

Laboratorio di Amministratore di Sistema

Lab1. Amministrazione di Linux

[Cisco ITESS II - Chapter 10]

Università di Venezia – Facoltà di Informatica
feb-mag 2013 - [A. Memo](#), [M. Squarcina](#)



Procedure di installazione di GNU/Linux



- 10.1 Interfaccia utente
- 10.2 Utenti e gruppi
- 10.3 Filesystem e servizi
- 10.4 Demoni



Metodi di login

- Gli utenti possono effettuare il login su un sistema GNU/Linux tramite Command-Line Interface (CLI), simile all'interfaccia presente in Windows 2000.
- Le informazioni richieste agli utenti sono soltanto username e password.

```
Ubuntu 9.04 linux-sandbox tty1
linux-sandbox login: marco
Password:
Last login: Thu Jul  9 14:12:55 CEST 2009 on tty1
Linux linux-sandbox 2.6.28-11-generic #42-Ubuntu SMP Fri Apr 17 01:57:59 UTC 2009 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

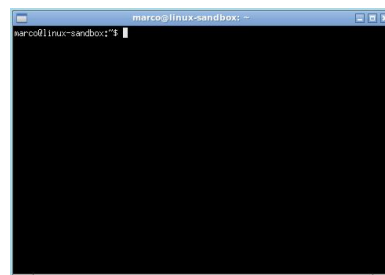
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
marco@linux-sandbox:~$
```

prompt ←



Interfaccia grafica (GUI)

- X.Org è l'implementazione open source ufficiale del sistema grafico X Window System.
- Gli utenti di GNU/Linux possono modificare a loro piacimento l'aspetto e il comportamento della GUI. Nella maggior parte dei casi, l'interfaccia grafica non risulterà troppo diversa da quella impiegata tradizionalmente in Windows.
- Un **terminale virtuale** è un'applicazione che fornisce un prompt dei comandi all'interno di una finestra nello schermo.
- **xterm** è il terminale più diffuso essendo incluso in X.Org.



Command Line Interface



- L'interfaccia a riga di comando di GNU/Linux permette agli utenti di interagire con il sistema in modo simile all'interprete dei comandi di Windows.
- L'inserimento casuale di comandi può seriamente compromettere l'integrità dei propri dati e del sistema.

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ whoami  
marco  
marco@linux-sandbox:~$ groups  
marco adm dialout cdrom plugdev lpadmin sambashare a  
dmin  
marco@linux-sandbox:~$ id  
uid=1000(marco) gid=1000(marco) groups=4(adm),20(dia  
lout),24(cdrom),46(plugdev),107(lpadmin),115(sambash  
are),123(admin),1000(marco)  
marco@linux-sandbox:~$
```

1. Using cd with an absolute pathname:

```
$ cd /home
```

2. Using cd with a relative pathname:

```
$ cd user2/dir1
```

3. Use cd without a directory name returns you to your home directory:

```
$cd
```

Command Line Interface



whoami

- mostra lo userid dell'utente corrente

groups

- mostra i gruppi a cui appartiene l'utente

id

- mostra gli id reali (ed effettivi) degli utenti e dei gruppi

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ whoami  
marco  
marco@linux-sandbox:~$ groups  
marco adm dialout cdrom plugdev lpadmin sambashare a  
dmin  
marco@linux-sandbox:~$ id  
uid=1000(marco) gid=1000(marco) groups=4(adm),20(dia  
lout),24(cdrom),46(plugdev),107(lpadmin),115(sambash  
are),123(admin),1000(marco)  
marco@linux-sandbox:~$
```

user-ID reale , group-ID reale - quelli passati al momento del login

user-ID effettivo , group-ID effettivo - quelli usati nella verifica dei permessi e

controllo accessi; coincidono con i reali, escluso x i bit Suid e Sgid

user-ID salvato , group-ID salvato - copia degli effettivi del proc. padre

user-ID di filesystem , group-ID di filesystem - copia degli effettivi, per rendere più sicuro NFS

Command Line Interface

comando “man”



- Il comando **man** presenta un'interfaccia ai manuali di riferimento in linea.
- Fornisce la guida all'utilizzo per ognuno dei comandi disponibili.
- Tramite il comando:
man intro
è possibile ottenere una introduzione all'utilizzo della riga di comando e una breve descrizione dei principali comandi.
- Una pagina del manuale è composta da varie parti identificate da differenti etichette (nome, sintassi, descrizione, etc)

The man -k mkdir Command

```
[root@systemA1 root]# man -k mkdir
mkdir      (1) - make directories
mkdirhier  (1k) - makes a directory hierarchy
[root@systemA1 root]#
```

Common Man Page Headings

| Command | Description |
|-------------|---|
| name | Contains the name of the command and other commands that may accomplish the same thing. |
| Synopsis | Shows the syntax of the command with any allowable options and arguments. |
| Description | Gives an overview of what the command does. |
| Operands | The target of the command or what the command will take effect on such as a directory or a file. |
| Options | Switches that can change the function or effect of the command. They are normally preceded by a dash (-) or minus sign. |
| see also | Refers users to other related commands and subjects, such as name, synopsis, description, options, and operands. |

Command Line Interface

comando “ls”



- Tutti i comandi devono avere almeno un nome, la relativa sintassi e la descrizione.
- Uno dei principali comandi è **ls**. Lo scopo di tale comando è quello di visualizzare il contenuto della directory corrente.
- La sintassi è la seguente:
ls [opzioni] [files]

The man Page for the ls Command

```
LS(1)                                FSF                                LS(1)
NAME
  ls - list directory contents
SYNOPSIS
  ls[OPTION]... [FILE]...
DESCRIPTION
  List information about the FILES (the current
  directory by default). Sort entries
  alphabetically if none of -cfuSUX nor --sort
  -a, --all
    do not hide entries starting with .
  -A, --almost-all
    do not list implied . and .1.
  -b, --escape
    print octal escapes for nongraphic
    characters
  --block-size = SIZE
    use SIZE-byte blocks
  -B, --ignore-backups
    do not list implied entries ending with ~
  -c
    with -lt: sort by, and show, ctime (time
    of last modification of file status
    information) with -l: show ctime and sort
    by name otherwise: sort by :
```

Command Line Interface

comando "ls"

