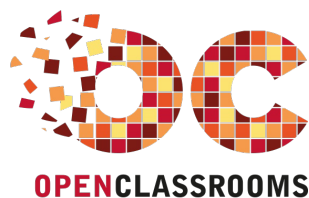


Installation et configuration de GRUB

Par Sébastien



www.openclassrooms.com

Sommaire

Sommaire	2
Installation et configuration de GRUB	3
Avant-propos	3
Pourquoi GRUB ?	3
Installation	4
Légende	4
Configuration générale	4
Section de boot	6
Légende	6
Les labels	7
Dernières indications	8
Mettre une image de fond	8
Fichier final	9
Partager	10



Installation et configuration de GRUB

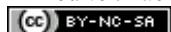


Par

Sébastien

Mise à jour : 01/01/1970

Difficulté : Facile



Quand on est débutant, se balader entre le xorg.conf, le rc.conf et autres joyeusetés n'est pas forcément chose aisée. Aussi, j'ai décidé de vous apprendre à installer et à bien configurer le célèbre bootloader GNU GRUB.

Sommaire du tutoriel :



- [Avant-propos](#)
- [Pourquoi GRUB ?](#)
- [Installation](#)
- [Configuration générale](#)
- [Section de boot](#)
- [Mettre une image de fond](#)
- [Fichier final](#)

Avant-propos

Alors déjà, il est mieux de savoir avec quoi nous allons travailler, non ? Eh bien GRUB (GRand Unified Bootloader), c'est un bootloader. Voici une bonne définition de Wikipédia :

Citation : Wikipédia

Un chargeur d'amorçage (ou Boot loader) est un logiciel permettant de lancer un ou plusieurs systèmes d'exploitation (multi-boot), c'est-à-dire qu'il permet d'utiliser plusieurs systèmes, à des moments différents, sur la même machine.

C'est-à-dire que vous pouvez le configurer pour qu'après le chargement du BIOS, vous puissiez choisir le système à démarrer (GNU/Linux, Mac, FreeBSD, Windows...). Et pour la configuration, il y a justement un fichier qui contient toutes les informations : [/boot/grub/menu.lst](#) (ou [grub.conf](#) selon les distributions).

Pourquoi GRUB ?

GRUB n'est pas le seul bootloader existant : il y a notamment son principal concurrent LILO (Linux LOader) et quelques autres projets libres mais moins connus comme [XOSL](#) ou [GAG](#).

LILO a été pendant très longtemps le seul utilisé. Il possède une dizaine d'années de solide développement et est par conséquent très stable. Mais ce développement semble depuis peu aller au ralenti, pour ne pas dire qu'il est abandonné.

GRUB est lui actif depuis moins longtemps, et est encore en version alpha, version 1.96. Cela ne veut pas dire qu'il marche mal et qu'il plantera au démarrage de votre ordinateur, mais c'est un projet assez jeune par rapport à LILO. S'il est aujourd'hui tant utilisé, c'est qu'il possède quand même des avantages sur lui :

- Son interface en ligne de commande est plus complète
- Si un problème survient avec GRUB, il sera souvent plus facilement réparable
- Il possède un site web, une FAQ, etc. contrairement à LILO qui lui n'a rien !

Pour compléter cette brève introduction, vous pouvez lire [cet article](#) (en anglais) très intéressant même s'il date un peu.

Installation

La plupart des distributions installent elles-mêmes GRUB. Mais si vous l'avez désinstallé par erreur, ou pour je ne sais quelle raison, vous devez l'installer par vos propres moyens. Suivez les étapes ci-dessous.

L'installation peut se faire par deux moyens : par la compilation, ou par l'installation d'un paquet. Nous verrons ici, bien évidemment, la deuxième option.

Sous Debian par exemple, en tant que root, commencez par cette commande :

Code : Console

```
# aptitude install grub
```

(Pour les autres distributions, vous devrez changer le gestionnaire de paquets.)

Et puis :

Code : Console

```
# grub
# grub> root (hd?1,?2)
# grub> setup (hd?3,?4)
```

Légende

- ?1 -> Le disque de l'OS sur lequel vous êtes actuellement. Une source courante d'erreur est de ne pas prendre en compte la numérotation débutant par 0 et plutôt par 1.
- ?2 -> La partition de l'OS sur laquelle vous êtes actuellement. De même que pour le disque, le décompte commence à partir de 0.
- ?3 -> Le disque de l'OS sur lequel vous voulez installer GRUB (c'est généralement la même valeur que ?1).
- ?4 -> La partition de l'OS sur laquelle vous voulez installer GRUB (c'est généralement la même valeur que ?2).

Avec cela, vous devriez être capables de l'installer correctement. 🤖

Configuration générale

Premièrement, ouvrons le fichier à modifier (oui, c'est toujours plus facile quand on le fait 🤖).

