西安电子科技大学

计算机学院

实验报告

题目： linux 系统调用的实现

班级： 1403013

姓名： 王鹏

学号： 14030130101

时间： 2016.10.25

理论分析

实验目的

1.了解 linux 操作系统，学习 linux 的基本操作命令，初步掌握在 linux 下的编程。
2.加深对系统功能调用的理解，理论结合实践，动手实现自己编写系统功能调用。

实验原理

系统功能调用：系统功能调用就是用户在程序中使用访管指令调用有操作系统提供的子功能集合。其中每一个子功能成为一个系统调用命令，也成为一条广义指令。按照我个人的理解，系统功能调用就是，用户在编写程序的时候，需要使用到一些涉及到有关计算机硬件系统资源，或者对于操作系统来说比较重要的内存资源等的时候，操作系统会提供一个关于这些基本资源的使用的函数的集合，作为用户，或者说编程人员，可以很方便的通过系统所提供的函数处理这些基本需求，一方面减轻了编程人员的编程实现负担，另一方面，使得程序的可移植性好，而且也不会应为编程人员的疏忽，造成系统的低效甚至奔溃。
Linux 的系统功能调用：系统调用其实就是函数调用，只不过调用的是内核态的函数，但是我们知道，用户态是不能随意调用内核态的函数的，所以采用软中断的方式从用户态陷入到内核态。在内核中通过软中断 0X80，系统会跳转到一个预设好的内核空间地址，它指向了系统调用处理程序，这里指的是在 entry.S 文件中的 system_call 函数。就是说，所有的系统调用都会统一跳转到这个地址执行 system_call 函数，那么 system_call 函数如何派发它们到各自的服务例程呢？我们知道每个系统调用都有一个系统调用号。同时，内核中一个有一个 system_call_table 数组，它是个函数指针数组，每个函数指针都指向了系统调用的服务例程。这个系统调用号是 system_call_table 的下标，用来指明到底要执行哪个系统调用。当 int ox80 的软中断执行时，系统调用号会被放进 eax 寄存器中，system_call 函数可以读取 eax 寄存器获得系统调用号，将其乘以 4 得到偏移地址，以 sys_call_table 为基地址，基地址加上偏移地址就是应该执行的系统调用服务例程的地址。

设计与实现

二、设计与实现（分值：30%）

修改 Linux 的系统调用的方法主要是修改 Linux 的内核。其修改方法为：

1、在内核源码/usr/src/linux-source-2.6.38/kernel/sys.c 中添加自定义函数

sys_my_syscall()

代码如下：

asmlinkage long sys_my_syscall(void)

```
{
    printk(KERN_INFO "Message from the simple syscall.");
}
```

2、 打开/usr/src/linux-source-2.6.38/arch/x86/kernel/syscall_table_32.S

133 common mknod sys_mknod

134 64 uselib

135 common personality sys_personality

.

.

.

323 common my_syscall sys_my_syscall (新添加的)

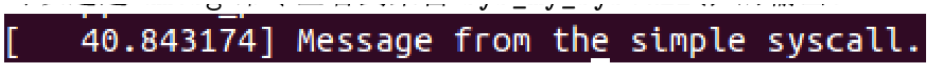
3、 安装新内核并重启系统

sudo make install

sudo make install_modules

4、测试新增的系统调用 编写测试程序，使用系统调用号的方式来测试新添加的系统调用。

- **实验结果**



- **心得与收获**

1.) 第一，开始的时候由于个人计算机上所安装的 linux 是 ubuntu16.04 和实验室的版本存在较大的出入，很多的问题，比如关于文件的路径几乎全部有改动，好不容易将其找到并且正确的处理，编译和安装内核，但是在调用自定义的系统功能调用的时候，结果总是返回-1，由此可知，调用出错，但是花了一天的时间各种方法都尝试了，还是没能够正确的调用自定义的系统功能调用，无奈只好放弃，使用实验室的版本。

通过前面几次的实践，很快就处理编译和安装内核了。并却也能够正确的调用结果正确。
2.)第二，由于对 linux 的编程有些生疏所以在买编写系统功能函数的时候，误把 printk 写成了 printf，导致编译的时候报错。
3．）至于说收获，通过这次的实验，基本明白了，操作系统的系统功能调用的概念，就是在用户态无法访问的系统资源的情况下，通过系统提供的函数来进行实现，这种方式体现了操作系统的安全和方便的特点。通过实验，也对 linux 操作系统的内核编写和内核的编译，有了一个概念，不想之前那样的模糊，抽象。也熟悉了 linux 操作系统的一些关键的命令行指令，方便快捷。总而言之，实验收获很大，但是距离目的地还是很遥远，比喻熟练使用 linux，清楚地明白 linux 的内核编程，编写更加使用能够解决实际问题的系统功能调用。所以在之后的试验中，还有课上课下会好好的学习这些知识。
4.）也希望老师，能够添加对新版内核的稍稍指导，指导手册太老旧，对于新的内核实在无法进行，然而网上网上的搜索对于新的改变也是很少，对于学生来说接触更新的版本还是有必要的，新的变动就说明具有新的优点和特性，但是出于知识的缺漏，个人解决起来实在是耗费时间，并且无功而反，丢失兴趣。