The ls Command

```
1. Using the basic ls command (from /home/user2)
$ ls
dante  dir1  dir3  file1  file3  practice
dante_1 dir2  dir4  file2  file4

2. Using the ls command with a relative pathname
$ ls dir1
coffees  fruit  trees

3. Using the ls command with an absolute pathname
$ ls /var/mail
user1  user2  user3  user4  user5
```

Common ls Command Options

| Option | Description |
|------------------|---|
| -a or -all | Normally the <code>ls</code> command only displays the files in a directory that are not hidden. Recall that hidden files in a Linux system are indicated by a leading period. For example, the <code>.bash_profile</code> file would be hidden and will only be listed when the <code>-a</code> or <code>-all</code> parameter is added to the command. |
| -color | When this parameter is added to the <code>ls</code> command the files, directories, symbolic links, and so on are all differentiated by a color code. This makes it easier to distinguish between the different types of files. |
| -d or -directory | When the <code>ls</code> command is issued without this parameter and the directory is listed as one of the <i>[files]</i> , then the entire contents of the directory will be listed. When this parameter is entered then just the directory name will be listed. |
| -l | When the <code>ls</code> command is entered without the <code>-l</code> parameter just the filenames of the files, directories, and so on are displayed. If the <code>-l</code> parameter is used along with the command much more information is displayed such as the file's permission string, the files owner, group, size, and creation date. This parameter is known as long listing. |
| -p or -filetype | This parameter attaches an indicator to the end of the output so the type of file is known. For example, <code>a</code> / is a directory, <code>@</code> is a symbolic link, <code>=</code> is a socket, and <code> </code> is a pipe. |
| -R or -recursive | This parameter causes the <code>ls</code> command to display the directories recursively. This means that when the <code>ls</code> command is issued it will display the contents of that directory plus all the files in the subdirectories. |

Command Line Interface

comando "ls"



| | |
|---------------------------------|---|
| <code>ls l*</code> | per elencare tutti i file che iniziano con l |
| <code>ls *.txt</code> | per elencare tutti i file con estensione txt |
| <code>ls c*o.txt</code> | per elencare tutti i file che iniziano con c, che terminano con o, ed hanno estensione .txt |
| <code>ls c?o.txt</code> | per elencare tutti i file con nome da tre lettere, la prima c, la seconda qualsiasi, la terza o |
| <code>ls a >lista.txt</code> | per creare un file lista.txt contenente l'output prodotto dal comando ls a (cioè ...) |
| <code>dotfile</code> | di solito i file nascosti |
| <code> more</code> | una pagina alla volta, e poi ENTER per una riga, SPACE per una pagina, q per terminare |

Alcuni semplici comandi



| | |
|----------------------|---|
| cat x | per visualizzare il contenuto del file |
| cat x y z | concatena a video i tre file |
| cp x y | copia il file x nel file y |
| mv x y | muove il file x nel file y (rinomina) |
| rm x | cancella il file x |
| rmdir x | cancella una directory (solo se vuota) |
| rm -r x | cancella la directory e tutto il suo contenuto |
| find x | cerca il file x |
| chmod x | cambia gli attributi al file x |
| | |
| X > y | redirezione dell'uscita di X nel file y |
| X >> y | accodamento dell'uscita di X nel file y |
| X < y | redirezione dell'input del file X dal file y |
| | |
| X y | pipeline, l'uscita del comando X diventa l'ingresso del comando y |
| ls -l lpr | esempio: stampo la lista dei file |
| x tee nomefile y | l'uscita del comando x viene salvata sul file nomefile e poi diventa l'ingresso del comando y |
| | |
| more x | visualizza il contenuto del file x a pagine |
| tail -N | visualizza le ultime N righe di un file di testo (default 10) |
| sort x | ordina i dati contenuti nel file x |
| grep x y | ricerca la stringa x nel file y |

Shell in GNU/Linux



- Le shell in GNU/Linux sono degli interpreti di comandi passati da standard input o da un file.
- L'interprete dei comandi di MS-DOS è simile, ma in GNU/Linux è possibile utilizzare una shell solo quando desiderato. Si hanno a disposizione varie shell:
 - Bourne Shell (sh), originale
 - Korn Shell, shell con sintassi C-like
 - Born Again Shell (bash), evoluzione di Bourne Shell
- La Born Again Shell, **bash**, è estremamente diffusa come shell di default in molti sistemi "UNIX-Like".

The Korn Shell (ksh)

```
[student1@student1]$ ps
PID TTY      TIME CMD
3914 pts/1    00:00:00 bash
3914 pts/1    00:00:00 ps
[student1@student1]$ echo $SHELL
/bin/bash
[student1@student1]$ ksh
$ ksh
[student1@student1 ~]$ ps
PID TTY      TIME CMD
3914 pts/1    00:00:00 bash
4162 pts/1    00:00:00 ksh
4167 pts/1    00:00:00 ps
$
```

The Linux Bourne Again Shell (bash)

```
[student1@student1]$ ps
PID TTY      TIME CMD
3914 pts/1    00:00:00 bash
3914 pts/1    00:00:00 ps
[student1@student1]$ echo $SHELL
/bin/bash
[student1@student1]$
```

The Linux Bourne Again (bash) shell

Editor di testo: VI(M)



- Le maggiori distribuzioni GNU/Linux includono un editor di testo non grafico estremamente potente: **vi**.
- **vi** permette di modificare e creare file di configurazione e script.
- **vi** opera secondo tre modalità:
 - *normal (command)* : modalità di avvio di vi. Tutto ciò che viene digitato è interpretato come un comando.
 - *insert*: permette di inserire del testo.
 - *visual*: consente selezioni sul testo.

```
marco@linux-sandbox: ~  
VIM - Vi IMproved (VI Migliorato)  
versione 7.2.79  
di Bram Moolenaar et al.  
Vim è 'open source' e può essere distribuito liberamente  
Diventa un utente Vim registrato!  
batti :help register<Invi> per informazioni  
batti :q<Invi> per uscire  
batti :help<Invi> o :f1 per aiuto online  
batti :help version7<Invi> per informazioni su versione  
0,0-1 Tut
```

gedit

awk



- **awk** è un linguaggio che permette di processare file, ordinare i dati, eseguire operazioni aritmetiche e molto altro.
- L'utilità **awk** esegue la scansione di un file di input riga per riga alla ricerca di *patterns* che facciano il *match* con quelli passati da command line. Nel caso in cui si verifichi un match, awk avanza nella sequenza di programmazione, altrimenti processa la riga successiva del file di input.

