

Quelques listes de commandes

Commandes liées au manuel

- lecture du manuel d'un programme, d'un fichier spécial ou de configuration, d'une notion : `man <programme>`, `man <fichier>`, `man <notion>`
- rechercher un mot clef dans les descriptifs des manuels : `man -k <mot_clef>`
- notion de sections, `man man` pour obtenir la liste
- manuel d'une section particulière : `man <section> <commande>` ou `man <commande>.<section>`

Commandes de navigation dans l'arborescence des répertoires

- créer un répertoire : `mkdir` (option `-p`)
- changer de répertoire : `cd`
- savoir dans quel répertoire on se trouve : `pwd` (voir l'option `-p`)
- lister le contenu d'un répertoire : `tree` et `ls` (options `-i`, `-l`, `-a`, `-h`)
- supprimer un fichier : `rm` (options `-d`, `-r`, `-f`)
 - supprimer un répertoire vide en toute sécurité : `rm -d` (ou son équivalent `rmdir`)
 - supprimer un répertoire et tout son contenu sans warning : `rm -rf attention`
- créer un fichier vide ou faire comme s'il venait d'être modifié : `touch`
- copier un fichier ou un répertoire, faire un lien (hardlink) : `cp` (options `-r`, `-a`, `-l`)
- copier un certain nombre d'octets d'un fichier : `dd`
- créer un lien symbolique : `ln -s`
- afficher le chemin pointé par un lien symbolique : `readlink`
- afficher le chemin absolu d'un fichier : `realpath`
- déplacer un fichier : `mv`
- déterminer le format d'un fichier (ascii, utf-8, html, pdf, png, odt, ...) : `file` (options `--mime*`, `-b`)
- itérer dans l'arborescence : `find` (nombreuses options `-exec`, `-execdir`, `-empty`, `-newer`, `-uid`, `-name`, `-perm`, `-type`, ...)

Commandes système de fichiers

- lister les systèmes de fichiers montés et leurs points de montage : `findmnt`
- installer un système de fichier dans un block device : `mkfs`
- vérifier l'intégrité d'un système de fichiers : `fsck`
- monter/démonter un système de fichiers sur un répertoire de l'arborescence : `mount`, `umount`
- voir les attributs d'inode d'un fichier : `stat`
- indiquer l'espace occupé par les systèmes de fichiers : `df` (options `-h`, `-i`)
- évaluer l'espace disque occupé par des fichiers : `du` (options `-s`, `-h`)

Gestion des périphériques de type bloc

- lister les périphériques de type bloc : `lsblk`
- partitionner un périphérique de type bloc : `parted`
- chiffrer un périphérique de type bloc : `cryptsetup` (options `luksFormat`, `open`, `close`)
- fusionner/redécouper/etc des périphériques de type bloc : `lvm`
- raid : `mdadm`
- créer un périphérique de type bloc à partir d'un fichier régulier : `losetup` (options `-f`, `-d`)

Commandes de manipulation de texte

Ces commandes permettent de manipuler du texte et se combinent très bien entre elles.

- éditer un fichier texte : **vim** ou **emacs**
- envoyer du texte dans la sortie standard : **echo**
- envoyer le contenu d'un fichier dans la sortie standard, concaténer plusieurs fichiers : **cat**
- afficher le contenu d'un fichier texte et pouvoir y naviguer (cf cours introduction, raccourcis proches de vim) : **less**
- rechercher une chaîne de caractère dans un fichier : **grep** (options **-v**, **-c**, **-i**)
- afficher les différences entre deux fichiers texte : **diff** (voir aussi **vimdiff**)
- **patch**
- trier les lignes d'un fichier texte dans l'ordre alphabétique : **sort** (option **-n** pour tri numérique)
- éliminer les lignes consécutives répétées : **uniq**
- numéroter les lignes d'un fichier texte : **nl**
- Concaténer et afficher des fichiers à l'envers (de la dernière ligne à la première) : **tac**
- Inverser les lignes au niveau des caractères : **rev**
- **join**
- **paste**
- récupérer des colonnes correspondant à certains champs d'un fichier : **cut**
- **column**
- **colrm**
- afficher les premières lignes d'un fichier texte : **head** (option **-n** pour spécifier le nombre de lignes)
- afficher les dernières lignes d'un fichier texte : **tail** (option **-n** pour spécifier le nombre de lignes)
- compter le nombre de lignes/mots/octets/caractères d'un fichier texte : **wc** (option **-l**)
- convertir des caractères : **tr**
- changer l'encodage d'un fichier texte : **iconv**
- couper chaque ligne de texte à une longueur donnée : **fold** (options **-s**, **-w**)
- substitution de motifs, suppressions de lignes : **sed**
- traitement de fichiers structurés en colonnes : **awk**
- afficher les portions imprimables d'un fichier binaire : **strings**
- afficher le contenu d'un fichier binaire en hexadécimal (4 bits par caractère) : **hexdump** (option **-C**)
- afficher le contenu d'un fichier binaire en octal (3 bits par caractère) : **od**

