

Commande	Description
clear	Vider le terminal Utilisez la commande clear pour effacer le contenu de l'écran. clear Vous obtiendrez un terminal vide avec une invite de commande (prompt). Au lieu d'utiliser cette commande, vous pouvez également vider votre terminal à l'aide de la combinaison de touches [CTRL] + [L].
exit	Fin de session La commande exit termine la session en cours et ferme le terminal. exit Vous pouvez également utiliser la combinaison de touches [CTRL] +[D].
help	Afficher une liste des commandes shell Utilisez la commande help pour afficher une liste de toutes les commandes shell intégrées. help Utilisez help directement avec une commande shell pour afficher une courte description de la commande correspondante. help COMMANDE
history	Emissions des fichiers de l'historique Dans le bash, les 500 dernières commandes que vous avez saisies (par ligne de commande) sont enregistrées dans history. Cette fonction peut aider à la saisie des commandes. Elle permet en effet d'effectuer une recherche dans la liste des commandes précédemment saisies (grâce aux touches fléchées) et de les exécuter à nouveau en confirmant avec la touche Entrée. L'historique peut être recherché par mots-clés à l'aide de la combinaison [CTRL] +[R]. Vous avez également la possibilité d'afficher la liste complète numérotée dans le terminal. Utilisez la commande history sans options ni arguments : history Si vous voulez filtrer la sortie, combinez history via le pipe avec le programme en ligne de commande grep (voir les options de recherche) et un mot-clé. History grep MOTDERECHERCHE

***** Les commandes de bases *****

apropos	Rechercher dans le manuel Utilisez apropos pour rechercher des mots-clés dans les titres de pages et les descriptions des pages de manuel de votre système d'exploitation. L'utilitaire en lignes de commande affiche toutes les correspondances, y compris une brève description dans le terminal. Orientez-vous vers le schéma suivant : apropos [OPTIONS] MOTDERECHERCHE La commande prend en charge différentes options. Limitez la recherche aux correspondances exactes avec l'option -e (--exact) ou utilisez des caractères de remplacement (-w'*MOTDERECHERCHE') et des expressions régulières (-r).
info	Accéder aux pages d'information GNU La commande info peut être utilisée pour produire des pages d'information GNU sur un sujet spécifique. Dans la plupart des cas, il s'agit de pages de manuel accessibles via man,

mais à la différence de celles-ci, des liens vont faciliter la navigation dans le manuel.

Utilisez la syntaxe suivante pour accéder à une page d'information GNU :

info [OPTION] SUJET

Un appel sans option et sans sujet spécifié vous amène au menu principal des pages d'information GNU.

man Utiliser le manuel

La commande man ouvre les pages de manuel de votre distribution Linux (Man-Pages) directement dans le terminal.

Utilisez le schéma suivant pour accéder à une page de manuel :

man [OPTION] SUJET

Les pages man Linux sont divisées en 10 domaines :

- Commandes utilisateur
- Appel système
- Fonctions du langage de programmation C
- Fichiers spécifiques
- Formats de fichier
- Jeux
- Divers
- Commandes pour l'administration du système
- Fonctions cœur (noyau Linux)
- Nouvelles commandes

Par exemple, si vous voulez ouvrir la page de manuel pour une commande Linux spécifique, utilisez-la avec le nom de la commande :

man clear

Vous pouvez également limiter la recherche en saisissant le numéro de domaine :

man 1 clear

Dans les deux cas, la page d'aide pour la commande clear s'ouvre. Utilisez la touche [Q] pour la fermer et revenir au prompt.

Pour rechercher des mots-clés dans les pages du manuel Linux, vous pouvez utiliser apropos.

pinfo Accéder aux pages d'information à la manière Lynx

pinfo est une variante du programme en lignes de commande info, centrée sur le navigateur Lynx et qui affiche des pages d'information avec des liens surlignés.

Utilisez pinfo selon le même schéma que la commande info :

pinfo [OPTIONS] SUJET

whatis Recherche par mot-clé dans les pages du manuel

Le programme en lignes de commande whatis est utilisé pour rechercher des mots-clés dans les pages de manuel. Appelez le programme avec le terme de votre choix pour rechercher des correspondances exactes dans le manuel de votre système d'exploitation. S'il trouve un équivalent, whatis émet une courte description dans le terminal.

whatis [OPTIONS] MOTDERECHERCHE

whatis prend également en charge les mots de remplacement (-w '*MOTDERECHERCHE') et les expressions régulières (-r).

cd Navigation dans l'arborescence des répertoires

La commande cd (pour change directory) est utilisée pour la navigation dans l'arborescence des répertoires.

Voici la syntaxe de la commande :

cd [OPTION] REPERTOIRE

Si aucun répertoire cible n'est spécifié, cd passe automatiquement au répertoire home de l'utilisateur.

Si cd est utilisé avec un signe moins (-), il passe au répertoire précédent.

chroot

Exécuter le programme dans un nouveau répertoire racine

La commande chroot (abréviation de change + root) est utilisée pour exécuter une commande dans un autre répertoire racine. Par exemple, chroot peut isoler les programmes critiques du reste du système de fichiers. On parle alors de chroot jail.

L'utilisation du programme nécessite des droits root. Il est basé sur le schéma suivant :
chroot REPERTOIRE COMMANDE

ls

Liste du contenu de l'annuaire

La commande ls (pour liste) sert à afficher le contenu d'un répertoire (les noms de tous les fichiers et dossiers du répertoire spécifié).

La syntaxe de la commande est la suivante :

ls [OPTIONS] REPERTOIRE

Si ls est utilisé sans spécification de répertoire, la commande liste le contenu du répertoire courant.

A l'aide des options supplémentaires, vous pourrez définir quelles informations à afficher et comment.

option	description
ls -a	list all files including hidden file starting with '.'
ls --color	colored list [=always/never/auto]
ls -d	list directories - with '/'
ls -F	add one char of */=>@ to enteries
ls -i	list file's inode index number
ls -l	list with long format - show permissions
ls -la	list long format including hidden files
ls -lh	list long format with readable file size
ls -ls	list with long format with file size
ls -r	list in reverse order
ls -R	list recursively directory tree
ls -s	list file size
ls -S	sort by file size
ls -t	sort by time & date
ls -X	sort by extension name

mkdir

Créer un répertoire

La commande mkdir signifie make directory. Il permet aux utilisateurs de Linux d'établir de nouveaux répertoires.

Utilisez la syntaxe suivante pour créer un nouveau répertoire dans le répertoire courant :

mkdir [OPTION] NOMREPERTOIRE

Par ailleurs, vous pouvez créer plusieurs répertoires en même temps en séparant leurs noms par un espace :

mkdir [OPTION] NOMREPERTOIRE1 NOMREPERTOIRE2

Si vous voulez en créer un dans un répertoire cible spécifique, spécifiez le chemin absolu ou relatif vers le répertoire.

mkdir /home/user/Desktop/test

mkdir ../Desktop/test

Dans les deux exemples, le répertoire test est créé dans le répertoire Desktop.

mkdirhier Créer une hiérarchie de répertoires
Avec mkdirhier, vous pouvez créer des hiérarchies de répertoires entiers avec une seule ligne de commande :
mkdirhier [OPTION] /home/user/repertoire1/repertoire2/repertoire3
Si repertoire1 et repertoire2 existent déjà, mkdirhier crée simplement repertoire3.
Autrement, ce sont les trois répertoires qui sont créés.

pwd Emission du nom de répertoire
Utilisez pwd (abréviation de print working directory) pour faire afficher le nom du répertoire de travail actuel.
Voici la syntaxe de la commande :
pwd [OPTIONS]

rmdir / rm Supprimer l'annuaire
Si vous voulez supprimer un répertoire spécifique, utilisez la commande rmdir (remove directory) selon la syntaxe suivante :
rmdir [OPTION] REPERTOIRE
Avec rmdir, vous ne pouvez supprimer que les répertoires vides. Pour supprimer un répertoire comprenant des fichiers et sous-dossiers, utilisez la commande rm (remove) avec l'option -r.
rm -R (enlèvement récursif!!!)
rm fichier
rm -i fichier (interactivement, avec demande de confirmation)
rm -f fichier (avec force, sans demande de confirmation)
rm -r fichier (avec récursivité, avec les sous répertoires)
rm -rf dossier (supprime le répertoire et tout son contenu, sans confirmation)

tree Supprimer des répertoires dans l'arborescence
Alors que ls ne liste que le contenu des répertoires, la hiérarchie complète peut être affichée de manière récursive sous la forme d'une arborescence grâce au programme en ligne de commande tree.
L'appel est effectué selon la syntaxe suivante :
tree [OPTIONS] [REPERTOIRE]

find Effectuer une recherche dans le système de fichier
find est un programme en ligne de commande utilisé pour rechercher des fichiers.
L'appel de programme est basé sur la syntaxe suivante :
find [OPTIONS] [REPERTOIRE] [CONDITIONDERECHERCHE][ACTIONS]
Le répertoire spécifié est le répertoire de départ de la recherche. La recherche est alors effectuée dans ce répertoire ainsi que ses sous-répertoires. Si vous ne spécifiez pas de répertoire, find lance la recherche à partir de votre répertoire courant.
Les options vous permettent de définir des critères de recherche et des actions. L'action par défaut est -print.
Les critères de recherche courants sont le nom du fichier (-name NOMFICHIER[SUFFIXE]), un nom d'utilisateur (-user NOMUTILISATEUR), la taille du fichier (-size n[cwbkMG]), le jour de l'accès au fichier (-atime [+]-n) ou le jour des modifications effectuées (-mtime [+]-n).
Les méta-caractères et caractères de remplacement peuvent être utilisés lors de la recherche de noms de fichiers. Mettez-les entre guillemets pour empêcher qu'ils ne soient interprétés par le Shell.
Exemple :
find /tmp -name "*.odt" -mtime -3 -size +20k
Le répertoire de départ est /tmp. Le programme en ligne de commande find émet sur la sortie standard tous les fichiers avec l'extension .odt qui sont plus grands que 20k et qui ont été modifiés pour la dernière fois il y a moins de 3 jours.
Pour plus d'options sur la commande find, reportez-vous à la page d'aide du programme.

grep

Rechercher dans les fichiers texte

La commande grep (abréviation de global regular expression print) peut être utilisée pour faire une recherche dans les fichiers texte (par exemple les fichiers log). N'importe quelle chaîne de caractères ou expression régulière peut être utilisée comme modèle de recherche.

Utilisez grep selon la syntaxe suivante :

grep [OPTIONS] MODELEDERECHERCHE [FICHIER(S)]

Si grep rencontre une chaîne de caractères correspondant au modèle de recherche, le numéro de ligne est édité dans le terminal avec le nom du fichier.

En principe, grep est appliqué à tous les fichiers du répertoire courant. L'option -r permet une recherche récursive dans les sous-répertoires.