Summary of Common awk Metacharacters

| Metacharacter | Meaning |
|---------------|------------------------------|
| . | Match any character |
| * | Match zero or more character |
| ^ | Match beginning of line |
| \$ | Match end of line |
| \ | Escape character following |
| [] | List |
| {} | Match range of instance |
| + | Match one more preceding |
| ? | Match zero or one preceding |
| | Separate choices to match |

Sintassi: **awk '{pattern + action}' {filenames}**

awk



```
echo uno due | awk '{print $1}'  
# uno
```

```
echo uno due | awk '{print $2}'  
# due
```

awk



data file

```
nomefile = inventory  
dresses    order  5  
shoes      good   12  
shirts     order  6  
socks      good  15  
pants      good   8
```

command and results

```
awk '{print $1, $3}' inventory  
dresses 5  
shoes 12  
shirts 6  
socks 15  
pants 8
```

command and results

```
awk '/good/ {print $3}' inventory  
12  
15  
8
```


awk



Come si possono estrarre solo i nomi dei file e le loro dimensioni della directory corrente?

```
ls
ls -l
ls -l | tail
ls -l | tail 4
ls -l | tail 4 > outlist
```

awk '{print \$4,\$8}' outlist

outlist →

```
-rw-r----- 1 rossi 413  Feb 14 13:37  dati1
-rw-r----- 1 rossi 2851 Feb 14 13:32  dati2
-rw-r----- 1 rossi 412  Aug 3  1990   prog1.c
-rw-r----- 1 rossi 668  Feb 14 13:28   prog2.c
```

outlist →

```
413  dati1
2851 dati2
412  prog1.c
668  prog2.c
```

Utenti e Gruppi in GNU/Linux



- In GNU/Linux, la possibilità di accedere contemporaneamente da parte di più persone - senza che queste possano interferire fra loro - a un sistema, è garantita dall'esistenza degli account utente.
- I termini “**utente**” e “**account**” sono spesso usati come sinonimi.
- Le informazioni riportate a destra sono specificate in due file nei sistemi GNU/Linux:

`/etc/passwd`
`/etc/shadow`

- Username
- Login privileges
- Password protection
- Permissions
- Home directory
- User and group ids
- Default shell

Utenti e Gruppi in GNU/Linux



- L'utente più importante è rappresentato dall'account di **root**, chiamato anche **superuser**.
- Questo account permette di svolgere le attività di amministrazione del sistema.
- In Ubuntu 9.04 tale account è disabilitato. Non è possibile quindi effettuare direttamente il login come root. E' infatti preferibile ricorrere al tool **sudo**.
- Generalmente l'account di root può essere utilizzato in vari modi:
 - root login
 - su (→ exit)
 - sudo
 - SUID & SGID root files

Aggiungere e rimuovere utenti



- E' possibile aggiungere nuovi utenti in modo interattivo tramite il comando:
sudo adduser username
- Il comando originale **useradd**, invece, è più conveniente se si volessero aggiungere utenti tramite uno script.
- Per maggiori dettagli sull'utilizzo dei comandi, consultare le relative pagine di manuale.
- La creazione di un utente è un processo che richiede numerose operazioni come la creazione della home directory e l'impostazione dei permessi di default.

```
studente@linux-sandbox: ~  
marco@linux-sandbox:~$ sudo adduser studente  
[sudo] password for marco:  
Adding user 'studente' ...  
Adding new group 'studente' (1001) ...  
Adding new user 'studente' (1001) with group 'studente' ...  
Creating home directory '/home/studente' ...  
Copying files from '/etc/skel' ...  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
Changing the user information for studente  
Enter the new value, or press ENTER for the default  
Full Name []: Mario Rossi  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [Y/n] Y  
marco@linux-sandbox:~$ su - studente  
Password:  
Added user studente.  
studente@linux-sandbox:~$ whoami  
studente  
studente@linux-sandbox:~$
```

Aggiungere e rimuovere utenti



- Un amministratore per assegnare o modificare una password ad un utente deve utilizzare il seguente comando:

`sudo passwd username`

- Per rimuovere un account:

`sudo userdel username`

- Per rimuovere un account, la sua home directory e tutte le directory contenute:

`sudo userdel -r username`

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ ls /home/  
marco studente  
marco@linux-sandbox:~$ sudo userdel -r studente  
[sudo] password for marco:  
marco@linux-sandbox:~$ ls /home/  
marco  
marco@linux-sandbox:~$
```

Gestione utenti



- Per disabilitare un account utente senza rimuoverlo è sufficiente rimuovere manualmente la sua password di accesso.
- Le password sono memorizzate in un unico file:
`/etc/shadow`
- Tale file può essere modificato con un editor di testo come `vi` o `gedit`.

| Task | Command Syntax | Example |
|--------------------|---|--------------------------------------|
| Change a user name | <code>usermod -l <newname> <oldname></code> | <code>usermod -l johndoe jdoe</code> |
| Change a password | <code>passwd <username></code> | <code>passwd johndoe</code> |
| Delete an account | <code>userdel <username></code> | <code>userdel johndoe</code> |

Gestione utenti

- Per disabilitare un account è possibile modificare il file `/etc/shadow` inserendo un asterisco "*" all'inizio del secondo campo relativo all'utente.

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ sudo grep studente /etc/shadow  
studente:*:14435:0:99999:7:::  
marco@linux-sandbox:~$ su - studente  
Password:  
su: Authentication failure  
marco@linux-sandbox:~$
```

Creazione gruppi e aggiunta di utenti ai gruppi

- L'appartenenza degli utenti ai gruppi è definita dal file `/etc/group`
- Per effettuare il log-in in un altro gruppo dopo l'accesso al sistema, si usa il seguente comando:
`newgrp <group name>`
- Per creare un nuovo gruppo:
`sudo groupadd <group name>`
- Il comando `useradd` è usato per aggiungere utenti a un gruppo.
- Il comando `gpasswd` permette di amministrare i gruppi attuali.

