

# linux@evening 2008

mercoledì 18/06/2008 – Ragusa

***“Sopravvivere alla linea di comando”***

*Mario Di Raimondo*

# Dopo l'installazione



- Ormai le moderne distribuzioni GNU/Linux sono molto **semplici** da installare:
  - interfacce grafiche;
  - procedure guidate intuitive ed automatizzate;
- questa semplificazione **non sempre è positiva**:
  - si tende a conoscere sempre meno i dettagli del sistema;
  - quando le procedure automatiche falliscono (nessuno è perfetto!) ci si ritrova bloccati!

# Un po' di informazioni per sopravvivere al cambiamento



- Oggi introdurremo alcuni **concetti basilari** essenziali all'utilizzo proficuo delle vostre installazioni di GNU/Linux:
  - gestione degli utenti;
  - il file-system;
  - la shell;
  - comandi di base;
  - ...
- durante la nostra trattazione faremo spesso riferimento alle distribuzioni Ubuntu e Debian.

# Utenti e convivenza



- Da sempre i sistemi UNIX sono sistemi **multiutente**:

*“per coesistere pacificamente sono necessarie delle regole”*

- ogni utente ha un proprio **“spazio vitale”** e non può interferire con il lavoro degli altri utenti:
  - home directory e proprietà dei file;
  - processi;
  - spazio su disco;
  - ...

# Ci sono sempre le eccezioni **solira**



- L'utente '**root**' può fare tutto:
  - operare sui processi/file degli altri utenti;
  - amministrare la macchina:
    - installare/rimuovere applicazioni;
    - avviare/fermare servizi;
    - modificare la configurazione;
    - ...
- questa separazione permette di rendere il sistema **più robusto** e ne preserva l'integrità anche in caso di azioni maldestre o maligne (virus/malware/trojan).

# E io chi sono?



- Il **modo corretto** di utilizzare una installazione è quello di impiegare un “utente semplice” per tutte le nostre **operazioni quotidiane** (navigare, inviare email, modificare documenti,...);
- riservare l'utilizzo dell'**utente amministratore** alle sole operazioni che lo richiedono (installare pacchetti nuovi, cambiare la configurazione,...);
- tipicamente queste due entità **sono separate** e hanno due password di accesso diverse.

# Il modello 'sudo'



- Una variante di questo modello consiste nella possibilità di cedere i diritti di amministrazione ad alcuni utenti specifici;
  - utilizzato in alcune distribuzioni GNU/Linux (come Ubuntu) e su Mac OS X;
- gli utenti in una **lista speciale** (i 'sudoers') possono eseguire operazioni di amministrazione, previa conferma della loro identità;
  - viene richiesta la **riconferma della password** (dell'utente semplice);
  - non viene chiesta la password di root (che può non esistere);
- vedremo dopo l'uso del comando 'sudo' per i comandi da shell.

