3 Licence Appliquée
Administration des Réseaux et Services

COURS/TP UNIX

Chapitre 1
Commandes et fichiers

Références:

Le Système Linux. ***M.K.Dalheimer et M.Welsh***
Préparation à la Certification LPIC-1. ***S.Rohaut.***

Ben Youssef Nihel – nihel.benyoussef@gmail.com

Travailler en lignes de commandes

Pour déclarer une variable locale:
[root@test] /# **VARNAME=SOMETHING**
Exemple: root@nihel:~# **MONNOM= NIHEL**
Pour supprimer une variable:
[root@test] /# **unset VARNAME**
Pour déclarer une variable qui doit être vue par tous les shells,
il faut utiliser :**Export.**
[root@test] /# **export VARNAME** Ou
[root@test] /# **declare -x VARNAME=SOMETHING**
Voir le contenu de la variable
[root@test] /# **echo $VARNAME**
Ajouter un contenu a une variable
root@nihel:~# **export MONNOM= $MONNOM-BEN YOUSSEF**
NIHEL-BENYOUSSEF
Verifier l'ajout de la variable pour tous les shells
[root@test] /# **env**

Travailler en lignes de commandes

Bash ,comme shell interactif, utilise les fichiers suivants :

1. /**etc/profile** : initialisation par défaut proposée sur cette machine pour Tous les utilisateurs du systeme.
2. Le premier des fichiers **$HOME/.bash_profile** (contient umask,path...), **$HOME/.bash_login**, **$HOME/.profile,** dans cet ordre et s'ils sont présents.
3. Lorsqu'un shell de login se termine il lit et exécute, s'ils sont présents, les scripts **/etc/.bash_logout** et **$HOME/.bash_logout**.
4. Bash exécutera /etc/bash.bashrc and $HOME/.bashrc (contient Les alias) s'ils sont présents. On notera que, dans la plupart des cas, les scripts .bash_profile appellent explicitement les .bashrc.
5. L'historique des commandes tapées est dans le fichier: .bash_history (commande history)

A travers le fichier /**etc/login.defs,** l'administrateur peut modifier des paramètres Et des variables pour tous les utilisateurs.

Travailler en lignes de commandes

La commande type permet d'afficher si la commande est fournit par un shell
Ou un prorgamme
nihel@nihel:~$ **type export**
export est une primitive du shell
nihel@nihel:~$ **type env**
env est /**usr**/**bin**/**env**
- Pour relancer la derniere commande , il faut taper **!!**
- Pour lancer des commandes l'une à la suite de l'autre
Cmd1; cmd2; cmd2
- Pour lancer deux commandes a la suite de l'autre meme si l'une d'elles Echouent:
cmd1 | |cmd2
- pour lancer deux commandes. La commande2 ne s'execute que si la Commande1 a reussi:
cmd1&&cmd2

Filtres de flux de texte

Cat : Concaténer les fichiers

```
user@ubuntu:~$ cat myfile.txt myfile2.txt
a
b
c
d
e
```

Tac Concaténer les fichiers en sens inverse

```
user@ubuntu:~$tac  myfile.txt myfile2.txt
c
b
a
a
d
```

Filtres de flux de texte

Head et **tail** affichent par défaut les 10 premiers et derniers lignes d'un fichier

[root@test] /# head file # Affiche les 10ere lignes du fichier.

[root@test] /# head -n 2 file # Affiche les 2 permieres lignes du fichier.

[root@test] /# tail -c 10 file # Affiche les 10 dernières octets du fichier.

[root@test] /# tail -f -s 1 /var/log/messages # Affiche les 10 dernières lignes du fichier,s'arrete et verifie tous les seconde des nouvelles données
L'option f permet d'afficher toujours a la suite.