| Option | Description | gpasswd |
|-----------------|---|---------|
| -a <user> | This option will add the specified user to the specified group | |
| -d <user> | This option will delete the specified user from the specified group. | |
| -R | Using this option will specify to this group not to allow new members to be added with the newgrp command. | |
| -r | This option will remove the password from the group. | |
| -A <user> [...] | This option can be issued by the root user to specific group administrators that will have privileges to add or remove users from groups as well as specify group passwords. Keep in mind that this option overrides any previous lists of administrators. All the group administrators | |

Creazione e condivisione di directory



- Creare file e directory da GNU/Linux richiede la conoscenza dei seguenti comandi.

| Function | Command Syntax | Example |
|-------------------------|---|--|
| Create a directory | <code>mkdir <directory></code> | <code>mkdir docs</code> |
| Copy a directory | <code>cp </path/olddirectory> </path/newdirectory></code> | <code>cp docs docs2</code> |
| Copy a file | <code>cp </path/oldfile> </path/newfile></code> | <code>cp /save/docs/notes /save/docs2/notescopy</code> |
| Move/Rename a directory | <code>mv </path/olddirectory> </path/newdirectory></code> | <code>mv /save/docs2 /docs2</code> |
| Move/Rename a file | <code>mv </path/oldfile> </path/newfile></code> | <code>mv /docs2/notescopy /save/docs/notescopy</code> |
| Delete a directory | <code>rmdir </path/directory></code> | <code>rmdir /docs2</code> |
| Delete a file | <code>rm </path/file></code> | <code>rm /save/docs/notescopy</code> |

Utilizzo dei comandi grep e find



- Il comando **find** è usato per trovare la posizione all'interno del file system di una risorsa, supponendo di conoscere almeno parzialmente il suo nome.
- E' possibile specificare opportuni filtri nella ricerca ed eseguire comandi attraverso tutto l'albero delle directory.

```
$ find /home/marco -type f -size -350K -atime -4
```

| | |
|--------------------------|---|
| <code>/home/marco</code> | <i>path where to search the object</i> |
| <code>-type f</code> | <i>the object to search is f=file, d=directory, c=character</i> |
| <code>-size -350K</code> | <i>the object has a dimension less than 350 KB</i> |
| <code>-atime -4</code> | <i>the object was last accessed less than 4 days ago</i> |

Utilizzo dei comandi grep e find



- Il comando **grep** permette la ricerca di un determinato pattern all'interno di una lista di file.
- Per effettuare la ricerca di una stringa all'interno di uno o più file, è necessario includere tale stringa fra singoli apici.

```
$ grep 'Ca Foscari' Lab_A Lab_B
```

'Ca Foscari' *pattern to search*

Lab_A Lab_B *the files' names where search the pattern*

Password e permessi



- Gli amministratori di sistema su GNU/Linux hanno a disposizione un controllo più granulare sui permessi rispetto a Windows, potendo operare su 3 differenti categorie: *esecuzione, lettura e scrittura*.
- Il permesso di esecuzione controlla la possibilità di "attraversare" una directory, mentre quello di lettura la possibilità di visualizzarne il contenuto.

```
-rwxr--r--  
0123456789
```

```
Terminal
Window Edit Options Help

$ touch pluto
$ ls -l
-rw-r--r-- 1 user2 staff 0 May 24 17:25 pluto

$ mkdir secure
$ ls -l
-rw-r--r-- 1 user2 staff 0 May 24 17:25 pluto
drwxr-xr-x 2 user2 staff 512 May 24 17:25 secure
```

| User Category | User(owner) | Group | Other |
|---------------|--------------------|--------------|--------------|
| New File | Read/Write | Read | Read |
| New Directory | Read/Write/Execute | Read/Execute | Read/Execute |

- 0 "-" is the type of the file. "d" for directory, "l" for link, "-" for regular file
- 1-3 "rwx" are permissions for the owner of the file
- 4-6 "r--" are permissions for the group
- 7-9 "r--" are permissions for others.

| | | |
|---|---------|--------------------|
| r | read | { modify delete |
| w | write | |
| x | execute | |

Password e permessi



- I permessi sui file e le directory sono gestiti tramite i comandi `chown` e `chmod`.
- `chown` imposta l'appartenenza di un file a un determinato utente e gruppo:
`chown username:group filename`
- Con il comando `chmod`, invece, vengono impostati i permessi relativi all'utente proprietario del file, al gruppo e a tutti gli altri. Il concetto di “ownership” di una risorsa è quindi fondamentale in GNU/Linux.

`chmod mode filename`

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ touch foo  
marco@linux-sandbox:~$ ls -la foo  
-rw-r--r-- 1 marco marco 0 2009-07-10 06:23 foo  
marco@linux-sandbox:~$ sudo chown marco:users foo  
marco@linux-sandbox:~$ chmod 755 foo  
marco@linux-sandbox:~$ ls -la foo  
-rwxr-xr-x 1 marco users 0 2009-07-10 06:23 foo  
marco@linux-sandbox:~$
```

Password e permessi



Sono inoltre presenti tre ulteriori bit con il seguente significato:

- **sticky bit** attribute
 - Se un utente ha il permesso di scrivere sulla directory, allora tale utente può modificare o rimuovere i file presenti in quella directory anche se non è il proprietario di quei file.
 - Con lo sticky bit per la directory attivo, solo i proprietari dei file possono rimuoverli o modificarli (sticky bit usato per /tmp)

