

Guide d'installation Archlinux

Environnement : GNOME
Systèmes : BIOS-MBR / UEFI-GPT
Image ISO : <http://mir.archlinux.fr/iso/>

Crée le : 04-07-2020
Révisé le : 19-12-2023

Sources : [https://wiki.archlinux.org/title/Guide_d%27installation](https://wiki.archlinux.org/title/Guide_d%27installation%20%5Bobsolète%5D)
[\[Obsolète\] Installons Archlinux à la mimine](#)

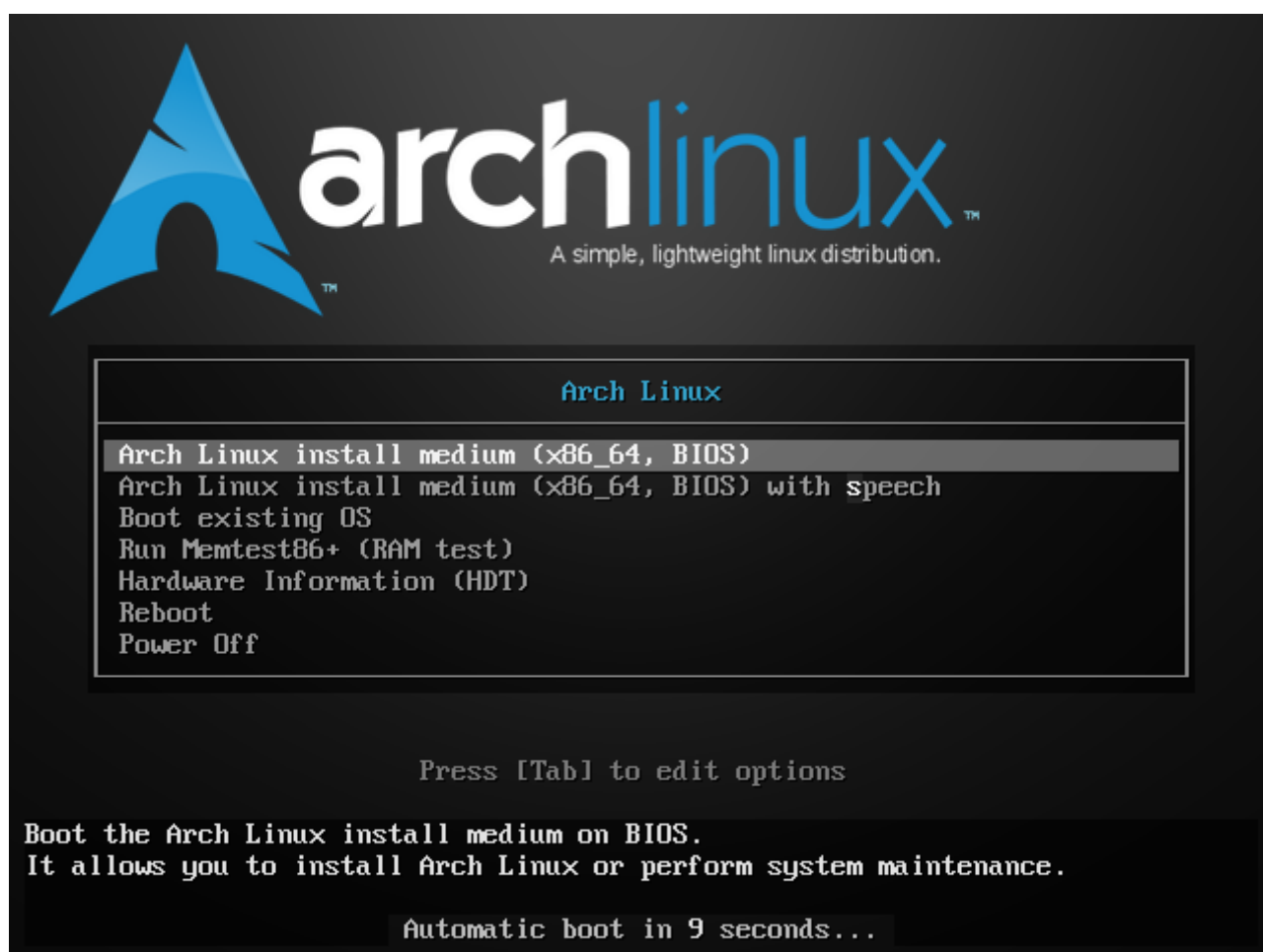


Table des matières

1	Préparation des disques.....	3
1.1	Prérequis.....	3
1.2	Schémas de partitionnement.....	3
1.3	Dimensionnement des partitions.....	3
1.4	Formatage des partitions.....	4
1.5	Attribution et montage des partitions.....	4
2	Installation de Archlinux.....	5
2.1	Installation de la base.....	5
2.2	Paramètres régionaux.....	6
2.3	Création du Kernel.....	6
2.4	Création du menu GRUB.....	7
2.5	Mot de Passe du 'Super-Utilisateur'.....	7
2.6	Gestionnaire de réseau.....	7
2.7	Gestionnaire de Miroirs.....	7
2.8	Gestionnaire de paquets PACMAN.....	8
2.9	Finalisation.....	8
3	Configuration de l'environnement.....	9
3.1	Outils de base.....	9
3.2	Configuration du niveau sonore.....	9
3.3	Fichiers Multimédias.....	9
3.4	Système de fenêtrage 'Xorg'.....	9
3.5	Installation des Polices (Fonts).....	9
3.6	Pilotes vidéo.....	10
3.7	Utilitaires d'impression.....	10
3.8	Compte utilisateur.....	11
3.9	Activation des services.....	11
4	Interface de bureau.....	12
4.1	Installation de l'environnement GNOME.....	12
4.2	Gestionnaire de démarrage.....	12
4.3	Navigateur Web.....	12
4.4	Gestionnaire de messagerie.....	12
4.5	Outils bureautique.....	13
4.6	Outils multimédia.....	13
4.7	Outils graphique.....	13
5	Dépôts 'Arch User Repository' (AUR).....	13
6	Extensions GNOME.....	13
6.1	Arch Linux Updates Indicator.....	14
7	Trucs & Astuces.....	14
7.1	Supprimer les erreurs 'mkinitcpio'.....	14
7.2	Lien Symbolique.....	14
7.3	Gestion des paquets.....	15
8	Droits & Permissions.....	16
8.1	Change Mode 'chmod'.....	16
8.2	Change Owner 'chown'.....	16
9	Création d'une clé USB Bootable.....	16
10	Dépannage 'chroot'.....	16
11	Wine Is Not an Emulator.....	17
12	Captvty.....	17
12.1	Captvty 2 (WINEARCH=win32).....	18
12.2	Captvty 2 & 3 Alpha (WINEARCH=win32) :.....	18
12.3	Captvty 3 Alpha (WINEARCH=win64).....	19
12.4	Captvty 3 Bêta.....	20
13	DCMOTO.....	21
14	Sketchup.....	22
15	GNOME Mastermind™ (PKGBUILD).....	23

