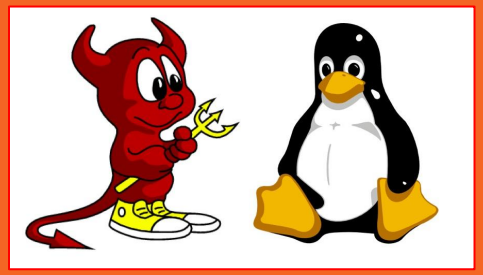


Unix/Linux
ISEN - AP4

2 - Unix concepts

2019 - J. Hochart

Infos

Référence

Supports disponibles en ligne après le cours (j+1):

www.hochart.fr/LIN

Contact

Pour toute question sur le cours ou les exos:

jul@hochart.fr

Licence

Références (2 - Unix concepts)

Livres

- Essential System Administration (3eme Ed)
  - Auteur: Aeleen Frisch
  - Editions: O'Reilly

Web

Fichiers

- **Les fichiers sont centraux** sous Unix (ce n'est pas le cas pour tous les OS)
  - Les **commandes** sont des fichiers **exécutables**
  - Les **privilèges** et **permissions** sont implémentés via **l'accès** aux fichiers
  - L'accès à un device et à un fichier ne varient qu'au plus bas niveau (abstraction)

Fichiers

- Unix à une structure de répertoire hiérarchique (en arbre) → **“le Filesystem”**
- La racine de l’arbre est appelée **root directory** et est désignée par **/**
- Une partie de la sécurité sous unix dépend de
  - La possession (ownership)
  - Le controle d’acces (permissions)

Possession (ownership)

- Sous Unix, un fichier à **2 propriétaires,** qui peuvent être totalement indépendants:
  - Un **utilisateur** propriétaire
  - Un **groupe** propriétaire
- Flexibilité

Possession (ownership)

- ls -l

```
jho@debian:~$ ls -l
total 4532
-rw-r--r-- 1 jho  jho      381 Oct 31 18:56 ages.txt
drwxr-xr-x 2 jho  jho     4096 Nov 15 21:56 asio
drwxr-xr-x 2 jho  jho     4096 Sep 27 21:27 build
drwxr-xr-x 6 jho  jho     4096 Dec  3 15:27 coin
drwxr-xr-x 4 jho  jho    20480 Nov 10 20:52 fsbuild
-rw-r--r-- 1 jho  jho  1500099 Nov 10 17:10 fsbuild-1.tgz
drwxr-xr-x 2 jho  jho     4096 Oct 31 22:33 fstest
-rw-r--r-- 1 jho  jho  1470142 Oct 31 22:34 fstest.tgz
```

Possession (ownership)

- Sur la plupart des unix
  - Le **propriétaire** est le **créateur** du fichier
  - Le groupe est le groupe de l'user qui à créé le fichier (BSD: le groupe est le groupe du répertoire parent)

Possession (ownership)

- Changement de propriétaire
  - **chown jho /tmp/toto**
- Changement de propriétaire récursif
  - **chown -R jho /tmp/works**
- Changement de propriétaire et de groupe
  - **chown jho:wheel /tmp/toto**
- Changement de groupe
  - **chgrp wheel /tmp/toto**

Accès

Access	Meaning for a file	Meaning for a directory
r	View file contents.	Search directory contents (e.g., use `ls`).
w	Alter file contents.	Alter directory contents (e.g., delete or rename files).
x	Run executable file.	Make it your current directory (`cd` to it).

Essential system administration - Oreilly - Table 2.1 - File access types

Accès - Droits requis par action type

	Minimum access needed	
Command	**On file itself**	**On directory file is in**
`cd /home/chavez`	N/A	*x*
`ls /home/chavez/*.c`	(none)	*r*
	r	*x*
`ls -l /home/chavez/*.c`	(none)	*rx*
	r	*x*
`cat myfile`	*r*	*x*
`cat >>myfile`	*w*	*x*
runme (executable)	*x*	*x*
cleanup.sh (script)	*rx*	*x*
`rm myfile`	(none)	*wx*

