

## Recette de Libre OS USB :

1. Créer une installation d'Ubuntu ou un de ses dérivés (Ubuntu MATE 64 bits dans mon cas de figure) sur le disque dur d'une machine réelle ou virtualisée (comme Virtualbox le permet).
2. Installer les logiciels complémentaires que vous souhaitez voir figurer dans la future distribution (attention aux logiciels propriétaires : Skype, Teamviewer, Google Chrome).
3. Installer si nécessaire des PPA pour que la mise à jour des logiciels hors distribution se fasse facilement (prudence toutefois, un PPA qui ne répond plus peut bloquer les mises à jour).
4. Pour créer son propre dérivé d'Ubuntu il existe le projet « LinuxRespin » :  
<https://github.com/ch1x0r/LinuxRespin>  
Configurez le logiciel dans « /etc/ » et utilisez la commande « **respin backup** » en tant que super utilisateur (à l'aide de sudo) pour qu'il génère un fichier ISO sortant, attention un fichier ISO ne peut dépasser 4 Go pour être opérationnel. L'on obtient une image ISO bootant avec SysLinux (que l'on contournera par la suite).
5. Maintenant nous allons construire un amorçage plus universel (UEFI et CSM). Installez sous Windows un utilitaire de Easy2Boot (<http://www.easy2boot.com>) se nommant « MakePartImage » pour pouvoir convertir l'image ISO en fichier transformable pour l'UEFI. Suivre le mode d'emploi et convertir l'image par Drag'n Drop sur l'icône « FAT32 », on obtient un fichier ayant l'extension « .imgPTN ».
6. Téléchargez Easy2Boot (<http://www.easy2boot.com>) sur votre système, je vous conseille d'être sous Gnu/linux, et installez le système d'amorçage sur une clef USB destinée à recevoir le système (au minimum de 16 Go), pour cela il y a un script Bash fourni qui fait cela très bien en utilisant un script Bash fourni : « **/Easy2Boot/\_ISO/docs/linux\_utils/fmt.sh** », on peut aussi défragmenter la partition en FAT32 de la clé USB avec un outil fourni, voir la doc (<http://www.easy2boot.com/make-an-easy2boot-usb-drive/make-using-linux/>).
7. Puis copiez le fichier « .imgPTN » dans le répertoire « \_ISO/Linux » de la clef.
8. Démarrez la clef sur un ordinateur supportant le boot CSM/Legacy, choisir l'image que l'on vient de créer dans le menu « Linux » puis demander la transformation de la clef pour que celle-ci devienne bootable UEFI/CSM, l'opération prend quelques secondes. **Attention l'installation de l'image .imgPTN corrige la disposition des partitions.**
9. La clef doit comporter deux partitions, la première en FAT32 doit faire environ 8Go et nous l'appelons « LIBREOSUSB » avec le flag « boot », la seconde en EXT4 peut être agrandie en fonction des capacités de la clef et nous l'appelons « HOME » elle servira pour le répertoire utilisateur. Je conseille de ne pas occuper tout l'espace de la clé, car cela sert à éviter que les capacités différentes en fonction des modèles de clefs USB ne nous jouent des tours lors des copies avec les utilitaires Linux « dd » ou « rufus » sous Windows.
10. Maintenant il y a un répertoire « EFI » à la racine de la clef, mais ce n'est pas encore suffisant, il faut aller récupérer le répertoire EFI présent sur l'ISO de Ubuntu comportant le répertoire « BOOT » et le copier avec les fichiers nécessaires qui sont « BOOTx64.EFI » et « grubx64.efi ». Il faut également copier le dossier « /boot » qui contiendra les éléments nécessaires à Grub.
11. La clef devient bootable en mode UEFI grâce aux actions précédentes. Nous allons ajouter la « persistance » pour notre système. Pour cela nous allons placer un fichier « casper-rw » à la racine de la clef (maximum 4 Go) avec la commande suivante : « **dd if=/dev/zero of=casper-rw bs=1M count=4000** »,

puis : « **mkfs.ext3 -F casper-rw -L casper-rw** ».

