


# **Planificación para la implantación de un SO**

**Estudio del Inicio de Distribuciones Linux**

# LINUX: Elementos de Inicio

- Gestores de arranque
  - Existe gran variedad de gestores de arranque para Linux:
    - bootman
    - GRUB
    - LILO
    - NTLDR
    - XOSL
    - BootX
    - loadlin
    - Gujin
    - Boot Camp
    - Syslinux
    - GAG
  - Los más extendidos son LILO y GRUB

# LINUX: Elementos de Inicio

- LILO (Linux Loader) 
  - Este programa se puede alojar en el MBR, en el bootsector de la partición del sistema o en un floppy
  - Es independiente del Sistema de Ficheros (soporta múltiples sistemas de ficheros)
  - Acepta hasta 16 entradas de SSOO
  - Es necesario modificarlo con cada cambio del kernel de Linux
  - Su configuración se realiza a través del archivo **/etc/lilo.conf**.  
[Existe un manual de configuración en Internet](#)
  - **LILO** trabaja con la extensiones BIOS de los dispositivos. Esto supone una limitación para acceder a datos alojados más allá de 1024 cilindros



# LINUX: Elementos de Inicio

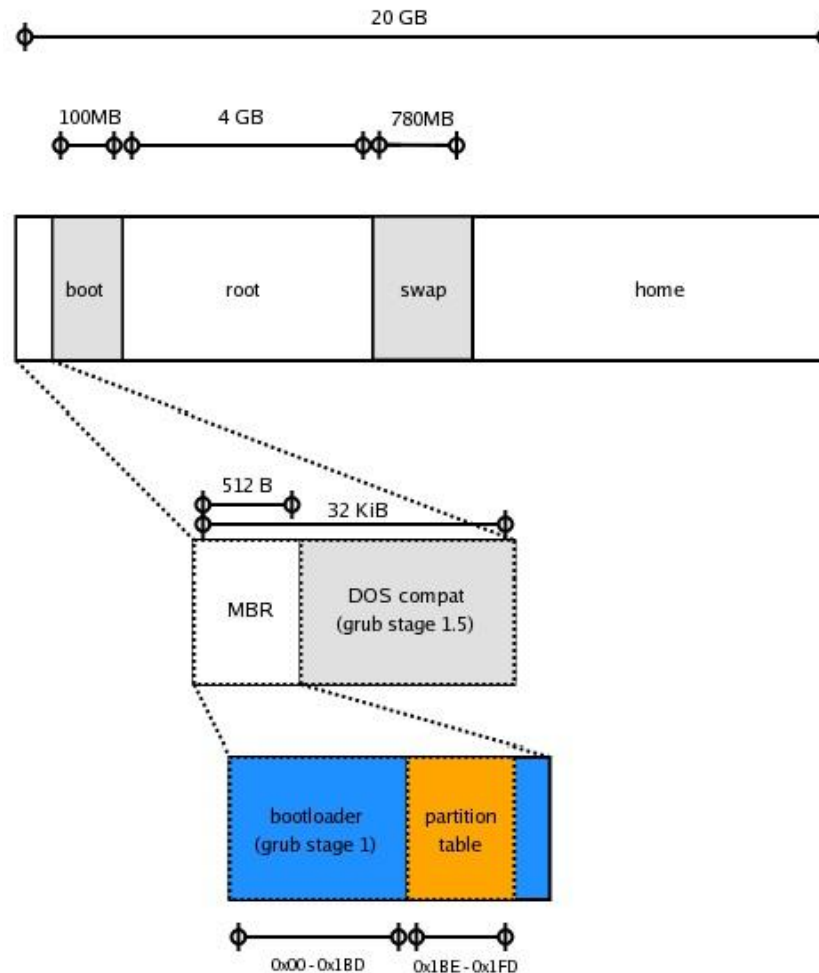
- **GRUB (GRand Unified Bootloader)**
  - GRUB es un programa GNU
  - Soporta múltiples Sistemas de ficheros lo que le hace ideal para gestionar el arranque de multiples SSOO de diferentes plataformas
  - Posee una consola para su recuperación y otra para su configuración
  - Permite arrancar sistemas desde la red
  - GRUB posee tres fases de arranque: fase 1, fase 1.5 y fase 2
  - Fase 1
    - Se aloja en el MBR o en el primer sector de la partición que aloja al sistema
    - Su misión es enlazar con la fase 1.5, aunque también puede pasar directamente a la fase 2