```
$ ls -ld testdir  
drwxrwxrwx 2 rtalbot rtalbot 512 Jul 15 21:23 testdir  
$ chmod +t testdir  
drwxrwxrwx 2 rtalbot rtalbot 512 Jul 15 21:23 testdir
```

Password e permessi



- **SUID**, o setuid (modifica l'ID dell'utente che esegue il file): se un file ha il bit SUID attivato, quando il file verrà eseguito da un qualsiasi utente, il relativo processo avrà gli stessi diritti del proprietario di quel file
- **SGID**, o setgid (modifica l'ID del gruppo che esegue il file): analogo per il gruppo.

Per le directory, il significato è diverso: quando viene creato un file in quella directory, questo eredita i permessi dal gruppo della directory e non dal gruppo di chi sta creando il file.

Mount e gestione del filesystem



I due comandi necessari a montare e smontare partizioni e filesystem sono **mount** e **umount**:

| The -o Option | |
|---------------|---|
| Options | Description |
| nodev | Does not interpret special devices on the filesystem. This is a useful security option. |
| Noexec | Does not allow execution of binaries on this filesystem. This is also a useful security option. |
| Nosuid | Does not interpret setuid or setgid flags on the filesystem. This is also a useful security option. |
| Device | This is the filename that is associated with a particular disk device like /dev/cdrom for example. |
| Mountpoint | This specifies the directory to which the device's content should be attached. |

| The mount Command Options | |
|---------------------------|--|
| Option | Description |
| -a | Mounts all the filesystems listed in /etc/fstab. Exceptions are those marked as 'noauto', excluded by the -t flag, or those that are already mounted. |
| - | Does everything except for the actual system call. This option is useful in conjunction with the -v flag to determine what the mount is actually trying to do. |
| -f | Forces the mount of an unclean filesystem, dangerous, or forces the revocation of write access when downgrading a file system's mount status from read-write to read-only. |
| -r | Mounts the filesystem as read-only. This is identical to using the rdonly argument to the -o option. |
| -t fstype | Mounts the given filesystem as the given filesystem type, or mounts only filesystems of the given type. If given the -a option, 'ufs' is the default filesystem type. |
| -w | Mounts the filesystem as read-only. This is identical to using the rdonly argument to the -o option. |

mount [-alrsvw] [-t fstype] [-o options] device dove
umount [-anrv] [-t fstype] [device|mountpoint]

Mount e gestione del filesystem

- Il comando **df** (*disk space of the file system*) fornisce informazioni sul disco fisso e sulle partizioni, come lo spazio totale, lo spazio usato, quello disponibile e il punto di mount.

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ df -hT  
Filesystem      Type      Size      Used Avail Use% Mounted on  
/dev/sda1       ext3      4,4G      1,4G  2,8G  34% /  
tmpfs           tmpfs     502M      0      502M  0% /lib/init/rw  
varrun          tmpfs     502M     320K  502M  1% /var/run  
varlock         tmpfs     502M      0      502M  0% /var/lock  
udev            tmpfs     502M    140K  502M  1% /dev  
tmpfs           tmpfs     502M      0      502M  0% /dev/shm  
lrn             tmpfs     502M     2,4M  499M  1% /lib/modules/2.6.28-11-generic/volatile  
marco@linux-sandbox:~$
```

| Options | Description |
|------------------------|--|
| -h or --human-readable | This option tells the command to display the output labeled in units of kilobytes (k), megabytes (M), or gigabytes (G) instead of the standard 1024 bytes blocks which is the default output without this option |
| -i or --inodes | Inodes are data structures on Linux filesystems that contain file information. Using this option displays information on the consumption of inodes as opposed to displaying disk space used, which is the default. |
| -l or --local | Using this option will tell the command to ignore network file systems and return information based on local or fixed disks only. |
| -T or --print-type | This will show output of the filesystem type code along with any other information. |

Mount e gestione del filesystem

- Il comando **du** (*disk usage statistics*) fornisce informazioni inerenti ad una determinata directory.

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ sudo du -h --max-depth=1 /usr/  
24K    /usr/games  
39M    /usr/sbin  
362M   /usr/share  
104K   /usr/local  
4,0K   /usr/X11R6  
220M   /usr/lib  
55M    /usr/src  
7,4M   /usr/include  
131M   /usr/bin  
811M   /usr/  
marco@linux-sandbox:~$
```

| The du Command Options | |
|-------------------------|--|
| Option | Description |
| -c or --total | This will display a list of the total space used by the file or directory. |
| -h or --human-readable | This option tells the command to display the output labeled in units of kilobytes (k), megabytes (M), or gigabytes (G) instead of the standard 1024 bytes blocks, which is the default output without this option. |
| -S or --separate-dirs | This option will list the output totals for the individual directories and subdirectories. Without this option the du command lists the total space with subdirectories included. |
| -s or --summarize | This option will summarize the output for each main file or directory instead of each subdirectory. |
| -max-depth=N | When using the du command the output can be sometimes very long and unmanageable. Using this command will limit the output to a maximum of N subdirectories deep. For example, if this command were used where N=1 then it would list directories and subdirectories for the directory listed plus all the subdirectories directly below it, but none below that subdirectory. |
| -x or --one-file-system | Using this command will limit the output of files and directories that are located on one partition. Otherwise the du command will scan for all files and subdirectories listed for the specified directory across all partitions. |

File di configurazione

Utente
Sistema