Syntaxes exactes aussi:
[root@test] /# head -n 2 < file
[root@test] /# head -2 file
[root@test] /# cat file|head -2

Filtres de flux de texte

nl affiche les numérotations de la ligne.

```
user@ubuntu:~$ nl myfile.txt
     1 a
     2 b
     3 c
```

Wc Compter les lignes -l , les mots -w et les caracteres -m d'un fichier

```
user@ubuntu:~$ wc myfile.txt
3 3 6 myfile.txt
```

Expand remplacer la tabulation par des espaces, unexpand le contraire

```
user@ubuntu:~$ cat expandfile.txt
        tab
user@ubuntu:~$ cat expandfile.txt | wc c
5
user@ubuntu:~$ expand expandfile.txt | wc c
12
```

Filtres de flux de texte

Cut afficher ou supprimer les sections de chaque ligne
user@ubuntu:~$ cat cutfile.txt
01:Harry:32423:!
02:William:53452:!
03:Camilla:342534:!
04:Elizabeth:235345:!
-b Supprimer l'octet de la position 4 – 7
user@ubuntu:~$ cut b 47 cutfile.txt
Harr
Will
Cami
Eliz
- d delimiteur, f- champs
user@ubuntu:~$ cut f3 d: cutfile.txt
32423
53452
342534
235345

Filtres de flux de texte

Tr convertit les caractères

Remplacer a par 1, b par 2 et c par C :

```
user@ubuntu:~$ cat myfile.txt
a
b
c
user@ubuntu:~$ cat myfile.txt | tr 'abc' '12C'
1
2
C
```

Remplacer les minuscules par des majuscules :

```
user@ubuntu:~$ cat myfile.txt | tr '[az]' '[AZ]'
A
B
C
```

Syntaxes possibles: tr 'az' 'AZ' < myfile.txt ou tr az AZ < myfile.txt

Attention syntaxe FAUSSE: tr 'az' 'AZ' myfile.txt

Filtres de flux de texte

Equivalent à **tr** mais plus evolué :
La fonction de substitution de la Commande **sed**
root@test] /# **sed** 's/toto/TOTO/' fichier
#va changer la première occurrence de la chaîne toto par TOTO (la première chaîne toto rencontrée dans le texte uniquement)

[root@test] /# **sed** 's/toto/TOTO/3' fichier
#va changer la troisième occurrence de la chaîne toto par TOTO (la troisième chaîne toto rencontrée dans le texte uniquement)

[root@test] /# **sed** 's/toto/TOTO/g' fichier
#va changer toutes les occurrences de la chaîne toto par TOTO (toutes les chaînes toto rencontrées sont changées

[root@test] /# **sed** 's/toto/TOTO/w resultat' fichier
#fichier en cas de substitution la ligne en entrée est inscrite dans un fichier résultat

Filtres de flux de texte

sed "s/[Ff]raise/FRAISE/g" fichier
#substitue toutes les chaînes Fraise ou fraise par FRAISE
La fonction y permet de changer un caractère par un autre(comme la commande tr)
sed '**y**/abcd/ABCD/' fichier

Autre fonctions de la commande sed
[root@test] /# sed '20,30**d**' fichier
#suppression des lignes 20 à 30 du fichier fichier.

[root@test] /# sed '/toto/**d**' fichier
#Suppression des lignes contenant la chaîne toto.

Si au contraire on ne veut pas effacer les lignes contenant la chaîne toto (toutes les autres sont #supprimées), on tapera :
[root@test] /# sed '/toto/**!d**' fichier

Filtres de flux de texte

Paste pour coller plusieurs lignes de fichiers

```
user@ubuntu:~$ cat paste1.txt
1
2
3
user@ubuntu:~$ cat paste2.txt
a
b
c
user@ubuntu:~$ paste paste1.txt paste2.txt
1	a
2	b
3	c
```

Join pour joindre plusieurs lignes d'un fichier

```
user@ubuntu:~$ cat join1.txt
1 a
2 b
3 c
user@ubuntu:~$ cat join2.txt
1 d
2 e
4 f
user@ubuntu:~$ join join1.txt
join2.txt
1 a d
2 b e
```

