

Installation Debian sans le réseau immédiat

- Objet : Installation debian sur eeepc sans le réseau immédiat
- Niveau requis :
[débutant, avisé](#)
- Commentaires : *Installer une Debian sans avoir de réseau*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
[à-tester](#)
 - Création par [smolski](#) le 14/04/2010
 - Mise à jour par [darien](#) le 10/06/2011
 - Testé par le
- Commentaires sur le forum : [C'est ici^{1\)}](#)

Introduction

Donc vous n'avez pas de réseau. C'est gênant. Mais vous pouvez quand même installer une Debian GNU/Linux avec un bureau complet. En effet, le premier CD de la distribution contient par exemple les logiciels les plus utilisés ainsi qu'un bureau Gnome.

Vous pouvez aussi appliquer la même méthode avec le CD d'installation KDE ou LXDE+Xfce, et comme on est fainéant et que c'est plus sûr, on va réutiliser le travail déjà fait à partir de ces images iso.

Installation

Préparation de l'environnement de travail

Mise à jour :

```
apt-get update
```

Les outils :

```
apt-get install mtools dosfstools syslinux mbr
```

Créer le répertoire de travail et s'y placer :

```
mkdir ~/debian && cd ~/debian
```

Syntaxe des commandes :

Téléchargement boot

```
wget  
http://ftp.fr.debian.org/debian/dists/stable/main/installer-i386/current/images/hd-media/boot.img.gz
```

Extraire l'image du boot :

```
gunzip boot.img.gz
```

Image iso

Ensuite, il vous faut une image iso de la branche stable du moment.
Pour choisir l'image iso de la branche stable actuelle, voir la page wiki :

- [Installation de debian pas à pas](#)



Pensez à utiliser des méthodes de téléchargement alternatives, comme [bittorrent](#) ou [Jigdo](#). En effet, ces méthodes réduisent la charge des serveurs de fichiers du projet debian.



Le plus important est sans doute de choisir une image CD qui soit de la même version que le fichier boot.img.gz téléchargé plus haut. Ceci pour pouvoir utiliser des modules supplémentaires lors du processus d'installation.

Identification de la clé USB

Branchez votre clé USB, et tapez dans un terminal :

Méthode 1

```
dmesg | tail
```

[retour de la commande](#)

```
[2165634.791468] scsi 5:0:0:0: Direct-Access          USB DISK 2.0  
PMAP PQ: 0 ANSI: 0 CCS  
[2165634.802119] sd 5:0:0:0: Attached scsi generic sg3 type 0  
[2165635.042119] sd 5:0:0:0: [sdc] 8036352 512-byte logical blocks:  
(4.11 GB/3.83 GiB)  
[2165635.042726] sd 5:0:0:0: [sdc] Write Protect is off  
[2165635.042738] sd 5:0:0:0: [sdc] Mode Sense: 23 00 00 00
```

```
[2165635.042743] sd 5:0:0:0: [sdc] Assuming drive cache: write through
[2165635.051244] sd 5:0:0:0: [sdc] Assuming drive cache: write through
[2165635.051264]   sdc: sdc1
[2165635.106004] sd 5:0:0:0: [sdc] Assuming drive cache: write through
[2165635.106019] sd 5:0:0:0: [sdc] Attached SCSI removable disk
```

Méthode 2

```
ls -l /dev/disk/by-id/*usb*
```

[retour de la commande](#)

```
lrwxrwxrwx 1 root root 9 jan 22 12:59 /dev/disk/by-id/usb-
disk2go_disk2go_PURE_S2_0B61EB60F1A12983-0:0 -> ../../sdc
lrwxrwxrwx 1 root root 10 jan 22 12:59 /dev/disk/by-id/usb-
disk2go_disk2go_PURE_S2_0B61EB60F1A12983-0:0-part1 -> ../../sdc1
```

Identification

Ici, notre périphérique est donc `/dev/sdc`.



Il est crucial d'identifier correctement votre clé USB, **sous peine de détruire la structure d'un de vos disques durs**. Vous avez été prévenu.

