

Rapport TP4 Antoine Maendly

Que se passera-t-il si nous déverrouillons un fichier (ou une partie du fichier) qui n'est pas verrouillé ?

déverrouiller un fichier qui n'est pas verrouillé

```
PID=12824> s u 0 1
Successfully removed lock from 0 to 1
PID=12824> s u 0 1
Successfully removed lock from 0 to 1
```

On voit que si on enlève un verrou ou il n'y a pas de verrou, il n'y aura pas de différence que si on venait de l'enlever. La fonction fcntl ne va pas retourner quelque chose en particulier et le programme va considérer que l'opération est réussie.

déverrouiller une partie d'un fichier

On met un write lock de 0 à 2, puis on l'enlève de 0 à 1

```
PID=12824> s w 0 2
Successfully put a write lock from 0 to 2
PID=12824> s u 0 1
Successfully removed lock from 0 to 1
```

Depuis un autre processus, on peut voir que il n'y a plus de lock de 0 à 1 mais que il y a toujours un lock de 1 à 2.

```
PID=12937> g r 0 1
No lock at the position
PID=12937> g r 1 1
There is a write lock detained by 12824 from 1 to 2
```

Donc cela déverrouille que la partie concerné

Que se passera-t-il si nous mettons un nouveau verrou sur une section déjà verrouillée? Le type de verrou changera-t-il le résultat? Expliquer dans la situation avec le même processus et avec 2 processus différents

```
PID=13296> s w 0 1
Successfully put a write lock from 0 to 1
PID=13296> s w 0 2
Successfully put a write lock from 0 to 2
PID=12937> g r 0 1
There is a write lock detained by 13296 from 0 to 2
```

On voit que si on met un verrou sur un section déjà verrouiller, cela ne change rien et l'opération se termine correctement. si on met un verrou plus long, cela rallonge aussi sur toute la section du nouveau verrou

```
PID=13626> s r 0 1
Successfully put a read lock from 0 to 1
PID=13626> s w 0 1
Successfully put a write lock from 0 to 1
PID=13626> s r 0 1
Successfully put a read lock from 0 to 1

PID=13593> g w 0 1
There is a read lock detained by 13626 from 0 to 1
PID=13593> g w 0 1
There is a write lock detained by 13626 from 0 to 1
PID=13593> g w 0 1
There is a read lock detained by 13626 from 0 to 1
```

Dans cet exemple, on met un lock puis on check dans l'autre processus ligne par ligne (on écrit pas les trois lignes d'un coup). On peut voir que mettre un lock d'un autre type va changer le type du lock et que l'opération se termine avec succès.