**Les commandes LINUX**

**Des raccourcis clavier utiles**

Ctrl-A 🡺 vous amène au début de la ligne courante

Ctrl-E 🡺 vous amène à la fin de la ligne courante

Ctrl-K 🡺 efface toute la ligne depuis la position du curseur jusqu'à la fin de la ligne

Ctrl-L 🡺 nettoie l'écran

Ctrl-D 🡺 sur une ligne vide fermera la session actuelle

Ctrl-C 🡺 va interrompre la commande en cours d'exécution, sauf si vous étiez en train d'éditer une ligne. Dans ce cas, ce sera l'édition en cours qui sera interrompue et vous serez ramené à l'invite.

Ctrl-Z 🡺 arrête une tâche de façon temporaire, elle est suspendue

Ctrl-S 🡺 sert à suspendre le flux de caractères sur un Terminal

Ctrl-Q 🡺 sert à restaurer le flux de caractères sur un Terminal.

**Chemins relatifs et chemins absolus**

Chemin **absolu**: chemin complet à partir de la racine ( **/** ).

Chemin **relatif**: chemin à partir de l'endroit où l'on se trouve lorsqu'on tape une commande. On fera alors appel au répertoire parent ( **..** ), par exemple, si on a à revenir en arrière.

Exemples

On a un fichier nommé cv.txt sous le répertoire etc, lui-même faisant partie du répertoire racine ( / ).

Son **chemin absolu** sera /etc/cv.txt

De même, nous avons deux fichiers toto.txt et test.txt, situé dans le répertoire tmp, faisant lui-même partie du répertoire racine ( / ).

Leurs **chemins absolus** respectifs seront alors /tmp/toto.txt et /tmp/test.txt

Supposons maintenant que nous sommes actuellement dans le répertoire /etc. On peut accéder au répertoire /tmp en utilisant le chemin absolu ou le chemin relatif.

**chemin absolu** : cd /tmp

**chemin relatif** : cd ../tmp

Les **chemins relatifs** sont souvent de bons raccourcis lors de la copie ou des déplacements de fichier dans des répertoires au même niveau d'une même branche, lorsque effectué **en ligne de commande**.

Cependant, **pour l'utilisation dans les batch files**, il est recommandé d'utiliser les **chemins absolus**, afin d'éviter d'éventuels problèmes liés au déplacement de l'exécutable du "batch file".

Faire partir un dossier

./execute-me**Opérateurs et filtres**

**\**

Opérateur qui découpe une ligne de commande sur plusieurs lignes, pour éviter de dépasser les 80 caractères d'une ligne.

La commande "cd /tmp" peut s'écrire sur deux lignes si on utilise l'opérateur \

cd \

/tmp

**Enchaînement inconditionnel des commandes**

**;**

; permet d'écrire une séquence de plusieurs commandes sur une même ligne.

Toutes les commandes sont exécutées même si l'une d'entre elle provoque une erreur.

**clear; ls –la**

**Enchaînement conditionnel des commandes**

Les séparateurs && (AND) et || (OR) sur la ligne de commande sont des séparateurs qui jouent les rôles d'opérateurs conditionnels, en ce sens que la 2ème commande sera exécutée en fonction du code de retour de la 1ère commande.

Dans **commande1 && commande2**, commande2 ne sera exécutée que si le code de retour de commande1 est 0 (exécution correcte).

**ls -l /toto && rm -rf /toto**

Dans **commande1 || commande2**, commande2 ne sera exécutée que si le code de retour de commande1 est différent de 0 (exécution erronée).

**ls -l /toto || mkdir /toto**

**Redirections des entrées-sorties**

**<**

Sert à rediriger les entrées

**more < /etc/X11/xorg.conf**

**>**

Sert à rediriger les sorties en écrasant le contenu actuel du fichier

**ls -l /root > /tmp/info.txt**

La commande ls n'affichera rien à l'écran, mais redirigera plutôt son résultat dans le fichier info.txt sous le répertoire /tmp

**>>**

Sert à rediriger les sorties à la fin d'un fichier existant, sans l'écraser

**grep -i "^r" /etc/passwd >> /tmp/info.txt**

**grep** grep username /etc/passwd

Trouvez les villes dont le code postal se termine par 123. Quelle est la 43ième ville de ce résultat?

grep "^...123" sample.csv | head -43

permet de trouver l’id usager c est le troisieme

**ON PEUT COMBINER LES SYMBOLES DE REDIRECTION**

**cat -v < /root/fichierdanger > /root/fichierdanger.lisible**

Permet de créer une version imprimable d'un fichier inconnu, afin de vérifier de quoi il s'agit, sans danger d'exécution.

On peut penser à un texte formaté que pour une raison inconnue, il nous est impossible d'ouvrir.

**On peut transformer ce fichier en texte brut en redirigeant le fichier vers la commande cat -v, dont le résultat serait lui-même redirigé dans un fichier.**

**|**

**pipeline** est un opérateur qui se sert du résultat de la commande précédente pour effectuer celle qui suit.

**ls /dev | grep sda**

**du /home | sort -nr**

**Les métacaractères**

Il est relativement rare de connaître parfaitement la syntaxe d'un nom de fichier particulier. De plus, il est souvent utile de chercher des groupes de fichiers nommés selon certaines caractéristiques qui se ressemblent. Nous pouvons avoir besoin de connaître les fichiers d'un répertoire qui commence par une lettre ou un groupe de lettre particulier, comme ceux commençant par "a" par exemple. L'existence des métacaractères est justifiée par des besoins semblables.

Il existe deux métacaractères : \* et ?

Ainsi que deux métacaractère évolué : [ ] et ^

\* remplace plusieurs caractères

Placé à la fin d'une expression, il signifie "*tout ce qui commence par* [expression], *peu importe la suite*". Ainsi, tes\* signifie " tout ce qui commence par tes, peu importe la suite ". C'est-à-dire que tes\* signifie aussi bien tes1, tes2, test, tessain, testeurs, que testeur26.

Placé au début d'une expression, il signifie "*tout ce qui finit par* [expression], *peu importe le début* ". Ainsi, \*s signifie "tout ce qui finit par s, peu importe le début".

Bien entendu, utilisé tout seul, \* signifie " *n'importe quoi* " et peut être très utile.

Il peut aussi s'utiliser au milieu d'une expression.

? remplace un seul caractère

Placé à la fin d'une expression, il signifie "tout ce qui commence par [expression] peut importe le caractère de fin". Ainsi, tes? signifie autant test que tes1, tes2, tes3, tess, ou teso. Mais il ne signifie pas tests, qui a deux caractères de plus (t et s ) que l'expression.

pour afficher tous les fichiers du répertoire courrant commençant par a.

**ls -l a\***

pour afficher tous les fichiers dont le nom commence par test, suivi d'un seul caractère (quelconque)

**ls -l test?**

[ ] regroupe un ensemble de caractères explicites ou d'intervalles à considérer

Supposons que nous voulons lister tous les fichiers commençant par a, b, c, d, e.

