

Práctica 2: Instalación y Arranque del Sistema

Objetivos

En esta práctica se analiza el proceso de instalación “manual” del sistema operativo. Estudiaremos una configuración típica que sirve para ejemplificar los aspectos más importantes de la instalación y sus características. Una vez configurado el sistema veremos el proceso de arranque y la configuración de servicios.

Preparando el entorno...

Vamos a realizar la instalación de un sistema operativo CentOS7 en una máquina virtual. Configuraremos inicialmente la máquina virtual con la que vamos a trabajar de la siguiente forma:

- El tipo de sistema operativo es Red-Hat Linux de 64 bits.
- La memoria virtual de la máquina tiene que ser suficiente para ejecutar el entorno gráfico de instalación (1024MB).
- El disco tendrá formato VDI de 10GB y **no** se asignará todo el espacio inicialmente.
- Además tendrá una unidad DVD con el dvd de instalación que está disponible en el path: `/mnt/DiscoVMs/ASR/CentOS-7-x86_64-DVD-1511.iso`. Se añadirá una vez creada la máquina virtual en la sección de almacenamiento. Finalmente, se configurará un interfaz de red en modo NAT.

El método de instalación no requiere el uso de la red, alternatively es posible instalar el sistema usando imágenes mínimas de instalación en un CD, memoria USB o descargadas por red con Kickstart y PXE.

Instalación del Sistema Operativo CentOS 7

La mayor parte del proceso de instalación es guiado y no debe presentar ningún problema, los pasos que se describen a continuación son una referencia básica que puede variar entre distintas versiones del sistema. **Leer atentamente las instrucciones del instalador.**

Fases de la Instalación

Las fases de la instalación son:

- **Arranque**, GRUB es el gestor de arranque que inicializa el proceso de instalación (y más tarde el del sistema instalado). Permite diversas opciones, en nuestro caso seleccionaremos la de instalación.

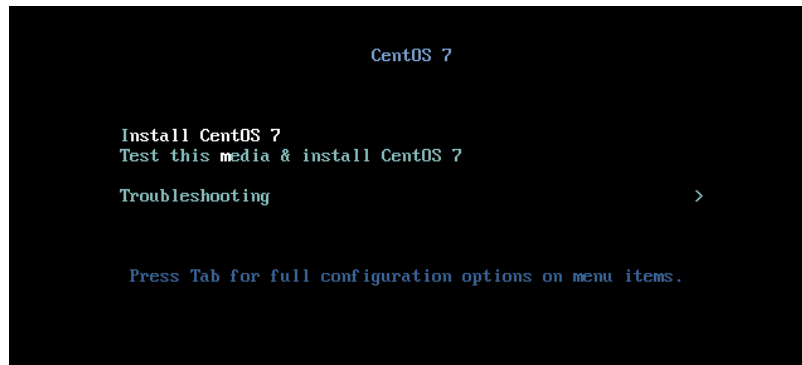


Figura 1. Ventana de arranque para iniciar la instalación del sistema

- **Selección del idioma**, este es el idioma del sistema que puede ser distinto al que se use para configurar los usuarios que se pueden añadir como idiomas adicionales en el menú principal de la instalación. Inglés es el idioma recomendado.
- **Zona horaria y fecha**, puede seleccionarse la sincronización horaria con servidores de tiempo. En nuestro caso no la activaremos ya que no dispondremos de conexión a Internet.
- **Teclado**, elegir la disposición en español.
- **Políticas de seguridad**, deshabilitar las políticas de seguridad.
- **Medio de instalación**, para la instalación desde CD-ROM no es necesario especificar otro medio, opcionalmente se puede verificar aunque en nuestro caso no es necesario hacerlo.
- **Selección de paquetes software**. Hay aproximadamente 2500 aplicaciones disponibles, el instalador ofrece pre-selecciones de software para instalar el sistema. Elegir Infrastructure Server y dedicar unos minutos a explorar las diferentes categorías de software, y grupos de aplicaciones.

Nota: Una buena práctica es elegir una instalación mínima y añadir sólo el software que se va a usar (menos paquetes implica menos vulnerabilidades, sistemas más fáciles de mantener, y dependencias software menos complejas).

