



# **Corso base GNU/Linux 2014**

## **Latina Linux Group**

**Sito web: [www.llg.it](http://www.llg.it)**

**Mailing list: <http://lists.linux.it/listinfo/latina>**



# Obiettivi di questo incontro

- ♦ Conoscere cosa accade quando si accende il PC, come avviene l'avvio del sistema operativo
- ♦ Conoscere la struttura del filesystem di GNU/Linux
- ♦ Imparare a gestire in maniera basilare permessi e file
- ♦ Conoscere le nozioni base per partizionare un disco



# Il Bootstrap: la fase di avvio

Da Wikipedia:

“L'insieme dei processi che vengono eseguiti da un computer durante la fase di avvio, in particolare dall'accensione fino al completo caricamento in memoria primaria del kernel del sistema operativo a partire dalla memoria secondaria.”

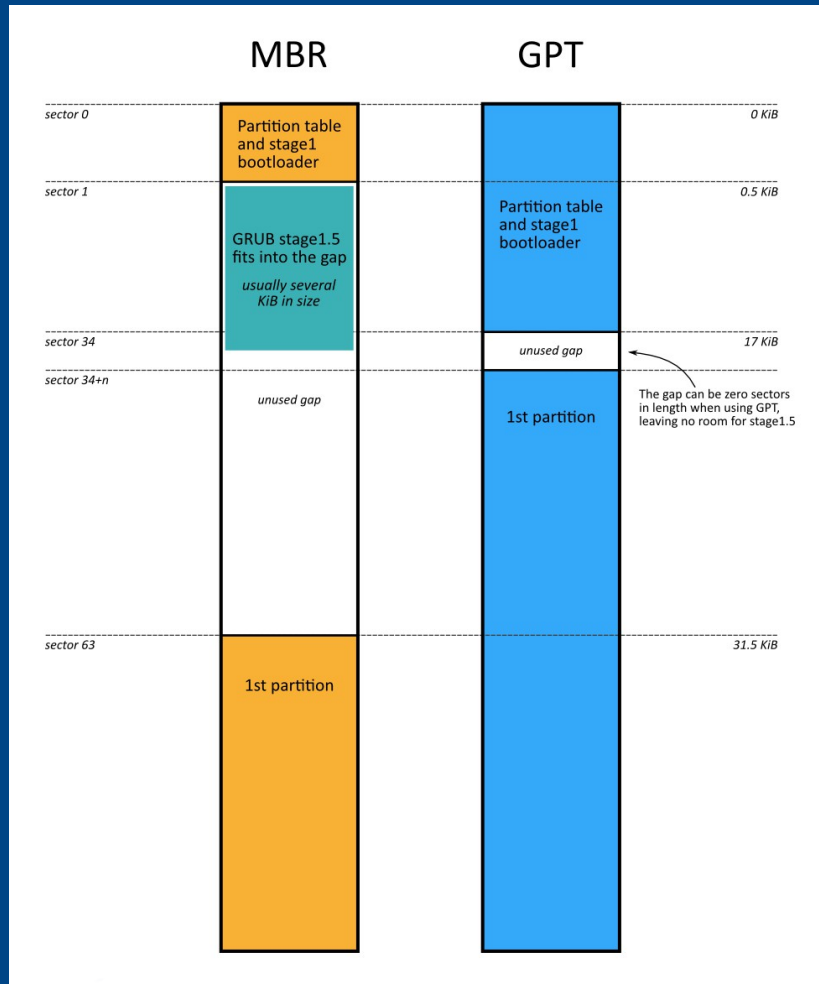


# L'avvio del Sistema Operativo

1. Accensione/Riavvio (BIOS-EFI/UEFI)
2. Bootloader, MBR (Stage 1)
3. Bootmanager, LiLo/GRUB (Stage 2)
4. Caricamento del Kernel Linux
5. Init (Processo ID = 1)
6. Run Level (User Space)



# MBR e GPT: due modi di partizionare il disco



MBR: Master Boot Record

Risiede nei primi settori del disco

GPT: GUID Partition Table

E' ridondato (copiato alla fine del disco) e fornisce un MBR "compatibile" per i software non aggiornati



# Differenze principali tra MBR e GPT

Funzionalità	MBR	GPT
Supporto ai sistemi operativi meno recenti (fino a Windows XP x64)	✓	✗
Supporto a partizioni maggiori di 2 TB	✗	✓
Supporto avvio sistemi operativi x86	✓	✗
Supporto avvio sistemi operativi x64	✓	✓
Supporto disco dati	✓	✓
Massimo numero di partizioni primarie	4	Fino a 128

Queste limitazioni affliggono per lo più i sistemi operativi Microsoft, GNU/Linux e Mac OSX NON hanno problemi sia riguardo l'installazione del sistema operativo che la gestione di file su dischi partizionati in GPT.



