Project 4

CS342 - Operating Systems

Efe Acer - 21602217

Yusuf Dalva - 21602867

Bilkent University, CS

Contents

1 Introduction 2

2 Developing a Kernel Module 2

3 Accessing Kernel Structures 2

4 Carrying out an Experiment 2

5 Experiment Results 4

1 Introduction

The aim of this project is to access certain Linux Kernel in-memory structures and retrieve some information related to files and file systems. To do this, a Linux Kernel module is developed. This module is installed into the Kernel using the `insmod` command and cleared with the `rmmod` command. This procedure is the most common way of adding functionality to the operating system's core.

2 Developing a Kernel Module

Kernel code is still implemented in `c` but it has a different structure than regular `c` files. There are certain constructs and macros that are required to write a kernel module. We followed these kernel development guidelines throughout the implementation of the project.

3 Accessing Kernel Structures

As mentioned earlier, we accessed many in-memory structures in order to retrieve information about the files and the file system. For this, we investigated and studied the Linux Kernel header files. Those headers were all open source, so they were easily available online. With a little bit of guidance from the project document and the help of the Kernel documentation, we succeeded to find the appropriate structures. These structures were mainly, file `inode`s, dentrys (directory entries) and `buffer_head`s (buffer caches).

4 Carrying out an Experiment

To check whether our kernel module prints the necessary information correctly or not, we implemented an experimental test program named `app.c`. The code inside `app.c` is provided below:

```c
 /**
 * CS342 Spring 2019 - Project 4
 * A simple application programs that creates, writes and reads files.
 * @author Yusuf Dalva - 21602867
 * @author Efe Acer - 21602217
 */

// Necessary include(s)
#include <stdio.h>
#include <unistd.h>
```

```

int main() {
	printf("Application started.\n");

	char byte_str[] = "BYTE"; // 4 bytes

	char* file1 = "file1.txt";
	FILE* fp1 = fopen(file1, "w+b");
	fwrite(byte_str, 1, sizeof(byte_str) - 1, fp1);
	char str_in1[sizeof(byte_str)];
	fclose(fp1); // restore the pointer
	fp1 = fopen(file1, "r");
	if (fgets(str_in1, sizeof(byte_str), fp1)) {
		printf("File 1 content: %s\n", str_in1);
	}

	char* file2 = "file2.txt";
	FILE* fp2 = fopen(file2, "w+b");
	char str2[] = "This is the second file created by the application.";
	for (int i = 0; i < 1000 * 5; i++) { // will fill up 5 pages
		fwrite(byte_str, 1, sizeof(byte_str) - 1, fp2);
	}
	fwrite(str2, 1, sizeof(byte_str) - 1, fp2);
	char str_in2[sizeof(byte_str) * 1000 * 5];
	fclose(fp2); // restore the pointer
	fp2 = fopen(file2, "r");
	if (fgets(str_in2, sizeof(byte_str) * 1000 * 5, fp2)) {
		printf("File 2 content: %s\n", str_in2);
	}

	char* file3 = "file3.txt";
	FILE* fp3 = fopen(file3, "w+b");
	for (int i = 0; i < 1000 * 10; i++) { // will fill up 10 pages
		fwrite(byte_str, 1, sizeof(byte_str) - 1, fp3);
	}
	char str_in3[sizeof(byte_str) * 1000 * 10];
	fclose(fp3); // restore the pointer
	fp3 = fopen(file3, "r");
	if (fgets(str_in3, sizeof(byte_str) * 1000 * 10, fp3) != NULL) {
		printf("File 3 content: %s\n", str_in3);
	}

	printf("process identifier: %d\n", getpid());
	printf("Application is infinite looping...(terminate yourself)\n");
```

```
        while(1);
        return 0;
}
```

The `app.c` program is designed in a way such that it creates three different files, populates the first file with information that fits into a single block, the second file with information that fits into 5 blocks and the third file with information that fits into 10 blocks. The program prints its process identifier and starts to loop infinitely instead of terminating. This is done to allow the kernel module to receive the process identifier of `app.c` and for `app.c` to continue using the files it created without closing them.