Puis il faut ajouter dans les fichiers de démarrage de Easy2Boot les éléments qui prennent cela en compte :

**A la racine de la clef se trouve le fichier « menu.lst », il convient de l'élaguer et de lui rajouter comme lignes :**

```
title Libre OS USB\n Start Libre OS USB
kernel /casper/vmlinuz boot=casper persistent noeject noprompt live-media-path=/casper
ignore_uuid quiet splash --
initrd /casper/initrd.gz
```

**Dans le répertoire « /boot/grub/grub.cfg » de la clef, il faut introduire les lignes :**

```
set default="0"
set timeout=10
```

```
menuentry "Libre OS USB" {
linux /casper/vmlinuz boot=casper persistent quiet splash --
initrd /casper/initrd.gz
}
```

```
menuentry "Libre OS USB in safe mode" {
linux /casper/vmlinuz boot=casper persistent xforcevesa quiet splash --
initrd /casper/initrd.gz
}
```

12. Ensuite il est nécessaire de rendre les menus plus lisibles, d'enlever le superflu, de modifier l'image de fond pour le menu CSM si on le souhaite.
13. Maintenant si la clef démarre avec la persistance dans les deux modes UEFI et CSM, la première étape est jouée. Nous allons maintenant configurer le système lui-même.
14. Nous allons tout d'abord modifier le « initrd.gz » ce dernier n'est au autre qu'un fichier gzip un peu spécial placé sur la clé dans « /casper » :

Pour décompresser ce fichier :

```
# mkdir tempdir
# cd tempdir
```

puis :

```
# gunzip -dc ../initrd.gz | cpio -imvd --no-absolute-filenames
```

Ou si cette méthode ne fonctionne pas correctement, il faut installer « dracut » puis utiliser un de ses scripts, comme suit :

```
# /usr/lib/dracut/skipcpio ../initrd.gz |zcat| cpio -id --no-absolute-filenames
```

Pour recompresser notre fichier « initrd.gz » il faudra utiliser :

```
# find . | cpio -o -H newc | gzip -9 > ../initrd.gz
```

voir cette adresse : <https://help.ubuntu.com/community/LiveCDCustomization>). Pour que le répertoire utilisateur « /home » soit monté avec la partition dédiée « HOME » de la clé que nous avons créée au préalable en EXT4 au démarrage de la clé, il faut modifier

« `usr/share/initramfs-tools/scripts/casper-bottom/12fstab` » se trouvant dans l'arborescence du « `initrd.gz` » que nous personnalisons, il faut utiliser l'uuid de la partition « HOME » et recopier les fichiers qui se trouvent dans l'ancien répertoire utilisateur (pensez au répertoires cachés).

15. Pour pouvoir mettre à jour et installer des logiciels il faut que le `/etc/apt/sources.list` de la clé soit bien configuré, pour cela il faut veiller à ne pas laisser de **ppa** douteux et ne pas inclure la source « CD-ROM » en désactivant le fichier `/scripts/casper-bottom/41apt_cdrom` situé lui aussi dans le « `initrd.gz` »
16. Ensuite si besoin (**LinuxRespin en mode backup permet déjà de créer un utilisateur spécifique**) nous allons créer un nouvel utilisateur se nommant « libreosusb » et nous allons en faire l'utilisateur par défaut. Il faut lui attribuer les groupes qui étaient attribués à l'utilisateur ubuntu auparavant (l'utilisateur des Live CD).

#### Option :

Il sera nécessaire pour cela de modifier le « `initrd.gz` ». Tous les scripts se trouvent dans le répertoire « `/usr/share/initramfs-tools/scripts/casper-bottom/` ». Dans un de ses scripts vous pourrez désactiver l'autologin si besoin.

#### Méthode :

Travailler directement sur le système de fichier en lecture seule :

Faites une copie du fichier « `filesystem.squashfs` » se trouvant dans le « `/casper` » de la clé, puis nous allons pouvoir le personnaliser :

```
# unsquashfs filesystem.squashfs
# mv filesystem.squashfs /path/to/backup/
```

Allez dans « `squashfs-root` », ajouter, modifier à votre convenance, et recréez le « `filesystem.squashfs` » comme ceci :

```
# cd /path/to/workdir
# sudo mksquashfs squashfs-root filesystem.squashfs -b 256k -comp lzo -e boot
```

L'utilisateur « `libreosusb` » devra être présent dans « `/etc/group` », « `/etc/passwd` » et « `/etc/shadow` » de ce système en lecture seule.

