

La Ligne de Commande Shell

La ligne de commande shell est un outil puissant pour les développeurs et les administrateurs système. Voici un guide des commandes shell et des commandes usuelles pour Linux (et souvent macOS).

```
override@Atul-HP:~$ ls -l
total 212
drwxr-xr-x  5 override override 4096 May 19 03:45 acadenv
drwxr-xr-x  2 override override 4096 May 27 15:30 acadenv_demo
drwxr-xr-x 12 override override 4096 May 3 15:14 anaconda3
drwxr-xr-x  6 override override 4096 May 31 16:49 Desktop
drwxr-xr-x  2 override override 4096 Oct 21 2016 Documents
drwxr-xr-x  7 override override 4096 Jun 1 13:09 Downloads
-rw-r--r--  1 override override 8980 Aug 1 2016 examples.desktop
-rw-r--r--  1 override override 20880 May 20 08:45 hs_err_pid2000.log
-rw-r--r--  1 override override 45147 Jun 1 03:24 hs_err_pid2001.log
drwxr-xr-x  2 override override 4096 Mar 2 18:22 Music
drwxr-xr-x 21 override override 4096 Dec 25 00:13 Mydata
drwxr-xr-x  2 override override 4096 Sep 20 2016 newbin
drwxr-xr-x  2 override override 4096 May 31 08:46 Pictures
drwxr-xr-x  4 override override 4096 May 31 08:46 Public
drwxr-xr-x  2 override override 4096 Aug 8 2016 Public
drwxr-xr-x  2 override override 4096 May 31 19:49 scripts
drwxr-xr-x  2 override override 4096 Aug 1 2016 Templates
drwxr-xr-x  2 override override 4096 May 31 14:49 Videos
drwxr-xr-x  2 override override 4096 Mar 11 13:27 Videos
drwxr-xr-x  2 override override 4096 Sep 1 2016 xdm-helper
override@Atul-HP:~$
```

1. Navigation dans le Système de Fichiers

```
/home/hannan/ron
hannan@hannan-Latitude-E5470:~/fun$ cd
hannan@hannan-Latitude-E5470: $ mkdir -p dir1/dir2/dir3
hannan@hannan-Latitude-E5470: $ cd dir1
hannan@hannan-Latitude-E5470:~/dir1$ ls
dir2
hannan@hannan-Latitude-E5470:~/dir1$ cd dir2
hannan@hannan-Latitude-E5470:~/dir1/dir2$ ls
dir3
hannan@hannan-Latitude-E5470:~/dir1/dir2$ cd dir3
hannan@hannan-Latitude-E5470:~/dir1/dir2/dir3$ pwd
/home/hannan/ron/dir1/dir2/dir3
hannan@hannan-Latitude-E5470:~/dir1/dir2/dir3$
```

a. `pwd` (Print Working Directory)

Affiche le chemin du répertoire actuel.

```
pwd
```

b. `ls` (List)

Affiche le contenu d'un répertoire.

```
ls
ls -l # Affiche les détails
ls -a # Affiche les fichiers cachés
ls -lh # Affiche les détails avec des tailles lisibles par l'homme
```

c. cd (Change Directory)

Change de répertoire.

```
cd /path/to/directory # Change vers le répertoire spécifié  
cd .. # Monte d'un niveau  
cd ~ # Va dans le répertoire personnel de l'utilisateur
```

2. Gestion des Fichiers et Répertoires

```
hannan@hannan-Latitude-E5470:~/fun$ cd  
hannan@hannan-Latitude-E5470: $ mkdir -p dir1/dir2/dir3  
hannan@hannan-Latitude-E5470: $ cd dir1  
hannan@hannan-Latitude-E5470:~/dir1$ ls  
dir2  
hannan@hannan-Latitude-E5470:~/dir1$ cd dir2  
hannan@hannan-Latitude-E5470:~/dir1/dir2$ ls  
dir3  
hannan@hannan-Latitude-E5470:~/dir1/dir2$ cd dir3  
hannan@hannan-Latitude-E5470:~/dir1/dir2/dir3$ pwd  
/home/hannan/dir1/dir2/dir3  
hannan@hannan-Latitude-E5470:~/dir1/dir2/dir3$ █
```

a. cp (Copy)

Copie des fichiers ou des répertoires.

```
cp source.txt destination.txt # Copie un fichier  
cp -r source_directory/ destination_directory/ # Copie un répertoire et son contenu
```

b. mv (Move)

