

---

# Une clef USB polyvalente à base de Slackware Linux

Damien Goutte-Gattat <dgouttegattat@incenp.org>

Copyright © 2010, 2011, 2012, 2013, 2019 Damien Goutte-Gattat

2019/12/21

## Table des matières

1. Créer une clef USB « multi-install » .....	1
1.1. Préparation de la clef .....	1
1.2. Installation de LILO .....	2
2. Un peu plus loin : « multi-install » + distribution live .....	2
3. Ma clef polyvalente en 2013 .....	3
4. Ma clef polyvalente en 2019 .....	5
4.1. Partitionnement .....	5
4.2. Copie de l'installateur Slackware .....	6
4.3. Copie de la Slackware Live Edition .....	6
4.4. Installation de GRUB .....	7

## 1. Créer une clef USB « multi-install »

La Slackware est fournie avec une image contenant un système minimal et un programme d'installation destiné à être copié sur une clef USB, ladite clef pouvant alors servir de média d'installation de la distribution. Le fichier `usb-and-pxe-installers/README_USB.txt` contient toutes les informations nécessaires pour utiliser cette image.

Néanmoins, ces instructions ne permettent que d'installer *une* copie du programme d'installation (soit environ 25 Mo) sur une clef USB. Si la clef est plus volumineuse (ce qui est aujourd'hui la norme), l'espace en sus est inutilisé et inutilisable.

Inutilisable ? Pas tout-à-fait. Il est possible d'aller un peu plus loin que ne le propose le seul fichier `README_USB.txt`. Je montre ici comment créer une clef USB contenant *plusieurs* versions du programme d'installation de la Slackware (par exemple, un pour la Slackware 32 bits et un pour la 64 bits), avec la possibilité pour l'utilisateur de sélectionner au démarrage celui qu'il veut lancer.

### 1.1. Préparation de la clef

Vous êtes bien sûr censés avoir une clef USB à votre disposition, ne contenant aucune donnée (du moins aucune donnée que vous préféreriez ne pas perdre — tout son contenu sera effacé). Branchez-là. S'il y a déjà un système de fichiers sur la clef (la plupart des clefs sont vendues pré-formatées en FAT32), ne le montez pas ; démontez-le s'il a déjà été monté automatiquement par HAL.

Utilisez `fdisk` (ou `cdisk`, ou ce que vous voulez) pour partitionner la clef. Créez autant de partitions que vous voulez de versions différentes de l'installateur Slackware. Les partitions doivent bien sûr être assez grandes pour accueillir l'image de l'installateur, et un peu de marge ne peut pas faire de mal — pour ma part j'opte pour des partitions de 50 Mo, soit le double du minimum requis. Inutile de créer un système de fichiers sur ces partitions.

Copiez les installeurs Slackware sur chacune des partitions nouvellement créées :

```
# dd if=slackware-13.1/usb-and-pxe-installers/usbboot.img of=/dev/sdb1
# dd if=slackware64-13.1/usb-and-pxe-installers/usbboot.img of=/dev/sdb2
```



Je vous laisse vérifier, avant de préférence, que `/dev/sdb` correspond bien à votre clef USB, et non par exemple à votre deuxième disque dur SATA...

À présent, montez les partitions et examinez leur contenu. Elles doivent contenir, entre autres fichiers, un noyau Linux. Lorsque l'image de l'installateur est copiée sur une clef non-partitionnée, le MBR de la clef contient le code nécessaire pour trouver et charger ce noyau au démarrage de la machine. Maintenant que nous avons plusieurs noyaux sur plusieurs partitions, pour pouvoir démarrer l'un d'eux nous avons besoin d'un chargeur de démarrage pour :

- afficher un prompt ou un menu permettant de sélectionner le noyau (et donc l'installateur) à démarrer ;
- charger effectivement le noyau sélectionné.

