

Compilation de commandes utiles régulièrement, ou pas

Ce document est réalisé à partir de mes expériences diverses sur GNU/Linux, principalement Debian et depuis peu avec Awesome WM.

Beaucoup de documentation existe sur le sujet, je ne vais donc pas réécrire ce qui existe déjà en mieux, mais plutôt partager des informations spécifiques qui m'ont demandé un peu de temps de recherche.

Table des matières

1	Commandes usuelles	4
2	Installations	4
2.1	Système	4
2.1.1	Créer une clé USB Bootable	4
2.1.2	Créer une clé USB live avec plusieurs ISOs	5
2.1.3	Vérification d'ISO	5
2.1.4	Retirer le bip du clavier	5
2.1.5	Faire apparaître GRUB au démarrage	5
2.2	Terminal	6
2.2.1	Changer de shell - Zsh	6
2.2.2	Ajouter des plugins facilement dans Vim	6
2.2.3	Colorer les codes couleurs html/rgb etc	6
2.3	Disques chiffrés	6
2.3.1	Monter un disque chiffré en ligne de commande	6
2.3.2	Monter un disque chiffré automatiquement au démarrage	7
2.4	Customisation, paramétrage	7
2.4.1	Installer une police au format fichier .ttf	7
2.4.2	Changer l'image de grub (écran de démarrage)	7
2.4.3	Utiliser le pavé numérique comme souris	8
2.4.4	Luminosité	9
2.4.5	Changement image de profil	10
2.5	Chercher et supprimer des fichiers en doublon	10
2.6	Compresser des images en ligne de commande	10
3	Manjaro - Arch Linux	10
3.1	Ressources diverses	10
4	Awesome WM et la customisation avancée	10
4.1	Où sont les fichiers ?	11
4.2	Dépendances	11
5	Yunohost, Brique Internet et autres serveur-like tips	12
5.1	Brique Internet sur clé 4G	12
5.1.1	Installation, post-installation	12
5.1.2	Monter la clé 4G au démarrage comme interface réseau	12
5.2	Utilisation de deux serveurs avec un domaine principal et un sous-domaine	13
5.2.1	Enlever le bouton Yunohost permanent dans les applications	13
6	Sauvegardes	14
6.1	Android - Utilisation de adb	14
6.2	GNU-Linux	14
6.2.1	Le paquet rsync	14
6.2.2	Sauvegarde complète de Debian avec un script	15
7	Logiciels	15
7.1	Graphisme	15
8	Développement Web	15
8.1	HTML	15
8.1.1	Centrer une image	15
8.1.2	Faire un fond de page web qui change aléatoirement de couleur	16
9	Internet	16
9.1	Flux RSS	16
9.2	Multimédia	17
9.2.1	youtube-dl	17
9.2.2	Spotify pour Debian 9	17
9.2.3	Télécharger des playlists Spotify	17

10 Vie privée	18
10.1 Utiliser le VPN Riseup Black	18
10.2 Changer son adresse mac	18
11 Bugs divers	19
11.1 Bugs après une installation	19
11.1.1 Driver wifi - Debian x bcm43xx	19
11.2 Bugs dans le terminal	19
11.2.1 ./configure say no compiler found	19
12 Divers	20
12.1 Wallabako - synchronisation wallabag et kobo	20
12.2 Contrôler son téléphone Android - scrpy	20
12.3 Stylet, PC tablette, etc	20
12.3.1 Activer le clic droit du stylet	20
12.4 ASCII art generator	20
12.5 Comment créer un livre électronique au format ePub avec LibreOffice	20
12.6 Taper le point median sur Linux	20
13 Windows	21
14 Annexes - Scripts	21
14.1 Trier des fichiers par année	21
14.2 Post-Installation automatique	21
14.3 Compression de PDF	22
14.4 Écrire soit-même des fortunes en semi-automatique	23
14.5 Convertir une archive Youtube en dossier de favoris	24
14.5.1 prog.sed	24
14.5.2 header.txt	24
14.5.3 rm.sh	25

1 Commandes usuelles

Si l'on utilise pas quotidiennement son terminal, on se retrouve régulièrement à avoir à rechercher des commandes usuelles sur internet car on les a oubliées.

L'objectif de cette page est donc de regrouper quelques commandes qui reviennent assez souvent.

Pour une vraie documentation/présentation de ces commandes, je conseille la très bonne page : <https://linuxpedia.fr/doku.php/commande/commande>.

Pipe ou envoyer le résultat d'une commande dans une autre

Le signe `|` permet de renvoyer le résultat d'une commande dans la suivante au lieu de l'afficher dans le terminal :

```
commande1 | commande2
```

Le résultat de la commande1 est utilisé comme argument de commande2.

Cat ou afficher le contenu d'un fichier

La commande `cat` permet d'afficher le contenu d'un fichier grâce à :

```
cat fichier.txt
```

Télécharger une page internet

La commande `curl exemple.com` affichera dans votre terminal le contenu de la page demandée.

La commande `wget exemple.com` téléchargera ce contenu dans un fichier html. Il est possible de l'utiliser récursivement pour télécharger tout un site.

Ajouter du contenu dans un fichier

Pour ajouter du contenu dans un fichier sans avoir à l'ouvrir, on peut utiliser `»`.

Ainsi `echo nouvelle ligne » fichier.txt` ajoutera "nouvelle ligne" dans fichier.txt.

On peut aussi l'utiliser avec une commande, par exemple :

```
curl exemple.com >> fichier.txt
```

permettra d'ajouter le contenu de exemple.com dans fichier.txt, et crée le fichier si il n'existe pas.

Trouver des caractères ou des fichiers

Pour ces tâches on utilisera les commandes `grep` et `find`, la première pour trouver des caractères/mots/lignes dans un fichier et la deuxième pour trouver des fichiers dans un répertoire selon leur nom ou extension par exemple.

2 Installations

2.1 Système

2.1.1 Créer une clé USB Bootable

Attention, cette commande nécessite de savoir précisément l'identification de votre clé usb, sinon vous risquez de perdre toutes vos données en formattant votre disque dur.

Pour plus d'informations : https://doc.ubuntu-fr.org/live_usb#en_ligne_de_commandes

```
sudo dd bs=4M status=progress if=votre.iso of=/dev/sdX && sync
```

2.1.2 Créer une clé USB live avec plusieurs ISOs

Il est également possible d'utiliser plusieurs images dans une seule clé. Pour cela on peut installer `multibootusb` (http://multibootusb.org/page_download/). Pour l'installer :

```
sudo dpkg -i python3-multibootusb_9.2.0-1_all.deb
```

Si vous rencontrez des problèmes de dépendances non satisfaites, relevez les noms de ces paquets, installez les puis répétez la commande ci-dessus.

Exécutez ensuite le programme avec `sudo multibootusb`.

2.1.3 Vérification d'ISO

La vérification d'iso consiste à vérifier l'authenticité et l'intégrité des images téléchargées. La méthode exacte sera spécifique à chaque .iso car les fichiers et signatures ne seront pas la même, cette partie présentera donc des commandes usuelles mais ne permet pas de réaliser toutes les étapes sans regarder la documentation spécifique à la vérification souhaitée.

TENTATIVE D'EXPLICATION : l'authenticité permet de vérifier que le fichier téléchargé a bien été créé par la personne qui dit l'avoir signée. L'intégrité permet de vérifier que le fichier n'a pas été modifié par une personne tiers. Ces méthodes **ne peut pas prouver que le fichier n'est pas malicieux**, on doit donc choisir de nous-même si l'on fait confiance aux personnes qui ont créé ces fichiers. Tout d'abord, télécharger dans le même fichier le .iso (ou autre format, mais le fichier de l'image logiciel) et les fichiers de signature SHA256sum.txt et SHA256sum.txt.gpg (de même, les extensions de fichier peuvent varier).

