

# SYSTÈMES D'EXPLOITATION : LES PROCESSUS - CORRECTION

## PROCESSUS

### Exercice 2 : liste des processus

1. `lfcltx@SshSzpiegServer:~ $ ps aux`  
OU  
`lfcltx@SshSzpiegServer:~ $ ps aux |more`
2. On les repère grâce à la colonne « USER »
3. `lfcltx@SshSzpiegServer:~ $ ps aux | grep lfclt`

### Exercice 3 : tuer un processus

- 1.
2. a) La commande « who »  
b) `lfcltx@SshSzpiegServer:~ $ who`
3. a) Dans un premier terminal : `lfcltx@SshSzpiegServer:~ $ cat`  
b)  

```
lfcltx@SshSzpiegServer:~ $ ps aux | grep cat
root      77  0.0  0.0   0   0 ?        I<  18:42   0:00 [DWC Notificatio]
lfcltx    3754 0.0  0.0  6592  380 pts/1  S+  20:07   0:00 cat
lfcltx    3762 0.0  0.0  7364  560 pts/2  S+  20:08   0:00 grep --color=auto cat
```

  
c) Le processus est dans l'état « S » de « Suspendu », car il attend que l'utilisateur valide la saisie dans le 1<sup>er</sup> terminal.  
d) `lfcltx@SshSzpiegServer:~ $ kill 3754`  
e) La commande « cat » s'est terminée et la main a été rendue à l'utilisateur.  
f) Dans le premier terminal, on ne trouve pas le processus : s'étant terminé il a disparu et libéré son PID.

```
lfcltx@SshSzpiegServer:~ $ ps aux | grep 3754
lfcltx    3789 0.0  0.0  7364  540 pts/2  S+  20:15   0:00 grep --color=auto 3754
```

## Exercice 4 : arbres

- 1.
2. a) `lfcltx@SshSzpiegServer:~ $ pstree -up`  
b) On voit les processus des autres élèves (*remarque : un nom d'utilisateur n'apparaît qu'une seule fois dans une branche. Les processus descendants appartiennent implicitement à cet utilisateur*)

3. Dans le 2<sup>e</sup> terminal : `lfcltx@SshSzpiegServer:~ $ cat`

4. a) `lfcltx@SshSzpiegServer:~ $ ps aux | grep cat`

```
lfcltx@SshSzpiegServer:~ $ ps aux | grep cat
root          76  0.0  0.0   0   0 ?        I<    sept.12   0:00 [DWC Notificatio]
lfclt1       6135  0.0  0.0  6592  336 pts/6    T    13:29   0:00 cat
lfclt5       6156  0.0  0.0  6592  376 pts/4    T    13:30   0:00 cat
lfclt1       6345  0.0  0.0  6592  324 pts/6    T    13:36   0:00 cat
lfclt6       6346  0.0  0.0  6592  364 pts/3    S+   13:36   0:00 cat
lfclt1       6355  0.0  0.0  7364  556 pts/11   T    13:38   0:00 grep --color=auto cat
lfcltx       6603  0.0  0.0  6592  328 pts/7    S+   13:51   0:00 cat
lfclt4       6604  0.0  0.0  6592  340 pts/13   S+   13:51   0:00 cat
lfclt1       6605  0.0  0.0  6592  332 pts/6    S+   13:51   0:00 cat
lfclt5       6606  0.0  0.0  6592  360 pts/10   S+   13:51   0:00 cat
lfclt2       6607  0.0  0.0  6592  376 pts/5    S+   13:51   0:00 cat
lfcltx       6611  0.0  0.0  7364  552 pts/1    S+   13:52   0:00 grep --color=auto cat
```

b)

```
lfcltx@SshSzpiegServer:~ $ pstree -ups 3849
systemd(1)---sshd(483)---sshd(3731)---sshd(3737,lfcltx)---bash(3738)---cat(3849)
```

# INTERBLOCAGE

## Exercice 5 : accès concurrents à un répertoire partagé 1 (verrou)

```
marion@SshSzpiegServer:~$ sudo lsof | grep verrou
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
flock    7001                lfclt5    3rW     REG     179,2      0      2054 /tmp/verrou
cat      7002                lfclt5    3r      REG     179,2      0      2054 /tmp/verrou
flock    7003                lfclt4    3r      REG     179,2      0      2054 /tmp/verrou
flock    7005                lfcltx    3r      REG     179,2      0      2054 /tmp/verrou
flock    7006                lfclt3    3r      REG     179,2      0      2054 /tmp/verrou
flock    7018                lfclt1    3r      REG     179,2      0      2054 /tmp/verrou
flock    7042                lfclt2    3r      REG     179,2      0      2054 /tmp/verrou
```

C'est lfclt5 qui a le verrou. Il le libère :

```
marion@SshSzpiegServer:~$ sudo lsof | grep verrou
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
flock    7003                lfclt4    3rW     REG     179,2      0      2054 /tmp/verrou
flock    7005                lfcltx    3r      REG     179,2      0      2054 /tmp/verrou
flock    7006                lfclt3    3r      REG     179,2      0      2054 /tmp/verrou
flock    7018                lfclt1    3r      REG     179,2      0      2054 /tmp/verrou
flock    7042                lfclt2    3r      REG     179,2      0      2054 /tmp/verrou
cat      7068                lfclt4    3r      REG     179,2      0      2054 /tmp/verrou
```