1 Préparation des disques

1.1 Prérequis

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

```
# loadkeys fr
```

1.2 Schémas de partitionnement

Création des partitions en mode graphique :

Pour le formatage des partitions, si vous avez peur de faire des bêtises, il est plus prudent de passer par un LiveCD/USB tel que [GNOME Partition Editor](#).

BIOS-MBR :



Partition	Système de fichiers	Point de montage	Étiquette	Taille	Utilisé	Inutilisé	Drapeaux
/dev/sda1	ext2	/boot	BOOT	488.00 Mio	130.86 Mio	357.14 Mio	boot
/dev/sda2	linux-swap		SWAP	3.72 Gio	0.00 o	3.72 Gio	swap
/dev/sda3	ext4	/	LINUX	14.90 Gio	7.97 Gio	6.93 Gio	
/dev/sda4	ext4	/home	DATA	10.70 Gio	399.78 Mio	10.31 Gio	

UEFI-GPT :



Partition	Nom	Système de fichiers	Point de montage	Étiquette	Taille	Utilisé	Inutilisé	Drapeaux
/dev/sda1	EFI System partition	fat32	/boot/efi	ESP	488.00 Mio	1.14 Mio	486.86 Mio	boot, esp
/dev/sda2	Exchange partition	linux-swap		SWAP	3.72 Gio	0.00 o	3.72 Gio	swap
/dev/sda3	Archlinux	ext4	/	LINUX	14.90 Gio	8.04 Gio	6.86 Gio	
/dev/sda4	Home	ext4	/home	DATA	10.70 Gio	400.61 Mio	10.31 Gio	

Ne SURTOUT pas oublier de créer les '**Table de partition**' et de **définir les drapeaux** (*flags*) sur la partition 'sda1' dédiée au démarrage du système :

Système	Table de partition	Drapeaux
BIOS-MBR	msdos	boot
UEFI-GPT	gpt	boot, esp

1.3 Dimensionnement des partitions

Disque	Partition	Taille	Système de fichiers
/dev/sda1	/boot	512Mo (488Mio)	ext2 (BIOS) fat32 (UEFI)
/dev/sda2	/swap	≥ RAM (x2 max.)	swap
/dev/sda3	/	≥ 32Go	ext4
/dev/sda4	/home	le reste du disque	ext4

Convertisseur d'unités informatique :

<https://lehollandaisvolant.net/tout/tools/mo-mio/>

1.4 Formatage des partitions

Formatage de la partition de démarrage '/boot'.

BIOS-MBR :

```
# mkfs.ext2 /dev/sda1
```

UEFI-GPT :

```
# mkfs.vfat -F32 /dev/sda1
```

Pour la suite, les instructions sont valables pour les systèmes BIOS & UEFI.

Formatage des partitions racine '/' et utilisateur '/home' :

```
# mkfs.ext4 /dev/sda3
# mkfs.ext4 /dev/sda4(*)
```

^(*) Si /home est placé sur un disque séparé remplacer 'sda4' par 'sdxx'

Sans oublier le formatage de la partition d'échange '/swap' :

```
# mkswap /dev/sda2
```

1.5 Attribution et montage des partitions

Commençons par activer la partition '/swap' :

```
# swapon /dev/sda2
```

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

BIOS-MBR :

```
# mount /dev/sda3 /mnt
# mkdir /mnt/{boot,home}
# mount /dev/sda1 /mnt/boot
# mount /dev/sda4(*) /mnt/home
```

UEFI-GPT :

```
# mount /dev/sda3 /mnt
# mkdir /mnt/home && mount /dev/sda4(*) /mnt/home
# mkdir -p /mnt/boot/efi && mount -t vfat /dev/sda1 /mnt/boot/efi
```

^(*) Si '/home' est placé sur un disque séparé remplacer 'sda4' par 'sdxx'

2 Installation de Archlinux

2.1 Installation de la base

Outils de base :

```
# pacstrap /mnt base base-devel pacman-contrib  
# pacstrap /mnt linux(*) linux-firmware
```

(*) Si on veut utiliser le noyau LTS remplacer '**linux**' par '**linux-lts**'

Outils bien pratiques à avoir dès le départ, à taper sur une seule ligne !

```
# pacstrap /mnt zip unzip p7zip vim mc alsa-utils syslog-ng mtools  
dosfstools lsb-release ntfs-3g exfat-utils bash-completion htop nano vi neofetch  
tree wget ufw man-db man-pages man-pages-fr dmidecode git
```

Selon la configuration la machine :

amd-ucode	<i>microcodes AMD</i>
intel-ucode	<i>microcodes INTEL</i>
tlp	<i>utile pour la batterie des portables</i>
hplip	<i>drivers pour les imprimantes HP</i>
efibootmgr	<i>application pour modifier le 'EFI Boot Manager'</i>
bluez	<i>pour le Bluetooth</i>

Générer le fichier 'fstab' qui liste les partitions présentes :

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

Au tour du chargeur de démarrage GRUB :

```
# pacstrap /mnt grub os-prober
```

Notes : le paquet 'os-prober' est indispensable pour les Dual-Boot.

