

Administración de Servidores

Arranque del Sistema

Recuperación del sistema

Creado por

Iñaki Fernández de Viana y González

Huelva, octubre 2020

Sobre Nosotros

Iñaki Fernández de Viana y González



Despacho 128. Escuela Técnica
Superior de Ingeniería



Dpto. De Tecnologías de la Información



i.fviana@dti.uhu.es



+34 959217378



Objetivos

- **Peso:** 4
- **Descripción:** Candidates should be able to properly manipulate a Linux system during both the boot process and during recovery mode. This objective includes using both the init utility and init-related kernel options. Candidates should be able to determine the cause of errors in loading and usage of bootloaders. GRUB version 2 and GRUB Legacy are the bootloaders of interest. Both BIOS and UEFI systems are covered.

Objetivos (II)

- **Áreas clave de conocimiento:** BIOS and UEFI. NVMe booting. GRUB version 2 and Legacy. grub shell ; boot loader start and hand off to kernel; kernel loading; hardware initialisation and setup; daemon/service initialisation and setup; Know the different boot loader install locations on a hard disk or removable device. Overwrite standard boot loader options and using boot loader shells. Use systemd rescue and emergency modes.

Objetivos (III)

- **Términos y utilidades:** mount. Fsck. inittab, telinit and init with SysV init. The contents of /boot/, /boot/grub/ and /boot/efi/. EFI System Partition (ESP). GRUB. grub-install. Efibootmgr. UEFI shell. initrd, initramfs. Master boot record. systemctl.

Índice

1. Proceso de Arranque
2. GRUB
3. GRUB2
4. Systemd y sus modos de rescate

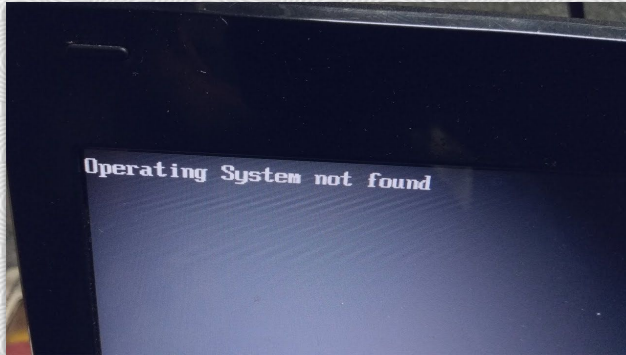
Proceso de arranque



1

Introducción

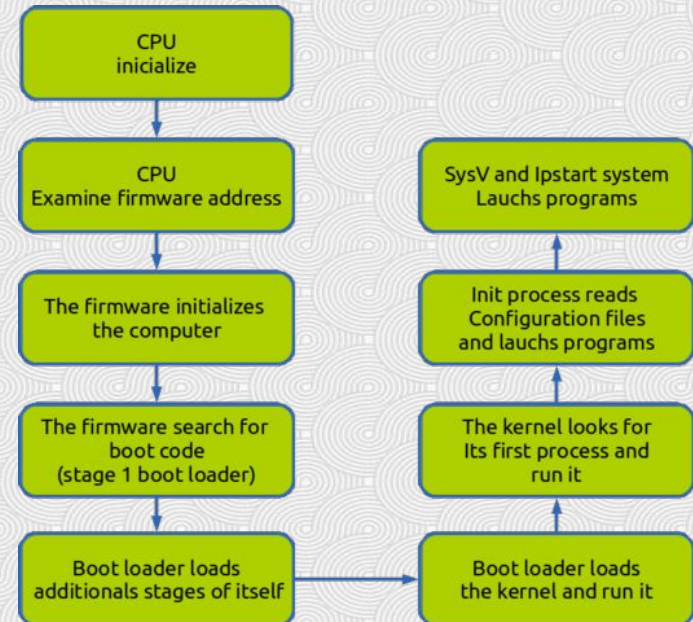
- ★ Nada es más frustrante que no poder arrancar un ordenador que antes, lo hacía
- ★ Estos problemas son de difícil solución, ya que no solemos disponer de los recursos que consultamos para solucionar cualquier otro tipo de problema



```
[ 0.552962] CPU: 3 PID: 1 Conn: swapper/0 Not tainted 4.8.0-44-generic #47~16
.04.1 Ubuntu
[ 0.553012] Hardware name: TOSHIBA Satellite C640/Portable PC, BIOS 2.10 11/0
9/2011
[ 0.553060] 0000000000000000 0000000000000000 ffff9575f4473df0 ffffffff8a2e
073
[ 0.553200] ffff9575f3ae5000 ffffffff92725a0 ffff9575f4473e70 ffffffff8b79e
5ad
[ 0.553341] ffff957500000010 ffff9575f4473e80 ffff9575f4473e20 0000000000000000
541
[ 0.553482] Call Trace:
[ 0.553519] [] dump_stack+0x63/0x90
[ 0.553562] [] panic+0xe4/0x226
[ 0.553606] [] mount_block_root+0x1fb/0x2c2
[ 0.553647] [] mount_root+0x33/0x35
[ 0.553680] [] prepare_namespace+0x13a/0x18f
[ 0.553731] [] kernel_init_freeable+0x1ee/0x217
[ 0.553775] [] kernel_init+0xe/0x100
[ 0.553817] [] ret_from_fork+0x1f/0x40
[ 0.553850] [] ? rest_init+0x80/0x80
[ 0.553932] Kernel Offset: 0x37600000 from 0xffffffff00000000 (relocation ran
ge: 0xffffffff00000000-0xffffffff00000000)
[ 0.553991] ---[ end Kernel panic - not syncing: VFS: Unable to mount root fs
on unknown-block(0,0)
-
```