Il faut ensuite générer le SHA256sum de l'iso téléchargé afin de le comparer au fichier téléchargé. Générer le SHA256sum :

```
sha256sum -b *.iso
```

Il faut ensuite comparer la signature obtenue à celle contenue dans SHA256sum.txt. Une méthode permet de le faire à partir d'une ligne de commande :

```
sha256sum --ignore-missing -c sha256sum.txt
```

Vérification d'authenticité

Pour vérifier l'authenticité de SHA256sum.txt il faut vérifier la signature de SHA256sum.txt.gpg. Il faut tout d'abord importer la clé de signature du système d'exploitation, pour *Debian* :

```
gpg --keyserver keyring.debian.org --recv-keys 0xDEADBEEF
gpg --verify sha256sum.txt.gpg sha256sum.txt
```

2.1.4 Retirer le bip du clavier

Il arrive qu'après une nouvelle installation le clavier *bip* en cas d'entrée vide. Pour l'enlever :

```
echo "blacklist pcspkr" | sudo tee /etc/modprobe.d/nobeep.conf
```

Le bruit aura disparu après un redémarrage.

2.1.5 Faire apparaître GRUB au démarrage

Après une nouvelle installation de Manjaro, je n'avais pas accès à GRUB au démarrage et je ne pouvais pas changer de noyau. Cela était dû au fait que GRUB était configuré comme caché. On peut le modifier dans son fichier de configuration : `/etc/default/grub`. Il suffit de changer

```
GRUB_TIMEOUT_STYLE="hidden"
```

en

```
GRUB_TIMEOUT_STYLE="menu"
```

Il est aussi possible de changer le temps d'affichage de GRUB avec la variable `GRUB_TIMEOUT`. Pour que les changements soient pris en compte :

```
sudo update-grub
```

2.2 Terminal

2.2.1 Changer de shell - Zsh

Le shell par défaut des principales distributions Linux est *bash*, un autre shell populaire offrant de nombreuses fonctionnalités est *zsh* et son extension populaire *oh-my-zsh*. Pour l'utiliser il faut d'abord l'installer :

```
sudo apt install zsh
```

pour Arch Linux ou Manjaro :

```
sudo pacman -S zsh
```

On peut vérifier qu'il est bien installé avec :

```
zsh --version
```

Pour changer de shell par défaut et pour passer à zsh :

```
chsh -s $(which zsh)
```

Il faut ensuite se déconnecter pour mettre à jour le changement.

Installer oh-my-zsh

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"
```

Wiki : <https://github.com/ohmyzsh/ohmyzsh/wiki>

Il est maintenant possible de configurer à nos souhaits ce shell via le fichier `.zshrc`.

Deux vidéos intéressantes pour comprendre les fonctionnalités disponibles et comment les utiliser :

<https://www.youtube.com/watch?v=4KBuPCeF9Gc>

<https://www.youtube.com/watch?v=soAwUq2cQHQ>

2.2.2 Ajouter des plugins facilement dans Vim

<https://github.com/VundleVim/Vundle.vim>

2.2.3 Colorer les codes couleurs html/rgb etc

Un plugin pour surligner de la couleur correspondante les codes couleurs dans Vim :

<https://github.com/gko/vim-coloresque>

2.3 Disques chiffrés

2.3.1 Monter un disque chiffré en ligne de commande

Monter un disque chiffré ne requiert pas les mêmes commandes que pour un disque normal, mais c'est finalement assez simple.

Tout d'abord vous devez bien entendu avoir `cryptsetup` d'installé sur votre système.

```
sudo apt install cryptsetup
```

Ensuite on aura ces opérations à réaliser :

```
sudo cryptsetup luksOpen /dev/sdc1 cryptdrive
sudo mkdir /media/MONDISQUE
sudo mount /dev/mapper/cryptdrive /media/MONDISQUE/
```

Le `/dev/sdc1` donne l'information de la **partition** que vous voulez monter, et `cryptdrive` est le *mapper* que vous nommez pour que le système reconnaisse le disque.

La deuxième ligne est optionnelle si l'emplacement de montage souhaité existe déjà.

La dernière commande est la commande habituelle de montage : identification du disque à monter et l'emplacement où le monter.

2.3.2 Monter un disque chiffré automatiquement au démarrage

Il est possible de monter automatique le disque chiffré sans avoir à renseigner la phrase de déchiffrement si vous configurez un fichier "clé" et l'utilisez au démarrage.

Mon "tuto" est seulement un rappel des commandes nécessaires, sans grandes explications car tout est bien détaillé dans la source que j'ai utilisé. Si cela ne fonctionne pas parfaitement il y a plus de détails dans l'article source. Source : <https://blog.tinned-software.net/automount-a-luks-encrypted-volume-on-system-start/>

On commence par créer un fichier `disksecretkey` rempli aléatoirement qui servira à contenir la clé de déchiffrement. Il faudra peut être créer le dossier `luks-keys` si il n'existe pas.

```
dd if=/dev/urandom of=/etc/luks-keys/disksecretkey bs=512 count=8
```

On peut ensuite lier ce fichier à notre disque et sa phrase de déchiffrement :

```
sudo cryptsetup -v luksAddKey /dev/sdb1 /etc/luks-keys/disksecretkey
```

Cela vous demandera la phrase de déchiffrement de votre disque (attention à bien renseigner votre partition à la place de `sdb1`).

Ensuite on récupère le UUID du disque pour l'utiliser dans les règles au démarrage :

```
sudo cryptsetup luksDump /dev/sdb1 | grep "UUID"
```

On ajoute une règle dans le fichier `/etc/crypttab` :

```
sdb1-crypt UUID=2a2375bf-2262-413c-a6a8-fbeb14659c85 /etc/luks-keys/disksecretkey luks
```

Enfin, pour qu'il soit monté au démarrage on ajoute une ligne au fichier dédié : `/etc/fstab`

```
/dev/mapper/sdb1-crypt /media/gerhard/Daten ext4 defaults 0 2
```

Pour des règles plus spécifiques consultez la page dédiée à `fstab` : <http://man7.org/linux/man-pages/man5/fstab.5.html>

2.4 Customisation, paramétrage

2.4.1 Installer une police au format fichier .ttf

Pour l'installer seulement pour l'utilisateur·ice :

```
cp exemple.ttf .fonts
```

Pour l'installer sur tout le système :