Passons aux réglages de l'OS :

```
# arch-chroot /mnt
```

2.2 Paramètres régionaux

Configuration du clavier, création du fichier 'vconsole.conf' :

```
# nano /etc/vconsole.conf  
  
KEYMAP=fr-latin9  
FONT=eurlatgr
```

Configuration de la localisation, création du fichier 'locale.conf' :

```
# nano /etc/locale.conf  
  
LANG=fr_FR.UTF-8  
LC_COLLATE=C
```

Configuration du fichier 'locale.gen' décommenter les lignes :

```
# nano /etc/locale.gen  
  
en_US.UTF-8 UTF-8  
fr_FR.UTF-8 UTF-8
```

Astuce :

Raccourci clavier pour effectuer une recherche dans les langages:

CTRL+W ou **F6**

On va maintenant générer les traductions :

```
# locale-gen
```

Spécifier la locale pour la session courante :

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

Configuration du nom de la machine, création du fichier 'hostname' :

```
# nano /etc/hostname
```

Création d'un lien symbolique pour le fuseau horaire :

```
# ln -sf /usr/share/zoneinfo/Europe/Paris /etc/localtime  
# hwclock --systohc --utc
```

2.3 Création du Kernel

Générer l'image du noyau :

```
# mkinitcpio -p linux(*)
```

^(*) Si on veut utiliser le noyau LTS remplacer '**linux**' par '**linux-lts**'

Bien entendu, aucune erreur ne doit apparaître.

2.4 Création du menu GRUB

BIOS-MBR :

```
# grub-install --no-floppy --recheck /dev/sda
```

UEFI-GPT :

```
# grub-install --target=x86_64-efi --efi-directory=/boot/efi --bootloader-id=Grub --recheck
```

Astuce :

Pour ne pas afficher GRUB au démarrage :

Note : <https://forum.ubuntu-fr.org/viewtopic.php?id=2027095>

```
# nano /etc/default/grub

GRUB_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
```

A chaque modification de GRUB, il faut **TOUJOURS** effectuer une mise à jour de la configuration :

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

2.5 Mot de Passe du 'Super-Utilisateur'

On donne un mot de passe au compte 'root' :

```
# passwd root
```

2.6 Gestionnaire de réseau

Activer le réseau avec 'NetworkManager' :

```
# pacman -S networkmanager
# systemctl enable NetworkManager
```

2.7 Gestionnaire de Miroirs

Note : [https://wiki.archlinux.org/title/Reflector_\(Fran%C3%A7ais\)](https://wiki.archlinux.org/title/Reflector_(Fran%C3%A7ais))

```
# pacman -S reflector
```

Édition du fichier 'reflector.conf' et activation du service:

```
# nano /etc/xdg/reflector/reflector.conf

--protocol http,https
--latest 10
--sort rate

# systemctl enable reflector.service
```

Aussi, on peut aussi directement gérer le fichier 'mirrorlist' :

```
# nano /etc/pacman.d/mirrorlist
```

Astuces :

Pour lancer « Recherche (pour remplacer) » :
ALT+R

Dans le champ « Search (to replace) » taper :
#Server

Pour décommenter tous les serveurs saisir :
Server

Raccourci clavier pour effectuer une recherche dans les serveurs :
CTRL+W ou **F6**

2.8 Gestionnaire de paquets PACMAN

Fichier de configuration :

```
# nano /etc/pacman.conf
```

Pour obtenir les dépôts nécessaires pour les outils 32 bits, décommenter les lignes :

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

Astuces :

Pour mettre de la couleur dans le terminal, décommenter la ligne :
color

Totalement inutile et (peut-être) indispensable...
A la fin de la rubrique # Misc options ajoutez la ligne :
ILoveCandy

```
:: Synchronisation des bases de données de paquets...  
core           133,5 KiB   340 KiB/s 00:00 [#####] 100%  
extra          1707,6 KiB  317 KiB/s 00:05 [#####] 100%  
community      3,7 MiB   509 KiB/s 00:02 [#####-----] 76%
```

```
:: Synchronisation des bases de données de paquets...  
core           133,5 KiB   3,26 MiB/s 00:00 [-----] 100%  
extra          1707,6 KiB   562 KiB/s 00:03 [-----] 100%  
community      2,6 MiB   273 KiB/s 00:08 [-----C o o o ] 53%
```

2.9 Finalisation

Démonter proprement les partitions :

```
# exit  
# umount -R /mnt  
# reboot
```

Voilà, on peut redémarrer.
Il faut éjecter le support d'installation pour éviter des problèmes au démarrage suivant.

3 Configuration de l'environnement

Une fois le système démarré, on se connecte en 'root'.

3.1 Outils de base

On ajoute 'ntp' (synchronisation de l'heure en ligne) et 'cronie' (pour les tâches d'administration à automatiser) :

```
# pacman -Syyu
# pacman -S ntp cronie
```

Pour avoir les logs en clair en cas de problème, il faut modifier le fichier 'journal.conf' :

```
# nano /etc/systemd/journal.conf
ForwardToSyslog=yes
```

3.2 Configuration du niveau sonore

```
# alsamixer
```

Pour conserver l'ensemble configuré tel quel :

```
# alsactl store
```

3.3 Fichiers Multimédias

Installation l'ensemble des greffons 'gstreamer' qui nous donneront accès aux fichiers multimédias une fois GNOME lancé :

```
# pacman -S gst-plugins-{base,good,bad,ugly} gst-libav
```

Note : 'gst-libav' prend en charge tout ce qui est x264 et apparenté.

3.4 Système de fenêtrage 'Xorg'

```
# pacman -S xorg-{server,xinit,apps} xf86-input-libinput xdg-user-dirs
```

3.5 Installation des Polices (Fonts)

Note : <https://fr.wikipedia.org/wiki/Aide:Unicode>

```
# pacman -S ttf-{bitstream-vera,liberation,freefont,dejavu} freetype2

# pacman -S noto-fonts noto-fonts-extra noto-fonts-emoji
# pacman -S ttf-droid ttf-hack ttf-inconsolata
```

Note : pour les polices Microsoft, le paquet '**ttf-ms-fonts**^(AUR)', elles sont sur les dépôts AUR, donc il faut utiliser un empaqueteur comme TRIZEN ou YAY.