Avec ce que nous savons, nous pourrions écrire

**ls -l a\* b\* c\* d\* e\***

Ce qui afficherait le résultat attendu.

Mais nous pourrions aussi écrire de façon plus concise:

**ls -l [abcde]\***

Ce qui fournit le même résultat, mais avec une syntaxe plus légère. Cette dernière façon indique de chercher tout les fichiers commençant par une des lettres explicites situées entre les crochets.

Nous pourrions de plus raffiner la syntaxe ainsi:

**ls -l [a-e]\***

car les crochets peuvent en effet interpréter les intervalles.

^ désigne un ou plusieurs caractères à exclure

Nous pourrions avoir besoin de tous les fichiers qui ne commence ni par h ni par j.

**ls -l [^hj]\***

**&**

**&** se place à la fin d'une commande pour permettre d'exécuter cette commande en "tâche de fond" et d'en exécuter une autre. Donc, le prompt revient.

pour que la commande xclock s'exécute en tâche de fond

**xclock &**

**alias**

est utilisé pour définir une longue ligne de commande par une plus courte

alias affiche la liste des alias existants

alias name=value création d'un alias

Si vous voulez exécuter la commande ls sans tenir compte de son alias, vous devez exécuter la commande /bin/ls

On peut ajouter un alias dans le fichier **/root/.bashrc**

Il est conseillé d'ajouter les alias et les fonctions "shell" dans le fichier **.bashrc** et non pas dans le fichier **.bash\_profile**.

**alias bye='shutdown -h 0'**

**alias today='date +"%A, %B %-d, %Y"'**

Il existe des alias prédéfinis dans Linux comme rm et md.

**rm='rm -i'**

**md='mkdir'**

**bg - fg - jobs**

Si on a omis &, et qu'on veut mettre un processus en "tâche de fond" 🡺 Ctrl-z et exécuter la commande "bg" (background).

* xclock
* Ctrl-z
* bg

note: la commande "**jobs**" permet d'afficher la liste des processus qui s'exécutent en "tâche de fond".

Pour remettre un job en particulier au premier plan, vous pourrez alors taper fg <n> où <n> désigne le numéro de job, par exemple

* fg 2
* sleep 15
* Ctrl-z
* bg
* fg

Le curseur va revenir à la fin de l'exécution de la commande "sleep 15".

**cal**

est utilisé pour afficher un calendrier

cal MM AAAA (MM et AAAA sont facultatifs)

pour afficher le calendrier du mois courant

**cal**

pour afficher le calendrier de l'année courante

**cal -y**

pour afficher le calendrier de l'année 2006

**cal 2006**

pour afficher le calendrier du mois de mai 2007

**cal 5 2007**

pour afficher le calendrier du mois de septembre de l'année 1752

**cal 9 1752**

**cat**

est utilisé pour afficher le contenu d'un fichier

Le nom cat vient de con**cat**éner.

pour afficher le contenu du fichier "/boot/grub/grub.conf"

**cat /boot/grub/grub.conf**

pour afficher le contenu du fichier "/boot/grub/grub.conf" et numéroter les lignes

**cat -n /boot/grub/grub.conf**

sudo cat /etc/shadow

affiche le mot de passe de l’usager a la premiere ligne

si \* pas de mot de passe

cat /etc/passwd est ce que l’usager a un home

si la 6 ieme colone est /usr/sbin/nologin

Si la sixième colonne de la ligne de l'utilisateur dans le fichier /etc/passwd est /usr/sbin/nologin, cela signifie que cet utilisateur ne peut pas se connecter au système via une connexion interactive en utilisant un shell tel que bash ou sh.

En d'autres termes, l'utilisateur peut être un compte système utilisé pour exécuter des tâches de fond ou des processus spécifiques, mais il n'est pas destiné à être utilisé pour une connexion interactive.

Lorsque vous essayez de vous connecter à un compte d'utilisateur qui a /usr/sbin/nologin comme shell, vous recevrez un message d'erreur indiquant que la connexion est refusée.**cd (change directory)**

est utilisé pour changer de répertoire

Pour accéder à un répertoire ayant qui a des espaces, on place le nom entre double-guillemet.

pour se déplacer directement dans le répertoire de l'utilisateur actif

**cd ~user\_name**

**cd ~**

Lorsque nous sommes connectés en terminal, quelle est la commande la plus simple pour se déplacer dans le répertoire "~"?

**cd**

pour se déplacer en montant d'un niveau

**cd ..**

pour se déplacer à la racine du disque

**cd /**

**clear**

est utilisé pour effacer l'écran

Le raccourci "**Ctrl-L**" permet également d'effacer l'écran.

**cp (copy)**

est utilisé pour copier un fichier et garder sa source intacte

Si le répertoire de travail est "/etc" et le fichier "cv.txt" est dans "/etc"

source absolue, destination absolue

**cp /etc/cv.txt /tmp/cv.txt**

source relative, destination absolue

**cp cv.txt /tmp/cv.txt**

source relative, destination relative

**cp cv.txt ../tmp/cv.txt**

Pour copier des répertoires on utilise l'option -R

cp -R [source] [destination]

cp -R /tmp /home/temporaire

**Si le répertoire "/home/temporaire" n'existe pas il sera créé.**

**Si le répertoire "/home/temporaire" existe on va avoir un répertoire "tmp" sous "/home/temporaire" donc "/home/temporaire/tmp".**

**cut**

est utilisé pour sélectionner une partie seulement des lignes envoyées à l'entrée standard

pour afficher le premier champ du fichier "/etc/passwd" en se basant sur le délimiteur :

**cat /etc/passwd | cut -d: -f1**

pour afficher les dix premiers caractèrs du fichier "/etc/passwd"

**cat /etc/passwd | cut -c1-10**

**date**

est utilisé pour afficher ou modifier la date du système

pour afficher la date et l'heure du système

**date**

pour modifier la date et l'heure du système pour le 23 mai, 14h30.

**date 05231430**

pour modifier la date et l'heure du système pour: 03 octobre - 11h25 - année 2006.

***date 100311252006***

pour avancer l'heure de 2 minutes

**date --set='+2 minutes'**

pour avancer la date de une journée

**date --set='+1 day'**

pour afficher la date avec un texte

**date '+Bonjour, nous sommes le %D'**

Paramètres utiles

%H Renvoie l'heure actuelle chiffrée seulement (s'il est 3h15m20s : **3**)

%M Renvoie les minutes actuelles chiffrées seulement (s'il est 3h15m20s : **15**)

%S Renvoie les secondes actuelles chiffrées seulement (s'il est 3h15m20s : **20**)

%T Renvoie l'heure actuelle sous la forme HH:MM:SS (03:15:20)

%d Renvoie le jour actuel du mois chiffré seulement (Le 4 septembre : **4**)

%B Renvoie le mois actuel lettré seulement (janvier, février...)

%Y Renvoie l'année actuelle chiffrée seulement (**2006**)

%D Renvoie la date actuelle sous la forme MM/JJ/AA (**06/06/06**)

%A Renvoie le jour actuel de la semaine lettré seulement (lundi, mardi...)