Introducción

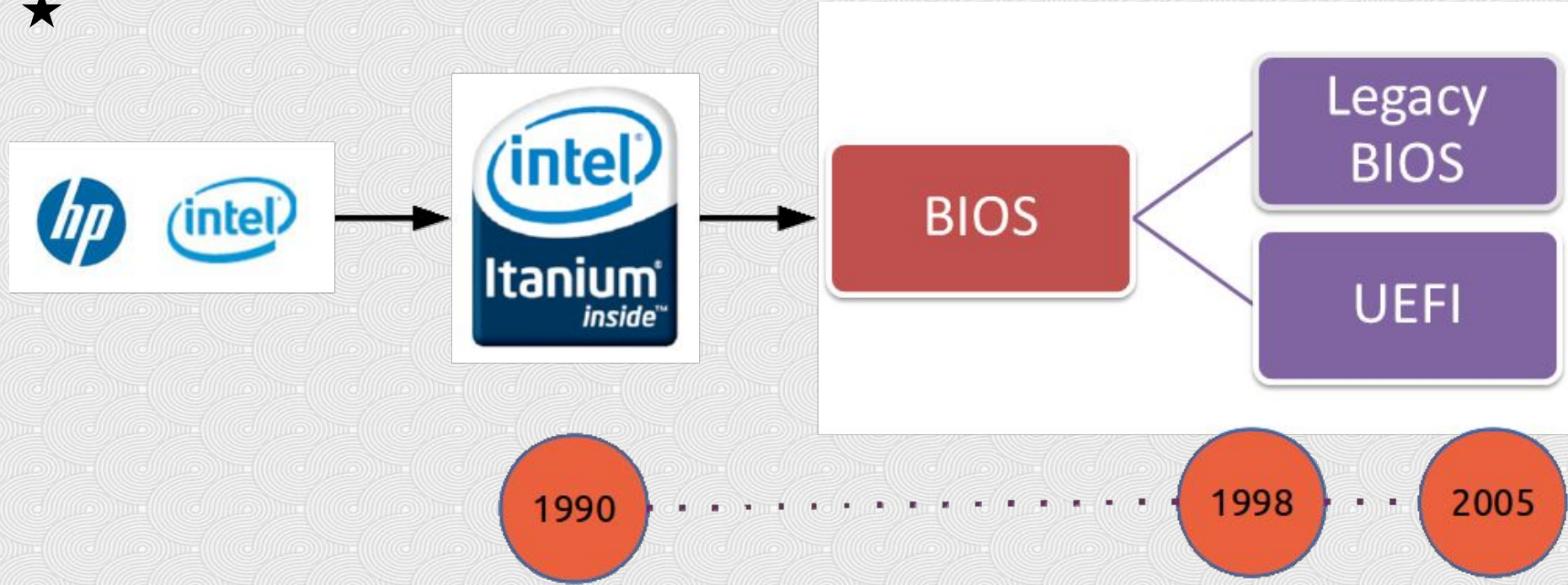
- ★ Para solucionarlos, es primordial conocer el arranque de un ordenador.
- ★ Fundamentalmente pasa por cuatro fases:
 - Fase de la BIOS/UEFI
 - Fase del bootloader
 - Fase del kernel
 - Fase del postkernel



Fase de la BIOS/UEFI

- ★ Tras encender el ordenador, un software es el encargado de iniciar el proceso de arranque.
- ★ Este software está incluido en el hardware y puede ser:
 - La **BIOS** (*Basic Input Output System*): en un firmware usando desde hace más de 30 años. Tiene muchas restricciones relacionadas con dimensiones y tipos de dispositivo.
 - **UEFI** (*Unified Extensible Firmware Interface*): es un software, no un firmware, que solventa las restricciones que presentaba la BIOS.

Fase de la BIOS/UEFI



Fase de la BIOS/UEFI

- ★ Tanto las placas base antiguas como las basadas en UEFI tienen BIOS ROMs, que contienen el firmware necesario para hacer test POST antes de cargar código de terceros en memoria y saltar hacia él.
- ★ La diferencia entre un legacy BIOS y UEFI firmware son:
 - Where they find that code,
 - How they prepare the system before jumping to it,
 - What convenience functions they provide for the code to call while running.

Inicialización de la Plataforma

```
Award Modular BIOS v6.00PG
Copyright (C) 1984-2010, Award Software, Inc.

P67A-UD5 F3

Main Processor : Intel(R) Core(TM) i5-2500K CPU 4.84GHz(101x48)
<CPUID:000206A7 Patch ID:00000009>
Memory Testing : 2032MB OK

Memory Frequency 1804MHz
IDE Channel 0 Master : SAMSUNG SP2504C UT100-33
IDE Channel 1 Master : HL-DT-ST DVDROM GH22LS50 TL02

Detecting IDE drives ...
Detecting IDE drives ...

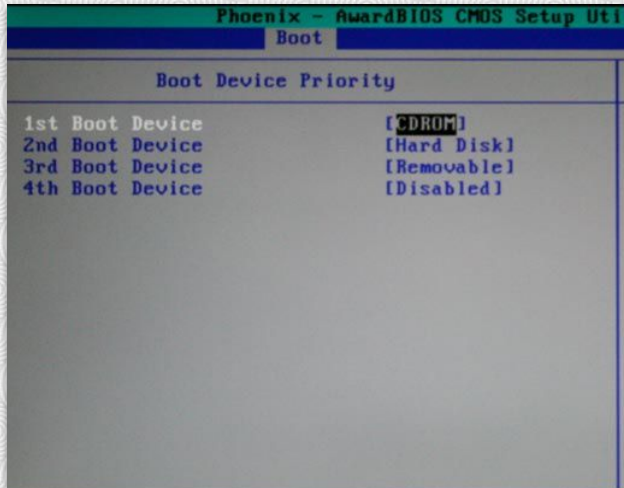
<DEL>:BIOS Setup <F9>:ExpressRecovery2 <F12>:Boot Menu <End>:Qflash
11/25/2010-P67-7A89UC00C-00
```


Inicialización de la Plataforma

- ★ En **Legacy BIOS**, realiza la inicialización de todos los dispositivos (memory controller configuration, PCI bus configuration and BAR mapping, graphics card initialization, etc.), y se lleva al sistema a modo de dirección real para mantener compatibilidades con sistemas antiguos.
- ★ En **UEFI** se realizan los mismos pasos pero, además, se prepara un modo protegido con segmentación y un modo larga con paginación para sistemas de 64 bits.
- ★ En el caso de **Legacy BIOS** es el bootloader el que se encarga de estas últimas acciones.

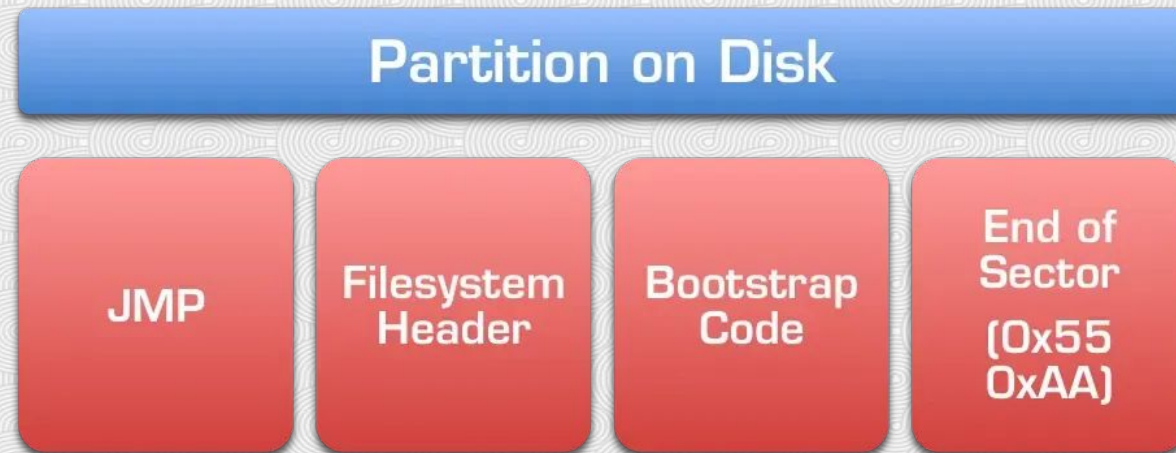
Mecanismo de arranque

- ★ En **Legacy BIOS** el MBR (primeros 512 bytes) del dispositivo marcado como de arranque se carga en la posición de memoria 7C00 y se le pasa el control al **Bootstrap Code**.