3.6 Pilotes vidéo

Note : [https://wiki.archlinux.org/title/Xorg_\(Fran%C3%A7ais\)](https://wiki.archlinux.org/title/Xorg_(Fran%C3%A7ais))

Tout d'abord, identifiez la carte graphique :

```
$ lspci -v | grep -A1 -e VGA -e 3D
```

AMD / ATI :

Type	Driver	OpenGL	
Libre	xf86-video-amdgpu	mesa	lib32-mesa
	xf86-video-ati		
Propriétaire	xf86-video-amdgpu	amdgpu-pro-oglp ^(AUR)	lib32-amdgpu-pro-oglp ^(AUR)

INTEL :

Type	Driver	OpenGL	
Libre	xf86-video-intel	mesa	lib32-mesa

Pour les cartes graphiques Intel de 4ème génération et plus, consultez [Intel graphics \(Français\)#Installation](#) pour connaître les pilotes disponibles.

NVIDIA :

Type	Driver	OpenGL	
Libre	xf86-video-nouveau	mesa	lib32-mesa
Propriétaire	nvidia	nvidia-utils	lib32-nvidia-utils
	nvidia-470xx-dkms ^(AUR)	nvidia-470xx-utils ^(AUR)	lib32-nvidia-470xx-utils ^(AUR)
	nvidia-390xx-dkms ^(AUR)	nvidia-390xx-utils ^(AUR)	lib32-nvidia-390xx-utils ^(AUR)

Pour les ordinateurs portables équipés de NVIDIA Optimus qui utilisent une carte vidéo intégrée combinée à un GPU dédié, consultez [NVIDIA Optimus](#).

D'autres pilotes vidéo peuvent être trouvés dans le groupe [xorg-drivers](#).

3.7 Utilitaires d'impression

Installation de 'cups' (gestionnaire d'imprimantes) :

```
# pacman -S cups
```

Note : <http://localhost:631/>

La série des paquets foomatic permet d'avoir le maximum de pilotes pour imprimantes :

```
# pacman -S foomatic-{db,db-ppds,db-gutenprint-ppds,db-nonfree,db-nonfree-ppds} gutenprint
```


3.8 Compte utilisateur

On crée un utilisateur avec la commande suivante sur une seule ligne !

```
# useradd -m -g wheel -c 'username' -s /bin/bash username
# passwd username
```

Pour autoriser l'utilisateur classique à accéder aux droits complets sur la machine, éditer le fichier 'sudoers' :

```
# visudo
ou
# nano /etc/sudoers
```

Trouver la ligne :

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

Et décommenter l'instruction associée :

```
%wheel ALL=(ALL) ALL
```

Pour sortir de l'éditeur vi et converser la modification, faire ':' puis taper les touches **W** et **Q** (Write & Quit) et valider. Taper donc :

```
:wq
```

Astuce :

Un autre truc totalement inutile et (peut-être) indispensable...

Trouver la rubrique :

```
'### Defaults specification' :
```

A la fin de la rubrique, ajouter :

```
Defaults insults
```

3.9 Activation des services

Liste des services systemd à activer :

# systemctl enable syslog-ng@default	(gestion des journaux)
# systemctl enable cronic	(pour les tâches récurrentes)
# systemctl enable avahi-daemon	(dépendance de Cups)
# systemctl enable avahi-dnsconfd	(autre dépendance de Cups)
# systemctl enable cups	(cups pour les imprimantes)
# systemctl enable ntpd	(pour synchroniser l'heure)
# systemctl enable ufw	(activation du Pare-Feu)
# systemctl enable sshd	(activation de openSSH)
# systemctl enable bluetooth	(si équipé de Bluetooth)

4 Interface de bureau

4.1 Installation de l'environnement GNOME

Logiciels de base :

```
# pacman -S gnome gnome-terminal gnome-tweaks eog gnome-icon-theme-extras  
gparted gufw system-config-printer unoconv
```

Quelques Jeux :

```
# pacman -S gnome-chess gnome-mahjongg gnome-sudoku gnome-taquin aisleriot  
leocad
```

Option (facultative) :

```
# pacman -S gnome-extra
```

L'installation de GNOME est maintenant terminée.

4.2 Gestionnaire de démarrage

Pour être certain d'avoir le bon clavier au démarrage de GDM :

```
# sudo localectl set-x11-keymap fr
```

Dans un premier temps, il ne faut pas activer le gestionnaire de connexion de l'environnement. On fait uniquement un 'systemctl start' suivi du nom du gestionnaire en question :

```
# sudo systemctl start gdm
```

Au démarrage suivant, GDM nous accueille, nous pouvons nous connecter. Une fois connecté ouvrir un Terminal et taper :

```
$ sudo systemctl enable gdm
```

4.3 Navigateur Web

On rajoute ensuite Mozilla Firefox en français avec les indispensables bloqueur de scripts et de publicités :

```
$ sudo pacman -S firefox firefox-i18n-fr firefox-noscript firefox-ublock-  
origin
```

4.4 Gestionnaire de messagerie

Après Firefox, on peut utiliser son ami 'Oiseau Tonnerre' :

```
$ sudo pacman -S thunderbird thunderbird-i18n-fr
```

4.5 Outils bureautique

La version de développement (*unstable*) :

```
$ sudo pacman -S libreoffice-fresh-fr hunspell-fr
```

La version officiel (*stable*) :

```
$ sudo pacman -S libreoffice-still-fr hunspell-fr
```

4.6 Outils multimédia

Installation de lecteurs de son et vidéo :

```
$ sudo pacman -S rhythmbox vlc
```

Pour les machines équipées d'un lecteur CD/DVD :

```
$ sudo pacman -S brasero
```

4.7 Outils graphique

Installation de GIMP (retouche d'images) :

```
$ sudo pacman -S gimp gimp-help-fr
```

5 Dépôts 'Arch User Repository' (AUR)

Installation de l'empaqueteur 'TRIZEN'.

Actions à effectuer en tant qu'utilisateur classique :

```
$ git clone https://aur.archlinux.org/trizen
$ cd trizen/
$ makepkg -sri
$ cd ..
$ sudo mv trizen /opt
```

Rechercher un paquet dans les dépôts (AUR) :

```
$ trizen -a pkgname
```

6 Extensions GNOME

Pour gérer les extensions GNOME via Firefox:

```
$ sudo pacman -S gnome-browser-connector
```

Dans le Navigateur Web Firefox installer l'extension :

Intégration à GNOME Shell

6.1 Arch Linux Updates Indicator

Aller dans 'Paramètres' et dans l'onglet 'Réglages avancés'.
Remplacer le champs 'Commande pour mettre à jour les paquets' par :

```
gnome-terminal -e 'sh -c "sudo pacman -Syu && trizen -Syua && flatpak update ;  
echo Terminé - Appuyez sur Entrée pour quitter ; read"'
```

7 Trucs & Astuces

7.1 Supprimer les erreurs 'mkinitcpio'

Note :

https://wiki.archlinux.org/title/Mkinitcpio#Possibly_missing_firmware_for_module_XXXX

==> WARNING: Possibly missing firmware for module: '*module_name*'

À adapter selon les erreurs affichées.

