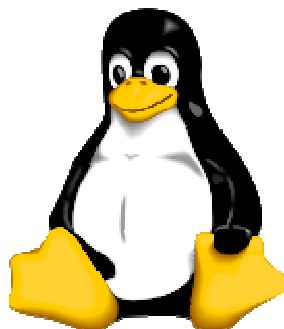

MANIPULATION DE FICHIERS SOUS LINUX

(COMMANDES DE BASE)



DR. CHAFIKA BENZAID

MAITRE DE CONFERENCES,
CLASSE B

LSI-Département
Informatique,
Faculté d'Electronique &
Informatique, USTHB

Email: benzaid@hotmail.com

MANEL CHENAIT

CHARGEЕ DE COURS

LSI-Département
Informatique,
Faculté d'Electronique &
Informatique, USTHB

Email: chenait@gmail.com

BAHIA ZEBBANE

CHARGEЕ DE COURS

LSI-Département
Informatique,
Faculté d'Electronique &
Informatique, USTHB

Email: zebbane@gmail.com

Sommaire

COMMANDES DE GESTION DES REPERTOIRES ET DES FICHIERS.....	3
Répertoires	3
Création : La commande mkdir	3
Affichage :	3
La commande pwd	3
La commande ls	3
Déplacement :.....	4
La commande cd	4
Copier : La commande cp -r	4
Renommage :	5
La commande mv	5
Suppression :	5
La commande rmdir	5
La commande rm	5
 Fichiers	6
Création :.....	6
La commande touch	6
La commande cat	6
Affichage :	6
La commande cat	6
La commande head	7
La commande tail	7
La commande more	7
La commande less	7
La commande file	8
Copier :La commande cp	8
Suppression : La commande rm	8
Renommage :La commande mv	9
Comparaison :.....	9
La commande cmp	9
La commande diff	9
Recherche :.....	10
La commande grep	10
La commande find	10
Tri :La commande sort	11
Archivage :La commande tar	11
Test :La commande test	12
 Liens	12
Lien physique	12
La commande ln	12
Lien symbolique	13
La commande ln -s	13
 COMMANDES DE GESTION DES ATTRIBUTS DES FICHIERS ET REPERTOIRES	13
Droits d'accès	13
La commande chmod	13
La commande umask	13
Propriétaire	14
La commande chown	14
Groupe	14
La commande chgrp	14

Commandes de gestion des répertoires et des fichiers

Répertoires

Création : La commande **mkdir** <nom-répertoire>

Crée un répertoire ou une liste de répertoires (**make directory**).

Exemple

```
[root@localhost ~]# mkdir Rep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog  Rep
[root@localhost ~]#
```

Affichage :

La commande **pwd**

Affiche le chemin absolu du répertoire courant (**print working directory**).

Exemple

```
[root@localhost ~]# pwd
/root
[root@localhost ~]#
```

La commande **ls**

Affiche (en ordre alphabétique par défaut) le contenu d'un répertoire (càd. les répertoires et les fichiers du répertoire courant). Elle permet aussi l'affichage des informations sur un ou plusieurs fichiers suivant les options :

ls	Affiche seulement les noms
ls nom_répertoire	Afficher les noms des fichiers et des sous répertoires du répertoire indiqué
ls -l	Affiche les informations sur chaque fichier et sous répertoire (affiche le format long : Type+droits+Nbre_de_liens+user+group+taille+date+heure+nom)
ls -a	Affiche les fichiers cachés
ls -la	Affichage long avec les fichiers cachés
ls -al	Affiche toute l'information
ls -i	Affiche les numéros d'inode des fichiers
ls -li	Affichage le plus complet (numéro_inode+infomations)
ls -R	Affiche récursivement les sous répertoires
ls -x	Affichage sur plusieurs colonnes
ls -t	Affichage par date de modification (dans l'ordre croissant)
ls -S	Affichage par taille (dans l'ordre décroissant)
ls -r	Inverse l'ordre d'affichage
ls -F	affiche le type de fichier dans l'ordre alphabétique
ls --help	Affiche l'aide en ligne

Exemple

```
[root@localhost ~]# ls -l
total 80
-rw----- 1 root root 1098 Feb 16 2008 anaconda-ks.cfg
drwxr-xr-x 3 root root 4096 Nov 3 19:40 Desktop
-rw-r--r-- 1 root root 39819 Feb 16 2008 install.log
-rw-r--r-- 1 root root 4593 Feb 16 2008 install.log.syslog
drwxr-xr-x 2 root root 4096 Nov 3 20:11 Rep

[root@localhost ~]# ls -li
4620297 anaconda-ks.cfg 4620290 install.log 4621227 Rep
4620325 Desktop 4620291 install.log.syslog
[root@localhost ~]#
```

Déplacement :

La commande `cd`

Change de répertoire courant (change directory) .