## 1.2. Installation de LILO

Créez un fichier de configuration pour LILO sur l'une quelconque des partitions de la clef. Ajoutez une section pour chaque installateur, en précisant le chemin vers leur noyau Linux et l'*initial ramdisk* correspondant. Par exemple, en supposant que la clef USB est `/dev/sdb` et que les partitions des installeurs 32 et 64 bits sont montées respectivement sur `/mnt/tmp/slack` et `/mnt/tmp/slack64` :

```
boot=/dev/sdb
compact
lba32
vga=769
prompt
timeout=20
install=text

# Slackware 13.1 installer
image=/mnt/tmp/slack/hugesmp.s
  initrd=/mnt/tmp/slack/initrd.img
  label=slack

# Slackware64 13.1 installer
image=/mnt/tmp/slack64/huge.s
  initrd=/mnt/tmp/slack64/initrd.img
  label=slack64
```

Installez ensuite LILO sur la clef, en lui précisant bien d'utiliser le fichier de configuration ci-dessus (et non le fichier `/etc/lilo.conf` de votre système) :

```
# /sbin/lilo -b /dev/sdb -C /mnt/tmp/slack/lilo.conf -m /mnt/tmp/slack/lilo.map
```

Voilà. La même clef peut désormais servir à installer les versions 32 et 64 bits de la Slackware, il suffit de sélectionner la version souhaitée au prompt de LILO.

## 2. Un peu plus loin : « multi-install » + distribution live

Plusieurs versions du programme d'installation sur une clef USB, c'est déjà bien pratique, mais à raison d'environ 25 Mo par installateur, on est loin de remplir une clef un tant soit peu récente. Qu'à cela ne tienne, il suffit d'utiliser la place encore disponible pour distribution GNU/Linux « Live » ! La clef se transforme ainsi en petit couteau suisse, permettant tantôt :

- d'installer Slackware sur une nouvelle machine ;
- d'utiliser une distribution GNU/Linux sur une machine qui n'en dispose pas, et sur laquelle on ne peut ou ne veut l'installer ;
- de dépanner une machine tombée en rade.

Parmi les distributions « live », j'ai opté pour Slax [<http://www.slax.org/>], mais ce qui suit devrait être adaptable à d'autres sans difficultés majeures.

Préparez une nouvelle partition sur la clef, d'au moins 250 Mo, et créez-y un système de fichiers VFAT. Téléchargez l'archive de la Slax for USB [[http://www.slax.org/get\\_slax.php?download=tar](http://www.slax.org/get_slax.php?download=tar)] (version 6.1.2 à l'heure où j'écris ces lignes) et détarrez-là à la racine de la partition nouvellement créée.

Ne suivez *pas* les instructions du site officiel, qui vous invitent à exécuter le script `boot/bootinst.sh` pour rendre la clef « bootable » : cela fonctionne très bien, mais ne permettrait par la suite que de démarrer Slax, laissant de côté nos installeurs Slackware sur les autres partitions.

À la place, nous allons installer nous-mêmes LILO en reprenant le fichier de configuration que nous avons déjà utilisé ci-dessus, en y ajoutant au moins une section pour démarrer Slax. En supposant que la partition de la Slax est montée sur `/mnt/tmp/slax` :

```
image=/mnt/tmp/slax/boot/vmlinuz
initrd=/mnt/tmp/slax/boot/initrd.gz
label=slax
append="ramdisk_size=6666 root=/dev/ram0 rw autoexec=xconf;telinit~4 changes=/slax/"
```

Vous pouvez ajouter d'autres section pour démarrer la Slax dans différents modes, en changeant les paramètres passés au noyau Linux (« append »). Pour connaître les modes disponibles et les paramètres correspondants, consultez le fichier `slax.cfg` fourni avec la Slax.

Reste à installer LILO. La commande est la même que précédemment. Assurez-vous que toutes les partitions de la clef (celles contenant les installeurs et celle contenant la Slax) sont montées, et que leur point de montage correspond aux chemins mentionnés dans le fichier de configuration de LILO.



Lors de son installation, LILO peut émettre le message d'avertissement suivant :

```
Warning: The initial RAM disk is too big to fit between the kernel and
the 15M-16M memory hole. It will be loaded in the highest memory as
though the configuration file specified "large-memory" and it will
be assumed that the BIOS supports memory moves above 16M.
```

qui nous apprend que la clef pourrait ne pas être bootable sur certaines vieilles machines. Vieilles comment ? Je ne sais pas. Je n'ai encore jamais rencontré ce problème avec aucun ordinateur. Je présume que la probabilité de trouver une machine concernée par ce problème est déjà faible et ne fera que diminuer encore plus avec le temps. Comme le message le laisse supposer, l'avertissement peut être supprimé en rajoutant l'option `large-memory` dans la section globale du fichier de configuration de LILO.

