

## **INSTALACIÓN DEL SISTEMA OPERATIVO:**

- Pasos para crear una nueva máquina virtual:
  1. Abrir Virtual Box → Nueva
  2. Nombre y sistema operativo:
    - Nombre y sistema operativo:
      - Nombre: Practica2
      - Sistema operativo: Linux
      - Versión: Red-Hat (64-bit)
  3. Tamaño de memoria (RAM): 1024 MB (1 GB)
  4. Disco duro: Crear un disco duro virtual ahora.
  5. Tipo de archivo de disco duro: Utilizaremos VDI.
    - VDI → Virtual Disk Image, formato utilizado por oracle para virtualizar.
    - VHD → Virtual Hard Disk, formato utilizado por Virtual PC o Citrix.
    - VMDK → Virtual Machine Disk, formato utilizado por VMware.
  6. Almacenamiento en unidad de disco duro física: Usaremos la opción de espacio reservado dinámica-mente.
    - Reservado dinámica-mente: El disco duro virtual se amplía a medida que necesitamos mas espacio, hasta que lleguemos a un tamaño máximo. Esta opción ocupa menos espacio, pero puede ser más lenta ya que el hipervisor tiene que hacer traducciones.
    - Reservado estática-mente: Establecemos el tamaño del disco virtual directamente, es más costoso de crear, pero esta opción suele ser más rápida.
  7. Ubicación del archivo y tamaño: 8 GB
  8. Abrir la configuración de la máquina virtual que acabamos de crear.

9. Abrir la opción de almacenamiento:

- Dispositivos de almacenamiento:
  - Controlador IDE: Es un controlador hardware que permite la realización de conexiones con dispositivos electrónicos como discos duros, unidades ópticas, disqueteras y algunos puertos.
  - Controlador SATA: Es otro estándar más moderno de controlador, es más rápido que IDE. Introdujo la novedad de que los dispositivos podían ser añadidos o quitados cuando el sistema se está ejecutando. La interfaz principal de esta tecnología es llamada Advanced Host Controller Interface (AHCI).
- Usaremos IDE para seleccionar la ISO del sistema CentOS. Así cuando la máquina arranque cargará desde una supuesta unidad de disco óptico y cargará el sistema operativo desde ella.

10. Seleccionar en el menú la máquina virtual Practica2 y seleccionar la opción de Iniciar. La máquina virtual arrancará y mostrará la interfaz GRUB, el cargador de arranque unificado. GRUB puede cargar cualquier cualquier sistema operativo que se adapte a la especificación de multiarranque.

11. Cuando se muestre el menú de GRUB, seleccionamos la opción de instalación gráfica del sistema.

12. Idioma (idioma por defecto del sistema operativo) → English.

13. Menú de instalación:

- LOCALIZATION:
  - Date & Time: Europe ; Madrid
  - La opción de Network Time sirve para calibrar la hora del sistema con la de otros servidores distribuidos por el mundo, esta opción requiere conexión a Internet.
  - Keyboard: Hay que añadir la configuración para el teclado en español y ponerla la primera en la lista. En este menú también podemos ver el mapa del teclado.
  - Lenguaje Support: Añadir el idioma Español de España.

- SYSTEM:
  - Instalation destination:
    - En “Other Storage Options”, “Partitioning” seleccionar la opción “I will configure partitioning”.
    - En el siguiente menú añadiremos las distintas particiones del sistema operativo. Usar la opción “Add new mount point”:
      - /boot ; 512 MB → Partición de arranque.
        - Será una partición estándar con un sistema de ficheros xfs.
      - swap / 1024 MB (1 GB) → Partición de intercambio.
      - / (root) → Raíz del sistema operativo. Con “” le asignamos todo el espacio restante.
    - Security Policy: “Apply Security Policy” → OFF.
  - Software:
    - Software selection: Base Environment → Miniall install.  
Add-Ons → Development tools.

14. Begin Installation.

15. Establecer la contraseña del root (administrador) y crear un primer usuario:

- Contraseña root: cursoasr
- Cuenta y contraseña de usuario común: cursoasr

16. Reboot.

## **GRUB, LA SHELL DE ARRANQUE:**

### **Ejercicio 1:**

set root='hd0,msdos1' (hd0,0): Este comando indica que el disco duro 0 va a ser el disco duro primario y que de este disco se procederá al montaje de la primera partición del disco.

insmod <FileName> : Carga un módulo del kernel.

initrd16: Proporciona un proceso de arranque en dos etapas. Primero configura un sistema de ficheros raíz temporal en un disco RAM que contenga los módulos que se necesitan añadir al kernel, luego los carga y monta el sistema de ficheros real. El RAM DISK que contiene el sistema de ficheros temporal es el fichero de dispositivo especial “/dev/initrd”.

linux16: Carga una imagen del kernel de linux. Cualquier texto que le siga al fichero donde está el kernel, le es pasado a este como un lote de comandos a ejecutar.

