

Petit guide d'installation d'Archlinux avec Gnome 3.6+

Dans ce petit guide, je vais détailler l'installation d'Archlinux avec Gnome 3.6 et suivante. L'installation terminée proposera un environnement suffisamment étoffé pour être utilisable. C'est une adaptation des deux précédents guide.

NOTE : La machine virtuelle utilise un BIOS « classique ». Si vous utilisez une machine en UEFI, il faudra se conférer au guide officiel en anglais pour les spécificités au niveau du partitionnement et de l'installation du gestionnaire de démarrage : https://wiki.archlinux.org/index.php/Beginners%27_Guide

Cependant, certains points plus techniques et / ou plus particuliers ont été volontairement oubliés : ce sont les manipulations liées à l'installation du pare-feu, ou encore du matériel comme des imprimantes avec scanners.

Je vous renvoie aux wiki anglophone <https://wiki.archlinux.org/> et francophone <http://wiki.archlinux.fr/Accueil> pour ce genre de manipulations.

La variété du dernier point ne rentre pas en effet dans le cadre de ce petit guide.

Pour cette version du tutoriel, je me suis basé sur la dernière ISO officielle, celle qui utilise les scripts d'installation. Celle datée de début Mars 2013, la 2013.03.01.

Enfin, j'ai utilisé une machine virtuelle pour la simple et bonne raison que cela permet d'avoir un environnement standardisé. Si vous suivez ce guide pour installer sur une machine réelle, il faudra penser à prendre des notes sur le matériel, spécialement votre carte vidéo.

Ce document est proposé sous licence CC-BY-SA.

<http://creativecommons.org/licenses/by-sa/3.0/>

Il a été créé avec LibreOffice 4.0.1, avec une distribution Archlinux, Gnome 3.6+ et les outils proposés.

I. Installons notre base.

Installer une ArchLinux, c'est comme construire une maison. On commence par les fondations, et on rajoute les murs et le reste par la suite. L'image ISO utilisée est la archlinux-2013.03.01-dual.iso

La machine virtuelle est une session Qemu à laquelle j'ai rajouté un disque virtuel de 128 Go. L'installation est comparable dans une machine VirtualBox, sauf précision complémentaire.

Voici donc le premier écran qui nous accueille.

