

5.2 - Surveiller et arrêter les processus

Un **processus** est une instance d'un programme en cours d'exécution. Il possède un identifiant unique, le PID, ainsi qu'un propriétaire et un état.

Lister les processus

```
ps
ps aux
ps -ef
ps -u "$USER"
ps -p 1234 -o pid,ppid,user,%cpu,%mem,etime,cmd
```

Rechercher par nom :

```
pgrep -a sshd
pidof sshd
```

Éviter `ps aux | grep nom` lorsque `pgrep -a nom` suffit.

Surveiller en direct

```
top
htop
```

Dans `top`, `P` trie par processeur, `M` par mémoire et `q` quitte. `htop` est plus convivial mais n'est pas toujours installé.

Résumé de la mémoire :

```
free -h
```

Sous Linux, une partie de la mémoire « utilisée » sert de cache récupérable. La colonne `available` est généralement plus pertinente que `free` seule.

Envoyer un signal

```
kill 1234          # SIGTERM : demande un arrêt propre
kill -15 1234      # identique
kill -9 1234       # SIGKILL : arrêt forcé par le noyau
pkill -TERM nom
```

Méthode recommandée :

1. vérifier le PID et le propriétaire ;
2. envoyer `SIGTERM` ;
3. attendre et contrôler ;
4. utiliser `SIGKILL` seulement si le processus ne répond pas.

`SIGKILL` empêche le programme de nettoyer ses fichiers temporaires ou de terminer une écriture.

Tâches au premier et à l'arrière-plan

```
commande-longue &
jobs
fg %1
bg %1
```

Suspendre le programme au premier plan avec `Ctrl + Z`, puis utiliser `bg` ou `fg`. `Ctrl + C` demande son interruption.

Garder une commande après la fermeture du terminal :

```
nohup commande-longue > commande.log 2>&1 &
```

Pour les tâches importantes, préférer un service `systemd`, `tmux` ou `screen`, qui offrent une gestion plus claire.

Priorité

```
nice -n 10 commande  
renice 10 -p 1234
```

Une valeur `nice` plus élevée réduit la priorité CPU. Des valeurs négatives demandent généralement des privilèges.

Pour un service, agir avec `systemctl` plutôt que tuer directement son processus : le gestionnaire peut sinon le relancer immédiatement.

Revision #2

Created 2026-08-08 01:24:52 CEST by Admin

Updated 2026-08-08 01:46:46 CEST by Admin