- **Dispositivo de instalación y particionado**, seleccionar el único disco disponible en la máquina. En este primer diálogo se pueden añadir discos adicionales en red o sistemas de almacenamiento externo.

Además permite seleccionar el modo de particionado del disco, automático o manual para personalizar la disposición de las particiones. En nuestro caso elegiremos particionado manual.

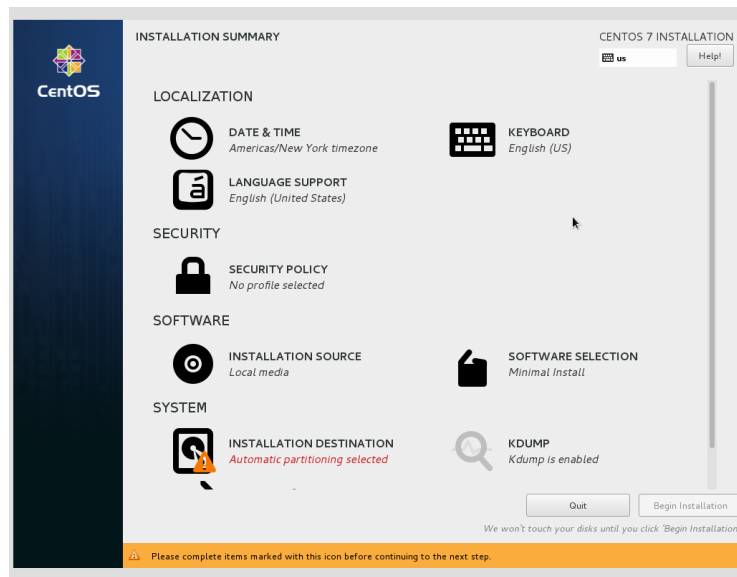


Figura 2. Ventana inicial principal de la instalación del sistema.

Una vez seleccionado el disco y modo de particionamiento el siguiente diálogo es la creación de particiones.

Todas las particiones se crearán en el único disco que hemos añadido que será accesible en el dispositivo `/dev/sda`. Cada partición se nombrará automáticamente con un número y el nombre de dispositivo (ej. `sda1` ó `sda5`, `sd` nombra discos SCSI). En una práctica posterior analizaremos en detalle la administración de discos, en esta práctica usaremos la siguiente disposición:

- `/boot` con 512MB, tipo partición estándar y sistema de ficheros xfs
- `swap` con 1024MB, tipo swap
- `/` con el resto (no poner tamaño), tipo partición estándar y sistema de ficheros xfs

Finalmente una vez creadas todas las particiones el instalador presenta un resumen de las acciones que realizará.

Ejercicio. La disposición del sistema de directorios en Linux sigue el estándar FHS (*file hierarchy standard*) que define el nombre, ruta y propósito de los directorios. Con ayuda del estándar, http://refspecs.linuxfoundation.org/FHS_2.3/fhs-2.3.html completar:

Directorio	Propósito
<code>/bin</code>	<code>/bin</code> contains commands that may be used by both the system administrator and by users, but which are required when no other filesystems are mounted (e.g. in single user mode). It may also contain commands which are used indirectly by scripts.
<code>/boot</code>	his directory contains everything required for the boot process except configuration files not needed at boot time and the map installer. Thus <code>/boot</code> stores data that is used before the kernel begins executing user-mode programs. This may include saved master boot sectors and sector map files

/dev	The /dev directory is the location of special or device files.
/etc	The /etc hierarchy contains configuration files. A "configuration file" is a local file used to control the operation of a program; it must be static and cannot be an executable binary
/home	/home is a fairly standard concept, but it is clearly a site-specific filesystem. The setup will differ from host to host. Therefore, no program should rely on this location.
/usr	/usr is the second major section of the filesystem. /usr is shareable, read-only data. That means that /usr should be shareable between various FHS-compliant hosts and must not be written to. Any information that is host-specific or varies with time is stored elsewhere. Large software packages must not use a direct subdirectory under the /usr hierarchy.
/usr/bin	This is the primary directory of executable commands on the system.
/usr/include	This is where all of the system's general-use include files for the C programming language should be placed.
/var	
/var/log	
/var/lib	

- **Red y nombre de host (hostname)**, este es el nombre de host, que puede ser distinto del nombre DNS. Elegir un nombre cualquiera y no configurar la red (estudiaremos la configuración de la red en prácticas posteriores).

