



- **Scanner sur port parallèle** : Ici c'est un tout petit peu plus compliqué : en effet sans scanner ne scanne pas les ports parallèles pour détecter le scanner. Il va donc falloir utiliser une petite ruse de sioux consistant à émuler le scanner comme SCSI. Le modèle de scanner testé sur cette partie est un Microtek Phantom 330CX sur une Mandriva 9.1. ( il faut ajouter que d'autres scanners fonctionnant sur un port parallèle peuvent nécessiter un autre type d'installation, par exemple le chargement d'un firmware spécifique). Pour l'installation de ce scanner, il faut veiller à ce que votre distribution de préférence récente dispose des modules suivants : `ppscsi` et `onscsi`. Donc le fonctionnement de ce scanner microtek s'effectue de la manière suivante, loggez-vous en root dans une console et tapez :

```
insmod parport_pc
insmod sg
insmod ppscsi
insmod onscsi
```

Vous noterez que le chargement des deux derniers modules s'effectuent avec des avertissements. Cela n'empêchera pas ce scanner de fonctionner. Il est à noter que si vous possédez une imprimante fonctionnant sur le port parallèle, l'émulation scsi de celui-ci ne portera pas préjudice à votre imprimante. Sous Linux ce scanner fonctionne de la même manière que sous windows ; en effet, sous ce système d'exploitation le pilote vous installera une carte scsi virtuelle.

- **Pour tous** : Il vous faut le module **scanner** de monté. Pour vérifier ce fait :

```
lsmod | grep scanner
```

Si la commande ne vous renvoie rien, il vous faut alors le monter par :

```
modprobe scanner
```

Après cela vérifier à nouveau que le module est bien monté avec la commande `lsmod | grep scanner`. Si cela n'était pas le cas, il y a de fortes chances pour que le module n'est pas été compilé avec le noyau. Pour vous en assurer, taper la commande suivante :

```
cat /boot/config-ide-x.x.xx | grep -i scanner
```

Celle-ci devrait vous renvoyer quelque chose comme :

```
#CONFIG_USB_SCANNER is not set
```

Ce qui signifie qu'il vous faut recompiler le noyau avec la bonne option. En ce cas suivez le guide : [ici](http://www.trustonme.net/didactels/285.html)<sup>3</sup>. Toutefois si vous optez une ligne non commenté misé à yes (y), cela signifie que le pilote est en dur avec le noyau donc rien à faire. Dans tous les cas de figure, s'il vous a fallu monter un ou des modules, pour que ce(s) dernier(s) soi(en)t pris en compte à chaque démarrage, il vous faut éditer en tant que root le fichier `/etc/rc.d/rc.local` et ajouter la ligne suivante :

```
/sbin/insmode nom_module
```

Une autre méthode consiste à utiliser les outils spécifiques à votre distribution.

---

<sup>3</sup><http://www.trustonme.net/didactels/285.html>

### 0.1.3 1.2 Vérification pour le device :

Selon que votre scanner soit USB, SCSI ou parallèle, le device du scanner sera différent. Dans la plupart des distributions, les devices sont déjà existants mais ne possèdent pas les droits nécessaires pour permettre l'utilisation du scanner par un simple utilisateur. Nous allons ici vérifier selon le type du scanner l'existence du device ainsi que les droits qui lui sont attribués.

– **Pour un scanner USB** : Les devices utilisés sont :

- /dev/usbscanner
- /dev/usb/scanner
- /dev/usb/scannerX

X valant 0,1,2, etc... Pour vérifier l'existence de l'un de ces devices (un seul suffit) en même temps que les droits attribués, saisissez les commandes :

```
ls -l /dev/usbscanner
ls -l /dev/usb/scanner
ls -l /dev/usb/scanner0
```

Si l'une ou plusieurs des commandes vous certifie l'existence du device, vérifiez que les droits sur celui-ci soient bien en lecture et écriture (rw) pour tous les utilisateurs (à moins de fonctionner par groupe). Si tel n'est pas le cas alors saisissez (à adapter) :

```
su
chmod 666 /dev/usbscanner
chmod 666 /dev/usb/scanner
chmod 666 /dev/usb/scannerX
exit
```

X valant 0,1,2, etc... Au contraire, si toutes les commandes "ls -l /dev/fichier" restent muettes, alors il vous faut le créer avec celles-ci :

```
su
mkdir /dev/usb
mknod -m 666 /dev/usb/scanner0 c 180 48
exit
```

Petites remarques : le répertoire /dev/usb existe peut-être déjà, en ce cas inutile de le créer. De plus l'option **-m 666** permet de donner les droits de lecture et d'écriture sur le device à tous les utilisateurs, si vous préférez fonctionner par les groupes rien ne vous en empêche en adaptant l'option à **-m 660**.