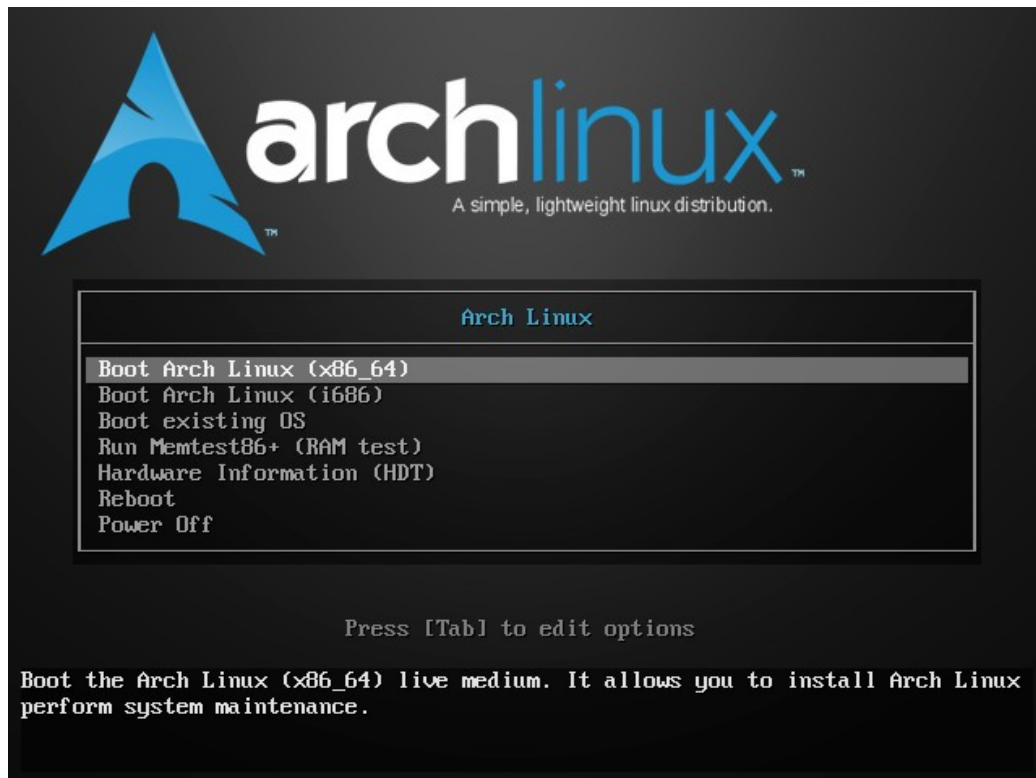


Illustration 1: Ecran de démarrage

On nous demande simplement si on veut faire une installation 32 bits, une 64 bits, avec un noyau « normal » ou à support « Long Terme » (LTS).

Pour les besoins du guide, j'ai pris une version 64 bits. Sachant que mis à part des outils spécifique comme Skype, le 64 bits n'a plus à rougir devant une version 32 bits coté équipement logiciel.

Donc si votre processeur supporte le 64 bits, n'hésitez pas ! Il est toujours possible d'utiliser des logiciels 32 bits sur un système 64 bits, nous verrons cela à la fin de l'étape d'installation.

La première chose à faire, c'est d'avoir le clavier français :

```
loadkeys fr
```

Nous passons ensuite à la première étape délicate, à savoir le partitionnement. Utilisant une machine virtuelle, je n'ai pas eu à me soucier outre mesure de ce point précis. Si vous utilisez une machine ayant déjà MS-Windows, il vous faudra avoir créé une partition auparavant... Ou utiliser un disque dédié.

Dans le cas de notre tutoriel, je vais utiliser le partitionnement suivant :

- partition /boot en /dev/sda1, 512 Mo en ext2
- une partition de swap d'au moins la taille de la mémoire vive en /dev/sda2
- une partition / en /dev/sda3 de 20 Go en ext4
- le reste sera la partition /home, en ext4

On peut utiliser l'outil de son choix, je suis resté classique en utilisant ce bon vieux cfdisk.

```
cfdisk (util-linux 2.22.2)

Disk Drive: /dev/sda
Size: 137438953472 bytes, 137.4 GB
Heads: 255 Sectors per Track: 63 Cylinders: 16709

Name      Flags      Part Type  FS Type      [Label]      Size (MB)
-----
sda1      Boot       Primary   Linux         509.97
sda2              Primary   Linux swap / Solaris  2048.10
sda3              Primary   Linux         20480.95
sda4              Primary   Linux         114399.95*

[ Bootable ] [ Delete ] [ Help ] [ Maximize ] [ Print ]
[ Quit ] [ Type ] [ Units ] [ Write ]

Quit program without writing partition table_
```

Illustration 2: Partitionnement avec cfdisk

On va formater les partitions :

```
mkfs.ext2 /dev/sda1
mkswap /dev/sda2
swapon /dev/sda2
mkfs.ext4 /dev/sda3
mkfs.ext4 /dev/sda4
```

On va ensuite créer les points de montages et y associer les partitions qui correspondent.

```
mount /dev/sda3 /mnt
mkdir /mnt/{boot,home}
mount /dev/sda1 /mnt/boot
mount /dev/sda4 /mnt/home
```

La connexion est normalement fonctionnelle par défaut. Pour choisir le miroir le plus rapide, il suffit de jeter un œil dans le fichier /etc/pacman.d/mirrorlist

Celui d'archlinux.fr, car c'est le plus rapide pour moi. Ensuite, c'est à vous de voir celui qui vous est le plus proche géographiquement parlant ! J'ai donc commenté les autres en rajoutant une # devant la ligne Server.

```
GNU nano 2.2.6      File: /etc/pacman.d/mirrorlist      Modified
## Score: 1.0, Italy
#Server = http://archlinux.openlabto.org/archlinux/$repo/os/$arch
## Score: 1.1, United States
#Server = http://archlinux.surlyjake.com/archlinux/$repo/os/$arch
## Score: 1.1, Germany
#Server = http://mirror.selfnet.de/archlinux/$repo/os/$arch
## Score: 1.1, Belarus
#Server = http://mirror.datacenter.by/pub/archlinux/$repo/os/$arch
## Score: 1.1, United Kingdom
#Server = http://archlinux.mirrors.uk2.net/$repo/os/$arch
## Score: 1.1, United States
#Server = http://mirrors.liquidweb.com/archlinux/$repo/os/$arch
## Score: 1.2, France
Server = http://mir.archlinux.fr/$repo/os/$arch
## Score: 1.2, Netherlands
#Server = http://arch.apt-get.eu/$repo/os/$arch
## Score: 1.2, United States
#Server = http://mirror.ece.vt.edu/archlinux/$repo/os/$arch
## Score: 1.2, Japan
#Server = http://ftp.jaist.ac.jp/pub/Linux/ArchLinux/$repo/os/$arch

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^U Next Page  ^U UnCut Text ^T To Spell
```