%w Renvoie quel nième jour de la semaine nous sommes (**3** [eme])

%j Renvoie quel nième jour de l'année nous sommes (**247** [eme])

%W Renvoie quel nième semaine de l'année nous sommes (**35** [eme])

**commande pour changer l'heure sur l'ordinateur**

* date -s "11:15"

**dd**

est utilisé pour copier des fichiers en spécifiant la taille des blocs

pour créer un fichier image "disque\_mbr.img" qui représente le contenu du MBR d'un disque dur

**dd if=/dev/hda of=/dev/fd0/disque\_mbr.img bs=512 count=1**

pour restaurer une copie du MBR de votre disque dur à partir d'un fichier sur une disquette

**dd if=/dev/fd0/disque\_mbr.img of=/dev/hda bs=512**

pour créer un fichier image "disquette.img" qui représente le contenu d'une disquette

**dd if=/dev/fd0 of=/tmp/disquette.img bs=512**

pour créer une disquette à partir d'un fichier image "floppy.img"

**dd if=/images/floppy.img of=/dev/fd0**

pour créer un fichier ISO à partir d'un CD-ROM

**dd if=/dev/cdrom of=/tmp/mdk1.iso bs=1024**

pour créer un fichier de 1 KO qui contient que des 0

**dd if=/dev/zero of=/tmp/test.txt bs=512 count=2**

**df**

est utilisé pour afficher les informations sur la quantité de l'espace totale, la quantité d'espace occupé, la quantité d'espace libre et le pourcentage de l'espace utilisé des systèmes de fichiers (les périphériques de stockages montés, ou volumes montés).

**dos2unix**

est utilisé pour convertir un fichier DOS en fichier UNIX

**dos2unix /tmp/info.txt**

**du**

est utilisé pour afficher les statistiques d'utilisation du disque en KO par tous les répertoires qu'elle peut lire, à partir d'un emplacement.

**du /home**

**du -s /home**

**du -sk /home**

**e2fsck**

est utilisé pour vérifier un système de fichiers Linux ext2 et ext3

L’option **-p** demande à **e2fsck** d’effectuer toutes les réparations nécessaires sans rien demander tandis que l’option **-y** suppose que vous répondez oui à toutes les questions.

**e2fsck -py /dev/sda1**

**fdformat**

est utilisé pour formater un périphérique

**fdformat /dev/floppy**

**file**

est utilisé pour indiquer le type de fichier

**file /boot/grub/grub.conf**

**file /boot/grub/menu.lst**

**file /boot/vmlinuz**

**find**

est utilisé pour trouver des fichiers sur le disque

find [chemin] [expression]

pour recherche tous les fichiers à partir de la racine du disque dont le nom commence par "xorg"

**find / -iname xorg\***

pour rechercher tous les fichiers à partir de la racine du disque dont le nom possède "clo" dans sa chaîne de caractères.

**On doit utiliser des apostrophes (' 🡺 alt-39) avant et après chaque \*.**

**find / -iname '\*clo\*'**

find / -iname shadow

pour rechercher tous les fichiers de plus de 1000000 octets à partir de la racine du disque.

(c: caractère, et 1 caractère = 1 octet)

**find / -size +1000000c**

pour rechercher tous les fichiers dont la taille est supérieure à 1000000 octets et dont le nom commence par "Pack" dans le répertoire /tmp

**find /tmp -size +1000000c -and -name Pack\***

Plusieurs options utiles

-type pour spécifier le type de fichier

-type d (un répertoire)

-type f (un fichier)

-type l (un lien)

-size pour spécifier un critère de taille

-size 1024c (exactement 1024 octets)

-size -1024c (moins de 1024 octets)

-size +1024c (plus de 1024 octets)

-**mmin n** pour spécifier un critère de temps en minutes sur la dernière modification

exemple: -mmin -10 (modifié dans les dernières dix minutes)

-**mtime n** pour spécifier un critère de temps en jours sur la dernière modification

exemple: -mtime -0.5 (modifié dans la dernière demi-journée)

-**inum n** fichier dont le numéro de i-noeud est n

**ftp**

open on doit fournir le nom du serveur FTP

on doit entrer un nom d'utilisateur et le mot de passe

help pour voir toutes les commandes disponibles

pwd permet de voir dans quel répertoire on est sur le serveur ftp

cd permet de changer de répertoire sur le serveur ftp

ls affiche la liste des fichiers qui sont sur le serveur

lcd permet de changer de répertoire sur l'ordinateur local

get copier un fichier (et non un répertoire) du ftp à l'ordinateur local

mget copier plusieurs fichiers du ftp à l'ordinateur local

put copier un fichier (et non un répertoire) de l'ordi local au ftp

mput copier plusieurs fichiers de l'ordinateur local au ftp

bin pour effectuer un transfert binaire

prompt n désactive la confirmation lorsqu'on utilise mget ou mput

! pour exécuter une commande du SHELL

quit quitter le serveur ftp

note: pour une connexion anonyme

nom: ftp

mot de passe: **n'importe quoi**

**grep** (**G**lobal **R**egular **E**xpression **P**rint)

Recherche des lignes de fichiers contenant un motif particulier

grep "motif\_de\_recherche" [fichier]

grep "root" /etc/passwd

cherche toutes les lignes du fichier passwd contenant le motif root.

Options utiles

-c affiche seulement le nombre de lignes qui correspondent

grep -c "" sample.csv

wc -l sample.csv

sed -n '$=' sample.csv

(Dans le fichier sample.csv) Combien y-a-t'il de code postaux dans l'état de New York (NY)?

grep -c "NY" sample.csv

-i ignore la casse des caractères du motif ou du mot

-l affiche seulement les noms des fichiers qui contiennent le mot ou le motif

-n numérote les lignes affichées (telle que numérotées dans le fichier)

grep -n "12345" sample.csv

-v affiche les lignes **où l'on ne retrouve pas** le motif

Quelle est l'option que l'on peut passer à la commande grep pour qu'il retourne l'inverse du modif/pattern recherché?

grep -v "example" file.txt

Combien de lignes ne possèdent pas de lettre N (ou n)?

grep -vc [Nn] sample.csv

trouve qqchose dans un fichier

grep "21298" sample.csv

Note : Les double guillemets ne sont pas nécessaires si l'on effectue une recherche sans utiliser de caractères servant à former une *expression régulière*. On peut toutefois utiliser indépendamment les double guillemets si le motif est précis (sans caractères spéciaux).

**Les expressions régulières**

Attention: Les caractères spéciaux utilisé pour former les expressions régulières **ne signifient pas** la même chose que ces mêmes caractères utilisés comme métacaractères.

Voici les caractères spéciaux servant à former des expressions régulières

\ : banalise le métacaractère qui le suit.

^ : cherche le motif depuis le début de la ligne.

$ : cherche le motif depuis la fin de la ligne.

. : remplace un caractère quelconque.

\* : un nombre quelconque (voire nul) de répétition du caractère qui le précède.

[ ] : n'importe quel caractère dans l'ensemble spécifié entre crochet.

[^ ]: n'importe quel caractère hors de l'ensemble spécifié entre crochet.