Exemple :

grep motif fichier

grep -i motif fichier (sans tenir compte de la casse)

grep -c motif fichier (en comptant les occurrences)

grep -v motif fichier (inverse la recherche, en excluant le "motif")

locate

Recherche dans l'index des fichiers

Le programme en ligne de commande locate permet également de rechercher des fichiers via le terminal. Contrairement à find, cependant, ce n'est pas le répertoire de fichiers qui est fouillé, mais plutôt une base de données dédiée à cette fin et régulièrement mise à jour. Par conséquent, locate permet d'obtenir des résultats de recherche beaucoup plus rapidement que find.

Pour rechercher un fichier spécifique dans la base de données, locate est appelé selon le schéma suivant :

locate MODELEDERECHERCHE

Le modèle de recherche peut contenir des méta-caractères tels que des caractères de remplacement (*). Mettez-les entre guillemets pour empêcher une interprétation par le Shell.

Dans l'exemple suivant, locate émet tous les fichiers avec l'extension .png.

locate "*.png"

La commande locate est sensible à la casse. Pour ignorer les différences de majuscules et minuscules dans la recherche, utilisez l'option -i.

Le fichier /var/lib/locatedb sert de base de données pour l'indexation des fichiers. Cette dernière contient une liste de tous les fichiers du système de fichiers à un moment donné et doit être mis à jour régulièrement. Pour ce faire, utilisez la commande updatedb.

tre-agrep

Recherche imprécise dans les fichiers texte

tre-agrep est également utilisé pour rechercher sur la base de modèles de recherche des strings dans les fichiers texte. Contrairement à grep, non seulement les correspondances exactes sont émises, mais aussi les correspondances moins précises telles qu'avec des inversions de lettres ou autres coquilles. Le programme est basé sur la bibliothèque TRE.

La syntaxe de tre-agrep correspond à celle de la commande grep :

tre-agrep [OPTIONS] MODELEDERECHERCHE FICHIER(S)

Dans les options, vous pouvez définir le nombre maximum d'erreurs autorisées. Dans l'exemple suivant, un seul écart est toléré par rapport au modèle de recherche.

tre-agrep -1 'Linux' test.txt

tre-agrep édite toutes les lignes du fichier texte test.txt contenant des mots qui correspondent au modèle de recherche ou qui en diffèrent au maximum d'une lettre : par exemple Linux.

updatedb

Mettre à jour l'index des fichiers

Une recherche via locate ne fonctionnera de manière fiable que si le fichier /var/lib/locatedb est continuellement tenu à jour. La commande updatedb permet donc de mettre à jour la base de données manuellement. Notez que vous avez besoin pour ce faire de disposer des droits root:

updatedb

whereis Rechercher des pages binaires, du code source ou les pages manuelles des programmes
Avec la commande whereis, vous pouvez localiser les fichiers binaires, de code source ou les manuels des programmes sélectionnés.
La syntaxe générale de la commande est :
whereis [OPTIONS] PROGRAMME
Exemple :
whereis firefox
En tant qu'output, whereis émet dans le terminal les chemins d'accès aux fichiers trouvés en les séparant par des espaces :
firefox: /usr/bin/firefox/ usr/lib/firefox/ usr/share/man/man1/firefox.1.gz
À l'aide des options, vous pouvez limiter la recherche à des types de fichiers ou des répertoires spécifiques.

which Retrouver des fichiers binaires à partir des programmes
Si vous voulez retrouver le fichier binaire d'un programme, utilisez la commande which selon le schéma ci-dessous, qui permet d'afficher le chemin dans le terminal.
which [OPTIONS] PROGRAMME
Exemple :
which firefox
Output :
/usr/bin/firefox
En mode par défaut which émet le premier fichier qui est trouvé. Utilisez l'option -a pour afficher tous les fichiers qui correspondent au critère de recherche.

finger Appeler des informations sur les utilisateurs
Le programme en ligne de commande finger est utilisé pour récupérer des informations sur les utilisateurs. La commande finger est donnée avec le nom de l'utilisateur en question :
finger [Options] [NOMUTILISATEUR]
Le résultat comprend les informations suivantes sur le compte utilisateur spécifié : nom de connexion, nom réel, temps de connexion, temps écoulé depuis la dernière activité (Idle-Time), répertoire d'accueil de l'utilisateur, login-shell, emplacement (Office number), mail et numéro de téléphone (si disponibles) ainsi que le contenu des fichiers .plan, .project, .pgpkey et .forward dans le répertoire d'accueil de l'utilisateur (si disponible).
Utilisez finger sans nom d'utilisateur pour récupérer des informations sur votre propre compte.

groups Appeler les membres d'un groupe
La commande groups dresse la liste des membres d'un groupe d'un compte utilisateur sélectionné.
Utilisez la commande en ligne de commande selon le schéma suivant :
groups [OPTIONS] [NOMUTILISATEUR]
Utilisez groups sans nom d'utilisateur pour lister tous les groupes auxquels votre compte utilisateur appartient.

id Récupération des ID d'utilisateur et de groupe
La commande id émet les identifiants d'utilisateur et de groupe du compte utilisateur sélectionné.
id [OPTIONS] [NOMUTILISATEUR]
Vous pouvez récupérer vos propres identifiants en passant la commande sans nom d'utilisateur.
La portée de la sortie peut être limitée à l'aide des options.

last Récupérer les informations sur les utilisateurs connectés en dernier
Utilisez la commande last selon le schéma ci-dessous pour obtenir la liste des utilisateurs

qui se sont connectés en dernier, avec leur temps de connexion et de déconnexion :

last [OPTIONS]

Les informations correspondantes sont extraites du fichier wtmp sous /var/log/wtmp. Si vous voulez seulement consulter les informations d'un compte spécifique, saisissez la commande avec le nom d'utilisateur désiré.

last [OPTIONS] [NOMUTILISATEUR]

w

Récupérer les utilisateurs effectifs et leurs processus

La commande w affiche une liste de tous les utilisateurs connectés, avec l'ensemble des processus qu'ils exécutent.

Utilisez w en combinaison avec un nom d'utilisateur pour restreindre la commande à cet utilisateur :

w [OPTIONS] [NOMUTILISATEUR]

Si nécessaire, l'étendue et le format de la sortie peuvent être ajustés à l'aide des options.

who

Obtenir des informations détaillées sur les utilisateurs enregistrés

La commande who permet d'afficher des informations détaillées sur les utilisateurs connectés sur le système.

La syntaxe générale de l'appel est la suivante :

who [OPTION] [FICHERSOURCE]

who prend en charge diverses options permettant d'ajuster la sortie d'informations.

Par défaut, who recouvre à partir du fichier /var/run/run/utmp les données des utilisateurs alors connectés.

Pour récupérer des informations sur les logins précédents, utilisez who avec le fichier source /var/log/wtmp :

who [OPTION] /var/log/wtmp

Pour obtenir des informations sur les connexions qui ont échoué, appelez who avec /var/log/btmp :

who [OPTION] /var/log/btmp

whoami

Récupérer votre propre nom d'utilisateur

Utilisez la commande whoami pour récupérer votre propre nom d'utilisateur.

whoami [OPTIONS]

adduser

Créer un compte utilisateur

La façon la plus simple de créer un compte utilisateur est d'utiliser le programme en ligne de commande adduser. Ceci nécessite des droits root. Voici la syntaxe utilisée :

adduser [OPTIONS] NOMUTILISATEUR

Si vous utilisez adduser sans options, un ID utilisateur, le répertoire d'accueil et un groupe d'utilisateurs du même nom seront automatiquement créés en plus du nouveau compte.

Exemple :

adduser test

Sortie du terminal :

Adding user 'test' (1001) ...

Adding new group 'test' (1001) ...

Adding new user 'test' (1001) with group 'test' ...

Creating home directory '/home/test' ...

Copying files from '/etc/skel' ...

Un dialogue est ensuite généré, dans lequel vous pouvez définir le mot de passe et les informations utilisateur avancées (nom réel, numéro de bureau, numéros de téléphone, etc.).

Cela peut être toutefois ajusté ou bloqué grâce à des options additionnelles.

Le script Perl adduser est basé sur le programme de bas niveau useradd. Il offre les mêmes fonctions mais de manière plus intuitive et conviviale.

chfn

Paramétrer des informations utilisateur avancées

La commande `chfn` (abréviation de change finger) permet de configurer les informations avancées d'un compte utilisateur telles que le nom réel, l'emplacement de son bureau et les numéros de téléphone privés et professionnels.

La syntaxe générale de `chfn` est :

`chfn [OPTION "NOUVELLE VALEUR"] [NOMUTILISATEUR]`

La commande doit être exécutée avec des droits root.

Vous pouvez définir les informations utilisateur que vous souhaitez remplacer par une nouvelle valeur à l'aide des options `-f` (nom réel), `-r` (emplacement de bureau), `-w` (numéro professionnel) et `-h` (numéro personnel).

Dans l'exemple suivant, l'emplacement précédent du bureau de `peter23` est remplacé par la nouvelle valeur `122`.

`chfn -r "122" peter23`

chsh

Modification du Shell par défaut

La commande `chsh` (abréviation de change shell) modifie le shell de connexion d'un utilisateur sélectionné.

Orientez-vous vers le schéma suivant lors de la saisie des données :

`chsh [OPTIONS] NOMUTILISATEUR`

Pour ajuster le shell de connexion d'un utilisateur, `chsh` est utilisé avec l'option `-s`. L'option est suivie du chemin d'accès vers l'interpréteur de commandes désiré (par exemple `/usr/bin/fish`).

À noter : les utilisateurs sans droits root sont seulement autorisés à personnaliser leur propre shell. Si vous voulez changer le shell des autres utilisateurs, exécutez la commande avec des droits root.

Exemple :

`sudo chsh -s /usr/bin/fish peter23`

Pour l'utilisateur `peter23`, le shell `fish` (friendly interactive shell) est alors défini par défaut.

Le changement d'interpréteur de commandes ne prend effet que lorsque l'utilisateur se déconnecte et se reconnecte.

deluser

Supprimer un compte utilisateur

`deluser` permet de supprimer dans les fichiers du compte système toutes les entrées d'un compte utilisateur sélectionné.

L'utilisation de `deluser` nécessite des droits root. La commande est basée sur le schéma suivant :

`deluser [OPTIONS] NOMUTILISATEUR`

Exemple :

`deluser peter23`

Le compte utilisateur `peter23` est supprimé.

Si vous souhaitez également supprimer tous les fichiers du répertoire personnel de l'utilisateur, utilisez la commande avec l'option `--remove-home`. Si tous les fichiers utilisateurs doivent être supprimés du système, l'option `--remove-all-files` est utilisée.

`deluser --remove-all-files peter23`

Si tous les fichiers de l'utilisateur doivent être sauvegardés avant suppression, utilisez `deluser` avec l'option `-backup-to` et spécifiez le répertoire désiré.

`deluser --backup-to /chemin/acces/au/repertoire peter23`

`deluser` est un script Perl qui offre les fonctions du programme de bas niveau `userdel` sous une forme plus conviviale.

groupadd

Création de groupes d'utilisateurs

La ligne de commande `groupadd` est utilisée pour établir de nouveaux groupes d'utilisateurs.