<code>cd</code>	Retour au répertoire de login
<code>cd /</code>	Retour au répertoire racine
<code>cd chemin_absolu</code>	Aller au répertoire indiqué dans le chemin absolu
<code>cd chemin_relatif</code>	Aller au répertoire indiqué
<code>cd ..</code>	Retour au répertoire père
<code>cd ../../</code>	Remonter de deux niveaux (aller au répertoire père du répertoire père du répertoire courant)
<code>cd ~</code>	Aller au répertoire de base

Exemple

```
[root@localhost ~]# cd Desktop/
[root@localhost Desktop]# cd ..
[root@localhost ~]# cd Desktop/
[root@localhost Desktop]# ls
foo-00ee7f7571.desktop  fumeur.c~  one.c  Screenshot.eps
fumeur~                larry-00eff57bf1.desktop  one.c~  Screenshot.png
fumeur.c               one        rep
[root@localhost Desktop]# cd rep/
[root@localhost rep]# cd ../../
[root@localhost ~]# cd Desktop/rep
[root@localhost rep]# cd ~
[root@localhost ~]#
```

Copier : La commande `cp -r <répertoire_source> <répertoire_destination>`

Copie un répertoire vers un autre répertoire (l'option `-R` ou `-r` pour récursif).

Exemple

```
[root@localhost ~]# mkdir newRep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog  newRep  Rep
[root@localhost ~]# cp -r Rep newRep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog  newRep  Rep
[root@localhost ~]# cd newRep
[root@localhost newRep]# ls
Rep
[root@localhost newRep]#
```

Renommage :

La commande mv **<source>** **<destination>**

Renomme un répertoire ou le déplace dans la hiérarchie.

Exemple

```
[root@localhost newRep]# mv Rep Repertoire
[root@localhost newRep]# ls
Repertoire
[root@localhost ~]# mkdir nouveauRep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog  newRep
nouveauRep
[root@localhost ~]# mv newRep nouveauRep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog  nouveauRep
[root@localhost ~]# cd nouveauRep/
[root@localhost nouveauRep]# ls
newRep
[root@localhost nouveauRep]# cd newRep/
[root@localhost newRep]# ls
Repertoire
[root@localhost newRep]#
```

Suppression :

La commande rmdir **<nom-repertoire>**

Supprime un répertoire vide ou une liste de répertoires vides (**remove directory**).

Exemple

```
[root@localhost newRep]# rmdir Repertoire/
[root@localhost newRep]# ls
[root@localhost newRep]#
```

La commande rm **<nom-repertoire>**

Permet de supprimer un répertoire et tout son contenu si elle est utilisée avec les options suivantes :

rm -r ou rm -R	permet de détruire un répertoire et tous ses fichiers.
rm -i	permet de détruire un répertoire et tous ses fichiers avec demande de confirmation pour chaque étape.
rm -f	Permet de détruire un répertoire et tous ses fichiers sans demande de confirmation pour chaque étape (forcer la destruction).

Exemple

```
[root@localhost ~]# rm -r nouveauRep/
rm: descend into directory `nouveauRep/'? n
[root@localhost ~]# rm -ir nouveauRep/
rm: descend into directory `nouveauRep/'? y
rm: remove directory `nouveauRep//newRep'? y
rm: remove directory `nouveauRep/'? y
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  install.log  install.log.syslog
[root@localhost ~]#
```

Fichiers

Création :

La commande **touch <Fichier>**

Permet de créer un fichier vide s'il n'existe pas. Elle permet aussi de changer la date de modification d'un fichier ou d'un répertoire.