# Kernel

Il Kernel rappresenta il cuore di ogni sistema operativo:

- ♦ Esegue il programma */sbin/init*, che, essendo il primo ad essere avviato, ha PID 1
- ♦ Avvia *initrd*, un RAM disk iniziale che contiene un sistema minimale che consente il montaggio del file system “reale” grazie ad un set di driver che permettono il riconoscimento dell'hardware



# Init

File di configurazione: */etc/inittab*

Esistono 6 Run Level:

0 – Halt (Arresto)

1 – Modalità Singolo utente

2 – Modalità Multiutente senza NFS

3 – Modalità Multiutente

4 – \*Inutilizzato\*

5 – X11 (Interfaccia grafica)

6 – Riavvio





# Run Level

In fase di avvio Linux eseguirà i programmi indicati nei file di configurazione di ogni Run Level:

- ♦ Run level 0 – /etc/rc.d/rc0.d/
- ♦ Run level 1 – /etc/rc.d/rc1.d/
- ♦ Run level 2 – /etc/rc.d/rc2.d/
- ♦ Run level 3 – /etc/rc.d/rc3.d/
- ♦ Run level 4 – /etc/rc.d/rc4.d/
- ♦ Run level 5 – /etc/rc.d/rc5.d/
- ♦ Run level 6 – /etc/rc.d/rc6.d/



## Run Level #2

In ogni directory */etc/rc.d/rc\*.d/* si trovano degli script (generalmente dei link) di tipo *S* e *K*:

- ♦ Gli script che iniziano con *S* vengono eseguiti durante l'avvio del sistema (*StartUp*)
- ♦ Gli script che iniziano con *K* vengono eseguiti durante l'arresto del sistema (*Kill*)
- ♦ I numeri ordinali che seguono il tipo *S* e *K* indicano la sequenza in cui gli script devono essere avviati o terminati



## Il File System (FS)

Il File System (sistema di organizzazione dei file) indica il modo in cui i file sono posizionati e organizzati sul disco (HD, USB, CD, DVD, RAM, ...).

Esistono diversi tipi di FS (FAT16, FAT32, NTFS, EXT4, XFS, JFS, HFS, HFS+, HPFS, ...) per sistemi operativi diversi e per esigenze diverse.



# Ext4

## (Fourth Extended FileSystem)

- E' il FS predefinito di GNU/Linux, i file sono organizzati in un albero che parte dalla *radice* (root, “/”) e in cui ci sono diverse directory che contengono file “*speciali*”
- E' l'evoluzione di ext3 di cui migliora le prestazioni e le caratteristiche, fu rilasciato il 25/12/2008 con il kernel 2.6.28
- E' un filesystem “*journalled*”, ogni attività di scrittura su disco viene interpretata come una transazione:
  1. Si scrivono in un log le operazioni da eseguire sul/i file
  2. Si eseguono le operazioni
  3. Si scrivono sul log i dettagli delle operazioni eseguite
- La dimensione massima del singolo file può essere di 16 TB



# La struttura del FS

- ♦ */bin*: applicazioni binarie importanti (comandi di sistema)
- ♦ */boot*: contiene i file necessari all'avvio (boot) del PC, ad esempio, il kernel
- ♦ */dev*: file dei device
- ♦ */etc*: file di configurazione, script di avvio
- ♦ */home*: directory personali degli utenti (dove, normalmente, vengono memorizzate tutti i dati dell'utente)
- ♦ */lib*: librerie di sistema
- ♦ */media*: dispositivi rimovibili montati (USB, CD, DVD, schede di memoria, ...)
- ♦ */mnt*: (altri) filesystem montati
- ♦ */opt*: directory per le applicazioni opzionali



