

# Aide mémoire Bash et commandes systèmes

## Bash

### Raccourcis clavier

```

Ctrl + A      Go to the beginning of the line you are currently typing on
Ctrl + E      Go to the end of the line you are currently typing on
Ctrl + L      Clears the Screen, similar to the clear command
Ctrl + U      Clears the line before the cursor position. If you are at the
end of the line, clears the entire line.
Ctrl + H      Same as backspace
Ctrl + R      Let's you search through previously used commands
Ctrl + C      Kill whatever you are running
Ctrl + D      Exit the current shell
Ctrl + Z      Puts whatever you are running into a suspended background
process. fg restores it.
Ctrl + W      Delete the word before the cursor
Ctrl + K      Clear the line after the cursor
Ctrl + T      Swap the last two characters before the cursor
Esc + T      Swap the last two words before the cursor
Alt + F      Move cursor forward one word on the current line
Alt + B      Move cursor backward one word on the current line
Tab        Auto-complete files and folder names

```

### Evaluer une expression

```

let z=z+1
z=$((z + 1))

```

## Boucles

### for

```

for i in * ; do          # * retourne la liste des fichiers
    echo $i
done
for i in `cat liste` ; do      # i va prendre les valeurs
                                # contenues dans le fichier "liste"
    ls $i
done

```

```
for i in 1 2 3 ; do # i prend successivement les valeurs 1, 2 et 3
    echo $i
done
for ((i=0;i<10;i=i+1));do echo $i; done
```

## until

```
a=0
until [ $a == 10 ] ; do
    let a=a+1; echo $a;
done
```

## while

```
while
    Commande_Test
do
    Commandes
done
```

## I/O

### read

```
echo n "Entrez votre nom :"
read nom
echo $nom
```

### select

```
select choix in \
    "Choix A" \
    "Choix B";
do
    case $REPLY in
        1) echo "$CHOIX > $REPLY";;
        2) echo "$CHOIX > $REPLY";;
        *) echo "Vous avez tapé n'importe quoi !";;
    esac
done
```

## Tests

```
if test 3 eq 3 ...
if [ 3 eq 3 ]....
if test $1 == $2 ...
if [ $1 == $2 ]...
```

### Tests sur les fichiers

- -d, si c'est un répertoire
- -e, si le fichier existe
- -f, si le fichier existe et si c'est un fichier standard

### Tests sur les chaînes

- s1 == s2, si les chaînes s1 et s2 sont identiques
- s1 != s2, si les chaînes sont différentes

```
if [ foo == foo ]; then echo true; fi
```

### Tests sur les entiers

- n1 eq n2, si n1 est égal à n2
- opérateurs ne, eq, gt, lt, le ge (not equal, equal, greater, lower, lower or equal, greater or equal)

## Variables spéciales

- Les paramètres sont passés dans \$1, \$2...\$n
- \${@}, donne la liste des paramètres
- \${#}, donne le nombre de paramètres
- \${\$}, donne le numéro (PID) du script
- \${0}, donne le nom du script
- \${?}, donne la valeur de retour d'un script ou d'une fonction

## Variables d'environnement

- Lister les variables d'environnement

```
env #ou# printenv #ou# export
```

- Fichier de configuration global. Contient certaines variables d'environnement valables pour tous les utilisateurs du système.

```
cat /etc/profile
```

- Fichiers de configuration utilisateur

\$HOME/.bash\_profile et \$HOME/.bashrc

- Placer une variable dans l'environnement (exemple : PS1, modèle de prompt).

```
PS1="[\u@\h \w]\$\\" export PS1
```

- Utiliser les variables d'un script dans un autre script <code bash> source monautrescript # ou .monautrescript </code>

## Regexp - Expressions régulières / rationnelles

- [Tutoriel complet \(en\)](#)
- [Tutoriel sur phpmanuel \(fr\)](#)
- [Aide mémoire Adobe](#)
- [Aide mémoire - Site du zéro](#)
- [http://www.lumadis.be/regex/tuto\\_pcre.php](http://www.lumadis.be/regex/tuto_pcre.php) - Excellent tutoriel PCRE

## Liens

- [Advanced Bash scripting guide](#)
- [Tutoriel](#)
- [Tutoriel Ksh](#)