– **Pour un scanner SCSI** : Les devices utilisés sont :

- /dev/scanner
- /dev/sgX

X valant 0,1,2, etc... Vérifiez qu'il en existe au moins un avec des droits de lecture et d'écriture pour tous les utilisateurs avec :

```
ls -l /dev/scanner
ls -l /dev/sgX
```

En cas de droits insuffisamment :

```
su
chmod 666 /dev/scanner
```

```
chmod 666 /dev/sg0
exit
```

**ATTENTION** : il est possible que **/dev/sg0** existe mais soit utilisé pour un autre périphérique que le scanner, en ce cas ne changez en rien ces droits, et adaptez avec **/dev/sg1**. De même si **/dev/sg1** est aussi utilisé, prenez le suivant ;-). Si les fichiers **/dev/scanner** et **/dev/sg0** n'existent pas, alors il vous faut créer le device de la façon suivante :

```
su
mknod -m 666 /dev/sg0 c 180 0
ln -s sg0 /dev/scanner
exit
```

De même que précédemment, si vous préférez fonctionner par groupe, rien ne vous en empêche en adaptant l'option **-m** de la commande **mknod**.

- **Pour un scanner Parallèle** : Comme nous l'avons vu dans la partie 1.1, un scanner sur port parallèle va être émulé scsi, donc ici reportez-vous à la partie dédiée au scanner scsi.

#### 0.1.4 2. Installation de sane et de sa suite :

La première chose à faire est de vérifier que votre scanner est bien reconnu par sane, vous pouvez consulter cette page<sup>4</sup> pour vous en informer. Par exemple pour moi, je consulte les scanners Mustek, et y découvre que le ScanExpress 1200 UB est en version stable. À l'occasion je télécharge le fichier présent dans la colonne **backend** (avant dernière) (attention vous êtes redirigé vers une nouvelle page spécifique au pilote). Cet étape n'est pas une obligation mais permet simplement de bénéficier du dernier pilote sans devoir attendre une nouvelle sortie de sane. Maintenant il vous faut télécharger les fichiers sane-backends, sane-frontends ou xsane dans leur dernières versions présentes sur l'un des ftps de cette page : ici<sup>5</sup>.

#### 0.1.5 2.1 Installation de sane :

Je supposerais ici que vous avez téléchargé le fichier backend propre à votre scanner. Pour les besoins la cause je le nommerai **fichier\_backend.tar**. L'installation de sane s'effectue par cette suite de commandes à taper dans un terminal, inutile de vous dire d'adapter les chemins des répertoires :

```
tar -xzf /le/chemin/vers/sane-backends-x.x.xx.tar.gz
cd sane-backends-x.x.xx
rm -fr backend/fichier_backend.*
tar -xzf /vers/où/est/fichier_backend.tar.bz
./configure --prefix=/usr --sysconfdir=/etc
make
su
make install
exit
```

<sup>4</sup><http://www.sane-project.org/sane-mfgs.html>

<sup>5</sup><http://www.sane-project.org/source.html>

Maintenant vous pouvez tester si l'installation s'est bien déroulée et si votre scanner est bien pris en charge par sane. Pour se faire, saisissez la commande suivante :

```
sane-find-scanner
```

Celle-ci devrait vous renvoyer quelque chose ressemblant à ce dialogue :

```
# No SCSI scanners found. If you expected something
  different, make sure that
## you have loaded a SCSI driver for your SCSI adapter.
## Also you need support for SCSI Generic (sg) in your
  operating system.
## If using Linux, try "modprobe sg".
found USB scanner (vendor=0x055f, product=0x0006) at /dev/usb/scanner0
## Your USB scanner was detected. It may or may not
  be supported by
## SANE. Try scanimage -L and read the backend's manpage.
## Scanners connected to the parallel port or other
  proprietary ports can't be
## detected by this program.
## You may want to run this program as root to find
  all devices. Once you
## found the scanner devices, be sure to adjust access
  permissions as
## necessary.
```

Et vous pouvez alors vous écrire "Gagné !" ;-). Si tel n'est pas le cas, vérifier ces quelques points :

- Que les modules concernant l'usb (usbcore, uhci ou usb-uhci) soient bien montés ainsi que le module scanner.
- Vérifiez que l'utilisateur possède les bons droits sur le device (exemple : /dev/usb/scanner0)
- Enfin vérifiez que votre scanner est bien pris en charge en référant au site officiel<sup>6</sup>. (lien mort !)

