**《操作系统》课第11次实验报告**

|  |  |
| --- | --- |
| 学院: | 软件学院 |
| 姓名: | 张怡桢 |
| 学号: | 2013747 |
| 邮箱: | 2662765987@qq.com |
| 时间: | 12/01/2022 |

1. **开篇感言**

“你长大后想成为什么人？”

“什么意思？长大后我就不能成为我自己了吗？”

-- 《阿甘正传》

2. **实验题目**

To implement a Linux kernel module

3. **实验要求**

write a c program to implement to list all processes using Linux kernel module mechanism

* 需要列出每个进程的名字(comm)、进程ID号(pid)、父进程ID号、进程状态、学号姓名等
* 统计共有多少个进程

4. **原理方法**

**An example of Linux Kernel Module**

|  |
| --- |
| Plaintext #include <linux/init.h> #include <linux/kernel.h> #include <linux/module.h> #include <linux/sched/task.h> #include <linux/sched/signal.h>  /\* init function */ static int hellokmodule\_init(void) {  printk(KERN\_ALERT "ALAL:simple module initialized\n");  printk("ALAL:Jiffies value: %lu\n", jiffies);  return 0; }  /* exit function - logs that the module is being removed \*/ static void hellokmodule\_exit(void) {  printk(KERN\_ALERT "ALAL:simple module is being unloaded\n");  print("ALAL:Jiffies value: %lu\n", jiffies); }  module\_init(hellokmodule\_init); module\_exit(hellokmodule\_exit);  MODULE\_LICENSE ("GPL"); MODULE\_AUTHOR ("LKD"); MODULE\_DESCRIPTION ("Simple Kernel Module"); MODULE\_VERSION("1.01"); |

**Compile the above Linux kernel module**

1. 创建Makefile文件:

|  |
| --- |
| Plaintext ModuleName=hellokmodule obj-m +=$${ModuleName}.o all:$${ModuleName}.ko $${ModuleName}.ko:$${ModuleName}.c  make -C /lib/modules/$$(shell uname -r)/build M=$$(PWD) modules testload:$${ModuleName}.ko  sudo dmesg -C  sudo insmod $${ModuleName}.ko  sudo dmesg | grep ALAL testunload:$${ModuleName}.ko  sudo dmesg -C  sudo rmmod $${ModuleName}.ko  sudo dmesg | grep ALAL |

1. 编译:

|  |
| --- |
| Plaintext make |

1. 执行（插入模块）:

|  |
| --- |
| Plaintext make testload |

1. 执行（卸载模块）:

|  |
| --- |
| Plaintext make testunload |

**write a c program to implement to list all processes using Linux kernel module mechanism**

* 需要列出每个进程的名字(comm)、进程ID号(pid)、父进程ID号、进程状态、学号姓名等
* 统计共有多少个进程

|  |
| --- |
| Plaintext  struct task\_struct \*p=&init\_task;  for\_each\_process(p)  {  printk("%s[%d]\n",p->comm, p->pid1);  } |

