

Processus et redirections

F. Lassabe

Printemps 2021

Contents

1	Exercice 1 : processus	2
1.1	États d'un processus	2
1.2	Commandes d'information sur les processus	2
2	Exercice 2 : manipulation des processus	3
3	Exercice 3 : les redirections	4
3.1	Découverte par l'exemple	4
3.2	Application	5

1 Exercice 1 : processus

1.1 États d'un processus

1. Définir les différents états d'un processus ?

2. Donner un schéma montrant toutes les transitions d'états possibles en utilisant les signaux indiqués dans le tableau ci-dessous. Préciser pour chaque transition le signal associé (joindre le schéma à ce pdf)

kill -TERM pid	Terminer proprement un processus avec le signal SIGTERM (non interceptable)
kill -INTR pid	Interruption du programme proprement avec le signal SIGINT, équivalent à Ctrl-C
kill -KILL pid	Force le programme à se terminer avec le signal SIGKILL, non interceptable.
kill -CONT pid	Reprend l'exécution d'un processus stoppé (avec STOP)
kill -STOP pid	Met en pause le processus (reprise avec CONT), équivalent à Ctrl-Z
kill -QUIT pid	Équivalent au caractère quitter non présent sur tous les claviers
commande&	Lancement de la commande <i>commande</i> en arrière plan
fg numéro	Remet le processus <i>numéro</i> au premier plan. S'il était suspendu, reprend au premier plan. <i>numéro</i> est obtenu avec la commande <i>jobs</i>

1.2 Commandes d'information sur les processus

Les commandes ci-dessous permettent d'avoir des informations sur les processus en cours d'exécution sur le système avec des indicateurs sur l'utilisation des ressources mémoire et CPU :

- ps
- ps -f
- ps -e
- ps -fH
- ps -efH
- ps a
- ps au
- ps aux
- pstree
- pstree *username*
- top
- jobs

1. Tester ces commandes et trouver les options listées dans le manuel de chacune des commandes ps, pstree, top et jobs, puis répondre aux questions qui suivent.

2. Lancer dans un terminal les 2 commandes gedit (en arrière plan) et man ps.
3. Lancer à partir du mode graphique les programmes Firefox et Libre Office Writer.
4. Ouvrir un nouveau terminal et exécuter la commande ps avec les options indiquées précédemment. Déterminer à chaque fois que c'est possible les informations suivantes : Id du processus, Id du processus père, le propriétaire du processus, consommation mémoire et CPU ; le terminal auquel est attaché le processus.

Programme	PID	PID père	Propriétaire	Mémoire	CPU	terminal
gedit 1						
gedit 2						
man ps						
Firefox						
Writer						

5. Quelle est la hiérarchie de tous vos processus lancés en 1 et 2.

2 Exercice 2 : manipulation des processus

Répondre à chacune des questions suivantes (en indiquant quelle commande a été utilisée) :

- Ouvrir deux terminaux, le terminal 1 pour manipuler les processus donc l'exécution des commandes ps et kill et le terminal 2 pour le lancement des autres processus.
- Exécuter la commande ps a (terminal 1).
- Afficher le manuel de la commande ln (terminal 2) et stopper le processus man dès l'affichage de la première page (à partir du terminal 1).
- Vérifier que le processus a bien été stoppé (terminal 2).
- Reprendre l'exécution de man (vérifier le man) puis le tuer (terminal 1).
- Vérifier si le processus man est également tué, tuer le processus si ce n'est pas le cas (terminal 2).
- Lancer de nouveau man ln (terminal 2), tuer le processus man (terminal 1), le processus man est-il tué ?
- Lancer en arrière plan l'éditeur gedit puis saisir quelques mots (terminal 2).
- Mettre gedit au premier plan.

- Stopper gedit (terminal 1), est-il possible de saisir du texte dans la fenêtre gedit ouverte ?
- Tuer le processus gedit (terminal 1).
- Stopper le processus bash du terminal 1, est-il possible de le relancer ?, si oui alors relancez-le sinon expliquer pourquoi.
- Lancer la commande man ln (terminal 1) puis stopper le processus. Est-il possible de le relancer ?
- Lancer la commande man ln (terminal 1) puis suspendre le processus. Est-il possible de le relancer ?

3 Exercice 3 : les redirections

3.1 Découverte par l'exemple

Exécuter chacune des commandes suivantes et dire quel est le résultat.

1. `ls -l > lsl.txt`
2. `cat lsl.txt`
3. `cat lsl.txt |wc -l`
4. `cat < lsl.txt`
5. `wc < lsl.txt`
6. `cat > lsl.txt`
7. `cat lsl.txt`
8. `echo "Que fait cette commande ?" > lsl.txt`

```

9. cat lsl.txt

10. echo "# Informations sur le répertoire courant " > lsl.txt

11. echo "Nombre de fichiers" >> lsl.txt

12. ls -l | wc {1 >> lsl.txt

13. echo "Liste et informations sur les fichiers " >> lsl.txt

14. ls {1 >> lsl.txt

15. less lsl.txt

16. echo "export PS1=\"/>

```

3.2 Application

Créer les cinq fichiers f1, f2, f3, f4 et f5 dont le contenu est le suivant :

f1	f2	f3	f4	f5
Firefox	vim	evince	Writer	
Chrome	nano	okular	Calc	
Edge	emacs	acroread	Impress	
	gedit	calibre		
	kate			

À l'aide des opérations vues dans la section 3.1, répondre aux questions suivantes en indiquant les commandes utilisées :

1. Copier le contenu des fichiers f1 à f4 vers f5, le contenu de f5 doit être trié.
2. Ajouter le contenu de f1 à f2
3. Copier le contenu de f3 vers f4 (en écrasant l'ancien contenu de f4).
4. Ajouter à f4 les lignes de 4 à 7 du fichier f2
5. Effacer tout le contenu de f3.
6. Remplir le fichier f3 avec les 10 premières et 10 dernières lignes du manuel de la commande cp. Le fichier doit commencer par la ligne "# Ceci sont les 10 premières et dix dernières lignes du manuel de la commande cp"
7. Supprimer la première ligne de f4
8. Remplacer le contenu de f1 par celui des fichiers f1, f2, f3, f4, f5, dans l'ordre. La première ligne doit indiquer le nombre de lignes du fichier résultat.
9. Supprimer les fichiers de f1 à f5.