Filtres de flux de texte

Uniq pour supprimer les lignes dupliquées

```
user@ubuntu:~$ cat uniqfile.txt
a
a
b
b
b
c
c
user@ubuntu:~$ uniq uniqfile.txt
a
b
c
user@ubuntu:~$ uniq c uniqfile.txt
      2 a
      3 b
      2 c
```

Split pour diviser les fichiers

```
user@ubuntu:~$ cat myfile.txt
a
b
c
user@ubuntu:~$ split -l 1 myfile.txt
user@ubuntu:~$ cat xaa
a
user@ubuntu:~$ cat xab
b
user@ubuntu:~$ cat xac
C
```

Pour regrouper
Cat xa*>fichregroup.txt

Filtres de flux de texte

Paste pour coller plusieurs lignes de fichiers

```
user@ubuntu:~$ cat paste1.txt
1
2
3
user@ubuntu:~$ cat paste2.txt
a
b
c
user@ubuntu:~$ paste paste1.txt paste2.txt
1	a
2	b
3	c
```

Join pour joindre plusieurs lignes d'un fichier

```
user@ubuntu:~$ cat join1.txt
1 a
2 b
3 c
user@ubuntu:~$ cat join2.txt
1 d
2 e
4 f
user@ubuntu:~$ join join1.txt
join2.txt
1 a d
2 b e
```

Filtres de flux de texte

Fmt pour formater les fichiers
[root@test] /# **fmt** -w 35 file # Display 35-character lines width.

Pr pour formater un fichier pour l'imprimante
[root@test] /# pr -d file # Format file with double-space.

Sort pour trier un fichier
user@ubuntu:~$ cat sortfile.txt
d
e
a
user@ubuntu:~$ sort sortfile.txt
a
d
e
user@ubuntu:~$ sort -r sortfile.txt
e
d
a

Gestion de Base des fichiers

Od Afficher un fichier binaire

[root@test] /# od -cx /bin/ls
0000000 177 E L F 001 001 001 \0 \0 \0 \0 \0 \0 \0 \0 457f 464c 0101 0001
0000 0000 0000 0000 0000020 002 \0 003 \0 001 \0 \0 \0 224 004 \b 4 \0 \0
\0
0002 0003 0001 0000 9420 0804 0034 0000 0000040 Â° Â2 \0 \0 \0 \0 \0 \0 4 \
0 \0 006 \0 (\0 b2b0 0000 0000 0000 0034 0020 0006 0028 0000060 032 \0
031 \
0 006 \0 \0 \0 4 \0 \0 \0 4 200 004 \b 001a 0019 0006 0000 0034 0000 8034
0804_

Options:
C: un caractere en 1 octet
X: 2 octets en hex

Gestion de Base des fichiers

Pwd Afficher le repertoire courant, **cd** accéder à un repertoire

user@ubuntu:~$ pwd
/home/user
user@ubuntu:~$ cd .
user@ubuntu:~$ pwd
/home/user

Aller au repertoire parent

user@ubuntu:~$ cd ..
user@ubuntu:/home$ pwd
/home

Aller au home directory

user@ubuntu:/bin$ cd
user@ubuntu:~$
Ou
user@ubuntu:/bin$ cd ~
user@ubuntu:~$

Aller au repertoir precedent

user@ubuntu:/bin$ cd
user@ubuntu:~$
user@ubuntu:~$ cd
user@ubuntu:/bin$

Gestion de Base des fichiers

Ls Lister le contenu
Afficher les fichiers caches
user@ubuntu:~$ ls a
.
..
.bash_history
.bash_logout
.bash_profile
.bashrc
Desktop
[..]