En este momento puede comenzar la instalación, durante este proceso es posible configurar:

- **Contraseña de root.**
- **Cuenta de usuario**, nunca hay que interactuar con el sistema entrando (*login*) con el usuario root. Es recomendable crear una cuenta adicional que tenga permisos de administración, probar a crear una cuenta ejemplo, más adelante estudiaremos en detalle la gestión de usuarios y grupos.

Finalmente simplemente **reiniciar el sistema**, en el primer arranque se solicitará alguna información adicional.

Para saber más...

- Estudiar cómo preparar una memoria USB como medio de instalación.
- Estudiar la aplicación kickstart.
- Consultar la guía del sistema operativo Red Hat Enterprise Linux. El primer arranque permite configurar algunos aspectos del sistema (e.j modelo de seguridad o actualizaciones automáticas).

Arranque del sistema

El arranque del sistema comprende desde el momento en que se pulsa el botón de encendido hasta que el sistema operativo ofrece la posibilidad de acceder él (*login*). En esta sección nos ocuparemos de la primera parte de este proceso que culmina cuando se ha cargado el kernel y comienza el arranque de los distintos servicios. Esta segunda parte se estudiará en la siguiente sección.

Actualmente hay dos estándares para la disposición de la información en el disco (particiones) y el proceso de arranque: el tradicional BIOS y UEFI. Aunque UEFI pretende solucionar algunas de las limitaciones del esquema tradicional presenta algunos problemas, p. ej. secure boot, firmware bugs, ó licencias, por lo que BIOS no se ha abandonado completamente. En este curso estudiaremos el esquema tradicional.

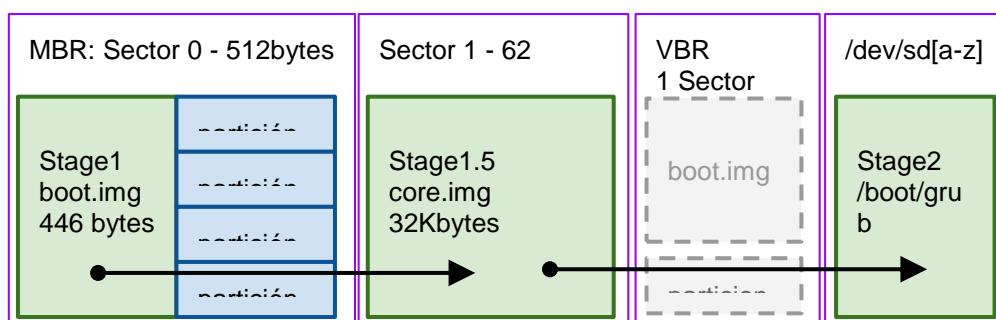


Figura 3. Versión simplificada de la disposición del cargador en disco, particionado MS-DOS

El arranque consiste en:

1. Auto comprobación del sistema por la BIOS
2. La BIOS carga del MBR (primer sector de 512 bytes del disco de arranque)
3. Cargador del sistema (paso 1) en nuestro caso GRUB que ocupa los 62 sectores siguientes, antes de la primera partición.
4. Cargador del sistema (paso 2) se leen sectores adicionales
5. El cargador (paso 2) carga en memoria el kernel y posiblemente un RAM disk.

GRUB: La shell de arranque

Una vez cargado GRUB (pasos 1 y 2) se muestra una consola para la selección del sistema o programa que se quiere cargar.

