

FILE SYSTEM

Cosa è:

Un **file system** è un insieme di strutture dati e di regole che permettono di organizzare, archiviare e gestire i file su un dispositivo di memorizzazione, come un hard disk, un SSD, o una chiavetta USB. Il file system definisce la modalità di memorizzazione dei dati, l'organizzazione dei file in cartelle, e le operazioni che è possibile eseguire sui file (lettura, scrittura, spostamento, cancellazione)

Gli elementi del file system:

- **Inode:** Struttura dati che contiene metadati del file, come permessi, proprietà, e posizione dei blocchi.
- **Block Data:** I dati reali del file, il contenuto vero e proprio degli stessi.
- **Superblock:** Contiene informazioni critiche sul file system stesso (Tipo di file system, dimensione dei blocchi e delle partizioni, numero di inode disponibili e utilizzati, Indirizzo del blocco di root, informazioni sullo spazio libero e sull'integrità del file system.
- **Journal:** Log delle modifiche al file system.

Gli inode

Cos'è: L'**inode** è una struttura dati fondamentale in molti file system Unix-like (come Linux). Ogni file o directory è associato a un inode, che contiene metadati relativi al file.

Contenuto dell' inode:

- **Tipo di file:** Se è un file normale, una directory, un link simbolico, ecc.
- **Permessi:** I permessi di accesso (lettura, scrittura, esecuzione) per il proprietario, il gruppo e gli altri utenti.
- **Proprietario e gruppo:** Il proprietario del file e il gruppo di appartenenza.
- **Dimensione del file:** La dimensione totale del file in byte.
- **Timestamps:** Data e ora della creazione, dell'ultima modifica e dell'ultimo accesso del file.
- **Numero di link:** Il numero di hard link che puntano a quel file.
- **Posizione dei dati:** L'inode contiene puntatori (indirizzi) ai blocchi di dati che effettivamente contengono il contenuto del file.

File system più diffusi:

FAT32 (File Allocation Table 32):

- **Utilizzo:** Spesso utilizzato per dispositivi di archiviazione portatili, come chiavette USB e schede SD.
- **Limiti:** File di dimensione massima di 4 GB, supporta solo partizioni fino a 8 TB.
- **Vantaggi:** Compatibile con molti dispositivi, sistemi operativi e piattaforme (Windows, macOS, Linux, ecc.).

NTFS (New Technology File System):

- **Utilizzo:** Il file system principale di Windows.
- **Caratteristiche:** Supporta file di 16TB, permessi di sicurezza avanzati, cifratura, e journaling (che aiuta a prevenire la corruzione dei dati in caso di crash).
- **Vantaggi:** Supporta funzionalità avanzate come la compressione dei file e la gestione degli accessi a livello di file.

exFAT (Extended File Allocation Table):

- **Utilizzo:** Utilizzato per dispositivi di memorizzazione esterni, come dischi esterni o schede SD ad alta capacità.
- **Caratteristiche:** Maggiore flessibilità rispetto a FAT32, supporta file fino a 128 PB.
- **Vantaggi:** Compatibile sia con Windows che con macOS, ma con una gestione dei permessi più semplice rispetto a NTFS.

HFS+ (Mac OS Extended):

- **Utilizzo:** File system tradizionale di macOS.
- **Caratteristiche:** Include supporto per metadati, file di grandi dimensioni 8 EB, e supporto per le directory a livello di journaling.
- **Vantaggi:** Ottimizzato per l'uso su dispositivi Apple.

EXT4 (Fourth Extended File System):

- **Utilizzo:** Il file system più comune su Linux.
- **Caratteristiche:** Supporta file di grandi dimensioni, gestione avanzata delle directory, e journaling.
- **Vantaggi:** Affidabile, con buone performance, e una lunga storia di utilizzo su Linux.

XFS è un file system avanzato utilizzato principalmente su sistemi Linux, noto per le sue capacità di gestire grandi volumi di dati.