user@ubuntu:~$ ls -l /bin/ls
-rwxr-xr-x 1 user user 46784 mar 23 2002 /bin/ls

Ne pas descendre dans les sous répertoires

user@ubuntu:~$ ls -ld /bin
drwxr-xr-x 2 user user 2144 nov 5 11:55 /bin

Gestion de Base des fichiers

File permet de determiner le type de contenu d'un fichier

```
user@ubuntu:~$ file /etc
/etc: directory
user@ubuntu:~$ file .bashrc
.bashrc: ASCII English text
```

Wich determine le chemin d'un programme

```
user@ubuntu:~$ which cut
/usr/bin/cut
```

Mkdir cree un repertoire

Avec permission

```
[root@test ] /# mkdir -m 0700 bin
```

Cree un parent s'il n'existe pas

```
[root@test ] /# mkdir -p bin/system/x86
```

Rmdir supprimer un repertoire

```
[root@test ] /# rmdir tmp
```

Gestion de Base des fichiers

Cp copier un fichier dans un autre ou dans un repertoire
-r copies sous repertoires et leurs contenu
-i: interactive: demande l'autorisation avant de copier.
-f : forcer ne pas demander
[root@test] /# cp *.[a-z] /tmp
[root@test] /# cp readme readme.orig
[root@test] /# cp ls /bin
[root@test] /# cp -ri bin/* /bin

Rm supprimer les fichiers ou les repertoires
i interactive: Affiche un prompte avant chaque suppression.
f force: Ne demande pas de confirmation et ne prévient pas en cas d'échec.
r récursive: Efface les sousrépertoires et leur contenu.
[root@test] /# rm *.[a-z]
[root@test] /# rm readme readme.orig
[root@test] /# rm -rfi /bin
[root@test] /# cd; rm -rf * .*

Gestion de Base des fichiers

Mv deplacer ou nommer un fichier ou un repertoire
i: interactive: Affiche un prompte à chaque action.
f: force: Force pour écraser.

[root@test] /# mv *.[a-z] /tmp
[root@test] /# mv readme readme.orig
[root@test] /# mv ls /bin
[root@test] /# mv -fi bin/* /bin

Wildcards
Liste tous les fichiers commencant pas f ou F
root@test] /# ls -l [fF]*
[root@test] /# ls *.c | more
[root@test] /# ls -l [a-s]*

Mecanismes de citations
echo What \'s happenning \?
echo "What's happenning?"

Recherche de fichiers

Pour chercher un fichier dans un sousrépertoire, il faut utiliser find.

find [subtrees] [conditions] [actions]

Parmi les conditions, il peut y avoir:

name [FNG]: Recherche les noms de FNG.
type c: Type de fichier [bcdfl].
size [+]#: A une taille de + en blocks (c:octets,k:kilo)
user [name]: Propriété de l'utilisateur
atime [+]#: Date d'accession aux fichiers: +n appelle les fichiers qui n'ont pas été ouverts les n derniers jours, n appelle les fichiers qui ont été accédés les n derniers jours.
mtime [+]#: Date de modification des fichiers.
perm nnn: Possèdent la permission nnn.
inum N: Fichiers avec un nombre inode N.

Parmi les actions possibles à exécuter pour chaque fichier trouvé:

print: Donne le chemin d'accès.
exec cmd {} \; : Exécute cmd avec le nom du fichier en argument.
ok cmd {} \; : pareil que exec.

Recherche de fichiers

Exemples

```
[root@test ] /# find . -name '*.[ch]' -print
[root@test ] /# find /var /tmp . -size +20 -print
[root@test ] /# find ~ -type c -name '*sys*' -print
[root@test ] /# find /tmp/toto -type f -size +2c -exec rm -i {} \;
[root@test ] /# find / -atime -3 -print
[root@test ] /# find ~jo ~toto -user chloe -exec mv {} /tmp \;
```

locate est un outil d'indexation et de recherche rapide parmis tous les fichiers du système. **slocate** est une version sécurisée de locate, et sur Ubuntu, c'est cette version qui est lancée par défaut, même si vous lancez locate.