Utilisez `groupadd` avec des droits root selon le schéma suivant :

`sudo groupadd [OPTIONS] GROUPE`

Chaque groupe créé reçoit un ID de groupe unique (GID). Les ID de groupe compris entre 0 et 99 sont réservés aux groupes systèmes. Si vous voulez définir vous-même le GID d'un nouveau groupe d'utilisateurs, utilisez la commande `groupadd` avec l'option `-g` (GID).

Dans l'exemple suivant, le groupe users est créé avec le GID 1425.

```
groupadd -g 1425 users
```

Si vous voulez créer un groupe système, utilisez l'option -r (root).

delgroup

Supprimer un groupe d'utilisateurs

La commande delgroup (abréviation de delete group) permet de supprimer un groupe d'utilisateurs existant.

La syntaxe générale de delgroup est :

```
delgroup [OPTIONS] GROUPE
```

Les droits roots sont nécessaires pour pouvoir exécuter la commande.

L'appel suivant entraîne la suppression du groupe users :

```
delgroup users
```

Tout comme deluser, il s'agit d'un script Perl qui fournit les fonctions du programme de bas niveau groupdel sous une forme plus conviviale.

groupmod

Personnaliser le groupe d'utilisateurs

Via groupmod, le nom et le GID d'un groupe d'utilisateurs existant peuvent être ajustés.

La commande en ligne de commande est utilisée avec les droits root selon le schéma suivant :

```
groupmod OPTIONS GROUPE
```

Utilisez groupmod avec l'option -g pour personnaliser le GID. L'utilisation de l'option -n permet de renommer le nom du groupe.

Exemple :

```
groupmod -g 1800 users
```

Le GID du groupe d'utilisateurs est défini sur 1800.

```
groupmod -n all users
```

Le groupe d'utilisateurs est renommé all.

newgrp

Modifier le groupe d'utilisateurs

newgrp (abréviation de new group) est une commande qui permet aux utilisateurs connectés de changer leur ID de groupe sans avoir à se connecter et se déconnecter à nouveau.

La syntaxe générale de la commande est :

```
newgrp [-] [GROUPE]
```

Si newgrp est remis avec le paramètre optionnel [-], le changement de groupe entraîne un redémarrage de l'environnement utilisateur, comme si l'utilisateur s'était reconnecté.

Les utilisateurs qui utilisent newgrp sans spécification de groupe bascule vers le groupe par défaut spécifié dans /etc/passwd.

En principe, un utilisateur doit être membre du groupe vers lequel il souhaite basculer. Les groupes protégés par mot de passe sont une exception. Si tel est le cas, le mot de passe sera demandé via le terminal avant le changement.

passwd

Changer le mot de passe de l'utilisateur

Utilisez le programme en ligne de commande passwd pour changer le mot de passe d'un utilisateur ou pour définir des intervalles de modification ou verrouillage de ce mot de passe.

La commande est basée sur la syntaxe suivante :

```
passwd [OPTIONS] NOMUTILISATEUR
```

Si vous voulez changer le mot de passe d'un autre utilisateur, vous avez besoin des droits root.

Utilisez la commande passwd sans nom d'utilisateur pour changer votre propre mot de passe.

```
passwd
```

Pour verrouiller le mot de passe de l'utilisateur sélectionné, utilisez la commande passwd avec l'option -l (lock).

```
passwd -l NOMUTILISATEUR
```

D'autres options offrent la possibilité de définir une période d'expiration pour les mots de passe (-x) ainsi que des intervalles de temps pour les avertissements (-w) ou le verrouillage

(-i) :

`passwd -x MAX_JOURS -w WARN_JOURS -i INACTIF_JOURS NOMUTILISATEUR`

L'exemple suivant détermine que l'utilisateur peter24 doit renouveler son mot de passe tous les 30 jours. Il recevra un avertissement 5 jours avant la date limite. Si le mot de passe n'est pas renouvelé, il expire et le compte utilisateur peter24 est bloqué 3 jours plus tard.

`passwd -x 30 -w 5 -i 3 peter24`

sudo

Exécuter des programmes avec les droits d'autres utilisateurs

La commande sudo (abréviation de substitute user do) peut être placée devant les appels de programme pour les exécuter avec les droits d'un autre utilisateur. Un mot de passe doit être en principe saisi à cet effet. La commande sudo exige toujours le mot de passe de l'utilisateur appelant.

Si la commande est donnée sans nom d'utilisateur, le super-utilisateur root est utilisé comme cible.

`sudo APPELPROGRAMME`

Dans /etc/sudoers, les administrateurs peuvent définir les personnes pouvant utiliser sudo et quels appels de programme sont autorisés. Pour utiliser la commande sudo, l'utilisateur doit appartenir au groupe sudo.

Si vous voulez sélectionner un autre utilisateur cible, utilisez sudo avec l'option -u et le nom d'utilisateur souhaité.

`sudo -u NOMUTILISATEUR APPELPROGRAMME`

Un tel changement d'utilisateur n'est possible que si cela est déjà autorisé dans /etc/sudoers.

Si vous voulez passer au shell root en permanence pour exécuter des commandes avec des droits administrateur, utilisez sudo avec l'option -i.

`sudo -i`

La commande sudo est utile car elle permet aux utilisateurs d'exécuter des commandes prédéfinies en tant qu'utilisateurs root, sans avoir à transmettre le mot de passe root.

su

Travailler avec les droits d'autres utilisateurs

La commande su permet également à un utilisateur d'exécuter temporairement des appels de programme avec les droits d'un utilisateur cible. Contrairement à sudo, les commandes ne sont toutefois pas exécutées directement. En effet, c'est plutôt un changement d'identité qui est opéré. De plus, su ne demande pas le mot de passe de l'utilisateur appelant, mais celui de l'utilisateur cible. Pour appeler des programmes en tant que super-utilisateur, un utilisateur doit par conséquent avoir en principe besoin du mot de passe root du système. De plus, contrairement à sudo, su n'est pas limité à un ensemble d'appels de programme prédéfinis par l'administrateur.

Voici la syntaxe générale de la commande :

`su [OPTIONS] [NOMUTILISATEUR]`

Un appel sans NOMUTILISATEUR définit automatiquement root comme utilisateur cible.

usermod

Personnaliser le compte utilisateur

La commande en ligne de commande usermod vous permet de remanier les comptes utilisateurs déjà créés.

Utilisez usermod avec les privilèges root selon le schéma suivant :

`usermod [OPTIONS] NOMUTILISATEUR`

Les options permettent de définir les modifications à apporter.

Modifier le nom utilisateur (-l NOUVEAU_NOM) :

`usermod -l peter24 peter23`

L'utilisateur peter23 s'appelle maintenant peter24.

Créer un nouveau répertoire home (répertoire -d) et déplacez les anciens fichiers (-m) :

`usermod -d /chemin/vers/repertoire/peter24 -m peter24`

Tous les fichiers de l'ancien répertoire d'accueil sont déplacés vers le nouveau répertoire d'accueil.

Verrouiller l'utilisateur (-L) :

`usermod -L peter24`

Le mot de passe de l'utilisateur peter24 est verrouillé.

Ajouter des utilisateurs au groupe (-a) en conservant tous les autres membres du groupe (-G) :

`usermod -aG users peter24`

Peter est ajouté au groupe users.

logger	Créer des entrées log Le programme en ligne de commande logger peut être utilisé pour créer des entrées dans le journal système. La commande logger suit le schéma suivant : <code>logger "VOTRE MESSAGE"</code> Vous trouverez le journal système sous <code>/var/log/syslog</code>.
reboot	Faire redémarrer le système <code>reboot</code> est une commande qui permet de faire redémarrer le système. Pour lancer un redémarrage, la commande doit être exécutée avec les droits root. <code>reboot [OPTIONS]</code>
rtcwake	Démarrage et veille automatique du système <code>rtcwake</code> est une commande qui permet de planifier le démarrage et la veille du système. Elle est basée sur la syntaxe suivante : <code>rtcwake [OPTIONS] [MODE] [Temps]</code> Sélectionnez un mode spécifique (-m MODE) sur lequel le système doit se mettre en veille pour une durée définie en secondes (-s TEMPS EN SECONDE). Autrement, vous avez aussi la possibilité de sortir votre système de la veille à une heure définie précisément selon l'heure UNIX (-t). Exemple 1 : <code>rtcwake -m standby -s 300</code> Le système est mis en mode veille pendant 5 minutes (300 secondes). Exemple 2 : <code>rtcwake -m off -t 1490997660</code> Le système est éteint et remis en route à l'heure Unix 1490997660, ce qui correspond à la date suivante : 01.04.2017 - 00:01:00. L'heure Unix correspond au nombre de secondes depuis le 1.1.1970 à 00:00. Comme l'information en heure Unix est difficile à saisir, il est recommandé de la convertir en utilisant la commande <code>date</code> (voir ci-dessous). <code>rtcwake -m off -t \$(date -d '20170401 00:01' +%s)</code>
shutdown	Arrêt du système La commande <code>shutdown</code> peut être utilisée par les utilisateurs root pour arrêter le système. Elle est basée sur la syntaxe suivante : <code>shutdown [OPTIONS] [TEMPS] [MESSAGE]</code> Si vous souhaitez procéder à un arrêt du système, vous pouvez définir une heure. Utilisez un temps spécifique (hh:mm) ou un compte à rebours (+m). Les autres utilisateurs du système reçoivent une annonce d'arrêt. Si nécessaire, il peut être complété par un message individuel : Dans l'exemple suivant, le système s'arrêtera dans 10 minutes : <code>shutdown +10</code> Si la commande <code>shutdown</code> est utilisée avec l'option -r, un redémarrage suit l'arrêt du système. <code>shutdown -r +10</code>

date	Récupérer la date système La commande <code>date</code> renvoie, comme son nom laisse le présager, à la date et heure du système. <code>date [OPTIONS] [FORMATDESORTIE]</code>
-------------	---

Si vous voulez travailler avec une certaine date dans un appel de programme (voir `rtctime`), ceci peut être défini avec l'option `-d 'DATE'`. Par ailleurs, diverses options pourront être prises en charge, avec lesquelles les informations de date et d'heure peuvent être communiquées dans le format souhaité.

Par exemple, utilisez l'option `+%s` pour afficher une date en heure Unix (nombre de secondes depuis 1970-01-01 00:00:00 UTC).

Exemple :

```
date -d '20170427 11:29' +%s
```

Sortie :

```
1493285340
```

1493285340 est l'heure Unix correspondant au 27.04.2017 à 11h29.

df

Obtenir l'espace libre du disque dur

Utilisez la commande `df` (disk free) selon le schéma suivant pour afficher l'espace disque disponible des partitions montées.

```
df [OPTIONS]
```

Si la commande est utilisée avec un fichier spécifique, le système ne spécifie que l'espace libre de la partition sur laquelle se trouve le fichier.

```
df [OPTIONS] [FICHIER]
```

L'option `-l` (local) limite `df` aux systèmes de fichiers locaux. Des options sont également prises en charge pour personnaliser le format de sortie. Pour une sortie lisible par l'homme, il est conseillé l'option `-h` (human readable) : par exemple 3K 124M 1G.

dmesg

Obtenir des annonces de la mémoire tampon du noyau

Le programme `dmesg` (abréviation de display message) émet dans le terminal des messages à partir de la mémoire tampon du noyau, ce qui vous permet de localiser les erreurs de matériel.

