



# Introduzione a Linux

## Lezione 3 Il file system

Ruggero Donida Labati

### Laboratorio di Sistemi Operativi

Università degli Studi di Milano  
Dipartimento di Informatica  
A.A. 2024/2025

*Materiale prodotto originariamente da Ruggero Donida Labati e Angelo Genovese*

RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

1

## Panoramica della lezione

- Verrà introdotta l'installazione per uso stabile
- Sarà introdotto il file system e le sue caratteristiche
  - Sicurezza del file system
- Saranno introdotti gli aspetti e i metodi per la gestione degli utenti



RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

2

## Sommario (1/2)

1. Installazione per uso stabile
  - Partizionamento del disco
  - Bootloader
2. Il file system
  - Aspetti generali
  - Il file system Ext
3. File e directory
  - Tipi di file
  - Directory principali
  - Creazione
  - Manipolazione



RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

3

## Sommario (2/2)

4. Sicurezza del file system
  - Privilegi associati ai file
  - Privilegi associati alle directory
  - Modifica dei privilegi
5. Gestione gruppi e utenti
  - Aggiungere un utente
  - Aggiungere un utente a un gruppo
  - Gestione gruppi e utenti tramite GUI
6. Esercizi



RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

4

## 1. Installazione per uso stabile

1. Motivazioni
2. Metodi
3. Partizionamento del disco
4. Bootloader



RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

5

1. INSTALLAZIONE PER USO STABILE – MOTIVAZIONI

### Motivazioni per un'installazione per uso stabile

- Un'installazione per uso stabile di linux non può prevedere emulazioni
  - Installazione del sistema operativo in partizioni apposite
    - File system non virtualizzato
  - Avvio di Linux come sistema operativo principale
- Attualmente, è relativamente semplice installare più sistemi operativi sulla stessa macchina
  - Es. Windows XP, Windows 8, Linux

RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

6

## Metodi

1. Masterizzare la ISO su un DVD
  - Procedimento lento
  - Noioso
  - Si spreca un DVD
2. Usare una chiavetta USB
  - Software trasferiscono la ISO su chiavetta e la rendono bootabile
    - Es. Unetbootin



## Partizionamento del disco (1/5)

- Una partizione è una suddivisione logica dell'hard disk
- Ogni partizione è vista come un'unità separata
  - Gestita in modo indipendente
- Un'installazione definitiva di Linux richiede 2 partizioni
  - *root*
  - *swap*



## Partizionamento del disco (2/5)

- La partizione di root contiene il sistema operativo e tutti i dati
- La partizione di swap contiene i dati immagazzinati nella memoria virtuale
  - Non è sempre indispensabile
- Altre partizioni possono essere create alla bisogna
  - Partizione con directory `/home`

## Partizionamento del disco (3/5)

- Le partizioni sono formattate per essere utilizzate dal sistema operativo
  - La formattazione crea sul disco le strutture dati per la memorizzazione dei dati
- Il file system descrive il tipo di struttura utilizzato
- Il file system più diffuso in ambito Linux è il Ext
  - Ultima versione Ext4
- La partizione di swap non ha file system
  - Raw data

## Partizionamento del disco (4/5)

- Le distribuzioni più user-friendly includono gli strumenti per gestire le partizioni
  - Ridimensionare le partizioni esistenti (es. Windows)
  - Formattare le partizioni create
- Esistono anche tool appositi
  - fdisk
  - gparted
    - ISO da masterizzare (o scrivere su pendrive) e avviare live



## Partizionamento del disco (5/5)

- In Linux le unità disco sono elencati in /dev
  - `/dev/sda` (disco principale)
  - `/dev/sdb` (disco secondario, chiavetta usb, ecc.)
  - ...
- Ogni partizione aggiunge un numero
  - `/dev/sda1`
  - `/dev/sda2`
- Elenco delle partizioni sul primo disco
  - `sudo fdisk -l /dev/sda`

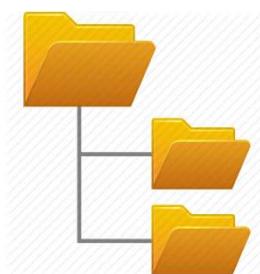
## Bootloader

- Il bootloader è un software che permette l'avvio del sistema operativo (o dei sistemi operativi)
- Contiene i puntatori al codice in ogni partizione che esegue i diversi sistemi operativi
- Nei sistemi Linux moderni più diffusi (es. Ubuntu) è praticamente autoconfigurante
  - Grub
  - Lilo