Commandes de gestion des users et groups

- changer les droits d'un fichier : **chmod** (option **-R**)
- changer l'utilisateur et/ou le groupe propriétaire d'un fichier : **chown** (option **-R**)
- créer un utilisateur : **adduser**
- supprimer un utilisateur : **deluser**
- créer un groupe : **addgroup**
- ajouter un utilisateur à un groupe : **adduser**

Commandes de gestion des processus

- lister les processus : **ps**, **pstree**
 - appels classiques : **ps aux** (BSD), **ps -eF** (POSIX)
 - sélection des colonnes : option **-eo** (colonnes utiles : **pid,ppid,pgid,uid,gid,stat,comm[and]**)
 - vue arborescente : option **--forest** ou **-f**
- visualiser les processus et les trier selon leur consommation : **htop**
- lister les PID des processus correspondant à nom de commande : **pidof**
- quels processus accèdent à quels fichiers : **fuser**, **lsof**
- informations contenues dans **/proc** : **man 5 proc**
- envoyer un signal à un ou plusieurs processus : **kill** (options **-l**, **-15**, **-9**)
- gestion des services : **systemd** (en tant que **root**)
 - démarrer un service : **systemctl start <service>**
 - éteindre un service : **systemctl stop <service>**
 - redémarrer un service : **systemctl restart <service>**
 - voir l'état d'un service, et la fin de son fichier de log : **systemctl status <service>**
 - activer le démarrage automatique d'un service : **systemctl enable <service>**

- désactiver le démarrage automatique d'un service : `systemctl disable <service>`
- voir si le démarrage automatique d'un service est activé : `systemctl is-enabled <service>`
- recharger la configuration d'un service : `systemctl reload <service>`

Commandes réseau

- gestion de la couche liaison : `ip link`
- gestion de l'inter-couche ARP (correspondance MAC-IP) : `ip neighbour`
- gestion de la couche réseau : `ip address`
- gestion de la table de routage : `ip route`
- scanner des ports d'une machine distante (voir sur lesquels des services écoutent) : `nmap`
- établir une connexion entre 2 extrémités (TCP, UDP, socket UNIX, fichier, terminal, ...) : `socat`
- observer la route empruntée par les paquets vers une machine distante `traceroute` (options `--tcp`, `--udp`)
- tester l'accessibilité d'une machine distante : `ping`
- lister les sockets (TCP, UDP, mais aussi UNIX) : `ss`

Commandes d'administration à distance

- shell distant `ssh` (man 5 `ssh_config` pour les options)
- générer une paire de clefs SSH : `ssh-keygen`
- envoyer une clef publique SSH vers une machine distante : `ssh-copy-id`
- synchronisation de répertoires entre machines locale et distante : `rsync` (options `-av`)
- copie de fichiers entre machines locale et distante : `scp`
- protocole d'échange de fichiers à travers SSH : `SFTP`
- système de fichiers permettant de monter un répertoire distant sur un point de montage local : `sshfs`
- gestionnaire d'archives : `tar` (regex options `(c|x|t)z?v?f`)
- téléchargement de fichiers sur le web : `wget` (nombreuses options)

Commandes de gestion des paquets (distributions de la famille Debian)

- à l'échelle de la distribution (DAG des dépendances des paquets) : `apt`
 - récupérer la liste des paquets disponibles avec leurs numéros de version, leurs dépendances, leurs descriptifs : `apt update`
 - télécharger et installer les nouvelles versions des paquets installés en tenant compte de leurs dépendances : `apt upgrade`
 - lister les paquets dont le descriptif contient un mot clef : `apt search`
 - montrer le descriptif d'un paquet : `apt show`
 - installer un paquet en tenant compte de ses dépendances : `apt install`
 - équivalent d'`apt` à utiliser dans les scripts ou les tubes : `apt-get` (moins joli mais plus prévisible)
- à l'échelle d'un paquet : `dpkg`
 - installer un paquet téléchargé manuellement `dpkg -i`
 - voir les fichiers fournis par un paquet (installé) : `dpkg -L`
 - voir quel paquet a fourni un fichier : `dpkg -S`

Commandes quasi-synonymes

- `apropos` ~ `man -k`
- `rmdir` ~ `rm -d`
- `link` ~ `cp -l` ~ `ln`
- `dlocate` ~ `dpkg -S`

Options transverses à plusieurs commandes

- `-R` (parfois `-r`) ou `--recursive` : récursivité, appliquer l'opération à la sous-arborescence du répertoire spécifié ou courant
- `-no-clobber` : ne pas écraser un fichier existant
- `--dereference` (ou `--no-dereference`), appliquer la commande au fichier pointé par le lien symbolique (ou pas)

- `-v, --verbose -vv`
- `-h, --human-readable` : lisible par un être humain, par exemple indiquer les grandes tailles dans l'unité adaptée (KiB, MiB, GiB, TiB) plutôt qu'en octets.
- `-F` décoration qui termine les noms de fichiers par `/` pour les répertoires, `*` pour les fichiers réguliers exécutables, `@` pour les liens symboliques, etc