

Boot sequence

BIOS [16bit] (exec POST power on self test) and → loads “STAGE 1 Boot loader” from MBR (512 byte)→ “loads STAGE 2” boot loader (loads OS)

il boot loader può caricare il kernel con un initrd:

Initial RAM disk

L' initial RAM disk o *initrd* o *initramfs*, è un file che contiene moduli kernel caricabili. solitamente è un file compresso e viene caricato in ram dal boot loader. il kernel vi accede come se fosse un filesystem montato.

su alcuni sistemi come fedora 12 può essere creato con una utility chiamata **dracut**. possiamo vederne il contenuto con il comando **lsinitrd**

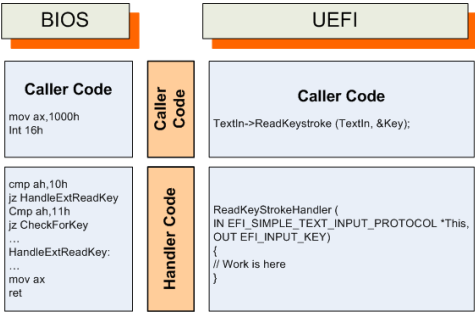
Chain loading (un boot loader può caricare un altro boot loader) es: grub → windows boot loader → windows

EFI/UEFI [32/64 bit]

- * EFI è di intel (abbandonato nel 2005) per contribuire allo sviluppo di UEFI
- * UEFI (unified Extensible Firmware Interface)

Uefi può caricare sistemi UEFI e Legacy

Uefi è un firmware complesso, richiede una partizione UEFI su disco per fare il boot, e ha supporto ai driver, una shell e può leggere il contenuto delle partizioni.



Boot loader (lilo, GRUB legacy; elilo, GRUB2 -supports uefi)

init

quando il kernel viene caricato, normalmente avvia */sbin/init*. questo processo è il primo ad essere avviato e ha PID 1.

ha il compito di avviare tutti gli altri processi e rimane attivo fino allo spegnimento

su sistemi che fanno uso del tradizionale System V init, o su sistemi più nuovi che usano l'inizializzazione di Upstart, il processo sarà chiamato init nei nuovi sistemi con systemd invece, */sbin/init* è un collegamento simbolico a */lib/systemd/systemd*.

sui sistemi in cui il processo di init è upstart, abbiamo il concetto di jobs.

mentre nei sistemi con systemd si utilizzano le units per definire cosa viene eseguito

esempi di units sono: *.service .socket .device .mount .automount .swap .target .path .timer .snapshot .slice .scope*

se il sistema carica il kernel ma fallisce nell'avviare *init*, possiamo provare a ripristinarlo avviando un programma differente all'avvio. Ad esempio, specificando *init=/bin/sh* avvia il sistema in una shell *sh* con privilegi di root, dal quale possiamo essere in grado di riparare il sistema

runlevel

dal boot loader, posso specificare tramite un parametro in quale runlevel fare il boot

Run Level	Name	Description
0	<i>Halt</i>	Shuts down all services when the system will not be rebooted.
1	<i>Single User</i>	Used for system maintenance. No Networking capabilities.
2	<i>MultiUser</i> <i>No Network Support</i>	Used for maintenance and system testing.
3	<i>MultiUser</i> <i>Network Support</i>	Non-Graphical Text Mode operations for server systems.
4	-	Custom Mode, used by SysAdmin
5	<i>Graphical</i> <i>X11</i>	Graphical login with same usability of Run Level 3.
6	<i>Reboot</i>	Shuts down all services when the system is being rebooted.