## Commandes

## Démarrage

En général les options de démarrage sont dans le fichier /etc/inittab, exemple :

```
# Default runlevel. The runlevels used by Mandrakelinux are:  
# 0 - halt (Do NOT set initdefault to this)  
# 1 - Single user mode  
# 2 - Multiuser, without NFS (The same as 3, if you do not have  
networking)  
# 3 - Full multiuser mode  
# 4 - unused  
# 5 - X11  
# 6 - reboot (Do NOT set initdefault to this)  
#  
id:3:initdefault:
```



**Dans ubuntu seulement**, depuis 6.06 c'est upstart qui gère les niveaux de démarrage, répertoire /etc/event.d/

- Connaître le runlevel

```
runlevel
```

- Changer le runlevel à 2

```
telinit 2
```

- Menu de démarrage

```
cat /boot/grub/menu.lst
```

## Arrêt

Ne pas utiliser les commandes halt et reboot

### shutdown

- shutdown arrête dans l'ordre les services, démonte les partitions, envoie un message d'arrêt, coupe l'alimentation.

```
#halt
shutdown -h now
#reboot
shutdwn -r now
#halt avec message d'avertissement et délai
shutdown -h+10 "les services réseaux vont s'arrêter et reprendre dans 2 heures"
```

## Dépannage

### Créer une disquette de démarrage

```
mkbootdisk --device /dev/fd0 `uname -r`
```

### Disquette de root ou rescue

```
#Recupérer l'image de la disquette sur le cdrom d'install. Il s'agit d'un
fichier ayant l'extension .img souvent dans le répertoire images
mkbootdisk --device /dev/fd0 `uname -r`
```

```
dd if=rescue.img of=/dev/fd0 bs=1440
```

## Mot de passe de root oublié

- booter avec les disquettes de boot et de root ou en mode single, le système ne vous demandera pas de mot de passe,
- ouvrez une session root (pas besoin de mot de passe),
- monter le disque dur exemple

```
root:~# mount -t ext3 /dev/sda1 /mnt
```

- modifiez le fichier /mnt/etc/passwd. Attention, ne touchez pas directement au fichier /etc/passwd si les mots de passes sont cachés. Il faut vous "chrooter".

```
cd /mnt chroot . # Là utiliser la commande passwd. passwd
```

## Démarrer en single-user

Au message boot : entrez

```
linux single root=/dev/hdxx initrd=initrd-`uname -r`.img
#Charger le clavier :
loadkeys /usr/share/keymaps/i386/azerty/fr-latin1.kmap.gz
#monter la racine : (-w = read+write ; -n = ne pas écrire dans /etc/mtab)
mount -w -n -o remount /
```

## Système de fichiers

### mkdir

- Créer un répertoire en créant les parents en même temps s'ils n'existaient pas

```
mkdir -p /chemin/de/mon/répertoire
```

### cp

- copier en conservant les droits

```
cp -cp
```

## rm

- supprimer un répertoire non vide

rm -Rf répertoire

- Supprimer uniquement les fichiers cachés et tous les sous-répertoires (Les ?? servent à exclure le répertoire parent)

rm -rf .??\*

## chmod

- changer les droits d'accès (u:user, g: group, o:other)

```
chmod -R 775 ./*
#Donner tous les droits pour utiliser avec Apache
find LeRépertoire -type d | xargs chmod a+x
find LeRépertoire -type f | xargs chmod a-x,+r
```

## droits ugoa

- Sticky bit

drwxrwxrwt 15 root root 8192 20051007 18:05 tmp

Seul le propriétaire de chaque fichier de ce répertoire pourra supprimer le fichier en question.

- suid, sgid

Permet de donner à un programme les droits de l'utilisateur ou du groupe "root" lors de son exécution. Remplace le "x" par un "s"

#exemple

1. rwsr-xr-x 1 root root 29104 2007-05-18 11:59 /usr/bin/passwd

- Valeurs des droits

	r	w	x
Propriétaire	400	200	100
Groupe	40	20	10
Autres	4	2	1
suid	4000		
sgid	2000		
sticky	1000		

Remarque : pour un répertoire le droit d'exécution permet d'accéder au répertoire mais pas de le lister.