Si tout vous semble correct et que pourtant vous n'arrivez pas à détecter votre scanner, il install\_sane se pourrait qu'il vous manque le firmware. Ici je ne m'avancerai pas n'ayant pas rencontré ce problème (hélas ou heureusement ?) mais par contre je suis preneur de tout ajout sur le didacticiel sur ce sujet...

### 0.1.6 2.2 Cas des scanners AGFA snapscan :

Suite à un topic du forum de [email]leo\_nard@no-log.org[/email], je décrirais ici l'installation du firmware pour le scanner AGFA snapscan 1212U. Normalement cette méthode est généralisable pour tous les scanners AGFA utilisant snapscan. Allez à cette page<sup>7</sup> puis recherchez votre scanner et consultez l'avant dernière colonne qui vous donne le nom du firmware que je nommerai pour les besoins de la cause **fichier\_firmware**. Si rien n'est inscrit, c'est que votre scanner n'en nécessite pas. Une fois que vous avez ce nom, il va falloir trouver le fichier pour pouvoir en disposer sur votre distribution et c'est bien là la tâche la plus difficile. Deux cas de figure se présentent :

<sup>6</sup><http://www.mostang.com/sane/sane-mfgs.html>

<sup>7</sup><http://snapscan.sourceforge.net/>

- Vous possédez windows : en ce cas rien de plus simple : recherchez le fichier sur windows et recopiez le par exemple dans votre \$HOME (/home/votre\_login). Par exemple Léo le trouve dans le répertoire `C:\Program Files\Agfa\ScanWise 1_20`.
- Vous n'avez pas windows : en ce cas vous rentrez dans la lutte :-D : en effet ces fichiers étant protégés généralement par des droits de propriétés, cela interdit théoriquement leur diffusion. Vous pouvez toujours essayer de le trouver sur le web tout de même, autrement essayer de prendre contact avec une personne pouvant vous le fournir.

De plus, il vous faut télécharger le fichier **agfafirm** ou **acerfirm** selon votre scanner. Ils sont présents dans la partie "Scanner firmware" de la page précédente. Passons maintenant à l'installation du firmware. Je supposerai que le fichier `fichier_firmware` ainsi que `agfafirm` ou `acerfirm` sont présents dans votre \$HOME. Si tel n'est pas le cas, adaptez. Vous devez saisir dans un terminal :

- Pour les utilisateurs de `agfafirm` :

```
cd
su
chmod a+x agfafirm
./agfafirm /dev/usb/scanner0 "SnapScan fichier_firmware"
exit
```

- Pour les utilisateurs de `acerfirm` :

```
cd
su
chmod a+x acerfirm
./acerfirm /dev/usb/scanner0 "SnapScan fichier_firmware"
exit
```

Attention j'ai supposé que le scanner avait pour device `/dev/usb/scanner0`, cela n'est peut-être pas le cas, il vous faut vous reporter à la partie 2.1 et au résultat de la commande `sane-find-scanner`. Maintenant il reste encore deux petites choses à faire : placer le firmware au bon endroit et éditer le fichier de configuration de `snapscan` pour lui indiquer l'emplacement du firmware. L'étape 1 se réalise facilement de la façon suivante :

```
cp /où_est/fichier_firmware /etc/sane.d/
```

Pour la seconde étape, éditez le fichier `/etc/sane.d/snapscan.conf` à l'aide de `kate`, `nano`, `vi` ou tout autre éditeur, et remplacez la ligne :

```
firmware /path/to/your/firmware/file.bin
```

par la ligne suivante :

```
firmware ./fichier_firmware
```

Remplacez bien **fichier\_firmware** par le vrai nom, dans le cas de Léo : `1212U_2.bin`. Si le nom de votre firmware possède des espaces blancs, supprimez-les en les remplaçant par des "\_", et adaptez en conséquence dans le fichier `snapscan.conf`. Cette étape réalisée, votre scanner devrait être exploitable.