## 2. Il file system

1. Aspetti generali
2. Il file system Ext



## Aspetti generali del file system (1/2)

- Definisce la struttura di allocazione dei dati sul disco
- Diversi file system sono utilizzati a seconda del tipo di device (hard disk, cd, ecc) e del sistema operativo
  - FAT32 (Windows 95, 98, ...)
  - NTFS (Windows XP e seguenti)
  - Ext (Linux)
  - ISO 9660 (CD)

## Aspetti generali del file system (2/2)

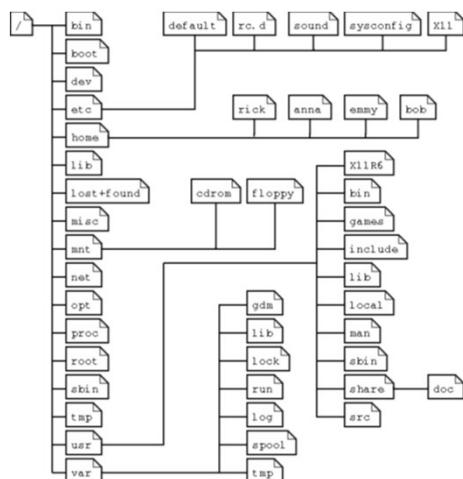
- Aspetti principali gestiti dai file system
  - Nomi dei file
    - Identificare una locazione all'interno del supporto
  - Directory
    - Come raggruppare diversi file
  - Metadati
    - Timestamp di creazione, modifica, tipo di file, creatore, etc.
  - Controllo dell'accesso
    - Accesso ai dati regolato in base all'utente e al programma
  - Integrità
    - Mantenere i dati leggibili in caso di eventi imprevisti (es. crash)

## Il file system Ext (1/3)

- Struttura ad albero

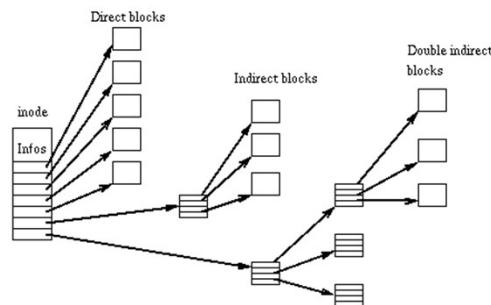
- Innestato sulla directory di root
  - /

- *"Everything is a file. If something is not a file, is a process"*



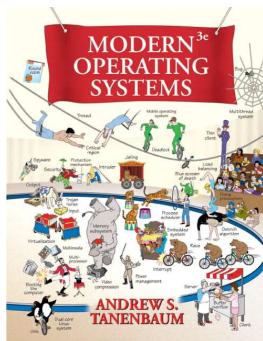
## Il file system Ext (2/3)

- Disco diviso in blocchi
- Ogni file o directory è rappresentato da *inode*
  - Inode contengono metadati
  - Puntatori ai blocchi che compongono il file
  - Puntatori organizzati in modo gerarchico



## Il file system Ext (3/3)

- Approfondimenti
  - *Andrew S. Tanenbaum – Modern Operating Systems*  
3° edition



## 3. File e directory

1. Tipi di file
2. Directory principali
3. Navigazione
4. Manipolazione di file e directory
  - Directory
  - File
  - Collegamenti



## Tipi di file

- Estensioni più comuni
  - *.bin* (file binario eseguibile)
  - *.deb* (pacchetto Debian per l'installazione di un software)
  - *.png, .jpg* (formati di immagine)
  - *.pdf* (formato di documento)
  - *.sh* (script Bash)
  - *.tgz, .tar.gz* (file compresso diffuso in ambito Linux)
  - *.zip* (file compresso)
  - *.txt* (file di testo)

## Directory principali

- Directory principali
  - */bin* (programmi comuni condivisi).
  - */boot* (file di avvio e kernel)
  - */dev* (riferimenti all'hardware)
  - */etc* (file di configurazione)
  - */home* (file personali degli utenti)
  - */opt* (software extra)
  - */root* (file personali dell'utente root)
  - */sbin* (programmi usati dal sistema e dall'utente root)
  - */usr* (programmi, librerie e documentazione dei programmi degli utenti)

## Navigazione (1/2)

- Navigare nelle directory
  - `cd /<directory>/<subdirectory>/<....>/`
  - Con il tasto «Tab» si può sfruttare l'autocompletamento
- Elencare il contenuto di una directory
  - `cd /<percorso_directory>/`
  - `ls`