5 Experiment Results

We run our kernel module with the `insmod` command giving the process identifier of `app.c` as an argument. Then the kernel module logged the following lines:

```
 May 25 14:50:51 efe-VirtualBox kernel: [24712.748353] project4_EA_YD module is started.
May 25 14:50:51 efe-VirtualBox kernel: [24712.748355]
May 25 14:50:51 efe-VirtualBox kernel: [24712.748355] Input process identifier: 7334
May 25 14:50:51 efe-VirtualBox kernel: [24712.748356] Information about the file descriptors and open files of
    the process:
May 25 14:50:51 efe-VirtualBox kernel: [24712.748356]
May 25 14:50:51 efe-VirtualBox kernel: [24712.748357] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748357] Descriptor Number: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748358] Current File Position Pointer: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748358] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748359] Process Access Mode: 393219
May 25 14:50:51 efe-VirtualBox kernel: [24712.748359] Name of the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748360] inode Number of the File: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748360] File Length in Bytes: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748361] Number of Blocks Allocated to the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748361] Number of Blocks (Pages) Cached: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748362] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748362] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748363] Descriptor Number: 1
May 25 14:50:51 efe-VirtualBox kernel: [24712.748363] Current File Position Pointer: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748363] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748364] Process Access Mode: 393219
May 25 14:50:51 efe-VirtualBox kernel: [24712.748364] Name of the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748365] inode Number of the File: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748365] File Length in Bytes: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748365] Number of Blocks Allocated to the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748366] Number of Blocks (Pages) Cached: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748366] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748366] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748367] Descriptor Number: 2
May 25 14:50:51 efe-VirtualBox kernel: [24712.748367] Current File Position Pointer: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748368] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748368] Process Access Mode: 393219
May 25 14:50:51 efe-VirtualBox kernel: [24712.748368] Name of the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748369] inode Number of the File: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748369] File Length in Bytes: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748370] Number of Blocks Allocated to the File: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748370] Number of Blocks (Pages) Cached: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748370] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748371] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748371] Descriptor Number: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748372] Current File Position Pointer: 4
May 25 14:50:51 efe-VirtualBox kernel: [24712.748372] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748372] Process Access Mode: 134381597
May 25 14:50:51 efe-VirtualBox kernel: [24712.748373] Name of the File: file1.txt
May 25 14:50:51 efe-VirtualBox kernel: [24712.748374] inode Number of the File: 1537
May 25 14:50:51 efe-VirtualBox kernel: [24712.748374] File Length in Bytes: 4
May 25 14:50:51 efe-VirtualBox kernel: [24712.748375] Number of Blocks Allocated to the File: 8
May 25 14:50:51 efe-VirtualBox kernel: [24712.748375] Number of Blocks (Pages) Cached: 1
May 25 14:50:51 efe-VirtualBox kernel: [24712.748375] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748376] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748377] Block Number: 298093
May 25 14:50:51 efe-VirtualBox kernel: [24712.748377] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748377] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748378] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748378] Descriptor Number: 4
May 25 14:50:51 efe-VirtualBox kernel: [24712.748379] Current File Position Pointer: 20004
May 25 14:50:51 efe-VirtualBox kernel: [24712.748379] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748379] Process Access Mode: 134381597
May 25 14:50:51 efe-VirtualBox kernel: [24712.748380] Name of the File: file2.txt
May 25 14:50:51 efe-VirtualBox kernel: [24712.748380] inode Number of the File: 1538
May 25 14:50:51 efe-VirtualBox kernel: [24712.748381] File Length in Bytes: 20004
May 25 14:50:51 efe-VirtualBox kernel: [24712.748381] Number of Blocks Allocated to the File: 40
May 25 14:50:51 efe-VirtualBox kernel: [24712.748382] Number of Blocks (Pages) Cached: 5
May 25 14:50:51 efe-VirtualBox kernel: [24712.748382] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748383] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748383] Block Number: 298094
May 25 14:50:51 efe-VirtualBox kernel: [24712.748384] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748384] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748385] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748385] Block Number: 298095