## chown, chgrp

- changer le propriétaire

```
chown user.group fichier
```

- changer le groupe

```
chgrp groupe fichier
```

## du

- Occupation disque

```
#Occupation du répertoire courant du -hs #Occupation des fichiers et répertoires courants for i in *; do du -hs $i; done;
```

## ln

- Créer des liens

```
ln Fichier.origine Nom.lien # crée un lien physique
ln -s Fichier.origine Nom.lien # crée un lien symbolique
```

## fuser

- Lister les pid de processus utilisant un fichier donnée

```
fuser 139/tcp #Donne le pid du processus utilisant le port 139 en tcp
```

## dd

- Créer un fichier vide de 100M

```
dd if=/dev/zero of=test_ext3fs bs=1M count=100
```

# Partitions

## df

- (disk free) : espace libre sur chaque partition des disques montés, informations en “human readable”.

```
df -h
```

## **fdisk**

- Lister tous les disques (montés et non montés)

```
fdisk -l
```

## **blkid**

- Lister les partitions avec leur UUID

```
blkid
```

## **mount / umount**

- Lister les disques montés

```
mount
```

- Monter un lecteur CD ou DVD

```
mount -t iso9660 -r /dev/hda /mnt/cdrom
```

- Monter et démonter une image iso (le périphérique loop sert à monter des systèmes de fichiers virtuels)
 

```
<code> cd /usr/local/fs # Copie ou téléchargement de l'image du CDROM dans un fichier cp /dev/cdrom image.iso # calcul de l'empreinte digitale de l'image du CDROM et de l'image # vérifiez que les deux empreintes sont identiques # c'est une bonne habitude à prendre. md5sum /dev/cdrom/ && md5sum image.iso mkdir /mnt/virtualcdrom mount -o loop -t iso9660 -v image.iso /mnt/virtualcdrom ls /mnt/virtualcdrom umount -d /mnt/virtualcdrom</code>
```

Inscription dans fstab

```
/usr/local/fs/image.iso /ou/vous/voulez iso9660 loop,noauto,ro,exec 0 0
```

## **mkisofs**

- Créer une archive au format iso

```
cd /tmp && mkisofs -o 'date +%d-%B-%Y'-backup.home.iso /home
```

## **losetup**

- Attribuer le type de périphérique loop à un fichier

```
dd if=/dev/zero of=testext3fs bs=1M count=100 mkfs.ext3 testext3fs losetup /dev/loop1  
test_ext3fs
```

- Utiliser la cryptographie sur un fichier <code> # On charge le module de cryptage modprobe cryptoloop losetup -e aes /dev/loop0 testext3fs # Il faudra saisir un mot de passe mkfs.ext3 /dev/loop0 mount -t ext3 -v /dev/loop0 /mnt/virtualfs rsync -av /home/ /mnt/virtualfs/home find /mnt/virtualfs umount /mnt/virtualfs losetup -d /dev/loop0 # Le système est démonté. Pour le lire il faudra # le remonter et entrer le mot de passe. mount -t ext3 -v -o loop,encryption=aes testext3fs /mnt/virtualfs </code>

## cfdisk

- changer les partitions d'un disque (attention efface le contenu !)

```
cfdisk /dev/hda
```

## mkfs

- Formater une partition en reiserfs

```
mkfs -t reiserfs /dev/hda1
```

## /etc/fstab

- Points de montage des partitions

```
cat /etc/fstab
```

## Liens

- [UUID et Label](#)
- <https://help.ubuntu.com/community/DataRecovery>

# Traitement des fichiers texte

## awk

- Exemples

```
awk -F ":" '{ $2 = "" ; print $0 }' /etc/passwd # imprime chaque ligne du fichier /etc/passwd après avoir effacé le deuxième champs  
awk 'END {print NR}' fichier # imprime le nombre total de lignes du
```

```
fichiers
awk '{print $NF}' fichier # imprime le dernier champs de chaque ligne
who | awk '{print $1,$5}' # imprime le login et le temps de connexion.
awk 'length($0)>75 {print}' fichier # imprime les lignes de plus de 75
caractères. (print équivaut à print $0)
```

- [Tutoriel \(fr\)](#)

## cut

Fait des coupes (sombres) dans les lignes qui lui sont transmises

- Afficher la liste des comptes système cat /etc/passwd | cut -d":" -f1

## sed

- Remplacer day par night dans old vers new

```
sed 's/day/night/' <old>new
```

- Supprimer des lignes vides en fin de fichier

```
find -name ".php" | xargs sed -i -e :a -e '/^$/{$d;N;ba' -e '}'
```

