Operating Systems
Spring 2018

S09

Professor : Philippe Cudré-Mauroux
Assistant : Ines Arous

Submitted by Sylvain Julmy

Exercise 2

a)

An i-node is a data structure that a filesystem object like a file or a directory. Each inode stores the attributes and disk block location(s) of the object's data. Filesystem object attributes may include metadata (times of last change, access, modification), as well as owner and permission data.
The i-node is associated with a filename, give a filename the directory knows where to find the i-node and the i-node knows where to find the file on the disk.

b)

There are 3 i-nodes : *usr*, *src* and *kernels* and 4 directory block : /, *usr*, *src* and *kernels*. So there are $4 + 3 = 7$ total disk access.

Exercise 3

`rename` in the current directory only changes the filename in the directory entry. `rename` in a different directory add an entry to the new directory which point on the same i-node and remove the entry in the old one.
`cp` just create a new i-node and `rm` decreases the reference counter of the i-node.

Exercise 4

A symbolic link is just a file which contains the path of the targeted file and a hard link is a filename which point to the same i-node.

a)

Advantages of hard links :

- The link is working even if a file is removed.
- Moving the link on the same disk does not break the link.

Advantages of symbolic links :

- Can point to any file where hard links can only point on the same disk

`b)`

A Windows shortcute is a symbolic link.

`c)`

An OSX's alias is a symbolic link.

Exercise 5

What are the i-node values of file1.txt and file2.txt? Are they the same or different? The i-node values are the same like in figure 1.

```
snipy@snipy-anarchy ~/Master/mcs-git/os/exercices/s10/exercises  master ± ls -li
total 24
282400 -rw-r--r-- 2 snipy users 32 15 mai 08:45 file1.txt
282400 -rw-r--r-- 2 snipy users 32 15 mai 08:45 file2.txt
282431 -rw-r--r-- 1 snipy users 33 15 mai 08:45 file3.txt
```

Figure 1: The i-node value of the file and the hard link is the same.

Do the two files have the same or different contents ? Both files owns the same content as seen in figure 2.

```
snipy@snipy-anarchy ~/Master/mcs-git/os/exercices/s10/exercises  master ± cat file1.txt
This is the first example file.
snipy@snipy-anarchy ~/Master/mcs-git/os/exercices/s10/exercises  master ± cat file2.txt
This is the first example file.
snipy@snipy-anarchy ~/Master/mcs-git/os/exercices/s10/exercises  master ± diff file1.txt file2.txt

snipy@snipy-anarchy ~/Master/mcs-git/os/exercices/s10/exercises  master ±
```

Figure 2: The content of both file is identical.

Are the cont ents of file1.txt and file2.txt the same or different? The contents remains the same for both file as seen in figure 3

Does file2.txt still exist as well ? After removing the file, the hard link still exists as in figure 4.

What system call is used for removing file2.txt ? As seen in figure 5, there are a lot of different system call used in order to remove a file.
System call used by rm (not all, only the most impactfull) :

- `mmap` : establish a mapping between an address space of a process and a memory object.
- `close` and `read` : close or read from a channel.
- `brk` : modify the data segment size.

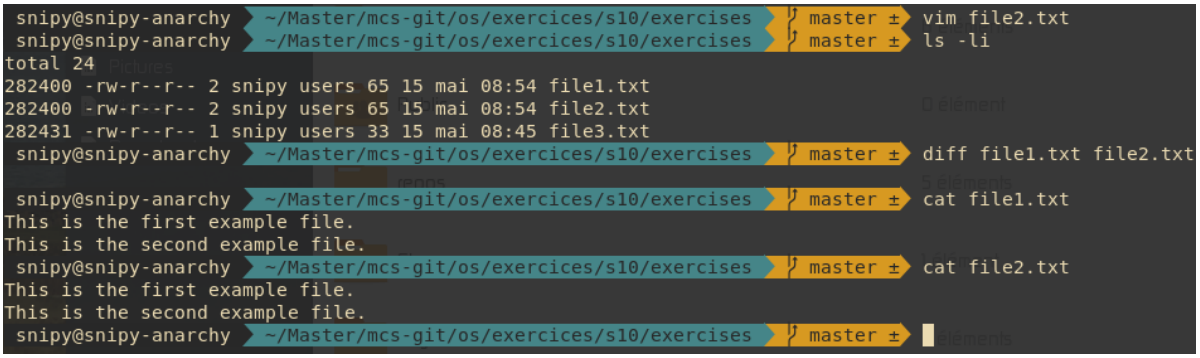

Figure 3: The content of both file is identical even after modifiying the content of the hard link.