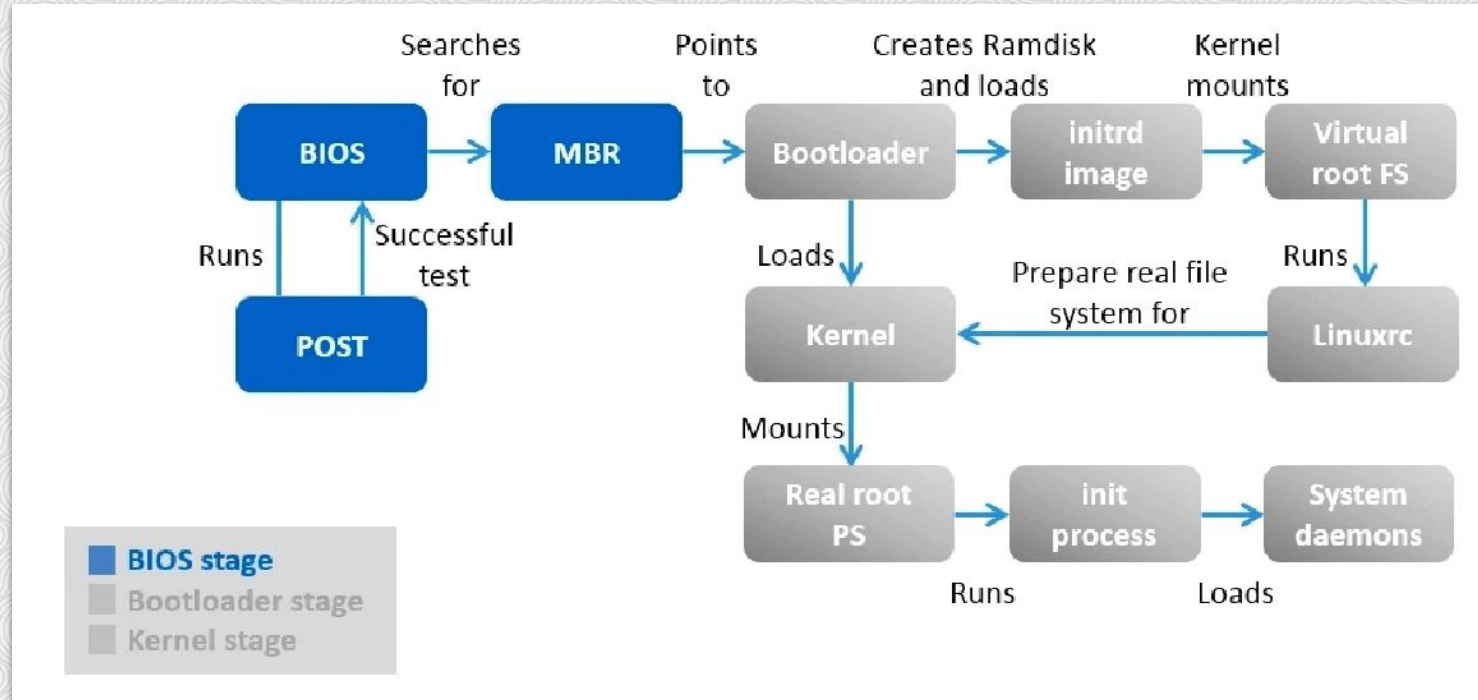


Mecanismo de arranque

- ★ El **Bootstrap Code** busca la partición activa en la tabla de particiones, y carga su primer sector en memoria e insta a la CPU para que ejecute el código contenido en dicho sector.



Mecanismo de arranque

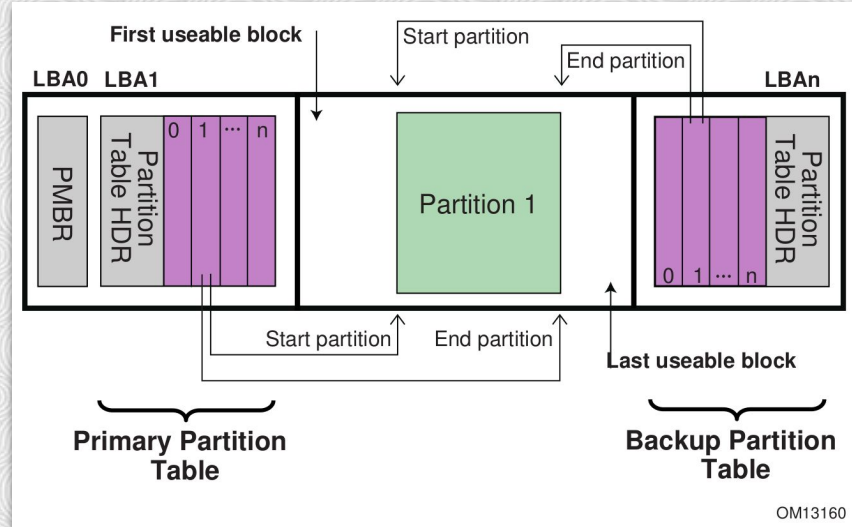


Mecanismo de arranque

- ★ En **UEFI** una **aplicación UEFI** de tamaño arbitrario (un fichero ejecutable) se carga en una posición de memoria seleccionada en tiempo de ejecución.
- ★ Este fichero se encuentra en la partición denominada **Extensible Firmware Interface (ESP)** contenida en un dispositivo con tabla de particiones de tipo **GPT**.
- ★ La CPU empieza a ejecutar la aplicación que puede devolver el control a la UEFI en cuyo caso buscará otro dispositivo de arranque.

Mecanismo de arranque

- ★ La tabla de particiones **GUID (GPT)** es un estándar para la colocación de la tabla de particiones en un disco duro físico definida en UEFI.
- ★ Sustituye al Master Boot Record (MBR) usado con el BIOS.



Mecanismo de arranque

- ★ Una **Extensible Firmware Interface (ESP)** contiene
 - Los **bootloaders** o las imágenes de los núcleos para todos los sistemas operativos instalados
 - Los **ficheros de los drivers** de los dispositivos presentes en el sistema.
 - **Programas de utilidades** del sistema que se ejecutarán antes de cargar el sistema operativo.
 - **Ficheros con datos** (logs)
- ★ El sistema de ficheros contenido en la ESP está basado en una FAT.