## Liens

- [Tutoriel court\(en\)](#)
- [Tutoriel semi-complet \(fr\)](#)
- [La référence sur sed](#)

## xargs

Exécute des commandes système en prenant l'entrée standard comme liste d'objets

- Exemples

```
#Liste toutes les droits du programme passwd
which passwd | xargs ls -al
#Faire la liste alphabétique sur une seule ligne des utilisateurs du système
cut -d: -f1 < /etc/passwd | sort | xargs echo
```

## Découpage de chaînes

Bash possède de nombreuses fonctionnalités pour découper des chaînes de caractères. L'une des plus pratiques est basée sur des motifs. La notation ## permet d'éliminer la plus longue chaîne en correspondance avec le motif :

```
$ Var='tonari no totoro'  
$ echo ${Var##*to}  
ro
```

ici le motif est \*to, et la plus longue correspondance "tonari no totoro"1. Cette forme est utile pour récupérer l'extension (suffixe) d'un nom de fichier :

```
$ F='rep/bidule.tgz'  
$ echo ${F##*.}  
tgz
```

La notation # (un seul #) est similaire mais élimine la plus courte chaîne en correspondance :

```
$ Var='tonari no totoro'  
$ echo ${Var#*to}  
nari no totoro
```

De façon similaire, on peut éliminer la fin d'une chaîne :

```
$ Var='tonari no totoro'  
$ echo ${Var%no*}  
tonari
```

Ce qui permet de supprimer l'extension d'un nom de fichier :

```
$ F='rep/bidule.tgz'  
$ echo ${F%.*}  
rep/bidule
```

% prend la plus courte correspondance, et %% prend la plus longue :

```
$ Y='archive.tar.gz'  
$ echo ${Y%.*}  
archive.tar  
$ echo ${Y%%.*}  
archive
```

## Hardware

### Ishw

- Lister la config matérielle

Ishw | less

## lsusb

- Lister les périphériques usb connectés

```
lsusb
```

## lspci

- Lister les composants pci

```
lspci
```

## paquets

### dpkg

- installer un fichier .deb

```
dpkg -install fichier.deb
```

### apt

- réparer des dépendances non résolues

```
apt-get -f install
```

- Sources d'apt

```
cat /etc/apt/sources.list
```

## Programmes

### symbole &

- Lancer un programme en tâche de fond

```
macommande &
```

### jobs

- Lister les travaux en cours

jobs

## fg, bg

- Passer du premier plan à l'arrière plan et réciproquement, exemple <code> \$ sleep 20 #On tape [ctrl]+z dans la foulée et la commande est mise en pause [1]+ Stopped sleep 20 \$ bg 1 #On la redémarre en arrière plan [1]- sleep 20 & \$ fg 1 #On la repasse en premier plan </code>

## nohup

- Lancer un programme insensible aux HangUPs et en tâche de fond

```
nohup macommande > fichier.log &
#Redirige également les messages d'erreur vers le néant
nohup macommande > /dev/null 2>&1 &
```

## ps, pstree, pidof, top

- Lister les processus <code bash> ps # liste des processus du terminal courant. ps -u utilisateur # liste des processus appartenant à l'utilisateur user. # liste de tous les processus du système. ps -ef ps aux # (syntaxe BSD) pstree | less # Affiche l'arborescence des processus pidof postgres # Affiche les pid du service postgres top # Affiche une liste dynamique des processus </code>

## Kill

Tuer ou envoyer des signaux à des processus

```
kill 546 #Tenter de détruire le processus n°546
kill -9 546 #Forcer la destruction du processus n°546
kill -15 -1 #Tente de détruire tous les processus de l'utilisateur courant
kill -9 -1 #Force la destruction de tous vos processus
kill `cat /var/run/squid.pid` #supprime le processus squid sur la base de son pid déclaré
kill -CONT pid #Demander à un procesus de redémarrer
```

## Killall

- Relancer un service sans changer son PID (Il relit ses fichiers de configuration)

killall -HUP service

- Tuer un processus dont on connaît le nom

```
killall -9 service
```

## réseau

- Connaître les machines actives sur le réseau local (ping de broadcast)

```
ping 192.168.1.255 - b
```

- Connaître les connexions actives et inactives

```
netstat -taupe | sort  
netstat -a | grep monservice
```

- Liste des ports prédéfinis

```
cat /etc/services
```

- Liste des alias de noms d'hôtes

```
cat /etc/hosts
```

- tester les ports ouverts

```
nmap adresseip  
nmap adresseip -p port
```

- Connaître les fichiers réseau ouverts

```
lsof -Pi
```

- Tracer une adresse IP

```
traceroute adresseIP
```

## sauvegardes

### Avec Tar

- décompresser en tar et gzip

```
tar xzvf cible.tgz
```

- Compresser en tar gzip

```
tar czvf cible.tgz source
```

- Consulter les fichiers de l'archive

```
tar tzvf archive.tgz
```

- Faire une sauvegarde de la distribution située en hda1 (source) vers hda2 (cible).