# Il file-system UNIX



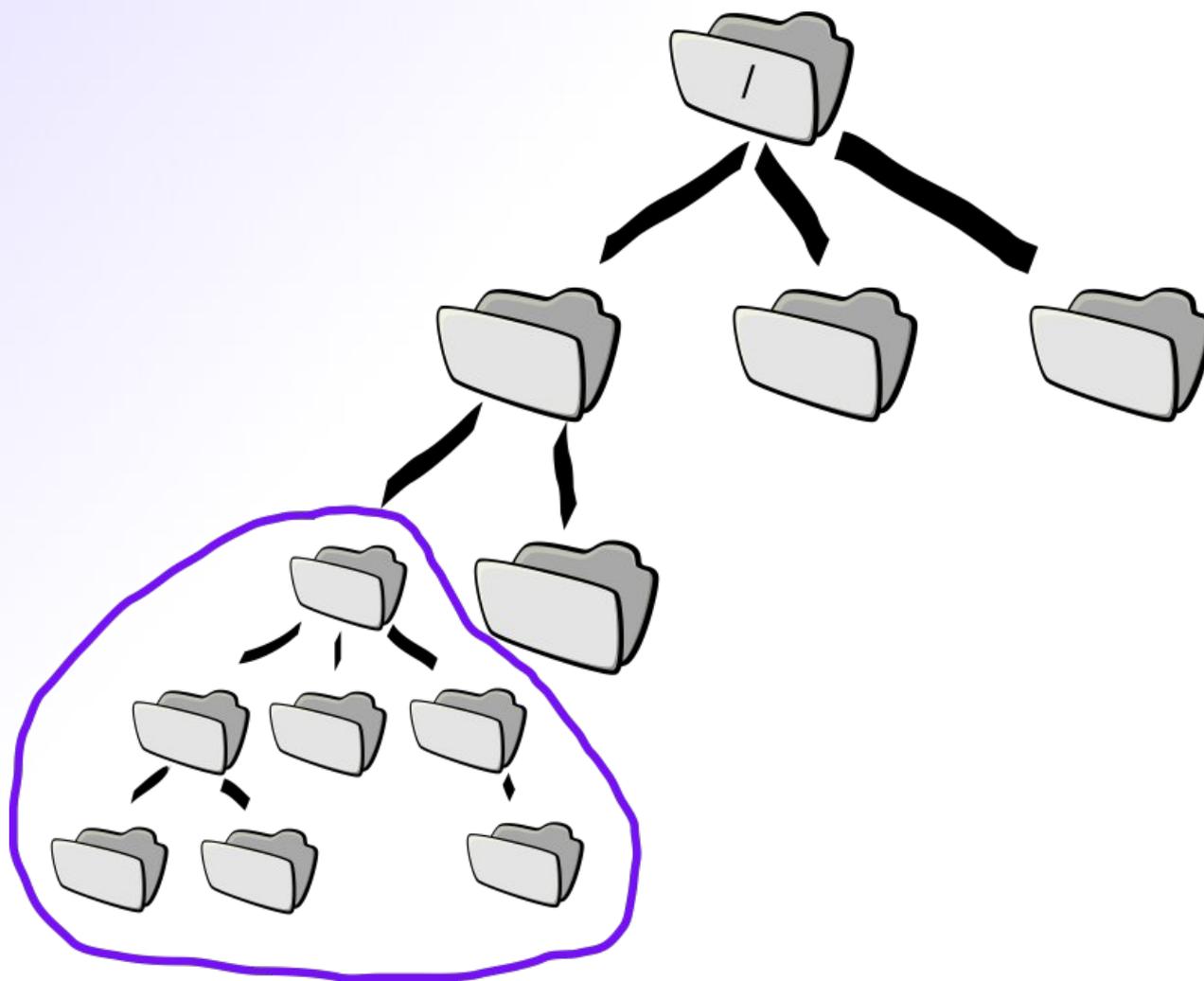
- Il **file-system** rappresenta l'organizzazione in cartelle e file dei contenuti dei nostri dischi;
- GNU/Linux supporta parecchi file-system (anche “stranieri”):
  - **ext3**, reiser-fs, ext2, xfs,...
  - vfat, ntfs, isofs, udf,...
- utilizzare file-system nativi UNIX permette di sfruttarne le loro peculiarità:
  - proprietà dei file/cartelle e diritti di accesso;
  - link simbolici e fisici;
  - ....

# Un unico albero



- Ad ogni disco o partizione (in generale volume) sul nostro sistema corrisponde un diverso file-system;
  - in Windows ogni volume ha una **lettera diversa**: A:, C:, D:,....;
    - i **percorsi** devono includere il volume:
      - C: \Windows \config.ini
      - D: \Documenti \Ufficio \Relaz.doc
  - nei sistem UNIX si usa un **modello ad albero unico**:
    - abbiamo un **file-system principale** (root=radice);
    - gli altri file-system sono **montati come foglie** di quello principale;
    - i percorsi partono tutti dalla radice:
      - /home/utente/documento.doc
      - /mnt/cdrom/foto.jpg

# Un unico albero



# Le cartelle di sistema



- Le cartelle che troviamo in una tipica installazione GNU/Linux sono:
  - /home/ : cartelle degli utenti;
  - /etc/ : file di configurazione;
  - /bin/ , /sbin/ : comandi principali di sistema;
  - /usr/ : tutti i file delle applicazioni installate;
  - /var/ : altri file di sistema (data-base, log, spool) e di applicazioni;
  - /mnt/ , /media/ : punti di montaggio per volume e dispositivi di memorizzazione removibili (pen-drive USB, iPod,...);
  - /dev/ : file speciali corrispondenti a tutti i dispositivi del sistema;