Avant de lancer locate, vous devez créer les index avec updatedb :

user@ubuntu:~$ **sudo updatedb**

Le fichier de configuration d'updatedb est /**etc**/**updatedb.conf** .

Pour trouver l'emplacement d'un binaire, de la source d'un fichier ou des pages de manuel, il faut utiliser whereis.

b: Recherche uniquement les binaires.

m: Recherche uniquement les manuels.

s: Recherche uniquement les sources

Recherche de fichiers

Pour trouver un fichier situé à un endroit défini par une variable PATH, il faut
utiliser which.

```
[root@test ] /# which -a ls
/bin/ls
```

L'option a permet de chercher toutes les possibilités du PATH, pas uniquement la première.

Pipes/ Redirections

Pour chaque commande exécutée dans un terminal, il y a:

une valeur d'entrée standard, 0 (le clavier par défaut sur une console).

une valeur de sortie standard,1 (par défaut le terminal).

une valeur de sortie standard pour les erreurs, 2 (par défaut le terminal).

Chaque canal peut aussi être identifié par une adresse:

&0 pour les entrées,

&1 pour les sorties,

&2 pour les erreurs.

Chaque canal [n] peut être redirigé.

[n]< file: La valeur par défaut de n est 0 et elle lit les entrées standards depuis le fichier.

[n]> file: La valeur par défaut est 1 et il envoie les sorties standards au fichier (clobber).

[n]>>file: la valeur par défaut est 1 et il appose les sorties standards au fichier.

<<word: Lit les le texte qui suit jusqu'à ce que word soit atteint. `command` ou $(command) : Remplace le nom de la commande par le résultat.

Pipes/ Redirections

[root@test] /# pwd>file # out=file in=none error=terminal
[root@test] /# cat chap* >book # out=book in=none error=terminal.
[root@test] /# mv /etc/* . 2>error # out=terminal in=none error=error.
[root@test] /# echo end of file >> book # out=book in=none
error=terminal.
[root@test] /# ls > list 2>&1 # ls and errors are redirected to list.
[root@test] /# ls 2>&1 > list # Errors are redirected to standard output
and ls output is redirected to list.
[root@test] /# cat `ls /etc/*.conf` > conffile 2>>/tmp/errors
[root@test] /# cat $(ls /etc/*.conf) > conffile 2>>/tmp/errors
Concatenate all the configuration files from /etc dir
in conffile and append errors in file /tmp/errors.
Utilisez le fichier /dev/null pour la suppression des messages d'erreur.
Essayez les commandes suivantes:
[root@test] /# grep try /etc/*
[root@test] /# grep try /etc/* 2>/dev/null

Redirection: command1 | command2

Il fait une redirection de la sortie standard de command1 vers l'entrée standard de la command2. **Exemples:**

[root@test] /# ls -l /dev | more
[root@test] /# ls -l /etc/*.conf | grep user | grep 500
[root@test] /# ls -l /bin | mail `users`

Pipes/ Redirections

La commande **tee** permet de dupliquer les sorties standards dans un fichier et sur la sortie standard en même temps.

```
[root@test ] /# ls -l /dev | tee file
[root@test ] /# ls -l /etc | tee -a file # Append to the file
```

La commande **xargs** construit une liste d'arguments pour une commande en utilisant les entrées standards. Elle est utilisée comme un pipe.

```
[root@test ] /# ls f* | xargs cat # Print to standard output the content of
all files starting with f
[root@test ] /# find ~ -name 'proj1*' print | xargs cat # Search in the Home
directory for files starting with proj1 and send it to the standard input of
cat.
```