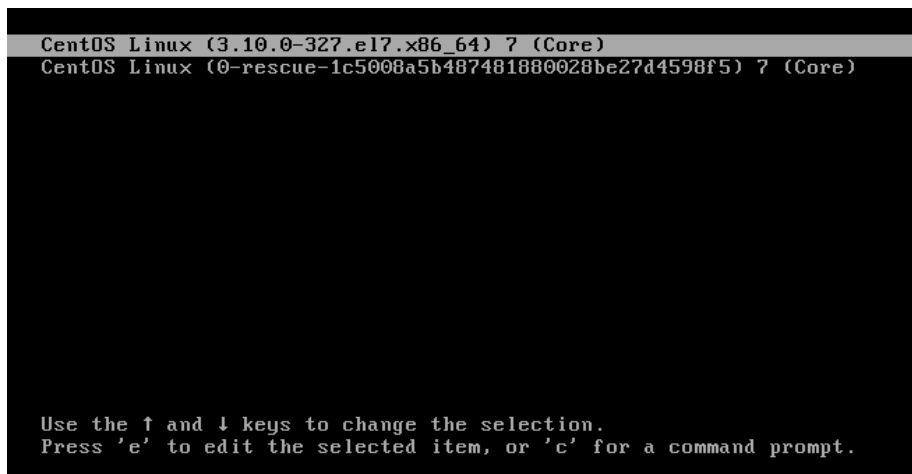


Figura 4. El menú de selección de GRUB

Ejercicio 1. Entrar en el modo edición de la carga del sistema operativo presionando 'e', mostrará la lista de comandos que se ejecutarán uno en cada línea, determinar cuál es el propósito de los siguientes comandos:

- `set root='hd0,msdos1'`
- `insmod`
- `linux16 e initrd16`

Utilizar la información sobre grub disponible en el sistema o la documentación online (<http://www.gnu.org/software/grub/manual/grub.html>).

Ejercicio 2. Muchas veces para reparar un sistema debe arrancar en modo mono-usuario, añadir la opción `init=/bin/bash` al final de la línea que carga el kernel. Arrancar el sistema con Ctrl-x.

Comprobar que no se solicita contraseña y la importancia de proteger BIOS/GRUB con *password* en entornos no seguros. Finalmente, reiniciar el sistema con el comando `reboot`.

Ejercicio 3. Estudiar los contenidos del directorio `/boot` y relacionarlos con los del fichero `/boot/grub2/grub.cfg`. Este fichero se genera automáticamente a partir de los ficheros ubicados en `/etc/grub.d/` y `/etc/default/grub`, durante la instalación del sistema o de un kernel nuevo. A modo de ejemplo cambiaremos el `timeout` por defecto a 10 segundos:

- Modificar la variable correspondiente en el archivo `/etc/default/grub`.

```
cd /etc/default ; vim grub. #modificar timeout=10
```

- Actualizar la configuración con `grub2-mkconfig -o /boot/grub2/grub.cfg`. Comprobar los cambios con `grep`, por ejemplo.

```
grub2-mkconfig -o /boot/grub2/grub.cfg; cat /boot/grub2/grub.cfg
```

Ejercicio 4. Para asegurar el acceso físico al sistema es necesario proteger el arranque. Por defecto la edición y selección de las entradas no está restringido. En este ejercicio modificaremos este comportamiento y asignaremos una contraseña de acceso:

- Estudiar el script `/etc/grub.d/10_linux`. Modificarlo para eliminar la clave `--unrestricted` en la generación de cada entrada del menú (variable `CLASS`).

cd /etc/grub.d/ ; cat -n 10_linux | grep unrestricted #para conocer la línea a modificar

vim 10_linux #eliminamos el unrestricted

- Añadir un usuario y contraseña. El usuario será de tipo superuser, para que pueda acceder a todas la entradas. Modificar el script /etc/grub.d/01_users para que defina el usuario con el siguiente formato:

Listado 1. Contenido del fichero /etc/grub.d/01_users

```
#!/bin/sh -e
cat <<EOF
set superusers="cursoasr"
password cursoasr cursoasr
EOF
```

- Actualizar la configuración con **grub2-mkconfig -o /boot/grub2/grub.cfg**.
- **(Opcional)** Guardar la contraseña cifrada usando el comando grub2-mkpasswd-pbkdf2. En este caso hay que cambiar el comando password por password_pbkdf2 en el fichero 01_users

Ejercicio 5. Con la ayuda del comando dmesg observar los mensajes que produce el kernel durante esta fase. Para consultar la salida del comando de forma cómoda combinarlo (tubería |) con less .

