实 验 报 告

课程名称： 操作系统实验

学 院： 计算机科学与工程学院

专 业： 软件工程 班 级： 软件18-1班

姓 名：KAFLE SAMRAT学 号： 201801060933

山 东 科 技 大 学

实 验 报 告

页

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 组 别 |  | 姓 名 | KAFLE SAMRAT | 同组实验者 |  |
| 实验项目  名称 | 添加最简单的Linux内核模块 | | 实验日期 |  | |
| 教师评语 |  | | | | |
| 实验成绩： | | | 指导教师（签名）：  年 月 日 | | |

实验目的:

熟练掌握基本的Linux内核模块开发框架和编译方法。

熟练掌握Linux内核模块添加流程。

理解Linux内核模块代码中的一些常见宏和参数。

掌握Linux内核模块程序和应用程序的差异。

深入理解操作系统为用户提供服务的方式、方法

深入理解计算机程序的运行方式

实验要求:

通过阅读、执行hello.c及其对应的Makefile文件，理解Linux内核模块LKM的基本框架和运行方式、原理。

结合操作系统知识，通过实验深入理解计算机程序在操作系统支持下的运行方式。

实验内容:

从教材提供的电子资源中找到或者按教材提示自己编写简单的Linux内核模块kello.c及其对应的Makefile文件

#include<linux/module.h>

static int \_\_init hello\_start(void)

{

printk(KERN\_INFO "\n Hello kernal!This is in kernel space!\n");

return 0;

}

static void \_\_exit hello\_end(void)

{

printk(KERN\_INFO "\n Goodbye kernal! \n");

}

module\_init(hello\_start);

module\_exit(hello\_end);



obj-m = hello.o

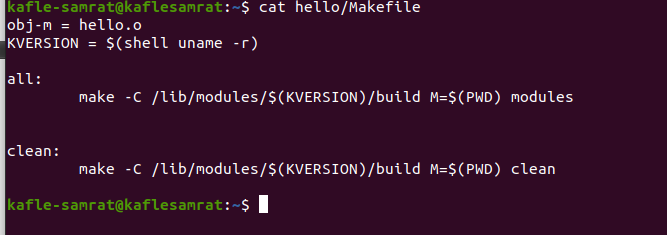
KVERSION = $(shell uname -r)

all:

make -C /lib/modules/$(KVERSION)/build M=$(PWD) modules

clean:

make -C /lib/modules/$(KVERSION)/build M=$(PWD) clean

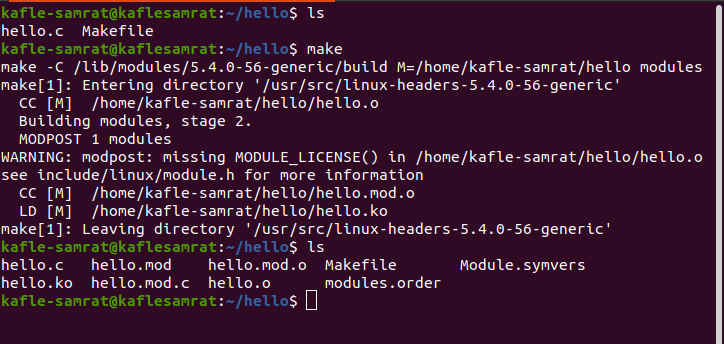


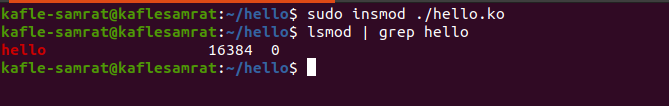
编译、安装、删除该模块，查看该模块的安装位置、运行情况

本次采用单独编译、动态插入内核；把将开发的内核代码文件直接进行编译，然后使用命令动态插入内核或者从内核卸载。

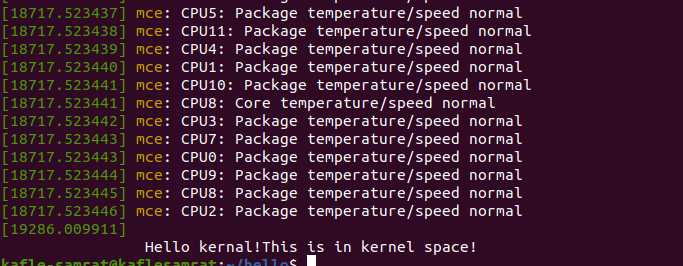
优点：编译速度快；单独调试代码

缺点：每次系统启动后都需要再加载代码

.

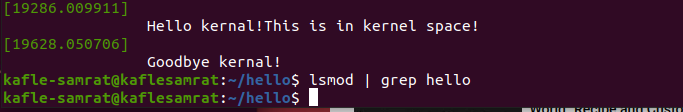


dmesg后显示:

.

删除模块:

sudo rmmod hello.ko



实验心得:

通过这次实验，我学习了基本的Linux内核模块开发框架和编译方法，熟悉了添加Linux内核模块的过程，了解了程序的工作原理。