```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ ls -a  
*      .dmrc      .sudo_as_admin_successful  
..     .gconf     .viminfo  
.bash_history  .gnome2    .Xauthority  
.bash_logout  .local     .xinitrc  
.bashrc       .noc       .xsession-errors  
.dbus         .profile  
marco@linux-sandbox:~$
```

- I file di configurazione **utente** contengono impostazioni come il windows manager utilizzato, eventuali script di autorun e configurazioni varie (es. Client mail)
- Il nome di tali file inizia con un “.” e per questo motivo sono considerati file nascosti.
- Per visualizzarli si può utilizzare il comando

ls -a

| User Configuration Files | |
|--------------------------|--|
| File Type | Description |
| .kde and .kderc | As mentioned before, these are the files that set the global desktop settings for the K Desktop Environment. |
| .gnome | This is the equivalent of the .kde directory for the KDE interface. |
| .bashrc | This is a script that runs automatically when the bash shell runs. It contains many of the default environmental variables that are set when a user logs on. |
| .netscape | This is the configuration file that holds information for the Netscape Web browser. |
| X Configuration Files | Which file is used here depends on what distribution of Linux is being used. This will either be the .xinitrc, .xsession, or .Xclients file. Like the .bashrc, this file actually runs automatically as a script when the X Window session is started. |
| Window Manager Files | Window managers were covered previously. Each of these Window managers have a configuration file associated with them that stores the settings for these files. |

File di configurazione

- I file di configurazione di **sistema** controllano il funzionamento dell'intero sistema operativo.
- Sono localizzati nella directory /etc
- Possono essere idealmente suddivisi in tre categorie:
 - File di configurazione dei *servizi d'avvio*
 - File di configurazione per determinate *funzioni del sistema operativo*
 - File di configurazione relativi ad *applicazioni di tipo servente* (webserver, server di posta, etc)

File di configurazione



File di configurazione dei servizi d'avvio:

- **init** è il programma usato per avviare i servizi al boot (da Ubuntu 9.04 si utilizza un nuovo software per avviare i servizi: **upstart**). Il file di configurazione per il processo di init è **/etc/inittab**.

| File Name | Description |
|------------------------------------|---|
| fstab | This configuration file specifies what partitions and devices will mount automatically. |
| X11 | This file contains information regarding how the X Window environment is supposed to be configured, including XF86Config, which was covered in previous chapters. |
| modules.conf | These files specify when and what kernel modules will be loaded. |
| passwd, shadow, group, and gshadow | These files were covered in the previous chapter but are also system function configuration files. |
| resolv.conf | This file stores information that the OS uses for the locations of name servers such as DNS servers. |

File di configurazione per determinate funzioni del sistema operativo:

- controllano il funzionamento del sistema dopo aver effettuato il boot.
- non sono script, vengono utilizzati da altri programmi.

File di configurazione



File di configurazione relativi ad applicazioni di tipo server:

- questi file controllano il comportamento di applicazioni eseguite in background.
- sono solitamente configurati in modo da poter avviare l'applicazione o modificare le sue impostazioni una volta avviata.

| Configure Files for Specific Servers | |
|--------------------------------------|--|
| File Type | Description |
| xinetd.conf | This file controls servers that run through this program like telnet or ftpd. These files respond to requests made from other servers or workstations and launch the appropriate server when needed. This method saves the systems memory and resources because the server does not need to run all the time, only when requested. |
| sendmail.cf | This is the file that controls the sendmail mail server. This is a popular e-mail server, which is used in many distributions. |
| smb.conf | This is the file that controls the Samba server. This file allows the sharing of files with Windows systems. |
| exports | This is the file that runs an NFS server that is used to share files with other UNIX and Linux systems. This type of server is configured through the /etc/exports file. |
| httpd.conf | This configuration file runs the Apache web server on a Linux system. |

File di configurazione

| File Name | Description |
|-----------|---|
| id | This represents the identification string for the runlevel process. |
| Runlevel | This specifies at which runlevel the process will run. |
| Action | This tells the init program how to handle the process. The options to include here are initdefault, which specifies the default runlevel; sysinit, which indicates the process will be run during system startup; respawn, which indicates the system will start the process whenever it quits; wait, which indicates the system will run the process and wait for it to quit; and ctrlaltdel, which tells init what process to run when the Ctrl+Alt+Del key sequence is used. |
| Process | This represents the command to be executed for this process. |

Il file `/etc/inittab` controlla il processo di init.

Formato della configurazione:
`id:runlevel:action:process`

```
# This file describes how the INIT process
# should set up the system in a certain
# run-level.
#
# Author:      Miquel van Smoorenburg,
#             <miquels@drinkel.nl.mugnet.org>
# Modified for RBS Linux by Marc Ewing
# and Donnie Barnes
#
# Default runlevel. The runlevels used by RBS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you
#    do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

# Things to run in every runlevel.
ud::once:/sbin/update

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

File di configurazione

Il file `/etc/fstab` fornisce le indicazioni per accedere alle partizioni di un disco e ai dispositivi rimovibili permettendo a un utente di conoscere i *mount point* dei vari file system. Nelle distribuzioni più recenti i dispositivi rimovibili sono totalmente gestiti e montati tramite HAL.

The /etc/fstab File

```
LABEL=/ / ext3 defaults 1 1
LABEL=/boot boot ext3 defaults 1 2
none /dev/pts devpts gid=5,mode=620 0 0
none /proc/proc defaults 0 0
none /dev/shm tmpfs defaults 0 0
/dev/hda3 swap swap defaults 0 0
/dev/cdrom /mnt/cdrom iso9660 noauto,owner,kudzu,ro 0 0
/dev/fd0 /mnt/floppy auto noauto,owner,kudzu 0 0
"/etc/fstab" 8L, 617C
```

The /etc/fstab/ File Descriptions

| Option | Description |
|-----------------|---|
| Device | This column specifies the mount device. These are device filenames that represent hard drives, floppy drives, or CD-ROMs. |
| Mount Point | This column shows the location of where the drive or partition is mounted in the directory structure relevant to the root directory. |
| Filesystem type | This represents the type of file system that the drive is formatted with. The type code that is used is the same that was used with the mount command. |
| Mount Options | This section represents the options available for how the kernel treats the file system. It is possible to specify more than one mount option here as well by separating them by commas. To learn what mount option can be used here, consult the man mount page. A few examples of the values that can be used here are as follows: <ul style="list-style-type: none">• The defaults option specifies that the default values will be used.• The users option allows regular users to mount or unmount the file system.• The noauto option specifies that the file system will not be mounted during startup. |
| dump operation | The value in this field is always either a one (1) or a zero (0). A number 1 is used to tell the dump utility to backup the partition, a 0 means that it will not. |
| fsck order | Recall from previous chapters that the fsck utility is used to check file system integrity. The value that is placed here represents if this utility is being used on the partition and in what order. A value of zero (0) indicates that no check will take place. The value of one (1) or two (2) represents the order in which fsck checks the disk or partition, with one (1) having the highest priority. The root partition should have a value of one (1), any others should have a value of two (2). |

Gestione dei runlevel



- I **runlevel** controllano quale insieme di applicazioni verrà avviato al boot del sistema.
- Tradizionalmente le configurazioni riguardanti l'ordine dei runlevel da eseguire al boot sono memorizzate nel file **/etc/inittab**.
- Il processo di **init** legge tale file e imposta il runlevel iniziale di conseguenza.
- Dopo il boot, i runlevel possono essere modificati utilizzando i comandi **init** o **telinit**.
- Poichè Ubuntu 9.04 utilizza **upstart** al posto di **init**, non è presente il file **/etc/inittab**. Tale sistema assicura comunque *backward compatibility* con **sysvinit**. Per gestire comodamente i runlevel è possibile installare **sysv-rc-conf**.

Gestione dei runlevel



- I runlevel sono numerati da 0 a 6.
- Passare al runlevel 0 è un caso speciale poichè causa l'arresto della macchina.
- Il metodo consigliato per spegnere il computer è tramite il comando **shutdown**.