Ecco le principali caratteristiche e vantaggi di XFS:

1. Alta scalabilità

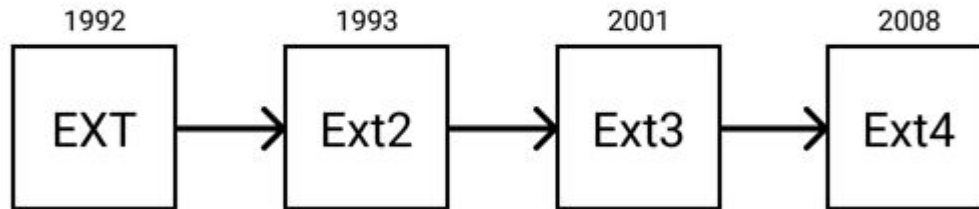
- XFS è progettato per supportare file system di grandi dimensioni e grandi quantità di dati. Può gestire volumi di storage fino a **8 exabyte**, rendendolo ideale per ambienti con necessità di archiviazione di grandi dimensioni, come data center o server aziendali.
- È in grado di gestire singoli file di dimensioni molto grandi (fino a 9 petabyte)

2. Gestione avanzata dei metadati

- I metadati in XFS sono gestiti in modo molto efficiente grazie alla struttura B-tree, che consente un accesso rapido alle informazioni relative ai file, alle directory e ai permessi.

File system ext

Chronology of the development of the EXT halyard system



Ext4 caratteristiche principali:

Volumi grandi: file grandi fino a 16 TiB, ed i file system potranno essere grandi fino a 1 EiB (exbibyte 2^{60}).

Extent: Riduce la quantità di metadati di un file. Invece di usare mapping dei data block indiretto (quindi tener traccia di ogni blocco), con il metodo extent i blocchi del file sono contigui, così che l'unica informazione necessaria è sapere gli N blocchi dove i dati del file si trovano. Con file di grandi dimensioni le performance aumentano drasticamente.

Comando mkfs:

Comando usato per formattare una partizione di un dispositivo di archiviazione.

1. -t (tipo di filesystem)

Specifica il tipo di filesystem da creare. Ad esempio:

- `mkfs -t ext4 /dev/sdX1` crea un filesystem **ext4** sulla partizione `/dev/sdX1`.
- `mkfs -t ntfs /dev/sdX1` crea un filesystem **NTFS** sulla partizione `/dev/sdX1`.

2. -V (verbose)

Abilita l'output verboso, che mostra maggiori dettagli su ciò che sta facendo `mkfs`.

- Esempio: `mkfs -V ext4 /dev/sdX1`

3. -L (label)

Consente di assegnare una **etichetta** (label) al filesystem, che può essere utilizzata per identificarlo.

- Esempio: `mkfs -t ext4 -L Mariodisk /dev/sdX1` assegna l'etichetta "Mariodisk" al filesystem ext4.

4. -c (check for bad blocks)

Questa opzione esegue un controllo dei blocchi danneggiati durante la creazione del filesystem. È utile se si sospetta che il disco possa contenere settori danneggiati. Il comando `mkfs -c` esegue una scansione completa del dispositivo di memorizzazione o della partizione, leggendone ogni settore.

- Esempio: `mkfs -t ext4 -c /dev/sdX1`

5. -b (block size)

Specifica la dimensione del blocco in byte. Questa opzione è utile quando si vuole ottimizzare il filesystem per specifiche esigenze di performance o uso dello spazio.

- Esempio: `mkfs -t ext4 -b 4096 /dev/sdX1` crea un filesystem ext4 con blocchi di 4096 byte.

6. -n (no write)

Esegui solo un "dry run" (simulazione) senza effettivamente scrivere il filesystem. Può essere utile per verificare le configurazioni senza rischiare di perdere dati.

