

## OS 第四次作業

### 一、分工

鄧智宇：撰寫報告、程式邏輯、相關資料查找

黃信維：撰寫程式、程式邏輯、相關資料查找

### 二、作業步驟以及遇到問題

1. 在「replace\_policy.c」中進行 paging 需調度系統資源，但引入<linux/xx>系列標頭時 make 會出錯。詢問助教後，知道這次作業不需要引入其他的標頭檔。
2. 如何比較輸入的 id 和已存在 page 裡面的 id 是否一致？需檢查是否與舊資料的 request id 相符，就要另外再檢查原始的 request 到底給的是多少，所以也需要在 ioctl 一開始就要去紀錄原始 request 與轉換後的 request 對應。
3. 在 1/8 後複製修改助教提供的提示後，更改如下(包含加分題)：

```
switch(cmd_choice){
    case CMD_WRITE_REQUEST:

        //紀錄新近來 request 的 metadata
        //紀錄 virtual addr, physical addr, 是否已經在 disk內, 是否屬於 update
        //Modify here
        reqMeta[currReqNo].virtual_addr = virtual_addr;
        reqMeta[currReqNo].physical_addr = virtual_addr%256;
        reqMeta[currReqNo].update = 0;
        reqMeta[currReqNo].alreadyInDisk = 0;

        paging_find_free_page(reqMeta[currReqNo]);
        break;

    case CMD_PRINT_INFO:
        print_information();
        break;
}
```

```
//步驟三之二：
//當前沒有可用空間，所以要先找 victim frame 並執行置換
else{
    //找 victim frame 之 free list node 的位置
    //並找出 victim frame 代表的 virtual address
    victim_physical = paging_victim_selection();
    int i=0;
    list_for_each_entry(tmp_node, &free_list_header->next, next){
        //Modify here
        if(tmp_node->virtual_addr%256 == victim_physical){
            break;
        }
    }
    victim_virtual = tmp_node->virtual_addr;

    //倒回去檢查 victim 在 reqMeta 的紀錄 (因為越後面越新的狀態)，找出 victim 最新的資訊
    for(i=currReqNo; i>=0; i--){
        //若 history 找到最新 victim frame 所存的 victim_virtual 資訊
        if(reqMeta[i].virtual_addr == victim_virtual){
            //更新 victim 在 free_list 的紀錄，把 victim frame 內容覆蓋成 user_req
            //Modify here
            tmp_node->virtual_address = reqMeta[i].virtual_address;
        }
    }
}
```

```

//回傳 victim physical frame
static int paging_victim_selection(void)
{
    int victim_no;
    if(replace_policy == FIFO){
        //Modify here
        victim_no = reqMeta[currentVictimNo].physical_addr;
        if(currentVictimNo == 255){
            currentVictimNo = 0;
        }
        else{
            currentVictimNo += 1;
        }
    }
    else if(replace_policy == LRU){
        //Modify here
        struct free_list *tmp_node;
        int ref_count_min = 9999;
        victim_no = 0;

        list_for_each_entry(tmp_node, &free_list_header->next, next){
            if(tmp_node->ref_count < ref_count_min){
                ref_count_min = tmp_node->ref_count;
                victim_no = tmp_node->virtual_addr%256;
            }
        }
    }
    return victim_no;
}

```

4. 不過 makefile 卻無法成功，錯誤訊息如圖

```

veridas@veridas-VirtualBox:~/linux-5.7.9/MyChrDevice$ sudo make
[sudo] password for veridas:
make -C /lib/modules/5.7.9406315017/build M=/home/veridas/linux-5.7.9/MyChrDevice modules
make[1]: Entering directory '/home/veridas/linux-5.7.9'
scripts/Makefile.build:44: /home/veridas/linux-5.7.9/MyChrDevice/Makefile: No such file or directory
make[2]: *** No rule to make target '/home/veridas/linux-5.7.9/MyChrDevice/Makefile'. Stop.
Makefile:1732: recipe for target '/home/veridas/linux-5.7.9/MyChrDevice' failed
make[1]: *** [/home/veridas/linux-5.7.9/MyChrDevice] Error 2
make[1]: Leaving directory '/home/veridas/linux-5.7.9'
makefile:12: recipe for target 'modules' failed
make: *** [modules] Error 2