Mecanismo de arranque

- ★ Podemos consultar los contenidos de la ESP usada en el proceso de arranque consultando `/boot/efi`.
- ★ Su contenido será similar a:

```
> ls /boot/efi/  
EFI  mach_kernel  System
```

- ★ Por defecto la **aplicación UEFI** que se ejecutará será `\EFI\BOOT\BOOTX64.EFI` pero en el directorio EFI tenemos las aplicaciones proporcionadas por distintos fabricantes:

```
> ls /boot/efi/EFI  
BOOT  fedora  Microsoft
```

Mecanismo de arranque

- ★ Podemos usar el comando `efibootmgr` para interactuar con el gestor de arranque de la UEFI
- ★ La opción `-v` nos permitirá consultar las aplicaciones y orden:

```
> efibootmgr -v
BootCurrent: 0004
Timeout: 1 seconds
BootOrder: 0004, 0000, 000E,0013,0001,001D
Boot0000* Windows Boot Manager
HD(1,GPT,65db246a-f46f-4213-ad38-2ddd173c0125,0x800,0x4a800)/File(\EFI\MICROSOFT\BOOT\BOOTMGFW.EFI)WINDOWS.....x...B.C.D.O.B.J.E.C.T.=.{.9.d.e.a.8.6.2.c.-.5.c.d.d.-.4
.e.7.0.-.a.c.c.1.-.f.3.2.b.3.4.4.d.4.7.9.5.}...,.....
Boot0001* UEFI OS
HD(1,GPT,65db246a-f46f-4213-ad38-2ddd173c0125,0x800,0x4a800)/File(\EFI\BOOT\BOOTX64.EFI)..BO
```

Mecanismo de arranque

★ La opción **-v** nos permitirá consultar las aplicaciones y orden:

```
Boot0004* Fedora
HD(1,GPT,65db246a-f46f-4213-ad38-2ddd173c0125,0x800,0x4a800)/File(\EFI\FEDORA\SHIMX64.
EFI)
Boot000E* ADATA SX8200PNP   BBS(HD,,0x0)..BO
Boot001C* UEFI OS
HD(1,GPT,65db246a-f46f-4213-ad38-2ddd173c0125,0x800,0x4a800)/File(\EFI\BOOT\BOOTX64.EF
I)..BO
Boot001D* Fedora
HD(1,GPT,65db246a-f46f-4213-ad38-2ddd173c0125,0x800,0x4a800)/File(\EFI\FEDORA\SHIM.EFI
)..BO
```


Mecanismo de arranque

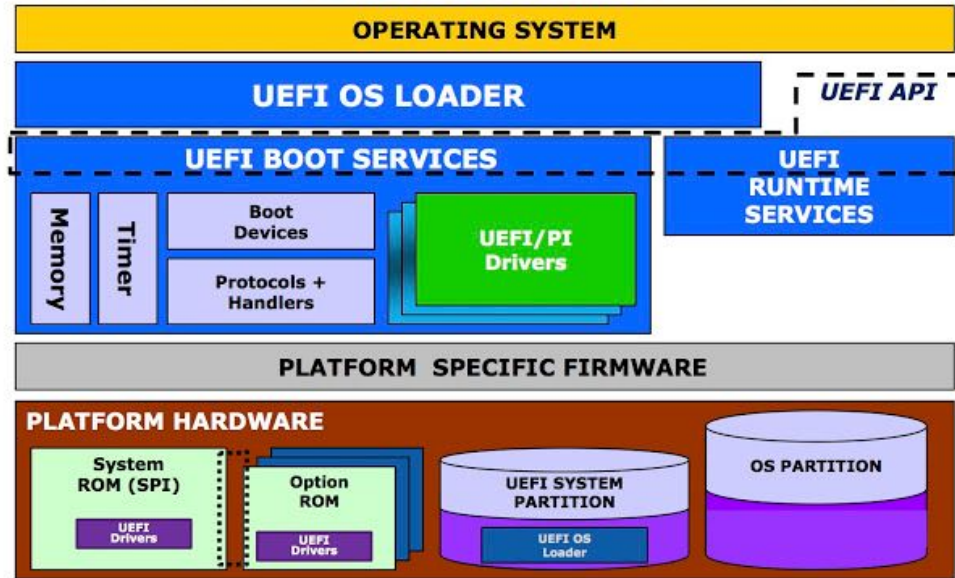
- ★ En ocasiones podemos encontrar aplicaciones UEFI denominadas shellx64.efi contenidas en `\EFI\` y que nos permitirá lanzar una UEFI shell.
- ★ Una UEFI Shell es una shell/terminal que nos permitirá lanzar aplicaciones EFI, obtener información del sistema o del propio firmware, ejecutar aplicaciones de particionado, modificar variables del boot manager, cargar driver UEFI, etc.

Mecanismo de arranque

```
UEFI Interactive Shell v2.1
EDK II
UEFI v2.40 (EDK II, 0x00010000)
Mapping table
FS0: Alias(s):HD15a0a1::BLK2:
    PciRoot(0x0)/Pci(0xD,0x0)/Sata(0x0,0x0,0x0)/HD(1,GPT,19C72019-C9C1-439
C-B9F3-E0E00469412A,0x800,0x100000)
BLK0: Alias(s):
    PciRoot(0x0)/Pci(0x1,0x1)/Ata(0x0)
BLK1: Alias(s):
    PciRoot(0x0)/Pci(0xD,0x0)/Sata(0x0,0x0,0x0)
BLK3: Alias(s):
    PciRoot(0x0)/Pci(0xD,0x0)/Sata(0x0,0x0,0x0)/HD(2,GPT,26984747-57FD-457
7-BCF0-E5E28F7F8036,0x100000,0x4D00000)
BLK4: Alias(s):
    PciRoot(0x0)/Pci(0xD,0x0)/Sata(0x0,0x0,0x0)/HD(3,GPT,6395C8EF-2EF9-419
6-A071-D4ABC465AE11,0x4E00000,0x1FF000)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> _
```