```
cp exemple.ttf /usr/share/fonts
```

2.4.2 Changer l'image de grub (écran de démarrage)

Il est possible de changer l'image d'arrière plan de GRUB, l'utilitaire de démarrage de Debian (et d'autres distribution) qui permet de choisir quelle distribution lancer.

L'image utilisée doit cependant respecter un format spécifique, elles peuvent être png, jpg/jpeg ou tga mais avec ces caractéristiques :

1. les images doivent être 8-bits (256 couleurs)
2. les images doivent être non-indexées, RGB

Des convertisseurs d'images pour les transformer sous ces caractéristiques sont trouvables sur *internet*.

Par défaut, si le paquet *desktop-base* est installé les images utilisables sont trouvables dans le répertoire `/usr/share/images/desktop`. GRUB va chercher l'image à afficher selon un ordre particulier présenté ici par ordre de priorité :

1. la ligne `GRUB_BACKGROUND` dans `/etc/default/grub`
2. la première image trouvée dans `/boot/grub/`
3. l'image spécifiée dans `/usr/share/desktop-base/grub_background.sh`
4. le fichier listé à la ligne `WALLPAPER` dans `/etc/grub.d/05_debian_theme`

Le plus simple étant donc de modifier ou créer la ligne `GRUB_BACKGROUND=` dans `/etc/default/grub` en ajoutant le chemin du fichier derrière le `=`. *pour modifier ce fichier, il est nécessaire d'avoir les droits administrateur.* **Pour avoir la modification effective, il faut utiliser la commande :**

update-grub

CHANGER LES COULEURS DU TEXTE :

On peut également changer les couleurs du texte grâce aux 3 options suivantes :

1. `menu_color_highlight` : la couleur de l'entrée menu surlignée et son fond à l'intérieur du menu.
2. `menu_color_normal` : la couleur de l'entrée menu non-sélectionnée et son fond
3. `color_normal` : la couleur du texte et son fond à l'extérieur du menu.

La syntaxe à utiliser est la suivante : `menu_color_highlight=fg-color/bg-color` Et la liste des couleurs supportées est :

black
blue
brown
cyan
dark-gray
green
light-cyan
light-blue
light-green
light-gray
light-magenta
light-red
magenta
red
white
yellow

Pour changer les couleurs, il faut ensuite ouvrir `/etc/grub.d/05_debian_theme` et trouver la ligne :

```
if [ -z "${2}" ] && [ -z "${3}" ]; then
    echo " true"
fi
```

Et la remplacer par :

```
if [ -z "${2}" ] && [ -z "${3}" ]; then
    # echo " true"
    echo "    set color_highlight=red/green"
    echo "    set color_normal=light-cyan/black"
fi
```

Ne pas changer le "black" dans `color_normal` sinon l'image ne sera pas transparente à l'endroit où le menu est affiché. Pour avoir la modification effective et la visualiser, il faut utiliser la commande suivante et redémarrer update-grub

Il est également possible d'expérimenter en "live" au démarrage en arrêtant le compte à rebours (en tapant n'importe quelle touche) et en tapant "C" pour entrer dans la ligne de commande GRUB.

Vous pouvez alors expérimenter à votre souhait :

```
grub> set color_highlight=red/green
grub> set color_normal=light-cyan/black
```

Pressez ESC pour voir les modifications et recommencez si cela ne vous convient pas.

2.4.3 Utiliser le pavé numérique comme souris

Il est possible d'utiliser le pavé numérique comme une souris ! Pour l'activer de manière non permanente il suffit d'utiliser :

```
setxkbmap -option keypad:pointerkeys
```

Ensuite pour l'utiliser il faut utiliser la combinaison **Shift+NumLock** et le pavé numérique fonctionnera comme une souris avec des raccourcis spécifiques.

Pour se déplacer :

- 7, 8, 9 pour un déplacement vers le haut
- 4, 6 pour gauche et droite
- 1, 2, 3 pour le bas

Pour cliquer c'est légèrement compliqué :

- / sélectionne le clic gauche (appuyer deux fois pour un double clic, trois fois pour un triple)
- * sélectionne le clic du milieu
- - sélectionne le clic droit

Une fois que le clic est sélectionné il est seulement **actif** et il faut donc utiliser une autre touche pour l'utiliser :

- 5 ou + pour réellement cliquer
- 0 / ins pour cliquer et garder actif (pour faire glisser)
- . / Del pour libérer

Pour garder cette configuration au redémarrage il faut ajouter l'option dans le fichier de configuration, ou le créer si besoin :
/etc/X11/xorg.conf.d/00-keyboard.conf

```
Section "InputClass"
    Identifier "system-keyboard"
    MatchIsKeyboard "on"
    Option "XkbOptions" "keypad:pointerkeys"
EndSection
```

SOURCE : https://linuxreviews.org/HOWTO_use_the_numeric_keyboard_keys_as_mouse_in_XOrg

2.4.4 Luminosité

Pour régler la luminosité en ligne de commande, on utilise **xrandr**.
Tout d'abord on cherche l'écran que l'on veut paramétrer :

```
~$ xrandr
Screen 0: minimum 320 x 200, current 1366 x 768, maximum 8192 x 8192
LVDS-1 connected primary 1366x768+0+0 (normal left inverted right x axis y axis)
```

On voit donc que l'écran connecté est LVDS-1. Pour le baisser à 50% de luminosité :

```
xrandr --output LVDS-1 --brightness 0.5
```

Pour toute autre valeur on ajuste le nombre renseigné (1 pour avoir 100%, 0.1 pour 10% etc).

J'ai trouvé une manière de changer la luminosité après quelques recherches, c'est probablement une manière beaucoup trop compliquée de faire mais c'est pratique pour apprendre les commandes et ça fonctionne.
J'ai fait trois scripts différents, un pour augmenter de 10% la luminosité, un pour la baisser et un dernier pour juste afficher la luminosité actuelle en pourcentage.
Ces scripts sont composés de seulement une ligne mais avec plusieurs commandes dedans. Ainsi, celle-ci permet de baisser la luminosité de 10% :

```
xrandr --output eDP-1 --brightness $(echo "$(xrandr --current --verbose|grep -i Brightness | sed s/"Brightness: " //g)-.1" | bc)
```

On retrouve la commande de départ avec le **-output** avec le nom de votre écran puis **-brightness** qui prend la valeur en décimale qu'on veut pour l'écran (1.0 pour 100% et 0.1 pour 10%). On cherche d'abord la luminosité actuelle.

```
xrandr --current --verbose|grep -i Brightness
```

Donne quelque chose comme :

```
Brightness: 1.0
```

On utilise alors :

```
| sed s/"Brightness: " //g
```

Le signe | permet d'envoyer le résultat de la commande à gauche dans la commande à droite.

Cela permet de récupérer seulement la valeur décimale. Puis on utilise **-.1** pour enlever 10% à la valeur trouvée.

En **bash** on ne peut pas faire du calcul en décimal puisque les valeurs sont considérés comme des chaînes de caractères. Il faut alors utiliser **bc** : <http://jlbicquelet.free.fr/unix/bc.php>.
bc prend comme argument la valeur de la parenthèse (à l'aide du signe \$) et fait le calcul, qui nous rend donc la valeur

actuelle de la luminosité moins 10%.

De la même manière, on a la commande pour augmenter de 10% :

```
xrandr --output eDP-1 --brightness $(echo "$(xrandr --current --verbose|grep -i Brightness | sed s/"Brightness: " //g)+.1" | bc)
```

Enfin, j'ai fait une commande pour récupérer la valeur actuelle de la luminosité et la convertir en pourcentage :

```
echo $(xrandr --current --verbose|grep -i Brightness | sed s/"Brightness: " //g)*100 | bc
```

On retrouve la même base sauf que le calcul est seulement une multiplication par 100 de la valeur actuelle. Cette commande est utilisée pour mon widget de luminosité.

2.4.5 Changement image de profil

Pour modifier l'image affichée sur l'écran de connexion.