Shutdown command options

| Option | Description |
|--------|---|
| -t sec | This is the time, in seconds, between the time shutdown issues the SIGTERM command and the shutdown process begins. If this parameter is not used then the default time of five seconds is used. |
| -a | Adding this parameter to the command will force the system to check the /etc/shutdown.allow file for a list of users who are allowed to issue this command. The system will only shutdown if one of those users listed in the /etc/shutdown.allow file is logged into the system. |
| -r | Using this option with force the system to reboot after shutdown. |
| -k | This option sends a message to users that the system is being shutdown but does not actually shutdown. |

shutdown [-t sec][-arkcfF] time [warning-message]

Documentazione della configurazione del sistema



E' opportuno tenere traccia delle operazioni compiute:

- Configurazione iniziale
- Installazione pacchetti
- Modifica file di configurazione
- Modifiche al filesystem
- Ricompilazioni del kernel
- Interventi hardware

Per effettuare un backup della directory contenente tutti i file di configurazione, si può utilizzare il comando **tar**:

```
mount /dev/sdb1 /mnt/backup  
tar cvfj /mnt/backup/etc.tar.bz2 /etc
```

[c] create [v] verbose [f] nome_file [i] ignora i blocchi EOF

Introduzione ai demoni



- Le funzioni che in Windows sono chiamate “servizi”, in Linux sono tipicamente identificate con il termine “**demoni**”.
- A differenza di Windows, i demoni non sono integrati nel sistema operativo, ma applicazioni distinte che svolgono determinate funzioni.
- I demoni vengono eseguiti in background senza produrre alcun output visibile. L'attività viene registrata su dei file di log.

FTPD: ftp server

HTTPD: web server

crond: esegue script a un determinato orario

syslogd: registra informazioni nei file di log di sistema

inetd: attende richieste in ingresso e le redirige al demone appropriato

Avvio, arresto e riavvio dei demoni



- Generalmente ci sono vari modi per gestire un demone su GNU/Linux:
 - Manualmente tramite script di avvio e arresto forniti dall'init system.
 - Permanentemente tramite script forniti dall'init system.
 - Tramite il superdemone xinetd.
 - Tramite script di avvio personalizzato.

Avvio, arresto e riavvio dei demoni



- Su Ubuntu 9.04 Server Edition, gli script del sistema di init sono collocati nella directory [/etc/init.d](#)
- Per eseguire questi script è necessario avviarli tramite il seguente comando:
`/etc/init.d/<script> start`
- Oltre a “start”, altre opzioni possibili sono stop, restart, reload, status, etc...
- E' bene tener presente che:
 - Il nome e la posizione degli script può variare in base alla distribuzione.
 - Gli script di startup sono scritti per funzionare su una determinata distribuzione, non è assicurata l'interoperabilità.
 - Non è sempre facile capire se uno script è stato eseguito correttamente.
 - Alcuni script supportano opzioni differenti in base alla distribuzione.

Avvio, arresto e riavvio dei demoni



- Tradizionalmente gli script di startup per [Sys V](#) sono posizionati nelle directory [/etc/rc.d/init.d](#) o [/etc/init.d](#), ma altre directory ([/etc/rc.d/rcX.d](#) o [/etc/rcX.d](#)) contengono link simbolici a questi script.
- Quando un sistema GNU/Linux viene avviato ad uno specifico runlevel, questi link simbolici, associati al runlevel, assicurano l'avvio permanente di servizi, programmi e demoni.
- I nomi dei link simbolici si presentano nella seguente forma:
 - [Kxxdaemon](#)
 - [Sxxdaemon](#)

S per "Start"

K per "Kill"

xx per l'ordine di esecuzione

Avvio, arresto e riavvio dei demoni



- I [Super-server](#) si pongono in ascolto di eventuali richieste di demoni o servizi.
- Si occupano di caricare un demone o un servizio solo dopo aver ricevuto una richiesta.
- Questo sistema permette di risparmiare memoria in quanto il demone viene avviato solo quando serve.
- Il problema è dovuto all'aumento del tempo necessario per accedere al servizio.
- I due super-server usati nelle distribuzioni GNU/Linux sono [inetd.d](#) e [xinetd.d](#)

Avvio, arresto e riavvio dei demoni



- **xinetd.conf** è il file di configurazione per **xinetd.d**
- **xinetd.conf** specifica le configurazioni necessarie ai file contenuti in **/etc/xinetd.d**

```
xinet.conf
Simple configuration file for xinetd
Some defaults, and include /etc/xinetd.d/

defaults
{
    instances                = 60
    log_type                  = SYSLOG authpriv
    log_on_success             = HOST PID
    log_on_failure             = HOST
    cps                       = 25 30
}

includedir /etc/xinetd.d
```

```
/etc/rc.d/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

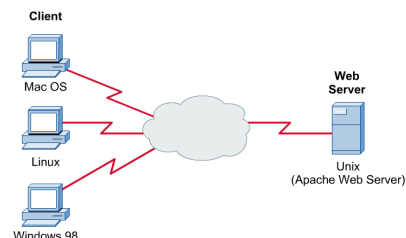
touch /var/lock/subsys/local
```

- Volendo avviare un demone tramite un proprio script, basta aggiungere l'opportuno comando in **/etc/rc.d/rc.local**
- L'unico modo per terminare i processi avviati in questo modo è quello di usare il comando **kill** o **killall** dopo averne individuato il PID

HTTP



- **Apache** è un demone HTTP per vari sistemi Unix-like (e non solo) al pari di Internet Information Services (**IIS**) per Windows.
- Sebbene la configurazione di Apache (a riga di comando) sia molto più difficile e complessa rispetto a quella di IIS, permette di controllare il demone HTTP a un livello più approfondito.
- Gli utenti possono utilizzare qualsiasi sistema operativo per accedere a un web server con Apache visto che il protocollo HTTP è indipendente dal sistema operativo utilizzato sul client.



HTTP

- Ubuntu 9.04 Server Edition fornisce di default una configurazione per Apache funzionante.
- Per testarlo basta inserire nel browser l'indirizzo IP della macchina sulla quale è stata eseguita l'installazione.