Ainsi:

grep "^.o" /etc/passwd

Affiche les lignes du fichier passwd, où l'on retrouve, **à partir du début de la ligne** (^): **un caractère quelconque** (.) puis un o.

grep -c "^r" /etc/passwd

Affiche **le nombre de lignes** (-c) du fichier où l'on retrouve un r au **début de la ligne** (^).

grep -iv "^r" /etc/passwd

Affiche les lignes du fichier passwd où l'on **ne retrouve pas** (-v) un r en début de ligne, qu'il soit minuscule ou majuscule, puisque l'on ignore la casse (-i).

grep -n "r.o." /etc/passwd

Affiche et numérote (-n) les lignes du fichier passwd où l'on retrouve, **n'importe où sur la ligne**, la séquence r [caractère quelconque] o [caractère quelconque].

grep [xyz] /etc/passwd

Affiche les lignes du fichier /etc/passwd qui contiennent un x, un y ou un z.

grep -l "^E" /etc/X11/xorg\*

Affiche les noms des fichiers (-l) du répertoire X11, dont le nom commence par xorg et qui contiennent des lignes qui commencent par un E.

**grep "^[^k/] " /etc/passwd | grep "sh$" /etc/passwd**

Affiche les lignes du fichier /etc/passwd où l'on retrouve un motif qui commence par n'importe quel caractère **sauf** ([^]) le caractère k et le caractère /, puis qui se termine par sh. (Ainsi les lignes seront celles des comptes d'usagers dont le shell de connexion n'est ni sh, ni ksh, ni cshx).

pour afficher les lignes du fichier "/boot/grub/grub.conf" qui ne débutent pas par t ou k

**find /boot -iname grub.conf -exec grep "^[^tk]" {} \;**

**note: la syntaxe est plus simple avec la commande xargs**

**head**

est utilisé pour afficher le contenu des premières lignes d'un fichier (le nombre est à spécifier en paramètre).

pour afficher les six premières lignes du fichier "/boot/grub/grub.conf"

**head -6 /boot/grub/grub.conf**

**history**

est utilisé pour afficher la liste des dernières commandes entrées

La variable d'environnement "**HISTSIZE**" contient le nombre de lignes maximum que peut contenir la commande "**history**".

La variable **HISTSIZE** est dans le fichier "**/etc/profile**".

Pour exécuter la commande portant ce numéro dans l'historique de commandes.

**![#\_de\_l'historique]**

Pour exécuter la dernière commande de l'historique

**!!**

Pour effacer l'historique des commandes.

**history -c**

**init**

est utilisé pour changer de niveau d'exécution (runlevel)

init 0 ARRÊT DE L'ORDINATEUR

init 1 "Single user" - mode mono-utilisateur ou maintenance

init 2 multi-utilisateurs sans le support NFS

**init 3 multi-utilisateurs (mode texte)**

init 4 pas utilisé

**init 5 X11 (environnement graphique)**

init 6 REDÉMARRE L'ORDINATEUR

On peut configurer le niveau d'exécution (runlevel) dans le fichier "/etc/inittab"

On remplace x par (1,2,3,4,5) sur la ligne sur la ligne 🡺 **id:x:initdefault**

**NE PAS REMPLACER x PAR (0, 6) SUR LA LIGNE 🡺 id:x:initdefault**

**jobs**

est utilisé pour afficher les processus qui s'exécutent en "tâche de fond"

**kill**

est utilisé pour terminer l'exécution d'un ou plusieurs processus en utilisant le numéro de PID

Le signal 9 tue systématiquement les processus.

pour arrêter le processus 4116

**kill -9 4116**

pour arrêter les processus 4116 et 5125

**kill -9 4116 5125**

**killall**

est utilisé pour terminer l'exécution d'un ou plusieurs processus en utilisant le nom du processus

pour arrêter le processus xclock

**killall -9 xclock**

pour arrêter les processus xclock et xconsole

**killall -9 xclock xconsole**

**kdpf**

est utilisé pour lire des fichiers PDF

**kpdf test.pdf**

**less**

est utilisé pour arrêter l'affichage au bas de l'écran si le texte est trop long et permet de visualiser la suite en appuyant sur une touche.

La commande **less** est bidirectionnelle.

pour afficher le contenu du fichier "/boot/grub/grub.conf" page par page

**cat /boot/grub/grub.conf | less**

**ln**

est utilisé pour créer des liens durs ou des liens symboliques

Un inode est une ligne dans un tableau spécial situé à l'intérieur d'une partition et l'inode enregistre des informations sur le nom, le propriétaire, les droits d'accès, la taille et l'emplacement du fichier dans la partition.

**Liens durs**

Un seul et même fichier est susceptible d'apparaître dans plusieurs répertoires, en utilisant simplement le même inode à plusieurs reprises.

Les limitations d'un lien dur

* un lien dur n'existe qu'avec des fichiers
* la source et la destination doivent se trouver sur la même partition

**Pour créer le lien physique: ln test lien\_physique**

**ls -li**

**2108285  -rw-r--r--  2  root  root  nov  9  0  test**

**2108285  -rw-r--r--  2  root  root  nov  9  0  lien\_physique**

Le nombre de lien physique est comptabilisé. Ce n'est que lorsque le dernier lien a été supprimé qu'il n'est plus possible d'accéder au fichier fichier, l'inode est alors supprimé et l'espace disque occupé par le fichier est libéré.

**Liens symboliques**

Un lien symbolique peut exister sur des disques différents et des partitions différentes.

Un lien symbolique est une entrée de répertoire qui ne fait pas référence à l'inode d'un fichier existant, mais par l'intermédiaire d'un nom de chemin.

**Pour créer le lien symbolique: ln -s test lien\_symbolique**

**ls -li**

**2108285  -rw-r--r--  1   root  root  nov  9  0  test**

**2108395  lrw-r--r--  1   root  root  nov  9  0  lien\_symbolique 🡪 test**

**Le fichier vmlinuz est un lien symbolique**

lrwxrwxrwx 16 Dec 11 1999 vmlinuz -> vmlinuz-2.0.36-3

-rw-r--r-- 454325 Oct 13 1998 vmlinuz-2.0.36-0.7

-rw-r--r-- 454434 Dec 29 1998 vmlinuz-2.0.36-3

**ls**

est utilisé pour afficher le contenu d'un répertoire

**Par défaut, la commande ls n'affiche pas les fichiers cachés**

Options utiles

-a : affiche tous les fichiers, fichiers cachés y compris.

-d : affiche tous les répertoires (sans le contenu).

-i : affiche le numéro de l'inode

-s : affiche la taille, en blocs, des fichiers. Un bloc = 1024 octets ( ou 1 Ko ).

Note: Parfois ce peut être des blocs de 512 octets.

-1 : affiche les résultats sur une seule colonne.

-l : affiche le format long des fichiers et des répertoires, incluant les droits d'accès, le nombre de liens, le propriétaire, la taille en octets, la date du dernier accès, et enfin le nom du fichier.

-h : affiche la taille des fichiers en K, M, ...