Exemple

```
[root@localhost ~]# touch Fichier1
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Fichier1  install.log  install.log.syslog
[root@localhost ~]#
```

La commande **cat**

Permet de créer un fichier de type texte. L'option **-n** de la commande **cat** permet de numéroté les lignes du fichier crée. Cette commande peut être utilisée pour concaténer deux fichiers.

Exemple

```
[root@localhost ~]# cat>Fichier2
c'est un exemple de creation de fichier avec la commande cat
[root@localhost ~]# cat>Fichier1
Bonjour,
c'est la section C

[root@localhost ~]# cat Fichier1
Bonjour,
c'est la section C
[root@localhost ~]# cat>> Fichier1
c'est la section B
c'est la section A

[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Fichier1  Fichier2  install.log
install.log.syslog
[root@localhost ~]# ls -l
total 92
-rw----- 1 root root 1098 Feb 16 2008 anaconda-ks.cfg
drwxr-xr-x 3 root root 4096 Nov 3 21:08 Desktop
-rw-r--r-- 1 root root 0 Nov 3 21:08 Fichier1
-rw-r--r-- 1 root root 61 Nov 3 21:11 Fichier2
-rw-r--r-- 1 root root 39819 Feb 16 2008 install.log
-rw-r--r-- 1 root root 4593 Feb 16 2008 install.log.syslog
[root@localhost ~]#
```

Affichage :

La commande **cat <Fichier>**

Affiche le contenu d'un ou plusieurs fichiers de type texte.

Exemple

```
[root@localhost ~]# cat Fichier2
c'est un exemple de creation de fichier avec la commande cat
[root@localhost ~]# cat Fichier1
Bonjour,
c'est la section C
```

```
c'est la section B
c'est la section A
[root@localhost ~]#
```

La commande **head** <Fichier>

Affiche les n premières lignes d'un fichier de type texte (n=10 par défaut).

Exemple

```
[root@localhost ~]# head -2 Fichier1
Bonjour,
c'est la section C
[root@localhost ~]# head -3 Fichier1
Bonjour,
c'est la section C
c'est la section B
[root@localhost ~]#
```

La commande **tail** <Fichier>

Affiche les n dernières lignes d'un fichier de type texte (n=10 par défaut).

Exemple

```
[root@localhost ~]# tail Fichier2
c'est un exemple de creation de fichier avec la commande cat
[root@localhost ~]# tail -1 Fichier1
c'est la section A
[root@localhost ~]# tail -2 Fichier1
c'est la section B
c'est la section A
root@localhost ~]#
```

La commande **more** <Fichier>

Affiche page par page un fichier de type texte.

- La barre *d'espacement* permet d'afficher la page suivante ;
- La touche *b* permet d'afficher la page précédente ;
- La touche *return* permet d'afficher une nouvelle ligne en bas de l'écran ;
- La touche *q* permet de quitter ;
- La touche *h* permet d'obtenir un écran d'aide.

Exemple

```
[root@localhost ~]# more Fichier1
Bonjour,
c'est la section C
c'est la section B
c'est la section A
[root@localhost ~]#
```

La commande **less** <Fichier>

Affiche page par page avec défilement arrière.

Exemple

```
[root@localhost ~]# less Fichier2
c'est un exemple de creation de fichier avec la commande cat
[root@localhost ~]#
```

La commande **file <Fichier>**

Affiche des informations sur la nature du fichier. Elle indique si un fichier binaire est exécutable, si un fichier texte s'appuie sur une table de codage particulière (ASCII...), etc.

Exemple

```
[root@localhost ~]# file Fichier1
Fichier1: ASCII text
[root@localhost ~]# file anaconda-ks.cfg
anaconda-ks.cfg: ASCII English text
[root@localhost ~]# cd Desktop/
[root@localhost Desktop]# ls
example  foo-00ee7f7571.desktop  fumeur.c  larry-00eff57bf1.desktop
one.c    rep                    Screenshot.png
example~ fumeur~                fumeur.c~  one
one.c~   Screenshot.eps
[root@localhost ~]# file rep
rep: directory
[root@localhost Desktop]# file one.c
one.c: ASCII text
[root@localhost Desktop]# file one
one: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for
GNU/Linux 2.6.9, dynamically linked (uses shared libs), for GNU/Linux
2.6.9, not stripped
[root@localhost Desktop]#
```

Copier : La commande **cp <fichier_source> <fichier_destination>**

Copie un fichier en changeant son nom et copie une liste de fichiers vers un répertoire.