Illustration 3: Sélection du serveur

On passe à l'installation de la base. La deuxième ligne rajoute certains outils bien pratique à avoir dès le départ.

```
pacstrap /mnt base base-devel
pacstrap /mnt zip unzip p7zip sudo vim mc alsa-utils syslog-ng
```

Maintenant que les outils de base, on va passer aux choses sérieuses. Mais avant tout, il faut générer le fichier /etc/fstab.

```
genfstab -U -p /mnt >> /mnt/etc/fstab
```

On installe le chargeur de démarrage. Pour mon tutoriel, j'utilise la version bios de Grub2.

```
pacstrap /mnt grub2-bios
```

On va passer aux réglages de l'OS qu'on vient d'installer. Il faut donc y entrer.

```
arch-chroot /mnt
```

Étant donné que depuis le 13 octobre 2012, le système installé utilise systemd, on

va lui créer les fichiers nécessaires à son bon fonctionnement.

Pour avoir le bon clavier en mode texte, il faut modifier le fichier `/etc/vconsole.conf` en y entrant ceci :

```
KEYMAP=fr-latin9
FONT=lat9w-16
```

Pour avoir la localisation française, le fichier `/etc/locale.conf` doit contenir :

```
LANG=fr_FR.UTF-8
```

Sans oublier qu'il faut vérifier que la ligne « `fr_FR.UTF-8 UTF-8` » dans le fichier `/etc/locale.gen` n'a pas de `#` devant elle. Ainsi que la ligne « `en_US.UTF-8 UTF-8` »

On va maintenant générer les traductions :

```
locale-gen
```

Le nom de la machine ? Il est contenu dans le fichier `/etc/hostname`. Il suffit de taper le nom sur la première ligne.

Le fuseau horaire ? Une seule étape. Prenons le cas d'une installation avec le fuseau horaire de Paris.

On commence par créer un lien symbolique :

```
ln -sf /usr/share/zoneinfo/Europe/Paris /etc/localtime
```

Ensuite, deux cas se présente. Soit on a une machine seule, et on peut demander à ce que l'heure appliquée soit UTC, soit un double démarrage avec MS-Windows. Prenons le premier cas.

On va demander à ce que l'heure soit par défaut réglée sur UTC :

```
hwclock --systohc --utc
```

Sinon, on ne touche à rien. Car MS-Windows est un goujat dans ce domaine:(

Étape suivante ? Générer le bon fichier grub, installer grub et générer le noyau.

```
grub-mkconfig -o /boot/grub/grub.cfg
grub-install --target=i386-pc --recheck /dev/sda
cp /usr/share/locale/en\@quot/LC_MESSAGES/grub.mo /boot/grub/locale/en.mo
mkinitcpio -p linux
```

```

Installation finished. No error reported.
sh-4.2# mkinitcpio -p linux
==> Building image from preset: 'default'
    -> -k /boot/vmlinuz-linux -c /etc/mkinitcpio.conf -g /boot/initramfs-linux.img
==> Starting build: 3.7.10-1-ARCH
    -> Running build hook: [base]
    -> Running build hook: [udev]
    -> Running build hook: [autodetect]
    -> Running build hook: [modconf]
    -> Running build hook: [block]
    -> Running build hook: [filesystems]
    -> Running build hook: [keyboard]
    -> Running build hook: [fsck]
==> Generating module dependencies
==> Creating gzip initcpio image: /boot/initramfs-linux.img
==> Image generation successful
==> Building image from preset: 'fallback'
    -> -k /boot/vmlinuz-linux -c /etc/mkinitcpio.conf -g /boot/initramfs-linux-fal
    lback.img -S autodetect
==> Starting build: 3.7.10-1-ARCH
    -> Running build hook: [base]
    -> Running build hook: [udev]
    -> Running build hook: [modconf]
    -> Running build hook: [block]

```

Illustration 4: Génération du noyau d'Archlinux

Bien entendu, aucune erreur ne doit apparaître.