```
sudo nano /etc/lightdm/lightdm-gtk-greeter.conf
```

Le paquet accountsservice résout une dépendance et permet d'utiliser mugshot qui définit la photo de profil du compte.

2.5 Chercher et supprimer des fichiers en doublon

On utilise le paquet fdupes :

```
fdupes [options] DIRECTORY
```

Pour chercher les doublons récursivement dans un dossier et les supprimer :

```
fdupes -r -d DIRECTORY
```

SOURCES : : <https://linux.die.net/man/1/fdupes>

Une autre solution est d'utiliser fslint :

```
sudo apt install fslint
```

2.6 Compresser des images en ligne de commande

```
sudo apt-get install jpegoptim
```

3 Manjaro - Arch Linux

3.1 Ressources diverses

Post-Installation :

<https://www.youtube.com/watch?v=jl4nyEA-F-o> https://wiki.archlinux.org/index.php/Xfce#Use_a_different_window_manager https://wiki.archlinux.org/index.php/Uniform_look_for_Qt_and_GTK_applications

4 Awesome WM et la customisation avancée

La plupart des distributions utilisent par défaut des gestionnaires de fenêtres "flottantes" : XFCE, MATE, GNOME, etc. Il existe cependant des gestionnaires en mosaïque (Tiling managers) qui permettent de gérer plus précisément ses fenêtres et tout son environnement graphique (à condition de bien vouloir y passer quelques heures de configuration). J'en ai rapidement testé plusieurs mais *Awesome* est celui dont j'ai le mieux réussi la prise en main et donc que j'utilise en ce moment. En parallèle de ce gestionnaire j'utilise : kitty en terminal, vim en éditeur, compton pour la transparence, ranger pour les fichiers et conky pour mon affichage de données en continu sur le bureau.

4.1 Où sont les fichiers ?

Souvent, l'emplacement des fichiers de configuration est "évidente" pour les personnes partageant leurs codes sur internet, et pourtant...

J'essaye d'utiliser *Vim* pour éditer mes fichiers. Parce que apparemment c'est la référence pour être efficace et qu'il est configurable pour avoir de la saisie semi-automatique dans LateX. Il est dur de prise en main mais super pratique quand on (commence) à s'y faire.

Pour l'édition des thèmes il est presque indispensable grâce à l'extension que j'ai installée qui surligne les codes des couleurs par leur couleur correspondante.

```
~/ .vim/  
~/ .vimrc
```

Pour les raccourcis : <https://vim.rtorr.com/> Pour le gestionnaire de fichier j'utilise au maximum ranger qui permet d'afficher les fichiers textes sans avoir à les ouvrir, très pratique pour explorer les dossier de code. Il s'utilise au clavier et peut être super rapide et pratique une fois qu'on s'y fait.

J'utilise : <https://ranger.github.io/cheatsheet.png> pour les raccourcis. Les fichiers de configurations sont dans :

```
~/ .config/ranger/
```

Ici les modifications sont le fait d'afficher les fichiers cachés et d'activer l'affichage des images directement dans kitty.

Pour le terminal, j'utilise kitty. Il est sobre et permet d'afficher les images en utilisant ranger.

```
~/ .config/kitty/
```

Contient le thème dans *theme.conf* basé sur mon thème actuel.

J'ai installé et configuré Conky pour avoir un affichage permanent de pas mal de données sur le Bureau. Tout est dans un fichier *.conkyrc* dans le dossier utilisateur-ice. Il utilise une image dans *Images/ImagesConfig/*. Il dépend surtout des polices installées et les données sur le réseau utilise le nom d'interface réseau qui peut être différente selon les ordinateurs (pour l'ethernet comme pour le wifi).

Pour la **transparence**, j'utilise compton et son fichier de configuration *.config/compton.conf*. Il permet de choisir la transparence des fenêtres actives/inactives, les exclusions, et "l'ombre" sur les fenêtres inactives.

Pour les thèmes, les fichiers sont à trouver et/ou copier ici :

```
~/ .themes/  
~/ .fonts/  
~/ .icons/
```

Pour Awesome j'utilise du code trouvé sur internet, et plusieurs dossiers de dépendances pour les widgets. Le fichier *rc.lua* sert à définir le fonctionnement global (comportement des fenêtres, programmes au démarrage, par défaut, raccourcis clavier, etc) puis dans le dossier **themes** on trouve tout ce qui est esthétique.

Le dossier où tout se passe est ici :

```
~/ .config/awesome/
```

4.2 Dépendances

```
sudo apt install awesome awesome-extra i3lock conky compton kitty vim-runtime ranger lxappearance pulsemixer  
papirus-icon-theme numix-icon-theme
```

J'utilise pulsemixer parce que j'ai réussi à le configurer dans mes raccourcis clavier pour qu'il soit compatible avec mon casque bluetooth.

Lxappearance est nécessaire pour changer le thème utilisé. En effet le programme de xfce ne fonctionne pas dans Awesome (les modifications ne sont pas prises en compte).

J'utilise kitty parce que c'est un terminal très épuré, il a aussi l'avantage de pouvoir afficher les images en étant dans ranger (gestionnaire de fichiers intégré au terminal, avec beaucoup de raccourcis clavier, les mêmes que Vim).

Vim est la référence apparemment pour éditer le code grâce à ses dizaines de raccourcis clavier, je l'ai pas mal configuré pour pouvoir écrire plus facilement du Latex. Et j'utilise un plugin très pratique qui affiche les couleurs du code (les codes couleurs RVB et HTML sont surlignés de la couleur correspondante, très pratique pour l'édition des thèmes).

Sur GitHub : [i3lock-fancy](#)

5 Yunohost, Brique Internet et autres serveur-like tips

5.1 Brique Internet sur clé 4G

Le projet `labriqueinter.net` permet d'installer un mini-serveur sur une carte ARM ou Raspberry (ou autre) avec la particularité de créer un réseau wifi lié à un VPN. Tous les appareils connectés sont alors *protégés* par le VPN si ils se connectent à ce réseau plutôt qu'à celui de la box. Le projet est basé sur `yunohost.org` lui même basé sur Debian, permettant de faciliter l'installation et l'utilisation d'un serveur pour les non-initiés.

Mon cas est particulier pour deux raisons :

1. J'utilise une carte OLIMEX LIME 1, pour laquelle il n'y a pas d'images pour la dernière version du projet, il faut donc réaliser une migration après avoir installé la version précédente.
2. J'utilise une clé 4G Huawei pour me connecter et non un câble ethernet, il faut donc configurer la carte pour qu'elle se connecte sur cette clé au démarrage.

Ces deux particularités m'ont causé pas mal de soucis avant de finalement réussir une installation. Une réinstallation est maintenant l'occasion pour moi de documenter les opérations nécessaires.

5.1.1 Installation, post-installation

J'ai utilisé le programme disponible sur `install.labriqueinter.net` pour générer mon fichier `install.hypercube`. Il faut ensuite utiliser le script fourni `install-sd.sh` pour installer les fichiers sur la carte sd. Comme il manque la dernière version de l'image pour la LIME1 il y a une erreur à l'exécution.

Deux solutions sont alors possibles : installer via une image brique internet Jessie puis faire la migration vers Stretch, ou faire une image brique internet à partir d'une image Yunohost. La première option n'a pas fonctionné pour moi et elle prend énormément de temps avec toutes les mises à jour à faire.

On préférera donc la méthode :

- Télécharger une image Yunohost pour LIME1 : <https://yunohost.org/#/images>
- Créer l'image Brique internet à l'aide du script fourni ici : <https://github.com/labriqueinternet/build.labriqueinter.net/>
- Utiliser le script d'installation sur carte SD avec l'image créée.

Ma commande d'exécution est celle-ci :

```
bash install-sd.sh -y install.hypercube -f internetcube-stretch-3.4.2-lime-stable.img
```

L'option `-s` permet de préciser quel périphérique utiliser si vous savez déjà l'emplacement de celui-ci. Vous avez besoin d'avoir `arp-scan` pour la suite :