Essential system administration - Oreilly - Table 2.2 - File protection examples

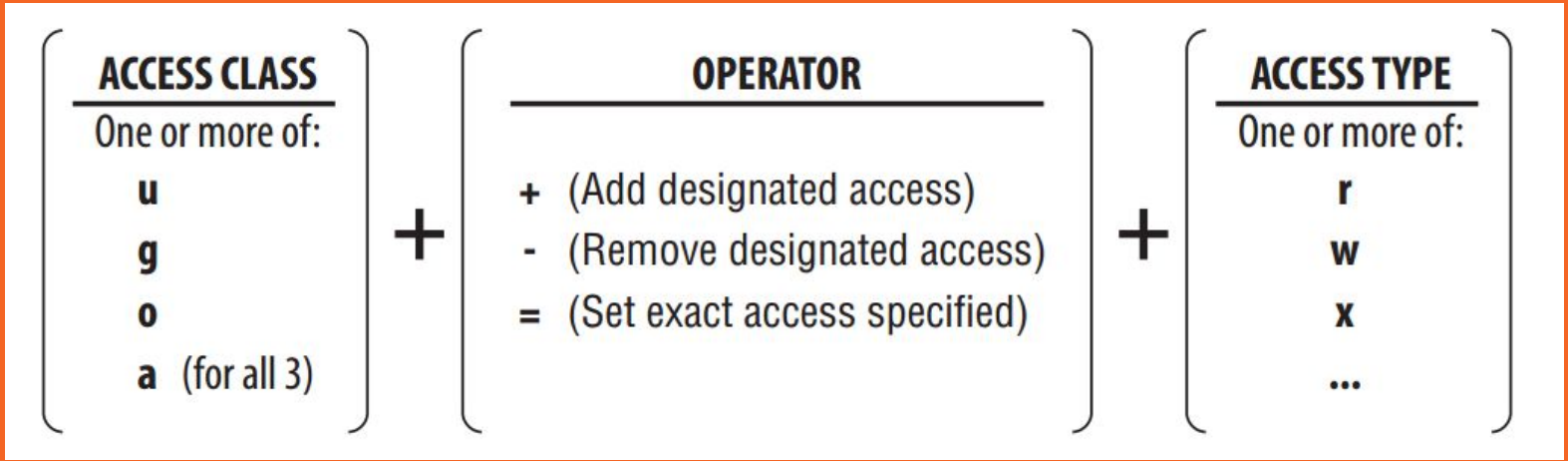
Accès & notation

```
$ ls -l
-rwxr-xr-x  1 root     system      120 Mar 12 09:32  bronze
-r--r--r--  1 chavez   chem         84 Feb 28 21:43  gold
-rw-rw-r--  1 chavez   physics   12842 Oct 24 12:04  platinum
```

		User access			Group access			Other access		
File	type 1	read 2	write 3	exec 4	read 5	write 6	exec 7	read 8	write 9	exec 10
bronze	-	r	w	x	r	-	x	r	-	x
gold	-	r	-	-	r	-	-	r	-	-
platinum	-	r	w	-	r	w	-	r	-	-
/etc/passwd	-	r	w	-	r	-	-	r	-	-

Essential system administration - Oreilly - Table 2.4 -Interpreting mode strings

Accès



Essential system administration - Oreilly - Table 2.1 -Constructing an access string for chmod

- chmod uw+,og+r-w /tmp/toto

Accès

	user			group			other		
Mode	r	w	x	r	-	x	r	-	-
Convert to binary	1	1	1	1	0	1	1	0	0
Convert to octal digit		7			5			4	
Corresponding absolute mode					754				

Essential system administration - Oreilly - Table - Specifying numeric modes

- chmod 754 /tmp/toto

Umask

- La commande umask permet de spécifier le **mode par défaut** pour les fichiers nouvellement créés
  - Le masque est le **complément octal** au mode numérique souhaité
  - umask 023: 777-023 = 754

Modes d’accès spéciaux

- **Sticky bit (t)**
  - **Fichier:** conserver l’exécutable en mémoire après exit
  - **Repertoire:** Restreindre la suppression aux seuls fichiers appartenant à l’user
- **Setuid bit (s)**
  - **Fichier**: Changer l’user ID du process à l’execution
- **Setgid bit (s)**
  - **Fichier**: Changer le group ID du process à l’exec
  - **Repertoire**: Les nouveaux fichiers héritent du group owner

Modes d’accès spéciaux

- **File locking (l):**
  - Fichier: Protection à la lecture et à l’écriture (Solaris, Tru64 et Linux). Affiché en S lors de ls -l

- **Examples**
  - chmod u+t /tmp
  - chmod g+s /tmp/dir

Modes d’accès spéciaux

```
# chmod 4755 uid          Setuid access
# chmod 2755 gid          Setgid access
# chmod 6755 both         Setuid and setgid access: 2 highest bits on
# chmod 1777 sticky       Sticky bit
# chmod 2745 locking      File locking (note that group execute is off)
# ls -ld
-rwsr-sr-x   1 root   chem        0 Mar 30 11:37 both
-rwxr-sr-x   1 root   chem        0 Mar 30 11:37 gid
-rwxr-Sr-x   1 root   chem        0 Mar 30 11:37 locking
drwxrwxrwt   2 root   chem     8192 Mar 30 11:39 sticky
-rwsr-xr-x   1 root   chem        0 Mar 30 11:37 uid
```