On donne un mot de passe au compte root :

```
passwd root
```

Pour le réseau, étant donné qu'on va utiliser Gnome par la suite, je trouve qu'installer et activer NetworkManager est une bonne idée.

```
pacman -S networkmanager
systemctl enable NetworkManager.service
```

Dernier réglage, **optionnel si on veut avoir accès à l'outil yaourt**, il faut ajouter ceci au fichier /etc/pacman.conf à la fin.

```
[archlinuxfr]
Server = http://repo.archlinux.fr/\$arch
```

Si vous voulez utiliser un outil comme Skype (qui est uniquement en 32 bits) et que vous installez un système 64 bits, il faut décommenter (enlever les #) des lignes suivantes :

```
#[multilib]
#Include = /etc/pacman.d/mirrorlist
```

On peut maintenant quitter tout, démonter proprement les partitions et redémarrer. C'est un peu plus délicat qu'auparavant. Mais au moins, on voit les étapes à suivre.

```
exit
umount /mnt/{boot,home,}
```

Voilà, on peut redémarrer. On va passer à la suite, largement moins ennuyeuse !

II. Installons maintenant Gnome !

Nous attaquons donc la partie la plus intéressante, l'installation de Gnome. C'est plutôt simple.

Une fois le système démarré, on se connecte en root. Etant donné que j'ai installé NetworkManager à l'étape précédente, le réseau fonctionne directement. On tape les lignes de commande suivante pour mettre à jour les dépôts et installer yaourt un outil qui va nous simplifier grandement la vie. Ainsi que ntp !

```
pacman -Syy
pacman -S yaourt ntp
```

Les outils en place, on lance alsamixer, pour configurer le niveau sonore :