# LINUX: Elementos de Inicio

- GRUB (**GR**and **U**nified **B**ootloader)
  - Fase 1.5
    - Se aloja en los siguientes 30KB del disco.
    - Su misión es enlazar con la fase 2
  - Fase 2
    - La fase 2 se encuentra en **/boot**
    - Carga el menú de arranque para que el usuario seleccione la opción deseada
    - Ejecuta el la imagen de **kernel** por defecto o la seleccionada por el usuario

Grub se distribuye entre tres partes: MBR o bootsector de la partición de Linux, en los 30KB siguientes y en la ruta /boot

# LINUX: Elementos de Inicio



# LINUX: Elementos de Inicio

- GRUB2 (**GR**and **U**nified **B**ootloader)
  - GRUB2 es un proyecto GNU independiente de GRUB
  - Pretende mejorar la robustez y seguridad de GRUB
- GRUB2 vs GRUB
  - GRUB no es compatible con GRUB2
  - ✗ GRUB lee la configuración de arranque de `/boot/grub/menu.lst`
  - GRUB2 utiliza dos archivos de configuración de arranque
    1. Configuración de arranque en `/boot/grub/grub.cfg`
    2. El usuario modifica su configuración a través de la edición del archivo `/etc/default/grub` CUIDADO
    3. El archivo `grub.cfg` se actualiza a través de un conjunto de scripts situados en `/etc/grub.d`. Los scripts son invocados por medio del comando `update-grub2`



# LINUX: Elementos de Inicio

- **KERNEL** SE PUEDE LOCALIZAR UNA VEZ REALIZADA LA PRÁCTICA MULTISISTEMA O EN EL ANFITRIÓN
  - En un sistema monolítico el kernel es la parte central y más importante del sistema operativo
  - El kernel **se carga en memoria RAM** y se mantiene hasta el apagado del sistema.
  - El kernel ejecuta el programa **init** (inicio clásico) o **systemd** (inicio evolucionado)
  - La imagen del **kernel** no se almacena en un ejecutable sino en un archivo comprimido en la ruta **/boot**
    - **Zimage** para imágenes menores de 512Kbytes
    - **BzImage** para imágenes superiores a 512kbytes LOCALIZAR

**uname -r** /\* Este comando nos permite ver la versión del kernel cargado \*/



# LINUX: Secuencia de Inicio

- **Inicio clásico (init)**

SEGÚN LA VERSIÓN SE PUEDE O NO LOCALIZAR  
SI NO ESTÁ, PASAR AL MODELO EVOLUCIONADO

- Adoptado por Linux con origen en los grandes sistemas operativos **SysV** y **BSD** (ambos versiones de **Unix**).
- Se basa en la ejecución de un script lanzado por el proceso **init** para poner en marcha el resto de procesos que intervienen en la secuencia de inicio.

- **Inicio evolucionado (systemd)** LOCALIZAR

- Desde 2015, la mayoría de las versiones que utilizan Linux como núcleo han incorporado **systemd** como demonio para el inicio del sistema.
- Este demonio sustituye a **init** y por tanto la secuencia de inicio varía

# LINUX: Secuencia de Inicio Clásico

- Proceso init
  - El kernel lanza al proceso **init** LOCALIZAR
  - El proceso init es el responsable de ejecutar alguno de los scripts indicados en **/etc/inittab**
  - Los scripts se organizan mediante “**run-levels**”. Al menos uno de ellos se ejecuta por defecto.
  - Cada script lanza los procesos del “**run-level**” asociado.

El proceso init se convierte en el **proceso padre de todos los procesos de la máquina**

# LINUX: Secuencia de Inicio Clásico

- Run-levels
  - Seleccionan el modo de arranque del SO.
  - El run-level establece el script debe ser ejecutado al arrancar la máquina.
  - Los run-levels se indican en el archivo **/etc/inittab**
  - Los scripts asociados a cada run-level ejecutan el conjunto de procesos asociados

- Run-levels de Debian

LOCALIZAR

0 (halt the system)  
1 (single-user / minimal mode),  
2 through 5 (multiuser modes), and  
6 (reboot the system).

Por defecto Debian no diferencia entre los modos 2 y 5. El modo 1 es para mantenimiento

# LINUX: Secuencia de Inicio Clásico

- Archivos rc#.d
  - Se encuentran en la ruta **/etc/rc.d**
  - Son los scripts que lanzan los procesos asociados a cada run-level
  - El caracter **#** se sustituye por el valor comprendido entre 0-6, en función del run-level adecuado

