

Introduction à GNU/Linux

Processus

Pascal Bessonneau

Starinix

11/2017

La théorie

Les signaux

En mode graphique

En mode console

Les systèmes *GNU/Linux*

Le propos sera simplifié et pourra choquer des spécialistes. Les processus et leur gestion est un sujet complexe.

Un processus est par exemple quand vous lancez une application. Elle possède son petit bout de mémoire propre, des informations comme le répertoire où vous l'avez lancé, le droit d'avoir un certain temps d'utilisation du CPU, ...

Sous *GNU/Linux*, un processus (une application au sens large) est toujours le fils d'un autre processus. Ainsi à chaque boot, le premier processus va se multiplier à chaque que nécessaire avec des infos et des droits adaptés.

Les systèmes *GNU/Linux*

Cette hiérarchie peut être vue avec la commande *pstree*.

Au quotidien

Les éléments importants à savoir au quotidien est principalement que chaque processus se voit affecter des droits (par exemple root, ou pascal ou gaston) et qu'il a un numéro unique qui l'identifie. Identifier les droits avec lequel est lancé le processus est important car cela conditionne ce qu'on pourra faire au sein du processus. C'est la raison pour laquelle on utilise *sudo*. On laisse en utilisant cette commande le processus en tant que root.

Les systèmes *GNU/Linux*

Un processus peut recevoir des signaux venant du noyau. Les signaux sont, en grande partie, la forme de communication pour gérer les processus.

Nous verrons ici les ordres envoyés aux processus par le noyau : ceux qui nous intéressent sont ceux qui vont gérer l'arrêt et le redémarrage des processus.

Ainsi les signaux à mémoriser sont :

TERM (15) pour dire à l'application de se fermer

KILL (9) pour tuer l'application

HUP (1) pour redémarrer l'application (par exemple un serveur pour qu'il relise ses fichiers de configurations)

Au quotidien

Le numéro de processus va permettre par exemple s'il se bloque de l'arrêter ou de le tuer en utilisant une commande et le numéro du processus.

Oui sous *GNU/Linux* on tue des processus, c'est cruel.

Terminal/Console

GNOME *Moniteur Système.*

KDE *KDE System Guard* dans Système.

Au quotidien

C'est la version simple. Si votre processus a un nom unique, la commande permet de tuer les processus. Par exemple pour stopper le navigateur Firefox vous pouvez taper :

```
kill firefox
```

S'il ne répond pas du tout vous pouvez le tuer :

```
kill -9 firefox
```

Au quotidien

```
pascal  31945  105  5.6 9244360 903364 ?      S1    19:32
pascal  32073  0.0  0.0  14376  1088 pts/0    S+    19:33
```

on voit que Firefox est le processus 31945 et qu'il est en sommeil.

Au quotidien

Pour le stopper, *kill 31945* ou le tuer *kill -9 31945*.

Vous pouvez visualiser les tâches en arborescence avec *ps* à l'aide de cette commande : *ps aux -ejHu*

Au quotidien

Il y a aussi l'utilitaire *top* il permet d'afficher en temps (presque) réel les processus actifs.

Vous pouvez classer différemment en utilisant les touches :

- F** trie selon une colonnes différentes

- u** n'affiche que cet utilisateur

- k** stoppe un processus

- q** quitter *top*