May 25 14:50:51 efe-VirtualBox kernel: [24712.748385] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748386] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748386] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748387] Block Number: 298096
May 25 14:50:51 efe-VirtualBox kernel: [24712.748387] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748388] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748388] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748388] Block Number: 298097
May 25 14:50:51 efe-VirtualBox kernel: [24712.748389] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748389] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748390] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748390] Block Number: 298098
May 25 14:50:51 efe-VirtualBox kernel: [24712.748390] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748391] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748391] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748392] Descriptor Number: 5
May 25 14:50:51 efe-VirtualBox kernel: [24712.748392] Current File Position Pointer: 40000
May 25 14:50:51 efe-VirtualBox kernel: [24712.748392] User's ID: 0
May 25 14:50:51 efe-VirtualBox kernel: [24712.748393] Process Access Mode: 134381597
May 25 14:50:51 efe-VirtualBox kernel: [24712.748393] Name of the File: file3.txt
May 25 14:50:51 efe-VirtualBox kernel: [24712.748394] inode Number of the File: 1539
May 25 14:50:51 efe-VirtualBox kernel: [24712.748394] File Length in Bytes: 40000
May 25 14:50:51 efe-VirtualBox kernel: [24712.748395] Number of Blocks Allocated to the File: 80
May 25 14:50:51 efe-VirtualBox kernel: [24712.748395] Number of Blocks (Pages) Cached: 10
May 25 14:50:51 efe-VirtualBox kernel: [24712.748396] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748396] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748397] Block Number: 298099
May 25 14:50:51 efe-VirtualBox kernel: [24712.748397] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748398] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748398] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748398] Block Number: 298100
May 25 14:50:51 efe-VirtualBox kernel: [24712.748399] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748399] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748400] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748400] Block Number: 298101
May 25 14:50:51 efe-VirtualBox kernel: [24712.748400] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748401] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748401] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748402] Block Number: 298102
May 25 14:50:51 efe-VirtualBox kernel: [24712.748402] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748402] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748403] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748403] Block Number: 298103
May 25 14:50:51 efe-VirtualBox kernel: [24712.748404] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748404] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748404] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748405] Block Number: 298104
May 25 14:50:51 efe-VirtualBox kernel: [24712.748405] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748406] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748406] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748406] Block Number: 298105
May 25 14:50:51 efe-VirtualBox kernel: [24712.748407] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748407] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748408] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748408] Block Number: 298106
May 25 14:50:51 efe-VirtualBox kernel: [24712.748408] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748409] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748410] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748410] Block Number: 298107
May 25 14:50:51 efe-VirtualBox kernel: [24712.748410] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748411] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748411] Storage Device (Search Key): 8388609
May 25 14:50:51 efe-VirtualBox kernel: [24712.748412] Block Number: 298108
May 25 14:50:51 efe-VirtualBox kernel: [24712.748412] Use Count: 3
May 25 14:50:51 efe-VirtualBox kernel: [24712.748412] ------------------------------
May 25 14:50:51 efe-VirtualBox kernel: [24712.748413] ==============================
May 25 14:50:51 efe-VirtualBox kernel: [24712.748413]
May 25 14:50:51 efe-VirtualBox kernel: [24712.748413] Current Directory of the Process: project4
May 25 14:51:19 efe-VirtualBox kernel: [24740.212036]
May 25 14:51:19 efe-VirtualBox kernel: [24740.212036] project4_EA_YD module is finished.
```

As it can be seen from the log, the file name information, file size information and number of block caches are printed correctly. User ID is also printed correctly since we have logged in as the `root`. Other information such as storage Device, block number, etc. are harder to check. However, we have printed the buffer content in our test runs and saw the file contents; which implies that the buffer is accessed correctly. Finally, we checked that the use count is incremented by 1 at each time we run the module. This is also the correct behavior since the module makes an additional reference to each block to retrieve the data associated with it.