

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

Лабораторная работа №2

по дисциплине

Операционные системы

Выполнил:
Студент группы Р33302
Ершова А. И.

Преподаватель:
Осипов С. В.

Санкт-Петербург, 2022

Вариант: debugfs: inode, pci_dev

Задание:

Лабораторная работа №2

Разработать комплекс программ на пользовательском уровне и уровне ядра, который собирает информацию на стороне ядра и передает информацию на уровень пользователя, и выводит ее в удобном для чтения человеком виде. Программа на уровне пользователя получает на вход аргумент(ы) командной строки (не адрес!), позволяющие идентифицировать из системных таблиц необходимый путь до целевой структуры, осуществляет передачу на уровень ядра, получает информацию из данной структуры и распечатывает структуру в стандартный вывод. Загружаемый модуль ядра принимает запрос через указанный в задании интерфейс, определяет путь до целевой структуры по переданному запросу и возвращает результат на уровень пользователя.

Интерфейс передачи между программой пользователя и ядром и целевая структура задается преподавателем. Интерфейс передачи может быть один из следующих:

1. syscall - интерфейс системных вызовов.
2. ioctl - передача параметров через управляющий вызов к файлу/устройству.
3. procfs - файловая система /proc, передача параметров через запись в файл.
4. debugfs - отладочная файловая система /sys/kernel/debug, передача параметров через запись в файл.

Целевая структура может быть задана двумя способами:

1. Именем структуры в заголовочных файлах Linux
2. Файлом в каталоге /proc. В этом случае необходимо определить целевую структуру по пути файла в /proc и выводимым данным.

Код:

<https://github.com/efobo/info-from-kernel-module>

Состав Makefile:

make – сборка файлов модуля ядра и пользовательской программы

make clean – удаление исполняемых файлов

make test – загрузка модуля в ядро

make close – выгрузка модуля из ядра

Порядок запуска:

make

make test

sudo ./user_prog <file_path> <vendor_id> <device_id>

Вывод программы:

```
anna@anna-VirtualBox:~/os2$ sudo ./user_prog file.txt -1 -1
Writing path to file.
Getting data from file.
Read chars count = 504.
```

```
Inode:
Inode number = 402131
Hard link count = 1
Size = 0 bytes
```

```
PCI_device:
```

```
Device #1:
Name 0000:00:00.0
Vendor ID = 32902, Device ID = 4663
Revision ID = 2
Interrupt Line = 0
Latency Timer = 0
Command = 0
```

```
Device #2:
Name 0000:00:01.0
Vendor ID = 32902, Device ID = 28672
Revision ID = 0
Interrupt Line = 0
Latency Timer = 0
Command = 7
```

```
Device #3:
Name 0000:00:01.1
Vendor ID = 32902, Device ID = 28945
Revision ID = 1
Interrupt Line = 0
Latency Timer = 64
Command = 7
```

```
Device #4:
Name 0000:00:02.0
Vendor ID = 5549, Device ID = 1029
Revision ID = 0
Interrupt Line = 10
Latency Timer = 64
Command = 7
```

```
Device #5:
Name 0000:00:03.0
Vendor ID = 32902, Device ID = 4110
Revision ID = 2
Interrupt Line = 9
Latency Timer = 64
Command = 7

Device #6:
Name 0000:00:04.0
Vendor ID = 33006, Device ID = 51966
Revision ID = 0
Interrupt Line = 11
Latency Timer = 0
Command = 3

Device #7:
Name 0000:00:05.0
Vendor ID = 32902, Device ID = 9237
Revision ID = 1
Interrupt Line = 11
Latency Timer = 64
Command = 5
```

```
Device #8:
Name 0000:00:06.0
Vendor ID = 4203, Device ID = 63
Revision ID = 0
Interrupt Line = 10
Latency Timer = 64
Command = 6

Device #9:
Name 0000:00:07.0
Vendor ID = 32902, Device ID = 28947
Revision ID = 8
Interrupt Line = 9
Latency Timer = 0
Command = 1

Device #10:
Name 0000:00:0b.0
Vendor ID = 32902, Device ID = 9820
Revision ID = 0
Interrupt Line = 9
Latency Timer = 64
Command = 22
```

Вывод:

Во время выполнения лабораторной работы я углубилась в работу ядра Linux. Собрала информацию сс структур `inode` и `pci_dev` и вывела на стороне пользователя.