Exemples :

```
[root@localhost ~]# cp Fichier1 Fichier2
cp: overwrite `Fichier2'? y
[root@localhost ~]# cat Fichier1
Bonjour,
c'est la section C
c'est la section B
c'est la section A
[root@localhost ~]# cat Fichier2
Bonjour,
c'est la section C
c'est la section B
c'est la section A
[root@localhost ~]# cp Fichier2 Rep
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Fichier1  Fichier2  install.log
install.log.syslog  Rep
[root@localhost ~]# cd Rep
[root@localhost Rep]# ls
Fichier2
[root@localhost Rep]#
```

Suppression : La commande **rm**

Supprime une liste de fichiers. Les options importantes de cette commande sont :

rm -i	permet de détruire un fichier avec demande de confirmation.
rm -f	Permet de forcer la destruction fichier (sans demande de confirmation).

Exemple

```
[root@localhost Rep]# rm Fichier2
rm: remove regular file `Fichier2'? y
[root@localhost Rep]# ls
[root@localhost Rep]#
```

Renommage : La commande **mv <source> <destination>**

Renomme un fichier ou le déplace dans la hiérarchie.

Exemple

```
[root@localhost ~]# mv Fichier2 Fichier
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Fichier  Fichier1  install.log
install.log.syslog  Rep
[root@localhost ~]# mv Fichier Rep/Fich
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Fichier1  install.log  install.log.syslog  Rep
[root@localhost ~]# cd Rep/
[root@localhost Rep]# ls
Fich
[root@localhost Rep]#
```

Comparaison :

La commande **cmp <fichier1> <fichier2>**

Compare 2 fichiers et affiche comme résultat les numéros du premier caractère différent et de la première ligne différente.

Exemple

```
[root@localhost ~]# cat Rep/Fich
Bonjour,
c'est le section C
c'est la section B
c'est la section A
[root@localhost ~]# cmp Fichier1 Rep/Fich
Fichier1 Rep/Fich differ: byte 17, line 2
[root@localhost ~]#
```

La commande **diff <fichier1> <fichier2>**

Compare 2 fichiers et les lignes différentes sont affichées.

Exemple

```
[root@localhost ~]# cat>> Fichier1
c'est une nouvelle section
[root@localhost ~]# cat Fichier1
Bonjour,
c'est la section C
c'est la section B
```

```

c'est la section A
c'est une nouvelle section
[root@localhost ~]# diff Fichier1 Rep/Fich
2c2
< c'est la section C
---
> c'est le section C
5d4
< c'est une nouvelle section
[root@localhost ~]#

```

Recherche :

La commande **grep**

Filtre le flux de texte qu'elle reçoit pour ne laisser passer que les lignes qui contiennent le texte indiqué comme critère de filtrage (autrement dit cherche une chaîne de caractères dans un fichier). Cette commande peut être utilisée avec les options suivantes :

-i	Sans tenir compte de la casse
-C	En comptant les occurrences
-v	Inverse la recherche en excluant la « motif »

Exemple

```

[root@localhost ~]# grep "c" Fichier1
c'est la section C
c'est la section B
c'est la section A
c'est une nouvelle section
root@localhost ~]# cat> Fichier
un mot
deux mot
trois mots
[root@localhost ~]# grep "mo" Fich*
Fichier:un mot
Fichier:deux mot
Fichier:trois mots
[root@localhost ~]# grep "u" Fich*
Fichier:un mot
Fichier:deux mot
Fichier1:Bonjour,
Fichier1:c'est une nouvelle section
[root@localhost ~]# [root@localhost ~]# grep -c "u" Fich*
Fichier:2
Fichier1:2

```

La commande **find**

Recherche en partant un répertoire donné des fichiers suivant différents critères (nom, type, date,...). Les options suivantes permettent de préciser ces critères :

-name	Recherche par nom
-type	Recherche selon le type
-print	Affiche le résultat trouvé
-size	Recherche selon la taille
-newer	Recherche les objets les plus récents que le fichier de référence
-mtime,-atime,-ctime	Recherche selon la date de modification, du dernier accès ou la

	date de création
-uid, -gid	Recherche le numéro du propriétaire ou le groupe de propriétaire
-user, -group	Recherche selon le nom du propriétaire ou le groupe du propriétaire
-nouser, -nogroup	Recherche les objets n'ayant pas de propriétaire ou groupe du propriétaire
-perm	Recherche selon les permissions (droits d'accès)
-prune	Limite la recherche au niveau hiérarchie du répertoire. Aucune recherche n'est effectuée dans les sous répertoires
-exec	Exécute une commande sur chaque élément trouvé.