```
sudo apt install arp-scan
bash install-sd.sh -l
```

La deuxième commande permet de scanner le réseau pour détecter la brique en cours d'installation. Cela vous donnera l'adresse IP de la brique pour suivre l'avancement de l'installation.

5.1.2 Monter la clé 4G au démarrage comme interface réseau

J'utilise une clé Huawei E3372h, j'ai suivi des tutoriels pour la passer de *hilink* à *stick* mais **je ne suis pas certain que cela soit nécessaire finalement, je vous conseille d'essayer d'abord les commandes suivantes avant d'essayer de faire le changement de mode.**

Je partage cependant les liens qui m'ont permis de réaliser ce changement (cela m'a nécessité l'utilisation de linux mais aussi windows) :

<https://www.0xf8.org/2017/01/flashing-a-huawei-e3372h-4g-lte-stick-from-hilink-to-stick-mode/>
<http://blog.asiantuntijakaveri.fi/2015/07/convert-huawei-e3372h-153-from.html>

https://markus.relix.de/index.php/Set_Huawei_E3372h_from_hilink_to_stick_mode

Lors de mes premiers tests j'ai testé plusieurs méthodes souvent sans résultats, la solution vient finalement de ce post : <https://askubuntu.com/questions/755570/how-to-enable-broadband-connection-with-huawei-e3372h-on-ubuntu-server>
Ces explications permettent pour mon cas d'ajouter deux lignes dans `/etc/rc.local` permettant de démarrer la brique en utilisant la clé 4G comme interface réseau. Ce fichier est celui qui permet de lancer des commandes au démarrage.

rc.local

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
echo -e "AT^NDISDUP=1,1,\"websfr\" \"r\" > /dev/ttyUSB0
dhclient wwan0
exit 0
```

Le **websfr** doit être modifié pour correspondre à l'**APN** de votre opérateur.

À la suite de ma réinstallation éditer ce fichier est tout ce qu'il m'a fallu faire pour que cela fonctionne.

5.2 Utilisation de deux serveurs avec un domaine principal et un sous-domaine

J'utilise deux serveur, un comme brique internet sur clé 4G, et un autre en serveur principal pour les gros services. Ils ont donc deux adresses IP publiques différentes.

Il me fallait donc bien configurer mes enregistrements DNS pour renvoyer mon sous-domaine dédié à la brique internet vers son adresse IP publique, ainsi que pouvoir lui installer un certificat SSL (pour l'https).

Il y a donc deux champs à rajouter aux enregistrements DNS :

- Un enregistrement **A** pour rediriger vers l'ip https://docs.gandi.net/en/domain_names/faq/record_types/a_record.html
- Un enregistrement **CAA** pour la création de certificat SSL https://docs.gandi.net/en/domain_names/faq/record_types/caa_record.html#caa-records

En suivant ces documentations on a donc :

```
brique 1800 IN A votre.ip
brique.votredomaine.com 3600 IN CAA 128 issue "letsencrypt.org"
```

5.2.1 Enlever le bouton Yunohost permanent dans les applications

Certaines applications affiche par défaut en bas à droite un lien avec le logo de yunohost. Pour l'enlever il faut modifier la configuration de **nginx** pour chaque application.

Il suffit de commenter la ligne :

```
include conf.d/yunohost_panel.conf.inc
```

Les fichiers de config sont dans : `/etc/nginx/conf.d/` (cela peut varier).

Pour info le script appelé est le suivant :

```
# Insert YunoHost button + portal overlay
sub_filter </head> '<script type="text/javascript" src="/ynh_portal.js"></script><link type="text/css" rel="
stylesheet" href="/ynh_overlay.css"></link><script type="text/javascript" src="/ynhtheme/custom_portal.js"></
script><link type="text/css" rel="stylesheet" href="/ynhtheme/custom_overlay.css"></link></head>';
sub_filter_once on;
# Apply to other mime types than text/html
sub_filter_types application/xhtml+xml;
```

```
# Prevent YunoHost panel files from being blocked by specific app rules
location ~ (ynh_portal.js|ynh_overlay.css|ynh_userinfo.json) {
}
```

Pour voir les changements il faut recharger nginx

```
sudo systemctl restart nginx
```

6 Sauvegardes

6.1 Android - Utilisation de adb

Installation :

```
sudo apt-get install android-tools-adb android-tools-fastboot
```

Détection :

```
adb devices
```

Sauvegarde :

```
adb backup -apk -shared -all -f backup.ab
```

Restaurer :

```
adb restore backup.ab
```

6.2 GNU-Linux

6.2.1 Le paquet rsync

Le paquet *rsync* permet de sauvegarder ses données par copies. Il copie tous les fichiers sources vers la destination que vous avez choisie. Ensuite, il ne mettra à jour que ceux qui ont été modifiés ou ajoutés. D'où un gain de temps. De plus, rsync peut aussi être utilisé à travers un réseau.

SYNTAXE DE BASE : **rsync [option] /source /destination**

Copier un dossier intégralement

```
rsync -av /source /destination
```

Copier seulement son contenu

```
rsync -av /source/ /destination
```

Transfert via SSH, avec un port spécifique

```
rsync -avzh -e "ssh -p $PORT" --progress /source/ user@domain.net:/destination/
```

Lors d'un téléchargement distant par ssh, en cas de coupure

```
rsync --partial --progress --rsh=ssh [source] [user]@[host]:[destination]
```

L'option **--partial** relance depuis le point d'arrêt, **--progress** affiche la progression et **--rsh=ssh** sécurise le transfert.

Exemple :

```
rsync -r -t -x -v -n --progress -l -s --filter "- /.cache/" --filter "- /.thumbnails/" --filter "+ jeVeuxCeluiCi.txt" /home/utilisateur/ /media/DisqueExterne/repertoire_sauvegarde/ | more
```

- r fait une copie récursive
- S, traite les fichiers à trous efficacement
- x, ne traverse pas les limites du système de fichiers
- n, simule la sauvegarde sans rien faire réellement

-progress affiche la progression
-filter permet de désigner les dossiers que l'on ne veut pas sauvegarder (signe -) ou que l'on veut ajouter à la liste (signe +)
enlever le -n et /more pour que la sauvegarde se fasse une fois satisfait.

Interfaces graphiques disponibles : grsync et luckybackup

DOCUMENTATION : <https://debian-facile.org/doc:reseau:rsync>

6.2.2 Sauvegarde complète de Debian avec un script

À l'aide du paquet présenté ci-dessus il est possible de créer un ou des scripts qui sauvegarderont vos données sur un disque externe ou même sur le réseau. Je décris ici la méthode pour une sauvegarde de la racine (fichier de configuration, paquets, réglages des paquets, etc) et des fichiers personnels.

Cette partie demande de connaître l'organisation des fichiers dans un système GNU/Linux et comment les contrôler dans un terminal.

Cette partie demande en particulier des précautions car elle consiste à modifier ses fichiers avec un contrôle root et cela n'est pas sans risques.

Il faut en particulier adapter les noms des partitions de ce guide à son propre système.

Pour identifier ses disques, tapez (*en root*) :

```
:~# fdisk -l
```

Vérifiez que vous n'avez aucun périphérique usb branché, éteignez et débranchez votre ordinateur, rallumez pour identifier votre disque source. Remettez hors tension, branchez le ou les disques externes, rallumez et identifiez les disques en root. Si vous utilisez un seul disque externe vous devriez trouver :

```
/dev/sda    ### disque source
```

```
/dev/sdb    ### disque des sauvegardes
```

DOCUMENTATION : https://wiki.debian-fr.xyz/Sauvegarder_sa_Debian_en_vue_d%27une_R%C3%A9installation_ou_Reconstruction_-_Racine_et_Home_s%C3%A9par%C3%A9s

7 Logiciels

Lister les paquets installés

```
dpkg-query -l
```

DOCUMENTATION : <https://wiki.debian.org/fr/ListInstalledPackages>

7.1 Graphisme

Logiciels utiles :

1. Dessin vectoriel : **Inkscape**
2. Édition basique d'images : **gthumb**

8 Développement Web

8.1 HTML

8.1.1 Centrer une image

Centrer une image dépendra de son conteneur, je transcris ici la version simple (dans un paragraphe) et ma source détaille différents cas de figure.