## Navigazione (2/2)

- *Directory speciali*
  - `.` (*directory corrente*)
  - `..` (*directory di livello superiore*)
- Se iniziate con `/` (*cioè la directory di root*) bisogna indicare il percorso completo
- Senza il `/` si intende il percorso relativo alla posizione attuale
  - Equivalente a `./`

## Manipolazione di file e directory

1. Manipolazione di directory
2. Manipolazione di file
3. Gestione dei collegamenti



## Manipolazione di directory (1/2)

- Creare directory
  - `mkdir </directory/>`
- Rimuovere directory
  - `rmdir </directory/>`
  - La directory deve essere vuota
- Rimuovere directory non vuote (molto utile)
  - `rm -rf </directory/>`
  - Attenzione!

## Manipolazione di directory (2/2)

- Copiare directory
  - `cp -a <directory_sorgente>/ <directory_destinazione>`
- Spostare directory
  - `mv <directory_sorgente>/ <directory_destinazione>`
- Rinominare directory
  - `mv <directory_sorgente>/ <directory_destinazione>`

## Manipolazione di file (1/3)

- Creare un file
  - Tramite GUI del programma
  - Linea di comando
- Aprire o creare un file da linea di comando ha la stessa sintassi
  - `<nome_software> <nome_file>`
  - Es. `nano esempio.txt`
- Eliminare un file
  - `rm <percorso>/<nome_del_file>`



## Manipolazione di file (2/3)

- Copiare un file
  - `cp <file_sorgente.ext> </directory_destinazione>`
- Spostare un file
  - `mv <file_sorgente.ext> </directory_destinazione>`
- Rinominare un file
  - `mv <file_sorgente.ext> </directory_destinazione/nuovo_nome_file>`

## Manipolazione di file (3/3)

- Tutti i comandi sono eseguibili tramite GUI
- L'uso del terminale permette di essere più veloci
- I comandi a terminale sono programmabili con script

## Gestione dei collegamenti (1/2)

- Due tipi di collegamenti
  - *Symbolic link*
  - *Hard link*
- **Symbolic link**
  - Permettono di riferirsi ad un file grazie ad un collegamento ad esso. Il collegamento avviene memorizzando come contenuto del file di link il percorso dell'originale.
  - Comando *ln*
    - *ln -s /bin/sh <collegamento\_sh>*



## Gestione dei collegamenti (2/2)

- **Hard link**
  - Sono tipici dei filesystem ad albero. Svolgono le stesse funzionalità dei collegamenti simbolici ma realizzano il collegamento condividendo l'i-node.
  - Non esiste un file principale ed uno secondario.
  - Lo spazio occupato è inferiore rispetto ai collegamenti simbolici.
  - Non è possibile effettuare collegamenti ad un file di un filesystem diverso
  - Comando *ln*
    - *ln <primo\_file> <secondo\_file>*

## 4. Sicurezza del file system

1. Privilegi associati ai file
2. Privilegi associati alle directory
3. Modifica dei privilegi



RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

33

4. SICUREZZA DEL FILE SYSTEM – PRIVILEGI ASSOCIATI AI FILE

### Privilegi associati ai file (1/2)

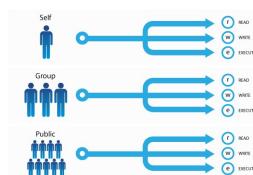
- Ad ogni file sono associati dei privilegi per tre categorie di utenti:
  - Chi possiede il file (*owner*)
  - Il gruppo dell’utente che possiede il file (*user group*)
  - Gli altri utenti
- Per ogni categoria di utenti, tre tipi di privilegi sono definiti
  - Lettura (*read*)
  - Scrittura (*write*)
  - Esecuzione (*execute*)

RUGGERO DONIDA LABATI – INTRODUZIONE A LINUX – LEZIONE 3 – IL FILE SYSTEM

34

## Privilegi associati ai file (2/2)

- I privilegi per ogni categoria sono definiti con 3 bit:
  - *r* (permesso di lettura)
  - *w* (permesso di scrittura)
  - *x* (permesso di esecuzione)
- Comando per visualizzare i privilegi associati al contenuto di una directory
  - *ls -l*



## Privilegi associati alle directory

- I privilegi associati alle directory sono definiti nel medesimo modo
  - 3 categorie di utenti
  - 3 tipi di privilegi
- Il significato è simile
  - L'accesso in lettura permette di conoscere l'elenco dei file contenuti
  - L'accesso in scrittura permette di creare, eliminare e rinominare dei file
  - L'accesso in esecuzione permette di attraversare una directory

