# Firejail

## Késako

Parfois, il est nécessaire utiliser des applications qui n'ont pas été bien testées dans différents environnements, mais vous devez les utiliser. Dans de tels cas, il est normal de se préoccuper de la sécurité de votre système. Une chose qui peut être faite sous Linux est d'utiliser des applications dans un **bac à sable** (en anglais “**Sandboxing**).

Exécuter une application dans un bac à sable offre la possibilité de lancer cette application dans un environnement limité. De cette façon, l'application dispose d'une quantité accrue de ressources, nécessaires à son exécution. Grâce à **Firejail**, vous pouvez exécuter en toute sécurité des applications non fiables sous Linux.

**Firejail** est une application **SUID** (Set Owner User ID) qui réduit l'exposition aux failles de sécurité.

Il permet à un processus et à tous ses descendants d'être exécuté dans une prison (un jail) afin de garantir la sécurité du système.

## Installation

Depuis les dépôts apt install firejail

sudo apt install firejail

Depuis le git git clone https://github.com/netblue30/firejail.git cd firejail ./configure && make && sudo make install-strip

## Syntaxe de base

firejail <OPTIONS> <APPLICATION>

Par exemple

firejail mousepad

Attention : Sans aucune option, le bac à sable se compose d'un système de fichiers construit dans un nouvel espace de noms de montage et de nouveaux espaces de noms PID et UTS. Les espaces de noms IPC, réseau et utilisateur peuvent être ajoutés à l'aide de la ligne de commande. Le système de fichiers Firejail par défaut est basé sur le système de fichiers hôte avec les principaux répertoires système montés en lecture seule. Ces répertoires sont /etc, /var, /usr, /bin, /sbin, /lib, /lib32, /libx32 et /lib64. Seuls /home et /tmp sont accessibles en écriture

# Usage avancé

## Les fichiers de configuration

Afin de sécurité le système, il y a toute une floppée d'options (voir le man de firejail pour en avoir un aperçu)qui s'ajoute aux profils de base.

Lors de l'exécution, Firejail cherche d'abord dans ~/.config/firejail/ un profil éponyme au nom de l'application et s'il n'en trouve pas, il cherche dans /etc/firejail/.

Si aucun profil approprié n'est trouvé, Firejail utilisera un profil par défaut.

Le profil par défaut est assez restrictif. Au cas où l'application ne fonctionne pas, utilisez l'option –noprofile pour le désactiver.

## Les profils par défaut

Ce sont ces profils qui donneront les privilèges au lancement d'une application via firejail.

Ces profils sont localisés dans

ls /etc/firejail/ | wc

```
 1101    1101   20335
```

Plus de mille profils sont disponibles... Un profil éponyme à l'application lancée pour les plus communes. Et aussi des héritages de profil que nous laisserons ici de côté.

Ce sont ces profils qui définiront les règles de sécurité d'ouverture / fermeture des portes de la prison.

Attention, il ne faut pas modifier ces profils sous /etc car en cas de mise à jour de firejail ils seront écrasés par les nouveaux. Dans ce cas, il suffit de copier le profil à modifier sous ~/.config/firejail/

## Lancement sans nom d'application

SI vous omettez de préciser le nom de l'application, Firejail démarre le shell préféré de l'utilisateur.

firejail Reading profile /etc/firejail/default.profile Reading profile /etc/firejail/disable-common.inc Reading profile /etc/firejail/disable-passwdmgr.inc Reading profile /etc/firejail/disable-programs.inc Warning: networking feature is disabled in Firejail configuration file

**Note: you can use –noprofile to disable default.profile**

Parent pid 211751, child pid 211752 Child process initialized in 78.39 ms

Ainsi lancer, le /home de l'$USER sera accessible comme d'habitude , un ls vous le confirmera.

Pour sortir de ce bac à sable, saissez simplement l'option exit

Pour lister les bacs à sable en cours d'exécution :

firejail –list 211751:ragnarok::firejail

## L'option --private

Sans doute la plus intéressante. C'est cette option qui va enfermer votre application à l'intérieure d'une prison.

En effet, cette option permet de monter /root et /home/$USER dans les systèmes de fichiers temporaires. A la fermeture du bac à sable toutes les modifications seront ignorées. Par conséquent, l'application se lancera comme si c'était la première fois qu'on la lançait sur un système vierge, les fichiers de configuration ne seront pas conséquent pas lus.

Exemple:

firejail –private claws-mail

## Le cas de Firefox

Vous voudrez sans doute peaufiner tout cela.