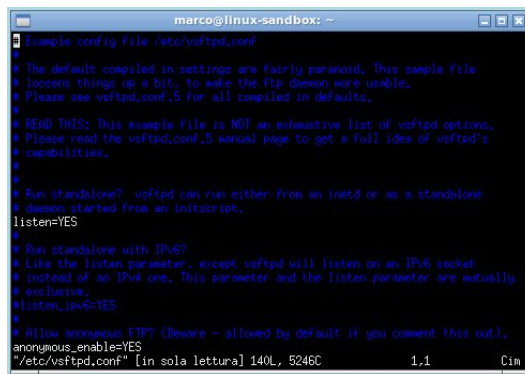


FTP

- Un server FTP estremamente sicuro e diffuso è [vsftpd](#).
- Una volta installato il demone basta avviarlo con il comando:

`sudo /etc/init.d/vsftpd start`

- Per informazioni più dettagliate:
`man vsftpd`



SSH



- **SSH** permette ad un utente di eseguire un login remoto in modo sicuro sul sistema tramite un'interfaccia a riga di comando.
- Per avviare tale servizio sul server basta eseguire:
`sudo /etc/init.d/ssh start`
- Per effettuare l'accesso, dal client:
`ssh -p porta user@host`

Server Message Block (SMB) Protocol



- **SMB** è un protocollo sviluppato per la condivisione di risorse.
- E' stato rinominato Common Internet FileSystems (**CIFS**) ma viene ancora usato per condivisione di file e stampanti.
- Tale protocollo è usato per permettere ad utenti non Linux o Unix di montare e usare file e stampanti montati su un server Linux nella rete.
- Il protocollo permette ad un client Windows di operare in una rete Linux come se si trovasse all'interno di una rete di sistemi Windows.
- L'applicazione che permette di utilizzare tale protocollo si chiama **Samba** ed è inclusa in tutte le distribuzioni recenti.

NFS (Network File System)

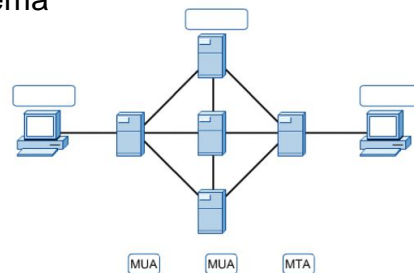


- **NFS** è un filesystem di rete che permette la condivisione di file fra vari sistemi Unix.
- La differenza principale rispetto a Samba è che NFS è stato progettato per funzionare esclusivamente su e fra sistemi Unix.
- Nel caso la rete sia composta soltanto da macchine Unix, NFS è preferibile rispetto a CIFS.

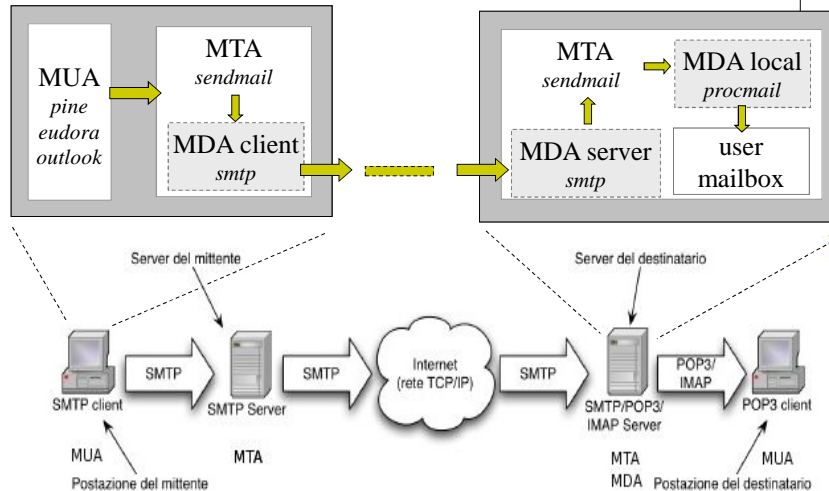
Client Mail



- Per avere un sistema di invio e ricezione mail funzionante sul server, l'amministratore di sistema deve predisporre:
 - **MUA**: Mail User Agent
 - **MTA**: Mail Transport Agent
 - **MDA**: Mail Delivery Agent
- Il client mail fornisce all'utente un'interfaccia per la lettura e la scrittura di e-mail.
- Il client mail utilizza l'MTA per trasferire le mail dal mittente alla mailbox del destinatario e viceversa.
- MTA molto diffusi sono **Postfix** (integrato in Ubuntu 9.04 Server Edition) e **Sendmail**.



Mail Client



Stampa in GNU/Linux

- Il sistema di stampa maggiormente diffuso nelle distribuzioni GNU/Linux recenti è il Common UNIX Printing System (CUPS).
- E' un sistema di stampa liberamente disponibile e altamente portabile. Per questi motivi è diventato il nuovo standard per la stampa in molte distribuzioni GNU/Linux.
- Se specificato in fase di installazione, CUPS verrà installato nel sistema.

Per avere maggiori dettagli sulla configurazione:

[man cupsd.conf](#)

```
marco@linux-sandbox: ~
cupsd.conf(5) Apple Inc. cupsd.conf(5)

NAME
    cupsd.conf - server configuration file for cups

DESCRIPTION
    The cupsd.conf file configures the CUPS scheduler, cupsd(8). It is
    normally located in the /etc/cups directory.

    Each line in the file can be a configuration directive, a blank line,
    or a comment. Comment lines start with the # character. The configura-
    tion directives are intentionally similar to those used by the popular
    Apache web server software and are described below.

DIRECTIVES
    The following directives are understood by cupsd(8). Consult the on-
    line help for detailed descriptions:

    AccessLog filename
        Defines the access log filename.

Manual page cupsd.conf(5) line 1
```

Scripts



- Le distribuzioni GNU/Linux permettono una vasta scelta di linguaggi di scripting. Il sistema più diffuso e integrato per eseguire script è tramite le funzionalità fornite dalla shell.
- Uno **shell script** è un file testuale che contiene un numero variabile di comandi scritti in successione.
- Ogni comando viene eseguito esattamente come se fosse stato inserito nel prompt.
- Tali script possono contenere logiche di programmazione come loop, condizioni, etc.



```
marco@linux-sandbox: ~  
marco@linux-sandbox:~$ cat foo.sh  
#!/bin/sh  
echo start  
for i in `seq 1 3`; do  
    echo $i foo  
done  
echo end  
marco@linux-sandbox:~$ ./foo.sh  
start  
1 foo  
2 foo  
3 foo  
end  
marco@linux-sandbox:~$
```