A continuación se pasa el control al proceso init (el primer proceso del sistema con PID=1) que en CentOS7 y muchas distribuciones de Linux es systemd que veremos en la siguiente sección.

dmesg | less

Ejercicio 6 (Opcional). En algunas circunstancias puede resultar útil hacer una copia de seguridad de la master boot record (incluida tabla de particiones). Para copiar en crudo bytes entre dispositivos se utiliza el comando dd:

- Estudiar man dd, especialmente las opciones if, of, bs, count, seek y skip
 - dd= convierte y copia un archivo**
 - if = input file (origen)**
 - of = output file (destino)**
 - bs= lee y escribe bytes de una vez**
 - count= copia bloques de tamaño en bytes determinado por ifs de fi, en vez de todo hasta el final del fichero**
 - seek= se salta bloques de tamaño en bytes determinado por obs de fo antes de la copia**
 - skip= se salta bloques de tamaño en bytes determinado por ifs de fi antes de la copia**
- En primer lugar, mostrar la tabla de particiones del dispositivo usando el comando **fdisk -l <nombre del disco>**
- Guardar la MBR (primeros 512bytes del disco) en un archivo, p. e.j /tmp/mbr.bck.

Copia de seguridad MBR:: dd if=/dev/sda of=/tmp/mbr.bck bs=512 count=1

- Poner a zeros los primeros 512bytes del disco, utilizar /dev/zero como archivo de entrada, para el comando dd. Comprobar que no se puede acceder a la tabla de particiones del sistema con el comando fdisk

Borrar MBR:: `if=dd /dev/zero of=/dev/sda bs=512 count=1`

`fdisk -l /dev/sda`

- Restaurar la copia con el comando dd y comprobar que se ha restaurado correctamente.

Restaurar MBR:: `dd if='/tmp/mbr.bck of=/dev/sda bs=512 count=1'`

Para saber más...

- Debido a la variedad de software disponible no se ha analizado UEFI. Un buen entendimiento de su funcionamiento es de gran ayuda.
- Una alternativa a GRUB es el proyecto SYSLINUX, muy versátil y utilizada en muchos liveCD. Estudiar sus características www.syslinux.org
- Muchas de las opciones de la línea del kernel se aplican una vez cargado el kernel se procesan por el RAM disk, mediante dracut. Consultar la información y opciones de este sistema en: <http://www.kernel.org/pub/linux/utils/boot/dracut/dracut.html#dracutcmdline7>
- GRUB dispone además de un modo comandos ('c') estudiar los comandos básicos: find, cat, boot, root

Niveles de Ejecución y Servicios

Una vez cargado el kernel empieza el proceso de arranque de servicios básicos con la ayuda del RAM disk (disco con un sistema de ficheros especial initramfs) que se descomprime en la RAM del sistema. En esta parte se inicializan los sistemas y periféricos básicos (CPU, RAM, discos y particiones...) y en el caso de RedHat/CentOS el sistema SELinux.

Systemd se estructura en *units*, que representan archivos de configuración relativos a servicios del sistema, niveles de arranque o dispositivos, entre otros. En la siguiente tabla se resumen las unidades más importantes y su ubicación en el sistema.

Tipo de unidad systemd	Propósito
<code>.service</code>	Define un servicio del sistema
<code>.target</code>	Conjunto de unidades de tipo service
<code>.mount</code>	Punto de montaje para un sistema de ficheros
<code>.timer</code>	Temporizador para ejecución programada de tareas
Diretorio	Descripción
<code>/usr/lib/systemd/system/</code>	Unidades distribuidas con paquetes RPM
<code>/etc/systemd/system/</code>	Unidades creadas por el administrador

--	--

Tabla 1. Resumen de algunas unidades y ubicaciones importantes de systemd

Servicios

Ejercicio 1. En primer lugar, determinar las servicios (*unit* de tipo `.service`) disponibles en el sistema con el comando `systemctl list-unit-files --type service`. Además se pueden ver las unidades activas con la opción `list-units`. Escribir una línea de comandos que muestre el número de servicios que se arrancan automáticamente (`enabled`).

```
systemctl list-unit-files --type service
```

```
systemctl list-units --type service
```