5. **实验步骤**

5.1  **Linux Kernel Module**

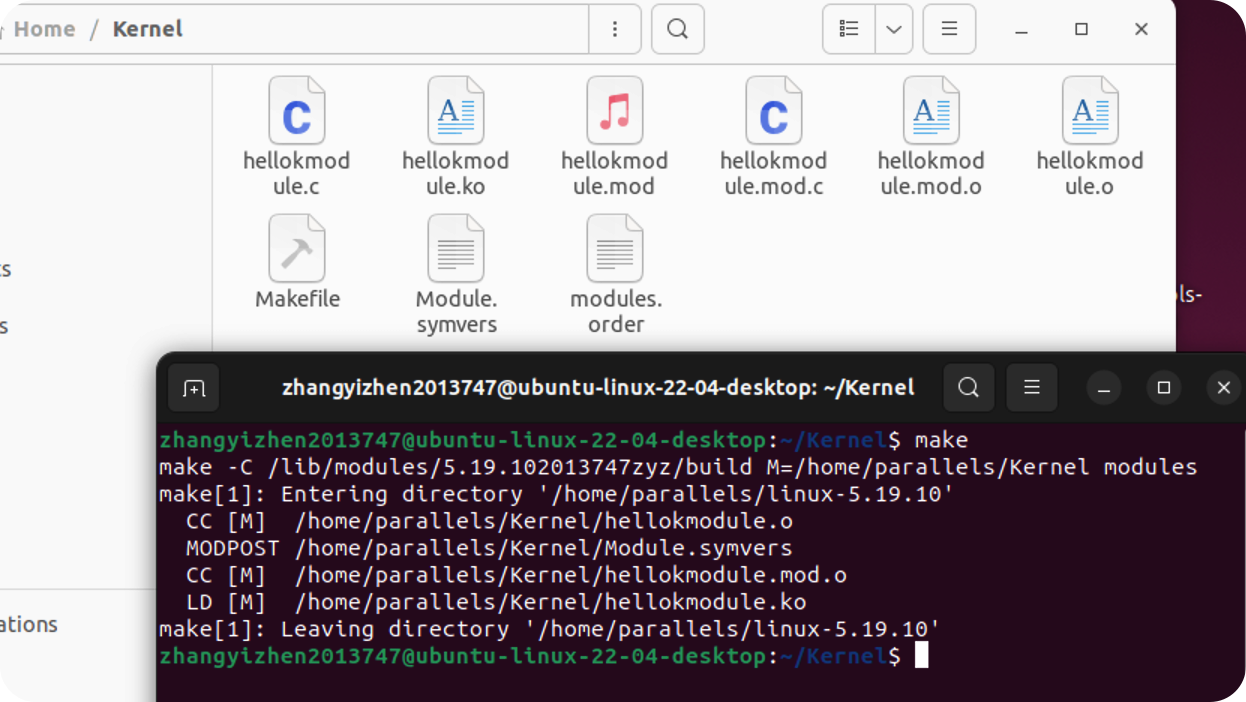
|  |
| --- |
| C #include <linux/init.h> #include <linux/kernel.h> #include <linux/module.h> #include <linux/sched/task.h> #include <linux/sched/signal.h>  /\* init function \*/ static int hellokmodule\_init(void) {  struct task\_struct \*p=&init\_task;  printk(KERN\_ALERT "2013747:simple module initialized\n");  printk("2013747:Jiffies value: %lu\n", jiffies);  //准备打印进程信息  printk("Hello new system call schello! 2013747 Zhang Yizhen\n");  printk("%-20s %-6s %-11s %-6s %-10s\n","Name","Pid","Parent Pid","Stat","2013747");  for\_each\_process(p){  printk("%-20s %-6d %-11d %-6c %-10s\n",p->comm,p->pid,p->parent->pid,task\_state\_to\_char(p),"2013747");  }  printk("2013747 Zhang Yizhen\n");   return 0; }  /\* exit function - logs that the module is being removed \*/ static void hellokmodule\_exit(void) {  printk(KERN\_ALERT "2013747:simple module is being unloaded\n");  printk("2013747:Jiffies value: %lu\n", jiffies); }  module\_init(hellokmodule\_init); module\_exit(hellokmodule\_exit);  MODULE\_LICENSE ("GPL"); MODULE\_AUTHOR ("LKD"); MODULE\_DESCRIPTION ("Simple Kernel Module"); MODULE\_VERSION("1.01"); |