Vous pouvez le faire par la console (en root) :

Code : Console

```
# cp /boot/grub/menu.lst /boot/grub/menu.lst.bak # Il serait préférable d'exécuter
# nano /boot/grub/menu.lst # Vous pouvez utiliser vi si vous voulez, mais comme on
```

Ou en graphique (en root toujours) :

Code : Console

```
# cp /boot/grub/menu.lst /boot/grub/menu.lst.bak # Il serait préférable d'exécuter  
# gedit /boot/grub/menu.lst # Si vous êtes sous Gnome.  
# kate /boot/grub/menu.lst # Si vous êtes sous KDE.
```

Le fichier commence par trois options générales :

- **timeout**
- **default**
- **color**
- **fallback**

timeout correspond au délai d'attente (en secondes) avant le démarrage de l'OS choisi par défaut.

default est l'OS à démarrer par défaut.

color sont les couleurs de GRUB (on s'en serait presque douté 😊).

fallback est l'OS à démarrer si celui par défaut ne boot pas.

Le début du fichier devrait donc ressembler à ceci :

Code : Autre

```
timeout    10 # J'ai mis 10, ça me paraît bien  
default    0 # On laisse en général 0 et l'ordre est indiqué plus tard  
fallback   1 # Si default ne boot pas  
color light-blue/black light-  
cyan/blue # Ici sont mes valeurs par défaut, vous mettez comme vous voulez, sachez  
plan, la deuxième modifie la couleur de 1er plan.
```

Il y a aussi quelques options facultatives :

- **hiddenmenu**
- **password**

hiddenmenu permet de cacher le menu. Il apparaît tout de même si l'on presse Échap.

password est utilisé pour mettre un mot de passe lors du démarrage de GRUB, pour éviter que n'importe qui utilise votre ordinateur.

Code : Autre

```
hiddenmenu  
# Deux utilisations de password : une avec md5, l'autre sans :  
password --md5 Le_Mot_De_Passe_En_MD5_Sinon_Il_Y_Aura_Un_Probleme  
# ou alors simplement :  
password Mon_Super_Mot_De_Passe
```

Pour plus d'information sur md5 :

[Lien Wikipédia](#)

Passons au plus dur : la section de boot.

Section de boot

La section de boot est celle qui pose souvent problème. En effet, il faut bien configurer les numéros de disque, de partition, etc.

Voyons la structure à utiliser pour un OS de type GNU/Linux où est installé GRUB :

Code : Autre

```
#(0) Mon Super OS GNU/Linux
title          Mon Super OS GNU/Linux
root           (hd?1,?2)
kernel         /boot/vmlinuz-2.x.x.xx root=/dev/sd?3?4 ?5
initrd         /boot/initrd.img-2.x.x.xx
```

Légende

- ?1 -> Le disque de l'OS. Une source courante d'erreur est de ne pas prendre en compte la numérotation débutant par 0 et plutôt par 1.
- ?2 -> La partition de l'OS. De même que pour le disque, le décompte commence à partir de 0.
- ?3 -> La racine du système. La lettre du disque (a, b, c).
- ?4 -> Comme ?3, pour le chiffre de la partition (1, 2, 3...).
- ?5 -> Les options. L'option **ro** sert à démarrer en "lecture seule", pour un mode sans échec par exemple, ou **quiet** qui permet un démarrage discret.

Et ci-dessous la structure pour Windows :

Code : Autre

```
#(1) Mon Super Windows
title          Mon Super Windows
root           (hd?1,?2)
chainloader    +1
makeactive
```

Pour chaque OS en plus de celui sur lequel est installé GRUB, il faut ajouter une ligne :

Code : Autre

```
rootnoverify    (hd?1,?2)
```

Un exemple :

Code : Autre

```
#(2) Mon Autre Super OS GNU/Linux
title          Mon Autre Super OS GNU/Linux
root           (hd?1,?2)
kernel         /boot/vmlinuz-2.x.x.xx root=/dev/sd?3?4 ?5
initrd         /boot/initrd.img-2.x.x.xx
rootnoverify    (hd?1,?2)
```

Les labels

Depuis tout à l'heure, nous indiquons la racine du système comme ceci :

Code : Autre

```
kernel /boot/vmlinuz-2.x.x.xx root=/dev/sd?3?4 ?5
```

Mais il existe une autre façon de faire. Voyez plutôt :

Code : Autre

```
kernel /boot/vmlinuz-2.x.x.xx root=UUID=bd54f622-87b7-4ac2-b2c4-acb3c7e56a75
```



C'est quoi, cette chose affreuse qui commence par **UUID** ? 🤔

Eh bien c'est un label. Il utilise un UUID (Universal Unique Identifier). Voici une définition de la documentation d'Ubuntu :

Citation : doc.ubuntu-fr.org

Il s'agit d'une suite plus ou moins longue de caractères alpha-numériques qui permet d'identifier de façon absolument sûre toute partition ou tout type de périphérique de stockage ("device") physique ou virtuel pouvant être raccordé au système de fichiers. L'utilisation de l'UUID est généralisée.