Ejercicio 2. Cada *servicio* puede manipularse individualmente con el comando `systemctl`, el formato general es `systemctl <orden> <nombre_unidad>` (el nombre de la unidad se puede poner con o sin la extensión `.service`):

- Consultar su estado con las órdenes: `status`, `is-enabled` (se arrancará durante el arranque) e `is-active` (está en ejecución).
- Arrancar (`start`), parar (`stop`) o reiniciar (`restart`) un servicio.
- Programar el arranque automático del servicio (`enable`) o deshabilitar (`disable`). En ocasiones un servicio se inicia ya que es requerido como dependencia de ejecución por otra unidad. Es posible deshabilitar completamente con `mask` (`unmask` para volver a habilitarlo)

Consultar el estado de los servicios `firewalld` y `NetworkManager`.

```
systemctl status firewalld; systemctl status NetworkManager
```

Parar y reiniciar el servicio `sshd`.

```
systemctl stop sshd; systemctl restart sshd;
```

Eliminar el arranque automático del servicio `cups`.

```
systemctl disable cups
```

Niveles de Ejecución

Ejercicio 1. Las unidades de tipo `.target` agrupan un conjunto de servicios que deben activarse cuando la unidad de tipo `.target` se activa. Estas unidades definen un nivel de ejecución, entendido como los servicios asociados que estarán disponibles. La siguiente tabla resume los niveles disponibles:

Unidad <code>.target</code>	Propósito
<code>poweroff.target</code>	Apagado completo del sistema
<code>rescue.target</code>	Modo de recuperación, ejecuta una consola de mantenimiento
<code>multi-user.target</code>	Entorno multi usuario sin escritorio gráfico

<code>graphical.target</code>	Entorno multi-usuario con escritorio
<code>reboot.target</code>	Apagado y reinicio del sistema

Tabla 2. Niveles de ejecución por defecto con systemd

- Determinar el nivel actual del sistema (unidades tipo target) con el comando `systemctl list-units`
- Determinar las unidades de tipo target disponibles en el sistema con el comando `systemctl list-unit-files --type target`. ¿En qué nivel de los mostrados en la Tabla 2 arranca el sistema?

Ejercicio 2. Determinar el nivel por defecto con el comando `systemctl get-default`. Además modificarlo para arrancar en modo multi-usuario sin entorno gráfico con el comando `systemctl set-default`. Observar los enlaces simbólicos que se crean.

Ejercicio 3. El nivel de arranque se puede seleccionar también desde GRUB, por ejemplo para reparar el sistema. Simplemente añadir al final de la línea que especifica el kernel la opción `systemd.unit=<unit>.target`. Probar arrancar con la unidad `emergency.target`.

Ejercicio 4. El nivel de ejecución actual se puede cambiar con el comando `systemctl isolate <unit>.target`.

- Pasar al nivel multiusuario y observar los servicios que se arrancan.
`systemctl set-default multi-user.target`
- Devolver el nivel de ejecución por defecto al gráfico.
`systemctl set-default graphical.target`
- Finalmente reiniciar la máquina usando la unidad adecuada.

Ejercicio 5. El comando `systemctl` permite apagar, parar, suspender e hibernar el servidor con los comandos `poweroff`, `halt`, `suspend` y `hibernate`, respectivamente. Probar el efecto de los comandos anteriores.

Ejercicio 6. Asociados a systemd se han desarrollado multitud de servicios, desde servidores DHCP y DNS hasta gestores de red. En particular `systemd-logind` es el gestor de acceso al sistema (login). Consultar la página de manual `logind.conf(5)`, especialmente las opciones para la gestión de los eventos ACPI.

Para saber mas...

- Estudiar el formato de las unidades, ver `systemd-unit(5)`, consultar algunas de las disponibles en el sistema.
- System V define un proceso de arranque de servicios y niveles de ejecución diferente a systemd. Hay algunos gestores que siguen este modelo, estudiar por ejemplo OpenRC, <http://wiki.gentoo.org/wiki/Project:OpenRC>.