Module	Package
aic94xx	aic94xx-firmware ^(AUR)
ast	ast-firmware ^(AUR)
bfa	linux-firmware-qlogic
bnx2x	linux-firmware-bnx2x
liquidio	linux-firmware-liquidio
mlxsw_spectrum	linux-firmware-mellanox
nfp	linux-firmware-nfp
qat_4xxx	<i>firmware non disponible</i>
qed	linux-firmware-qlogic
qla1280	linux-firmware-qlogic
qla2xxx	linux-firmware-qlogic
wd719x	wd719x-firmware ^(AUR)
xhci_pci	upd72020x-fw ^(AUR)

7.2 Lien Symbolique

Note : <https://www.malekal.com/lien-symbolique-linux/>

```
$ ln -s [fichier/dossier existant] [chemin du lien symbolique]
```

7.3 Gestion des paquets

Mises à jour des PKG :

```
$ sudo pacman -Syyu && trizen -Syua && flatpak update
```

Suppression du cache :

```
$ paccache -rvk1
```

Lister et effacer les PKG inutiles et/ou Orphelins :

```
$ pacman -Qtd  
$ sudo pacman -Rns $(pacman -Qtdq)
```

Suppression du verrou des Mises à Jour :

```
$ sudo rm /var/lib/pacman/db.lck
```

Mises à jour des Sources & des Clés :

```
$ sudo pacman -Sy archlinux-keyring && pacman -Su
```

Supprimer un PKG seul :

```
$ sudo pacman -Rc
```

Supprimer un PKG & ses Dépendances :

```
$ sudo pacman -Rs
```

Changer la version d'un PKG via le cache PACMAN :

```
$ sudo pacman -U /var/cache/pacman/pkg/pkgname.pkg.tar.zst
```

Changer la version d'un PKG via <https://archive.archlinux.org/packages/> :

```
$ sudo pacman -U https://archive.archlinux.org/packages/v/vlc/vlc-3.0.11.1-4-x86_64.pkg.tar.zst
```

Réinstaller l'intégralité des PKG :

```
$ pacman -S $(pacman -Qqn) --overwrite '*'
```

Corriger l'erreur Flatpak 'CRITICAL-ERROR' :

```
$ sudo pacman --overwrite="*" -S flatpak
```

8 Droits & Permissions

Note : <https://www.leshirondellesdunet.com/chmod-et-chown>

```
$ ls -la filename  
-rw-r--r-- 1 username wheel 15908  9 déc.  11:32 filename
```

8.1 Change Mode 'chmod'

Utiliser la commande 'chmod' :

```
r=4  (droit de lecture : read)  
w=2  (droit d'écriture : write)  
x=1  (droit d'exécution : execute)  
  
$ sudo chmod 644 filename
```

8.2 Change Owner 'chown'

```
$ chown -R $USER:wheel filename  
ou  
$ chown -R username:wheel filename
```

9 Création d'une clé USB Bootable

```
$ sudo fdisk -l  
$ sudo umount /dev/sdx  
$ sudo dd if=image.iso of=/dev/sdx bs=4M status=progress && sync
```

Exemple :

```
$ sudo dd if=/home/username/Téléchargements/debian-12.0.0-amd64-netinst.iso of=/dev/sdb bs=4M status=progress && sync
```

10 Dépannage 'chroot'

Pour effectuer la maintenance des systèmes où il n'est plus possible de démarrer et/ou de se connecter.

Il faut être munie d'un clé USB Bootable.

```
# loadkeys fr  
# sudo fdisk -l
```

Montage de la partition racine '/' :

```
# mount /dev/sda3 /mnt/  
# arch-chroot /mnt
```

[dépannez, réparez...]

```
# exit  
# umount -a  
# reboot
```

11 Wine Is Not an Emulator

Installation de Wine :

```
$ sudo pacman -Syyu && trizen -Syua
$ sudo pacman -S winetricks
$ wget -P /home/$USER/Téléchargements
https://github.com/Winetricks/winetricks/archive/master.zip
$ unzip /home/$USER/Téléchargements/master.zip -d
/home/$USER/Téléchargements
$ sudo mv /usr/bin/winetricks /usr/bin/winetricks.old
$ sudo cp /home/$USER/Téléchargements/winetricks-master/src/winetricks
/usr/bin
```

Supprimer le fichier 'master.zip' & le dossier 'winetricks-master' :

```
$ rm -r ~/Téléchargements/master.zip
$ rm -r ~/Téléchargements/winetricks-master/
```

Afin d'éviter des conflits avec d'autres programmes utilisant Wine, il est préférable de créer un environnement 'WINEPREFIX' pour chaque applications.

12 Captvty

Note : <https://captvty.fr/>

Organisation des fichiers :

```
$ tree ~/Vidéos/
Vidéos/
├── captvty
│   ├── captvty-2
│   ├── captvty-3-alpha
│   └── captvty-3-beta
└──
```

```
$ tree ~/.wine/
.wine/
├── captvty
├── captvty-2
├── captvty-3-alpha
├── captvty-3-beta
└──
```

Prérequis :

```
$ sudo pacman -S lib32-gnutls lib32-libldap
$ trizen -S ttf-ms-fonts
```

À la fin de l'installation supprimer les lanceurs créés par 'VLC' et 'Wine' :

```
$ rm -r ~/.local/share/applications/wine/Programs/VideoLAN/
$ rm -r ~/Bureau/VLC\ media\ player.desktop
$ rm ~/.local/share/applications/wine-*
```