Le chiffre de l'UUID est calculé automatiquement au moyen d'un algorithme intégrant notamment certaines données de l'ordinateur hôte. Ce mode de calcul ne présente aucun risque de sécurité crédible.

Comme elle le précise, il y a un avantage et une contrainte.

Il évite tout problème en cas de changement de désignation du disque / de la partition, mais il est un peu plus difficile à manipuler.

Il est en général préférable de l'utiliser à la place de l'ancienne méthode.

Voici la commande permettant d'obtenir l'UUID (en root) :

Code : Console

```
# vol_id -u /dev/sd?2?3
```

Vous pouvez dès lors mettre à jour votre ligne :

Code : Autre

```
kernel /boot/vmlinuz-2.x.x.xx root=UUID=?1
```

Légende

- ?1 -> La valeur que vous renvoie vol_id.
- ?2 -> La racine du système. La lettre du disque (a, b, c).
- ?3 -> La partition de l'OS. Le décompte commence à partir de 0.

Dernières indications



T'es bien gentil, mais on ne sait toujours pas ce que sont kernel et initrd !

Justement, nous y venons.

kernel, eh bien c'est le chemin vers le noyau de l'OS. Je ne pourrais pas le deviner à votre place, mais la commande suivante pourra vous le dire, pour remplacer les "x" :

Code : Console

```
uname -r
```

initrd, c'est un système temporaire utilisé au démarrage. Vous devez remplacer les petits "x" de la même manière qu'avec le kernel.

Mettre une image de fond

Disons-le franchement, GRUB est généralement assez moche. 🙄

Pour pallier ce problème, des développeurs ont créé un patch permettant d'insérer une image de fond à GRUB.

Il faut installer un paquet pour pouvoir utiliser une image. Sur Debian par exemple, il se nomme grub-splashimages (commande à lancer en tant que root, comme toujours) :

Code : Console

```
# aptitude install grub-splashimages
```

(Je ne pourrai pas détailler chaque distribution, à vous de fouiller la doc.)

Il faut que l'image en question soit dans un format spécial : le [.xpm.gz](#).

La convertir se fait en une petite commande :

Code : Console

```
$ convert -resize 640x480 -  
colors 14 image_à_convertir.format splashimage.xpm && gzip splashimage.xpm
```

S'il y a une erreur, il faut installer ImageMagick :

Code : Console


```
# aptitude install imagemagick
```

Puis déplacez [splashimage.xpm.gz](#) dans `/boot/grub/` :

Code : Console

```
# mv splashimage.xpm.gz /boot/grub/
```

Dernière étape : l'ajout d'une petite ligne dans [menu.lst](#) :

Code : Autre

```
timeout 10
default 0
color light-blue/black light-cyan/blue

# La ligne en question de l'image
splashimage=(hd?1,?2)/boot/grub/splashimage.xpm.gz
```

Il faut remplacer les points d'interrogation de la même manière que précédemment.

Un exemple :

Code : Autre

```
splashimage=(hd0,0)/boot/grub/splashimage.xpm.gz
```

C'est tout pour les images.

En bonus, pour vous, voici un lien où se trouvent de belles images prêtes à l'emploi pour votre GRUB. 😊

Schultz-net.dk

Fichier final

Allez, je suis de bonne humeur, je vais vous aider un peu en vous montrant un exemple de fichier final. 😊

Code : Autre

```
# Configuration générale

timeout 10
default 0
fallback 1
color light-blue/black light-cyan/blue

password --
md5 f02368945726d5fc2a14eb576f7276c0 # Ici le mot de passe est bonjour (changez

#Hiddenmenu

# Section de boot
# Ce sont des exemples, ils ne fonctionneront sûrement pas chez vous, modifiez-les
```

```
#(0) Ubuntu
title          Ubuntu
root           (hd0,4)
kernel         /boot/vmlinuz-2.6.24-21-generic root=UUID=bd54f622-87b7-4a
acb3c7e56a75 ro quiet splash
initrd         /boot/initrd.img-2.6.24-21-generic

title          Microsoft Windows XP Professionnel
root           (hd0,0)
makeactive
chainloader    +1

title          Arch Linux
root           (hd0,2)
kernel         /boot/vmlinuz26 root=/dev/disk/by-uuid/e2c590c0-56f6-4684-
initrd         /boot/kernel26.img
```

Et voilà : c'est tout.

Voilà : comme vous le voyez, la configuration de GRUB n'est pas d'une difficulté insurmontable.

Si vous voulez aller encore plus loin, je ne peux que vous conseiller la [documentation officielle de GNU](#) (en anglais).

Je suis ouvert à toutes remarques : à bientôt. 😊

Partager

