Commandes linux-debian 11+

Commandes Linux — Debian 11+ (et fed unbun)

1. Navigation & fichiers de base

`pwd` — Où suis-je ?

Affiche le dossier dans lequel tu te trouves actuellement.

```
pwd
# Exemple de résultat : /home/tonnom
```

`ls` — Lister les fichiers

Affiche le contenu d'un dossier.

```
ls              # liste simple
ls -l           # liste détaillée (permissions, taille, date)
ls -la          # idem + fichiers cachés (ceux qui commencent par un point)
ls -lh          # tailles lisibles (Ko, Mo...)
ls /var/log     # liste le contenu d'un autre dossier
```

`cd` — Se déplacer dans les dossiers

```
cd /etc             # aller dans /etc
cd ..               # remonter d'un niveau
cd ~                # aller dans ton dossier personnel
cd -                # revenir au dossier précédent
```

```
cd /home/tonnom/docs # chemin absolu
```

`mkdir` — Créer un dossier

```
mkdir mondossier                # crée un dossier
mkdir -p /home/user/a/b/c       # crée tous les dossiers intermédiaires si besoin
```

`rm` — Supprimer des fichiers ou dossiers

Attention : pas de corbeille, c'est définitif !

```
rm fichier.txt           # supprime un fichier
rm -r mondossier/        # supprime un dossier et tout son contenu
rm -rf mondossier/       # force la suppression sans confirmation (dangereux !)
rm -i fichier.txt        # demande confirmation avant de supprimer
```

`cp` — Copier

```
cp fichier.txt copie.txt            # copie un fichier
cp -r dossier/ destination/         # copie un dossier entier
cp fichier.txt /home/user/Bureau/   # copie vers un autre emplacement
```

`mv` — Déplacer ou renommer

```
mv fichier.txt /tmp/                # déplace un fichier
mv ancien_nom.txt nouveau_nom.txt   # renomme un fichier
mv dossier1/ /home/user/dossier2/   # déplace un dossier
```

`touch` — Créer un fichier vide

```
touch notes.txt         # crée un fichier vide (ou met à jour sa date)
```

`cat` — Afficher le contenu d'un fichier

```
cat fichier.txt          # affiche tout le fichier
cat -n fichier.txt       # affiche avec numéros de lignes
```

`less` — Lire un fichier page par page

```
less /var/log/syslog     # parcourir avec les flèches, quitter avec q
```

`head` / `tail` — Début ou fin d'un fichier

```
head fichier.txt         # affiche les 10 premières lignes
head -n 20 fichier.txt   # affiche les 20 premières lignes
tail fichier.txt         # affiche les 10 dernières lignes
tail -f /var/log/syslog  # suit le fichier en temps réel (logs live)
```

`find` — Chercher des fichiers

```
find / -name "fichier.txt"          # cherche partout
find /home -name "*.log"            # cherche les fichiers .log dans /home
find /etc -type f -name "*.conf"    # cherche uniquement des fichiers
find /tmp -type d                   # cherche uniquement des dossiers
```

`grep` — Chercher du texte dans un fichier

```
grep "erreur" fichier.txt           # cherche le mot "erreur"
grep -i "erreur" fichier.txt        # insensible à la casse
grep -r "erreur" /var/log/          # cherche dans tous les fichiers du dossier
grep -n "erreur" fichier.txt        # affiche les numéros de lignes
```

---

2. Gestion des paquets (installer des logiciels)

Debian / Ubuntu — `apt`

```
apt update                      # met à jour la liste des paquets disponibles
apt upgrade                     # installe les mises à jour disponibles
apt install nom_du_paquet       # installe un logiciel
apt remove nom_du_paquet        # désinstalle un logiciel
apt autoremove                  # supprime les paquets inutiles
apt search mot_clé              # cherche un paquet
apt show nom_du_paquet          # infos sur un paquet
apt list --installed            # liste les paquets installés
```

Fedora / RHEL — `dnf`

```
dnf check-update                # vérifie les mises à jour
dnf upgrade                     # installe les mises à jour
dnf install nom_du_paquet       # installe un logiciel
dnf remove nom_du_paquet        # désinstalle un logiciel
dnf autoremove                  # supprime les paquets inutiles
dnf search mot_clé              # cherche un paquet
dnf info nom_du_paquet          # infos sur un paquet
dnf list installed              # liste les paquets installés
```

---

3. Gestion des utilisateurs

`adduser` / `useradd` — Créer un utilisateur

Debian → `adduser` (plus simple, interactif) :

```
adduser tonnom
# Il te pose les questions : mot de passe, nom complet, etc.
```