Essential system administration - Oreilly - Exemple - Numerical equivalents for special access modes

En cas de probleme suspecté de droits

- Quelque chose (un programme) *marchait avant mais ne marche plus*
- **Essayer avec root.** Si ca marche, c'est probablement un problème de droits sur un des fichiers utilisés (répertoire, fichier de configuration, /dev, /tmp, …)
- **Classique:** edition d'un fichier en tant que root
  - Certains éditeurs font un chown root lorsqu'on sauve
  - Certains éditeurs créent des backups dans le répertoire, et modifient le propriétaire du parent

Type de fichiers

- **Fichiers réguliers** (regular files)
- **Repertoires** (directories)
- **Fichiers spéciaux** (char and block device files)
  - **Character**: Utilisé pour du transfer non bufferisé (terminal)
  - **Block**: Utilisé pour du transfer par morceaux de taille fixe (disques)
- **Lien**
  - Symlink (Symbolic)
  - Hard link: En général inutile, sauf pour optimiser des cas très précis (performance, backup incrémental…)

Symlink (ln -s TARGET LINK)

```
jho@debian:~$ ls -al *.tgz
-rw-r--r-- 1 jho jho 1500099 Nov 10 17:10 fsbuild-1.tgz
-rw-r--r-- 1 jho jho 1470142 Oct 31 22:34 fstest.tgz
-rw-r--r-- 1 jho jho     773 Sep 27 14:31 mystack.tgz
-rw-r--r-- 1 jho jho    3277 Oct 11 10:56 sbuild.tgz
jho@debian:~$ ln -s fsbuild-1.tgz fsbuild-current.tgz
jho@debian:~$ ls -al *.tgz
-rw-r--r-- 1 jho jho 1500099 Nov 10 17:10 fsbuild-1.tgz
lrwxrwxrwx 1 jho jho      13 Jan  3 12:58 fsbuild-current.tgz ->
sbuild-1.tgz
-rw-r--r-- 1 jho jho 1470142 Oct 31 22:34 fstest.tgz
-rw-r--r-- 1 jho jho     773 Sep 27 14:31 mystack.tgz
-rw-r--r-- 1 jho jho    3277 Oct 11 10:56 sbuild.tgz
jho@debian:~$
```

Sockets

- **Unix socket**
- Type de fichier spécial utilisé pour la **communication inter-process**
- Comme un **tuyau** qui permet de communiquer avec un composant du système
- Exemple: sous BSD, /dev/printer et une socket qui sert à communiquer avec le programme lpd (line printer spooling daemon)

Pipes nommés

- **FIFOs**
- Tuyaux ouverts par des applications pour des **communications inter-process**
- “Nommés” car ils ont ouverts en utilisant leur chemin de fichier
- Fonctionnalité apparue sous System V qui à été intégrée à tous les unix
- Principalement dans /dev

Identification des types de fichiers

-	Plain file (hard link)
d	Directory
l	Symbolic link
b	Block special file
c	Character special file
s	Socket
p	Named pipe

```
-rw------- 2
-rw------- 2
drwx------ 2
lrwxrwxrwx 1
brw-r----- 1
crw-r----- 1
srw-rw-rw- 1
prw------- 1
```

Essential system administration - Oreilly - Exemple - Using ls to identify file types

Identification des types de fichiers

- Identification du contenu des fichiers
  - **file**

```
jho@debian:~$ file * | grep -v directory
ages.txt:                              ASCII text
fsbuild-1.tgz:                         gzip compressed data, last modified: Sat Nov 10 16:10:41 2018, from Unix
fsbuild-current.tgz:                   symbolic link to fsbuild-1.tgz
fstest.tgz:                            gzip compressed data, last modified: Wed Oct 31 21:34:29 2018, from Unix
index.html:                            HTML document, ISO-8859 text, with very long lines
index.html.1:                          HTML document, ISO-8859 text, with very long lines
index.html.2:                          HTML document, ISO-8859 text, with very long lines
index.html.3:                          HTML document, ISO-8859 text, with very long lines
Makefile:                              ASCII text
myniceprogram.tar.gz:                  gzip compressed data, last modified: Sun Nov  4 15:54:01 2018, from Unix
mystack.tgz:                           gzip compressed data, last modified: Thu Oct 12 18:15:10 2017, from Unix
sbuild.tgz:                            gzip compressed data, last modified: Thu Oct 11 08:56:12 2018, from Unix
tsmsp1002s_20181012-112109_server.log: ASCII text, with very long lines
jho@debian:~$
```