Déplace ou renomme des fichiers ou des répertoires.

```
mv oldname.txt newname.txt # Renomme un fichier  
mv file.txt /path/to/directory/ # Déplace un fichier
```

c. rm (Remove)

Supprime des fichiers ou des répertoires.

```
rm file.txt # Supprime un fichier  
rm -r directory/ # Supprime un répertoire et son contenu
```

d. mkdir (Make Directory)

Crée un nouveau répertoire.

```
mkdir new_directory
```

e. **touch**

Crée un nouveau fichier vide ou met à jour l'horodatage d'un fichier existant.

```
touch newfile.txt
```

3. Affichage et Recherche de Fichiers

| 80+ ESSENTIAL LINUX COMMANDS | | | | |
|------------------------------|-------------------------|-----------------------|-------------------------|-------------|
| 1. pwd | 21. Useradd & Userdel | 41. fdisk | 61. sudo | 81. crontab |
| 2. ls | 22. Usermod | 42. ps | 62. Sudo | 82. wget |
| 3. cat | 23. groupadd & groupdel | 43. Top | 63. umask | 83. curl |
| 4. touch | 24. passwd | 44. htop | 64. getfacl | 84. wc |
| 5. mkdir | 25. gpasswd | 45. prep | 65. setfacl | 85. export |
| 6. cd | 26. find | 46. pkill and killall | 66. stat | |
| 7. rm | 27. locate | 47. kill | 67. who | |
| 8. rmdir | 28. whereis | 48. kof | 68. whoami | |
| 9. cp | 29. grep -D | 49. strace | 69. w | |
| 10. mv | 30. sed | 50. hostname | 70. last | |
| 11. file | 31. awk | 51. date | 71. getent | |
| 12. vi/vim/nano | 32. sort | 52. systemctl | 72. ip addr | |
| 13. more and less | 33. cut | 53. journalctl | 73. netstat -tuln | |
| 14. head and tail | 34. diff | 54. timedatectl | 74. ifconfig | |
| 15. cat | 35. tar | 55. shutdown | 75. ping | |
| 16. zip & unzip | 36. df | 56. reboot | 76. traceroute | |
| 17. gzip & gunzip | 37. du | 57. history | 77. mtr | |
| 18. tar | 38. mount | 58. clear | 78. nslookup | |
| 19. chmod | 39. free | 59. exit | 79. dmesg grep eth0 | |
| 20. chown & chgrp | 40. lsblk | 60. man | 80. yum/apt-get/dnf/rpm | |

a. **cat (Concatenate)**

Affiche le contenu d'un fichier.

```
cat file.txt
```

b. **less et more**

Affiche le contenu d'un fichier une page à la fois.

```
less file.txt  
more file.txt
```

c. **head et tail**

Affiche le début ou la fin d'un fichier.

```
head -n 10 file.txt # Affiche les 10 premières lignes  
tail -n 10 file.txt # Affiche les 10 dernières lignes
```

d. grep (Global Regular Expression Print)

Recherche des motifs dans un fichier.

```
grep "search_term" file.txt  
grep -r "search_term" /path/to/directory # Recherche récursive
```

4. Gestion des Processus



```
tecmin@tecmin:~$ ps -ef | grep mariadb  
mysql      3383     1  0 08:40 ?        00:00:00 /usr/sbin/mariadbd  
tecmin      3475   2115  0 08:50 pts/3    00:00:00 grep mariadb  
tecmin@tecmin:~$ sudo kill -9 3383  
tecmin@tecmin:~$  
tecmin@tecmin:~$ ps -ef | grep mariadb  
tecmin      3484   2115  0 08:50 pts/3    00:00:00 grep mariadb  
tecmin@tecmin:~$
```

a. ps (Process Status)

Affiche les processus en cours d'exécution.

```
ps  
ps aux # Affiche tous les processus avec plus de détails
```

b. top et htop

Affiche en temps réel les processus en cours et leur utilisation des ressources.

```
top  
htop # Nécessite une installation préalable : sudo apt-get install htop (Linux)
```

c. kill

Envoie un signal à un processus pour le terminer.

```
kill PID # Termine le processus avec l'ID donné  
kill -9 PID # Force la terminaison du processus
```