Utilisez `dmesg` selon le schéma suivant :

```
dmesg [OPTIONS]
```

La sortie `dmesg` contient tous les messages du processus de démarrage et est donc tout aussi longue. Le programme en ligne de commande est souvent utilisé en combinaison avec un pager tel que `more`, `less` ou `tail`.

Exemple :

```
dmesg | tail
```

La sortie `dmesg` est transmise au pager `tail` à l'aide du pipe operator (`|`). Ceci permet que seuls les 10 derniers messages soient affichés dans le terminal.

Une combinaison avec la commande `grep` permet de rechercher des messages spécifiques.

du

Obtenir l'espace occupé sur le disque dur

Si vous souhaitez savoir combien d'espace disque est occupé par des répertoires sur votre système, utilisez la commande `du` (abréviation de disk usage) selon le schéma suivant :

```
du [OPTIONS] [REPERTOIRE]
```

Spécifier un répertoire n'est pas obligatoire. L'espace disque occupé peut être édité dans un format compréhensible par l'homme en utilisant l'option `-h`.

free

Obtenir les disponibilités de la mémoire vive

La commande `free` émet des informations sur la mémoire vive.

La syntaxe générale est :

```
free [OPTIONS]
```

La sortie émet deux données : Mem (Memory) et Swap.

Mem est la mémoire de votre système. Si elle est épuisée, Linux transfère une partie des données stockées dans la RAM vers le disque dur. C'est ce qu'on appelle le swap space.

`free` peut également se servir de l'option `-h` pour sortir les informations de la mémoire dans un format lisible par l'homme.

hostname

Obtenir le nom d'hébergement

Utilisez la commande `hostname` selon le schéma ci-dessous pour afficher le nom DNS du système.

hostname [OPTIONS]

uname Obtenir des informations sur la machine d'exécution
La commande uname (pour unix name) est utilisée pour obtenir des informations système sur la machine d'exécution.
La commande prend en charge différentes options permettant de filtrer les informations de la sortie.
uname [OPTIONS]

uptime Retrouver le temps d'exécution du système
Si vous voulez retrouver la durée d'exécution de votre système depuis le dernier processus de démarrage, utilisez la commande en ligne de commande uptime selon le schéma suivant :
uptime

vmstat Obtenir des statistiques sur la mémoire virtuelle
L'outil de monitoring vmstat peut être utilisé pour récupérer des informations sur la mémoire virtuelle, les opérations de lecture/écriture du disque ainsi que l'activité du processeur (CPU).
Pour faire afficher les valeurs moyennes depuis le dernier démarrage du système, appelez vmstat comme suit :
vmstat [OPTIONS]
Autrement, vmstat permet un monitoring continue, dans lequel les valeurs du système sont appelées aussi souvent que nécessaire dans un intervalle de temps défini en secondes.
vmstat [OPTIONS] [INTERVALE [REPETITIONS]]
Exemple :
vmstat 4 8
La requête s'effectue en huit passages toutes les 4 secondes.
Pour interrompre la requête en cours, utilisez la combinaison de touches [CTRL] + [C].

lscpu Informations sur le processeur
Utilisez lscpu (abréviation de list cpu) selon le schéma suivant pour afficher dans le terminal les informations sur l'architecture CPU.
lscpu [OPTIONS]
Pour les options possibles, reportez-vous aux pages d'aide de votre système d'exploitation.

lshw Informations sur le matériel (hardware)
La commande lshw (pour list hardware) affiche des informations sur les composants hardware du terminal. Les informations portent sur le processeur, les modules de mémoire et les périphériques tels que les cartes son, cartes graphiques ou les lecteurs connectés à des interfaces PCI, USB ou IDE.
Utilisez lshw selon le schéma suivant :
lshw [OPTIONS]
La commande prend en charge différentes options pour personnaliser le format de sortie (-html, -xml, -short, -businfo) et la quantité d'informations (par exemple -sanitize pour cacher les informations sensibles).

lspci Informations sur les périphériques PCI
Utilisez lspci (abréviation de list pci) selon le schéma suivant pour afficher des informations détaillées sur les périphériques PCI :
lspci [OPTIONS]
Reportez-vous aux pages d'aide de votre système d'exploitation pour connaître les options possibles.

lsusb Informations sur les périphériques USB
Utilisez lsusb (abréviation de list usb) pour afficher dans le terminal des informations détaillées sur les périphériques USB.

lsusb [OPTIONS]

Reportez-vous aux pages d'aide de votre système d'exploitation pour connaître les options possibles.

chrt

Attributs en temps réel

chrt est un programme en ligne de commande pour les contrôles de processus avancés qui permettent de déterminer et d'ajuster les attributs en temps réel (politique d'ordonnancement et priorité) des processus en cours d'exécution ou d'exécuter des commandes et leurs arguments avec des attributs en temps réel spécifiques.

La syntaxe générale de la commande est :

chrt [OPTIONS] [PRIORITE] PID/COMMANDE [ARGUMENTS]

Utilisez chrt sans spécifier de priorité mais avec l'option -p pour déterminer les attributs en temps réel des processus sélectionnés :

chrt -p PID

Exemple :

chrt -p 1234

chrt édite les attributs temps réel du processus 1234.

La commande est très souvent utilisée comme suit pour exécuter une commande et ses arguments avec une priorité en temps réel spécifique.

chrt [OPTIONS] PRIORITE COMMANDE [ARGUMENTS]

Exemple :

chrt 99 firefox

Le programme Firefox est lancé avec une priorité en temps réel de 99.

Si, en revanche, la priorité en temps réel des processus en cours d'exécution doit être ajustée, la syntaxe suivante est utilisée :

chrt -p PRIORITE PID

Exemple :

chrt -p 20 1234

La priorité en temps réel du processus 1234 est fixée à 20.

De plus, chrt offre la possibilité de personnaliser ou de définir la règle de planification de l'exécution ou du redémarrage des processus à l'aide d'options.

chrt utilise la politique d'ordonnancement SCHED_RR (Round Robin, explicitement avec l'option -r) comme valeur par défaut. Cela signifie que tous les processus prêts pour le calcul se voient attribuer un temps de traitement CPU les uns après les autres pour une période de temps donnée. Ceci permet de spécifier combien de temps un processus peut durer jusqu'à ce qu'il soit supplanté par un autre processus. La taille pour le laps de temps du processus dépend de la priorité. Linux présente 140 niveaux de priorité pour les processus (0 = priorité la plus élevée, 139 = priorité la plus faible). Les niveaux de 1 à 99 sont réservés aux processus dits prioritaires en temps réel. Les processus utilisateurs, par contre, sont généralement exécutés à un niveau de processus de 100 à 139. Ceci correspond à une valeur nice de -20 à +19 (voir nice).

En plus de SCHED_RR, Linux a une autre politique d'ordonnancement pour les processus en temps réel : SCHED_FIFO (option -f). Comme SCHED_RR, SCHED_FIFO fonctionne en tant qu'algorithme first-in/first-out. Cependant, aucun intervalle de temps n'est utilisé. Les processus démarrés avec SCHED_FIFO s'exécutent jusqu'à ce qu'ils soient arrêtés ou remplacés par un processus ayant une priorité temps réel plus élevée. Les processus remplacés se déplacent vers la fin de la file d'attente.

ionice	Affecter des classes d'ordonnancement d'E/S La commande ionice est utilisée pour influencer la priorité d'un processus qui utilise l'interface d'E/S du noyau. La syntaxe générale de la commande est : ionice [OPTIONS] COMMANDE Pour pouvoir exécuter ionice, des droits root sont nécessaires. La commande distingue trois classes d'ordonnancement qui sont passées avec l'option -cNOMBRE. Les valeurs numériques possibles sont 1, 2 et 3. 1 = Temps réel : l'action E/S est exécutée immédiatement. 2 = Best-Effort : l'action E/S est exécutée le plus rapidement possible. 3 = Idle : l'action E/S n'est exécutée que si aucun autre processus ne prend en compte du temps E/S. Le PID d'un processus en cours d'exécution est transmis avec l'option -pPID. Exemple : ionice -c2 -p1234 La classe d'ordonnancement 2 (Best Effort) est transférée au processus avec le PID 1234.
kill	Arrêt et fin du processus via PID kill est un programme en ligne de commande qui peut être utilisé pour arrêter et mettre fin aux processus. La commande est passée selon le schéma ci-dessous en spécifiant le signal souhaité et l'ID du processus sélectionné. kill [OPTIONS] [-SIGNAL] PID Les signaux courants sont : - TERM : fait en sorte qu'un processus se termine tout seul (par défaut). - KILL : force un processus à prendre fin (par le système). - STOP : arrête un processus. - CONT : reprend un processus arrêté. L'appel suivant envoie un signal au process 1234, lui demandant de se terminer lui-même. Comme aucun signal n'a été transmis, kill envoie le signal par défaut TERM. kill 1234 Permettez toujours aux processus de se terminer eux-mêmes et ne forcez l'action via KILL que si le processus adressé ne réagit pas comme prévu. kill -KILL 1234 Si vous voulez seulement faire une pause, utilisez les appels suivants pour arrêter temporairement le processus ou lui permettre de reprendre : kill -STOP 1234 kill -CONT 1234 Utilisez la commande kill avec l'option -l (--list) pour afficher tous les signaux possibles qui peuvent être transmis aux processus via kill.
killall	Arrêter et terminer les processus par leur nom Utilisez killall avec un terme de recherche spécifique pour ne terminer que les processus dont les noms correspondent exactement (les 15 premiers caractères sont comparés). killall [OPTIONS] [-SIGNAL] [NOMPROCESSUS] L'option -e (--exact) permet d'étendre le réglage à tous les caractères du nom du processus.
nice	Définir les priorités du processus La commande nice attribue une valeur comprise entre -20 et +19 à un processus au démarrage, après quoi la puissance de traitement disponible du système est distribuée. L'intervalle de -20 à +19 correspond aux niveaux de priorité Linux de 100 à 139. Un processus avec une valeur nice de -20 a ainsi une priorité plus élevée qu'un processus avec une valeur nice de 19. La syntaxe seule est : nice [OPTION] [COMMANDE] Sans autre spécification, chaque processus commence avec une valeur nice de 0. Utilisez l'option -n pour définir les priorités du processus. Notez que les priorités négatives ne

peuvent être assignées qu'avec des droits root.
Dans l'exemple suivant, l'éditeur nano est lancé avec une priorité de 4 :
nice -n 4 nano

nohup

Déclencher le processus d'une session
Normalement, tous les processus dépendants d'un utilisateur se terminent automatiquement dès que l'utilisateur ferme la session terminal (par exemple, via exit).
La commande nohup (abréviation de no hangup) déclenche un processus à partir de la session en cours et permet de poursuivre son exécution même si vous vous déconnectez du système. Le signal HUP associé (hangup), qui provoque normalement l'arrêt automatique d'un processus, est supprimé par nohup.
L'appel de programme est basé sur le schéma suivant :
nohup COMMANDE

pgrep

Récupérer un PID grâce à un terme de recherche
Le programme en ligne de commande pgrep compare la liste des processus en cours d'exécution à un terme de recherche et affiche leurs PID respectifs s'ils correspondent.
La syntaxe générale de l'appel est :
pgrep [OPTIONS] TERMEDERECHERCHE
Par défaut, pgrep édite les PID de tous les processus qui contiennent le terme de recherche.
Exemple :
pgrep ssh
Tous les processus qui contiennent le terme de recherche ssh dans leur nom sont répertoriés.
Si vous souhaitez limiter la recherche à des correspondances exactes, utilisez la commande avec l'option -x.
Exemple : pgrep -x sshd
Seuls les processus dont le nom exact est sshd sont listés.
Si vous voulez récupérer le nom du processus avec le PID, utilisez pgrep avec l'option -l.
pgrep prend en compte (comme grep) les termes de recherche basés sur des expressions régulières.