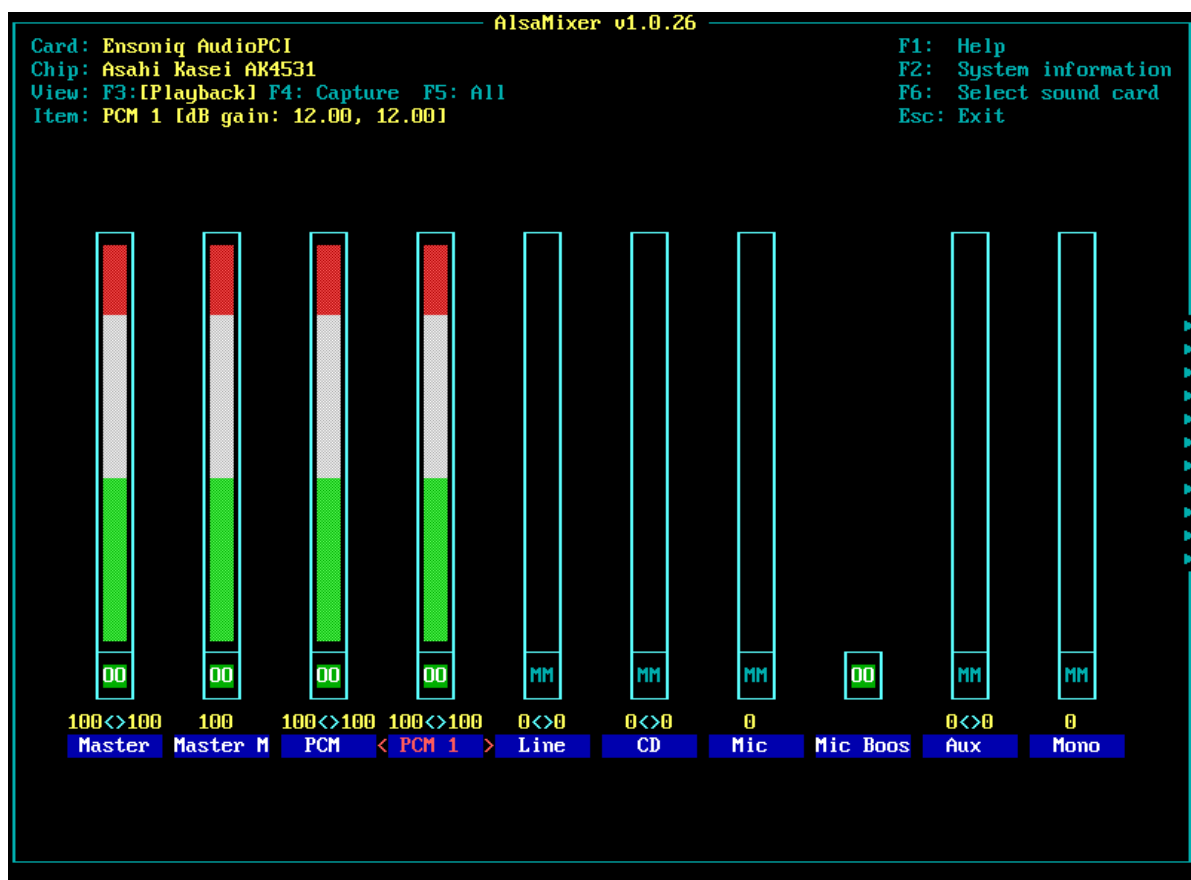


Illustration 5: Alsamixer en action

Une fois l'ensemble configuré, pour le conserver tel quel, il suffit d'entrer :

```
alsactl store
```

Nous sommes dans le multimédia ? Restons-y.

On va installer l'ensemble des greffons gstreamer qui nous donneront accès aux fichiers multimédias une fois Gnome lancé. Si vous n'avez pas installé yaourt, il

faudra le remplacer par pacman -S ou sudo pacman -S quand vous utiliserez votre compte utilisateur « normal ».

```
yaourt -S gstreamer0.10-plugins gst-plugins-good gst-plugins-bad gst-plugins-ugly
```

Passons à l'installation de Xorg. Il suffit de rentrer :

```
yaourt -S xorg-server xorg-xinit xorg-xmessage xorg-utils
```

Il faut ensuite choisir le pilote pour le circuit vidéo. Sur ce point précis, je ne peux que vous conseiller de lire le wiki anglophone qui donne la liste des principaux pilotes.

<https://wiki.archlinux.org/index.php/Xorg#Graphics>

Dans le cas d'une machine virtuelle, j'ai utilisé le paquet xf86-video-vesa. On passe ensuite à l'installation des polices. Voici la ligne de commande pour les principales, y compris les polices de Microsoft (Times, Arial, Courier). Pour les polices microsoft, elles sont sur le dépôt AUR, donc il faut utiliser yaourt pour les récupérer et les installer.

```
yaourt -S ttf-bitstream-vera ttf-liberation ttf-freefont ttf-dejavu ttf-msfonts
```

Si vous faites une installation dans VirtualBox, deux étapes supplémentaires. D'abord installer le paquet virtualbox-guests-modules.

Ensuite, il faut créer un fichier /etc/modules.load.d/vbox.conf qui contient les lignes suivantes :

```
vboxguest  
vboxvideo  
vboxsfnet
```

Ensuite, on va rajouter des outils, histoire de ne pas voir un gnome vide au premier démarrage. On commence par tout ce qui est graphique : gimp, cups et hplip (si vous avez une imprimante scanner Hewlett Packard) :

```
yaourt -S cups gimp hplip
```

On rajoute LibreOffice et sa traduction en français. LibreOffice étant « saucissonné », il faut sélectionner les parties qui nous intéressent. Il suffit d'entrer les numéros des parties qui sont désirées.

```
yaourt -S libreoffice libreoffice-fr
```

On rajoute ensuite Mozilla Firefox :

```
yaourt -S firefox firefox-i18n-fr
```

Vous préférez Chromium ?

```
yaourt -S chromium
```


On passe enfin au morceau de choix : l'installation de Gnome.

```
yaourt -S gnome gnome-extra gnome-tweak-tool system-config-printer xdg-user-dirs
```

L'installation de Gnome terminée, on crée un utilisateur avec la commande suivante :

```
useradd -m -g users -G wheel -s /bin/bash nom-de-l'utilisateur  
passwd nom-de-l'utilisateur
```

Pour définir le vrai nom complet de l'utilisateur, il faut utiliser la commande suivante :

```
chfn nom-de-l'utilisateur
```

Avant de finir, on va configurer sudo en utilisant visudo. En effet, il nous suffit de modifier une ligne pour que l'on puisse accéder en tant qu'utilisateur classique aux droits complets sur la machine de manière temporaire.