### **Ejercicio 2:**

```
linux16 ... init=/bin/bash
```

No están activados los servicios poweroff ni reboot, hay que apagar la máquina cerrándola con el aspa de la ventana.

### **Ejercicio 3:**

Lo siguiente funciona si no instalamos mas sistemas operativos:

```
cd /boot/grub2
```

```
gedit grub.cfg & → timeout=10
```

```
reboot
```

Cambiar la variable GRUB\_TIMEOUT de /etc/default/grub, esto hará que los cambios sean permanentes aunque instalemos más sistemas operativos:

```
cd /etc/default/grub
```

```
gedit grub & → GRUB_TIMEOUT=10
```

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

```
reboot
```

### **Ejercicio 4:**

gedit /etc/grub.d/10\_linux & → Quitar de CLASS el --unrestricted para que nos pregunte la clave de usuario al arrancar.

gedit /etc/grub.d/01\_users & → Añadir antes del EOF lo siguiente:

```
set superusers="cursoar"  
password cursoar cursoar
```

“o”

```
set superusers="cursoasr"
password_pbkdf2 cursoasr
grub.pbkdf2.sha512.10000.E2D3B9CF811832852797D31BBBEFDB52A89C7C2D5
F47E4BAFE6BA228F42E5AE47A0FA690047FF9D352B993C4871F531C3309EB9
D9287952B4C06AE87CD65FA7F.DBC2CBDC558C5AD84ACBC6ED69B8DF770
AC6E9B5A1C9A0E49899FB2C723D138427376F50C015131B94DD7069AEC7F0
FC33C63E22F8EFDD38B1EC38A7E8076918
```

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

### **Ejercicio 5:**

dmesg | less → Muestra todos los mensajes generados por el kernel.

### **Ejercicio 6:**

Copia el primer bloque de 512 bytes del disco en el archivo copia.

```
dd if=/dev/sda of="$PWD/copia" bs=512 count=1
```

Machaca con ceros el MBR

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

Comprobar que no se pudo leer la tabla de particiones con fdisk -l

Restaura el MBR con la copia de seguridad

```
dd if="$PWD/copia" of=/dev/sda bs=512 count=1
```

## **SERVICIOS:**

### **Ejercicio 1:**

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

```
systemctl list-units
```

```
systemctl list-units | wc -l
```

## **Ejercicio 2:**

systemctl status ksm

systemctl is-enabled ksm

systemctl is-active ksm

systemctl status firewalld

systemctl stop NetworkManager

systemctl status NetworkManager

systemctl start NetworkManager

systemctl disable cuts

## **NIVELES DE EJECUCIÓN:**

### **Ejercicio 1:**

systemctl list-units | grep "target "

|                          |                      |
|--------------------------|----------------------|
| basic.target             | loaded active active |
| Basic System             |                      |
| cryptsetup.target        | loaded active active |
| Encrypted Volumes        |                      |
| getty.target             | loaded active active |
| Login Prompts            |                      |
| graphical.target         | loaded active active |
| Graphical Interface      |                      |
| local-fs-pre.target      | loaded active active |
| Local File Systems (Pre) |                      |
| local-fs.target          | loaded active active |
| Local File Systems       |                      |
| multi-user.target        | loaded active active |
| Multi-User System        |                      |
| network-online.target    | loaded active active |
| Network is Online        |                      |
| network.target           | loaded active active |
| Network                  |                      |
| nfs-client.target        | loaded active active |
| NFS client services      |                      |

|                             |                      |
|-----------------------------|----------------------|
| nss-user-lookup.target      | loaded active active |
| User and Group Name Lookups |                      |
| paths.target                | loaded active active |
| Paths                       |                      |
| remote-fs-pre.target        | loaded active active |
| Remote File Systems (Pre)   |                      |
| remote-fs.target            | loaded active active |
| Remote File Systems         |                      |
| slices.target               | loaded active active |
| Slices                      |                      |
| sockets.target              | loaded active active |
| Sockets                     |                      |
| sound.target                | loaded active active |
| Sound Card                  |                      |
| swap.target                 | loaded active active |
| Swap                        |                      |
| sysinit.target              | loaded active active |
| System Initialization       |                      |
| timers.target               | loaded active active |
| Timers                      |                      |

systemctl list-unit-files --type target | grep enabled

|                   |         |
|-------------------|---------|
| default.target    | enabled |
| graphical.target  | enabled |
| nfs-client.target | enabled |
| remote-fs.target  | enabled |

## **Ejercicio 2:**

systemctl get-default → graphical.target

systemctl set-default multi-user.target

Removed symlink /etc/systemd/system/default.target.

Created symlink from /etc/systemd/system/default.target to /usr/lib/systemd/system/multi-user.target.

## **Ejercicio 3:**

linux16 ... systemd.unit=emergency.target

#### **Ejercicio 4:**

systemctl isolate multi-user.target

systemctl list-units

systemctl isolate graphical.target

systemctl reboot

#### **Ejercicio 5:**

systemctl poweroff

systemctl halt

systemctl suspend

systemctl hibernate

#### **Ejercicio 6:**

man logind.conf