Linux

- Comment installer Linux sur un Macintosh et dual boot avec macOS FR
- Comment installer Linux sur un Macintosh et dual boot avec macOS EN

- Raspbian ou Linux pour débutant FR
- Différences entre GNU/Linux et Windows
- Introduction à Linux

Ressources Linux

Les liens:

Français

- - documentation Ubuntu francophone
- - débutant - autre tutoriel
- - avancé - administration - debian
- - avancé - flux d'information et bidouille sur du Libre et autre
- - wiki fr d'ArchLinux, distribution en rolling release
- Le site officiel du projet Debian
- Le Wiki officiel du projet Debian
- Le site de l'association debian-facile - distribution "DFLinux" en téléchargement.
- Site de nouvelles et de tutoriels autours des logiciels libres (pratico-pratique !)
- cours linux simplifié

Anglais

- - cours - install - liste de commande unix
- - liste de commandes
- - english wiki of ArchLinux (une référence sous Linux, souvent très complet)

Livres

- Le cahier de l'administrateur debian Debian 10
- PDF - français et autre langue dispo Projet Manuel d'Ubuntu
- 700 pages ancien 2006 traduction de "Running Linux"Le Système Linux o'reilly
- Simple Comme Ubuntu- Framasoft - 2010 un peu ancien
- - Framasoft - 2010 - 261 pages, format PDF Unix et la ligne de commande

Commandes Linux

- Les commandes les plus utiles
- Exécuter des programmes en arrière-plan
- Linux - commandes fondamentales

- commande shell bash

Verr Num au demmarrage

Verr Num sous linux

Changer de disposition clavier avec son terminal

```
   setxkbmap fr
```

Commandes de commande

man

```
  <nom de la commande> : permet de consulter le manuel associé à la commande passée en paramètre (man = manual). C’est à mon avis la commande la plus utile qui soit puisque grâce à elle, vous pourrez utiliser n’importe quelle commande facilement 
```

whereis

```
  <nom de la commande> : permet de connaître quel script est exécuté lors de l’appel à la commande.
```

Système

top :

- permet de lister les processus du système.

htop :

- Permet de verifier les informations systemes : processus , memoires, Cpu ..., (plus facile à lire que top , mais il faut l'installer)

kill :

- permet de terminer un processus.

df :

- permet de connaître l’espace disponible sur les disques.

fdisk

- fdisk permet de manipuler les tables de partitions. Il permet de créer, de supprimer, de lister les partitions sur un disque dur.
    - Lister les partitions : *sudo fdisk -l*
    - Modifier /dev/sda : *sudo fdisk /dev/sda* ** ATTENTION commande pouvant etre dangereuse pour le disque **

mkfs

- permet de formater une partition
    - *sudo mkfs -t ext4 /dev/sdc* **formate la partition /dev/sdc/ en ext4**

uname :

- permet d'afficher des informations importantes du système (version du système…).

su :

- permet de changer d'utilisateur courant (su = switch user).
- Fichiers & repertoires

pwd :

- permet de savoir dans quel répertoire vous vous trouvez actuellement (pwd = print working directory).

cd <nom ou chemin du répertoire> : permet de changer de répertoire courant (cd = change directory). Cette commande permet de réaliser les mêmes actions que lorsque vous vous baladez avec votre souris dans l'arborescence de votre système.

mkdir :

- mkdir <nom du répertoire> : permet de créer un nouveau répertoire dans le répertoire courant (mkdir = make directory). Pour supprimer un répertoire, il suffit d'utiliser la commande rmdir <nom du répertoire>.

ls :

- permet de lister les fichiers et répertoires contenus dans le répertoire courant.

touch :

- touch <nom du fichier> permet de créer un nouveau fichier vide. A la base, cette commande

permet de changer la date de modification du fichier cible mais losqu'il n'existe pas, elle le créé. Pour supprimer un fichier, il suffit de saisir rm <nom du fichier>.

chmod et chown :

- permettent de changer les permissions sur les fichiers et les répertoires.