Processes

- Un process est un exécutable **qui s'exécute dans son propre espace d’adressage**
- **Process != commande** (qui peut faire appel à plusieurs processes)
- 2 grands types de processes
  - Processes **interactifs**: entrée / sortie sur le terminal
  - **Daemons**: tâches de fond souvent démarrées avec le système, qui attendent des évènements

Job control

- Permet de contrôler les processes interactifs depuis le shell

&	Run command in background. `$ long_cmd &`

^Z	Stop foreground process.
	`$ long_cmd`
	`^Z  Stopped`
	`$`

jobs	List background processes.
	`$ jobs`
	`[1] - Stopped emacs`
	`[2] - big_job &`
	`[3] + Stopped long_cmd`

%*n*	Refers to background job number *n*. `$ kill %2`
fg	Bring background process to foreground. `$ fg %1`
%?*str*	Refers to the background job command containing the specified characters. `$ fg %?em`

bg	Restart stopped background process.
	`$ long_cmd`
	`^Z Stopped`
	`$ bg`
	`[3] long_cmd &`

~^Z	Suspend `rlogin` session.
	`bridget-27 $ ~^Z`
	`Stopped`
	`henry-85 $`

Daemons classiques

- **Init**: Premier process créé
- **Syslogd**: Le daemon du service de journalisation syslog
- **Crond**: Le daemon de l'ordonnanceur cron
- **Httpd**: Daemon http (serveur www)
- **Named**: Daemon d'un serveur de noms (DNS)
- ...

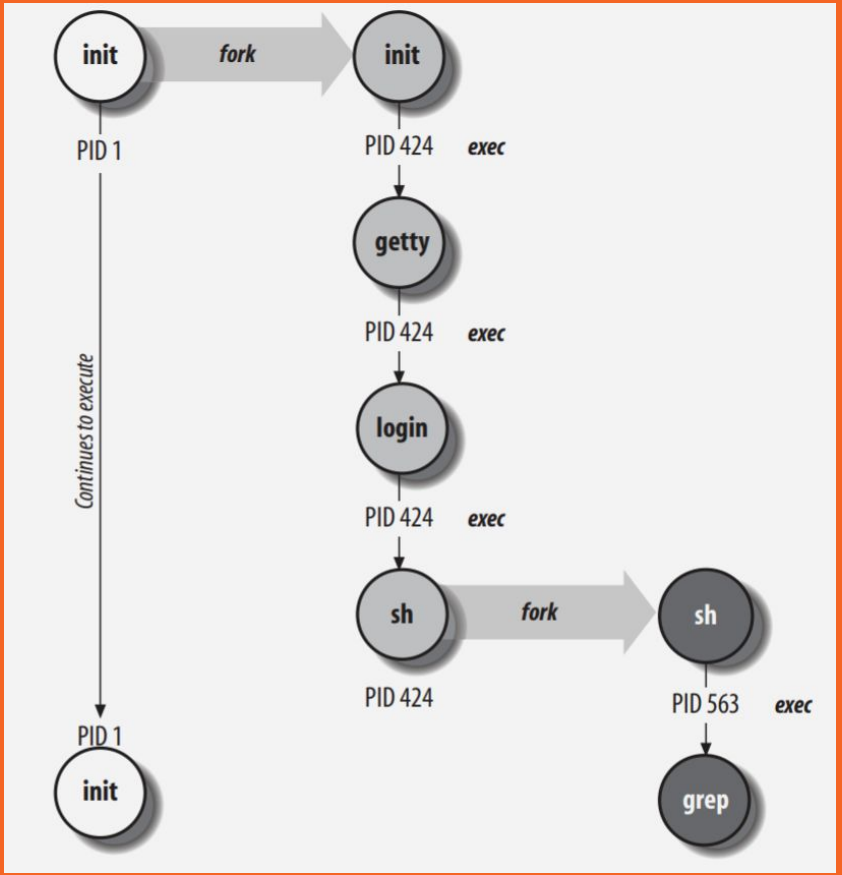
Attributs d’un process

- **PID**: ID du process
- **PPID**: ID du process parent
- **Nice number**: Priorité d’execution
- **TTY**: Terminal associé au process
- **RUID**: (Real): Id de l’user qui à lancé le process
- **EUID**: (Effective): Id après setuid ou privesc
- **RGID, EGID**

