

Chapitre 6. Installation et démarrage à partir d'un noyau

Table of Contents

- [Chapitre 6. Installation et démarrage à partir d'un noyau](#)
- [Utilisation des scripts d'installation d'une distribution](#)
- [Installation manuelle](#)
- [Modification du chargeur de démarrage pour le nouveau noyau](#)
 - [GRUB](#)
 - [LILO](#)

Source: [Linux Kernel in a Nutshell: 6 Installing and Booting From a Kernel](#)

Les chapitres précédents ont montré comment télécharger et construire un noyau. Maintenant qu'on a un fichier exécutable - ainsi que tous les modules nécessaires - il faut installer le noyau et essayer de le démarrer. Dans ce chapitre, contrairement aux précédentes, toutes les commandes doivent être exécutées en tant qu'utilisateur root. Cela peut être fait en préfixant chaque commande avec `sudo`, en utilisant la commande `su` pour devenir root, ou en se connectant en tant que root.

Pour voir si **sudo** est installé et si on a la configuration d'accès appropriée, procéder comme suit:

```
sudo ls ~/linux/linux-2.6.17.11/Makefile
```

```
Password:  
Makefile
```

Entrer soit un nouveau mot de passe à l'invite de mot de passe, soit le mot de passe de l'administrateur système (root). Le choix dépend de la configuration de la commande `sudo`. Si cela réussit, la ligne suivante est renvoyée:

```
Makefile
```

on peut alors passer à la section suivante.

Si **sudo** n'est pas installé ou ne vous donne pas les droits appropriés, essayez d'utiliser la commande `su`:

```
su  
Password:  
...  
exit  
exit
```

À l'invite de mot de passe, entrer le mot de passe de l'administrateur système (root). Lorsque le programme se accepte avec succès le mot de passe, on est autorisé à tout exécuter avec les privilèges root complets. Il faut très prudent en tant que root, et ne faire que le minimum nécessaire; puis quitter le programme pour continuer en tant que compte d'utilisateur normal.

Utilisation des scripts d'installation d'une distribution

Presque toutes les distributions sont livrées avec un script appelé **installkernel** qui peut être utilisé par le système de construction du noyau pour installer automatiquement un noyau construit à l'emplacement approprié et modifier le chargeur de démarrage afin que rien de plus ne soit nécessaire par le développeur. ¹⁾



Les distributions qui offrent **installkernel** le placent généralement dans un package appelé **mkinitrd**, il faut donc installer ce package si on ne trouve pas le script sur votre machine.

Lorsqu'on a construit des modules et que l'on souhaite utiliser cette méthode pour installer un noyau, entrer d'abord:

```
make modules_install
```

Cela installera tous les modules que construits et les placera à l'emplacement approprié dans le système de fichiers pour que le nouveau noyau les trouve correctement. Les modules sont placés dans le répertoire `/lib/modules/KERNEL_VERSION`, où **KERNEL_VERSION** est la version du noyau du nouveau noyau que l'on viens de construire.

Une fois les modules correctement installés, l'image principale du noyau doit être installée:

```
make install
```

Cela lancera le processus suivant:

- Le système de construction du noyau vérifiera que le noyau a été correctement construit.
- Le système de construction installera la partie du noyau statique dans le répertoire `/boot` et nommera ce fichier exécutable en fonction de la version du noyau du noyau intégré.
- Toutes les images **ramdisk** initiales nécessaires seront automatiquement créées, en utilisant les modules qui viennent d'être installés pendant la phase **modules_install**.
- Le programme de chargeur de démarrage sera correctement averti qu'un nouveau noyau est présent, et il sera ajouté au menu approprié afin que l'utilisateur puisse le sélectionner au prochain démarrage de la machine.

Une fois cette opération terminée, le noyau est installé avec succès et vous pouvez redémarrer en toute sécurité et essayer votre nouvelle image de noyau. Notez que cette installation ne remplace pas les anciennes images du noyau, donc s'il y a un problème avec votre nouvelle image du noyau, l'ancien noyau peut être sélectionné au démarrage.

Installation manuelle

Si la distribution n'a pas de commande **installkernel**, ou lorsqu'on souhaite simplement faire le travail à la main pour comprendre les étapes impliquées, voici les étapes impliquées.