pidof

Retrouver les PID
La ligne de commande pidof affiche les numéros d'identification de tous les processus d'un programme (PID).
Retrouver les PID via pidof selon le schéma suivant :
pidof [OPTIONS] PROGRAMME
Avec l'appel ci-dessous, les ID de tous les processus en cours du programme nano sont affichés dans le terminal :
pidof nano
Si vous souhaitez seulement afficher le premier ID de processus, utilisez pidof avec l'option -s (abréviation de single shot).

pkill

Arrêter et terminer des processus grâce à des termes de recherche
Comme kill, pkill envoie aussi un signal à un processus sélectionné. Cependant, l'adressage ne s'effectue pas par le PID. À la place, un terme de recherche est comparé avec les noms des processus en cours d'exécution. Il peut être formulé comme une expression régulière.
pkill utilise également par défaut le signal TERM si aucun autre signal n'a été défini. La syntaxe générale de la commande est :
pkill [OPTIONS] [-SIGNAL] [TERMEDERECHERCHE]
Des options supplémentaires peuvent être utilisées pour restreindre la commande aux processus d'un utilisateur particulier (-U UID), aux processus enfants d'un processus parent spécifique (-P PID), ou encore aux processus les plus récents (-n) ou les plus anciens (-o).
Alors que pkill s'adresse à tous les processus dont les noms contiennent le terme de recherche, la commande killall ne s'adresse qu'aux processus ayant des correspondances exactes.

ps	Obtenir une liste de tous les processus en cours d'exécution La commande ps affiche une liste de tous les processus en cours d'exécution dans le terminal. ps [OPTIONS] Si vous avez besoin d'une sortie détaillée, utilisez ps avec les options -f (détaillé) ou -F (très détaillé). Vous trouverez d'autres options possibles dans les pages de manuel de votre système d'exploitation.
ps tree	Récupérer les processus en cours sous forme d'arborescence Utilisez ps tree pour afficher tous les processus en cours d'exécution dans une arborescence. La syntaxe générale de la commande est : ps tree [OPTIONS] Le format et l'étendue de la sortie peuvent être personnalisés à l'aide de différentes options.
renice	Ajuster les priorités des processus en cours d'exécution La commande renice permet d'ajuster la priorité d'un processus en cours d'exécution. La syntaxe générale est : renice PRIORITE [OPTIONS] L'adressage se fait à l'aide d'options via l'ID du processus (-p PID), l'ID de groupe (-g GID) ou un nom d'utilisateur (-u UTILISATEUR). Exemple : renice 12 -p 1234 La priorité 12 est attribuée au processus avec l'ID 1234. renice 3 -g 3456 Tous les processus en cours du groupe avec le GID 3456 se voient attribuer une priorité de 3. sudo renice -6 -u peter24 Tous les processus en cours d'exécution de l'utilisateur peter24 se voient attribuer une priorité de -6. Si vous utilisez renice sans option, la valeur par défaut -p est prise en compte et la chaîne qui suit est interprétée comme ID du processus.
sleep	Retarder l'exécution du processus La commande sleep vous permet d'interrompre la session du terminal en cours pendant une période de temps donnée. La syntaxe générale de la commande est : sleep NOMBRE[SUFFIXE] Si vous utilisez sleep sans suffixe, le nombre spécifié est interprété comme un temps en secondes (s). Toutefois, vous pouvez également interrompre la session du terminal pendant des minutes (m), heures (h) ou jours (d) donnés. L'appel suivant interrompt la session pendant 4 minutes : sleep 4m La commande peut être utilisée par exemple pour retarder l'exécution de la commande qui suit : sleep 1h && reboot Le système attend une heure, puis exécute boot, ce qui provoque le redémarrage du système.
taskset	Affecter des processus à des processeurs spécifiques taskset est une commande pour les contrôles de processus avancés qui est utilisée pour les systèmes multiprocesseurs afin d'assigner des processus ou des commandes à des processeurs spécifiques. La commande nécessite des droits root et utilise l'un des schémas suivants : taskset [OPTIONS] MASQUE COMMANDE taskset [OPTIONS] -p PID L'affectation de process/commande au processeur se fait à l'aide d'un masque en bits

hexadécimaux. Par exemple :

0x00000001 = Processeur #0

0x00000003 = Processeur #0 et #1

0xFFFFFFFF = tous les processeurs (#0 jusqu'à #31)

Comme une telle affectation des masques n'est pas très intuitive, taskset est généralement utilisé avec l'option -c (--cpu-list), qui permet un classement numérique des processeurs (par exemple 0, 5 7, 9-11).

La commande suivante demande au processus 1234 d'utiliser les processeurs 1 et 2 :

taskset -p 1234 -c 1,2

top

Vue d'ensemble dynamique des processus

Utilisez la commande top pour obtenir une vue d'ensemble dynamique de tous les processus en cours d'exécution.

L'appel repose sur le schéma suivant :

top [OPTIONS]

La sortie des informations des processus peut être personnalisée à l'aide de diverses options. En outre, l'aperçu des processus top prend en charge (entre autres choses) les touches de raccourci suivantes pour trier les sorties :

[P] = Trie les sorties par charge CPU

[M] = Trie les sorties en fonction de l'utilisation de la mémoire.

[N] = Trie les sorties numériquement par PID.

[A] = Trie les sorties suivant l'ancienneté.

[T] = Trie les sorties en fonction du temps.

[U NOMUTILISATEUR ou UID] = Trie les sorties en fonction de l'utilisateur.

Utilisez la touche de raccourci [H] pour afficher une page d'aide et [Q] pour quitter l'aperçu du processus.

head

Émission des premières lignes d'un fichier

Le pager head est utilisé pour sortir la première partie d'un fichier.

La syntaxe générale de la commande est :

head [OPTIONS] Fichier

Utilisez l'option -n NOMBRE_LIGNES pour définir le nombre de lignes à éditer en commençant par la première ligne.

Exemple :

head -n 3 exemple.txt

Les trois premières lignes du fichier exemple.txt sont éditées. Si aucune ligne n'est spécifiée, head édite les 10 premières lignes du fichier donné.

less

Afficher les fichiers texte dans le terminal

Le programme de ligne de commande less permet d'afficher le contenu d'un fichier texte dans le terminal.

La syntaxe générale est :

less [OPTIONS] FICHER

La sortie est automatiquement en mode interactif. Ceci vous permet de parcourir plus facilement le document sélectionné ou d'effectuer des recherches par mots-clés.

La touche [Q] met fin au mode lecture interactive. Pour d'autres touches de commande et les options possibles, reportez-vous au manuel du programme.

more

Afficher les fichiers texte dans le terminal

Le pager more remplit la même fonction que less, mais offre une plus petite gamme de fonctions.

Utilisez more selon le schéma suivant pour appeler un fichier texte et afficher son contenu

dans le terminal :

more [OPTIONS] FICHIER

Le programme en ligne de commande affiche toujours une page d'écran complète du fichier sélectionné. Si un fichier contient plusieurs pages, more lance également un mode interactif qui vous permet de faire défiler le document ou de rechercher des mots-clés à l'aide des touches de commande.

La touche [Q] permet de quitter le mode interactif. Pour les autres touches de commande et les options utilisables, reportez-vous aux pages d'aide de votre système d'exploitation.

tail

Sortie des dernières lignes d'un fichier

Tandis que head affiche par défaut les 10 premières lignes d'un fichier sélectionné, tail affiche les 10 dernières.

Les deux pagers sont utilisés selon le même schéma (voir head).

emacs

L'éditeur de texte Emacs

Emacs est un éditeur de texte indépendant de la plateforme.

Il démarre par défaut avec une interface utilisateur graphique, mais peut aussi être ouvert dans le terminal en utilisant l'option --no-window-system.

emacs --no-window-system

Emacs dispose d'un tutoriel intégré auquel vous pouvez accéder en appuyant sur [CTRL] + [H], [T].

nano

L'éditeur de texte Nano

Nano est une réplique GNU de l'éditeur de texte basé sur le terminal Pico qui est utilisé dans le cadre du client de messagerie Pine. Si Nano offre une plus petite gamme de fonctions que les éditeurs comparables (par exemple Vim), il est particulièrement agréable à utiliser.

Voici la syntaxe générale :

nano [OPTIONS] FICHIER

Le programme ouvre le fichier spécifié dans une fenêtre Edit du terminal.

Si vous appelez Nano sans nom de fichier, un nouveau fichier texte est créé et enregistré dans le répertoire actuellement sélectionné.

nano [OPTIONS]

Les raccourcis pour piloter le programme sont listés au bas de la fenêtre d'édition. Vous trouverez de plus amples informations sur Nano dans les pages de manuel du programme.

vim

L'éditeur de texte Vim

Vim (abréviation de Vi Improved) est une suite de l'éditeur de texte Vi. Il se distingue par de nombreuses améliorations telles que le surlignement syntaxique, un système d'aide complet, un script natif, des autocomplétions du code et une sélection visuelle du texte.

Le programme open source offre différents modes de fonctionnement pour le traitement de fichiers texte pur et peut être utilisé soit dans le terminal, soit comme application autonome avec interface utilisateur graphique (GVim). Le traitement du code de programme est au cœur du domaine d'application du programme.

Si vous démarrez Vim dans la console, il est commandé par le clavier. Habituellement, le programme est appelé avec un fichier texte selon le schéma suivant :

vim [OPTIONS] FICHIER

Vim charge les fichiers ouverts dans une mémoire tampon (buffer). Toutes les modifications que vous apportez au fichier ouvert sont également stockées ici. Si vous ouvrez Vim sans spécifier un fichier, le programme démarre avec un buffer vide. Le fichier d'application d'origine n'est pas ajusté tant que le processus de sauvegarde n'est pas ordonné à l'aide de la combinaison de touches correspondante. S'il n'existe pas de fichier correspondant au nom donné lors de l'appel du programme, ce dernier sera créé lors du processus d'enregistrement.

