

Il filesystem unix



Partizionamento

Strumenti da utilizzare:

- ***fdisk*** : per veri duri (sconsigliato)
- ***cdisk*** : versione di fdisk con un minimo di grafica (consigliato se non è richiesto il ridimensionamento)
- ***parted*** : molto potente, permette anche di ridimensionare partizioni, ma poco intuitivo
- ***qtparted*** : interfaccia grafica per parted, unisce la potenza di parted alla facilità di un'interfaccia grafica (equivalente libero di PowerQuest Partition Magic)



Quale filesystem scegliere?

- **Ext3** : evoluto da Ext2, ha introdotto il “journaling”. Il mantenimento delle strutture di ext2 garantisce piena compatibilità.
- **Reiserfs** : Anch'esso journaled, ma la sua struttura interna è radicalmente differente. In particolare, fa uso di schemi ad albero binario ispirati dal software usato per gestire i database.
- **Jfs** : progettato e usato da IBM. Nato proprietario, è diventato OpenSource di recente. Struttura interna simile a quella di ReiserFS.
- **Xfs** : journaled, con prestazioni paragonabili a reiserfs, è però meno usato.



[1] **Struttura e cartelle**

<i>/bin</i>	Binari di sistema (usabili da tutti)
<i>/boot</i>	Kernel e file relativi al boot-loader
<i>/dev</i>	File dei device
<i>/etc</i>	Configurazioni di sistema
<i>/lib</i>	Librerie condivise e moduli del kernel
<i>/opt</i>	Software opzionale (openoffice, kde..)
<i>/proc</i>	Contiene informazioni utili al kernel
<i>/sbin</i>	Binari di sistema (usabili solo da root)
<i>/tmp</i>	File temporanei
<i>/usr</i>	Comandi e strumenti dell'utente
<i>/var</i>	File variabili (log, database, html..)



[2] **Struttura e cartelle**

<i>/usr/bin</i>	Comandi dell'utente
<i>/usr/include</i>	Directory per i file standard da includere
<i>/usr/lib</i>	Librerie per programmazione
<i>/usr/local</i>	Come /usr, ma annidata
<i>/usr/sbin</i>	Binari non essenziali usabili solo da root
<i>/usr/share</i>	Dati indipendenti dall'architettura
<i>/usr/src</i>	Codice sorgente (opzionale)
<i>/usr/X11R6</i>	Binari di Xorg (o XFree86)



[3] **Struttura e cartelle**

<i>/var/cache</i>	Cache delle applicazioni
<i>/var/lib</i>	Informazioni variabili sullo stato del sistema
<i>/var/lock</i>	File di lock
<i>/var/log</i>	File di log
<i>/var/run</i>	Variabili run-time
<i>/var/htdocs</i>	Documenti html di apache (opzionale)



[4] Struttura e cartelle

<i>/home</i>	Contiene le home di tutti gli utenti
<i>/root</i>	Home dell'utente root
<i>/home/utente</i>	Home di un generico utente "utente"

Nella home di un utente si trovano generalmente tutti i file di impostazione relativi all'uso che fa delle varie applicazioni, la sua posta, i suoi documenti.

NB:

all'avvio del desktop environment (KDE, Gnome, etc) vengono eseguiti tutti gli script contenuti in */home/utente/Desktop/Autostart*



E adesso?

“Montare” una partizione

Trasparenza del filesystem: tutto è organizzato all'interno di una stessa gerarchia: niente più C: e D: .

mount

mount è il tool che ci permette di caricare i nostri dispositivi (o partizioni) nel mount-point desiderato (una cartella a nostra scelta).

```
mount [-t fstype] something somewhere
```



Montare una partizione

```
$ mount -t vfat /dev/sda1 /mnt/usb/dati  
$  
$ cd /mnt/usb/dati  
$ ls  
Star Wars Lego - Revenge of the Brick [dummy].avi
```

Un po' complicato?

Ci si può semplificare la vita editando il file `/etc/fstab`, che dirà al sistema quali sono i device da montare e dove montarli.



/etc/fstab

#device	mountpoint	fs	opzioni	dump	check
/dev/hda1	/boot	ext2	noauto,noatime	1	1
/dev/hda6	/	reiserfs	noatime	0	0
/dev/hda5	none	swap	sw	0	0
/dev/hda7	/home	reiserfs	defaults	0	0
/dev/cdrom	/mnt/cdrom	iso9660	noauto,ro,users	0	0
none	/proc	proc	defaults	0	0
/dev/sda1	/mnt/usb/dati	auto	rw,noauto,users,umask=0	0	0
zuurg:/mnt/fat	/mnt/remoto	nfs	rw,users,noauto	0	0
/dev/hda3	/mnt/windows	auto	rw,noauto,users,umask=0	0	0

adesso per montare la chiavetta è sufficiente

```
$ mount /dev/sda1
```



[1] Gestione dei diritti

3 tipi di diritti:

- lettura (r),
- scrittura (w)
- esecuzione (x)

per 3 tipi di utenza:

- proprietario
- appartenenti dello stesso gruppo
- tutti gli altri

```
$ ls -l filesystem.sxi  
-rw-r--r-- 1 mel users 33357 May 20 19:32 filesystem.sxi
```



[2] Gestione dei diritti

chown

con chown si può cambiare il proprietario di un file e il gruppo a cui il file appartiene

```
chown [OPTION]... OWNER[:[GROUP]] FILE...
```

chmod

con chmod si possono cambiare i diritti per l'accesso ad un file

```
chmod [OPTION]... MODE[,MODE]... FILE...
```

```
chmod [OPTION]... OCTAL-MODE FILE...
```



Riferimenti

<http://openskills.info/topic.php?ID=13>

<http://kerneltrap.org/node/2761>

http://www.slacky.it/tutorial/slackware_for_dummies/slackware_for_dummies.txt