Cycle de vie d’un process

- **Tous les processes sur un host linux en fonctionnement proviennent d’un enchainement des 2 appels fork() et exec()**
  - **Fork**: Création d’une copie exacte d’un process, mais **avec un nouveau PID**
  - **Exec**: Écrasement de l’espace mémoire par un nouveau programme et exécution

Cycle de vie d'un process



Essential system administration - Oreilly - Exemple - Unix process creation - fork&exec

Commades VS fichiers

- Certaines commandes sont **builtin** dans le shell et sont interprétées à la volée: **cd, echo, …**
- Les autres sont des fichiers **exécutables** situés dans le **search** path: la variable $**PATH**
- Sur un unix, la plupart des exécutables sont dans
  - **/bin, /sbin**
  - **/usr/bin, /usr/sbin**
  - Certains binaires sont dans des répertoires différents en fonction de l’unix ou de la distribution

Devices

- Point fort d’unix: **abstraction sur les devices**
  - Fichiers spéciaux dans /dev
  - Peu importe ce qu’il y a physiquement derrière pour l’utilisateur

Devices

Device/use	Special file forms	Example
Floppy disk	/dev/[r]fd*n** /dev/floppy	*/dev/fd0*
Tape devices[a]	/dev/rmt*n*	*/dev/rmt1*
	/dev/rmt/*n*	*/dev/rmt/0*
nonrewinding	/dev/nrmt*n*	*/dev/nrmt0*
SCSI	/dev/rst*n*	*/dev/rst0*
default tape drive	/dev/tape	
CD-ROM devices	/dev/cd*n* /dev/cdrom	*/dev/cd0*
Serial lines	/dev/tty*n* /dev/term/*n*	*/dev/tty1* */dev/tty01* */dev/term/01*
Slave virtual terminal (windows, network sessions, etc.)	/dev/tty[p-s]*n* /dev/pts/*n*	*/dev/ttyp1* */dev/pts/2*
Master/control virtual terminal devices	/dev/pty[p-s]*n*	*/dev/ptyp3*
Console device	/dev/console	
some System V	/dev/syscon	
AIX	/dev/lft0	

Essential system administration - Oreilly - Common unix special file names

Devices

Device/use	Special file forms
Process controlling TTY (used to ensure I/O comes from/goes to terminal, regardless of any I/O redirection)	/dev/tty
Memory maps:	
physical	/dev/mem
kernel virtual	/dev/kmem
Mouse interface	/dev/mouse
Null devices: all output is discarded; reads return nothing (0 characters, 0 bytes) or a zero-filled buffer, respectively.	/dev/null /dev/zero

Essential system administration - Oreilly - Common unix special file names

Arborescence classique (linux)

- “**bin**”: Endroit traditionnel ou on trouve les exécutables (**binaires**). Souvent des symlinks individuels vers des fichiers dans /usr/bin
  - **/bin:** Binaires utiles pour booter en single user (réparation)
  - **/sbin:** Binaires pour booter en multi users
  - **/usr/bin:** Binaires non système lancés par les utilisateurs
  - **/usr/local/bin, ...:** binaires locaux

Arborescence classique (linux)

- **/etc:** Fichiers de conf système
- **/home:** L'endroit par défaut pour les homedir des users
- **/lib:** Les bibliothèques (**libs**) partagées **nécessaires pour booter** (avant que /usr soit monté)
- /lost+found: Fichiers perdus, **produits par fsck** au démarrage après un incident. Structures de données sur le disque, mais qui ne sont listées dans aucun répertoire

Arborescence classique (linux)

- **/mnt**: Point de montage temporaire
- **/opt:** La ou les logiciels optionnels sont installés
- **/proc:** Manipulation des processes. Contient aussi des fichiers d'information sur le système: Interruptions, IO, CPU info, …
- Fichiers de boot, incluant le kernel
  - **/kernel: Solaris**
  - **/stand: FreeBSD**
  - **/boot: Linux**

Arborescence classique (linux)

- **/tmp:** Répertoire temporaire. Les scripts de démarrage vident en général ce répertoire
- **/usr:** Répertoire traditionnel pour les programmes compilés en local
  - **/usr/include:** Les headers C pour la compilation
  - **/usr/lib:** Les libs qui vont avec les headers
- **/var:** Données variables (spooling, logs, …)

Arborescence classique (linux)

- **/var/log:** Les journaux maintenus par plusieurs sources
- **/var/mail:** Les mailboxes locales
- **/var/run:** Les PID de plusieurs daemons système
- **/var/spool:** Contient les différents spoolers (print, mail, cron)