# La Shell



- La **shell** (o terminale o riga di comando) è uno strumento presente su tutte le installazioni di GNU/Linux:
  - presente ma **nascosto** (per semplificare l'interazione dell'utente non esperto);
- comunque **utile**:
  - modificare **configurazioni** per esigenze particolari;
  - impartire **comandi specifici** (magari reperiti su Internet);
  - **strumento universale** invariante su tutte le distribuzioni;
  - apre le porte all'**amministrazione remota** (essenziale in ambiti server);
  - possibilità di costruire comandi molto potenti e complessi racchiusi all'interno di **script** (automazione).

# Dov'è? Non la trovo!



- Su Ubuntu:  
Applicazioni > Accessori > Terminale
- Si presenta un **prompt** di questo tipo:  
**utente@macchina:cartellacorrente\$ \_**
- da qui possiamo impartire comandi che verranno eseguiti, in modo interattivo, dal nostro utente.
- Primi esempi di comandi:
  - **ls** : lista il contenuto di una cartella;
  - **cd** : cambia la directory corrente;
  - **pwd** : visualizza il nome della cartella corrente;

# Tante opzioni!



- I comandi hanno tipicamente **varie opzioni** per modificare il loro comportamento:
  - **ls -l** : visualizza più dettagli;
  - **ls -a** : visualizza anche file/cartelle nascosti (che iniziano con il punto '.')
  - **ls -r** : visualizza anche il contenuto delle sotto-cartelle;
- ricordarle tutte non è né necessario né utile; possiamo sempre **chiedere aiuto**:
  - **ls --help**: visualizza una breve lista di opzioni;
  - **man ls**: visualizza una pagina di manuale sul comando 'ls';
    - per averle **in italiano** potete installare il pacchetto 'manpages-it' (vedrete dopo come fare);
    - quanto detto vale per quasi tutti i comandi.

# Altri comandi di base



- **whoami**: visualizza il nome dell'utente attuale;
- **cat nome-file**: visualizza il contenuto di un file;
- **mkdir nome-cartella**: crea una cartella;
- **rmdir nome-cartella**: cancella una cartella vuota;
- **cp nome-file destinazione**: copia uno o più file;
- **mv nome-file destinazione**: sposta uno o più file o rinomina;
- **rm nome-file**: cancella un file;
  - **rm \*.doc**: cancella tutti i file con estensione .doc;
  - **rm -r nome-cartella**: cancella la cartella con tutto il suo contenuto;

# Completamento automatico



- Digitare comandi da riga di comando può diventare **noioso**, ma la shell **ci aiuta**:
  - **auto-completamento** pigiando il tasto tabulazione (TAB):
    - completa il **comando** o **nome di file/cartella** che stiamo digitando considerando **le possibilità disponibili**.

# Invocare un programma

- I programmi che utilizziamo ogni giorno (browser, programma posta, editor, ...) volendo possono essere lanciati da riga di comando;  
**\$ gedit**
- questo però, tipicamente, “blocca” la shell finché non termina la sua esecuzione;
- per evitare ciò possiamo avviarli in “**modalità background**” (non bloccante):  
**\$ gedit &**
- un modo più semplice per invocare un programma è quello di utilizzare lo strumento “Esegui...” (ALT-F2);
  - in questo caso non vedremo l'output del comando.

# Uso del comando 'sudo'

- Torniamo al discorso legato all'amministrazione della nostra installazione;
- alcuni comandi lanciati dalla shell richiedono i **diritti di amministratore** per essere efficaci:
  - esempio: installazione pacchetti, modifica di file di configurazione;
- per eseguire tali comando dovremo far precedere il nostro comando dalla parola chiave '**sudo**';
  - ci verrà eventualmente richiesta la conferma della password;
  - esempio:

```
$ sudo gedit /etc/apt/sources.list
```

**Ancora terrorizzati dalla shell?**