Mecanismo de arranque

Overview of the UEFI Boot Process

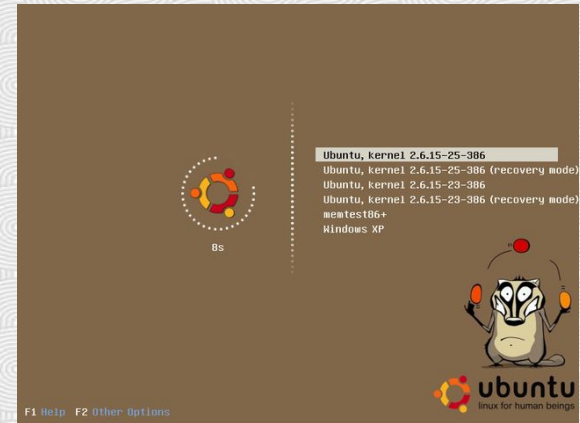


4

IDF2012
INTEL DEVELOPER FORUM

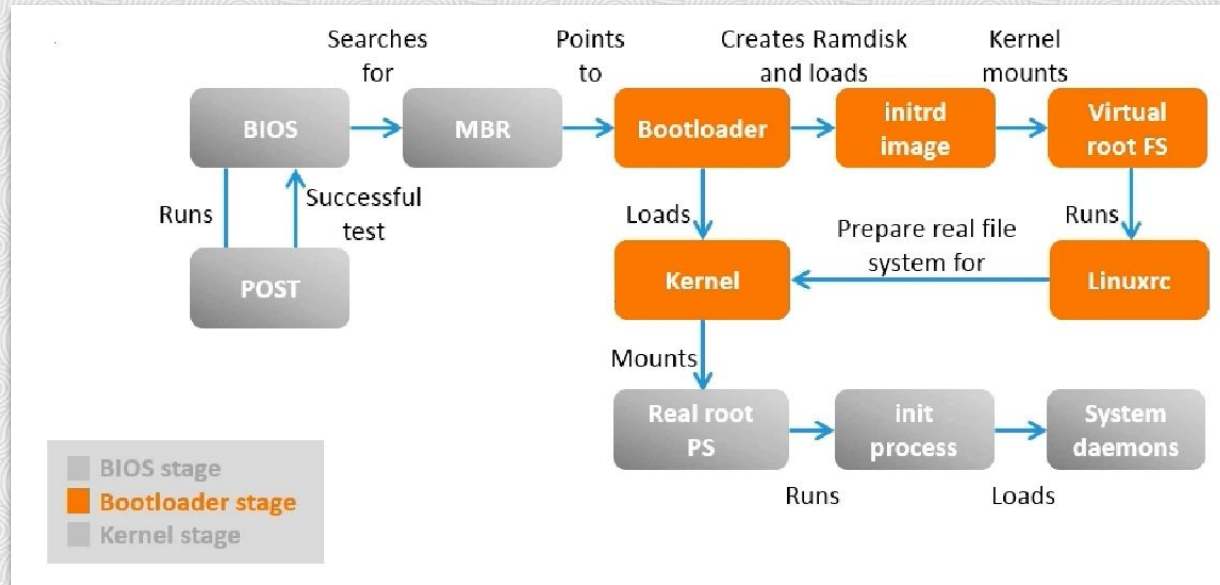
Fase bootloader

- ★ El bootloader es el software proporcionado por el sistema operativo encargado de leer el núcleo y comenzar el proceso de arranque del mismo.
- ★ Algunos de los bootloader más conocidos son:
 - LILO
 - Grub
 - Grub2

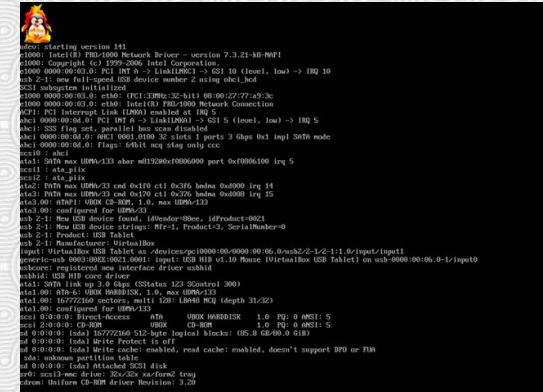


Fase bootloader (II)

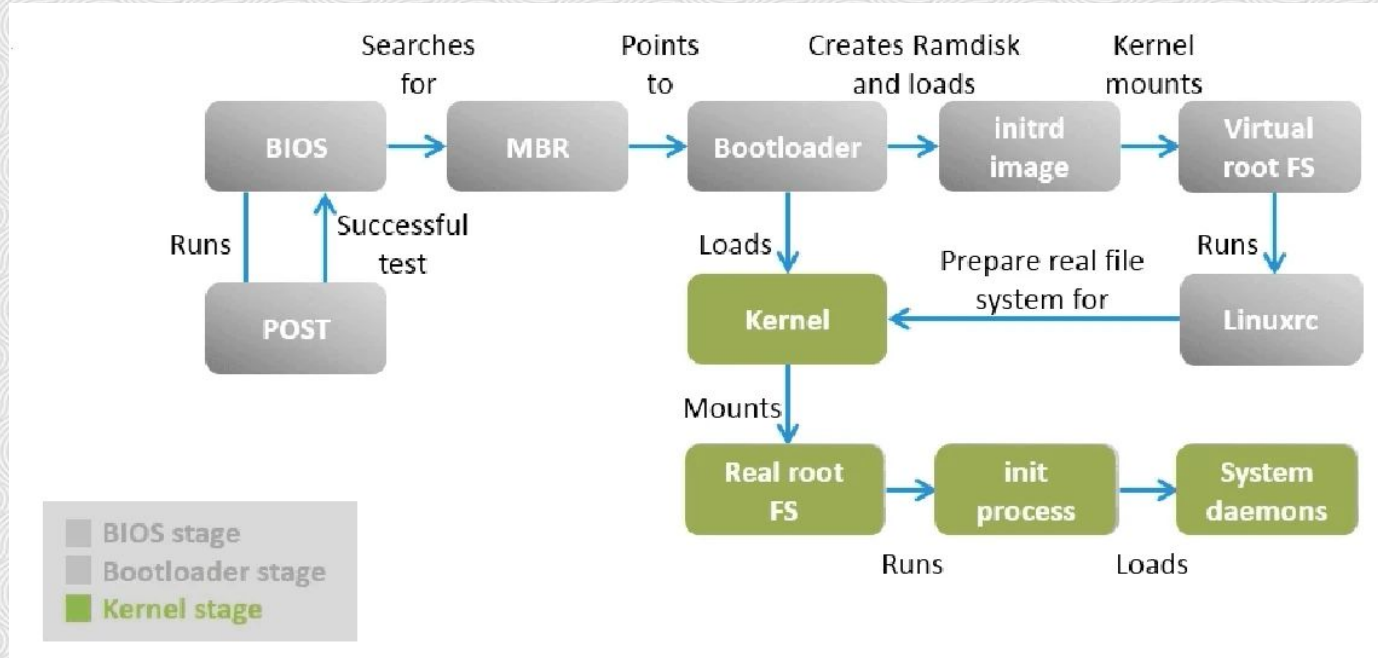
- ★ El bootloader carga la ramdisk y el núcleo y crea el sistema de ficheros raíz (virtual)



- ★ Configura el sistema asignando espacio de direcciones
- ★ Prueba el hardware y realiza diversas operaciones de configuración
- ★ Descomprime el RAM disk.
- ★ Inicializa los meta dispositivos
- ★ Monta el sistema de ficheros raíz en modo solo lectura
- ★ Da comienzo a las etapas posteriores




Fase del núcleo (II)



Fase del post-núcleo

- ★ Lleva al sistema al nivel de arranque indicado tal y como se vio en el tema pasado.



```
Configuring ISA PNP
Setting system time from the hardware clock (localtime).
Using /etc/random-seed to initialize /dev/urandom.
Initializing basic system settings ...
Updating shared libraries
Setting hostname: engpc23.murdoch.edu.au
INIT: Entering runlevel: 4
rc.M ==> Going multiuser...
Starting system logger ... [ OK ]
Initialising advanced hardware
Setting up modules ... [ OK ]
Initialising network
Setting up localhost ... [ OK ]
Setting up inet1 ... [ OK ]
Setting up route ... [ OK ]
Setting up fancy console and GUI
Loading fc-cache ... [ OK ]
rc.vlinit ==> Going to runlevel 4
Starting services of runlevel 4
Starting dnsmasq ... [ OK ]
==> rc.X Going to multiuser GUI mode ...
XFree86 Display Manager
Framebuffer /dev/fb0 is 307200 bytes.
Grabbing 640x480 ...
```

GRUB

2

Introducción

- ★ **Grub** (*GRand Unified Bootloader*) surgió como alternativa de otros gestores de arranque como fue LILO.
- ★ Actualmente la última revisión es la 0.97
- ★ No sigue un desarrollo activo ya que se apuesta por GRUB 2
- ★ Distribuciones como CentOS o RedHat lo siguen usando como boot loader por defecto.