Exemple

```
[root@localhost ~]# find -name "*.c"
./Desktop/one.c
./Desktop/fumeur.c
[root@localhost ~]#
```

Tri : La commande **sort**

Trie les lignes d'un fichier. Les principales options sont :

-n	Le tri est numérique
-r	L'ordre est inversé
-u	N'affiche qu'une seule ligne quand plusieurs lignes sont identiques
-k	Définit une clé de tri

Exemple

```
[root@localhost ~]# sort Fichier
deux mot
trois mots
un mot
[root@localhost ~]# sort -n Fichier
deux mot
trois mots
un mot
[root@localhost ~]# sort -r Fichier
un mot
trois mots
deux mot
[root@localhost ~]# sort Fichier1
Bonjour,
c'est la section A
c'est la section B
c'est la section C
c'est une nouvelle section
[root@localhost ~]# sort -r Fichier1
c'est une nouvelle section
c'est la section C
c'est la section B
c'est la section A
Bonjour,
[root@localhost ~]#
```

Archivage : la commande **tar**

Génère un fichier contenant une arborescence complète qu'elle regroupe en un seul fichier au format tar. La syntaxe générale est

```
$ tar -options fichier_archive répertoire
```

Les principales options sont :

-c	Création de l'archive
-v	Affichage du détail des fichiers et des répertoires archivés
-x	Extraction de la hiérarchie
-t	Affichage du contenu du fichier archive
-f	Indication du nom du fichier tar à utiliser

Exemple

```
[root@localhost ~]# tar -cvf F1.tar Rep
Rep/
Rep/Fich
Rep/Fich~
[root@localhost ~]# ls
anaconda-ks.cfg  F1.tar  Fichier1  install.log.syslog  Desktop
Fichier  install.log  Rep
[root@localhost ~]#
```

Test : La commande **test**

Teste les caractéristique d'un fichier et délivre une valeur de retour de 0 si le test est vrai et de 1 sinon. La syntaxe générale est : **test <expression> <fichier>**

Exemple

```
[root@localhost ~]# test -w F1.tar; echo $? //F1.tar est-t-il modifiable?
0
[root@localhost ~]# test -e Fich; echo $? //Fich existe-t-il?
1
[root@localhost ~]#
```

Liens

Lien physique

La commande **ln <nom_fichier> <nom_lien>**

Permet de créer un nouveau lien **physique** pour un fichier déjà existant, autrement dit le fichier devient alors accessible sous deux noms différents.

Exemple

```
[root@localhost ~]# ln Fichier1 FICH
[root@localhost ~]# ls -il
total 136
4620297 -rw----- 1 root root 1098 Feb 16 2008 anaconda-ks.cfg
4620325 drwxr-xr-x 3 root root 4096 Nov 3 22:07 Desktop
4621242 -rw-r--r-- 1 root root 10240 Nov 3 22:00 F1.tar
4621232 -rw-r--r-- 2 root root 93 Nov 3 21:29 FICH
4621241 -rw-r--r-- 1 root root 27 Nov 3 21:37 Fichier
4621232 -rw-r--r-- 2 root root 93 Nov 3 21:29 Fichier1
4620290 -rw-r--r-- 1 root root 39819 Feb 16 2008 install.log
4620291 -rw-r--r-- 1 root root 4593 Feb 16 2008 install.log.syslog
```

```
4621101 drwxr-xr-x 2 root root 4096 Nov 3 21:30 Rep
[root@localhost ~]#
```

Lien symbolique

La commande **ln -s <nom_fichier> <nom_lien>**

Permet la création d'un lien **symbolique**.