Une introduction complète à Vim est proposée par le programme vimtutor, que vous pouvez également lancer à partir de la ligne de commande.

arp	Affichage et manipulation du cache ARP Le programme en ligne de commande arp vous permet de consulter le cache ARP de votre système d'exploitation et de le manipuler. Utilisez arp sans modificateur pour afficher le contenu de la table ARP dans le terminal. <pre>arp</pre> Autrement, vous avez la possibilité de limiter la sortie aux entrées d'un nom d'hébergement spécifique grâce à l'option -a (au lieu d'une adresse IP). <pre>arp -a NOMHEBERGEMENT</pre> Exemple : <pre>arp -a exemple.com</pre> Si vous voulez créer une entrée ARP, utilisez l'option -s selon le schéma suivant : <pre>arp -s NOMHEBERGEMENT MAC_ADRESSE</pre> Exemple : <pre>arp -s exemple.com 00:05:23:73:e6:cf</pre> Si une entrée existante doit être supprimée, arp est utilisé avec l'option -d. <pre>arp -d NOMHEBERGEMENT</pre>
dig	Consulter des informations sur le DNS dig est un outil de recherche permettant de consulter des informations des serveurs DNS et de les émettre dans le terminal. L'utilitaire en ligne de commande est généralement utilisé pour consulter l'adresse IP et d'autres informations DNS pour un nom de domaine donné. Le schéma suivant est alors utilisé : <pre>dig [@SERVER] [DOMAIN] [TYP]</pre> SERVER est le serveur DNS dans lequel l'information désirée est recherchée. Si aucun serveur n'est spécifié, dig récupère le serveur DNS par défaut à partir de /etc/resolv.conf. DOMAIN représente le nom de domaine pour lequel l'information DNS doit être retrouvée. Avec TYP, le type de requête peut être fixé, par exemple ANY (toutes les entrées), A (enregistrement IPv4 d'un hôte) ou AAAA (enregistrement IPv6 d'un hôte). Le type défini par défaut est A. Utilisez dig avec l'option -x pour consulter le nom de domaine d'une adresse IP donnée lors d'une recherche inversée. <pre>dig [@SERVER] [-x IP-ADRESSE]</pre> Les arguments NAME, TYP et CLASSE ne sont pas nécessaires dans ce cas.
ftp	Transfert de fichiers via FTP Grâce au programme en ligne de commande ftp, la plupart des distributions Linux disposent d'un programme client préinstallé pour le transfert de données via FTP (File Transfer Protocol). Ceci vous permet d'échanger des fichiers entre le système local et un autre ordinateur du réseau. Utilisez ftp selon le schéma ci-dessous pour établir une connexion avec le serveur FTP de l'ordinateur cible. <pre>ftp [OPTIONS] [HOST[PORT]]</pre> L'adressage s'effectue via le nom d'hôte ou l'adresse IP. La spécification d'un numéro de port n'est pas obligatoire. Pour que la connexion soit établie, un nom d'utilisateur et le mot de passe correspondant vous sont généralement demandés. Si la connexion réussit, ftp lance un interpréteur de ligne de commande qui accepte les entrées de l'utilisateur sous forme de commandes. Le programme prend en charge différentes commandes avec lesquelles vous pouvez parcourir et gérer le système de fichiers de l'ordinateur cible et transférer des fichiers d'un système à un autre.
ip	Consulter et configurer des interfaces réseau Le programme en ligne de commande ip fait partie de la série de programmes iproute2,

avec laquelle les interfaces réseau peuvent être consultées et configurées via le terminal.

La syntaxe générale de la commande est :

`ip [OPTIONS] OBJET [COMMANDE [ARGUMENTS]]`

Pour définir quelle action est exécutée via ip, il est nécessaire d'utiliser des objets, des sous-commandes et leurs arguments.

Le programme prend en charge divers objets tels qu'address (adresse IP), link (interface réseau), route (entrée dans la table de routage) ou tunnel, auxquels des sous-commandes comme add, change, del, list ou show peuvent être appliquées.

Par exemple, si vous voulez obtenir l'adresse IP d'une interface réseau spécifique (par exemple eth0), utilisez la commande ip avec l'objet address, la commande show et l'argument dev eth0 :

`ip address show dev eth0`

Bon à savoir : les objets et les commandes peuvent également être écrits de manière courte :

`ip a s dev eth0`

Si vous voulez afficher toutes les informations d'une interface réseau (par exemple eth0), utilisez la commande ip avec l'objet link, la commande show et l'argument dev eth0 :

`ip link show dev eth0`

et de la même manière :

`ip l s dev eth0`

Pour activer ou désactiver une interface telle que eth0, procédez comme suit :

`ip link set eth0 up`

`ip link set eth0 down`

Avec sa large gamme de fonctions, la série de programmes iproute2 remplace un certain nombre d'outils réseau plus anciens tels que ifconfig, route et netstat.

Vous pourrez trouver une liste de toutes les options, objets, sous-commandes et arguments possibles pour ip ainsi que des informations sur les autres programmes iproute2 dans le manuel de votre système d'exploitation.

iw

Consulter et configurer des interfaces WLAN

Le programme en ligne de commande iw est utilisé pour configurer les interfaces WLAN et s'est imposé comme une alternative plus moderne à iwconfig.

La syntaxe est similaire à celle de la commande ip :

`iw [OPTIONS] OBJET [COMMANDE]`

Les objets possibles sont :

dev NOM_INTERFACE = interface réseau

phy NOM_PERIPHERIQUE = périphérique réseau sans fil (via Nom)

phy#INDEX_PERIPHERIQUE = périphérique réseau sans fil (via Index)

reg = regulatory agent pour la configuration des paramètres régionaux.

Utilisez iw avec la commande help pour afficher la syntaxe du programme et les options et commandes possibles.

`iw help`

Voici quelques exemples d'application du programme en ligne de commande iw ci-dessous.

Émettre les propriétés des périphériques de toutes les interfaces WLAN :

`iw list`

Obtenir l'état de connexion (vitesse de transmission et puissance du signal) d'une interface WLAN (par ex. wlan0) :

`iw dev wlan0 link`

Environnement WLAN :

`iw dev wlan0 scan`

Utilisez iw avec la commande scan pour émettre tous les réseaux WLAN dans la zone de réception avec leurs propriétés (canal radio, cryptage, puissance du signal, etc.).

Lire les paramètres régionaux :

`iw reg get`

Ajuster les paramètres régionaux :

`iw reg set FR`

Obtenir les propriétés du périphérique (par ex. de wlan0) :
iw list dev wlan0
Propriétés détaillées du périphérique :
iw dev wlan0 station dump
Obtenir les événements :
iw event

netstat Obtenir le statut des interfaces réseau
Le programme en ligne de commande netstat est utilisé pour consulter l'état des interfaces réseau.
La syntaxe générale de la commande est :
netstat [OPTIONS]
Utilisez netstat sans option pour afficher toutes les sockets ouvertes dans le terminal.
Vous pouvez également utiliser les options suivantes pour afficher les tables de routage (-r), les statistiques d'interface (-i), les connexions masquées (-M) ou les messages netlink (-N).
Le programme ss de la série de programmes iproute2 est une alternative à netstat.

nslookup Consulter des informations sur le DNS
Comme dig, nslookup est aussi utilisé pour consulter des informations sur le DNS. Le programme en ligne de commande est disponible selon deux modes : interactif et non interactif.
Lancez le mode interactif en entrant la commande nslookup dans le terminal sans autre spécification.
nslookup
Le programme peut maintenant accepter les commandes. Entrez par exemple un nom d'hôte (Domain) pour faire afficher l'adresse IP correspondante.
Vous pouvez également lancer une requête de recherche inversée en entrant une adresse IP pour que le nom d'hôte qui lui est associé s'affiche.
Le programme nslookup utilise automatiquement le serveur DNS par défaut du système.
Entrez la commande exit pour quitter nslookup.
Si vous voulez utiliser nslookup en mode non interactif, appelez le programme avec un nom d'hôte ou une adresse IP.
nslookup [OPTIONS] [HOST/IP]
Comme le programme est très ancien, il est recommandé aux utilisateurs d'utiliser dig à la place.

ping Vérifier la connexion réseau
Utilisez l'utilitaire en ligne de commande ping pour vérifier l'accessibilité des autres ordinateurs du réseau.
La commande est basée sur la syntaxe suivante :
ping [OPTIONS] CIBLE
Pour vérifier la connexion réseau, ping envoie un petit paquet de données au système cible spécifié (nom d'hôte ou IP) et évalue le temps jusqu'à ce que la réponse soit reçue.
En plus du temps aller-retour (Round trip time, RTT, ce qui désigne le temps écoulé entre l'envoi du paquet de données et la réception de la réponse), ping émet l'adresse IP du système cible sur le terminal. Le programme en ligne de commande est donc également adapté pour déterminer l'adresse IP d'un domaine.
Si ping est exécuté sans option, le programme s'exécute jusqu'à ce qu'il soit arrêté manuellement par les touches [CTRL] +[C].
Si vous voulez définir une heure de fin lors de l'appel du programme, utilisez les options -c CHIFFRE (nombre de demandes ping à envoyer) ou -w SECONDES (temps en secondes avant l'arrêt de ping).

route Affichage et traitement des tables de routage IP
Le programme en ligne de commande route peut être utilisé pour consulter et éditer les tables de routage IP du noyau.
La commande est basée sur la syntaxe suivante :
route [OPTIONS]

route [OPTIONS] [add|del] [-net|-host] CIBLE

Utilisez la commande sans option pour afficher la table de routage complète du noyau :
route

Pour définir un itinéraire vers un réseau, utilisez la sous-commande add.

route add -net 10.0.0.0

Si la destination est un sous-réseau, le masque du sous-réseau doit être spécifié avec l'option netmask MASQUE :

route add -net 10.0.0.0 netmask 256.245.155.0

Vous pouvez également configurer un itinéraire vers un ordinateur :

route add -host 218.89.72.191

Si le système possède plusieurs interfaces réseau, l'option dev INTERFACE doit être utilisée pour spécifier l'interface à utiliser :

route add -net 10.0.0.0 netmask 256.245.155.0 dev eth0

Si la destination ne peut être atteinte que par l'intermédiaire d'un routeur, il faut également le spécifier avec l'option gw ROUTER.

route add -net 10.0.0.0 netmask 256.245.155.0 gw 10.0.1.261

Si vous voulez supprimer un routeur, utilisez la sous-commande del.

route del -host 218.89.72.191

rsync

Synchroniser les fichiers

Le programme en ligne de commande rsync permet de synchroniser des fichiers localement ou sur un réseau. Tout d'abord, la taille et la date de modification des fichiers concernés sont comparés.

Si la source et la cible sont sur le même système, des fichiers différents sont copiés dans leur intégralité. Lors de la synchronisation sur un réseau, rsync utilise un algorithme de transfert delta, de sorte que seules les parties de fichier modifiées soient transférées du support de données source vers le système cible.