### 3. Ma clef polyvalente en 2013

J'ai récemment remplacé ma clef « multi-install » de 2 Go par une nouvelle clef de 16 Go. Je l'ai partitionnée comme suit :

- une partition (primaire) de 50 Mo pour un programme d'installation de Slackware, comme expliqué ci-dessus ;

## Une clef USB polyvalente à base de Slackware Linux

---

- une partition (primaire) de 6 Go pour l'intégralité des paquets de la Slackware64 14.0 ;
- une partition (primaire) de 500 Mo pour un programme d'installation via le réseau (« netinst ») de Debian 6.0.2 ;
- une partition (étendue) de 2 Go pour la distribution live Slax 7.0.8, comme expliqué ci-dessus ;
- une partition (étendue) occupant le reste de l'espace disponible (environ 7,5 Go), actuellement inutilisée et pouvant servir à stocker des données quelconques.

Cette version de la clef permet donc d'installer une Slackware64 en toute autonomie (puisqu'elle contient à la fois le programme d'installation et tous les paquets) ; elle permet également d'initier l'installation d'une Debian, une connexion Internet étant toutefois requise pour télécharger l'essentiel des paquets ; et comme précédemment, elle permet de démarrer sur un système live.

Pour mémoire, voici en bref comment reproduire cette clef, une fois le partitionnement ci-dessus réalisé (en supposant que la clef est représentée par le périphérique /dev/sdb).

Création de l'installateur Slackware :

```
# dd if=slackware64-14.0/usb-and-pxe-installers/usbboot.img of=/dev/sdb1
# mkdosfs /dev/sdb2
# mount /dev/sdb2 /mnt/tmp
# cp -r slackware64-14.0/{extra,kernels,patches,testing,slackware64} /mnt/tmp/
# umount /dev/sdb2
```

Création de l'installeur Debian :

```
# dd if=debian-6.0.2.i-386-netinst.iso of=/dev/sdb3
```

Installation de Slax :

```
# mkdosfs /dev/sdb5
# mount /dev/sdb5 /mnt/tmp
# unzip slax-English-US-7.0.8-x86_64.zip -d /mnt/tmp
# rm /mnt/tmp/slax/slax.txt
# umount /dev/sdb5
```

Pour installer le chargeur de démarrage, commencer par monter les trois partitions contenant un système bootable dans une arborescence temporaire :

```
# mkdir /mnt/tmp/{slackboot,debian,slax}
# mount /dev/sdb1 /mnt/tmp/slackboot
# mount /dev/sdb3 /mnt/tmp/debian
# mount /dev/sdb5 /mnt/tmp/slax
```

Puis, créer le fichier de configuration de LILO comme suit :

```
boot=/dev/sdb
compact
lba32
large-memory
vga=769
prompt
timeout=20
install=text

# Slackware64-14.0 installer
#####
image=/mnt/tmp/slackboot/huge.s
  initrd=/mnt/tmp/slackboot/initrd.img
  label=slackware

# Debian (i386) 6.0.2.1 (netinst)
#####
image=/mnt/tmp/debian/install.386/vmlinuz
```

```
initrd=/mnt/tmp/debian/install.386/initrd.gz
label=debian

# Slax 7.0.8
#####

# Normal mode with full autoconfiguration
image=/mnt/tmp/slax/slax/boot/vmlinuz
initrd=/mnt/tmp/slax/slax/boot/initrfs.img
label=slax
append="load_ramdisk=1 prompt_ramdisk=0 printk.time=0 slax.flags=xmode,perch"

# Normal mode without persistent changes
image=/mnt/tmp/slax/slax/boot/vmlinuz
initrd=/mnt/tmp/slax/slax/boot/initrfs.img
label=slax-fresh
append="load_ramdisk=1 prompt_ramdisk=0 printk.time=0 slax.flags=xmode"

# Normal mode + copy everything to RAM
image=/mnt/tmp/slax/slax/boot/vmlinuz
initrd=/mnt/tmp/slax/slax/boot/initrfs.img
label=slax-cp2ram
append="load_ramdisk=1 prompt_ramdisk=0 printk.time=0 slax.flags=xmode,toram"

# Console mode
image=/mnt/tmp/slax/slax/boot/vmlinuz
initrd=/mnt/tmp/slax/slax/boot/initrfs.img
label=slax-text
append="load_ramdisk=1 prompt_ramdisk=0 printk.time=0 slax.flags=perch"

Enregistrer le fichier dans /mnt/tmp/slackboot/lilo.conf puis installer effective-
ment LILO sur la clef :

# /sbin/lilo -b /dev/sdb -C /mnt/tmp/slackboot/lilo.conf -m /mnt/tmp/slackboot/lilo.map
```