Le code dans le corps de la page :

```
<p class="aligncenter">  
    
</p>
```

Ce code appelle donc la classe `aligncenter` définie comme :

```
<style>
.aligncenter {
    text-align: center;
}
</style>
```

SOURCE : <https://html-online.com/articles/image-align-center/>

8.1.2 Faire un fond de page web qui change aléatoirement de couleur

Ce code permet de changer toutes les 5 secondes la couleur du fond de la page. Il utilise 10 couleurs qui peuvent être modifiée, et on peut rajouter plus de couleurs en rajoutant des choix et en changeant à la bonne valeur la gamme de génération d'entier aléatoire.

```
<!-- this script got from www.htmlbestcodes.com-Coded by: Krishna Eydat -->
<html>
<head>
<script type="text/javascript">
    function setbackground()
    {
        window.setTimeout( "setbackground()", 5000); // 5000 milliseconds delay

        var index = Math.round(Math.random() * 9);

        var ColorValue = "FFFFFF"; // default color - white (index = 0)

        if(index == 1)
            ColorValue = "FFCCCC"; //peach
        if(index == 2)
            ColorValue = "CCAFFF"; //violet
        if(index == 3)
            ColorValue = "A6BEFF"; //lt blue
        if(index == 4)
            ColorValue = "99FFFF"; //cyan
        if(index == 5)
            ColorValue = "D5CCBB"; //tan
        if(index == 6)
            ColorValue = "99FF99"; //lt green
        if(index == 7)
            ColorValue = "FFFF99"; //lt yellow
        if(index == 8)
            ColorValue = "FFCC99"; //lt orange
        if(index == 9)
            ColorValue = "CCCCCC"; //lt grey

        document.getElementsByTagName("body")[0].style.backgroundColor = "#" + ColorValue;
    }
</script>
</head>
<body onload="setbackground();">
</body>
</html>
```

SOURCE : https://www.htmlbestcodes.com/Set_background_color_randomly.htm

9 Internet

9.1 Flux RSS

Les flux RSS permettent de suivre les publications de la majorité des sites internet sur une plateforme dédiée, il existe une multitude de services dédiés (Firefox intègre d'ailleurs un lecteur de flux).

Un aspect pratique des flux RSS est que cela fonctionne pour s'abonner à des **Chaînes YouTube** et cela sans compte, certains lecteurs de flux peuvent même afficher la vidéo et donc vous n'avez pas à aller sur YouTube en soi.

S'ABONNER À UNE CHAÎNE YOUTUBE

URL : <https://www.youtube.com/feeds/videos.xml?user=nom de user ou code trouvable dans l'URL sur YouTube>

S'ABONNER À UNE PLAYLIST

URL : https://www.youtube.com/feeds/videos.xml?playlist_id=nom de playlist ou id de l'URL

9.2 Multimédia

9.2.1 youtube-dl

Installation :

```
sudo curl -L https://yt-dl.org/downloads/latest/youtube-dl -o /usr/local/bin/youtube-dl
sudo chmod a+rx /usr/local/bin/youtube-dl
sudo apt-get install ffmpeg
```

Mettre à jour

```
youtube-dl -U
```

Options :

```
youtube-dl [OPTIONS] URL
-i, --ignore-errors
--playlist-start NUMBER
--playlist-end NUMBER
```

Metadata

```
--write-description Write video description to a .description file
--write-info-json Write video metadata to a .info.json file
--write-annotations Write video annotations to a .annotations.xml file
--write-thumbnail Write thumbnail image to disk
--write-sub
--all-subs
```

Audio

```
-x, --extract-audio Convert video files to audio-only files (requires ffmpeg or avconv and ffprobe or avprobe)
--add-metadata Write metadata to the video file
--metadata-from-title "%(artist)s - %(title)s"
```

DOCUMENTATION : <https://github.com/rg3/youtube-dl/blob/master/README.md>

9.2.2 Spotify pour Debian 9

```
sudo apt instal dirmngr
# 1. Add the Spotify repository signing keys to be able to verify downloaded packages
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 931FF8E79F0876134EDDBDCCA87FF9DF48BF1C90

# 2. Add the Spotify repository
echo deb http://repository.spotify.com stable non-free | sudo tee /etc/apt/sources.list.d/spotify.list

# 3. Update list of available packages
sudo apt-get update

# 4. Install Spotify
sudo apt-get install spotify-client
```

9.2.3 Télécharger des playlists Spotify

J'utilise : <https://github.com/SwapnilSoni1999/spotify-dl>

```
sudo npm install -g spotify-dl
```

Il peut être nécessaire d'installer **npm** avec un simple apt install.

10 Vie privée

10.1 Utiliser le VPN Riseup Black

1. Créer un compte black.riseup.net
2. Installer **Bitmask**
3. Ouvrir Bitmask et se connecter

Installation de Bitmask pour Debian Stretch

DOCUMENTATION : <https://dl.bitmask.net/linux/>

Il y a besoin d'autoriser les stretch-backports pour installer le dernier paquet leap-archive-keyring

Ne pas faire cette étape si stretch-backports est déjà activé

```
sudo echo "deb http://deb.debian.org/debian stretch-backports main" > /etc/apt/sources.list.d/stretch-backports.list
sudo apt update && sudo apt install -t stretch-backports leap-archive-keyring
```

```
sudo sh -c 'echo "deb http://deb.leap.se/client release stretch" > /etc/apt/sources.list.d/bitmask.list'
sudo apt update && sudo apt install bitmask
```

Résolution de problème

10.2 Changer son adresse mac

Installer macchanger :

```
sudo apt-get install macchanger
```

On peut ensuite connaître son adresse MAC pour la noter :

```
macchanger eth0
ou
macchanger wlan0
ou
macchanger wlp3s0
```

Créer le fichier de configuration :

```
sudo nano /etc/init/macchanger.conf

# macchanger - set MAC addresses
#
# Set the MAC addresses for the network interfaces.

description    "change mac addresses"

start on starting network-manager

pre-start script
    /usr/bin/macchanger -A wlan0
    /usr/bin/macchanger -A eth0
    /usr/bin/macchanger -A wmaster0
    /usr/bin/macchanger -A pan0
    #/usr/bin/logger wlan0 '/usr/bin/macchanger -s wlan0'
    #/usr/bin/logger eth0 '/usr/bin/macchanger -s eth0'
end script
```

Attention à rajouter le nom de votre interface si elle n'est ni wlan0 ni eth0 (c'était wlp3s0 pour moi).

Créer le fichier `/etc/network/if-post-down.d/random-mac` à partir de :

```
#!/bin/sh
```

```
MACCHANGER=/usr/bin/macchanger

[ "$IFACE" != "lo" ] || exit 0

# Bring down interface (for wireless cards that are up to scan for networks), change MAC address to a random
# vendor address, bring up the interface
/sbin/ifconfig "$IFACE" down
macchanger -A "$IFACE"
```

Il reste à rendre le script exécutable et à redémarrer le network manager :

```
sudo chmod +x /etc/network/if-post-down.d/random-mac
sudo service network-manager restart
```

Sources : <https://riseup.net/fr/security/network-security/mac-address>

11 Bugs divers

11.1 Bugs après une installation

11.1.1 Driver wifi - Debian x bcm43xx

Pour connaître les composants connectés au bus PCI, on utilise la commande `lspci`

Version raccourcie pour les réseaux :

```
~$ lspci -nn | grep Network
03:00.0 Network controller [0280]: Intel Corporation PRO/Wireless 3945ABG Network Connection [8086:4227] (rev 02)
```

Présentation ici d'un cas particulier mais la méthode peut être adaptée à d'autres modèles de drivers.

