Operating Systems
Jacobs University Bremen
Dr. Jürgen Schönwälder

Module: CO-562
Date: 2020-11-05
Due: 2020-11-12

OS 2020 Problem Sheet #9

Problem 9.1: *open files and file updates or meta data changes* (1+1+1+1+1 = 5 points)

```
/*
 * catloop.c --
 */

#define _POSIX_C_SOURCE 201112L

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <fcntl.h>

int
main(int argc, char *argv[])
{
    int fd;
    pid_t pid;

    if (argc != 2) {
        fprintf(stderr, "catloop: missing 'file' argument\n");
        return EXIT_FAILURE;
    }

    fd = open(argv[1], O_RDONLY);
    if (fd == -1) {
        perror("catloop: open");
        return EXIT_FAILURE;
    }

    while (1) {
        pid = fork();
        if (pid == -1) {
            perror("catloop: fork");
            (void) close(fd);
            return EXIT_FAILURE;
        }
        if (pid == 0) {
            char c;
            (void) lseek(fd, 0, SEEK_SET);
            while (read(fd, &c, 1) == 1) {
                write(STDOUT_FILENO, &c, 1);
            }
            (void) close(fd);
            return EXIT_SUCCESS;
        }
        (void) waitpid(pid, NULL, 0);
        sleep(1);
    }

    (void) close(fd);
    return EXIT_SUCCESS;
}
```

Save the source code shown above into the file `catloop.c`. Compile the C code to produce the executable `catloop` and afterwards execute the following shell commands on a Linux system (the behavior may be different on a Windows system):

```
$ rm -f foo
$ touch foo
$ ./catloop foo &
$ echo -n "hello " > foo
```

Answer the following questions, always with the same initial setup.

a) What is the program doing?

b) What happens if you append content to the file `foo` while `catloop` is running?

```
$ echo "world" >> foo
```

What happens if you truncate the file `foo` while `catloop` is running?

```
$ truncate -s 0 foo
```

c) Discuss the advantages and disadvantages of the behavior you have observed in the previous step. Could there be other file system update semantics?

d) What happens if you change the permissions of the file `foo` while `catloop` is running?

```
$ chmod 0 foo
$ ls -l foo
```

e) What happens if you remove the file `foo` while `catloop` is running?

```
$ rm -f foo
```

What are possible implications of this behavior?

Problem 9.2: *file system permissions* (1+1+1+1+1 = 5 points)

Unix file system objects have basic permissions associated with (i) the file owner, (ii) the file's group members, and (iii) everybody else with access to the file system. Answer the following questions:

a) Who has which access permissions for the file `foo`?

```
$ ls -l foo
-rwxrw-r-- 1 schoenw adm 0 Nov 30 14:53 foo
```

b) Who has which access permissions for the directory `bar`?

```
$ ls -ld bar
drwx-wx--- 2 schoenw adm 4096 Nov 30 14:56 bar
```

c) Can a member of the group `adm` (who is different from `schoenw`) read the content of the directory bar? Can a member of the group `adm` (who is different from `schoenw`) create a file in the director bar? Explain.

d) A regular user (with a umask of 0022) executes the following shell command. What are the file permissions of the file that is created and who is the owner of the file? Explain.

```
$ rm -f world
$ sudo echo hello > world
```

e) What is the meaning of the following access permissions?

```
$ ls -l /usr/bin/sudo
-rwsr-xr-x 1 root root 157760 Jan 11  2016 /usr/bin/sudo
```