Configuración (I)

- ★ **Grub** se configura mediante los ficheros `/boot/grub/menu.lst` o `/boot/grub/grub.conf`.
- ★ Estos ficheros son leídos durante el proceso de arranque.
- ★ Este fichero contiene un conjunto de **opciones globales**

```
default=0
timeout=15
splashimage=(hd0,0)/grub/bootimage.xpm.gz
hiddenmenu
```

Configuración (II)

★ ... y opción por núcleo

```
title Fedora (2.6.32)
    root (hd0,0)
    kernel /vmlinuz-2.6.32 ro root=/dev/hda5 mem=2048M
    initrd /initrd-2.6.32

title Debian (2.6.36-experimental)
    root (hd0,0)
    kernel (hd0,0)/bzImage-2.6.36-experimental ro root=/dev/hda6
```

Opciones globales

➤ Opción	Descripción
<code>default</code>	Núcleo a arrancar por defecto (la numeración empieza por 0)
<code>timeout</code>	Segundos de espera antes de arrancar el núcleo por defecto
<code>splashimage</code>	Imagen de fondo que se mostrará en el proceso de arranque (path relativo a la partición de arranque o absoluto). Resolución 640x480, 14 colores y extensión .xpm.gz
<code>hiddenmenu</code>	No aparece el menú de arranque (a no ser que pulsemos alguna tecla), sólo el contador indicado en timeout

Opciones por núcleo

Opción	Descripción
<code>title</code>	Etiqueta a mostrar para cada núcleo
<code>root</code>	Partición raíz para el grub (dónde se encuentra el fichero kernel e ram disk)
<code>kernel</code>	Localización del núcleo y opciones del núcleo
<code>initrd</code>	Localización del fichero con la RAM disk, relativo a root
<code>rootnoverify</code>	Como root para núcleos que GRUB no puede cargar
<code>chainloader</code>	Grub que pase el control a otro boot loader. Indicamos el sector donde se encuentra

Interacción

- ★ Interaccionamos con GRUB durante el proceso de arranque.
- ★ Podemos: elegir un kernel, modificar los parámetros de arranque, arrancar una consola

```
GNU GRUB version 0.97 (639K lower / 1046784K upper memory)

Ubuntu, kernel 2.6.20-15-generic
Ubuntu, kernel 2.6.20-25-generic (recovery mode)
Ubuntu, memtest86+
Other operating systems:
Windows Vista/Longhorn (loader)
Windows Vista/Longhorn (recuperación)

Use the ▲ and ▼ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for command-line.

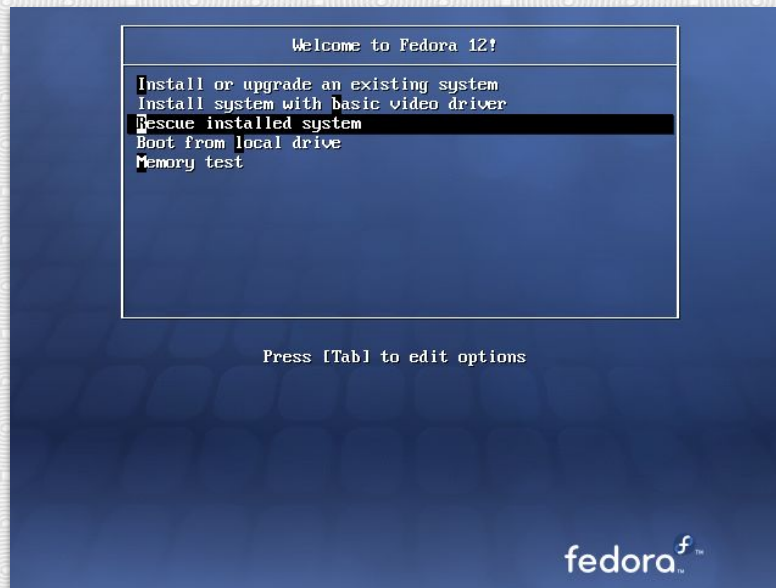
The highlighted entry will be booted automatically in 7 seconds
```

Interacción (II)

- ★ Si hemos indicado la opción `hiddenmenu`, el menú no aparecerá a no ser que pulsemos la tecla **ESC**.

Seleccionar un núcleo

- ★ La acción más habitual con GRUB es querer elegir (cursores + enter) un núcleo diferente al indicado por defecto.



Modificar los parámetros de arranque

- ★ Usamos los cursores para elegir el núcleo que queremos modificar y pulsamos **e** (editar la entrada completa) o **a** (sólo línea kernel).

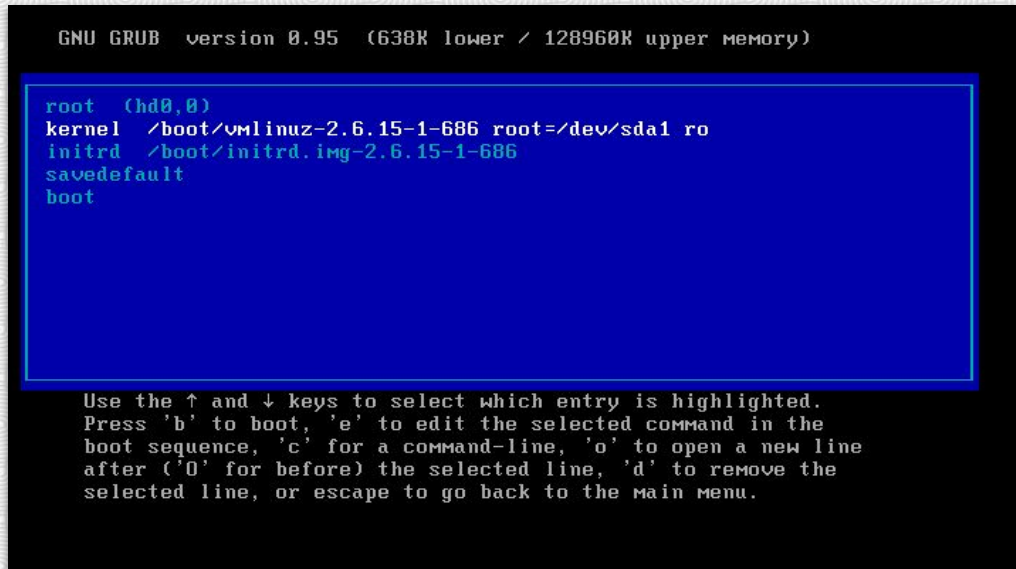
```
GNU GRUB version 0.95 (638K lower / 128960K upper memory)

root (hd0,0)
kernel /boot/vmlinuz-2.6.15-1-686 root=/dev/sda1 ro
initrd /boot/initrd.img-2.6.15-1-686
savedefault
boot
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.

Modificar los parámetros de arranque (II)

- ★ Con los cursores podemos elegir la línea a modificar y si pulsamos **e** comenzaremos a modificarla:



```
GNU GRUB version 0.95 (638K lower / 128960K upper memory)

root (hd0,0)
kernel /boot/vmlinuz-2.6.15-1-686 root=/dev/sda1 ro
initrd /boot/initrd.img-2.6.15-1-686
savedefault
boot
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.

Modificar los parámetros de arranque (III)

- ★ La edición finalizará el pulsar **ENTER** o **ESC** (finalizar sin grabar)
- ★ Arrancamos la entrada modificada pulsando **b**.