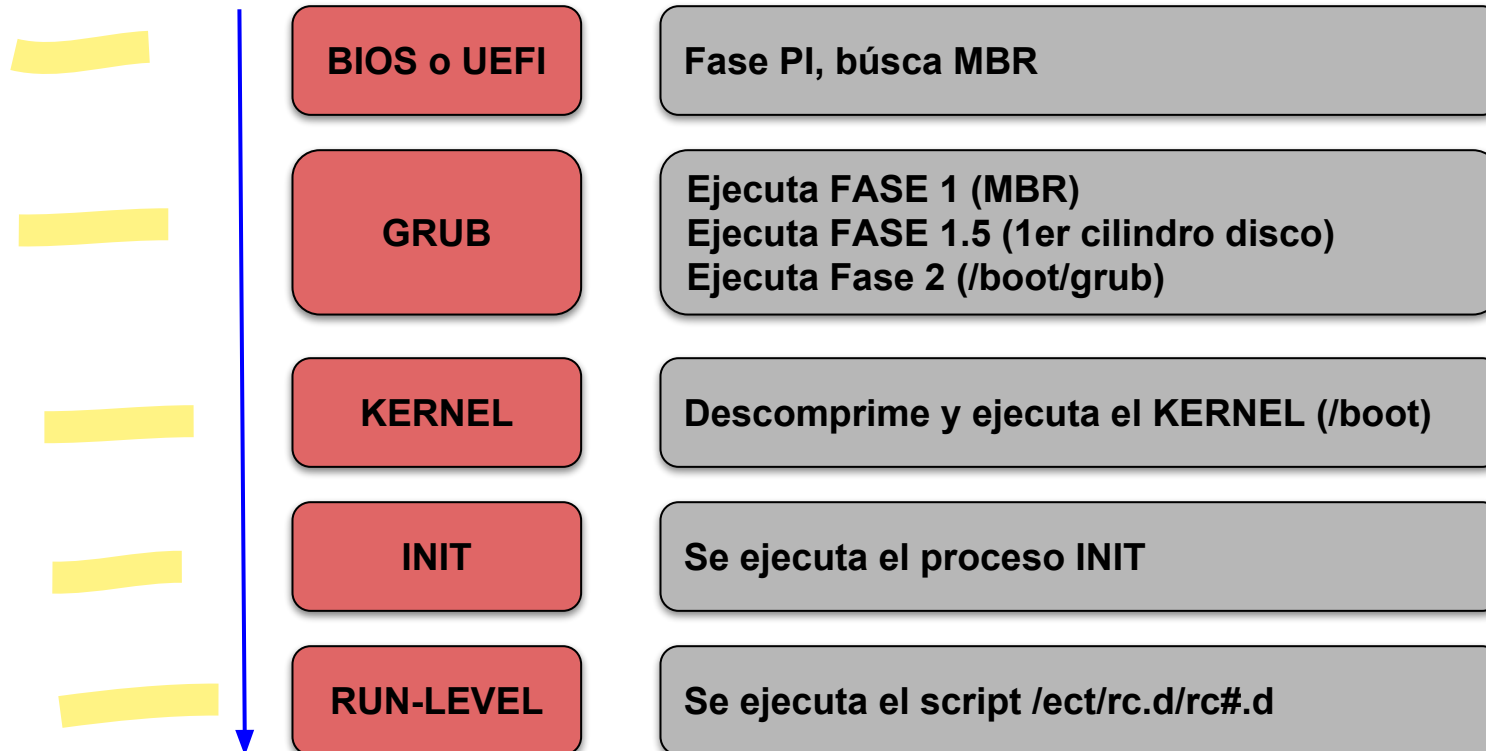
LOCALIZAR

# LINUX: Secuencia de Inicio Clásico

- Archivos init.d LOCALIZAR
  - Es un directorio en la ruta **/etc/init.d** (Debian)
  - Contiene los procesos (**demonios**) del sistema apuntados por los scripts contenidos en **/etc/rc.d**
  - El administrador puede realizar sobre ellos operaciones como **start, stop, restart, reload**

# LINUX: Secuencia de Inicio Clásico

- Inicio clásico (init)



# LINUX: Secuencia de Inicio evolucionado

- Proceso **systemd** BUSCAR
  - El kernel lanza al proceso **systemd** BUSCAR
  - Systemd utiliza archivos **target** para establecer su configuración en tiempo de arranque.
  - El archivo target de configuración por defecto es **/etc/systemd/system/default.target**. Se trata de un enlace simbólico al archivo de configuración real (target).
  - Systemd también mira si existe un arranque init heredado, en caso de existir utilizará archivos proporcionados para garantizar la compatibilidad

BUSCAR

El proceso systemd se convierte en el **proceso padre de todos los procesos de la máquina**

