**OS project one 实验报告**

**——Adding a System call to the Linux Kernel**

**5140309201**

**黄晟**

1. **实验要求**

在linux内核中增加一个新的系统调用，并重新编译新的内核，扩展操作系统的功能。

1. **实现过程**
2. 在kernel.org网站上下载4.4.8版本的Linux内核；
3. 在/uapi/asm-generic/unistd.h中增加；



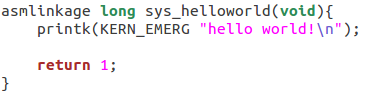
1. 在arch/x86/entry/syscalls/syscall\_64.tbl中增加新的系统调用号，为了不与内核本来的系统调用相冲突，此处定为546；



1. 在include/linux/syscalls.h中声明系统调用函数，由于此处函数必须要有参数，所以在pdf文档里的函数基础上加上参数void；



1. 在kernel/sys.c中完成系统调用函数的功能实现；



1. 完成内核编译，其步骤分别为：

make mrproper

make clean

make oldconfig

make –j8（使用多线程编译加快编译速度）

sudo make modules\_install

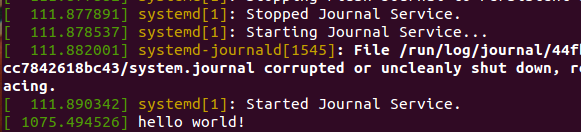
sudo make install

sudo reboot

1. **实现效果**
2. 通过调用uname -r,可以看到系统内核已升级为4.4.8；



1. 编写测试函数。在此采用了调用系统调用中最为简单的一种方法，即直接使用syscall函数，并使用之前定义的系统调用号546。在调用测试函数后，通过dmesg -c可以看到其打印出了“hello world!”。



1. **心得与体会**
2. 最初采用了书上的介绍，使用2.6版本的内核，但是由于Ubuntu15.10的gcc版本较高，无法编译2.6版本的内核，故采用了4.4.8的Linux内核；
3. 起初调用测试函数时，要调用两次才能一次性打印出两条“hello world!”，这是由于Linux是按行缓冲的，所以它只有在看到“\n”后才会打印出该行内容，所以在系统调用函数中要打印的东西后面加上“\n”即可解决该问题。