

# Project1 实验报告

---

余北辰 519030910245

## # 1 实验概述

---

### 1.1 实验名称

Introduction to Linux Kernel Modules

### 1.2 实验内容

1. 创建simple模块，实现模块的加载与卸载
2. 创建hello模块，实现在/proc文件系统读写信息
3. 创建jiffies模块，获取jiffies的值
4. 创建seconds模块，通过简单计算，获取首次加载模块后经过的秒数

## # 2 实验环境

---

- Ubuntu 18.04.5 LTS
- Linux version 5.4.0-72-generic
- VirtualBox 6.1.18

## # 3 实验过程与结果展示

---

### 3.1 simple模块

该部分的要求为：

1. Print out the value of `GOLDEN_RATIO_PRIME` in the simple `init()` function.
2. Print out the greatest common divisor of 3,300 and 24 in the simple `exit()` function.
3. Print out the values of `jiffies` and `HZ` in the simple `init()` function.
4. Print out the value of `jiffies` in the simple `exit()` function.

`simple_init()` 函数如下：

```

1  int simple_init(void)
2  {
3      printk(KERN_INFO "Loading Kernel Module\n");
4      printk(KERN_INFO "The Golden ratio prime is equal to %lu\n",
        GOLDEN_RATIO_PRIME);
5      printk(KERN_INFO "The value of jiffies is %lu\n", jiffies);
6      printk(KERN_INFO "The value of HZ is %lu\n", HZ);
7      return 0;
8  }

```

`simple_exit()` 函数如下：

```

1  void simple_exit(void)
2  {
3      printk(KERN_INFO "Removing Kernel Module\n");
4      printk(KERN_INFO "The gcd of 3300 and 24 is %lu\n", gcd(3300, 24));
5      printk(KERN_INFO "The value of jiffies is %lu\n", jiffies);
6  }

```

注意因为涉及到 `jiffies` 的计算和最大公约数的运算，需要 `#include <linux/gcd.h>` 以及 `#include <linux/jiffies.h>`。

使用 `sudo insmod simple.ko` 命令，加载 `simple` 模块；

使用 `sudo rmmod simple` 命令，移除 `simple` 模块；

使用 `dmesg` 命令，检查内核日志缓冲区的信息；

运行结果如下所示：

```

polaris@polaris-VirtualBox: ~/course/Operating-Systems/Project/Project1/simple
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ sudo insmod simple.ko
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ dmesg
[88843.353704] Loading Kernel Module
[88843.353729] The Golden ratio prime is equal to 7046029254386353131
[88843.353730] The value of jiffies is 4317109779
[88843.353730] The value of HZ is 250
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ sudo dmesg -c
[88843.353704] Loading Kernel Module
[88843.353729] The Golden ratio prime is equal to 7046029254386353131
[88843.353730] The value of jiffies is 4317109779
[88843.353730] The value of HZ is 250
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ sudo rmmod simple
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ dmesg
[88902.220936] Removing Kernel Module
[88902.220938] The gcd of 3300 and 24 is 12
[88902.220938] The value of jiffies is 4317124496
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ sudo dmesg -c
[88902.220936] Removing Kernel Module
[88902.220938] The gcd of 3300 and 24 is 12
[88902.220938] The value of jiffies is 4317124496
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/simple$ |

```

## 3.2 hello模块

该部分非实验作业内容，而是属于实验的教学部分。

该部分的要求为：create a new entry in the /proc file system. If a user enters the command `cat /proc/hello`, the infamous `Hello World` message is returned.

`proc_init()` 与 `proc_exit()` 代码如下：

```
1  int proc_init(void)
2  {
3      proc_create(PROC_NAME, 0, NULL, &proc_ops);
4
5      printk(KERN_INFO "/proc/%s created\n", PROC_NAME);
6
7      return 0;
8  }
9
10 void proc_exit(void) {
11
12     remove_proc_entry(PROC_NAME, NULL);
13
14     printk( KERN_INFO "/proc/%s removed\n", PROC_NAME);
15 }
16
```