```

在助教的建議下，刪去 EXTRA\_FLAG，並重新編譯 kernel，但問題變成無法讀取檔案：

```

veridas@veridas-VirtualBox:~/Desktop/oshw_2020/hw4/hw$ ./paging_user ~/home/veridas/Desktop/oshw_2020/hw4/hw/skew_dataset
Open dataset failed.
: No such file or directory

```

後來自己探索後，發現更改檔名後就可以執行。不過仍會出現以下錯誤：

```

$ sudo ./paging_user.o skew_dataset.txt
Now send request:2610
[1] 174937 killed sudo ./paging_user.o skew_dataset.txt

```

```

[100830.187053] user_req:52 found in memory
[100830.187039] user_req:461 found in memory
[100830.187045] user_req:521 found in memory
[100830.187051] user_req:229 found in memory
[100830.187056] user_req:68 found in memory
[100830.187070] user_req:75 found in memory
[100830.187075] user_req:237 found in memory
[100830.187089] user_req:295 found in memory
[100830.187095] user_req:195 found in memory
[100830.187101] user_req:166 found in memory
[100830.187111] user_req:247 found in memory
[100830.187147] user_req:79 found in memory
[100830.187153] user_req:85 found in memory
[100830.187160] user_req:591 found in memory
[100830.187175] user_req:40 found in memory
[100830.187180] user_req:123 found in memory
[100830.187245] user_req:49 found in memory
[100830.187250] user_req:164 found in memory
[100830.187269] BUG: unable to handle page fault for address: ffffffff23fa000
[100830.187271] #PF: supervisor write access to kernel mode
[100830.187273] #PF: error code(0x0002) - not-present page
[100830.187276] PGD 40fc0e067 P4D 40fc0e067 PUD 40fc10067 PMD 7e9aa3067 PTE 0
[100830.187280] Oops: 0002 [#1] SMP NOPTI
[100830.187283] CPU: 15 PID: 159119 Comm: paging_user.o tainted: P D OE 5.4.0-53-generic #59-Ubuntu
[100830.187285] Hardware name: System manufacturer System Product Name/ROG STRIX X570-E GAMING, BIOS 1405 11/19/2019
[100830.187286] RIP: 0010:0xffffffffc23e60ed
[100830.187290] Code: 00 00 00 48 8b 4a 08 48 89 42 08 48 89 10 48 89 48 08 48 89 01 48 63 05 91 24 01 00 48 89 c2 48 c1 e0 04 48 8d 88 60 85 3e c2 <44> 89 b0 60 85 3e c2 41 0f b6 c6 89 4
04 48 c7 41 08 00 00 00 00
[100830.187292] RSP: 0018:ffffb5d34a187e18 EFLAGS: 00010216
[100830.187293] RAX: 00000000000011aa RBX: fffff9b83da2d300 RCX: ffffffff23fa000
[100830.187295] RDX: 00000000000011aa RSI: 0000000040016301 RDI: fffff9b83da2d300
[100830.187296] RBP: fffffb5d34a187e58 R08: 0000000000000000 R09: 0000000000000000
[100830.187298] R10: 0000000000000000 R11: 0000000000000000 R12: fffff9b83da2d300
[100830.187300] R13: 0000000000000000 R14: 0000000000000000 R15: fffff9b83da2d300
[100830.187302] FS: 00000000023c9880 (0000) GS: fffff9b8bcb0000 (0000) knlGS:0000000000000000
[100830.187303] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
[100830.187304] CR2: ffffffff23fa000 CR3: 000000071ffbe000 CR4: 0000000000340ee0
[100830.187305] Call Trace:
[100830.187311] ? tty_write_unlock+0x31/0x40
[100830.187310] do_vfs_ioctl+0x407/0x670
[100830.187319] ksys_ioctl+0x67/0x90
[100830.187321] _x64_sys_ioctl+0x1a/0x20
[100830.187324] do_syscall_64+0x57/0x190
[100830.187327] entry_SYSCALL_64_after_hwframe+0x44/0xa9

```

目前猜測是記憶體操作錯誤，問題應是出在 vicim\_select。

### 三、引用資料

list\_for\_each\_entry

<https://www.itread01.com/p/158186.html>

ioctl

<https://blog.xuite.net/yang44/dd/49077684>