# LINUX: Secuencia de Inicio evolucionado

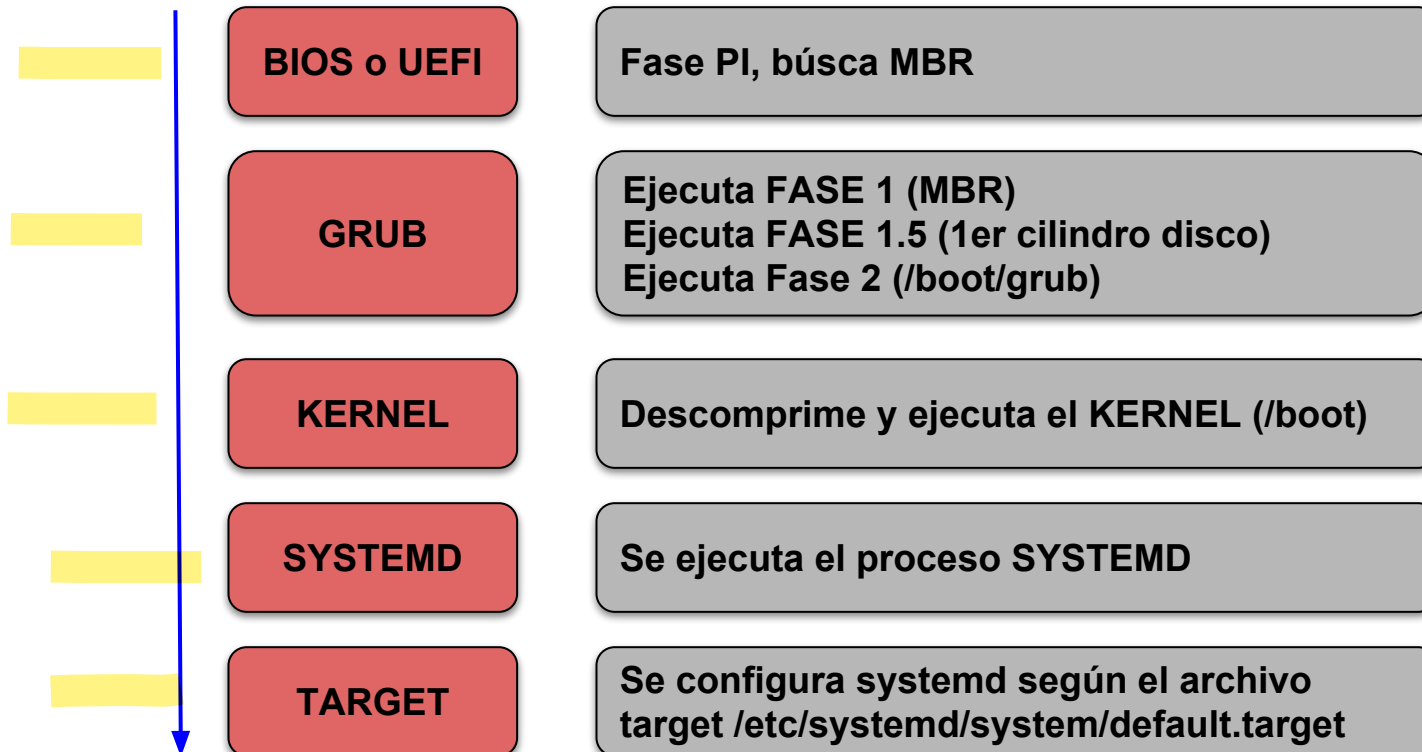
- Archivos Target
  - Cada **target** tiene descritas un conjunto de dependencias y el orden de arranque de los servicios.
  - Existe una correlación entre los archivos target y los run-levels del sistema clásico.

level	systemd target	Runlevel target alias	Description
0	poweroff.target	runlevel0.target	Halts the system and turns the power off.
1	rescue.target	runlevel1.target	A base system including mounting the filesystems with only the most basic services running and a rescue shell on the main console.
3	multi-user.target	runlevel3.target	All services running but command line interface (CLI) only.
5	graphical.target	runlevel5.target	multi-user with a GUI.
6	reboot.target	runlevel6.target	Reboot



# LINUX: Secuencia de Inicio evolucionado

- Inicio evolucionado (systemd)



# LINUX: Reparación del arranque

- **Rescataux** ES UNA ISO MULTIHERRAMIENTA DE REPARACIÓN
  - Se trata de una herramienta GNU que te permite entre otras cosas arreglar el arranque GRUB/GRUB2
  - Se distribuye como una imagen para USB o CDROM
- **SuperGrub2**
  - Te permite arrancar un sistema con arranque Grub/Grub2 “roto”
  - Está incluida en **rescataux** pero se puede descargar de forma independiente
- Consola de recuperación Grub
  - Mediante comandos de consola es posible arreglar el arranque

# LINUX: Secuencia de Inicio

## PRACTICA MULTISISTEMAS