- Les modules doivent être installés: `make modules_install`
- L'image du noyau statique doit être copiée dans le répertoire `/boot`. Pour un noyau basé sur i386, procéder comme suit
 - récupérer la version du noyau avec la commande: `make kernelversion` (la version du noyau sera probablement différente pour votre noyau. Utilisez cette valeur à la place du texte **KERNEL_VERSION** dans toutes les étapes suivantes)
 - copier le fichier `bzImage` dans le répertoire `/boot` : `cp arch/i386/boot/bzImage /boot/bzImage-KERNEL_VERSION`
 - copier le fichier `System.map` dans dans le répertoire `/boot` : `cp System.map /boot/System.map-KERNEL_VERSION`
- Modifier le chargeur de démarrage pour qu'il connaisse le nouveau noyau. Cela implique la modification d'un fichier de configuration pour le chargeur de démarrage qu'i est traité plus loin dans la section intitulée «Modification du chargeur de démarrage pour le nouveau noyau» pour les chargeurs de démarrage **GRUB** et **LILO**.

Si le processus de démarrage ne fonctionne pas correctement, c'est généralement parce qu'une image initiale du disque virtuel est nécessaire. Pour créer cela correctement, utilisez les étapes au début de ce chapitre pour installer automatiquement un noyau, car les scripts d'installation de distribution savent comment créer correctement le disque virtuel à l'aide des fichiers nécessaires. scripts et outils. Parce que chaque distribution fait cela différemment, il est hors de la portée de ce livre de couvrir toutes les différentes méthodes de construction de l'image du disque virtuel.

Voici un script pratique qui peut être utilisé pour installer le noyau automatiquement au lieu d'avoir à taper les commandes précédentes tout le temps:

```
#!/bin/sh
#
# installs a kernel
#
make modules_install

# find out what kernel version this is
for TAG in VERSION PATCHLEVEL SUBLEVEL EXTRAVERSION ; do
    eval `sed -ne "/^$TAG/s/ //gp" Makefile`
done
SRC_RELEASE=$VERSION.$PATCHLEVEL.$SUBLEVEL$EXTRAVERSION

# figure out the architecture
```

```
ARCH=`grep "CONFIG_ARCH " include/linux/autoconf.h | cut -f 2 -d "\"`

# copy the kernel image
cp arch/$ARCH/boot/bzImage /boot/bzImage-"$SRC_RELEASE"

# copy the System.map file
cp System.map /boot/System.map-"$SRC_RELEASE"

echo "Installed $SRC_RELEASE for $ARCH"
```

Modification du chargeur de démarrage pour le nouveau noyau

Il existe deux chargeurs de démarrage du noyau Linux courants: **GRUB** et **LILLO**. **GRUB** est celui qui est le plus couramment utilisé dans les distributions modernes, et fait certaines choses un peu plus facilement que **LILLO**, mais **LILLO** est toujours vu également.

Pour déterminer quel chargeur de démarrage le système utilise, recherchez dans le répertoire `/boot/`. S'il existe un sous-répertoire `grub`:

```
ls -F /boot | grep grub

grub/
```

alors on utilise le programme **GRUB** pour démarrer. Si ce répertoire n'est pas présent, rechercher la présence du fichier `/etc/lilo.conf`:

```
ls /etc/lilo.conf

/etc/lilo.conf
```

Si c'est le cas, on utilise le programme **LILLO** pour démarrer.

Les étapes impliquées dans l'ajout d'un nouveau noyau à chacun de ces programmes sont différentes, donc suivre uniquement la section qui correspond au programme que l'on utilise.

GRUB

Pour informer **GRUB** qu'un nouveau noyau est présent, il suffit de modifier le fichier `/boot/grub/menu.lst`. Pour plus de détails sur la structure de ce fichier et toutes les différentes options disponibles, consulter les pages d'informations de **GRUB**:

info grub

La façon la plus simple d'ajouter une nouvelle entrée de noyau au fichier `/boot/grub/menu.lst` est de copier une entrée existante. Par exemple, si on considère le fichier `menu.lst` suivant d'un système Gentoo:

```
timeout 300
default 0

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

title 2.6.16.11
    root (hd0,0)
    kernel /bzImage-2.6.16.11 root=/dev/sda2 vga=0x0305

title 2.6.16
    root (hd0,0)
    kernel /bzImage-2.6.16 root=/dev/sda2 vga=0x0305
```