### 0.1.7 2.3 Installation d'un scanner basé sur chipset niash :

Après avoir téléchargé les fichiers sane-backends, sane-frontends ou xsane dans leur dernières versions, Il vous faudra télécharger le pilote testtool ici<sup>8</sup>. Procédons maintenant à l'installation du pilote. L'exemple qui suit fait en sorte que que tout les fichiers qui vont être détaré sont dans le même dossier. L'installation qui suit se faire a partir d'un terminal. Veuillez taper les commandes ci dessous :

```
$ tar -xvzf /le/chemin/vers/testtool-xxxxxxx_x.tar.gz
$ cd testtool
$ vi Makefile
```

La dernière ligne permet l'édition du fichier Makefile. Aussi vous pouvez remplacer vi par kwrite, kate, gedit ou par votre éditeur préféré. Il vous faut modifier le passage suivant :

```
CFLAGS += -DLIBUSB_SUPPORT
LDFLAGS += -lusb
```

de la manière suivante (autrement dit le mettre en commentaire) :

```
#CFLAGS += -DLIBUSB_SUPPORT
#LDFLAGS += -lusb
```

Il vous faut ensuite éditer le fichier hp3300c\_xfer\_cfg.h. Vous pouvez utiliser vi et dans le terminal taper :

```
$ vi hp3300c_xfer_cfg.h
```

ou utilisez n'importe quel autre éditeur. Remplacez alors :

```
/* #define LINUX_USB_SUPPORT */
```

par :

```
#define LINUX_USB_SUPPORT
```

Terminez ensuite en tapant les deux commandes suivantes :

```
$ make clean
$ make
```

Maintenant il nous reste une dernière phase : patcher SaneBackend. Détarez le code source de SaneBackend et utilisez "patch-sane.sh". Ceci se fait en saisissant les commandes suivantes dans le terminal :

```
$ cd
$ tar -xvzf sane-backend-x.x.x.tar.gz
$ cd /le/chemin/vers/testtool/
$ ./patch-sane.sh ~/sane-backend-x.x.x
```

Maintenant SaneBackend est patcher pour supporté les scanner sur chipset niash. Il vous faut alors copier le fichier "hp3300c\_linux.c" dans /le/chemin/vers/sane-backends-x.x.x/backend :

```
$ cp hp3300c_linux.c ~/sane-backends-x.x.x/backend/hp3300c_linux.c
$ cd ~/sane-backends-x.x.x/
```

À partir d'ici, vous pouvez reprendre le didacticiel normalement avec l'installation de sanebackend (partie 2.1) et de sa suite.

<sup>8</sup>[http://prdownloads.sourceforge.net/hp3300backend/testtool-20030401\\_1.tar.gz?download](http://prdownloads.sourceforge.net/hp3300backend/testtool-20030401_1.tar.gz?download)

### 0.1.8 2.4 Installation des autres paquetages :

Les plus grosses difficultés ont été passées. Maintenant il ne vous reste plus qu'à installer sane-frontends **ou** xsane. Le paquetage sane-frontends nécessite une version de GTK égale ou supérieure à la 0.99.13 (oubliez les 2.x). Ces deux paquetages apportent une gui à sane, celle de xsane étant la plus avancée. Vous pourrez trouver les paquetages sur l'un des ftps de cette page<sup>9</sup>. (!lien mort !)

- Pour ceux préférant sane-frontends :

```
tar -xvzf /Où/est/sane-frontends-x.x.xx.tar.gz
cd sane-frontends-x.x.xx
./configure --prefix=/usr
make
su
make install
exit
```

- Pour ceux choisissant xsane :

```
tar -xvzf /Où/est/xsane-x.xx.tar.gz
cd xsane-x.xx
./configure --prefix=/usr
make
su
make install
exit
```

Vous pouvez bénéficier de votre scanner avec gimp. Pour cela :

```
ln -s /usr/bin/xscanimage /usr/lib/gimp/x.x/plug-ins/xscanimage
```

### 0.1.9 3. Utilisation :

Pour scanner, vous avez le choix entre scanimage installé pas sane en ligne de commande, ou bien encore xsane en mode graphique. Pour scanimage, la ligne de commande la plus simple est la suivante :

```
scanimage --mode color > mon_fichier.pnm
```

Le format par défaut étant le .pnm. Vous pouvez convertir ce dernier en .png à l'aide du programme convert de la suite ImageMagic par :

```
convert mon_fichier.pnm mon_fichier.png
```

scanimage admet beaucoup d'options permettant de jouer sur le gamma et autres réjouissances. Je vous laisse le soin de consulter le manuel avec la commande :

```
man scanimage
```

Si vous choisissez d'utiliser xsane, il vous suffit de le lancer dans le terminal avec la commande :

```
xsane &
```

Vous pouvez faire une pré-visualisation avant l'acquisition afin d'effectuer les réglages qui vous semblent les plus justes. Le programme est très facile d'utilisation.

<sup>9</sup><http://www.mostang.com/sane/source.html>