Modèles de Recherche

Un caractère quelconque	.	Ab.a	Abla ou Abca ou ...
Un ensemble de caractères	[ ]	Ab[sd]a	Absa ou Abda seulement
Une gamme de caractères	[-]	Ab[a-z]a	Abaa ou Abba ou ...
N'est pas dans l'ensemble	[^]	Ab[^a-z]a	Ab6a ou AbZa ou ...
0 et plus	*	Ab*a	Aa ou Aba ou Abbbba ou ...
Début de ligne	^	^Aba	La ligne commence par Aba
Fin de ligne	$	Aba$	La ligne termine avec Aba
Caractères littéraux	\	Aba\$	Aba$

Modèles de Recherche

N'importe quelle chaîne de caractères	.*	Ab.*a	Abrahma ou Abaa ou ...
Une chaîne à partir de ...	[ ]*	th[aersti]*	there ou this ou ...
Gammes multiples	[- -]	Ab[0-2a-c]a	Ab0a ou Abca ou ...
Recherche de \	\\	\\[a-zA-Z]*	\Beethoven

m caractères	\{m\}	b[0-9]\{3\}	b911
m ou plus	\{m,\}	b[0-9]\{2,\}	b52
Entre m et n caractères	\{m,n\}	b[0-9]\{2,4\}	b1234
Début de mot	\<	\<wh	where
Fin de mot	\>	[0-9]\>	bin01

Recherche de Motifs

grep [recherhe du texte dans des fichiers
i: Ignore la casse
l: Liste les noms de fichiers seulement si au moins un correspond
c: Affiche uniquement le nombre de lignes qui correspondent
n: Affiche aussi le numéro de la ligne
v: Ne doit pas correspondre
r: Descendre récursivement dans les répertoires
h: Evite d'ajouter des préfixes aux noms de fichiers lorsque plusieurs fichiers sont recherchés

```
[root@test ] /# grep host /etc/*.conf
[root@test ] /# grep -l '\<mai' /usr/include/*.h
[root@test ] /# grep -n toto /etc/group
[root@test ] /# grep -vc root /etc/passwd
[root@test ] /# grep '^user' /etc/passwd
[root@test ] /# grep '[rR].*' /etc/passwd
[root@test ] /# grep '\<[rR].*' /etc/passwd
```

L'éditeur VI

Selectionner la premiere ligne qui commence par motif a l'ouverture du fichier
Vi +/^motif fich
la premiere ligne qui se termine par motif a l'ouverture du fichier
Vi +/motif$ fich
Mode édition:

- i avant le curseur
- I au début de la ligne
- a après le curseur
- A à la fin de la ligne
- o ajout d'une ligne après la ligne courant
- O ajout d'une ligne avant la ligne courante

Enregister et quitter:
Shift ZZ , ESC ZZ, :wq, :x,
Executer une commande
:!

Afficher les numéros de ligne à l'ouverture
vi +"set number" mytestfile.txt
Selectionner la premiere occurence de motifs à l'ouverture du fichier
Vi +/motif fich
Suppression:

- dw jusqu'au mot suivant
- dd la ligne courante
- D jusqu'à la fin de la ligne
- x un caractère

Déplacement du curseur:
l un caractère à droite
h un caractère à gauche
j une ligne en bas
k une ligne en haut
à la fin de la ligne
^ au début de la ligne
w jusqu'au mot suivant
e jusqu'à la fin du mot

Droits d'accès/Privilèges

Extra access rights			user			group			others		
SUID (s)	SGID (s)	Sticky Bit(t)	r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1	4	2	1

Les permissions par défaut sont:

0666 pour la création d'un fichier

0777 pour la création d'un dossier

La plupart des systèmes attribuent ces permissions lors du démarrage grâce à la commande **umask**. En général, la valeur du umask est 022.

Pour modifier les permissions sur un fichier ou un répertoire, il faut utiliser **Chmod**.