C'est maintenant lfclt4 qui a le verrou. Il le libère :

```
marion@SshSzpiegServer:~$ sudo lsof | grep verrou
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
flock    7005                lfcltx    3rW     REG     179,2      0      2054 /tmp/verrou
flock    7006                lfclt3    3r      REG     179,2      0      2054 /tmp/verrou
flock    7018                lfclt1    3r      REG     179,2      0      2054 /tmp/verrou
flock    7042                lfclt2    3r      REG     179,2      0      2054 /tmp/verrou
cat      7074                lfcltx    3r      REG     179,2      0      2054 /tmp/verrou
```

C'est maintenant lfcltx qui a le verrou. Il le libère :

```
marion@SshSzpiegServer:~$ sudo lsof | grep verrou
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
flock    7006                lfclt3    3rW     REG     179,2      0      2054 /tmp/verrou
flock    7018                lfclt1    3r      REG     179,2      0      2054 /tmp/verrou
flock    7042                lfclt2    3r      REG     179,2      0      2054 /tmp/verrou
cat      7079                lfclt3    3r      REG     179,2      0      2054 /tmp/verrou
```

C'est maintenant lfclt3 qui a le verrou. Il le libère :

```
marion@SshSzpiegServer:~$ sudo lsof | grep verrou
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
flock    7018                lfclt1    3rW     REG     179,2      0      2054 /tmp/verrou
flock    7042                lfclt2    3r      REG     179,2      0      2054 /tmp/verrou
cat      7085                lfclt1    3r      REG     179,2      0      2054 /tmp/verrou
```





## Exercice 7 : interblocage

1.

```
lfcltx@SshSzpiegServer:~ $ cp /home/lfcltxpartage/Processus/Interb1.sh ./Processus/
lfcltx@SshSzpiegServer:~ $ cp /home/lfcltxpartage/Processus/Interb2.sh ./Processus/
```

2.

Script Interb1.sh :	Script Interb2.sh :
Un processus P1 : <ul style="list-style-type: none"><li>• pose un premier verrou (n°8) sur le fichier R1.txt</li><li>• fait une pause de 15 secondes</li><li>• alors qu'il a toujours le verrou n°8 bloqué, il bloque un 2<sup>e</sup> verrou n°9 sur le fichier R2.txt</li><li>• il attend 5 secondes</li><li>• il déverrouille le verrou n°9</li><li>• puis il déverrouille le verrou n°8</li><li>• il fait apparaître un message à l'écran annonçant qu'il a libéré les deux verrous</li></ul>	Un processus P2 : <ul style="list-style-type: none"><li>• pose un premier verrou (n°9) sur le fichier R2.txt</li><li>• fait une pause de 15 secondes</li><li>• alors qu'il a toujours le verrou n°9 bloqué, il bloque un 2<sup>e</sup> verrou n°8 sur le fichier R1.txt</li><li>• il attend 5 secondes</li><li>• il déverrouille le verrou n°8</li><li>• puis il déverrouille le verrou n°9</li><li>• il fait apparaître un message à l'écran annonçant qu'il a libéré les deux verrous</li></ul>

3.

4. a) b) c) Après avoir un message disant qu'un processus d'un certain PID se réveille dans chaque fenêtre, il ne se passe rien. Tout est bloqué car on vient de créer une situation d'interblocage.

5. a)

```
lfcltx@SshSzpiegServer:~ $ lsof | grep R2.txt
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
bash      10635      lfcltx      9w      REG      179,2      0      261978 /home/lfcltx/Processus/R2.txt
bash      10639      lfcltx      9w      REG      179,2      0      261978 /home/lfcltx/Processus/R2.txt
flock     10640      lfcltx      9w      REG      179,2      0      261978 /home/lfcltx/Processus/R2.txt
bash      10641      lfcltx      9w      REG      179,2      0      261978 /home/lfcltx/Processus/R2.txt
flock     10642      lfcltx      9w      REG      179,2      0      261978 /home/lfcltx/Processus/R2.txt
```

```
lfcltx@SshSzpiegServer:~ $ lsof | grep R1.txt
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1000/gvfs
Output information may be incomplete.
bash      10630      lfcltx      8w      REG      179,2      0      261893 /home/lfcltx/Processus/R1.txt
bash      10639      lfcltx      8w      REG      179,2      0      261893 /home/lfcltx/Processus/R1.txt
flock     10640      lfcltx      8w      REG      179,2      0      261893 /home/lfcltx/Processus/R1.txt
bash      10641      lfcltx      8w      REG      179,2      0      261893 /home/lfcltx/Processus/R1.txt
flock     10642      lfcltx      8w      REG      179,2      0      261893 /home/lfcltx/Processus/R1.txt
```

b)

```
lfcltx@SshSzpiegServer:~ $ pstree -sup 10640
systemd(1)---sshd(519)---sshd(10520)---sshd(10538,lfcltx)---bash(10539)---bash(10629)---bash(10630)---bash(10639)---flock(10640)
lfcltx@SshSzpiegServer:~ $ pstree -sup 10642
systemd(1)---sshd(519)---sshd(10552)---sshd(10558,lfcltx)---bash(10559)---bash(10634)---bash(10635)---bash(10641)---flock(10642)
```

c) On peut tuer les processus qui correspondent aux commandes flocks. Donc ici 10640 (ce qui débloquent le script Interb1.sh : il finira son exécution, puis Interb2 finira la sienne à son tour) ou 10642 (ce qui débloquent le script Interb2.sh).