Il faut aller jusqu'à la ligne :

```
##Uncomment to allow members of group wheel to execute any command
```

Et enlever le # sur la ligne qui suit.

Étant donné que systemd est utilisé, voici la liste des services à activer :

```
systemctl enable syslog-ng.service  
systemctl enable cronie.service  
systemctl enable avahi-daemon.service  
systemctl enable avahi-dnssconfd.service  
systemctl enable cups.service  
systemctl enable bluetooth.service  
systemctl enable ntpd.service  
systemctl enable gdm.service
```

Au démarrage suivant, GDM nous accueille. Et nous pouvons nous connecter.

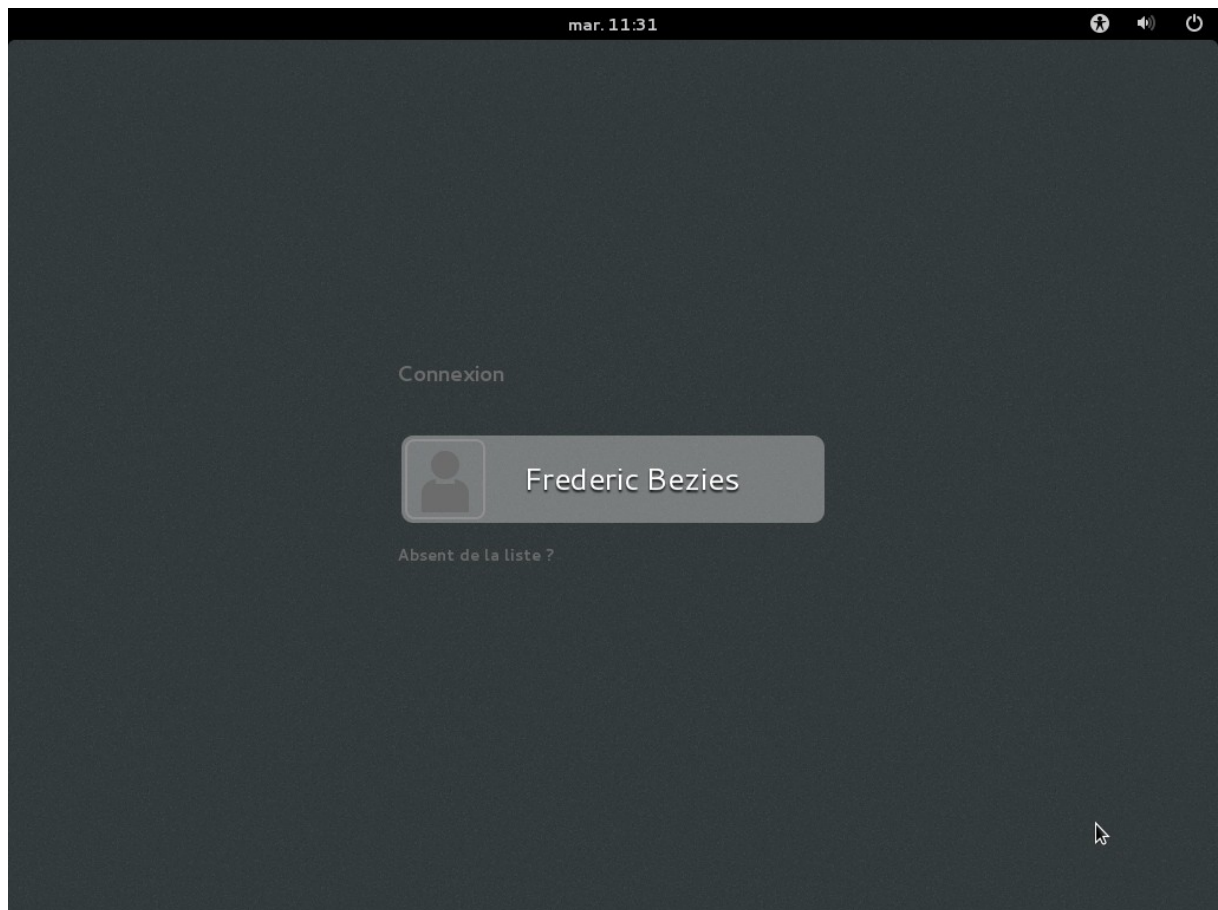


Illustration 6: GDM 3.6.x

III. Finalisons notre installation.

Quelques outils à rajouter : xsane (pour le scanner), mais aussi unoconv (pour l'aperçu des fichiers dans Gnome Documents) et les greffons grilo pour Totem... et Flash...

```
yaourt -S xsane unoconv grilo grilo-plugins flashplugin
```

Si on ouvre le gestionnaire de fichiers, on s'aperçoit que les répertoires classiques « Documents », « Téléchargements » manquent à l'appel.

Il suffit d'entrer la commande suivante dans un terminal et ils arrivent :

```
xdg-user-dirs-update
```