Formatage de la clé

Ici nous avons une clé de 4GB. Il serait bête de ne créer qu'une partition de quatre gigaoctets, quand on pourrait faire les choses proprement.

Nous allons donc diviser ce "disque dur" en deux partitions :

1. une de 700 Mo, bootable, en FAT 16,
2. l'autre partition dépend de vos goûts / besoins...

Vous pouvez décider d'utiliser votre outil favori : gparted, kparted, parted... Ou mieux, en ligne de commande avec [cfdisk](#) !

Pour ma part, c'est [fdisk](#) et **mkdosfs** :

```
install-mbr /dev/sdc
```

```
fdisk /dev/sdc
```

[séquence interactive](#)

Command (m for help): p

Disk /dev/sdc: 4114 MB, 4114612224 bytes
 127 heads, 62 sectors/track, 1020 cylinders
 Units = cylinders of 7874 * 512 = 4031488 bytes
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 1

First cylinder (1-1020, default 1): 1

Last cylinder or +size or +sizeM or +sizeK (1-1020, default 1020):
 +700M

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 2

First cylinder (176-1020, default 176):

Using default value 176

Last cylinder or +size or +sizeM or +sizeK (176-1020, default 1020):
 Using default value 1020

Command (m for help): p

Disk /dev/sdc: 4114 MB, 4114612224 bytes
 127 heads, 62 sectors/track, 1020 cylinders
 Units = cylinders of 7874 * 512 = 4031488 bytes
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		1	175	688944	83	Linux
/dev/sdc2		176	1020	3326765	83	Linux

Command (m for help): t

Partition number (1-4): 1

Hex code (type L to list codes): 6

Changed system type of partition 1 to 6 (FAT16)

Command (m for help): a

Partition number (1-4): 1

```
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): c
Changed system type of partition 2 to c (W95 FAT32 (LBA))

Command (m for help): p

Disk /dev/sdc: 4114 MB, 4114612224 bytes
127 heads, 62 sectors/track, 1020 cylinders
Units = cylinders of 7874 * 512 = 4031488 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1   *           1          175        688944    6   FAT16
/dev/sdc2             176         1020       3326765    c   W95 FAT32 (LBA)

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: If you have created or modified any DOS 6.x
partitions, please see the fdisk manual page for additional
information.
Syncing disks.
```

```
mkdosfs -F 16 /dev/sdc1
```

```
mkdosfs -F 32 -n usb-data /dev/sdc2
```

```
exit
```

Notre clé est prête. Nous avons créé une première partition bootable, et l'avons formaté en FAT 16. Nous l'avons rendu bootable avec la commande `fdisk 'a'`.



À noter que nous avons commencé par réinstaller un MBR vide à l'aide de la commande `install-mbr`, ceci afin d'éviter tout tracas éventuel en cas de MBR défectueux.

Nous avons aussi créé une partition FAT 32 avec un label "usb-data".

Installation de syslinux sur la clé

Maintenant, passons aux choses sérieuses.

Nous allons utiliser le noyau ainsi que les fichiers donnés dans le `boot.img.gz` précédemment téléchargé pour construire notre menu de démarrage.

Mais d'abord, un coup de syslinux :

```
syslinux /dev/sdc1
```

```
pwd
```

[retour de la commande](#)

```
/home/user/debian/
```

```
ls
```

[retour de la commande](#)

```
boot.img debian-505-i386-CD-1.iso
```

```
mkdir loopdir
```

```
mount -o loop boot.img ./loopdir
```

```
mount /dev/sdc1 /mnt
```

```
cp -R ./loopdir/* /mnt
```

```
cp ./debian-505-i386-CD-1.iso /mnt
```

```
sync
```

```
umount /mnt
```

```
exit
```

Bon bah... y'a plus qu'à essayer ! 😊

A noter que certaines vieilles cartes mères peuvent ne pas supporter le boot via USB.



1)

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

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

Permanent link:
<http://debian-facile.org/doc:install:install-usb-no-reseau>

Last update: **26/11/2015 18:10**