-t : trie l'affichage des fichiers par date de dernière modification (les plus récents en premier)

-x : trie l'affichage ligne par ligne, de gauche à droite.

-r : inverse l'ordre de tri, quel qu'il soit.

-R : affiche récursivement les fichiers des répertoires.

**lspci**

est utilisé pour afficher la liste des périphériques PCI

**lynx**

est utilisé pour naviguer sur internet en mode texte

**man**

**LA COMMANDE "man" EST LA PLUS UTILE DE LINUX.**

contient les pages de manuels UNIX qui est constitué de 9 sections.

Les fichiers de la commande "man" sont dans les répertoires

* /usr/share/man/fr/man1 Commandes utilisateur
* /usr/share/man/fr/man2 Appels système
* /usr/share/man/fr/man3 Sous-routines
* /usr/share/man/fr/man4 Périphériques
* /usr/share/man/fr/man5 Formats de fichiers
* /usr/share/man/fr/man6 Jeux
* /usr/share/man/fr/man7 Divers
* /usr/share/man/fr/man8 Administration système
* /usr/share/man/fr/man9 Kernel (noyau)

**man ls**

**man cd**

**man passwd**

**man 5 passwd**

**cherche les commandes qui contiennent le mot efface**

**man -k efface**

**Accéder à une page du manuel à partir de Konqueror**

Dans la barre d'adresse internet de Konqueror.

Pour accéder à la page du manuel 1 de la commande ls

**man:/ls**

Pour accéder à la page du manuel 5 de la structure du fichier passwd

**man:/passwd(5)**

**mkdir (make directory)**

est utilisé pour créer des répertoires

On peut créer plusieurs répertoire à la fois !!!

mkdir [chemin], mkdir [chemin] [chemin],mkdir [chemin] [chemin] [chemin], ...

pour créer plusieurs répertoires avec une seule commande mkdir

**mkdir /tmp/noyau2 /tmp/noyau3**

**ou**

**mkdir /tmp/{noyau2,noyau3}**

pour créer une arborescence complexe

**mkdir -p /tmp/{prog/{java,delphi},lib/{info,kernel}}**

pour créer un répertoire et attribuer les autorisations en même temps

**mkdir /tmp/test -m 666**

**mkfs.ext3**

est utilisé pour formater un disque ou une partition en ext3

vérifie les blocs défectueux en mode verbose lors de la création de la partition

mkfs.ext3 -v -c /dev/sdb1

**CRÉATION D'UN SYSTÈME DE FICHIER VIRTUEL**

**création d'un fichier de 1 GO**

dd if=/dev/zero of=mondisque count=2048000

**vérifie la taille du fichier mondisque**

du -sh mondisque

**création d'un système de fichier ext3 en utilisant le fichier mondisque**

mkfs.ext3 mondisque

**création d'un point de montage**

mkdir /mnt/image

**monter le fichier mondisque dans le dossier /mnt/image**

mount -o loop mondisque /mnt/image

**mkfs.msdos**

est utilisé pour formater une disquette sous le format DOS

**Les étapes pour créer une disquette**

# commande pour formater la disquette

mkfs.msdos /dev/fd0

# créer un répertoire

mkdir /mnt/disquette

# commande pour monter la disquette

mount /dev/fd0 /mnt/disquette

# par exemple, si on copie le fichier de réponse sur la disquette

cp /root/anaconda-ks.cfg /mnt/disquette/ks.cfg

# commande pour forcer l'écriture du contenu sur la disquette

sync

# commande pour démonter la disquette

umount /mnt/disquette

**mkswap**

est utilisé pour créer une zone d’échange pour la pagination sur disque

crée une zone de mémoire virtuelle pour la pagination et vérifie la présence de secteurs défectueux

**dd if=/dev/zero of=/root/fichierswap bs=1024 count=2048**

**mkswap /root/fichierswap**

**swapon /root/fichierswap**

crée une zone de mémoire virtuelle pour la pagination et vérifie la présence de secteurs défectueux

**mkswap -c /dev/sda5**

**swapon /dev/sda5**

**more**

est utilisé pour arrêter l'affichage au bas de l'écran si le texte est trop long et permet de visualiser la suite en appuyant sur une touche.

La commande **more** est unidirectionnelle vers le bas.

pour afficher le contenu du fichier "/boot/grub/grub.conf" page par page

**cat /boot/grub/grub.conf | more**

**mount - sync - umount - eject**

la commande "**mount**" va monter un périphérique pour le rendre disponible

la commande "**sync**" force l'écriture sur le périphérique

la commande "**umount**" défait le lien créé par la commande "**mount**", en prenant soin de forcer l'écriture physique vers le périphérique avant de défaire ce lien.

**Attention, la commande "umount" doit être effectuée en-dehors du répertoire sur lequel on l'applique.**

la commande "**eject**" permet d'éjecter le CD-ROM du lecteur

**Exemple avec le lecteur de disquette**

Pour accéder à une disquette dans le lecteur A:

Le contenu du lecteur de disquette est placé en mémoire et son contenu est mis à jour à certains intervalles.

**mount /mnt/floppy** 🡺 assurez-vous d'avoir une disquette dans le lecteur

**sync** 🡺 force l'écriture sur la disquette et la disquette reste accessible

**umount /mnt/floppy** 🡺 force l'écriture sur la disquette et n'est plus accessible

**Exemple avec le lecteur de CD-ROM**

créer le répertoire /mnt/cdrom/

**mount /dev/cdrom /mnt/cdrom/** 🢂 assurez-vous d'avoir un CD dans le lecteur

**umount /mnt/cdrom/** 🡺 le contenu du CD-ROM n'est plus accessible

**ou**

**eject /mnt/cdrom/** 🡺 le CD-ROM est éjecté du lecteur et n'est plus accessible

**Exemple avec un fichier ISO**

**mount -o loop -t iso9660 fichier.iso /mnt/isoimage/**

**umount /mnt/isoimage**

**Exemple avec un fichier IMG**

**mount -o loop -t vfat /tmp/test1.img /mnt/fd1/**

**umount /mnt/fd1**

**Accéder à une partition NTFS à partir de Linux**

Le paquetage "NTFS 3G" est disponible sur le site <http://www.ntfs-3g.org> et permet d'avoir accès en lecture/écriture à une partition NTFS.

**mv (move)**

est utilisé pour déplacer un fichier ou pour renommer un fichier

pour renommer un fichier la destination est la même que la source

**mv /etc/cv.txt /etc/resume.txt**

Si le répertoire de travail est "/etc" et le fichier "cv.txt" est dans "/etc"

source absolue, destination absolue

**mv /etc/cv.txt /tmp/cv.txt**

source relative, destination absolue

**mv cv.txt /tmp/cv.txt**

source relative, destination relative

**mv cv.txt ../tmp/cv.txt**

**newgrp**

est utilisé pour se connecter avec un nouveau groupe

**nice**

est utilisé pour changer la priorité d'un processus

Par défaut, **nice** assigne une priorité de 10.