Fedora → `useradd` (moins interactif) :

```
useradd tonnom              # crée l'utilisateur
passwd tonnom               # définit le mot de passe séparément
```

`passwd` — Changer un mot de passe

```
passwd              # change ton propre mot de passe
passwd tonnom       # change le mot de passe d'un autre utilisateur (en root)
```

`userdel` — Supprimer un utilisateur

```
userdel tonnom          # supprime l'utilisateur mais garde son dossier home
userdel -r tonnom       # supprime l'utilisateur ET son dossier home
```

`usermod` — Modifier un utilisateur

```
usermod -l nouveau_nom ancien_nom      # renomme un utilisateur
usermod -d /home/nouveau_home tonnom   # change le dossier home
usermod -aG sudo tonnom                # ajoute l'utilisateur au groupe sudo (Debian)
usermod -aG wheel tonnom               # ajoute l'utilisateur au groupe wheel (Fedora)
usermod -L tonnom                      # verrouille le compte
usermod -U tonnom                      # déverrouille le compte
```

`id` — Voir les infos d'un utilisateur

```
id                  # affiche ton uid, gid et groupes
id tonnom           # affiche les infos d'un autre utilisateur
```

`who` / `w` — Qui est connecté ?

```
who     # liste les utilisateurs connectés
w       # liste + ce qu'ils font
```

`su` / `sudo` — Changer d'utilisateur ou devenir root

```
su -                    # devient root (demande le mot de passe root)
su - tonnom             # devient un autre utilisateur
sudo commande           # exécute une commande en tant que root
sudo -i                 # ouvre un shell root complet
```

> **Debian** : root est souvent désactivé, on utilise `sudo`.
> **Fedora** : pareil, le groupe `wheel` donne accès à `sudo`.

---

4. Permissions et droits

Comprendre les permissions avec `ls -l`

```
-rwxr-xr--  1  user  groupe  taille  date  fichier
|└─┘└─┘└─┘
|  |    |   └── autres utilisateurs (r = lire)
|  |    └────── groupe (r-x = lire + exécuter)
|  └─────────── propriétaire (rwx = tout)
└──────────────── type : - fichier, d dossier, l lien
```

r = read (lire) = 4
w = write (écrire) = 2
x = execute (exécuter) = 1

`chmod` — Modifier les permissions

Méthode numérique :

```
chmod 755 script.sh      # propriétaire : rwx | groupe : r-x | autres : r-x
chmod 644 fichier.txt    # propriétaire : rw- | groupe : r-- | autres : r--
chmod 700 secret.sh      # propriétaire : rwx | personne d'autre
```

```
chmod 777 fichier        # tout le monde peut tout faire (déconseillé !)
```

Méthode symbolique :

```
chmod +x script.sh       # ajoute le droit d'exécution
chmod -w fichier.txt     # retire le droit d'écriture
chmod u+x script.sh      # ajoute exécution pour le propriétaire (u=user)
chmod g-w fichier.txt    # retire écriture pour le groupe (g=group)
chmod o-r fichier.txt    # retire lecture pour les autres (o=others)
```

`chown` — Changer le propriétaire

```
chown tonnom fichier.txt             # change le propriétaire
chown tonnom:mongroupe fichier.txt   # change propriétaire ET groupe
chown -R tonnom /home/tonnom/        # change récursivement (tout le dossier)
```

`chgrp` — Changer le groupe

```
chgrp mongroupe fichier.txt          # change le groupe d'un fichier
chgrp -R mongroupe /var/www/         # change récursivement
```

---

5. Gestion des groupes

`groupadd` — Créer un groupe

```
groupadd devs          # crée le groupe "devs"
```

`groupdel` — Supprimer un groupe

```
groupdel devs
```

`gpasswd` — Gérer les membres d'un groupe

```
gpasswd -a tonnom devs      # ajoute tonnom au groupe devs
gpasswd -d tonnom devs      # retire tonnom du groupe devs
```

`groups` — Voir les groupes d'un utilisateur

```
groups              # tes groupes
groups tonnom       # groupes d'un autre utilisateur
```

---

6. Processus et système

`ps` — Voir les processus en cours

```
ps aux              # tous les processus de tous les utilisateurs
ps aux | grep nginx # filtrer pour un processus précis
```

`top` / `htop` — Moniteur de ressources interactif

```
top                 # moniteur basique (q pour quitter)
htop                # version améliorée (à installer si absent)
```

`kill` — Tuer un processus

```
kill 1234           # envoie un signal d'arrêt au processus PID 1234
kill -9 1234        # force la fermeture (dernier recours)
pkill nginx         # tue par nom de processus
```