`proc_read()` 代码如下：

```
1  ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos)
2  {
3      int rv = 0;
4      char buffer[BUFFER_SIZE];
5      static int completed = 0;
6
7      if (completed) {
8          completed = 0;
9          return 0;
10     }
11
12     completed = 1;
13
14     rv = sprintf(buffer, "Hello World\n");
15
16     // copies the contents of buffer to userspace usr_buf
17     copy_to_user(usr_buf, buffer, rv);
18
19     return rv;
20 }
21
```

运行结果如下：

```
polaris@polaris-VirtualBox: ~/course/Operating-Systems/Project/Project1/hello
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$ sudo insmod hello.ko
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$ dmesg
[ 94557.401363] /proc/hello created
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$ cat /proc/hello
Hello World
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$ sudo rmmod hello
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$ dmesg
[ 94557.401363] /proc/hello created
[ 94591.008418] /proc/hello removed
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/hello$
```

### 3.3 jiffies模块

该部分的要求为：

creates a /proc file named /proc/jiffies that reports the current value of jiffies when the /proc/jiffies file is read

这与hello模块类似， `proc_init()` 与 `proc_exit()` 代码与hello模块中相同。

对hello模块中的 `proc_read()` 函数进行修改，使其读取jiffies的值：

```
1  ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos)
2  {
3      int rv = 0;
4      char buffer[BUFFER_SIZE];
5      static int completed = 0;
6
7      if (completed) {
8          completed = 0;
9          return 0;
10     }
11
12     completed = 1;
13
14     rv = sprintf(buffer, "%lu\n", jiffies);
15
16     // copies the contents of buffer to userspace usr_buf
17     copy_to_user(usr_buf, buffer, rv);
18
19     return rv;
```

运行结果如下：

```
polaris@polaris-VirtualBox: ~/course/Operating-Systems/Project/Project1/jiffies
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ sudo insmod jiffies.ko
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ dmesg
[94945.277654] /proc/jiffies created
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ cat /proc/jiffies
4318641730
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ sudo rmmod jiffies
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ dmesg
[94945.277654] /proc/jiffies created
[94982.593303] /proc/jiffies removed
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/jiffies$ |
```

### 3.4 seconds模块

该部分的要求为：

creates a proc file named `/proc/seconds` that reports the number of elapsed seconds since the kernel module was loaded.

根据提示，我们知道用jiffies之差除以HZ就是所求的秒数了。

`proc_init()` 实现的时候，用变量  $T_0$  记录当前jiffies值：

```
1 int proc_init(void)
2 {
3     proc_create(PROC_NAME, 0, NULL, &proc_ops);
4     T0 = jiffies;
5     printk(KERN_INFO "/proc/%s created\n", PROC_NAME);
6
7     return 0;
8 }
```

的 `proc_exit()` 代码与hello模块中相同。

对hello模块中的 `proc_read()` 函数进行修改：

```
1 ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos)
2 {
3     int rv = 0;
4     char buffer[BUFFER_SIZE];
5     static int completed = 0;
6
7     if (completed)
8     {
```

```
9         completed = 0;
10        return 0;
11    }
12
13    completed = 1;
14
15    rv = sprintf(buffer, "%lu\n", (jiffies - T0) / HZ);
16
17    // copies the contents of buffer to userspace usr_buf
18    copy_to_user(usr_buf, buffer, rv);
19
20    return rv;
21 }
```

运行结果如下：

```
polaris@polaris-VirtualBox: ~/course/Operating-Systems/Project/Project1/seconds
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ sudo insmod seconds.ko
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ dmesg
[95476.578795] /proc/seconds created
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ cat /proc/seconds
22
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ cat /proc/seconds
31
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ cat /proc/seconds
39
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ sudo rmmod seconds
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$ dmesg
[95476.578795] /proc/seconds created
[95527.194185] /proc/seconds removed
polaris@polaris-VirtualBox:~/course/Operating-Systems/Project/Project1/seconds$
```

## # 4 实验总结

---

1. 经过学习网上的资料，成功安装虚拟机并配置好实验所需的linux环境
2. 本次实验较为顺利，没有碰到太多的难点
3. 以此次实验为契机，学习了一些基本的命令行操作
4. 以此次实验为契机，学习了Makefile文件的编写

## # 5 实验参考资料

---

- 实验参考书籍：Operating System Concept, 10<sup>th</sup> edition
- 实验源代码网址：<https://github.com/greggagne/osc10e>