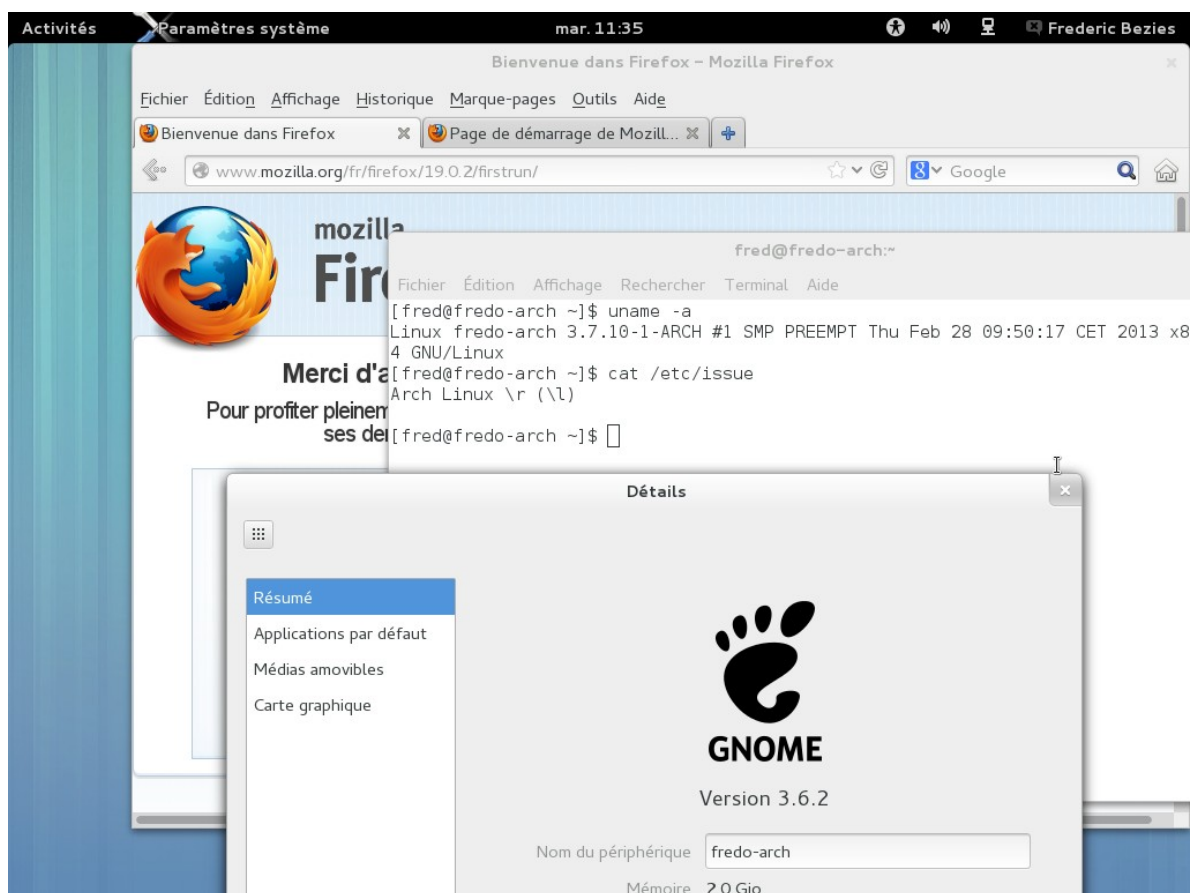


Illustration 7: Mozilla Firefox 19.0.2 et Gnome 3.6.2

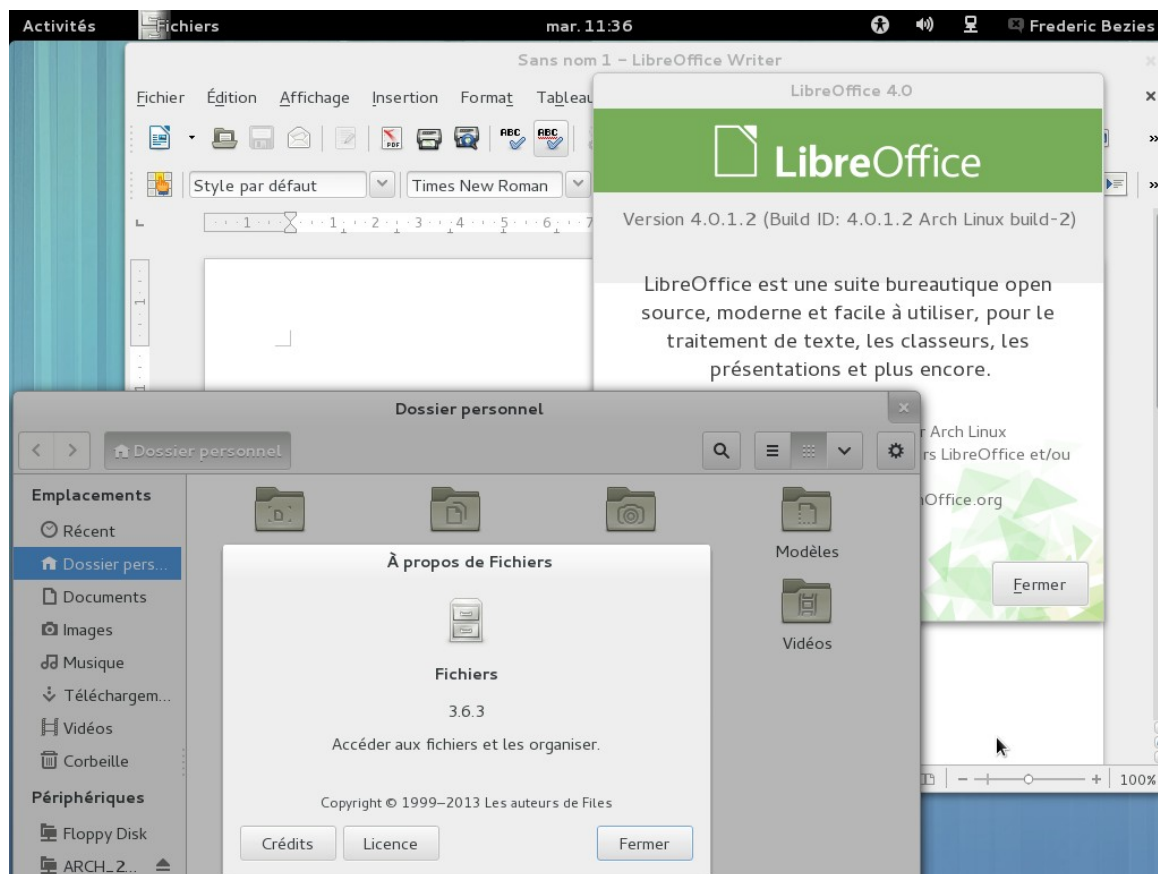


Illustration 8: LibreOffice 4.0.1 et Nautilus en action.

Voilà, le guide est maintenant fini. Cependant, je n'ai pas abordé l'installation d'un parefeu. C'est quelque chose de plus technique.

J'ai surtout voulu faire un **guide rapide**, histoire de montrer qu'en une petite heure on pouvait avoir un environnement installé et assez complet pour le finir par la suite.

Bonne découverte !