- Esempio: `mkfs -n -t ext4 /dev/sdX1`

Come creare un filesystem:

```
eugenio@ubuntu:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0    4K  1 loop /snap/bare/5
loop1        7:1      0   63,9M  1 loop /snap/core20/2318
loop2        7:2      0   63,7M  1 loop /snap/core20/2434
loop3        7:3      0   73,9M  1 loop /snap/core22/1663
loop4        7:4      0   73,9M  1 loop /snap/core22/1722
loop5        7:5      0  237,2M  1 loop /snap/firefox/2987
loop6        7:6      0  240,3M  1 loop /snap/firefox/3358
loop7        7:7      0  349,7M  1 loop /snap/gnome-3-38-2004/143
loop8        7:8      0   497M  1 loop /snap/gnome-42-2204/141
loop9        7:9      0  505,1M  1 loop /snap/gnome-42-2204/176
loop10       7:10     0   91,7M  1 loop /snap/gtk-common-themes/1535
loop11       7:11     0   12,9M  1 loop /snap/snap-store/1113
loop12       7:12     0   12,2M  1 loop /snap/snap-store/1216
loop13       7:13     0   44,3M  1 loop /snap/snapd/23258
loop14       7:14     0   44,4M  1 loop /snap/snapd/23545
loop15       7:15     0    476K  1 loop /snap/snapd-desktop-integration/157
loop16       7:16     0    568K  1 loop /snap/snapd-desktop-integration/253
sda          8:0      0    25G   0 disk
├─sda1       8:1      0     1M   0 part
├─sda2       8:2      0   513M   0 part /boot/efi
└─sda3       8:3      0   24,5G   0 part /var/snap/firefox/common/host-hunspell
```

Non ci sono dischi disponibili alla partizione e formattazione

```

loop1 7:1 0 4K 1 loop /snap/bare/5
loop2 7:2 0 73,9M 1 loop /snap/core22/1663
loop3 7:3 0 63,7M 1 loop /snap/core20/2434
loop4 7:4 0 237,2M 1 loop /snap/firefox/2987
loop5 7:5 0 73,9M 1 loop /snap/core22/1722
loop6 7:6 0 240,3M 1 loop /snap/firefox/3358
loop7 7:7 0 349,7M 1 loop /snap/gnome-3-38-2004/143
loop8 7:8 0 505,1M 1 loop /snap/gnome-42-2204/176
loop9 7:9 0 497M 1 loop /snap/gnome-42-2204/141
loop10 7:10 0 91,7M 1 loop /snap/gtk-common-themes/1535
loop11 7:11 0 12,9M 1 loop /snap/snap-store/1113
loop12 7:12 0 12,2M 1 loop /snap/snap-store/1216
loop13 7:13 0 476K 1 loop /snap/snapd-desktop-integration/157
loop14 7:14 0 44,3M 1 loop /snap/snapd/23258
loop15 7:15 0 568K 1 loop /snap/snapd-desktop-integration/253
loop16 7:16 0 44,4M 1 loop /snap/snapd/23545
sda 8:0 0 25G 0 disk
├─sda1 8:1 0 1M 0 part
├─sda2 8:2 0 513M 0 part /boot/efi
└─sda3 8:3 0 24,5G 0 part /var/snap/firefox/common/host-hunspell
/
sdb 8:16 0 10G 0 disk
sr0 11:0 1 51M 0 rom /media/eugenio/VBox_GAs_7.0.12
eugenio@ubuntu:~$

```



```

eugenio@ubuntu:~$ sudo fdisk /dev/sdb
[sudo] password for eugenio:

```

```

Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

```

```

Device does not contain a recognised partition table.
Created a new DOS disklabel with disk identifier 0xd845e15d.

```

```

Command (m for help): n

```

Rifacendo lsblk dopo:

```
sdb      8:16    0    10G    0 disk /
└─sdb1    8:17    0    10G    0 part
sr0      11:0    1    51M    0 rom  /media/eugenio/VBox_GAs_7.0.12
eugenio@ubuntu:~$
```