# La struttura del FS #2

- ♦ */proc*: directory speciale il cui contenuto è dinamico: i file contenuti rappresentano lo “*stato*” del sistema e dei processi in esecuzione
- ♦ */root*: home directory dell'utente root
- ♦ */sbin*: file binari di sistema importanti
- ♦ */sys*: contiene informazioni riguardo il sistema
- ♦ */tmp*: file temporanei, viene ripulita ad ogni avvio
- ♦ */usr*: file e applicazioni che sono disponibili per tutti gli utenti di sistema
- ♦ */var*: file variabili, di solito log e database

Di solito un programma installato dall'utente copia l'eseguibile in */usr/bin*, le librerie in */usr/lib* e gli eventuali file di configurazione in */etc*



# Utenti, Gruppi, permessi

- Il FileSystem di GNU/Linux è gerarchico, basato su standard POSIX e gestito tramite ACL (Access Control List):

Ogni utente appartiene a uno o più gruppi per cui sono definiti dei permessi di accesso (lettura, scrittura, esecuzione)

```
andrea@VUbuntu13:~/test_dir$ ll
totale 8
drwxr-xr-x  2 andrea andrea 4096 apr 15 00:23 ./
drwxr-xr-x 22 andrea andrea 4096 apr 15 00:20 ../
-rw-r--r--  1 andrea andrea   0 apr 15 00:20 file01.txt
-rwxr-xr-x  1 andrea andrea   0 apr 15 00:21 file02.testo*
-rwxrwxrwx  1 andrea andrea   0 apr 15 00:21 file03.odt*
-rw-r--r--  1 andrea andrea   0 apr 15 00:22 file04.odp
-rwxr--r--  1 andrea andrea   0 apr 15 00:23 file05.ods*
andrea@VUbuntu13:~/test_dir$
```



# Tipi di permesso

Valore	Tipo di permesso
0	Nessun permesso
1	Esecuzione (x)
2	Scrittura (w)
3	Esecuzione + Scrittura (wx)
4	Lettura (r)
5	Lettura + Esecuzione (rx)
6	Lettura + Scrittura (rw)
7	Lettura + Scrittura + Esecuzione (rwx)





# Partizionamento del disco

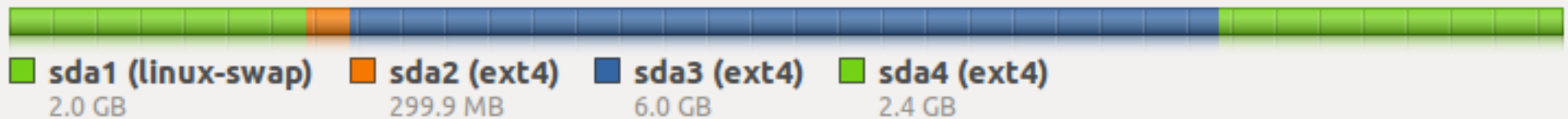
Per poter installare GNU/Linux è necessario, oltre che selezionare il disco, decidere come suddividere lo spazio che ospiterà il file del sistema operativo e i dati.

L'installazione di base prevede la presenza di almeno due partizioni:

- ♦ Una di *swap*, che contiene dati temporanei che non possono essere trattieneuti in RAM
- ♦ Una di *root (/)*, che contiene il sistema operativo, i programmi e i dati



# Esempio di partizionamento con quattro partizioni



Device	Tipo	Punto di mount	Formattare?	Dimensione	Utilizzato	Sistema
/dev/sda						
/dev/sda1	swap		<input type="checkbox"/>	2046 MB	sconosciuto	
/dev/sda2	ext4	/boot	<input checked="" type="checkbox"/>	299 MB	sconosciuto	
/dev/sda3	ext4	/	<input checked="" type="checkbox"/>	5999 MB	sconosciuto	
/dev/sda4	ext4	/home	<input checked="" type="checkbox"/>	2388 MB	sconosciuto	

+ - Change...

Nuova tabella partizioni...

Ripristina

Device per l'installazione del boot loader:

/dev/sda ATA VBOX HARDDISK (10.7 GB)



# La riga di comando

Alcuni esempi di comandi:

- ♦ man, ls, cp, mv
- ♦ cat, more, less, tail, echo
- ♦ grep
- ♦ Operatori (<, >, !, |, ...)
- ♦ ps, kill, pkill, top
- ♦ chmod, chown
- ♦ ...



# ABS: Advanced Bash Scripting (Guide)

Un buon punto di partenza per imparare l'arte dello scripting:

<http://www.tldp.org/LDP/abs/html/>

Tanta voglia, passione, dedizione, pratica e tempo faranno il resto!!



# Domande?

