Lab 7 Lab Report: CprE 308

Matt McKillip

1. Introduction

Lab 7 introduces us to the inner workings of the Linux Kernel. To help us understand the what is going on, we are asked to download and compile LInux-3.18.8.

2. Questions

2.1 Linux Kernel

2.1.1 List five architectures that the Linux kernel supports

1. x86
2. ia64
3. m32r
4. powerpc
5. arm64

2.1.1 List three filesystem types Linux supports

1. fat
2. ext4
3. nfs

six operations that files can support.

1. read
2. write
3. readdir
4. poll
5. mmap
6. open

Comment on why the prototypes are not identical and what the extra paramaters are for

The added parameters are offset and count. Count is needed to determine the number of bytes the program will read and the offset will offset the memory location. These are need since the file is using a FILE instead of a filedescriptor

3. Results/Output

Hello-world

```
root@matt:/host/cpre308/labs/Lab7/hello-world# lsmod
Module                  Size  Used by
root@matt:/host/cpre308/labs/Lab7/hello-world# modinfo hello_world.ko
filename:       /host/cpre308/labs/Lab7/hello-world/hello_world.ko
description:    A simple hello world driver
author:         Jeramie Vens vens@iastate.edu
license:        GPL
depends:
vermagic:       3.18.8mamckill mod_unload
root@matt:/host/cpre308/labs/Lab7/hello-world# insmod hello_world.ko
Hello World from kernel!
root@matt:/host/cpre308/labs/Lab7/hello-world# lsmod
Module                  Size  Used by
hello_world              811  0
root@matt:/host/cpre308/labs/Lab7/hello-world# rmmod hello_world.ko
Goodbye World from kernel!
root@matt:/host/cpre308/labs/Lab7/hello-world# lsmod
Module                  Size  Used by
root@matt:/host/cpre308/labs/Lab7/hello-world#
```

hello-file

Expected output when the file is opened, written to, read from, and closed

opened

When the file is opened it will print the lines in __init device_init such as “I was assigned….”

read

The output when the the file is read from will be the “i already told you __ time Hello World” from device_open

written

Since device_write is not implemented, the output will be “sorry, this operation is not supported”

closed

When the file is closed no output will happen since there is no printk statements in device_release

actual

```
root@matt:/host/cpre308/labs/Lab7/hello-file# insmod hello_file.ko
hello_file: module license 'unspecified' taints kernel.
Disabling lock debugging due to kernel taint
I was assigned major number 254. To talk to
the driver, create a dev file with
'mknod /dev/cpre308 c 254 0'.
Try various minor numbers. Try to cat and echo to
the device file.
root@matt:/host/cpre308/labs/Lab7/hello-file# mknod /dev/cpre308-0 c 254 0
mknod: ‘/dev/cpre308-0’: File exists
root@matt:/host/cpre308/labs/Lab7/hello-file# cat /dev/cpre308-0
I already told you 0 times Hello World!
root@matt:/host/cpre308/labs/Lab7/hello-file# cat /dev/cpre308-0
I already told you 1 times Hello World!
root@matt:/host/cpre308/labs/Lab7/hello-file# echo "hello" > /dev/cpre308
root@matt:/host/cpre308/labs/Lab7/hello-file# mknod /dev/cpre308-1 c 254 1
root@matt:/host/cpre308/labs/Lab7/hello-file# cat /dev/cpre308-0
I already told you 2 times Hello World!
root@matt:/host/cpre308/labs/Lab7/hello-file# cat /dev/cpre308-1
I already told you 3 times Hello World!
root@matt:/host/cpre308/labs/Lab7/hello-file# rmmod hello_file.ko
root@matt:/host/cpre308/labs/Lab7/hello-file# cat /dev/cpre308-0
cat: /dev/cpre308-0: No such device or address
root@matt:/host/cpre308/labs/Lab7/hello-file#
```

4. Design Decision

My design is very similar to the device_read. What needs to be changed is the instead of put_user I used get_usser. The other change was I checked each character for a new line character, to print the first line and return.

5. Issues

I had a lot of issues getting the linux kernel to compile and run. I am not sure why I had this issue since I followed the directions. This was where most of my time on this lab was spent.

6. Conclusion

I was happy with my Lab7 pre-lab and implementation of printer-driver. This lab was a nice break from the long lab5 and lab6. After this lab I have a better understanding of how linux kernels work.

7. Suggestions

I liked that this lab was not depended on lab5 and lab6, or else it would be really tough to get done.