5. Gestion des Permissions et Propriétés

```
zaira@Zaira:~/FreeCodeCamp$ ls -l
total 3856
-rw-r--r--  1 zaira zaira    89 Apr  5 20:46 CODE_OF_CONDUCT.md
-rw-r--r--  1 zaira zaira   210 Apr  5 20:46 CONTRIBUTING.md
-rw-r--r--  1 zaira zaira  1513 Apr  5 20:46 LICENSE.md
-rw-r--r--  1 zaira zaira 19933 Apr  5 20:46 README.md
drwxr-xr-x  4 zaira zaira  4096 Apr  6 22:45 api-server
drwxr-xr-x  1 zaira zaira   67 Apr  5 20:46 babel.config.js
drwxr-xr-x 10 zaira zaira  4096 Apr  6 22:55 client
drwxr-xr-x  5 zaira zaira  4096 Apr  6 22:54 config
```

MODE OWNER GROUP SIZE MODIFICATION DATE FILE/FOLDER NAME

a. chmod (Change Mode)

Change les permissions d'un fichier ou d'un répertoire.

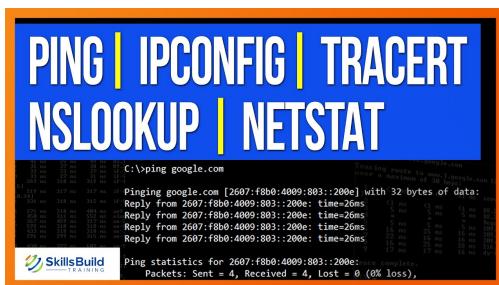
```
chmod 755 file.txt # Définit les permissions à rwxr-xr-x
```

b. chown (Change Owner)

Change le propriétaire d'un fichier ou d'un répertoire.

```
chown user:group file.txt
```

6. Réseau



a. ping

Teste la connectivité réseau avec une adresse IP ou un nom de domaine.

```
ping google.com
```

b. ifconfig et ip

Affiche et configure les interfaces réseau.

```
ifconfig  
ip a
```

c. curl et wget

Télécharge des fichiers depuis le web.

```
curl -O http://example.com/file.txt  
wget http://example.com/file.txt
```

7. Compression et Archivage



a. tar

Archive des fichiers et répertoires.

```
tar -cvf archive.tar directory/ # Crée une archive  
tar -xvf archive.tar # Extrait une archive
```

b. zip et unzip

Compresse et décompresse des fichiers.

```
zip -r archive.zip directory/  
unzip archive.zip
```

8. Utilitaires Divers

```
vivek@ntxcraft-asus:~$ date
Tue Dec 18 11:08:15 IST 2018
vivek@ntxcraft-asus:~$ now=$(date)
vivek@ntxcraft-asus:~$ echo "Snow"
Tue Dec 18 11:08:20 IST 2018
vivek@ntxcraft-asus:~$ printf "%s\n" $now
Tue
Dec
18
11:08:20
IST
2018
vivek@ntxcraft-asus:~$ printf "%s\n" "Snow"
Tue Dec 18 11:08:20 IST 2018
vivek@ntxcraft-asus:~$ backup_dir=${date%/*}
vivek@ntxcraft-asus:~$ echo "Backup dir for today: /nas04/backups/${backup_dir}"
Backup dir for today: /nas04/backups/12/18/2018
vivek@ntxcraft-asus:~$
```

© www.cyberciti.biz

a. echo

Affiche une ligne de texte.

```
echo "Hello, World!"
```

b. date

Affiche ou définit la date et l'heure du système.

```
date
```

c. man (Manual)

Affiche le manuel d'utilisation d'une commande.

```
man ls
```

9. Scripts Shell



```
newscript.sh
1 #!/bin/bash
2
3 function greeting() {
4     hello="Hello, $name"
5     echo "$hello"
6 }
7
8 echo "Enter name"
9 read name
10
11 val=$(greeting)
12 echo "Return value of the function is $val"
```

Vous pouvez créer des scripts shell pour automatiser des tâches.

Exemple de script shell (script.sh):

```
#!/bin/bash
# Script pour afficher un message de bienvenue
echo "Bienvenue, $(whoami)!"
echo "Nous sommes le $(date)."
```

Rendre le script exécutable et l'exécuter :

```
chmod +x script.sh
./script.sh
```

Conclusion

En maîtrisant ces commandes, vous serez en mesure de naviguer, gérer des fichiers, contrôler des processus, et bien plus encore, directement depuis la ligne de commande.