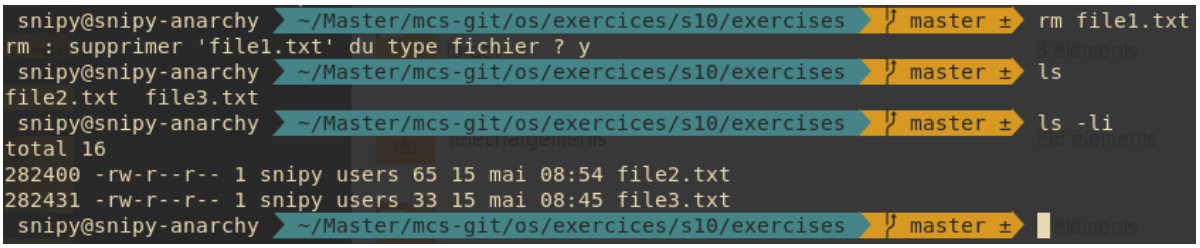

Figure 4: The hard link still exists after removing the real file.

- `unlinkat` : delete a name and possibly the file it refers to.

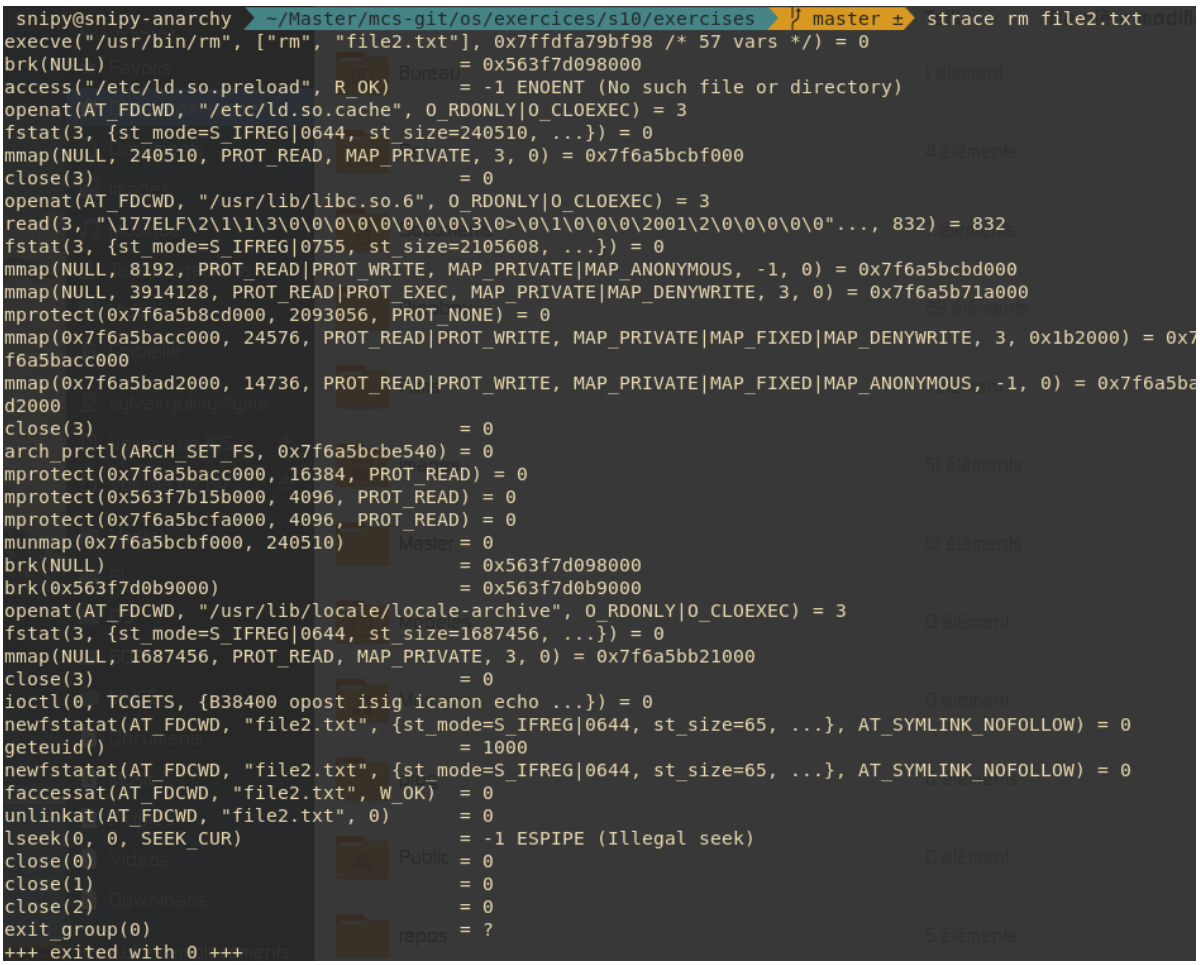

Figure 5: Output of the strace command when removing a hard link.