12.1 Captvty 2 (WINEARCH=win32)

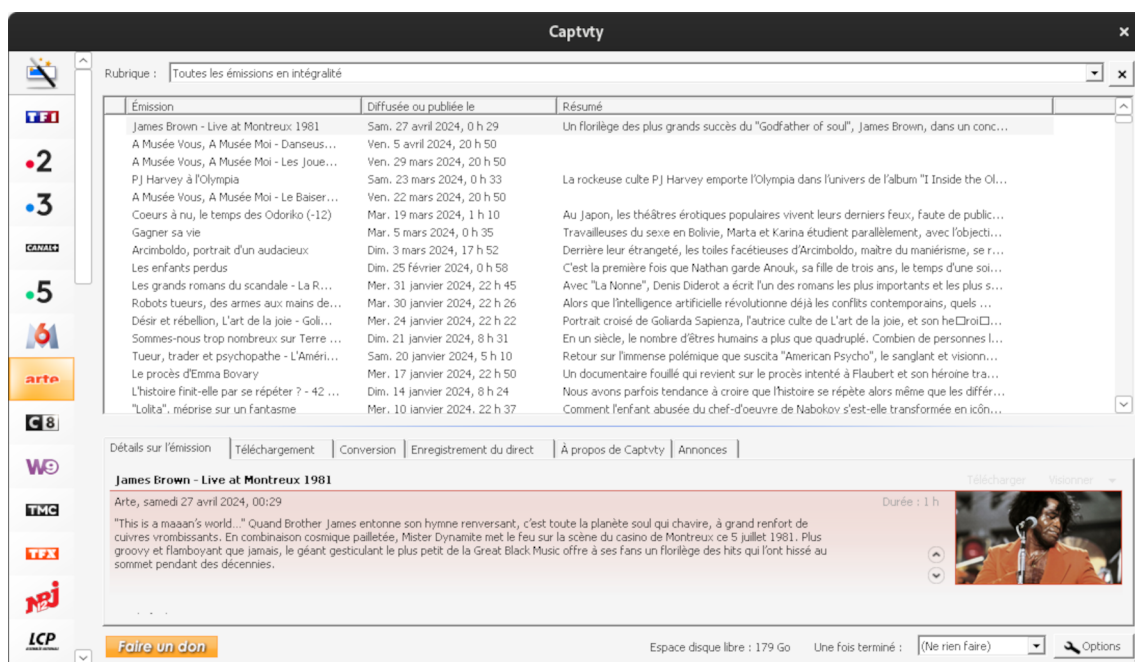
Installation :

```
$ export WINEPREFIX="/home/$USER/.wine/captvty-2"
$ export WINEARCH=win32 winecfg

$ winetricks dotnet40_kb2468871 gdiplus
$ winetricks comctl32 corefonts ie8
$ winetricks vcrun2008 vlc
```

Pour démarrer Captvty 2 :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/captvty-2 wine
$HOME/Vidéos/captvty/captvty-2/Captvty.exe"
```



12.2 Captvty 2 & 3 Alpha (WINEARCH=win32) :

Installation :

```
$ export WINEPREFIX="/home/$USER/.wine/captvty"
$ export WINEARCH=win32 winecfg

$ winetricks dotnet452 gdiplus
$ winetricks comctl32 corefonts ie8
$ winetricks vcrun2008 vcrun2010 vlc
```

Pour lancer Captvty 2 :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/captvty wine
$HOME/Vidéos/captvty/captvty-2/Captvty.exe"
```

Pour lancer Captvty 3 Alpha :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/captvty wine
$HOME/Vidéos/captvty/captvty-3-alpha/Captvty.exe"
```


12.3 Captvty 3 Alpha (WINEARCH=win64)

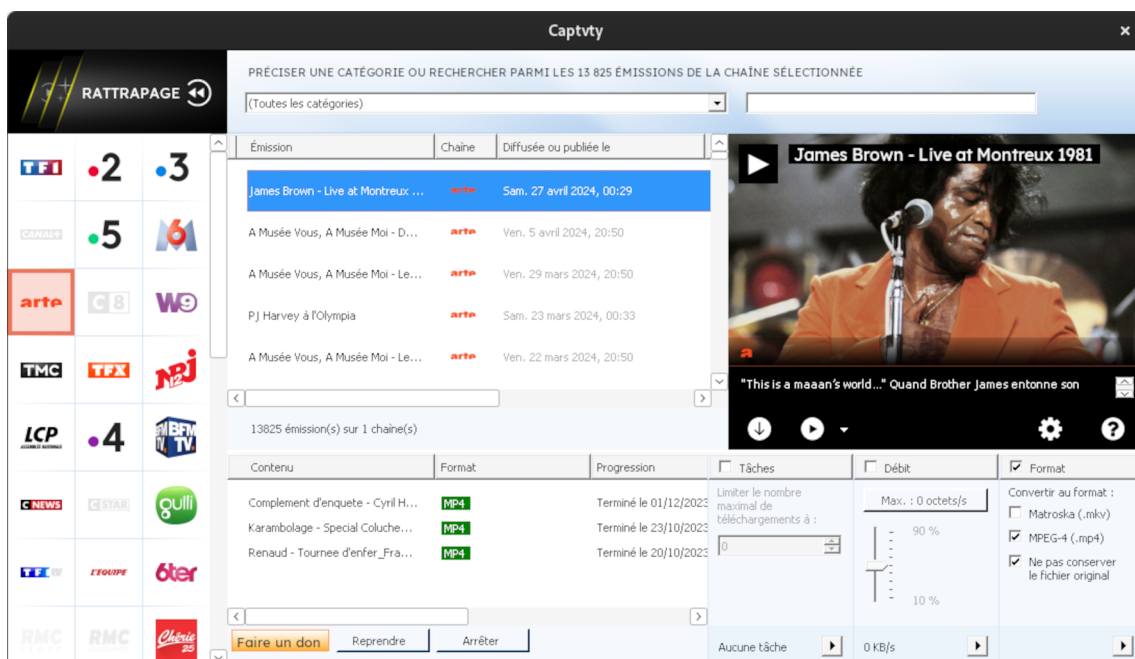
Installation :

```
$ export WINEPREFIX="/home/$USER/.wine/captvty-3-alpha"
$ export WINEARCH=win64 winecfg

$ winetricks dotnet452 gdiplus
$ winetricks comctl32 corefonts
$ winetricks vcrun2010 vlc
```

Pour démarrer Captvty 3 Alpha :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/captvty-3-alpha wine
$HOME/Vidéos/captvty/captvty-3-alpha/Captvty.exe"
```