La syntaxe de l'appel est :

rsync [OPTIONS] SOURCE(S) CIBLE

Exemple :

rsync -a home/user/documents/ /home/user/backup

Tous les fichiers de home/user/documents/ sont comparés avec les fichiers du répertoire /home/user/backup.

La commande rsync est généralement exécutée avec l'option -a, ce qui garantit que tous les sous-répertoires et liens symboliques sont également copiés ainsi que tous les droits d'utilisateur.

sftp

Transfert de fichiers via SFTP

Le programme en ligne de commande sftp sert comme ftp au transfert de données dans le réseau. Toutes les opérations sont effectuées via une connexion SSH chiffrée (Secure Shell).

Comme ftp, sftp établit une connexion à un ordinateur cible sur le réseau et démarre ensuite un mode de commande interactif.

scp

Transfert de fichiers via SCP

scp (abréviation de secure copy) est un autre programme pour le transfert sécurisé des données dans le réseau directement via le terminal : scp copie les données d'un ordinateur à un autre, en utilisant également le protocole réseau SSH.

Le programme client fonctionne de la même manière que l'option fichier cp, mais il est utilisé selon la syntaxe suivante :

scp [OPTIONS] FICHIER [[user@]remote_host:]CHEMINACCES

Lorsque vous spécifiez le chemin d'accès de l'hôte distant, le nom d'utilisateur et le nom de l'hôte correspondant sont indiqués. Vous pouvez utiliser des chemins d'accès relatifs ou absolus pour les fichiers locaux.

Exemple :

scp /home/max/images/image.jpg max@exemple.com:/home/max/archiv

Le fichier image.jpg est copié à partir du répertoire local images vers le répertoire archiv sur

un ordinateur cible avec l'adresse exemple.com.

Le programme scp prend également en charge le transfert de données dans le sens inverse ainsi qu'entre deux systèmes distants.

scp [OPTIONS] [[user@]host:]FICHER CHEMINACCES

scp [OPTIONS] [[user@]host1:]FICHER [[user@]host2:]CHEMINACCES

D'autres options vous permettent d'ajuster le mode de transfert et les paramètres de cryptage.

traceroute Suivi des paquets de données
Utilisez la commande traceroute selon le schéma ci-dessous pour tracer le chemin de transfert d'un paquet de données IP entre votre système et un ordinateur cible.
traceroute [options] NOMHOTE
traceroute peut être utilisé pour identifier quel routeur et points d'échange Internet un paquet IP traverse en allant vers l'ordinateur cible (par exemple pour rechercher la cause d'un retard).

tty Emission des noms de terminal
La commande tty émet le nom de fichier du terminal défini comme entrée par défaut.
La syntaxe générale de la commande est : tty [OPTIONS]

tar Écrire et extraire des fichiers dans les archives tar
La commande tar signifie tape archiver, un programme développé à l'origine pour sauvegarder les données sur des lecteurs de bandes. Aujourd'hui encore, tar est l'un des programmes les plus populaires pour l'archivage de données sous Linux.
Le programme permet d'écrire séquentiellement différents fichiers et répertoires dans un fichier tar et de les restaurer si nécessaire. Contrairement au format zip couramment utilisé sous Windows, tous les droits d'utilisateur du fichier archivé sont conservés même après que ce dernier est décompressé.
Le programme en ligne de commande tar est appelé selon la syntaxe suivante :
tar [OPTIONS] FICHIERS
Si vous voulez créer une nouvelle archive, utilisez tar avec les options -c (create new archive) et -f (write archive or read from specified file).
Dans l'exemple suivant, fichier1.txt et fichier2.txt sont écrits dans l'archive exemple.tar qui vient d'être créée.
tar -cf exemple.tar fichier1.txt fichier2.txt
Pour visualiser le contenu d'une archive, utilisez tar avec les options -t (afficher le contenu d'une archive), -v (sortie détaillée) et -f (voir ci-dessus).
tar -tvf exemple.tar
Si les fichiers archivés doivent être décompressés dans le dossier courant, tar est utilisé avec les options -x (extraire les fichiers de l'archive) et -f (voir ci-dessus).
tar -xf exemple.tar
Avec -j (bzip2), -J (xz), -z (gzip) et -Z (compress), tar offre également des options qui permettent de compresser ou décompresser des archives en appelant un autre programme pendant le processus.
Dans l'exemple suivant, fichier1.txt et fichier2.txt sont archivés dans exemple.tar.gz et compressés avec gzip.
tar -czf exemple.tar.gz fichier_1.txt fichier_2.txt
La commande suivante extrait et décompresse tous les fichiers archivés dans exemple.tar.gz.
tar -xzf exemple.tar.gz

gzip / Compresser ou dézipper des fichiers avec gzip
gunzip gzip (abréviation de GNU zip) est un programme qui facilite la compression et la décompression des fichiers.
La syntaxe générale qui s'applique est :
gzip [OPTIONS] FICHER(S)

Par exemple, utilisez gzip comme suit pour convertir le fichier exemple.txt au format compressé exemple.txt.gz :

```
gzip exemple.txt
```

Notez que gzip supprime le fichier d'application original dans le cadre du processus de dézippage par défaut. Ceci peut être modifié avec l'option -k.

```
gzip -k exemple.txt
```

Si nécessaire, le programme peut être appliqué à plusieurs fichiers en même temps. Chaque fichier source est converti en un fichier gz indépendant.

La commande

```
gzip exemple_1.txt exemple_2.txt exemple_3.txt
```

génère les fichiers exemple_1.txt.gz, exemple_2.txt.gz et exemple_3.txt.gz.

Si vous voulez écrire plusieurs fichiers dans une archive compressée commune, utilisez gzipavec le programme d'archivage tar.

Pour décompresser un fichier gz, utilisez la commande gzip avec l'option -d. Autrement, vous pouvez utiliser la commande gunzip.

```
gzip -d exemple.txt.gz
```

```
gunzip exemple.txt.gz
```

Les fichiers gz décompressés sont également supprimés par défaut. Si vous voulez conserver le fichier gz en plus du fichier extrait, utilisez l'option -k.

La compression gzip est basée sur l'algorithme Deflate (une combinaison d'encodage Huffman et LZ77). Par rapport à d'autres méthodes de compression, gzip se caractérise par sa rapidité. Cependant, le niveau de compression est relativement faible.

bzip2 / Compression et décompression de fichiers avec bzip

bunzip2

Une alternative populaire à gzip est le programme en ligne de commande bzip2. Il utilise la même syntaxe que gzip, mais est basé sur un processus de compression en trois étapes qui permet un niveau de compression bien plus élevé.

Tout d'abord, les fichiers donnés sont sujet à la transformation Burrows-Wheeler puis move-to-front. Enfin, la compression réelle des données s'effectue dans le cadre d'un codage Huffman.

Les fichiers compressés avec bzip2 vont porter l'extension.bz2. Voici le schéma pour compresser les fichiers :

```
bzip2 [OPTIONS] FICHER(S)
```

bzip2 peut également être appliqué aux archives tar.

La décompression se fait de la même manière que gzip avec l'option -d. Autrement, vous pouvez avoir recours à la commande bunzip2.

Le haut niveau de compression se répercute sur une certaine lenteur d'exécution.

xz Compresser et décompresser des fichiers avec xz

Le programme en ligne de commande xz convertit les fichiers au format de compression de données du même nom xz. L'appel de programme suit le même schéma que pour gzip et bzip2.

```
xz [OPTIONS] FICHER(S)
```

La décompression se fait avec l'option -d ou avec la commande unxz.

Comme les fichiers gz et bz2, les fichiers xz ne sont pas des fichiers d'archive. Si vous voulez écrire plusieurs fichiers dans le même fichier xz compressé, vous devez également utiliser l'outil d'archivage tar.

xz prend en charge différents algorithmes de compression. L'algorithme Lempel-Ziv-Markov est généralement utilisé comme standard (LZMA/LZMA2).

cpio Écrire et extraire dans un fichier d'archive

cpio (abréviation de copy in, copy out) est un programme d'archivage avec lequel les données peuvent être écrites et extraites d'un fichier d'archives (.cpio).

mount Monter les systèmes de fichiers

/unmount Si un système de fichiers doit être intégré dans la structure de répertoire du système

d'exploitation, le programme en ligne de commande mount est utilisé sous Linux.

La syntaxe générale de la commande est :

mount [OPTIONS] PERIPHERIQUE MOUNTPPOINT

PERIPHERIQUE = Le chemin d'accès du périphérique de stockage que vous voulez monter en tant que partition.

MOUNTPPOINT = L'emplacement de la structure de répertoire de votre système d'exploitation où vous voulez monter la partition. Le point de montage est généralement spécifié comme un chemin absolu.

Exemple :

mount /dev/sdd /media/usb

Le périphérique sdd est monté dans le répertoire /media/usb.

En règle générale, Linux reconnaît automatiquement le système de fichiers de l'appareil. Si ce n'est pas le cas, l'option -t vous permet de spécifier explicitement le système de fichiers (par exemple ext4) :

mount -t ext4 /dev/sdd /media/usb

Si un système de fichiers déjà intégré doit être démonté, la commande unmount est utilisée :
unmount [OPTIONS] PERIPHERIQUE

ou

unmount [OPTIONS] MOUNTPPOINT

Si vous voulez afficher tous les systèmes de fichiers qui sont inclus dans votre système d'exploitation, utilisez la commande mount avec l'option -l.

mount -l

La sortie peut être limitée aux systèmes de fichiers d'un certain type via -t.

lsblk

Liste des informations sur les périphériques de stockage de type bloc connectés

Utilisez la commande lsblk (abréviation de list block devices) pour afficher tous les périphériques de stockage de type bloc et partitions connectés sous forme d'arborescence.

L'appel est basé sur la syntaxe suivante :

lsblk [OPTIONS]

La sortie comprend les informations suivantes :

NAME = Nom de l'appareil (par ex. sda) ou nom de la partition (par ex. sda1, sda2 etc.)

MAJ:MIN = major:minor Numéro de l'appareil

RM = support amovible (1 = applicable, 0 = non applicable)

SIZE = capacité de mémoire de l'appareil

RO = dispositif de lecture seule (1 = vrai, 0 = faux)

TYPE = type de périphérique

MOUNTPPOINT = point de montage

Si nécessaire, la sortie peut être personnalisée à l'aide de l'option -o (--output) tout comme la liste des attributs souhaités pour récupérer des informations supplémentaires telles que le numéro d'identification (UUID), le système de fichiers (FSTYPE) ou l'état (STATE).

Exemple :

lsblk -o NAME,FSTYPE,UUID, SIZE,OWNER,GROUP,MODE,TYPE,MOUNTPPOINT

Les périphériques de stockage vides sont ignorés par défaut. Si vous voulez les inclure tout de même dans votre visualisation, utilisez lsblk en combinaison avec l'option -a (--all).

