


GRUB 2 (vision d'ensemble)

- Objet : GRUB 2, le chargeur d'amorçage version 2 (présent à partir de **squeeze**).
- Niveau requis :
[avisé](#)
- Commentaires : *installation et configuration de GRUB 2 lorsqu'un autre bootloader est présent, en particulier GRUB Legacy.*
- Suivi :
[à-tester](#)
 - Créé par  [smolski](#) le 16/02/2010
 - Mise à jour par ... le ...
 - Testé par ... le ...
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) ¹⁾

Présentation

Le code de grub2 est le seul maintenu et il n'y aura plus d'ajout de nouvelles fonctionnalités à grub (Legacy) premier du nom.

Le code de grub2 a fait l'objet d'une réécriture complète.

L'installation est divisée en plusieurs modules qui sont destinés à différentes architectures et plateformes, avec une base commune.

Exit les écritures directement dans un fichier, mais à la place une configuration au petit poil grâce aux commandes et aux scripts personnalisés propre à grub2.

Au cœur du fonctionnement de grub2 on trouve :

1. L'image amorçable « boot.img »
2. Le mini-initrd "core.img"

Ce dernier est suffisamment réduit pour être implanté entre le secteur d'amorçage du disque et le début de la première partition, ce qui rend possible de nombreuses utilisations avec :

- des `multiboot`
- des systèmes non-GNU/Linux
- des partitionnements variés, par exemple utilisant *raid* et/ou *lvm* rendant disponibles les volumes de ce type dès le menu grub !

Installation

le choix entre grub-pc et grub-efi est déterminé par la façon d'amorcer l'installateur, en mode EFI ou BIOS/legacy. Exception : quand l'installateur est amorcé en mode EFI mais détecte un système déjà installé en mode BIOS/legacy, il propose de continuer comme s'il avait été amorcé en mode BIOS/legacy afin d'installer un chargeur d'amorçage compatible avec le système déjà installé.

De raleur sur le forum :

- <https://debian-facile.org/viewtopic.php?pid=264471#p264471>

La table de partition indique que Windows a été installé en mode EFI. Il faut donc installer un chargeur GRUB EFI pour pouvoir lancer Windows à partir de GRUB.

Lors de l'installation, grub ne sera pas effacé du secteur d'amorçage (MBR) du disque, grub2 sera *chainé* à partir du grub déjà existant pour permettre de vérifier que tout fonctionne correctement.

Au démarrage suivant il faudra choisir dans le menu grub habituel l'entrée commençant par :

```
"chainload into grub2 on ...".
```

Si tout se passe bien, il est alors temps de faire totalement le remplacement de grub par grub2, en exécutant la commande :

```
upgrade-from-grub-legacy
```

Après cela :

1. grub2 sera installé dans le secteur d'amorçage
2. Le fichier **/boot/grub/grub.cfg** prendra le relais de **/boot/grub/menu.lst**.

/boot/grub/menu.lst reste présent pour servir de référence, cependant, il peut maintenant être supprimé.

Nota :

Sur certaines distributions²⁾ la procédure est un peu différente.

Pour basculer sur grub2 définitivement il faut l'installer sur le **mbr** en utilisant la commande :

```
grub-install
```

Commandes

L'utilisation des commandes spécifiques de grub2 devient la ligne de conduite obligée pour le fonctionnement de grub2 :

1. configurer,
2. réparer,
3. administrer.

- [Les commandes de grub2](#)

Configuration

- Pour grub, la configuration consistait à retoucher les entrées du fichier menu .lst directement.
- Avec grub2 on intervient sur un ensemble de fichiers de configuration qui seront *sourcés* au

moment de la génération du **grub.cfg** (à la manière du *sources.list* pour [apt](#)).

La configuration modulaire est basée sur des scripts personnels et/ou modifiables.

Procédure :

1. Créer un script (basé sur les modèles fournis),
2. Le copier dans le répertoire correspondant (par exemple **/etc/grub.d/**),
3. Mettre à jour le fichier **/boot/grub/grub.cfg** (`update-grub`).



Toute modification éditée directement dans le *grub.cfg* sera écrasée lors des appels automatiques à **update-grub** ou **grub-mkconfig**.