Exemple

```
[root@localhost ~]# ln -s Fichier1 Symbol
[root@localhost ~]# ls -il
total 140
4620297 -rw----- 1 root root 1098 Feb 16 2008 anaconda-ks.cfg
4620325 drwxr-xr-x 3 root root 4096 Nov 10 19:21 Desktop
4621242 -rw-r--r-- 1 root root 10240 Nov 3 22:00 F1.tar
4621232 -rw-r--r-- 2 root root 93 Nov 3 21:29 FICH
4621241 -rw-r--r-- 1 root root 27 Nov 3 21:37 Fichier
4621232 -rw-r--r-- 2 root root 93 Nov 3 21:29 Fichier1
4620290 -rw-r--r-- 1 root root 39819 Feb 16 2008 install.log
4620291 -rw-r--r-- 1 root root 4593 Feb 16 2008 install.log.syslog
4621101 drwxr-xr-x 2 root root 4096 Nov 3 21:30 Rep
4621246 lrwxrwxrwx 1 root root 8 Nov 10 19:22 Symbol -> Fichier1
[root@localhost ~]#
```

Commandes de gestion des attributs des fichiers et répertoires

Droits d'accès

La commande **chmod <à qui><opération><permission> <nomdufichier>**

Permet de modifier les droits d'accès d'un fichier ou d'un répertoire.

A qui	u (user), g (group), o (other), a (all)
Opération	+ (ajouter), - (supprimer), = (remplacer)
Permission	r (read), w (write), x (execute)

Exemple

```
[root@localhost ~]# touch F1 F2 F3
[root@localhost ~]# chmod u+x,g+wx F1
[root@localhost ~]# ls -l F1
-rwxrwxr-- 1 root root 0 Nov 10 19:24 F1
[root@localhost ~]# chmod a+w F2
[root@localhost ~]# ls -l F2
-rw-rw-rw- 1 root root 0 Nov 10 19:24 F2
[root@localhost ~]# chmod 761 F3
[root@localhost ~]# ls -l F3
-rwxrw---x 1 root root 0 Nov 10 19:24 F3
[root@localhost ~]# chmod a= F3
[root@localhost ~]# ls -l F3
----- 1 root root 0 Nov 10 19:24 F3
[root@localhost ~]#
```

La commande **umask**

Permet de définir les droits d'accès à retirer lors de la création des fichiers ou répertoires.

Exemple

```
[root@localhost ~]# umask
0022
[root@localhost ~]# touch F4
[root@localhost ~]# umask 002
[root@localhost ~]# touch F5
[root@localhost ~]# ls -l F4 F5
-rw-r--r-- 1 root root      0 Nov 10 19:28 F4
-rw-rw-r-- 1 root root      0 Nov 10 19:29 F5
[root@localhost ~]#
```

Propriétaire

La commande **chown** <nouveau_propriétaire> <fichier>

Permet de changer le propriétaire d'un fichier ou d'un répertoire.

-R	Applique récursivement les modifications au répertoire et à son contenu
-h	Applique le changement sur le lien symbolique pas sur l'objet pointé

Exemple

```
[root@localhost ~]# chown etudiant F4
chown: `etudiant': invalid user
[root@localhost ~]# chown etudiant1 F4
[root@localhost ~]# ls -l F4
-rw-rw-r-- 1 etudiant1 root      0 Nov 10 19:28 F4
[root@localhost ~]#
```

Groupe

La commande **chgrp** <nouveau_propriétaire> <fichier>

Permet de changer le groupe d'un fichier ou d'un répertoire.

Remarque : on peut attribuer en une fois un nouveau propriétaire et un nouveau groupe:

chown <nouveaupropriétaire>.<nouveaugroupe> <fichier>

Exemple

```
[root@localhost ~]# chgrp section F4
[root@localhost ~]# ls -l F4
-rw-rw-r-- 1 etudiant section      0 Nov 10 19:28 F4

[root@localhost ~]# chown etudiant.section F1 F2 F3
[root@localhost ~]# ls -l F1 F2 F3 F4
-rwxrwxr-- 1 etudiant section      0 Nov 10 19:24 F1
-rw-rw-rw- 1 etudiant section      0 Nov 10 19:24 F2
----- 1 etudiant section      0 Nov 10 19:24 F3
-rw-rw-r-- 1 etudiant section      0 Nov 10 19:28 F4
[root@localhost ~]#
```