

Documentation ArchLinux

Prérequis

- un iso **Archlinux** (dans cette documentation 6.13.5)
- un PC ou une VM (On va utilisé une VM)

Debut

On devrait être sur cette page :

```
Arch Linux 6.13.5-arch1-1 (tty1)
archiso login: root (automatic login)

To install Arch Linux follow the installation guide:
https://wiki.archlinux.org/title/Installation_guide

For Wi-Fi, authenticate to the wireless network using the iwctl utility.
For mobile broadband (WWAN) modems, connect with the mmcli utility.
Ethernet, WLAN and WWAN interfaces using DHCP should work automatically.

After connecting to the internet, the installation guide can be accessed
via the convenience script Installation_guide.

root@archiso ~ #
```

Changement clavier fr

On utilise **loadkeys** pour passer le clavier avec le standard fr :

```
loadkeys fr
```

Changement de la police

On utilise **setfont** pour passer la police en ter-132b :

```
setfont ter-132b
```

Partitionnement

Recommandation

Recommandation de l'ANSI :

Id	MountPoint	Size	Options	Type
sda1	/	1Go	None	ext4
sda2	/boot	512Mo	nosuid, nodev, noexec	fat32
sda3	None	2Go	None	swap
sda5	/var	4Go	nosuid, nodev, noexec	ext4
sda6	/usr	4Go	nodev	ext4
sda7	/home	Reste	nosuid, nodev, noexec	ext4

Liste des volumes

On affiche les volumes de stockage disponible avec **fdisk**:

```
fdisk -l
```

`-l` : liste les volume sde stockage

Résultat:

```
root@archiso ~ # fdisk -l
Disk /dev/sda: 16 GiB, 17179869184 bytes, 33554432 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop0: 836.7 MiB, 877342720 bytes, 1713560 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

On a deux volumes de stockage disponible:

- `/dev/sda` : le disque principal
- `/dev/loop0` : le disque du live boot

 **Warning**

Il ne faut pas modifier le disque du live boot

Création des partitions

On utilise fdisk pour créer les nouvelles partitions :

```
fdisk /dev/sda
```

```
Welcome to fdisk (util-linux 2.40.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS (MBR) disklabel with disk identifier 0x4d049784.

Command (m for help):
```

Volume '/'

```
Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-33554431, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-33554431, default 33554431):
+1G

Created a new partition 1 of type 'Linux' and of size 1 GiB.
```

Volume 'boot'

```
Command (m for help): n
Partition type
  p   primary (1 primary, 0 extended, 3 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2):
First sector (2099200-33554431, default 2099200):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2099200-33554431, default 33554431):
+512M

Created a new partition 2 of type 'Linux' and of size 512 MiB.
```

Volume 'swap'

```

Command (m for help): n
Partition type
  p   primary (2 primary, 0 extended, 2 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (3,4, default 3):
First sector (3147776-33554431, default 3147776):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (3147776-33554431, default 33554431): +2G

```

Volume étendue ''

```

Command (m for help): n
Partition type
  p   primary (3 primary, 0 extended, 1 free)
  e   extended (container for logical partitions)
Select (default e): e

Selected partition 4
First sector (7342080-33554431, default 7342080):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (7342080-33554431, default 33554431):

Created a new partition 4 of type 'Extended' and of size 12.5 GiB.

```

Volume 'var'

```

Command (m for help): n
All primary partitions are in use.
Adding logical partition 5
First sector (7344128-33554431, default 7344128):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (7344128-33554431, default 33554431): +4G

Created a new partition 5 of type 'Linux' and of size 4 GiB.

```

Volume 'usr'

```

Command (m for help): n
All primary partitions are in use.
Adding logical partition 6
First sector (15734784-33554431, default 15734784):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (15734784-33554431, default 33554431): +4G

Created a new partition 6 of type 'Linux' and of size 4 GiB.

```

Volume 'home'

```
Command (m for help): n
All primary partitions are in use.
Adding logical partition 7
First sector (24125440-33554431, default 24125440):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (24125440-33554431, default 33554431):

Created a new partition 7 of type 'Linux' and of size 4.5 GiB.
```

Écriture des partitions

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Formatage des partitions

On utilise **mkfs** pour formater chaque partitions:

/dev/sda1 `/`

```
root@archiso ~ # mkfs.ext4 /dev/sda1
mke2fs 1.47.2 (1-Jan-2025)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 0bcf9a00-55a5-481e-8aa0-7e35a502f656
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

/dev/sda2 `boot`

```
root@archiso ~ # mkfs.fat -F 32 /dev/sda2
mkfs.fat 4.2 (2021-01-31)
```

/dev/sda5 `var`

```
1 root@archiso ~ # mkfs.ext4 /dev/sda5
mke2fs 1.47.2 (1-Jan-2025)
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: 3fdb175e-f002-4628-a3c9-709ec99d846f
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

`/dev/sda6` `usr`

```
root@archiso ~ # mkfs.ext4 /dev/sda6
mke2fs 1.47.2 (1-Jan-2025)
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: 1618e205-90bb-46be-ad0e-b1e7f98bb038
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

`/dev/sda7` `home`

```
root@archiso ~ # mkfs.ext4 /dev/sda7
mke2fs 1.47.2 (1-Jan-2025)
Creating filesystem with 1178624 4k blocks and 294912 inodes
Filesystem UUID: 860e93aa-fae8-4218-964a-e67bec79020d
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

`/dev/sda3` `swap`

Pour le swap (`/dev/sda3`) on utilise `mkswap` :

