بسمه تعالی

گزارش تمرین عملی سوم – سوال دوم – درس سیستم‌های عامل دکتر اسدی – نیمسال اول۴۰۴-۴۰۵

نویسندگان:

۱. سیداحمد موسوی‌اول – ۴۰۲۱۰۶۶۴۸

۲. عرفان تیموری – ۴۰۲۱۰۵۸۱۳

در این تمرین در ابتدا مخزن گفته شده را clone کردیم و در ادامه ابتدا make را زدیم و دستور گفته شده را وارد کردیم ولی pid آن فرق داشت و یک نبود که در زیر می‌توانید آن را ببینید:

```
/Desktop/zocker$ cd zocker/
s-ahmad-mousavi-awal@s-ahmad-mousavi-awal-Victus-by-HP-Gaming-Laptop-15-fa1xxx:~
/Desktop/zocker/zocker$ make
gcc -Wall -Wextra -std=c99 -o zocker main.c
s-ahmad-mousavi-awal@s-ahmad-mousavi-awal-Victus-by-HP-Gaming-Laptop-15-fa1xxx:~
/Desktop/zocker/zocker$ ./zocker run --name hello 'readlink /proc/self/ns/pid'
Running child with pid: 4305
pid:[4026531836]
[Parent] Stoping...
```

در ادامه ابتدا به سوال مطرح شده پاسخ می‌دهیم:

این دستور در واقع برای بررسی و نمایش شناسه و همچنین PID namespace برای PID فرایندی که در حال اجرا است استفاده میشود.

مسیر proc/<pid>/ns/pid حاوی یک لینک به فضای نام PID مربوطه است؛ محتوای این لینک که با دستور readlink خوانده می‌شود دقیقا همان pid:[inode-number] است که در این تمرین از این برای بررسی ایزوله‌سازی فضای نام استفاده می‌کنیم و inode-number منحصر به فرد تضمین می‌کند که فضای نام‌ها جدا هستند؛ اگر inode یکسان باشد، ایزوله‌سازی درست کار نکرده‌است.

در ادامه به بخش ج می‌رسیم که در مورد مقایسهٔ آنها می‌توان گفت که تفاوت‌های زیر را دارند:

- clone(2): این این سیستم‌کال یک فرآیند فرزند جدید ایجاد می‌کند و کنترل دقیق بر منابع مشترک (مانند حافظه، فایل‌ها، سیگنال‌ها) و فضای نام‌ها (namespaces) را فراهم می‌کند. با استفاده از فلگ‌هایی مانند CLONE_NEWPID، فرزند مستقیماً در فضای نام جدید قرار می‌گیرد. این روش پیچیده‌تر است زیرا نیاز به مدیریت استک (stack)، TLS و فلگ‌های متعدد برای کنترل اشتراک‌گذاری دارد. مناسب برای موارد پیشرفته مانند کتابخانه‌های threading یا ایزوله‌سازی سفارشی.

- fork(2) + unshare(2): ابتدا unshare فضای نام را از فرآیند فعلی جدا می‌کند (برای CLONE_NEWPID، فرزندان آینده را در فضای نام جدید قرار می‌دهد، اما والد در فضای قدیمی می‌ماند). سپس fork فرزند را ایجاد می‌کند. این روش ساده‌تر است زیرا جداسازی فضای نام را از ایجاد فرآیند جدا می‌کند و نیاز به تنظیمات پیچیده مانند استک ندارد.

به همین دلیل می‌توان گفت که روش دوم منطقی‌تر است چرا که از fork استاندارد استفاده می‌کند و پیچیدگی‌های clone مانند مدیریت استک یا فلگ‌های اشتراک‌گذاری را ندارد.

برای تغییرات هم به روش زیر کد run_container را عوض می‌کنیم:

```c
int run_container(struct config cfg) {
    pid_t pid;

    if (unshare(CLONE_NEWPID) == -1) {
        perror("[ERR] unshare failed");
        return 1;
    }

    pid = fork();
    if (pid < 0) {
        perror("[ERR] fork failed");
        return 1;
    }
    if (pid == 0) {
        printf("Running child with pid: %d\n", getpid());
        execl("/bin/sh", "sh", "-c", cfg.command, NULL);
        perror("[ERR] execl failed");
        return 1;
    } else {
        int status;
        waitpid(pid, &status, 0);
        printf("[Parent] Stoping...\n");
    }
    return 0;
}
```

که در اینجا ابتدا unshare زده شده و در ادامه دستور fork همانطور که گفته شد وارد شد که نتیجه مانند عکس زیر شد:

که همانطور که مشاهده می‌کنید بدون sudo کار نمی‌کند.

سوال بعدی که دقیقا همین سوال است که چرا بدون sudo کار نمیکند را به این شرح جواب می‌دهیم:

دلیل آن این است که طبق مستندی که داده شده unshare(2) که برای ایجاد فضای نام PID جدید است یک عملیات سطح سیستم است که نیاز به قابلیت CAP_SYS_ADMIN دارد و این قابلیت معمولا فقط برای کاربر root در دسترس است، زیرا عملیات‌هایی مانند ایزوله‌سازی فضای نام می‌تواند بر شناسایی فرایندها تاثیر بگذارد و برای جلوگیری از سو استفاده اینطوری محدود شده است.

در ادامه فایل Makefile را به طریق زیر تغییر دادیم تا درست شود:

```
CC = gcc
CFLAGS = -Wall -Wextra -std=c99
TARGET = zocker
```

```
4.  SOURCES = main.c
5.
6.  $(TARGET): $(SOURCES)
7.    $(CC) $(CFLAGS) -o $(TARGET) $(SOURCES)
8.
9.  setcap:
10.   sudo setcap cap_sys_admin=ep $(TARGET)
11.
12. clean:
13.   rm -f $(TARGET)
14.
15. .PHONY: clean setcap
```

در نهایت هم ابتدا make clean زدیم تا بیلد قبلی پاک شود و در ادامه make کردیم و درنهایت هم make setcap را زدیم تا درست شود و دستور دیگر بدون sudo بتواند کار کند.

کدهای تغییریافته در کنار این فایل قرار گرفته‌اند.