Passez en root :

```
su
```

Montez la cible en /mnt/disk par exemple :

```
mount /dev/hda2 /mnt/disk
```

Utilisez l'option -t pour préciser le type du système de fichiers si vous rencontrez un problème.

Placez-vous dans le répertoire cible :

```
cd /mnt/disk
```

Lancez la copie de la source :

```
(cd / ; tar cf - *) | tar xvf
```

## Avec rsync

- fichier exclude.lst <code> /proc/ /media/disk/ /tmp/ /dev/ /sys/ /etc/fstab /var/tmp/ /var/cache/ /cache/ /Cache/ /var/lock/ /var/run/ /var/spool/ </code>
- commande

```
rsync -uav -delete -exclude-from=/home/murielle/.rsync/exclude.lst / /media/disk/
```

## Liens

- Sauvegardes incrémentales (en)
- Hotcloning(fr)

## Avec cpio

- Créer une archive sur disquette

```
ls Document1 Photo1 Diaporama1 | cpio -ocvB > /dev/fd0
```

- Consulter l'archive

```
cpio tv < archive1
```

- Ouvrir l'archive

```
cpio i < archive1
```

## Avec gzip

- Compresser et décompresser

```
#Compresser
gzip docu
gzip -9 docu #Précise le niveau de compression le plus élevé
#Décompresser
gzip -d docu.gz
```

## Avec dd

- copie d'une disquette
- ```
dd if=/dev/fd0 of=Disquette
```
- Sauvegarder et restaurer le MBR (Master Boot Record)

```
dd if=/dev/hda of=sauve.mbr count=1 bs=512 dd if=sauve.mbr of=/dev/hda
```

## Liens

- [Les sauvegardes sur Lea-linux](#)
- [Politique de sauvegarde](#)

## Utilisateurs et groupes

### groupadd

- Créer un groupe <code> groupadd web #en précisant le gid (supérieur à 500, sinon réservé au système) <code> groupadd -g gid groupe </code>

### useradd

- Créer un utilisateur, fichier de conf : /etc/adduser.conf, fichier squelette (contenant une arborescence type): /etc/skel <code bash> ## Créer l'utilisateur ulysses avec un commentaire, un répertoire home spécial ##+ une date d'expiration et un délai de 8 jours avant la désactivation, ##+ appartenant au groupe principal ulysses et aux groupes secondaires web et ##+ formateurs. L'option -m permet de créer le répertoire home si il n'existe pas. useradd -c "ceci est mon premier compte" -d /home/formation/ulysses -e 20081015 -f 8 -g ulysses -G

web,formateurs -m ulyssse

# En précisant l'uid (supérieur à 100 sinon réservé au système)

useradd -u uid login </code>

## passwd

- Changer le mot de passe d'un utilisateur <code> #en tant que root passwd <nomutilisateur> #en tant qu'utilisateur passwd #Changer le mot de passe dans un script (pour des créations en série) echo compte:motdepasse | chpasswd </code>

## usermod / userdel

- Ajouter un groupe à un utilisateur

usermod -a -G nouveau-groupe nom\_utilisateur

- Supprimer un compte utilisateur avec son répertoire personnel et ses fichiers

userdel -r nom\_utilisateur

## w, who, id

- w liste les utilisateurs et ce qu'ils font
- who liste les utilisateurs
- id indique les groupes d'un utilisateurs.

## liens

- [Toutes les commandes sur tldp](#)
- [commandes](#)
- [tutoriel](#)
- [memento-commandes.pdf - Aide mémoire](#)

From:  
<https://wiki.pielo.net/> - **Pielo.net - Wiki**

Permanent link:  
<https://wiki.pielo.net/aide-memoire-bash-et-commandes-systemes?rev=1765614330>

Last update: **2025/12/13 08:25**