NOTA :

grub2 numérote les disques à partir de 0³⁾ mais les **partitions** sont maintenant numérotées à partir de un et non plus de zéro !

Donc, pour la première partition du premier disque qui était pour **grub** :

```
(hd0,0)
```

devient pour **grub2** :

```
(hd0,1)
```

Résolution

grub2 supporte des images de plus haute résolution. Et même, au stade expérimental, la vidéo paraît-il !

Shell

grub2 propose aussi un environnement "shell" (console) développé, avec notamment :

- des fonctions de recherche,
- un support des boucles,
- il est scriptable,
- il est chaînable,
- il supporte le format **ext4**.

MODULE : GRUB - Détail

Dans un terminal, taper :

```
nano /etc/default/grub
```

Vous y lisez :

```
# Ce fichier est sourcé par update-grub ou grub-mkconfig, et ses variables
sont
# propagées
# aux fichiers "fils" dans /etc/grub.d.

# Entrée choisie automatiquement par défaut, "0" = première entrée du
fichier
# grub.cfg .
GRUB_DEFAULT=0

# Durée avant exécution du choix par défaut.
GRUB_TIMEOUT=4

# Reconnaissance de la distribution.
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`

# Chargement des modules additionnels, ici RAID, afin que ces volumes soient
# disponibles dès le
# menu grub. Indispensable pour des partitions /boot ou / (root) sur raid.
GRUB_PRELOAD_MODULES="raid mdraid"

# Options supplémentaires à passer au noyau (resume=/dev/sda2 ,
# acpi=off...). Concerne aussi les entrées "recovery mode"
GRUB_CMDLINE_LINUX="acpi=off"

# Options supplémentaires à passer au noyau, ne concerne QUE les entrées par
# défaut, pas les entrées "recovery"
GRUB_CMDLINE_LINUX_DEFAULT="quiet vga=0x0365"

# Décommenter pour inhiber le terminal graphique et forcer un terminal texte
# (grub-pc seulement).
#GRUB_TERMINAL=console

# Résolution et type du terminal graphique (défaut 640x480, gfxgrub). Si une
# liste de résolution est donnée, elles serviront d'alternatives de secours
en
# cas d'impossibilité d'utiliser la principale.
GRUB_GFXMODE="1440x900x24 800x600x24 640x480"
GRUB_TERMINAL=gfxterm

# Décommenter pour empêcher l'usage des UUID "root=UUID=xxx" dans grub.cfg .
#GRUB_DISABLE_LINUX_UUID=true
```

GRUB_PRELOAD_MODULES

Cette variable indique à GRUB2 de charger des modules additionnels, par exemple raid, lvm, video ... Ces modules seront ajoutés à l'image de démarrage /boot/grub/core.img (qui agit à la manière d'un initrd (Initial Ram Disk) minimal).

GRUB_CMDLINE_LINUX

Cette variable concerne toutes les entrées du menu, y compris **recovery**.

Cette variable indique les options permanentes à transmettre au noyau au moment de son chargement.

C'est ici qu'on indique :

- les options liées à l'**acpi**,
- la verbosité du démarrage ...

Les options doivent être séparées par un espace et l'ensemble de la liste encadrée par des guillemets.

GRUB_CMDLINE_LINUX_DEFAULT

Cette variable ne concerne QUE les entrées par défaut, pas les entrées **recovery**.

Elle permet de spécifier les options supplémentaires à passer au noyau :

- **quiet**,
- **vga=**,
- etc...

Voir ici : [TP - Utilisation de la configuration par DEFAULT](#).

GRUB_GFXMODE - La Résolution

Permet de choisir la résolution du terminal graphique gfxgrub lorsqu'il est utilisé.

Tout d'abord, pour connaître les modes supportés par votre carte vidéo, vous pouvez utiliser le programme (à installer si absent) **hwinfo** :

```
hwinfo --framebuffer
```

Pour être certain que la résolution sera disponible on peut passer par un shell grub lors du démarrage.

1. Passer en mode édition avec la touche **E**
2. puis **Ctrl+C** pour passer sur le shell grub,
3. **Echap** pour en sortir).