Vous pourrez recopier le nouveau « `filesystem.squashfs` » à la place de l'ancien sur votre clé USB.

17. Pour personnaliser l'image de démarrage de Libre OS USB, nous pouvons modifier les logos du thème par défaut de « Plymouth » toujours dans le « `initrd.gz` ». Attention il faut que ce fichier soit présent sur la partition « LIBREOSUSB » mais aussi dans le « `filesystem.squashfs` », c'est-à-dire notre environnement une fois booté. Dans ces fichiers en lecture seule les thèmes (graphique et texte) seront copiés dans « `/usr/share/plymouth/themes/` ». Attention aux liens symboliques et aux fichiers de configuration des thèmes. Thème utilisé : **[Spinning Dandelions Linux Mint 18\(Ubuntu 16.04\) Bootsplash](#)**
18. **Si l'on veut rendre le répertoire « `/home/utilisateur` » totalement crypté**, et le décrypter pendant la session de l'utilisateur après la saisie du mot de passe, Nous utiliserons « `Ecryptfs` » et ses « `ecryptfs-utils` », en utilisant un script par ce biais : « `ecryptfs-migrate-home -u libreosusb` », voir la documentation de cette page : **<https://wiki.archlinux.org/index.php/ECryptfs>**

19. Si l'on ne veut que seulement rendre quelques répertoires utilisateurs cryptés, nous utiliserons « Encfs » et son outil « Gnome Encfs Manager » qui démarrera avec la session utilisateur. Voir <https://doc.ubuntu-fr.org/encfs> pour plus d'informations.
20. Dans le fichier « initrd.gz » il faut mettre en accord le fichier « /etc/hostname » et « /etc/hosts », la commande « sudo » affiche un message d'avertissement dans le cas contraire.
21. Il est nécessaire de bloquer la version du kernel pour éviter sa mise à jour qui provoquerait un plantage de la clef. En éditant le nouveau fichier « /etc/apt/preferences.d/kernel » (dans le filesystem.squashfs) en y ajoutant le texte suivant :

```
Package: linux-generic linux-headers-generic linux-image-generic
Pin: version <insert version here>
Pin-Priority: 1001
```

<http://askubuntu.com/questions/178324/how-to-skip-kernel-update>

22. Il ne reste plus qu'à faire des arrangements d'ordre esthétique et ergonomique, comme ajouter le choix du clavier la barre de tray, s'assurer que la configuration du réseau est disponible ainsi que celle du son. Aussi il est utile de désinstaller « ubiquity » pour ne pas se retrouver avec l'icône d'installation, ainsi que « **grub2** » qui n'est pas nécessaire mais dont la mise à jour pourrait être problématique.
23. Enfin il faut supprimer les traces laissées par le passage en mode création, logs, historiques, etc. La commande « history -w » suivi de « history -c » effacent les traces du terminal, ou encore en vidant le fichier ~/.**bash\_history**  
Il est bon aussi si besoin de désactiver des groupes à l'origine pour l'utilisateur « live » ubuntu et lui supprimer son répertoire home.
24. Pour utiliser Virtualbox avec l'image physique de la clé, il faut créer au préalable une configuration avec un disque virtuel relié au fichier image en appliquant cette commande :  
VBoxManage internalcommands createrawvmdk -filename LibreOSUSB.vmdk -rawdisk 'Libre\_OS\_USB\_VX (virtualbox).img'
25. Pour localiser le layout du clavier dans LightDM si besoin, il faut créer le fichier « /etc/X11/xorg.conf.d/20-keyboard.conf » avec ce contenu :

```
Section "InputClass"
    Identifier "keyboard"
    MatchIsKeyboard "yes"
    Option "XkbLayout" "fr"
    Option "XkbVariant" "nodeadkeys"
EndSection
```