12.4 Captvty 3 Bêta

Installation :

```
$ export WINEPREFIX="$HOME/.wine/captvty-3-beta"
$ export WINEARCH=win64
$ wineboot -u

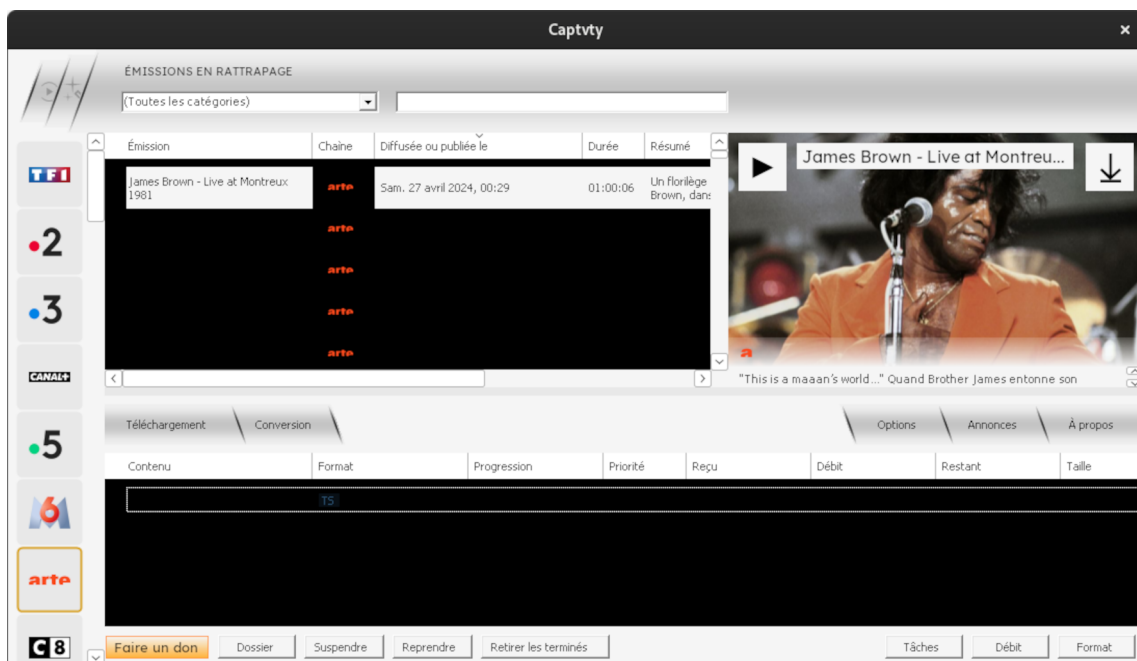
$ winetricks -q dotnet48 gdiplus comctl32 corefonts fontsmooth=rgb
vcrun2010 vlc win7

$ mkdir tmp
$ cd tmp
$ cabextract ~/.cache/winetricks/win7sp1/windows6.1-KB976932-X64.exe
$ cp -rf ./x86_microsoft.windows.common-
controls_6595b64144ccf1df_6.0.7601.17514_none_41e6975e2bd6f2b2/comctl32.dll
~/.wine/captvty-3-beta/drive_c/windows/syswow64/comctl32.dll
$ cd ~
$ rm -rf tmp

$ WINEDEBUG=fixme-all wine $HOME/Vidéos/captvty/captvty-3-beta/Captvty.exe
```

Par la suite Captvty 3 Bêta pourra être démarré ainsi :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/captvty-3-beta wine
$HOME/Vidéos/captvty/captvty-3-beta/Captvty.exe"
```



13 DCMOTO

Note : <http://dcmoto.free.fr/>

Organisation des fichiers :

```
$ tree ~/VMs/
```

```
VMs/
├──
└── Thomson
    ├── DCMOTO
    │   ├── dcmoto.exe
    │   ├── dcmoto_20230825.exe
    │   └── dcmoto_20230825.zip
    └──
```

Prérequis :

Pour ne pas avoir le message d'erreur 'waveOutOpen failed'

```
$ sudo pacman -S lib32-alsa-plugins lib32-libpulse lib32-openal
```

Renommer le fichier '.exe' en retirant la date de version, et le lui attribuer le nom '**dcmoto.exe**', ceci vous évitera de modifier le fichier '.desktop' à chaque mise à jour :

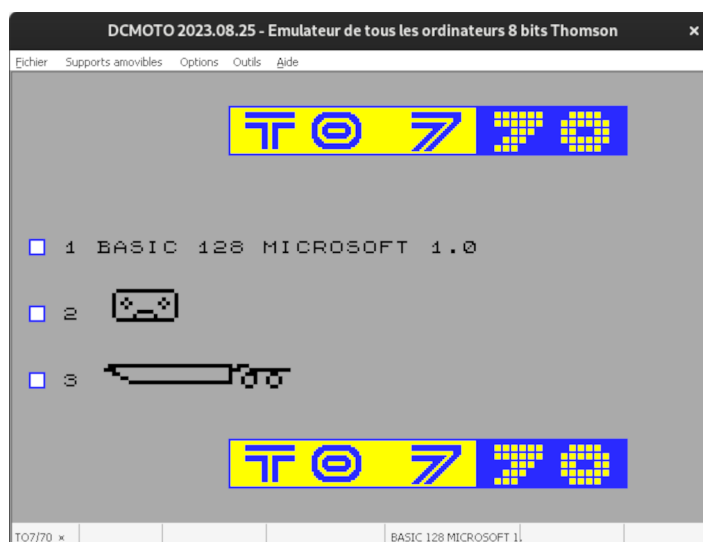
```
$ cp ~/VMs/Thomson/DCMOTO/dcmoto_20230825.exe
~/VMs/Thomson/DCMOTO/dcmoto.exe
```

Installation :

```
$ export WINEPREFIX="/home/$USER/.wine/dcmoto"
$ export WINEARCH=win32 winecfg
```

Pour lancer DCMOTO :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/dcmoto wine
$HOME/VMs/Thomson/DCMOTO/dcmoto.exe"
```



14 Sketchup

Note :