Si vous souhaitez créer une image ISO à partir du DVD d'installation de CentOS, vous utilisez d'habitude:

**dd if=/dev/cdrom of=/tmp/CentOS.iso**

Mais sur certains systèmes avec un lecteur DVD IDE standard, le fait de copier de grands volumes de données peut utiliser énormément de ressources.

Pour que cela n'empêche pas les autres personnes ou processus de fonctionner correctement, on peut démarrer le processus de copie avec une priorité affaiblie:

**nice -n 19 dd if=/dev/cdrom of=/tmp/CentOS.iso**

**parted**

est utilisé pour gérer les partitions des disques de type MBR ou GPT

liste les partitions d'un disque

**parted -l**

**pr**

est utilisé pour convertir du texte pour l'impression

**man ls | pr**

**man ls | pr > /tmp/ls\_man.txt**

**ps**

est utilisé pour afficher la liste des processus du système

Le processus **init** a le PID 1 et n'a pas de parent.

La signification des différentes colonnes est la suivante:

|  |  |
| --- | --- |
| **UID** | correspond au nom de l'utilisateur qui a lancé le processus |
| **PID** | correspond au numéro du processus |
| **PPID** | correspond au numéro du processus parent |
| **C** | correspond au facteur de priorité |
| **STIME** | correspond à l'heure de lancement du processus |
| **TTY** | correspond au nom du terminal |
| **TIME** | correspond à la durée de traitement du processus |
| **CMD** | correspond au nom du processus |

paramètres

a 🡺 affiche aussi les processus lancés par les autres utilisateurs

x 🡺 affiche aussi les processus n’ayant pas de terminal de contrôle (c’est le cas de pratiquement tous les serveurs) ou un terminal de contrôle différent de celui que vous êtes en train d’utiliser ;

u 🡺 affiche pour chaque processus le nom de l’utilisateur qui l’a lancé et l’heure de son lancement

par défaut, cette commande affiche uniquement les processus lancés par l’utilisateur à partir du terminal.

**ps**

pour afficher une liste des processus système en cours d'exécution, y compris les processus appartenant à d'autres utilisateurs et le nom du propriétaire.

**ps aux**

pour afficher les processus de l'utilisateur "root" seulement

**ps -u root**

**pstree**

est utilisé pour afficher l'arborescence des processus

pour afficher l'arborescence avec le numéro de PID

**pstree -p**

**pwd**

est utilisé pour afficher le répertoire actif

**renice**

Chaque processus actif sur le système s'exécute avec une priorité donnée.

Cette valeur peut varier de -20 (priorité la plus haute) à 19 (priorité la plus basse).

Si elle n'est pas définie, tous les processus s'exécutent avec une priorité de 0 par défaut.

Les processus ayant la priorité maximale (la valeur la plus basse, jusqu'à -20) s'exécutent plus souvent que les processus ayant une priorité inférieure (jusqu'à 19), et se voient ainsi offrir plus de temps processeur.

Les utilisateurs normaux peuvent seulement diminuer la priorité de leurs processus dans la fourchette 0 à 19.

Le super-utilisateur "root" peut définir n'importe quelle priorité à tous les processus.

**EXEMPLE 1**

Vous venez de lancer un processus qui est très long et qui a le numéro PID 3974.

Et pendant que ce processus tourne vous souhaitez jouer à un jeu qui nécessite que vous libériez des ressources système.

**renice +15 3974**

**EXEMPLE 2**

Si vous êtes l'administrateur d'un système partagé et constatez qu'un utilisateur lance beaucoup de processus accaparant les ressources du système, vous pouvez changer la priorité de tous les processus de cet utilisateur avec la commande:

**renice +20 -u richard**

Après cela, tous les processus de richard auront une priorité plus basse et ne bloqueront pas les processus des autres utilisateurs.

**rm (remove)**

est utilisé pour effacer des fichiers mais peut effacer des répertoires et son contenu

**Par défaut, la commande "rm" demande une confirmation de l'usager avant d'effacer.**

pour effacer un fichier (avec confirmation)

**rm /tmp/toto.txt**

pour effacer un fichier (sans confirmation)

**rm -f /tmp/toto.txt**

pour effacer un répertoire et son contenu (avec confirmation)

**rm -r /tmp/test**

pour effacer un répertoire et son contenu (sans confirmation)

**rm -rf /tmp/test**

**rmdir (remove directory)**

est utilisé pour effacer des répertoires vides

pour effacer un répertoire vide (sinon l'opération est refusée)

**rmdir /home/toto**

**si le dossier est plein rm -r**

**rsync**

est un outil qui permet le transfert des fichiers de façon incrémentale, c’est-à-dire qu’il ne copie que les nouveautés et modifications

pour télécharger les mises à jour de CentOS 5.2 (version 32 bits)

**rsync -av rsync://rsync.arcticnetwork.ca/centos/5.2/updates/i386 /tmp/updates/i386**

**seq**

affiche une séquence de nombres

seq 1 1 100

complète la largeur des nombres par des zéro

seq -w 1 1 100

**shutdown**

est utilisé pour arrêter la machine

pour arrêter maintenant

**shutdown -h now**

pour arrêter dans 30 minutes

**shutdown 30**

pour arrêter à une heure précise

**shutdown 12:35**

pour arrêter la machine après l'arrêt du système

**shutdown -h**

pour redémarrer la machine après l'arrêt du système

**shutdown -r**

pour annuler l'arrêt du système

**shutdown -c**

**sleep**

est utilisé pour faire attendre le processeur avant d'exécuter une autre commande.

Cette commande s'utilise surtout en conjonction d'autres commandes, à l'aide du séparateur point-virgule (;).

pour attendre 10 secondes avant d'éxécuter la commande "cat /root/info.txt"

**cat /boot/grub/grub.conf; sleep 10; cat /root/info.txt**

**sort**

* permet de trier un fichier de données contenant plusieurs champs
* il faut mentionner quel sera le caractère de séparation de champs dans le fichier

1151:Pierre Malvoisin:Repentigny:98.90

1234:Marie Zoltoc:Montreal:235.23

1275:Louise Miville:Laval:0.58

1352:Mario Brioche:Montreal:18.95

1425:Maurice Titué:Montreal:25.75

le symbole séparteur est :

pour trier à partir du 1er champ

**sort –t: nomfichier**

pour trier en ordre inverse à partir du 1er champ

**sort –t: -r nomfichier**

pour trier à partir du 2ème champ

**sort –t: -k 2 nomfichier**

pour trier à partir du 5ème caractère du 3ème champ

**sort –t: -k 3.5 nomfichier**

pour trier le 4ème champ par nombre

**sort –t: -n -k 4 nomfichier**

sort -t, -r -k 5.3 sample.csv

Mettre en ordre décroissant selon la troisième lettre du cinquième champ du fichier sample.csv. Les champs du fichier sont séparés par des virgules

**split**

est utilisé pour découper un fichier en plusieurs petits fichiers

pour diviser le fichier "f1.txt" en plusieurs fichiers contenant 5 lignes chacun qui se nommeront "f1aa", "f1ab", "f1ac", ….