Exemples :

- chmod -R 777 /var/www/html/data : permet de changer les autorisations du repertoire /var/www/html/**data** et des sous repertoires
- chown -R www-data /var/www/html/data : permet de changer l'utilisateur du repertoire /var/www/html/**data** et des sous repertoires
- chown -R www-data:www-data /var/www/html/data : permet de changer l'utilisateur et le groupe du repertoire /var/www/html/**data** et des sous repertoires

cp :

- cp <fichier source> <répertoire de destination> : permet de copier le fichier source et de le coller dans le répertoire de destination (cp = copy). Il est également possible de réaliser un « couper/coller » à l'aide de la commande mv <fichier source> <répertoire de destination> (mv = move).

cat :

- cat <nom du fichier> : permet de lire le contenu d'un fichier (cat = catenate, littéralement concaténer).

Exemple :

- cat /etc/passwd | awk -F: '{print $1}' : permet de lister les utilisateurs sous Debian

grep :

- grep <mots à trouver> <fichiers cibles> : permet de rechercher une chaîne de caractères dans un ou plusieurs fichiers. Cette commande est extrêmement utile et puissante et je ne peux que vous conseiller de bien lire sa documentation (man grep).

find :

- permet de rechercher des fichiers ou des répertoires selon leur nom.

Téléchargements

wget :

- wget <adresse de la ressource> : permet de télécharger la ressource hébergée à l'adresse définie.

curl :

- permet de télécharger mais également de transmettre un fichier vers un serveur distant.

ftp :

- permet de se connecter à l'aide du protocole FTP à un serveur distant. Vous pourrez ensuite télécharger ou transmettre plusieurs fichiers.

Réseau

ping :

- ping <ip ou nom de domaine> : permet de vérifier si la machine cible est bien accessible par le réseau.

ifconfig :

- permet d'afficher des informations concernant les interfaces réseau. On l'utilise fréquemment pour trouver son IP.

nmap :

- programme à installer, permet de scanner les ip et les ports ouverts des PC sur le même réseau
- exemple :map -sP -PT -PI -T 3 192.168.2.*

Enfin, notez que lorsque vous installez un nouveau logiciel sur votre machine, de nouvelles commandes seront disponibles. Par exemple, après avoir installé le logiciel de traitement de texte Emacs, la commande « emacs » sera disponible en ligne de commande pour lancer le programme.

speedtest

- permet de tester sa connexion internet en ligne de commande

installation de speedtest ==⇒ sudo apt-get install speedtest

- utlisation ==⇒ » speedtest

iftop

- connaitre la vitesse de la carte reseaux

installation de iftop ==⇒ sudo apt-get install iftop

- utilisation ==⇒ iftop

rsync

- rsync (pour remote synchronization ou synchronisation à distance), est un logiciel 1) de synchronisation de fichiers. Il est fréquemment utilisé pour mettre en place des systèmes de sauvegarde distante.rsync travaille de manière unidirectionnelle c'est-à-dire qu'il synchronise, copie ou actualise les données d'une source (locale ou distante) vers une destination (locale ou distante) en ne transférant que les octets des fichiers qui ont été modifiés. La notion d'unidirectionnalité semble souvent mal comprise : elle signifie qu'en une commande, la synchronisation ne peut se faire que dans un sens. Rien n'empêche ensuite de relancer la commande une seconde fois dans l'autre sens !

Voici un exemple d'une commande, utilisant le protocole SSH, qui copie à l'identique le dossier <source> vers le dossier <destination>.

Copie du dossier source vers le serveur:

```
  rsync -e ssh -avz --delete-after /home/source
user@ip_du_serveur:/dossier/destination/
```

où :

1. -delete-after : à la fin du transfert, supprime les fichiers dans le dossier de destination ne se trouvant pas dans le dossier source.
2. z : compresse les fichiers (Limite la bande passante mais augmente l'utilisation processeur et le temps de transfert : inutile en réseau locale ou avec très bon débit)
3. v : verbeux
4. e ssh : utilise le protocole SSH

