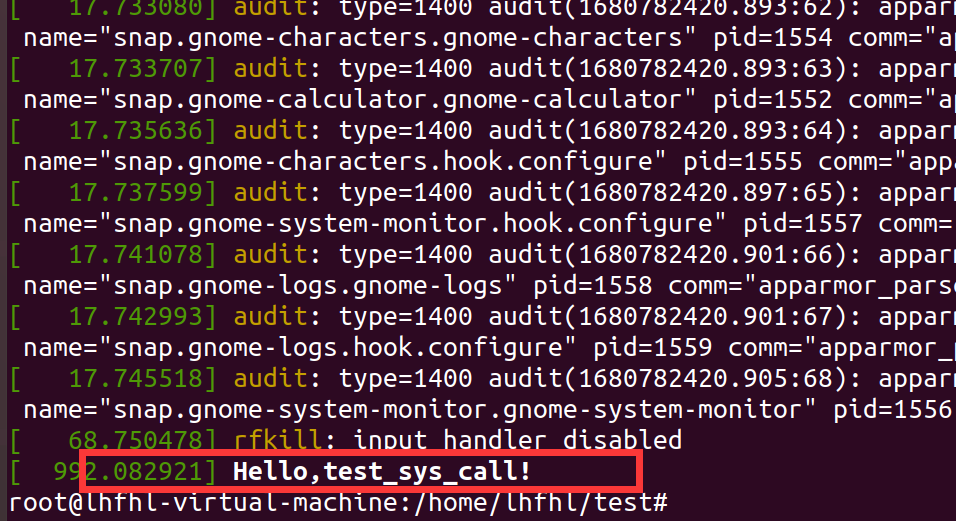
  


编译该文件，运行，可以看到系统调用返回0，说明调用成功



在输入dmesg命令，查看日志输出，可以看到我们的系统调用成功输出了：





# 实验结果及分析

通过实验步骤部分里的测试系统调用可以看到，我们添加的系统调用再重新编译安装内核后，可以成功调用了，说明实验成功。

# 五、程序代码