`df` — Espace disque

```
df -h               # espace disque de toutes les partitions (lisible)
df -h /home         # espace disque d'une partition précise
```

`du` — Taille d'un dossier

```
du -sh /var/log/    # taille totale d'un dossier
du -sh /*           # taille de chaque dossier à la racine
```

`free` — Mémoire RAM

```
free -h             # affiche la RAM utilisée/libre en lisible
```

`uname` — Infos système

```
uname -a            # toutes les infos (kernel, architecture...)
uname -r            # version du kernel uniquement
```

`uptime` — Depuis quand le système tourne

```
uptime
```

---

7. Réseau

`ip` — Configuration réseau

```
ip a                        # voir les interfaces réseau et IPs
ip r                        # voir les routes (passerelle...)
ip link show                # état des interfaces
```

`ping` — Tester la connectivité

```
ping 8.8.8.8                # ping Google DNS
ping -c 4 google.com        # envoie 4 paquets puis s'arrête
```

`ss` — Voir les connexions réseau

```
ss -tulnp                   # ports en écoute avec les processus associés
```

`curl` — Faire des requêtes HTTP

```
curl https://example.com            # affiche le contenu d'une URL
curl -O https://example.com/fichier # télécharge un fichier
curl -I https://example.com         # affiche les en-têtes HTTP
```

`wget` — Télécharger des fichiers

```
wget https://example.com/fichier.zip
wget -O nom_local.zip https://example.com/fichier.zip
```

`hostname` — Nom de la machine

```
hostname                    # affiche le nom de la machine
hostnamectl                 # infos détaillées + changer le nom
hostnamectl set-hostname nouveau-nom
```

8. Services (systemd)

(nginx = exemple)

```
systemctl status nginx           # état d'un service
systemctl start nginx            # démarrer un service
systemctl stop nginx             # arrêter un service
systemctl restart nginx          # redémarrer un service
systemctl reload nginx           # recharger la config sans couper
systemctl enable nginx           # démarrer automatiquement au boot
systemctl disable nginx          # ne plus démarrer au boot
systemctl enable --now nginx     # activer ET démarrer immédiatement
systemctl list-units --type=service  # lister tous les services
journalctl -u nginx              # voir les logs d'un service
journalctl -u nginx -f           # suivre les logs en direct
journalctl -xe                   # voir les derniers logs système
```

---

9. Éditeurs de texte en CLI

`nano` — Simple et accessible

```
nano fichier.txt
# Ctrl+O = sauvegarder | Ctrl+X = quitter | Ctrl+K = couper une ligne
```

`vim` — Puissant mais complexe (déconseillé)

```
vim fichier.txt
# i = passer en mode insertion
# Echap = revenir en mode commande
# :w = sauvegarder
# :q = quitter
# :wq = sauvegarder et quitter
# :q! = quitter sans sauvegarder
```

---

10. SSH — Connexion à distance

```
ssh user@ip_ou_domaine              # se connecter à une machine distante
ssh -p 2222 user@ip                 # connexion sur un port personnalisé
ssh-keygen -t ed25519               # générer une paire de clés SSH
ssh-copy-id user@ip                 # copier sa clé publique sur le serveur
scp fichier.txt user@ip:/tmp/       # copier un fichier vers un serveur
scp user@ip:/tmp/fichier.txt .      # copier un fichier depuis un serveur
```

---

11. Commandes utiles diverses

```
history                         # voir l'historique des commandes
!!                              # relancer la dernière commande
!commande                       # relancer la dernière commande commençant par "commande"
clear                           # effacer le terminal
echo "texte"                    # afficher du texte
```

```
echo $HOME                    # afficher une variable d'environnement
env                           # voir toutes les variables d'environnement
which python3                 # trouver où est installé un programme
whereis nginx                 # trouver les fichiers liés à un programme
man ls                        # afficher le manuel d'une commande (q pour quitter)
--help                        # option disponible sur presque toutes les commandes
alias ll='ls -la'             # créer un raccourci de commande
```

Résumé des différences Debian vs Fedora

Action	Debian	Fedora
Installer un paquet	`apt install`	`dnf install`
Mettre à jour	`apt update && apt upgrade`	`dnf upgrade`
Chercher un paquet	`apt search`	`dnf search`
Supprimer un paquet	`apt remove`	`dnf remove`
Créer un utilisateur	`adduser` (interactif)	`useradd` + `passwd`
Groupe sudo	`sudo`	`wheel`
Logs système	`journalctl`	`journalctl`
Pare-feu	`ufw`	`firewalld`

Revision #11
Created 2026-05-12 12:33:18 UTC by Admin
Updated 2026-10-07 15:57:58 UTC by Admin