Ajouter contrib non-free à la ligne du fichier `/etc/apt/sources.list` puis mettre à jour les sources :

```
# Debian 9 "stretch"
deb http://deb.debian.org/debian/ stretch main contrib non-free
apt-get update
```

Installer le bon firmware selon le driver

BCM4306 revision 3, BCM4311, BCM4318, BCM4321 ou BCM4322 :

`firmware-b43-installer`

BCM4312 LP-PHY :

`firmware-b43-installer`

BCM4301 chip ou BCM4306 revision 2 :

`firmware-b43legacy-installer`

SOURCE : <https://wiki.debian.org/bcm43xx>

11.2 Bugs dans le terminal

11.2.1 ./configure say no compiler found

Erreur

```
configure: error: no acceptable C compiler found in $PATH
```

Solution

```
sudo apt-get install build-essential
```

SOURCE : <https://stackoverflow.com/questions/18076157/why-does-configure-say-no-c-compiler-found-when-gcc-is>

12 Divers

12.1 Wallabako - synchronisation wallabag et kobo

Si vous avez une instance Wallabag et une liseuse Kobo, il existe un programme permettant de synchroniser les articles non-lus en les téléchargeant en ePUB sur votre liseuse.

La documentation et les fichiers sont ici : <https://gitlab.com/anarcat/wallabako>

12.2 Contrôler son téléphone Android - scrpy

Ce programme permet à un téléphone Android d'être entièrement contrôlé via l'ordinateur auquel il est branché. L'écran est simplement en miroir sur une fenêtre flottante et permet d'utiliser le clavier et la souris de l'ordinateur.

Il est disponible sur les principales distributions : <https://github.com/Genymobile/scrpy>

12.3 Stylet, PC tablette, etc

Pour installer les drivers du stylet dans Arch Linux (nécessite un redémarrage) :

```
sudo pacman -S xf86-input-wacom
```

Pour activer la reconnaissance automatique de l'orientation et changer celle de l'écran et de la souris en même temps, installer les programmes suivants (disponibles en AUR) :

```
iiio-sensor-proxy-git screenrotator-git
```

Ressources : https://wiki.archlinux.org/index.php/Tablet_PC#Stylus

12.3.1 Activer le clic droit du stylet

D'abord, trouver l'id du stylet avec :

```
xinput --list
```

```
xsetwacom set <id> button 2 3
```

Nécessite d'être relancé à chaque démarrage.

12.4 ASCII art generator

Convertir une image en texte ASCII : <https://github.com/uvipen/ASCII-generator>

12.5 Comment créer un livre électronique au format ePub avec LibreOffice

Tuto : <https://framablog.org/2013/10/11/creer-epub-libreoffice-logiciel-libre/>

12.6 Taper le point median sur Linux

En fonction de votre disposition voici des combinaisons de touches à utiliser :

```
- AltGr + Maj + ;  
- AltGr + Maj + 1  
- AltGr + :  
- AltGr + .  
- Maj + AltGr +
```

Plus d'informations : <https://rebellyon.info/Comment-taper-facilement-le-point-median-16767>

13 Windows

Clé Windows : `sudo cat /sys/firmware/acpi/tables/MSDM`

14 Annexes - Scripts

Attention, ces scripts sont des scripts que j'ai réalisés il y a parfois longtemps et souvent pour une utilisation unique. Il sont donc non-optimisés et potentiellement cassés. Ils ont fonctionné pour moi et fonctionnent probablement encore mais je conseille de les utiliser avec prudence pour éviter toute perte de fichiers.

14.1 Trier des fichiers par année

J'avais un gros dossier en vrac de vieilles photos, que je voulais garder mais en créant des dossiers par année pour les retrouver plus facilement. Il se trouve que l'on ne peut pas trouver dans les métadonnées la date de création mais seulement de dernière modification. J'ai reproduit ici ce script comme exemple de script simple et pour qu'il soit réutilisable.

```
#!/bin/bash
echo "Chemin du dossier a trier :"
read chemin
cd $chemin
ls
#Teste chaque fichier du dossier chemin
for i in *; do
    if [ -d "$i" ]; then
        #-d est l'option pour savoir si un dossier existe, ici on teste si il est utile de l'analyser ou pas
        echo "C'est un dossier"
    else
        annee='stat -c '%y' "$i" | awk -F - '{print $1}''
        #Sort l'annee de derniere modification
        #-c utilise le format precise plutot que celui par default. '%y' date de la derniere modification d'
            etat en format lisible.
        #awk -F - precise que le separateur est le tiret et pas un espace
        #'{print $1}' s'arrete d'afficher au premier separateur
        #" $i " entre guillemet pour eviter les erreurs avec les espaces dans les noms

        if [ -d $annee ]; then
            mv "$i" $annee
        else
            mkdir $annee
            mv "$i" $annee
        fi
    fi
done
```

14.2 Post-Installation automatique

Quand on veut réinstaller son ordinateur on sait généralement déjà de quoi on a besoin/envie. Et la force de GNU/Linux est de pouvoir réaliser de nombreuses étapes de post-installation automatiquement.

Celui-ci permet surtout d'installer les quelques programmes non-installés par défaut mais utiles comme les paquets pour lire les fichiers d'un téléphone android. Il permet aussi d'installer les librairies qui permettent de lire les DVD.

Si les lignes correspondantes sont décommentées, il permet d'installer VirtualBox.

Enfin, il peut être utilisé pour copier les fichiers de configuration que vous avez déjà, sous réserve de quelques modifications/ajouts pour les lignes commençants par `cp -r`.

Il nécessite bien entendu d'être exécuté en root ou avec sudo.

```
#!/bin/sh
#echo "Dossier des fichiers de config"
#read dossier
#echo "nom utilisateur/trice"
#read id
echo "Verifier que sudo est bien installe, sinon le script ne pourra pas fonctionner [0/n]"
read answer
if [ $answer = "0" ]; then
```

```

sudo apt update
#Utiles et dependances
sudo apt install curl ffmpeg accountsservice git tree rsync apt-transport-https numix-gtk-theme numix-icon-
theme
#Android
sudo apt install android-tools-adb android-tools-fastboot gvfs-backends jmtarfs libmtp-common
#Bureautique et multimedia
sudo apt install gthumb inkscape texmaker handbrake vlc
#Utilitaires
sudo apt install keepassx gnome-disk-utility
#Internet
sudo apt install thunderbird transmission pidgin

#cp -r $dossier/.config /home/$id/
#cp -r $dossier/.theme home/$id/
#cp -r $dossier/.mozilla home/$id/

#Install libdvdcss (necessite apt-transport-https)
wget -O - https://download.videolan.org/pub/debian/videolan-apt.asc | sudo apt-key add - && echo "deb https
://download.videolan.org/pub/debian/stable/ /" | sudo tee /etc/apt/sources.list.d/videolan.list
echo "deb-src https://download.videolan.org/pub/debian/stable/ /" | sudo tee /etc/apt/sources.list.d/
videolan.list

#Install youtube-dl
sudo curl -L https://yt-dl.org/downloads/latest/youtube-dl -o /usr/local/bin/youtube-dl
sudo chmod a+rx /usr/local/bin/youtube-dl

#Install VirtualBox
#wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc -O- | sudo apt-key add -
#wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo apt-key add -
#echo "deb https://download.virtualbox.org/virtualbox/debian stretch contrib" |sudo tee -a /etc/apt/sources
.list.d/virtualbox.list

sudo apt update
#sudo apt-get install virtualbox-6.0
sudo apt install libdvdcss2
#echo "URL de telechargement VMware : https://my.vmware.com/fr/web/vmware/free#desktop_end_user_computing/
vmware_workstation_player/12_0"
#echo "Pour installer, utiliser la commande sudo bash VMware-xxxx.bundle"
#Dependances VMware
#sudo apt install libx11-6 libx11-dev x-window-system-core x-window-system xorg-docs libxtst6 psmisc build-
essential
echo "Installation finie"
else
exit
fi

```

14.3 Compression de PDF

Ce script permet de compresser des PDF sans avoir à passer par un utilitaire tiers.