<https://appdb.winehq.org/objectManager.php?sClass=version&iId=34500#notes>

Téléchargement des fichiers '.exe' :

Version Anglaise,

<https://archive.org/download/sketchupmake-2017-2-2555-90782-en-x64>
md5 = 5c98d5afc594cdac0bcc0e5bf8c37f73

Version Française,

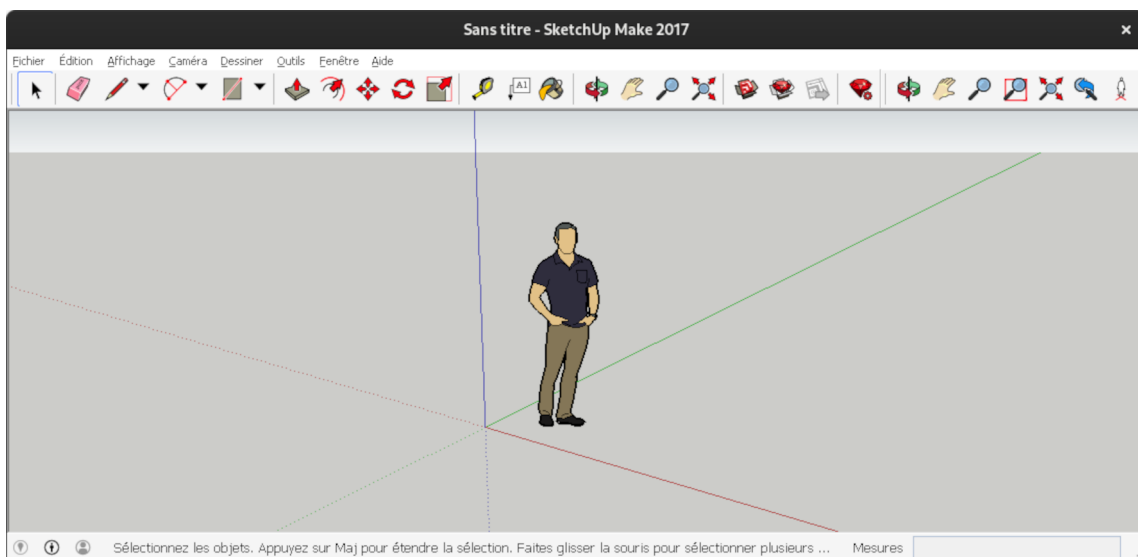
<https://archive.org/download/sketchupmake-2017-2-2555-90783-fr-x64>
md5 = f504fa0d7e108402239553e9d254962b

Installation :

```
$ export WINEPREFIX="/home/$USER/.wine/sketchup"  
$ export WINEARCH=win64 winecfg  
  
$ /usr/bin/wineutils win7  
$ /usr/bin/wineutils dotnet46  
$ wineutils corefonts riched30 richtx32 tahoma vcrun2015
```

Pour lancer SketchUP :

```
$ sh -c "env WINEPREFIX=$HOME/.wine/sketchup wine  
$HOME/Documents/config/sketchup/sketchupmake-2017-2-2555-90783-fr-x64.exe"
```



Pour supprimer les lanceurs créés :

```
$ rm ~/.local/share/applications/wine/Programs/SketchUp\  
2017/LayOut.desktop  
$ rm ~/.local/share/applications/wine/Programs/SketchUp\ 2017/Style\  
Builder.desktop  
  
$ cd ~/Bureau/  
$ rm *.desktop *.lnk
```

15 GNOME Mastermind™ (PKGBUILD)

Note :

<https://www.autistici.org/gnome-mastermind/>

<https://github.com/fargiolas/gnome-mastermind/>

Contrairement à Debian, ce jeu n'est pas disponible pour Archlinux.

Créer un dossier à l'emplacement de votre choix :

```
$ mkdir gnome-mastermind/
```

Dans ce dossier, avec un éditeur de texte, copier les lignes de code suivantes dans un fichier appelé 'PKGBUILD'

Note : [https://wiki.archlinux.org/title/PKGBUILD_\(Fran%C3%A7ais\)](https://wiki.archlinux.org/title/PKGBUILD_(Fran%C3%A7ais))

```
-----
# GNOME Mastermind™ v0.4.0

# PKGBUILD v1.1 - 2023/08/27
# Maintainer: daminus

pkgname=gnome-mastermind
pkgver=0.4.0
pkgrel=1
pkgdesc="Clone of the Mastermind™ board game for the GNOME Desktop"
arch=('x86_64')
url="https://www.autistici.org/gnome-mastermind/"
license=('GPL')
depends=('cairo' 'dconf' 'gdk-pixbuf2' 'glib2' 'glibc' 'gtk3' 'hicolor-icon-theme')
makedepends=('gnome-common')
source=(https://github.com/fargiolas/gnome-mastermind/releases/download/v0.4.0/"$pkgname-$pkgver.tar.gz")
md5sums=('799a92a34d094e2695b52da42ca8ef73')
sha256sums=('f49609712a91e1d46ef88edc6ba9322fec09433a7f3695de9d9d10449e129598')

build() {
    cd "$pkgname-$pkgver"
    ./autogen.sh
    ./configure --prefix=/usr
    make
}

check() {
    cd "$pkgname-$pkgver"
    make -k check
}

package() {
    cd "$pkgname-$pkgver"
    make DESTDIR="$pkgdir/" install
}
-----
```

Vérifier que le 'PKGBUILD' existe bien :

```
$ tree gnome-mastermind/  
gnome-mastermind/  
└─ PKGBUILD
```

Pour la création du PKG, installer le paquet 'gnome-common' qui est une dépendance nécessaire à compilation, il pourra être supprimé par la suite :

```
$ sudo pacman -S gnome-common
```

Dans le répertoire contenant le fichier 'PKGBUILD', exécutez simplement la commande 'makepkg' :

```
$ cd gnome-mastermind/  
$ makepkg
```

Et voilà vous avez un package installable !

Pour l'installer, toujours dans le même répertoire :

```
$ sudo pacman -U gnome-mastermind-0.4.0-1-x86_64.pkg.tar.zst
```



Pour ignorer la mise à jour de ce dépôt personnalisé, éditer le fichier 'pacman.conf' :

```
# nano /etc/pacman.conf  
  
IgnorePkg = gnome-mastermind
```