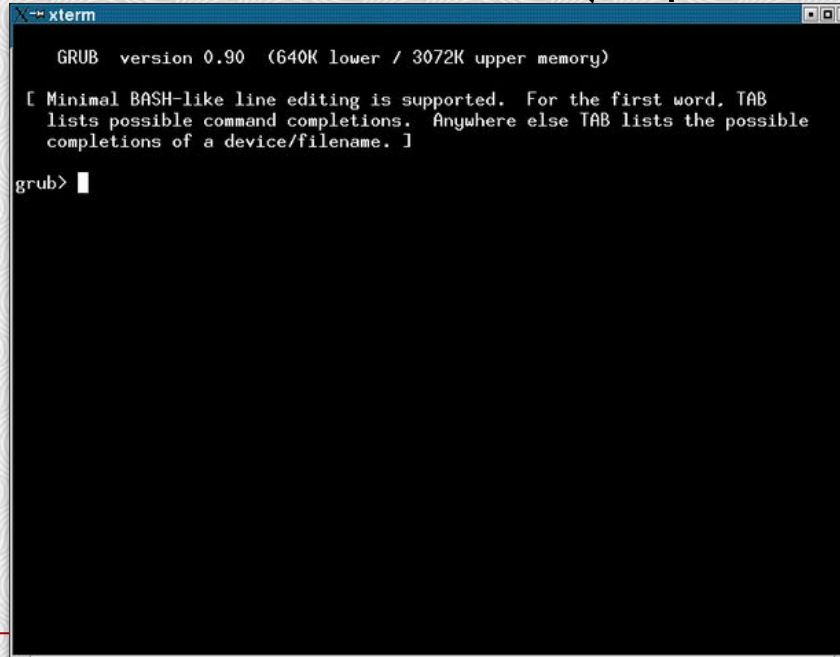
```
GNU GRUB version 0.95 (638K lower / 128960K upper memory)

root (hd0,0)
kernel /boot/vmlinuz-2.6.15-1-686 root=/dev/sda1 ro
initrd /boot/initrd.img-2.6.15-1-686
savedefault
boot
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.

Arrancar consola

- ★ Si GRUB no encuentra el fichero de configuración o pulsamos la opción c, entraremos en el modo shell de GRUB (help command)



```
xterm
GRUB version 0.90 (640K lower / 3072K upper memory)

[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename. ]

grub> █
```

Arrancando en modo monousuario

- ★ En ocasiones es necesario arrancar un sistema monousuario y hacernos pasar por root aunque no tengamos la clave.
- ★ Para lograrlo, añadimos la opción **s** o **single** al final de la línea kernel:

```
grub edit> kernel /boot/vmlinuz-2.6.15-1-686 root=/dev/sda1 ro Single_
```

- ★ Una vez arrancado el sistema usaremos el comando **passwd** para cambiar la clave del root... siempre que el SF no esté en modo sólo lectura

Protegiendo GRUB

- ★ Una medida de seguridad básica es evitar el acceso físico a un servidor.
- ★ Si no lo conseguimos deberíamos proteger la interacción (modificación) con el gestor de arranque.
- ★ GRUB permite asignar una clave para protegerlo.
- ★ Pare ello usamos el comando **grub-md5-crypt** que te pide la clave y te devuelve el hash md5 de la misma:

```
[root@localhost:] # grub-md5-crypt
Password:
Retype Password:
$1$KGnIT$FU80Xxt31J1qU6FD104QF/
```


Protegiendo GRUB (II)

- ★ El valor devuelto se debe poner en el fichero de configuración de GRUB mediante la opción **password --md5**:

```
default=0
timeout=15
splashimage=(hd0,0)/grub/bootimage.xpm.gz
hiddenmenu
password --md5 $1$KGnIT$FU80Xxt31J1qU6FD104QF/
```

- ★ Esta opción también se puede indicar en un title, en este caso sólo afectará al title
- ★ Ahora, cuando arranquemos, tendremos que pulsar **p** seguido de la clave si queremos modificar la configuración de GURB

Instalación

- ★ Los gestores de arranque se instalan como última fase del proceso de instalación del sistema operativo.
- ★ Se suelen instalar en el MBR o en el primer sector de una partición.
- ★ Si, por algún motivo, deseamos reinstalar GRUB deberemos usar el comando **grub-install**:

```
[root@localhost:] # grub-install /dev/hda  
[root@localhost:] # grub-install '(hd0)'  
[root@localhost:] # grub-install /dev/hda1  
[root@localhost:] # grub-install '(hd0,0)'
```

Instalación (II)

- ★ Las razones por las que tenemos que reinstalar grub son:
 - Redimensionado o cambio de la partición root del GRUB
 - Reinstalación de Windows (borra el MBR)
 - Cambio de disco

GRUB2

3

Introducción

- ★ Es usado por la mayoría de las distribuciones modernas y representa una reescritura completa de GRUB Legacy.
- ★ GRUB2 incrementa la flexibilidad y rendimiento.
- ★ Las principales diferencias entre GRUB y GRUB2:
 - GRUB2 soporta más SO
 - GRUB2 permite usar UUID (Universally Unique Identifiers) para identificar dispositivos
 - GRUB2 soporta RAID y LVM
 - La forma de configurarlos es totalmente distinta

Configuración

- ★ Es el elemento en el que más difiere respecto a GRUB Legacy.
- ★ El fichero de configuración principal es `/boot/grub2/grub.cfg` o `/boot/grub/grub.cfg`.
- ★ Estos ficheros no se editan directamente, en su lugar editamos el fichero `/etc/default/grub` y/o modificamos los contenidos del directorio `/etc/grub.d/`.

/etc/default/grub

★ El fichero `/etc/default/grub` y permite establecer opciones globales

```
GRUB_DEFAULT=0
#GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
GRUB_CMDLINE_LINUX=""

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"
```

/etc/default/grub (II)

```
# Uncomment to disable graphical terminal (grub-pc only)
#GRUB_TERMINAL=console
# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports via VBE
# you can see them in real GRUB with the command `vbeinfo'

#GRUB_GFXMODE=640x480
# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux
#GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

# Uncomment to get a beep at grub start
#GRUB_INIT_TUNE="480 440 1"
```

/etc/default/grub (II)

Opción	Descripción
GRUB_DEFAULT	El número indica el kernel por defecto
GRUB_SAVEDEFAULT	Recuerda el último núcleo arrancado
GRUB_TIMEOUT	Segundo a esperar antes de arrancar el núcleo por defecto (-1 → inf)
GRUB_TIMEOUT_STYLE	Hidden -> el título del menu no aparece en la cuenta atrás
GRUB_CMDLINE_LINUX	Parámetros del kernel a añadir a todos los title
GRUB_CMDLINE_LINUX_DEFAULT	Como el anterior pero no se añade al recovery

/etc/grub.d/

- ★ Los ficheros en `/etc/grub.d/` controlan distintas acciones del GRUB, como la búsqueda automática de nuevos núcleos:

```
[root@localhost:/etc/grub.d]$ ls
00_header          20_linux_xen      30_os-prober      41_custom
05_debian_theme    20_memtest86+    30_uefi-firmware  README
10_linux           25_custom        40_custom
```

- ★ En ocasiones queremos añadir un nuevo title o modificar las opciones de uno ya creado, para ello debemos modificar el fichero `40_custom`

40_custom

```
menuentry "Fedora (2.6.32)" {  
    set root=(hd0,1)  
    linux /vmlinuz-2.6.32 ro root=/dev/hda5 mem=2048M  
    initrd /initrd-2.6.32  
}  
menuentry "Debian (2.6.36-experimental)" {  
    set root=(hd0,1)  
    linux (hd0,1)/bzImage-2.6.36-experimental ro root=/dev/hda6  
}  
#  
# Other operating systems  
#  
menuentry "Windows" {  
    set root=(hd0,2)  
    chainloader +1  
}
```

Interacción

- ★ Es muy similar a la interacción descrita en GRUB.
- ★ El acceso al menú se obtiene pulsado **MAY**.