Si vous voulez seulement récupérer des informations sur un périphérique spécifique, utilisez lsblk selon le schéma suivant :

lsblk [OPTIONS] PERIPHERIQUE

Exemple :

lsblk /dev/sda

blkid

Liste des informations sur les périphériques de stockage de type bloc connectés

Semblable à lsblk, blkid fournit également des informations sur les périphériques de stockage de type bloc connectés.

Utilisez blkid selon la syntaxe ci-dessous pour récupérer le numéro d'identification (UUID) et le type de système de fichiers (TYPE) de tous les périphériques de stockage en bloc connectés.

blkid [OPTIONS]

Pour obtenir un tableau, utilisez l'option -o avec la valeur list.

blkid -o list

blkid peut également être limité à un appareil sélectionné :

blkid [OPTIONS] PERIPHERIQUE

Exemple :

blkid /dev/sda1

dd

Copie au bit près de fichiers, partitions ou supports de données

Le programme en ligne de commande dd permet un processus de copie dans lequel les données sont lues bit par bit à partir d'un fichier d'entrée (if) et écrites dans un fichier de sortie (of).

La syntaxe pour appeler le programme est la suivante :

dd if=Source of=Cible [OPTIONS]

La source et la cible données peuvent très bien être des fichiers individuels, tout comme des partitions entières (par ex. /dev/sda1) ou un périphérique de stockage complet (par ex. /dev/sda).

Exemple :

dd if=/dev/sda5 of=/dev/sdb1

La cinquième partition entière de /dev/sda est copiée au bit près sur la première partition de /dev/sdb.

Les options permettent de limiter le processus de copie à un nombre de blocs de mémoire de la taille désirée.

alias

Définir des diminutifs pour utiliser les programmes

Le programme en ligne de commande alias permet de définir des diminutifs pour appeler des programmes.

Utilisez alias selon la syntaxe suivante :

alias DIMINUTIF= 'COMMANDE'

La mention 'COMMANDE' ci-dessus peut alors être remplacée par n'importe quel programme en ligne de commande avec ses options comprises. La commande et ses options seront alors liées à la chaîne de caractères utilisée dans DIMINUTIF.

Exemple :

alias ll='ls -l'

La chaîne ll est définie comme alias pour la commande ls avec l'option -l (sortie détaillée).

at

Exécuter une commande au moment voulu

Pour exécuter une commande à un temps précis, appeler le programme en ligne de commande at selon la syntaxe suivante :

at TEMPS

Exemple :

at 10:00 AM 6/22/2017

Entrez ensuite la commande à exécuter et fermez le mode interactif avec [CTRL] +[D].

cal

Afficher le calendrier

Utilisez cal selon le schéma suivant pour afficher un calendrier dans le terminal :

cal [OPTIONS] [[MOIS] Année]

Exemple :

cal 12 2017

Le système affiche une synthèse mensuelle pour décembre 2017.

echo

Émission de chaînes de caractère sur la sortie standard

Utilisez la commande echo pour émettre des chaînes de caractères (strings) ligne par ligne sur la sortie standard (généralement le terminal).

La syntaxe générale de la commande est :

echo [OPTIONS] STRING

pr

Préparer les fichiers texte pour impression

Utilisez le programme en ligne de commande pr pour préparer les fichiers texte à l'impression.

La syntaxe générale de la commande est :

pr [OPTIONS] Fichier

Par défaut, pr crée un en-tête de page contenant le nom du fichier, la date du jour et le numéro de page.

script

Enregistrer les sessions du terminal

Le programme en ligne de commande script vous permet d'écrire une session de terminal dans le fichier typescript. Si typescript contient déjà un enregistrement d'une session précédente, il sera écrasé.

L'enregistrement démarre automatiquement lorsque le programme est appelé :

script

Utilisez les touches [CTRL] + [D] pour arrêter l'enregistrement.

Si vous voulez enregistrer dans un fichier autre que typescript, utilisez script avec un nom de fichier ou un chemin d'accès.

script FICHIER

seq

Affichage de séquences de nombres

Utilisez la commande seq pour émettre une série de chiffres sur la sortie par défaut. Définissez une valeur de début, une valeur de fin et éventuellement un incrément.

seq [OPTIONS] VALEURDEBUT INCREMENT VALEURFIN

Exemple :

seq 0 2 10

A partir de la valeur de 0, le programme compte par pas de deux jusqu'à 10.

tasksel

Aide à l'installation pour les applications standard

Le programme en ligne de commande tasksel sert d'aide à l'installation pour les applications standard (serveur de messagerie, DNS, OpenSSH, LAMP, etc. Utilisez cet outil pour installer automatiquement et dans l'ordre approprié tous les paquets et programmes requis pour une tâche.

Appelez tasksel avec l'option --list-tasks pour obtenir une liste de toutes les applications standard disponibles.

tasksel --list-tasks

Si vous voulez obtenir plus d'informations sur une application standard de cette liste, utilisez tasksel avec l'option --task-desc et la tâche correspondante.

Exemple :

tasksel --task-desc mail-server

Les informations sur la tâche « mail-server » sont affichées.

Si vous voulez lister tous les paquets appartenant à la tâche « mail-server », utilisez tasksel avec l'option --task-packages.

tasksel --task-packages mail-server

Pour installer tous les paquets d'une application standard, utilisez la sous-commande install. Des droits root sont nécessaires.

Exemple :

tasksel install mail-server

Le programme en ligne de commande démarre l'installation de tous les paquets nécessaires

à la tâche « mail-server ».

tee	Doubler la sortie du programme tee est un programme en ligne de commande qui peut être utilisé pour doubler la sortie d'un programme. Une sortie est émise à la sortie standard et une autre est écrite dans un fichier stipulé avec la commande tee. La syntaxe générale de la commande est la suivante : tee [OPTIONS] FICHIER En règle générale, tee est utilisé avec la redirection pipe (). ls tee exemple.txt La commande ls liste le contenu du répertoire courant. La sortie du programme est transférée via le pipe au programme en ligne de commande tee, qui l'affiche dans le terminal mais aussi l'écrit dans le fichier exemple.txt.
time	Mesure de la durée d'exécution des programmes Utilisez la commande time selon le schéma suivant pour déterminer la durée d'exécution des programmes que vous lancez via le terminal. time [OPTIONS] Commande [ARGUMENTS]
tr	Remplacer des caractères dans les fichiers texte Utilisez tr pour supprimer ou remplacer autant de caractères que souhaité par d'autres. Pour ce faire, tr lit le flux de données de l'entrée standard et l'écrit sur la sortie standard selon la modification souhaitée. Si un jeu de caractères doit être remplacé par un autre, tr est utilisé avec deux arguments. tr OPTION JEUDECARACTERE1 JEUDECARACTERE2 Le deuxième argument (JEUDECARACTERE1) remplace le premier (JEUDECARACTERE1). Si vous voulez supprimer une chaîne de caractères, utilisez tr avec l'option -d et donnez comme argument la séquence à supprimer. tr -d JEUDECARACTERE Le programme en ligne de commande est généralement utilisé en combinaison avec des opérateurs de redirection (< et >) pour ajuster les fichiers. tr 'a-z' 'A-Z' < exemple1.txt > exemple2.txt tr lit le contenu du fichier exemple1.txt, remplace les lettres minuscules de a à z par des lettres majuscules et émet le résultat dans le fichier exemple2.txt.
wall	Envoi de messages à tous les utilisateurs connectés Le programme en ligne de commande wall vous permet d'envoyer un message à tous les utilisateurs connectés au système. Pour envoyer un message, le programme commence par : wall Confirmez l'appel du programme avec [Enter] et écrivez votre message. Confirmez également avec [Enter] et validez l'envoi en appuyant sur les touches [CTRL]+[D]. Tous les utilisateurs connectés au système reçoivent votre message dans le terminal sous forme de broadcast message. Attention : pour recevoir des messages, vous devez autoriser les autres utilisateurs à accéder en écriture à votre terminal. Pour ce faire, utilisez la commande mesg : mesg [y/n] Statut actuel : mesg Accorder l'accès : mesg y Refuser l'accès : mesg n Si vous voulez envoyer le contenu d'un fichier à tous les utilisateurs connectés, utilisez wall avec une touche de redirection et le nom de fichier correspondant : wall < NOMFICHIER

watch Exécution de commandes à intervalle régulier
Le programme en ligne de commande watch vous permet d'exécuter une commande à intervalles réguliers.
La syntaxe pour le programme est la suivante :
watch [OPTIONS] COMMANDE
L'intervalle de temps est défini à l'aide de l'option -n SECONDES.
Pour mettre fin à watch, appuyez sur [CTRL] + [C].
Voici un exemple sur l'affichage de la disponibilité de la mémoire vive à intervalle de 10 secondes.
watch -n 10 free

wc Compter les lignes, les mots, les lettres, les caractères et/ou les octets des fichiers texte
Le programme en ligne de commande wc (abréviation de word count) affiche le nombre de lignes, de mots, de lettres, de caractères et/ou d'octets d'un fichier texte.
La syntaxe générale de la commande est :
wc [OPTIONS] FICHIER
Exemple :
wc exemple.txt
Sortie :
14 18 143 exemple.txt
Si wc est utilisé sans option, la sortie correspond au schéma LIGNES MOTS CARACTÈRES FICHIER. Pour filtrer la sortie, le programme en ligne de commande permet les options suivantes : -l (lignes), -c (octets), -m (caractères), -L (longueur de la ligne la plus longue) et -w(mots).

xargs Convertir l'entrée standard en lignes de commande
xargs est utilisé pour passer une sortie de commande précédente en argument à une commande. En général, on fait appel au symbole (|).
Voici la syntaxe générale de xargs :
COMMANDE1 | xargs [OPTIONS] COMMANDE2
xargs est par exemple utilisé avec la commande find.
Ci-dessous, find identifie tous les fichiers du répertoire courant qui correspondent au modèle de recherche *.tmp et émet leurs noms dans la sortie standard. Les noms de fichiers sont alors captés par xargs et passés en arguments à la commande rm.
\$ find . -name '*.tmp' | xargs rm

sort Trier les entrées puis afficher les résultats sur un moniteur ou les écrire dans un fichier.
\$ sort test.txt : afficherait les résultats triés du texte de test.txt.
\$ sort : afficherait les résultats triés du texte entrée par le clavier.
NB : Sans argument : Croissant

Option	Signification
-b	Saute les colonnes constituées de blancs.
-d	Trie de type dictionnaire.
-n	Trie par ordre numérique.
-f	Aucune différenciation n'est faite entre minuscules et majuscules.
-b	Ignore les espaces placés en début de champ.
-r	Trie inverse.
-M	Trie chronologiquement les mois.
-t:	Trie suivants les champs séparés par les caractères deux points (`:`').

ln

Créer des liens entre fichiers.

ln [options] [source] [dest]

-f, --force

Forcer l'écrasement du fichier destination s'il existe.

-s, --symbolic

Créer des liens symboliques à la place des liens physiques. Cette option produit souvent des messages d'erreurs sur les systèmes qui ne supportent pas les liens symboliques.