Dans le shell grub, pour connaître la liste des résolutions disponibles, on tape :

```
# vbeinfo
```

(#videoinfo avec un pc-efi)

Si le matériel le supporte on pourra par exemple utiliser une résolution de 1280×1024 avec des couleurs codés sur 24bit :

```
GRUB_GFXMODE=1280x1024x24
```

Ce qui se traduira dans votre **grub.cfg** par la mention:

```
"set gfxmode=1280x1024x24"
```

La première valeur donnée est celle par défaut, les suivantes (facultatives) seront utilisées en secours si la première échoue :

```
GRUB_GFXMODE="1440x900 800x600 640x480"
```



Attention !

1. Si vous utilisez une résolution élevée supportée par votre matériel
2. et une image de résolution inférieure

celle-ci ne sera pas affichée, ou ne couvrira qu'une partie de l'écran.

Il faut donc créer une image à la bonne résolution et en indiquer le chemin dans `/etc/grub.d/05_debian_theme` (*voir plus bas*).

GRUB_DISABLE_LINUX_UUID=true

Permet d'empêcher l'utilisation automatique des **UUID** pour l'adressage de la partition racine dans le `grub.cfg`.

Modifier l'ordre des OS

Pour modifier l'ordre de 2 OS affichés dans le GRUB tu édites `/etc/default/grub` et tu remplaces par 2 le 0 de `GRUB_DEFAULT=0` :

```
nano /etc/default/grub
```

[retour de la commande](#)

```
GNU nano 2.2.6          Fichier : /etc/default/grub

# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
```

```
# info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
[...]
```

Et pour finir :

```
update-grub
```

Voir la Résolution sur le forum :

- <https://debian-facile.org/viewtopic.php?id=15433>

Un kro merci à Blogus pour sa question et aux intervenants qui l'ont aidé à la résoudre finement ! 😊

SCRIPTS de configuration

Les autres fichiers de configuration se trouvent dans le répertoire `/etc/grub.d/`

Sur Debian on y trouve :

- 00_header
- 05_debian_theme
- 10_hurd
- 10_linux
- 20_memtest86+
- 30_os-prober
- 40_custom

On remarque le préfixe numérique placé devant (00_, 05_, 10_, ...) !

Ce préfixe détermine l'ordre dans lequel les scripts seront lus par `grub-mkconfig`.

/etc/grub.d/00_header

Gère la plupart des variables contenues dans `/etc/default/grub`.

Il n'y a normalement aucune raison de modifier ce script.

On y trouve des fonctions d'évaluation des variables contenu dans le **/etc/default/grub**, afin de les passer au **grub.cfg**

Exemple:

```
[...]
for i in ${GRUB_PRELOAD_MODULES} ; do
    echo "insmod $i"
```

done

```
if [ "x${GRUB_DEFAULT}" = "x" ] ; then GRUB_DEFAULT=0 ; fi
if [ "x${GRUB_TIMEOUT}" = "x" ] ; then GRUB_TIMEOUT=4 ; fi
if [ "x${GRUB_GFXMODE}" = "x" ] ; then GRUB_GFXMODE=640x480 ; fi

cat << EOF
set default=${GRUB_DEFAULT}
set timeout=${GRUB_TIMEOUT}
EOF
[...]
```

Ce fragment de fichier montre que le script **00_header** source les valeurs des variables relatives au chargement de modules spécifiques :

- du choix par défaut dans le menu du temps avant démarrage automatique
- du choix par défaut de la résolution attribuée à **gfxgrub**.

/etc/grub.d/05_debian_theme

Ce sera spécifique à votre distribution.

Il gère les éléments du thème graphique utilisé, comme l'image de fond du menu GRUB2 et les couleurs des polices.