5.2  **Compile the above Linux kernel module**

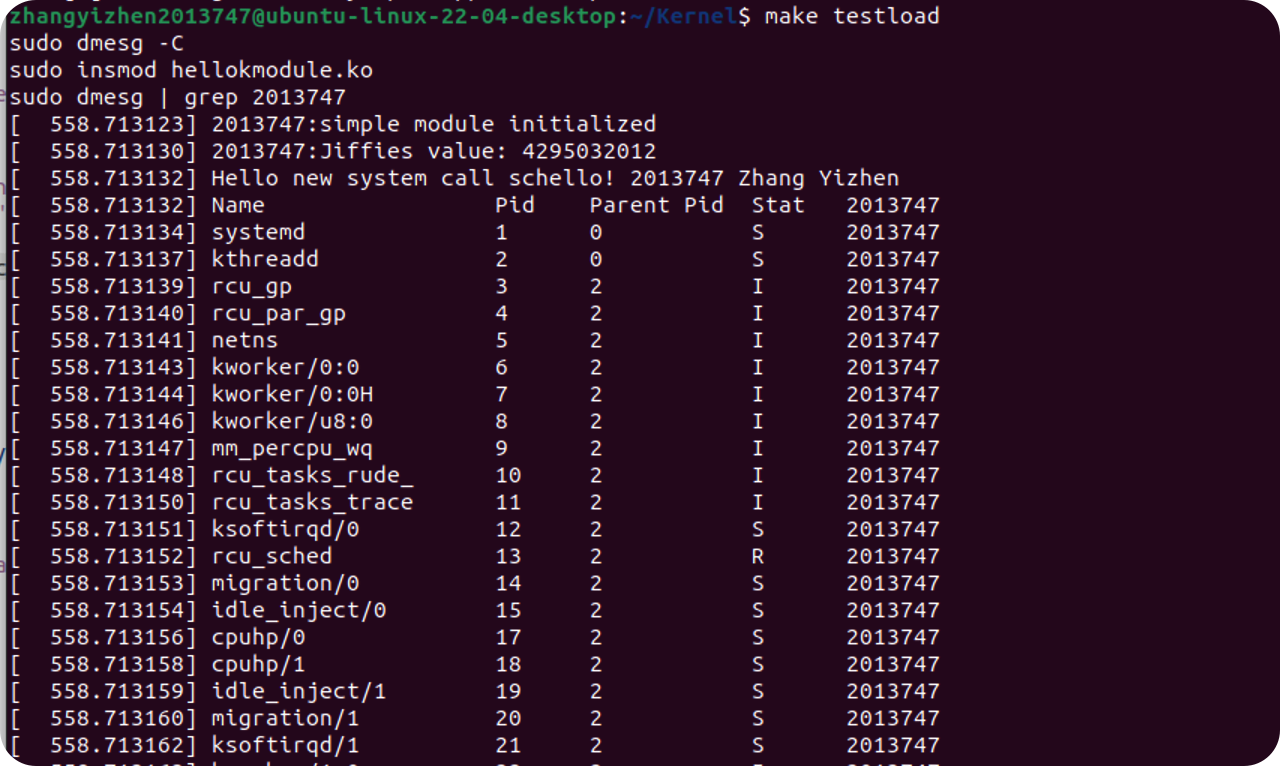
|  |
| --- |
| Makefile ModuleName=hellokmodule obj-m +=${ModuleName}.o all:${ModuleName}.ko ${ModuleName}.ko:${ModuleName}.c  make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules testload:${ModuleName}.ko  sudo dmesg -C  sudo insmod ${ModuleName}.ko  sudo dmesg | grep 2013747 testunload:${ModuleName}.ko  sudo dmesg -C  sudo rmmod ${ModuleName}.ko  sudo dmesg | grep 2013747 |

5.3 **编译并实现**

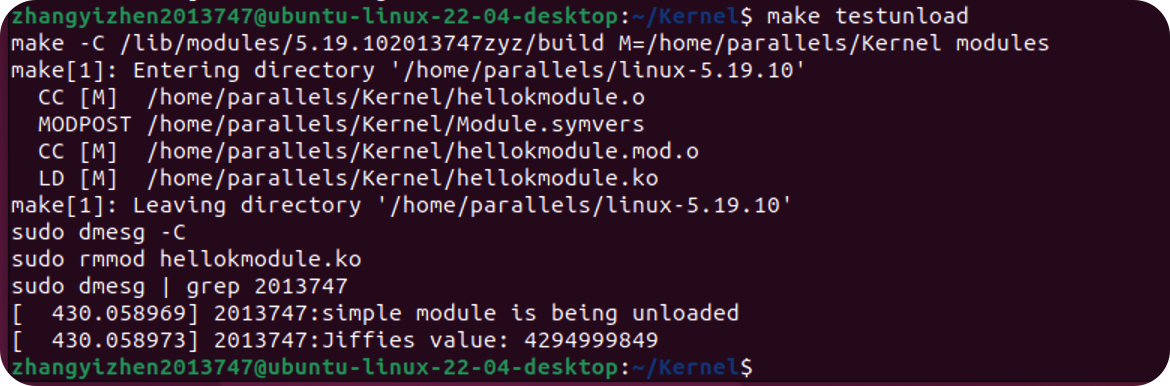
**make**



**make testload**



**make testunload**



6. **资料**

[老师的github实验文档](https://github.com/albertleecn/osplab)

7. **附件**

📎代码在压缩包中