Però...

```
eugenio@ubuntu:~$ lsblk -f
NAME FSTYPE FSVER LABEL UUID                                 FSAVAIL FSUSE% MOUNTPOINTS
loop0
  squash 4.0                                0    100% /snap/core20/2318
loop1
  squash 4.0                                0    100% /snap/bare/5
loop2
  squash 4.0                                0    100% /snap/core22/1663
loop3
  squash 4.0                                0    100% /snap/core20/2434
loop4
  squash 4.0                                0    100% /snap/firefox/2987
loop5
  squash 4.0                                0    100% /snap/core22/1722
loop6
  squash 4.0                                0    100% /snap/firefox/3358
loop7
  squash 4.0                                0    100% /snap/gnome-3-38-2004/143
loop8
  squash 4.0                                0    100% /snap/gnome-42-2204/176
loop9
  squash 4.0                                0    100% /snap/gnome-42-2204/141
loop10
  squash 4.0                                0    100% /snap/gtk-common-themes/1535
loop11
  squash 4.0                                0    100% /snap/snap-store/1113
loop12
  squash 4.0                                0    100% /snap/snap-store/1216
loop13
  squash 4.0                                0    100% /snap/snapd-desktop-integration/157
loop14
  squash 4.0                                0    100% /snap/snapd/23258
loop15
  squash 4.0                                0    100% /snap/snapd-desktop-integration/253
loop16
  squash 4.0                                0    100% /snap/snapd/23545
sda
├─ sda1
├─ sda2
│   vfat  FAT32      AD4C-1C0C          505,9M    1% /boot/efi
└─ sda3
    ext4   1.0        58867e85-5179-492b-8fed-2bcfddd084d8  8G       61% /var/snap/firefox/common/host-hunspell/
sdb
└─ sdb1
sr0 iso966 Jolie VBox_GAs_7.0.12
    2023-10-12-17-41-58-38
    0    100% /media/eugenio/VBox_GAs_7.0.12
```

Flag -f per indicare il filesystem


```
eugenio@ubuntu:~$ sudo mkfs.ext4 -L Mariodisk /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2621184 4k blocks and 655360 inodes
Filesystem UUID: 8417096c-bca2-4043-8613-9f6100e70049
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

mkfs.ext4 è un alias per: mkfs -t ext4

Adesso la partizione è formattata ed è montata su /mnt

```
eugenio@ubuntu:~$ lsblk -f
```

NAME	FSTYPE	FSVER	LABEL	UUID	FSAVAIL	FSUSE%	MOUNTPPOINTS
loop0	squashfs	4.0			0	100%	/snap/core20/2318
loop1	squashfs	4.0			0	100%	/snap/bare/5
loop2	squashfs	4.0			0	100%	/snap/core22/1663
loop3	squashfs	4.0			0	100%	/snap/core20/2434
loop4	squashfs	4.0			0	100%	/snap/firefox/2987
loop5	squashfs	4.0			0	100%	/snap/core22/1722
loop6	squashfs	4.0			0	100%	/snap/firefox/3358
loop7	squashfs	4.0			0	100%	/snap/gnome-3-38-2004/143
loop8	squashfs	4.0			0	100%	/snap/gnome-42-2204/176
loop9	squashfs	4.0			0	100%	/snap/gnome-42-2204/141
loop10	squashfs	4.0			0	100%	/snap/gtk-common-themes/1535
loop11	squashfs	4.0			0	100%	/snap/snap-store/1113
loop12	squashfs	4.0			0	100%	/snap/snap-store/1216
loop13	squashfs	4.0			0	100%	/snap/snapd-desktop-integration/157
loop14	squashfs	4.0			0	100%	/snap/snapd/23258
loop15	squashfs	4.0			0	100%	/snap/snapd-desktop-integration/253
loop16	squashfs	4.0			0	100%	/snap/snapd/23545
sda							
└sda1							
└sda2	vfat	FAT32		AD4C-1C0C	505,9M	1%	/boot/efi
└sda3	ext4	1.0		58867e85-5179-492b-8fed-2bcfddd084d8	8G	61%	/var/snap/firefox/common/host-hunspell
sdb							
└sdb1	ext4	1.0	Mariodisk	8417096c-bca2-4043-8613-9f6100e70049	9,2G	0%	/mnt

Prova creazione di un file e verifica:

```
eugenio@ubuntu:/mnt$ sudo touch marioo.txt
eugenio@ubuntu:/mnt$ df /mnt/marioo.txt
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sdb1	10217748	24	9677104	1%	/mnt

Precisazione, il comando mke2fs:

mke2fs è stato progettato esclusivamente per i filesystem **ext**:

Un esempio di creazione di filesystem con mke2fs : `mke2fs -t ext4 /dev/sdb1`