Modifier le thème par défaut



ATTENTION ! Consulter aussi le *précédent* script de configuration pour la résolution, là :
GRUB_GFXMODE - La Résolution

Pour spécifier un chemin vers une image particulière on pourra modifier la partie concernant le thème par défaut :

```
[...]
# check for usable backgrounds
use_bg=false
if [ "$GRUB_TERMINAL_OUTPUT" = "gfxterm" ] ; then
  for i in
    {/boot/grub,/usr/share/images/desktop-base}/moreblue-orbit-grub.{png,tga} ;
  do
    [...]
  done
fi
[...]
```

Inclure un chemin d'image

Pour inclure le chemin vers une image différente (ici les images incluses dans le paquet **grub2-**

splashimages):

```
[...]
# check for usable backgrounds
use_bg=false
if [ "$GRUB_TERMINAL_OUTPUT" = "gfxterm" ] ; then
    for i in
    {/boot/grub,/usr/share/images/desktop-
    base,/usr/share/images/grub}/Windbuchencom.{png,tga}; do
    [...]

```

Couleur de la police

Modifier la couleur de la police :

```
[...]
# set the background if possible
if ${use_bg} ; then
    prepare_grub_to_access_device `${grub_probe} --target=device ${bg}`
    cat << EOF
insmod ${reader}
if background_image `make_system_path_relative_to_its_root ${bg}` ; then
    set color_normal=black/black
    set color_highlight=magenta/black
else
EOF
fi
[...]
```



Ne modifiez pas ce qui concerne le thème par défaut, il constitue le choix de secours si votre personnalisation ne fonctionne pas.

/etc/grub.d/10_hurd - /etc/grub.d/10_linux

Des scripts spécifiques au chargement d'un type de noyau, respectivement pour les systèmes GNU-HURD ou GNU-Linux.

Ces scripts génèrent les entrées pour le système par défaut du **grub.cfg**.

Il n'y a aucune raison de les modifier directement.

Ces scripts sourcent les informations de **/etc/default/grub**, et évaluent un grand nombre de paramètres comme :

- la présence d'un initrd,
- l'emplacement,
- l'UUID et le type de volume sur lequel est situé la racine du système (/)
- ... etc.

/etc/grub.d/20_memtest86+

Entrée pour lancer l'application de test de mémoire `memtest` si elle est installée.

/etc/grub.d/30_os-prober

Est issu du paquet **os-prober**.

C'est un script additionnel à grub2 qui détecte les autres systèmes d'exploitation installés, et ajoute une entrée dans le **grub.cfg** automatiquement.

/etc/grub.d/40_custom

C'est le script qui pourra faire l'objet de vos modifications.

On peut le copier, lui donner un chiffre et un nom différent, puis y intégrer :

- un script spécifique,
- une entrée supplémentaire pour un système qui ne serait pas détecté par **os-prober**,
- ...etc.

Par défaut il ne contient rien.



Veillez à ce que le script créé soit exécutable.

Voir des TP d'utilisation ici : [Ajouter des distributions variées](#)

FICHER grub.cfg

Avec **GRUB2**, le fichier **/boot/grub/grub.cfg** remplace le fichier **/boot/grub/menu.lst** habituel du **GRUB**



Toute modification du `grub.cfg` faite directement sera écrasée lors d'un appel à **update-grub** ou **grub-mkconfig**.

NOTA 1 :

GRUB2 numérote les disques à partir de 0 (zero) comme GRUB, mais les partitions sont maintenant numérotées à partir de 1 et non plus de zéro (0).

Donc :

- (hd0,0) pour GRUB (première partition du premier disque) devient
- (hd0,1) avec GRUB2.

NOTA 2 :

Pour changer temporairement une entrée du **grub.cfg** on utilisera plutôt le **shell** de GRUB2 lors du démarrage.

Pour un changement permanent on créera un script dans **/etc/grub.d**

GRUB2 et ext4

Le système de fichier **ext4**, fait son apparition dans plusieurs distributions.

Sur Debian vous pouvez démarrer directement sur une partition système en **ext4** à partir de Squeeze. Si vous utilisez grub⁴⁾ vous devez avoir une partition **/boot** séparée, formatée en **ext3**.

Voir le tuto pour plus de renseignements :

<http://www.linuxpedia.fr/doku.php/expert/grub2>



¹⁾

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

²⁾

Ubuntu Hardy

³⁾

zero

⁴⁾

ou une version plus ancienne de grub2

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:grub2:detail>

Last update: **09/05/2018 11:56**