Note : je n'ai plus aucun souvenirs de comment je l'ai créé, je l'ai juste testé avant de le reproduire ici et il fonctionne.

```

#!/bin/bash
echo "Nom et chemin du fichier a compresser, sans l'extension .pdf"
read chemin
echo "Nom et chemin du fichier de sortie, sans l'extension .pdf"
read sortie

gs -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -dPDFSETTINGS=/screen -dNOPAUSE -dQUIET -dBATCH -sOutputFile=$sortie
.pdf $chemin.pdf

```

14.4 Écrire soit-même des fortunes en semi-automatique

Les **fortunes** sont les fichiers permettant d'afficher des citations aléatoires dans un terminal ou dans certains widgets. Il en existe de nombreuses versions mais on peut aussi les créer nous-même, comme c'est un peu fastidieux, j'avais fait un script pour l'occasion. **Attention il nécessite d'être exécuté en root car il écrit des fichiers dans des répertoires protégés en écriture.**

```
#!/bin/bash
#Ce script a pour but de faciliter la mise a jour des fichiers fortune.
echo -e "\033[1;45;41m Ce script necessite les droits d'administrateur pour fonctionner \033[0m"
echo -e "\033[1;44m Bienvenue dans ce script de mise a jour de fortune \033[0m"
#Creation d'un fichier texte avec la liste des fichiers fortune (1 par ligne)
cd /usr/share/games/fortunes
ls -lI "*.dat" -lI "*.u8"
echo -e "\033[1;44mTaper le nom du fichier a modifier, ou " "\033[1;42mNouveau\033[0m""\033[1;44m pour un nouveau
fichier\033[0m"
#cree la variable fichier en lui assignant la valeur tapee
read num
if [ $num = Nouveau ]; then
    echo -e "\033[1;42mEntrer un nom pour le nouveau fichier\033[0m"
    read fichier
else
    for i in *
    do
        if [ $num = $i ]; then
            fichier=$i
        fi
    done
fi

if [ -z "$fichier" ]; then
    echo "Le fichier n'existe pas"
else
    echo -e "\033[1;45;41mModification avec nano ou avec le script ?\033[0m"
    echo -e "\033[1;45;31mNano : 1\033[0m"
    echo -e "\033[1;45;34mScript : 2\033[0m"
    read choix
    if [ $choix = 1 ]; then
        nano /usr/share/games/fortunes/$fichier
        echo -e "\033[1;45;41mFin de la modification\033[0m"
    elif [ $choix = 2 ]; then
        #initialise la variable rep a 0 pour la boucle while
        rep=0
        while [ $rep = 0 ]
        do
            #boucle while : tant que la variable rep reste a "0" on peut continuer a rajouter des lignes.
            #on demande donc a chaque insertion si l'on continue ou pas, si une autre valeur que 0 est renseignee, le
            script sort de la boucle while
            echo "Entrer la citation a ajouter et appuyer sur [ENTREE]"
            #assigne la variable ligne a l'insertion au clavier
            read ligne
            #ajoute apres la derniere ligne du fichier la variable ligne
            echo $ligne >> /usr/share/games/fortunes/$fichier
            #rajoute une nouvelle ligne avec un %
            echo % >> /usr/share/games/fortunes/$fichier
            echo "Continuer ? [o/n]"
            #enregistre la reponse pour continuer dans rep
            read rep
        done
    fi
    #mets a jour les autres fichiers fortune
    strfile /usr/share/games/fortunes/$fichier /usr/share/games/fortunes/$fichier.u8
    strfile /usr/share/games/fortunes/$fichier /usr/share/games/fortunes/$fichier.dat
fi
```

```
exit 0;
```

14.5 Convertir une archive Youtube en dossier de favoris

Ce projet répondait à un besoin spécifique de récupérer mes playlists Youtube avant de supprimer mon compte. Youtube permet de récupérer une archive de toutes ses données, mais les donne sous forme de fichier `.json` qui est complètement inutilisable tel quel.

Mon idée était de récupérer toutes les données et d'en faire un `.html` exploitable par firefox comme une sauvegarde de favoris et donc en faire un dossier avec par exemple toutes mes vidéos "À voir" ou mes playlists.

Ce script utilise des fichiers annexes `header.txt` et `prog.sed` pour fonctionner et écrit plusieurs fichiers dans le dossier `tmp`. Il faut les supprimer à chaque nouvelle utilisation. Pour cela vous pouvez utiliser le script `rm.sh`.

```
#!/bin/bash
#Conversion .json to id and title
echo -e "name of .json :"
read path
# .json file gives the video id under "videoId" : "example",
# .json file gives the video title under "title" : "example"
#For each entry videoId is given twice
#Remove every lines without "videoId" : " and writes the results in a temp file
sed 's/"/"videoId" : "/!d' $path >> /tmp/tmpIds
#Same thing with titles
sed 's/"/"title" : "/!d' $path >> /tmp/Titles
#There is two lines for each Id but one ends with ", so it removes all those lines
sed 's/"/",/d' /tmp/tmpIds >> /tmp/Ids
#We replace "videoId" by the HTML
sed 's/      "videoId" : "/<DT><A HREF="https://www.youtube.com/watch?v=g' /tmp/Ids >> /tmp/Ids2

while read line
do
    echo "${line/%\"/</A>}" >> /tmp/Titles2
done < /tmp/Titles
sed 's/"title" : "/>g' /tmp/Titles2 >> /tmp/Titles3

paste -d "" /tmp/Ids2 /tmp/Titles3 >> links.html

sed -f prog.sed links.html >> linksdone.html

cat header.txt linksdone.html >autobookmarks.html

echo -e \</DL>><p> >> autobookmarks.html
echo -e \</DL>><p> >> autobookmarks.html
cp autobookmarks $path.html
```

14.5.1 prog.sed

```
i\ \<DL>><p>
a\ \</DL>><p>
```

14.5.2 header.txt

```
<!DOCTYPE NETSCAPE-Bookmark-file-1>
<!-- This is an automatically generated file.
      It will be read and overwritten.
      DO NOT EDIT! -->
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=UTF-8">
<TITLE>Bookmarks</TITLE>
<H1>Bookmarks</H1>
```


14.5.3 rm.sh

```
#!/bin/bash
rm /tmp/Titles
rm /tmp/Titles2
rm /tmp/Titles3
rm /tmp/Ids
rm /tmp/Ids2
rm /tmp/tmpIds
rm autobookmarks.html
rm links.html
rm linksdone.html
rm linksdone2.html
rm linkstmp
```