Exemple:

[root@test] /# chmod +r readme # Add read permission for everybody

[root@test] /# chmod -r readme # Remove read permission for everybody

[root@test] /# chmod u+x,g=r readme # Add execution for user and set read for group

[root@test] /# chmod u=rwx,go=rx readme # Set read write and execution for user, read and execution for group and others

[root@test] /# chmod -R +x /sbin/* Pour changer les permissions en mode récursif

Droits d'accès/Privilèges

Chattr permet de modifier les attributs de fichiers

A: l'enregistrement atime n'est pas modifié
a: ouvrir seulement en mode append pour l'écriture
D: écrire les modifications de répertoire sur le disque de façon synchrone
d: ne pas sauvegarder lorsque le programme de dump est lancé
i: ce fichier ne peut pas être supprime meme par root
S: écrire les modifications de fichier sur le disque de façon synchrone

Lsattr permet lister les attributs
Exemple

Desktop ne peut pas etre supprimer

user@ubuntu:~$ sudo chattr +i Desktop/

Pour enlever l'attribut (-i)

user@ubuntu:~$ sudo chattr -i Desktop/

Droits d'accès/Privilèges

Pour changer le propriétaire d'un fichier ou d'un répertoire, il faut utiliser **chown.**

[root@test] /# **chown** nouveauowner fich

Pour modifier le groupe d'un fichier ou d'un répertoire, il faut utiliser chgrp.

[root@test] /# **chgrp** nouveaugroup fich

Changer en meme temps de proprietaire et le groupe

[root@test] /# **chown** owner.group fich

Les programmes **gpasswd** et **yast2** permettent d'administrer les Groupes.

Les membres du groupe administrateurs peuvent ajouter ou enlever un membre du groupe.

[root@test] /# **gpasswd** -d toto users
[root@test] /# **gpasswd** -a toto users

Les membres du groupe administrateurs peuvent modifier ou enlever le mot de passe pour le groupe.

[root@test] /# **gpasswd** users
[root@test] /# **gpasswd** -r users

Lien Symbolique

On utilise des liens pour:

Créer un nouveau chemin pour accéder à un fichier.(lien physique)

Créer un alias plus court ou corrigé à un ficher(lien symolique)

•

Pour lier un fichier à un autre, il faut utiliser ln.

ln [options] filename linkname

ln [options] filename linkdirectory

Principales options:

f: force: Ecrase les liens existants

s: Lien symbolique.

Par défaut, les liens sont des liens physiques.

Dans Windows, ils sont appelés « raccourcis » (shortcut).

Exemple:

root@nihel:/home/nihel# ln fo fli (lien physique/ hard link)

root@nihel:/home/nihel# ln **-s** fo fls (lien symbolique)

root@nihel:/home/nihel# ls -li fo fli fls

61935 -rw-r--r-- 2 root root **418** 2010-04-08 11:41 fli

65677 lrwxrwxrwx 1 root root 2 2010-04-08 11:29 fls -> fo

61935 -rw-r--r-- 2 root root **418** 2010-04-08 11:41 fo

Lien Symbolique

Propriétés des liens physiques

Un lien physique peutêtre créé seulement vers un fichier existant.
On ne pourra supprimer un fichier qu'après avoir supprimé tous ses liens physiques.
Le lien et sa cible doivent être sur le même lecteur. Les liens physiques ne peuvent pas s'étendre à travers un système de fichiers ou plusieurs partitions.
Le lien physique référence sa cible grâce à son nombre **inode**.
Même avec les bons droits, il est impossible de créer un lien vers un répertoire.

Propriété des liens symboliques

Un lien symbolique peutêtre créé vers un fichier qui n'existe pas encore et est identifié par la commande ls. Ainsi, il est possible de casser un lien symbolique en supprimant sa cible.
Le lien symbolique référence sa cible avec son nom et son chemin, et non pas avec son nombre inode.
Les liens symboliques peuvent être déployés sur plusieurs systèmes de fichiers ou lecteurs.
La taille d'un lien symbolique dépend de la taille du nom de la cible.
Attention: Il est possible de créer un symlink avec un chemin absolu ou relatif vers la cible.