```
GNU GRUB  version 1.97~beta3

recordfail=1
save_env recordfail
set quiet=1
insmod ext2
set root=(hd0,1)
search --no-floppy --fs-uuid --set 904bf39-9234
linux /boot/vmlinuz-2.6.31-9 root=UUID=904bf39-9234 ro quiet splash
initrd /boot/initrd.img-2.6.31-9-generic

Minimum Emacs-like screen editing is supported. TAB lists
completions. Press Ctrl-x to boot, Ctrl-c for a command-line
or ESC to return menu.
```

Grabando la configuración

- ★ Una vez modificados los ficheros de configuración, haremos los cambios persistentes invocando a `update-grub` o `grub2-mkconfig`
- ★ Durante la ejecución de estos comando GRUB2 detectará automáticamente los nuevos núcleos instalados y creará los títulos oportunos.
- ★ El resultado de invocar estos comando se ve reflejado en el fichero `/boot/grub/grub.cfg`

Arrancando en modo monousuario

- ★ Deberíamos seguir los pasos indicados para GRUB, pero en vez de indicar `s` o `single`, ponemos `init=/bin/bash`

```
grub edit> kernel /boot/vmlinuz-2.6.15-1-686 root=/dev/sda1 ro Single
```



- ★ Una vez hecho esto montamos el sistema de ficheros raíz en modo `rw`

```
[root@localhost:]$ mount -o remount,rw /
```

- ★ Ejecutamos el comando `passwd`
- ★ Ejecutamos el comando siguiente si SELinux está activo

```
[root@localhost:]$ touch /.autorelabel
```

Protegiendo GRUB2

- ★ En ese caso podemos definir usuarios y claves para controlar el acceso a GRUB2
- ★ Creamos cuenta de superusuario añadiendo a `/etc/grub.d/01_users`

```
cat <<EOF
set superuser="bo"
Password bo bopass
EOF
```

- ★ Para usuarios regulares

```
cat <<EOF
set superuser="bo"
password bo bopass
password sa sapass
EOF
```


Protegiendo GRUB2 (II)

- ★ Si queremos dejar titles sin protección usamos la opción **--unrestricted**:

```
menuentry 'test1' --unrestricted{  
[...]  
}
```

- ★ Si queremos indicar los usuarios que tienen acceso usamos la opción **--users**

```
menuentry 'test1' --users jane{  
[...]  
}
```

Protegiendo GRUB2 (III)

- ★ Podemos usar claves cifradas gracias al comando `grub2-mkpasswd-pbkdf2`
- ★ En este caso el fichero `/etc/grub.d/01_users` sería:

```
cat <<EOF
set superuser="bo"
password_pbkdf2 bo
[valor devuelto por grub2-mkpasswd-pbkdf2]
EOF
```

Systemd y modos de rescate

4

Introducción

- ★ Ante situaciones de emergencia systemd proporciona dos modos de arranque.
- ★ *Emergency mode* se caracteriza por:
 - Only mounts the root filesystem, read-only.
 - No network interfaces are activated.
 - Some basic service are started.
 - Requires root password to enter this mode.

Introducción

★ *Rescue mode:*

- It tries to mount all local filesystems.
- Starts important services.
- No network interface are activated.
- No users are allow on.
- Is same as traditional single user mode.
- Requires the root password to login.

Arrancando en modo rescate o emergencia (I)

- ★ Debemos modificar el arranque del GRUB y añadir a la línea linux16 `systemd.unit=rescue.target`

```
linux16 /vmlinuz-3.10.0-1160.36.2.el7.x86_64 root=/dev/mapper/centos-r\
oot ro crashkernel=auto rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet\
LANG=es_ES.UTF-8 systemd.unit=rescue.target
initrd16 /initramfs-3.10.0-1160.36.2.el7.x86_64.img
```

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.

- ★ Pulsamos Ctrl-x y el resultado final sería

```
Welcome to rescue mode! After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or ^D to
boot into default mode.
Give root password for maintenance
(or press Control-D to continue): _
```

Arrancando en modo rescate o emergencia (II)

- ★ Debemos modificar el arranque del GRUB y añadir a la línea linux16 `systemd.unit=emergency.target`

```
linux16 /vmlinuz-3.10.0-1160.36.2.el7.x86_64 root=/dev/mapper/centos-r\
oot ro crashkernel=auto rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet\
LANG=es_ES.UTF-8 systemd.unit=emergency.target
initrd16 /initramfs-3.10.0-1160.36.2.el7.x86_64.img
```

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.

- ★ Pulsamos Ctrl-x y el resultado final sería

```
Welcome to emergency mode! After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or ^D to
try again to boot into default mode.
Give root password for maintenance
(or press Control-D to continue): _
```

Arrancando en modo rescate o emergencia

- ★ Si queremos acceder al sistema sin que nos pide la clave de administrador podemos lanzar una *systemd debug shell*
- ★ Debemos modificar el arranque del GRUB y añadir a la línea linux16 *systemd.debug-shell*

```
linux16 /vmlinuz-3.10.0-1160.36.2.el7.x86_64 root=/dev/mapper/centos-r\
oot ro crashkernel=auto rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet\
LANG=es_ES.UTF-8 systemd.debug-shell
initrd16 /initramfs-3.10.0-1160.36.2.el7.x86_64.img ↓

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.
```

- ★ Pulsamos Ctrl-x y el sistema arranca como siempre.

Arrancando en modo rescate o emergencia

★ Para acceder a la *systemd debug shell* pulsamos CTRL+9:

```
[root@localhost ~]#
```


Arrancando en modo rescate o emergencia

- ★ Con el sistema ya arrancado podemos ir al modo rescate ejecutando `systemctl rescue`:

```
systemctl rescue
```

- ★ Si queremos evitar mandar mensajes de *broadcast* a todos los usuarios conectados:

```
systemctl --no-wall rescue  
systemctl isolate rescue.target
```


Arrancando en modo rescate o emergencia

- ★ Con el sistema ya arrancado podemos ir al modo emergencia ejecutando `systemctl emergency`:

```
systemctl emergency
```

- ★ Si queremos evitar mandar mensajes de *broadcast* a todos los usuarios conectados:

```
systemctl --no-wall emergency  
systemctl isolate emergency.target
```