## 4. Ma clef polyvalente en 2019

Six ans après la dernière mise à jour de cet article, j'utilise toujours ma « clef polyvalente », mais celle-ci a subi quelques changements majeurs, rendant les sections précédentes complètement obsolètes. Cette clef contient désormais :

- le programme d'installation de la version de développement de Slackware (*-current*) ;
- une copie complète des paquets de deux versions de Slackware (la dernière version stable, 14.2, et la version *-current*) ;
- une copie de Slackware Live Edition [<http://www.slackware.com/~alien/liveslak/>], la version « live » de Slackware conçue par Eric « Alien BOB » Hameleers.

Par ailleurs, j'utilise désormais GRUB [<https://www.gnu.org/software/grub/index.html>] au lieu de LILO et la clef peut démarrer à la fois sur un ordinateur équipé indifféremment d'un BIOS ou d'un *firmware* UEFI (les versions précédentes ne permettaient de démarrer qu'à partir d'un BIOS, ou un firmware UEFI en mode *legacy CSM*).

### 4.1. Partitionnement



Dans tout ce qui suit, on supposera, comme précédemment, que la clef USB est représentée par le périphérique `/dev/sdb`.

Utilisez l'outil de votre choix (j'utilise généralement `cgdisk` [<https://www.rodsbooks.com/gdisk/index.html>]) pour partitionner la clef, en vous assurant de créer une table de partitions au format GPT et non MBR (c'est automatiquement le cas si vous utilisez `cgdisk`). Créez les partitions suivantes sur la clef :

- une partition de type BIOS boot partition (code EF02 dans cgdisk) de 1 Mo ;
- une partition de type EFI system partition (code EF00) de 50 Mo ;
- une partition de type Linux filesystem (code 8300) de 50 Mo ; cette partition accueillera le programme d'installation de Slackware, nous l'appellerons *SLACKINST* ;
- une seconde partition de type Linux filesystem, d'au moins 4 Go (8 Go chez moi) ; cette partition accueillera les paquets de la distribution, nous l'appellerons *SLACKDATA* ;
- une troisième partition Linux filesystem d'au moins 2 Go, pour accueillir la distribution Slackware Live ; nous l'appellerons *SLACKLIVE*.

S'il vous reste encore de l'espace libre, vous pouvez soit augmenter la taille de la partition *SLACKLIVE*, soit créer des partitions supplémentaires à votre guise. Personnellement j'ai ajouté une partition de type Microsoft basic data (code 0700).

Formatez les partitions nouvellement créées à l'exception de la première (celle de type BIOS boot partition : cette partition ne sera utilisée que par GRUB, qui n'a pas besoin qu'un système de fichiers y soit présent) :

```
# mkfs.fat -F32 /dev/sdb2
# mkfs.ext4 /dev/sdb3
# mkfs.ext4 /dev/sdb4
# mkfs.ext4 /dev/sdb5
# mkfs.fat /dev/sdb6
```

Créez une série de dossiers qui serviront de points de montage pour chacune des partitions pour les prochaines opérations :

```
# mkdir /mnt/multikey/{efi,slackinst,slackdata,slacklive}
# mount /dev/sdb2 /mnt/multikey/efi
# mount /dev/sdb3 /mnt/multikey/slackinst
# mount /dev/sdb4 /mnt/multikey/slackdata
# mount /dev/sdb5 /mnt/multikey/slacklive
```