La ligne commençant par le mot **title** définit une nouvelle entrée de noyau, donc ce fichier contient deux entrées. Copier simplement les lignes d'une occurrence du mot **title** à la suivante, comme:

```
title 2.6.16.11
    root (hd0,0)
    kernel /bzImage-2.6.16.11 root=/dev/sda2 vga=0x0305
```

à la fin du fichier et modifier le numéro de version pour qu'il contienne le numéro de version du nouveau noyau que l'on viensz d'installer. Le titre n'a pas d'importance, tant qu'il est unique, mais il est affiché dans le menu de démarrage, il faut donc en faire quelque chose de significatif. Dans l'exemple, on a installé le noyau 2.6.17.11, donc la copie finale du fichier ressemble à:

```
timeout 300
default 0

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

title 2.6.16.11
    root (hd0,0)
    kernel /bzImage-2.6.16.11 root=/dev/sda2 vga=0x0305

title 2.6.16
    root (hd0,0)
    kernel /bzImage-2.6.16 root=/dev/sda2 vga=0x0305

title 2.6.17.11
    root (hd0,0)
    kernel /bzImage-2.6.17.11 root=/dev/sda2 vga=0x0305
```

Après avoir enregistré le fichier, redémarrer le système, le titre de la nouvelle image du noyau apparaît dans le menu de démarrage. Utiliser la flèche vers le bas pour mettre en surbrillance la nouvelle version du noyau et appuyer sur Entrée pour démarrer la nouvelle image du noyau.

LILLO

Pour informer **LILLO** qu'un nouveau noyau est présent, il faut modifier le fichier de configuration `/etc/lilo.conf`, puis exécuter la commande **lilo** pour appliquer les modifications apportées au fichier de configuration. Pour plus de détails sur la structure du fichier de configuration **LILLO**, consulter la page de manuel **LILLO**:

```
man lilo
```

La façon la plus simple d'ajouter une nouvelle entrée de noyau au fichier `/etc/lilo.conf` est de copier une entrée existante. Par exemple, si on considère le fichier de configuration **LILLO** suivant à partir d'un système Gentoo:

```
boot=/dev/hda
prompt
timeout=50
default=2.6.12

image=/boot/bzImage-2.6.15
    label=2.6.15
    read-only
    root=/dev/hda2

image=/boot/bzImage-2.6.12
    label=2.6.12
    read-only
    root=/dev/hda2
```

La ligne commençant par le mot **image=** définit une nouvelle entrée de noyau, donc ce fichier contient deux entrées. Copier simplement les lignes d'une occurrence du mot **image=** à la suivante, comme:

```
image=/boot/bzImage-2.6.15
    label=2.6.15
    read-only
    root=/dev/hda2
```

à la fin du fichier et modifier le numéro de version pour qu'il contienne le numéro de version du nouveau noyau que l'on vient d'installer. Le label n'a pas d'importance, tant qu'il est unique, mais il est affiché dans le menu de démarrage, il faut donc en faire quelque chose de significatif. Dans l'exemple, on a installé le noyau 2.6.17.11, donc la copie finale du fichier ressemble à:

```
boot=/dev/hda
prompt
timeout=50
default=2.6.12

image=/boot/bzImage-2.6.15
    label=2.6.15
    read-only
    root=/dev/hda2

image=/boot/bzImage-2.6.12
    label=2.6.12
    read-only
    root=/dev/hda2

image=/boot/bzImage-2.6.17
    label=2.6.17
    read-only
    root=/dev/hda2
```

Après avoir enregistré le fichier, exécuter le programme **/sbin/lilo** pour écrire les modifications de configuration dans la section de démarrage du disque:

```
/sbin/lilo
```

Maintenant, le système peut être redémarré en toute sécurité. Le nouveau choix de noyau doit apparaître dans la liste des noyaux disponibles au démarrage. Utiliser la flèche vers le bas pour mettre en surbrillance la nouvelle version du noyau et appuyer sur Entrée pour démarrer la nouvelle image du noyau.

1)

Les exceptions notables à cette règle sont Gentoo et d'autres distributions de type "from scratch", qui attendent des utilisateurs qu'ils sachent installer eux-mêmes les noyaux. Ces types de distributions incluent de la documentation sur la façon d'installer un nouveau noyau.

From:
<http://www.ouarte.garden/> - **dwndoc**

Permanent link:
<http://www.ouarte.garden/doku.php?id=ifs:lkn-c06-installing-and-booting-from-a-kernel>

Last update: **2025/02/19 10:59**