**split -5 f1.txt f1**

pour afficher le contenu des fichiers f1aa, f1ab, f1ac, …

**ls f1??**

pour réunir les morceaux de fichiers en un seul fichier

**cat f1?? > f1\_assembler.txt**

pour diviser le fichier "f2.txt" en plusieurs fichiers de 2 KO chacun qui se nommeront "f2aa", "f2ab", "f2ac", ….

**split -b 2k f2.txt f2**

**startx**

est utilisé pour démarrer l'environnement graphique

utilise la commande startx sans paramètre

**startx &**

Il est possible d'ouvrir deux environnements graphiques en même temps.

pour démarrer un environnement graphique dans l'écran virtuelle 7

**startx -- :0 vt7 &**

pour démarrer un environnement graphique dans l'écran virtuelle 8

**startx -- :1 vt8 &**

**su (switch user)**

est utilisé pour exécuter une commande "shell" avec un user-id et un group-id différents. Il faut connaître le mot de passe de l'utilisateur au niveau duquel on passe.

ne donne pas accès aux variables d'environnement du root

**su**

donne accès aux variables d'environnement du root

**su -**

permet d'exécuter une commande en tant que root

**su -c ifconfig**

**su -c 'ls -l /bin/'**

**swapon**

active la pagination sur disque

active la zone de mémoire virtuelle

**swapon /dev/sda5**

affiche les zones de la mémoire virtuelle active

**swapon -s**

**swapoff**

désactive la pagination sur disque

désactive toutes les zones de mémoire virtuelle

**swapoff -a**

**tac**

est utilisé pour afficher le contenu d'un fichier mais à l'envers !!!

**tail**

est utilisé pour afficher le contenu des dernières lignes d'un fichier (le nombre est à spécifier en paramètre)

pour afficher les trois dernières lignes du fichier "/boot/grub/grub.conf"

**tail -3 /boot/grub/grub.conf**

**Combinaisons de head et tail**

Il est possible de combiner head et tail avec | pour visualiser des lignes qui sont au milieu d'un fichier.

pour afficher les lignes 10 à 12 (inclusivement) du fichier "/boot/grub/grub.conf"

**head -12 /boot/grub/grub.conf | tail -3**

**top**

est utilisé pour afficher la liste des processus de façon dynamique à l'écran

**touch**

est utilisé pour créer un fichier vide

pour créer un fichier vide "toto.txt" dans le répertoire "root"

**touch /root/toto.txt**

**type**

la commande type affiche un message qui décrit le type de fichier (alias, fonction "shell" , exécutable)

**type -a shutdown**

**type -a cd**

**type -a pwd**

pwd is a shell builtin

pwd is /bin/pwd

si on veut exécuter l'instruction builtin pwd

**builtin pwd**

**type -a ls**

ls is aliased to `ls -F --color=auto'

ls is /bin/ls

si on veut exécuter /bin/ls à la place de l'alias

**command ls**

**unalias**

est utilisé pour défaire un alias

pour défaire l'alias bye

**unalias bye**

**uname**

est utilisé pour afficher des informations sur le système

pour afficher la version du noyau

**uname -r**

pour afficher le nom de la machine, identique à la commande hostname

**uname -n**

**unix2dos**

est utilisé pour convertir un fichier UNIX en fichier DOS

**unix2dos /tmp/info.txt**

**uptime**

est utilisé pour indiquer le temps depuis l'ouverture du serveur et le nombre de login actif

**watch**

est utilisé pour vérifier les modifications: dans un dossier, la taille d'un fichier

pour vérifier le téléchargement d'un fichier ISO à chaque 2 secondes

**watch ls -l fichier.iso**

pour vérifier le téléchargement d'un fichier ISO à chaque 10 secondes

**watch -n 10 ls -l fichier.iso**

pour vérifier le changement dans un dossier en affichant la différence

**watch -d ls -l /tmp/**

**wget**

un programme non interactif de téléchargement de fichiers depuis le WEB.

il supporte les protocoles FTP, HTTP, HTTPS

pour télécharger plusieurs fichiers à partir d'un serveur FTP

**wget ftp://0123456:aammjj@extranet.cvm.qc.ca/rjean/\*.rpm**

pour télécharger un fichier à partir d'un serveur HTTP

**wget http://www.cvm.qc.ca/rjean/r1chard\_jean\_b22\_b32\_adresses.htm**

**which**

est utilisé pour retourner le chemin d'un utilitaire (comme **cat**, ou **cal**...).

pour afficher le chemin de l'utilitaire cat

**which cat**

**xargs**

est utilisé pour exécuter des lignes de commandes à partir de l'entrée standard

pour afficher le chemin des fichiers menu.\*

**find / -iname menu.\* | xargs ls -l**

pour afficher le numéro des lignes avec le mot "title" dans les fichier "menu.\*"

**find / -iname menu.\* | xargs grep -n title**

pour afficher les lignes du fichier "/boot/grub/grub.conf" qui ne débutent pas par t ou k

**find /boot -iname grub.conf | xargs grep "^[^tk]"**

**sed**

Permet le remplacement de texte à partir d’un fichier

Remplace le mot ancien pour le mot nouveau dans le fichier monFichier.txt et créer le fichier final.txt

**sed –i ‘/s/ancien/nouveau/g’ /home/fred/monFichier.txt > final.txt**

**tr**

Permet de faire un remplacement de texte

Remplace chateau par bateau. Sans l’option s, ça ferait : bbateau

**echo -s chateau | tr ch b**

**Comment effacer un disque ou une partition**

Commande pour remplacer le contenu d’un disque par des 0

* cat /dev/zero > /dev/sda

Commande pour remplacer le contenu d’une partition par des 0

* cat /dev/zero > /dev/sda1

Commande pour remplacer le contenu d’un disque par des valeurs aléatoires

* cat /dev/urandom > /dev/sda

Commande pour remplacer le contenu d’une partition par des valeurs aléatoires

* cat /dev/urandom > /dev/sda1

**Le temps d'exécution peut être très long selon la capacité des partitions et des disques.**

La commande shred permet d’effacer le contenu d’un fichier en exécutant plusieurs passes d’écriture successive.

* shred -vzn 7 /tmp/info.txt

On peut effacer une partition parce qu'une partition utilise un nom de fichier précis.

