EI338 Project 1

`516030910510 潘佳萌`

1. Jiffies

Question

Design a kernel module that creates a `/proc` file named `/proc/jiffies` that reports the current value of jiffies when the `/proc/jiffies` file is read, such as with the command

> cat /proc/jiffies

Be sure to remove `/proc/jiffies` when the module is removed.

Answer

This question requires the jiffies, so the `<linux/jiffies.h>` package needs to be include firstly. Then with the command `cat /proc/jiffies`, the jiffies output. In the `proc_read` function, the `sprintf` function should print `jiffies`.

proc_init()

In this function, `proc_create` creates the new `/proc/jiffies`, and passed `proc_ops`, which contains a reference to a struct `file-operations`. This struct initializes the `.owner` and `.read` members. The value of `.read` is the name of the function `proc_read()` that is to be called whenever /proc/hello is read.

proc_read()

In this function, we see that the unsigned long int `jiffies` is written to the variable buffer where buffer exists in kernel memory. Since `/proc/jiffies` can be accessed from user space, we must copy the contents of buffer to user space using the kernel function `copy_to_user()`. This function copies the contents of kernel memory buffer to the variable `usr_buf`, which exists in user space. Each time the `proc/hello` file is read, the `proc_read()` function is called repeatedly until it returns 0, so there must be logic to ensure that this function returns 0 once it has collected the data that is to go into the corresponding `/proc/jiffies` file.

proc_exit()

`/proc/hello` file is removed in the module exit point `proc_exit()` using the function `remove_proc_entry()`.

makefile

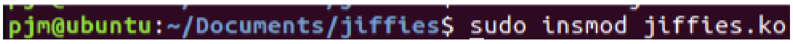
```
obj-m:=jiffies.o
jiffiesmodule-objs:=module
KDIR:=/lib/modules/$(shell uname -r)/build
MAKE:=make
default:
    $(MAKE) -C $(KDIR) SUBDIRS=$(PWD) modules
clean:
    $(MAKE) -C $(KDIR) SUBDIRS=$(PWD) clean
```

`obj-m=*.o` generates `*.ko` file, while `obj-y=*.o` only compiles the code into kernel.

Command line

1. Loading kernel

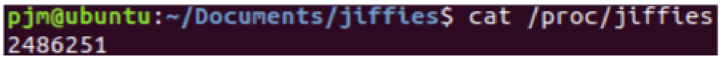
> sudo insmod jiffies.ko



```
pjm@ubuntu:~/Documents/jiffies$ sudo insmod jiffies.ko
```

2. Print jiffies

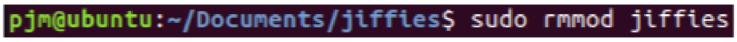
> cat /proc/jiffies



```
pjm@ubuntu:~/Documents/jiffies$ cat /proc/jiffies
2486251
```

3. Removing kernel

> sudo rmmod jiffies



```
pjm@ubuntu:~/Documents/jiffies$ sudo rmmod jiffies
```

2 Second

Question

Design a kernel module that creates a proc file named `/proc/seconds` that reports the number of elapsed seconds since the kernel module was loaded. This will involve using the value of `jiffies` as well as the HZ rate. When a user enters the command

> cat /proc/seconds

your kernel module will report the number of seconds that have elapsed since the kernel module was first loaded. Be sure to remove `/proc/seconds` when the module is removed.

Answer

Similar to 1, the diffience is this question needs a global variable to record the jiffies in the `proc_init` function.

proc_init()

t is the global variable.

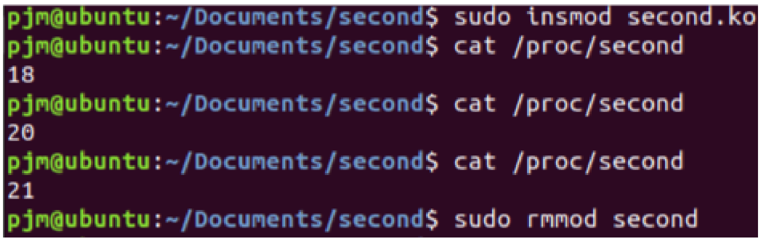
```
{
/* creates the /proc/second entry */
    proc_create(PROC_NAME, 0666, NULL, &proc_ops);
    t = jiffies;
    return 0;
}
```

proc_read()

`HZ` is the varible in `<linux/param.h>`

```
ssize_t proc_read(struct file *file, char __user *usr_buf,
size_t count, loff_t *pos)
{
    int rv = 0;
    char buffer[BUFFER_SIZE];
    static int completed = 0;
    if (completed) {
        completed = 0;
        return 0;
    }
    completed = 1;
    rv = sprintf(buffer, "%lu\n", (jiffies-t)/HZ);
    /* copies kernel space buffer to user space usr buf */
    copy_to_user(usr_buf, buffer, rv);
    return rv;
}
```

Command line



```
pjm@ubuntu:~/Documents/second$ sudo insmod second.ko
pjm@ubuntu:~/Documents/second$ cat /proc/second
18
pjm@ubuntu:~/Documents/second$ cat /proc/second
20
pjm@ubuntu:~/Documents/second$ cat /proc/second
21
pjm@ubuntu:~/Documents/second$ sudo rmmod second
```

In the command line, we can see that the second is the time since `second` is loaded