Si chemin(s) avec des espaces, on peut écrire les chemins entre guillemet en échappant les espace :

```
  rsync -e ssh -avz --delete-after "/home/source\ avec\ espace/"
user@ip_du_serveur:"/dossier/destination\ avec\ espace/"
```

Avec l'option -n la commande liste ce qu'elle va faire sans l'exécuter:

```
  rsync -e ssh -avzn --delete-after /home/mondossier_source
user@ip_du_serveur:/dossier/destination/
```

blkid

commande : **sudo blkid**

Qu'est-ce que l'UUID ? :

Il s'agit d'une suite plus ou moins longue de caractères alpha-numériques qui permet d'identifier de

façon absolument sûre chaque périphérique de stockage et partition.

Le chiffre de l'UUID est calculé automatiquement au moyen d'un algorithme intégrant notamment certaines données de l'ordinateur hôte, au moment de la création ou du formatage de la partition ou de la table des partitions. Ce mode de calcul ne présente aucun risque de sécurité crédible.

Un UUID est de la forme : 1124d9e8-6266-4bcf-8035-37a02ba75c69.

Il est à noter que l'UUID d'une partition est stockée dans le système de fichier. Une partition vierge (non formatée) ne peut donc avoir d'UUID et une partition clonée a le même UUID que l'originale. On peut ajouter que ceci est vrai même avec un partitionnement gpt. Bien que GPT définisse un UUID qui est stocké dans la table de partition, ce n'est pas ce dernier qui est utilisé par linux pour identifier les partitions.

mount et fstab

Le montage automatique ou manuel des systèmes de fichiers

éditer le fichier :** sudo nano /etc/fstab**

fstab

Le fichier /etc/fstab liste les partitions qui seront montées automatiquement au démarrage ou à la connexion du périphérique, avec toujours les mêmes options. Il est composé de plusieurs lignes décrivant chacune les conditions de montage de chaque partition / système de fichier. Le fichier fstab est créé lors de l'installation, avec le montage de la partition principale, et de la swap. Mais on peut y ajouter manuellement des partitions supplémentaires, qui seront alors automatiquement lues par le système au démarrage.

Variables d'environnement

Variables du prompt PS1

Le prompt Unix / Linux est défini par la variable PS1, généralement dans le .bashrc. Les variables suivantes peuvent être utilisées :

```
  \a an ASCII bell character (07)
  \d the date in "Weekday Month Date" format (e.g., "Tue May 26")
  \D{format} - the format is passed to strftime(3) and the result is
inserted into the prompt string; an empty format results in a locale-
specific time representation. The braces are required
  \e an ASCII escape character (033)
  \h the hostname up to the first part
  \H the hostname
  \j the number of jobs currently managed by the shell
  \l the basename of the shell's terminal device name
  \n newline
  \r carriage return
```

```
  \s the name of the shell, the basename of $0 (the portion following the
final slash)
  \t the current time in 24-hour HH:MM:SS format
  \T the current time in 12-hour HH:MM:SS format
  \@ the current time in 12-hour am/pm format
  \A the current time in 24-hour HH:MM format
  \u the username of the current user
  \v the version of bash (e.g., 2.00)
  \V the release of bash, version + patch level (e.g., 2.00.0)
  \w the current working directory, with $HOME abbreviated with a tilde
  \W the basename of the current working directory, with $HOME abbreviated
with a tilde
  \! the history number of this command
  \# the command number of this command
  \$ if the effective UID is 0, a #, otherwise a $
  \nnn the character corresponding to the octal number nnn \\ a backslash
  \[ begin a sequence of non-printing characters, which could be used to
embed a terminal control sequence into the prompt
  \] end a sequence of non-printing character
```

On utilise généralement les combinaisons :

export PS1="\u@\h \w> "

Exemple :

jdoe@laptop-jdoe /var/www/jdoe/sites/default/themes/vincentl >

ou, plus concis

export PS1="\u@\h \W> "

Exemple :

jdoe@laptop-jdoe vincentl >

From:
https://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
https://chanterie37.fr/fablab37110/doku.php?id=start:rasberry:linux&rev=1723111726

Last update: **2024/08/08 12:08**