* shred -vzn 7 /dev/sda1

**Le temps d'exécution peut être très long selon la capacité des partitions et des disques.**

**Commandes qui existent seulement dans RedHat et CentOS**

system-config-authentication

system-config-date

system-config-display

system-config-httpd

system-config-kdump

system-config-keyboard

system-config-kickstart

system-config-language

system-config-lvm

system-config-netboot

system-config-network

system-config-network-cmd

system-config-network-gui

system-config-network-tui

system-config-nfs

system-config-packages

system-config-printer

system-config-rootpassword

system-config-samba

system-config-securitylevel

system-config-securitylevel-tui

system-config-services

system-config-soundcard

system-config-time

system-config-users

system-control-network

system-install-packages

Vrai ou faux : Linux contient du code Unix Faux

Question #25 (45sec) Le créateur du C a également contribué au développement de … Unix

**Commandes Linux – Aide-mémoire**

|  |  |
| --- | --- |
| [Ctrl-A] | Amener le curseur au début de la ligne courante |
| [Ctrl-E] | Amener le curseur à la fin de la ligne courante |
| [Ctrl-K] | Tout effacer de la position du curseur jusqu’à la fin de la ligne |
| [Ctrl-L] | Nettoyer l’écran (équivalent à la conmmande *clear*) |
| [Ctrl-D] | Fermer la session shell actuelle (si ligne vide)  Indiquer la fin d’un fichier |
| [Ctrl-C] | Interrompre la commande en cours d’exécution (si ligne vide).  Interrompre l’édition d’une commande (si ligne non vide) |
| [Ctrl-Z] | Arrêter une tâche de façon temporaire |
| [Ctrl-S] | Suspendre le flux de cacactères sur un terminal |
| [Ctrl-Q] | Restaurer le flux de caractères sur un terminal |
| [Tab] | Compléter le nom d’une commande, d’un fichier, d’un répertoire ou d’une variable d’environnement |
| $ | Invite de commande d’un usager normal |
| # | Invite de commande de l’usager *root* |
| / | Référer au répertoire racine |
| . | Référer au répertoire courant |
| .. | Référer au répertoire parent |
| ~ | Référer au répertoire de l’usager actuel du shell |
| ~*nom* | Référer au répertoire de l’usager ayant ce nom |
| /home/*nom* | Référer au répertoire de l’usager ayant ce nom |
| ? | Désigner n’importe quel caractère pour l’expansion d’un nom de fichier |
| \* | Désigner plusieurs caractères pour l’expansion d’un nom de fichier |
| [ac] | Désigner les caractères a ou c pour l’expansion d’un nom de fichier |
| [a-c] | Désigner n’importe quel des caractères inclus dans l’intervalle a à c pour l’expansion d’un nom de fichier |
| [^a-c] | Désigner tous les caractères sauf ceux inclus dans l’intervalle a à c pour l’expansion d’un nom de fichier. |
| \ | Découper une ligne de commande sur plusieurs lignes |
| ; | Enchaîner inconditionnellement des commandes. Toutes les commandes sont exécutées même si l’une d’entre elles provoque une erreur. |
| && | Enchaîner conditionnellement des commandes. La prochaine commande n’est exécutée que si la commande actuelle retourne un code 0 (exécution correcte). |
| || | Enchaîner conditionnellement des commandes. La prochaine commande n’est exécutée que si la commande actuelle retourne un code différent de 0 (exécution erronée). |
| < | Rediriger un fichier dans l’entrée d’une commande |
| > | Rediriger la sortie d’une commande dans un fichier (écrase le contenu actuel) |
| >> | Rediriger la sortie d’une commande dans un fichier (ajoute le contenu à la fin) |
|  |  |
| | | (pipeline) Rediriger la sortie d’une commande dans l’entrée de la commande suivante. |
| & | Mettre une commande en tâche de fond (quand placé à la fin d’une ligne). Voir bg, jobs et fg. |

|  |  |
| --- | --- |
| alias | Définir une commande sur mesure par un identifiant (souvent utilisé pour ajouter des options à une commande existante |
| bg | Mettre une commande en tâche de fond |
| cal | Afficher un calendrier |
| cat | Concaténer plusieurs fichiers  Afficher le contenu complet d’un fichier |
| cd | Changer le répertoire courant |
| clear | Nettoyer l’écran |
| cp | Copier un fichier |
| cut | Sélectionner une partie de chacune des lignes |
| date | Afficher ou modifier la date du système |
| dd | Copier des fichiers en spécifiant la taille des blocs |
| df | Afficher des informations sur l’espace utilisé |
| dos2unix | Convertir un fichier DOS en Unix |
| du | Afficher les statistiques d’utilisation du disque |
| e2fsck | Vérifier un système de fichiers Linux et2 et ext3 |
| echo | Envoyer un texte en sortie |
| eject | Éjecter le CD-ROM |
| fdformat | Formater un périphérique |
| fg | Remettre une tâche de fond en commande d’avant-plan |
| file | Indiquer le type de fichier |
| find | Trouver des fichiers sur le disque |
| ftp | Ouvrir un client ftp |
| grep | Chercher des lignes contenant un motif |
| head | Afficher le contenu des premières lignes d’un fichier |
| history | Afficher l’historique des commandes de l’utilisateur courant |
| init | Change rle niveau d’exécution |
| jobs | Afficher les processus qui s’exécutent en tâche de fond. Voir aussi *bg* et *fg*. |
| kill | Terminer l’exécution d’un ou plusieurs processus (avec le numéro de processus)  Envoyer un signal à un processus |
| killall | Terminer l’exécution d’un ou plusieurs processus (avec le nom de processus) |
| less | Afficher un texte. Voir aussi *more* |
| locate | Trouver un fichier |
| ln | Créer des liens sur des fichiers |
| ls | Afficher le contenu des répertoires |
| lynx | Naviguer sur Internet en mode texte |
| man | Afficher le manuel d’utilisation d’un élément (commande, appel système, etc.) |
| mkdir | Créer des répertoires |
| mkswap | Créer une zone d’échange pour la pagination sur disque |
| more | Afficher un texte page par page. Voir aussi *less*. |
| mount | Monter un périphérique pour le rendre disponible |
| mv | Déplacer ou renommer un ficher/répertoire |
| newgrp | Se connecter avec un nouveau groupe |
| nice | Changer la priorité d’un processus |
| pr | Convertir du texte pour l’impression |
| ps | Afficher la liste des processus du système |
| pstree | Afficher l’arborescence des processus |
| pwd | Afficher le répertoire actif |
| renice | Changer la priorité d’un processus |
| rm | Effacer des fichiers/répertoire |
| rmdir | Effacer des répertoires (vides) |
| rsync | Synchroniser deux répertoires (transfert des modifications seulement) |
| seq | Afficher une séquence de nombres |
| shutdown | Arrêter la machine |
| sleep | Faire une pause |
| sort | Trier les lignes d’un texte |
| split | Découper un fichier en plusieurs fichiers |
| startx | Démarrer l’environnement graphique |
| su | Exécuter une commande shell avec un utilisateur différent |
| swapon | Activer la pagination sur disque |
| swapoff | Désactiver la pagination sur disque |
| sync | Forcer l’écriture sur le périphérique |
| tac | Afficher le contenu d’un fichier à l’envers |
| tail | Afficher les dernières lignes d’un fichier. Voir aussi *head*. |
| top | Afficher la liste des processus de façon dynamique |
| touch | Créer un fichier vide |
| type | Afficher le type de fichier exécutable |
| umount | Défaire le lien créé par mount |
| unalias | Défaire un alias |
| uname | Afficher les informations du système |
| unix2dos | Convertir un fichier Unix en fichier DOS |
| uptime | Indiquer le temps depuis le dernier démarrage du système |
| wget | Télécharger des fichiers avec les protocoles HTTP(S) et FTP |
| which | Retourner le chemin d’un fichier exécutable |
| xargs | Exécuter des commandes à partir des éléments de l’entrée standard |