## Modifica dei privilegi (1/3)

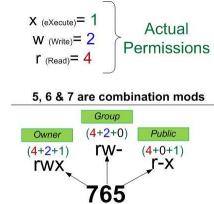
- Comando per la modifica dei privilegi
  - *chmod [parametri]*
- Parametri
  - Categoria di utenti a cui si vuole modificare i privilegi
    - *u* (user), *g* (group), *o* (other), *a* (all)
  - Aggiungere o rimuovere privilegi
    - + (aggiungere), - (rimuovere)
  - Tipo di privilegio da modificare
    - *r* (lettura), *w* (scrittura), *x* (esecuzione)

## Modifica dei privilegi (2/3)

- Esempi
  - *chmod g+rw esempio.txt* (aggiunge i permessi di lettura e scrittura al gruppo dell’utente che possiede il file)
  - *chmod o-rx esempio.txt* (rimuove i permessi di lettura ed esecuzione agli altri utenti)
  - *chmod ug+x esempio.txt* (aggiunge i permessi di esecuzione all’utente e al gruppo che possiedono il file)
  - *chmod a+rwx file\_pubblico.txt* (sconsigliato!)

## Modifica dei privilegi (3/3)

- Spesso si trovano i permessi espresso in formato numerico
  - 7 = permessi di lettura, scrittura, esecuzione (tutti e 3 i bit settati a 1)
  - 777 = permessi di lettura, scrittura, esecuzione a tutte le categorie di utenti
- Impostare i permessi di sola lettura:
  - `chmod 555 <nome_file>`



## 5. Gestione gruppi e utenti

1. Aggiungere un utente
2. Aggiungere un utente a un gruppo
3. Gestione gruppi e utenti tramite GUI



## Aggiungere un utente

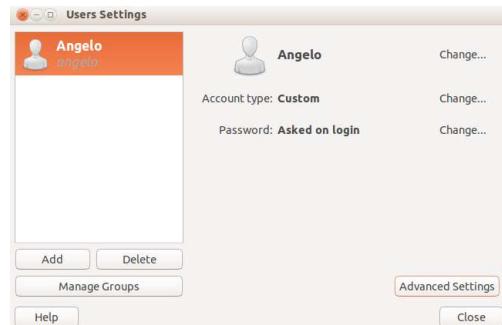
- Tramite linea di comando
  - `sudo adduser <nome_utente>`
  - Inserite la password del nuovo utente
- Il nuovo utente viene inserito in un gruppo appositamente creato
- Gestione dei privilegi di amministratore (capacità di eseguire `sudo`)
  - visudo

## Aggiungere un utente a un gruppo

- Creare un gruppo
  - `sudo groupadd <nome_gruppo>`
- Creare un nuovo utente direttamente nel nuovo gruppo
  - `sudo useradd -G <nome_gruppo> <nome_utente>`
- Aggiungere un utente esistente ad un gruppo
  - `usermod -a -G <nome_gruppo> <nome_utente>`

## Gestione gruppi e utenti tramite GUI

- Installare gnome-system-tools
  - Ubuntu software center
  - Linea di comando
    - `sudo apt-get install gnome-system-tools`
- Applicazione  
*'Users and groups'*
- Più intuitiva della linea di comando



## In sintesi

1. Installazione per uso stabile
2. Il file system
3. File e directory
4. Sicurezza del file system
5. Gestione gruppi e utenti



## 6. Esercizi (1/4)

- Provate i seguenti comandi e interpretate il risultato:
  - `cd`
  - `pwd`
  - `cd .`
  - `pwd`
  - `cd ..`
  - `pwd`
  - `ls`
    - A cosa servono le opzioni *d*, *i*, *R* ?
  - `ls -al`

## 6. Esercizi (2/4)

- Create le sottodirectory «dir1» e «dir2»
- Eliminate «dir2»
- Create «dir3» e spostatela dentro «dir1»

## 6. Esercizi (3/4)

- Eliminate una directory non vuota
- Create un file «hello.txt» nella directory home usando il terminale
  - Scrivete contenuti e salvatelo
- Create un symbolic link sul desktop al terminale
- Create un hard link sul desktop ad un file nella directory home

## 6. Esercizi (4/4)

- È possibile creare file che iniziano con «#» ?
  - E con «-» ?
- Create un file e modifiatene i permessi
  - es. r--r--r--
- Create un altro utente sulla macchina
- Rendete la vostra directory home inaccessibile all'altro utente
  - Funziona?
  - Ripristinate i permessi