## 4.2. Copie de l'installateur Slackware

Montez le fichier `usb-and-pxe-installers/usbboot.img` (dans l'arborescence de la distribution Slackware) sur un point de montage temporaire et copiez le contenu de l'image sur la troisième partition de la clef :

```
# mount -o loop $SLACKROOT/usb-and-pxe-installers/usbboot.img /mnt/tmp
# rsync -r --exclude=EFI /mnt/tmp/ /mnt/multikey/slackinst
# umount /mnt/tmp
```

Puis, copiez les paquets de la distribution vers la partition *SLACKDATA* :

```
# mkdir -p /mnt/multikey/slackdata/slackware64-14.2
# rsync -r $SLACKROOT/slackware64 /mnt/multikey/slackdata/slackware64-14.2
```

Si la taille de la clef vous le permet, vous pouvez copier les paquets de plus d'une seule version de Slackware, par exemple la dernière stable (14.2 à l'heure où ces lignes sont écrites) et un « snapshot » de la *-current*. Lors de l'installation, vous pourrez choisir quelle version de Slackware installer simplement en choisissant le dossier correspondant lorsque l'installateur vous demandera où trouver les paquets.

## 4.3. Copie de la Slackware Live Edition

Téléchargez l'image ISO de la Slackware Live Edition ou créez votre propre image avec les scripts du projet Liveslak.

Liveslak fournit un script `iso2usb.sh` conçu pour générer automatiquement une clef USB bootable à partir d'une image ISO de la Slackware Live Edition. Toutefois ce script n'est pas assez flexible pour ce que nous voulons faire ici, nous nous en passerons donc et à la place allons copier « manuellement » les fichiers de la Slackware Live Edition avant de rendre la clef bootable.

Montez l'image de la Slackware Live Edition sur `/mnt/cdrom` et copiez son contenu vers la partition `SLACKLIVE` créée précédemment :

```
# mount -o loop slackware-live-xfce-current.iso /mnt/cdrom
# rsync -rlptD --no-inc-recursive --info=progress2 --exclude=EFI \
  /mnt/cdrom/* /mnt/multikey/slacklive
```

#### 4.4. Installation de GRUB

Installez sur la partition EFI une version de GRUB ciblant les systèmes UEFI :

```
# grub-install --target=x86_64-efi --recheck --removable \
  --efi-directory=/mnt/multikey/efi --boot-directory=/mnt/multikey/efi
```

Assurez-vous que le binaire EFI de GRUB a été installé à l'emplacement `/mnt/multikey/efi/EFI/BOOT/BOOTX64.EFI`.

Installez GRUB une seconde fois, cette fois-ci en ciblant les systèmes utilisant toujours un BIOS:

```
# grub-install --target=i386-pc --recheck \
  --boot-directory=/mnt/multikey/efi /dev/sdb
```

Cela permettra de démarrer sur la clef quel que soit le mode de démarrage de la machine cible (UEFI ou BIOS). Les deux variantes de GRUB partageront une même configuration, dans le dossier `/mnt/multikey/efi/grub`.

Ne reste plus qu'à configurer GRUB, justement. Au préalable, obtenez les UUID des systèmes de fichier sur les partitions `SLACKINST` et `SLACKLIVE` :

```
# blkid /dev/sdb3
/dev/sdb3: UUID="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
# blkid /dev/sdb5
/dev/sdb5: UUID="zzzzzzzz-zzzz-zzzz-zzzz-zzzzzzzzzzzz"
```

Vous pouvez ensuite créer un fichier de configuration minimaliste pour GRUB tel que celui-ci :

```
set default=0
set timeout=10

insmod all_video
insmod ext2
insmod search_fs_uuid

menuentry "Slackware installer" --hotkey i {
  search --no-floppy --set=root --fs-uuid xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
  linux /huge.s
  initrd /initrd.img
}

menuentry "Slackware Live Edition" --hotkey b {
  search --no-floppy --set=root --fs-uuid zzzzzzzz-zzzz-zzzz-zzzz-zzzzzzzzzzzz
  linux /boot/generic load_ramdisk=1 prompt_ramdisk=0 rw printk.time=0
  initrd /boot/initrd.img
}
```

N'hésitez pas à consulter le fichier `EFI/BOOT/grub.cfg` fourni avec le projet Liveslak pour un exemple de configuration plus avancée, permettant notamment de choisir la disposition du clavier et la langue du système avant de démarrer la Slackware Live Edition.