```
root@archiso ~ # mkswap /dev/sda3
Setting up swapspace version 1, size = 2 GiB (2147479552 bytes)
no label, UUID=218aa667-bdae-4a48-ac66-ed9c3ca44885
```

Montage des systèmes de fichier

On utilise **swapon**:

```
swapon /dev/sda3
```

On utilise **mount** pour monter les systèmes de fichier:

```
mount /dev/sda1 /mnt
mount --mkdir /dev/sda2 /mnt/boot -o noexec,nodev,nosuid
mount --mkdir /dev/sda5 /mnt/var -o noexec,nodev,nosuid
mount --mkdir /dev/sda6 /mnt/usr -o nodev
mount --mkdir /dev/sda7 /mnt/home -o noexec,nodev,nosuid
```

`--mkdir` : crée le dossier si il n'est pas existant

`-o` : permet l'utilisation d'options

`nodev` : Empêche l'interprétation des fichiers spéciaux de caractères ou de blocs sur le système de fichiers.

`noexec` : Empêche l'exécution de binaires sur le système de fichiers.

`nosuid` : Empêche les bits set-user-identifier (SUID) et set-group-identifier (SGID) de prendre effet sur le système de fichiers.

Installation des paquets

On installe les paquets de base pour linux avec **pacstrap** :

```
pacstrap -K /mnt base linux linux-firmware nano
```

`-K` : permet de ne pas demander de confirmation avant d'installer les paquets.

`/mnt` : le système monté dans le quel installer les paquets.

`base` : contient les paquets essentiels nécessaires pour démarrer un système Arch Linux de base.

`linux` : contient le noyau Linux.

`linux-firmware` : contient les microprogrammes (firmware) nécessaires pour divers périphériques matériels.

`nano` : éditeur de texte (terminal).

Generation des information des systèmes de fichiers montés

On les génère grace a **genfstab** :

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

`-U` : Permet d'utiliser les UUID (Universally Unique Identifiers) des partitions pour les identifier dans le fichier.

`/mnt/etc/fstab` : emplacement du fichier `fstab` du nouveau système de fichiers monté.

Dans l'environnement monté

Changement d'environnement

On utilise `arch-chroot` pour changer l'environnement racine vers le nouveau système de fichiers monté :

```
arch-chroot /mnt
```

Configurations

Fuseau horaire

On utilise `ln` pour changer de fuseau horaire :

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

`ln` : permet de créer des liens entre des fichiers ou des répertoires.

`-s` : permet de créer un lien symbolique.

`-f` : force la création du lien symbolique, en remplaçant tout fichier existant au même emplacement.

`/usr/share/zoneinfo/Europe/Paris` : Zone horaire

`/etc/localtime` : fichier utilisé par le système pour déterminer le fuseau horaire local.

Synchronisation de l'horloge système

On utilise `hwclock` :

```
hwclock --systohc
```

`--systohc` : permet de synchroniser l'horloge matérielle avec l'horloge système.

Localisation

On utilise `nano` pour éditer `locale.gen` :

```
nano /etc/locale.gen
```

On de-commente la localisation voulu (`fr_FR.UTF-8 UTF-8`) et sauvegarde.

Ensuite on applique les changements avec **locale-gen**:

```
locale-gen
```

On crée le **locale.conf** avec :

```
echo "LANG=fr_FR.UTF-8" > /etc/locale.conf
```

Pareil pour **vconsole.conf** :

```
echo "KEYMAP=fr-latin1" > /etc/vconsole.conf
```

Change le mot de pass root

On utilise **passwd** :

```
passwd
```

```
[root@archiso /]# passwd
New password:
Retype new password:
passwd: password updated successfully
```

Initramfs

On crée des images initramfs avec **mkinitcpio**.

```
mkinitcpio -P
```

-P : permet de construire toutes les images initramfs définies dans les presets.

Warning

Si il ne trouve pas le fichier il faut ajouter dans le fichier `/etc/mkinitcpio.conf` :
usr dans la ligne HOOKS

Boot loader

On utilise **pacman** pour installer **grub** :

```
pacman -S grub
```

`grub` : pour le bootloader GRUB.

Installation

```
grub-install --target=i386-pc /dev/sda
```

`--target` : spécifie la plateforme cible pour l'installation de GRUB.

`i386-pc` : spécifie que GRUB doit être installé pour une architecture BIOS traditionnelle.

`/dev/sda` : chemin du disque sur lequel GRUB sera installé.

Configuration

On génère un fichier de configuration GRUB avec `grub-mkconfig` :

```
grub-mkconfig -o /boot/grub/grub.cfg
```

`-o` : spécifie le fichier de sortie.

Fin

On quitte le chroot avec CTRL+D, et on de-monte chaque partition avec `umount` :

```
umount -a
```

`-a` : précise toutes partitions

On peut ensuite éteindre la machine avec:

```
shutdown now
```

et on retire le périphérique de l'iso, et on relance la machine.

On fait une vérification avec **lsblk**:

```
Arch Linux 6.13.7-arch1-1 (tty1)

archlinux login: root
Password:
[root@archlinux ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO  TYPE MOUNTPOINTS
sda          8:0    0   16G  0  disk
├─sda1       8:1    0    1G  0  part /
├─sda2       8:2    0   512M 0  part /boot
├─sda3       8:3    0    2G  0  part
├─sda4       8:4    0    1K  0  part
├─sda5       8:5    0    4G  0  part /var
├─sda6       8:6    0    4G  0  part /usr
└─sda7       8:7    0   4,5G 0  part /home
sr0         11:0    1 1024M  0  rom
[root@archlinux ~]# _
```