

# Sistemas informáticos

Jesús Beas Arco



Contenidos digitales  
[www.sintesis.com](http://www.sintesis.com)

EDITORIAL  
SINTESIS

INFORMÁTICA Y COMUNICACIONES  
Módulo Transversal

# Sistemas informáticos

Jesús Beas Arco



Contenidos digitales  
[www.sintesis.com](http://www.sintesis.com)



EDITORIAL  
SÍNTESIS



# Sistemas informáticos

Consulte nuestra página web: [www.sintesis.com](http://www.sintesis.com)  
En ella encontrará el catálogo completo y comentado



Queda prohibida, salvo excepción prevista en la ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra sin contar con autorización de los titulares de la propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (arts. 270 y sigs. Código Penal). El Centro Español de Derechos Reprográficos (www.cedro.org) vela por el respeto de los citados derechos.

# Sistemas informáticos

Jesús Beas Arco



**ASESOR EDITORIAL:**

Juan Carlos Moreno Pérez

© Jesús Beas Arco

© EDITORIAL SÍNTESIS, S. A.  
Vallehermoso, 34. 28015 Madrid  
Teléfono 91 593 20 98  
[www.sintesis.com](http://www.sintesis.com)

ISBN: 978-84-135750-3-2

Impreso en España - Printed in Spain

Reservados todos los derechos. Está prohibido, bajo las sanciones penales y el resarcimiento civil previstos en las leyes, reproducir, registrar o transmitir esta publicación, íntegra o parcialmente, por cualquier sistema de recuperación y por cualquier medio, sea mecánico, electrónico, magnético, electroóptico, por fotocopia o por cualquier otro, sin la autorización previa por escrito de Editorial Síntesis, S. A.



**PRESENTACIÓN** ..... 11

<b>1. FUNDAMENTOS DE LOS SISTEMAS INFORMÁTICOS Y LAS MÁQUINAS VIRTUALES</b> .....	13
Objetivos .....	13
Mapa conceptual .....	14
Glosario .....	14
1.1. Introducción .....	15
1.2. Arquitectura de un sistema informático. Modelos .....	15
1.3. Componentes hardware de un sistema informático .....	17
1.3.1. Microprocesador .....	17
1.3.2. Memoria principal .....	19
1.3.3. Placa base .....	21
1.3.4. Dispositivos de almacenamiento secundario .....	28
1.3.5. Fuente de alimentación .....	30
1.3.6. Periféricos .....	31
1.4. Controladores de dispositivos. Instalación de drivers .....	32
1.4.1. Administración de dispositivos en Microsoft Windows .....	33
1.4.2. Administración de dispositivos en Ubuntu Desktop .....	33
1.5. Componentes software de un sistema informático .....	34
1.5.1. Tipos de software .....	35
1.5.2. El sistema operativo .....	35
1.6. Proceso de arranque de un sistema informático. POST .....	36
1.7. Máquinas virtuales .....	37
1.7.1. Concepto y usos .....	38
1.7.2. Software de virtualización .....	38

ÍNDICE

<b>1.8. Oracle VM VirtualBox .....</b>	40
1.8.1. Proceso de instalación de Oracle VM VirtualBox .....	40
1.8.2. Entorno de Oracle VM VirtualBox .....	41
1.8.3. Creación de una máquina virtual en Oracle VM VirtualBox .....	41
1.8.4. Creación de instantáneas .....	44
<b>1.9. Normas de seguridad y prevención de riesgos laborales .....</b>	44
Resumen .....	47
Ejercicios propuestos .....	48
Actividades de autoevaluación .....	50
<b>2. SISTEMAS OPERATIVOS. INTRODUCCIÓN .....</b>	53
Objetivos .....	53
Mapa conceptual .....	54
Glosario .....	54
2.1. Introducción .....	55
2.2. Funciones y características .....	55
2.3. Tipos de sistemas operativos .....	58
2.4. Arquitecturas de los sistemas operativos .....	61
2.4.1. Sistemas con capas o anillos .....	61
2.4.2. Sistemas monolíticos .....	63
2.4.3. Microkernel .....	63
2.4.4. Kernel híbrido .....	64
2.4.5. Arquitecturas de sistemas operativos actuales .....	64
2.5. Versiones de los sistemas operativos más utilizados .....	65
2.5.1. Sistemas operativos de Microsoft .....	65
2.5.2. Sistemas operativos GNU/Linux .....	65
2.5.3. Sistemas operativos de Apple .....	66
2.6. Instalación de un sistema operativo .....	67
2.6.1. Requisitos .....	67
2.6.2. Planificación y consideraciones previas .....	68
2.6.3. Proceso de instalación de Ubuntu Desktop en Oracle VM VirtualBox .....	69
2.6.4. Proceso de instalación de Microsoft Windows 10 Pro en Oracle VM VirtualBox .....	74
2.7. Instalaciones desatendidas .....	76
2.7.1. Instalación desatendida de Windows 10 .....	77
2.7.2. Instalación desatendida de Ubuntu .....	78
2.8. Proceso de arranque del sistema operativo. Gestores de arranque .....	78
2.8.1. Conceptos previos: esquemas de particiones .....	79
2.8.2. Gestor de arranque de Windows .....	82
2.8.3. Gestor de arranque de Linux .....	83
2.9. Actualización del sistema operativo .....	84
2.9.1. Administración de actualizaciones en Windows .....	84
2.9.2. Administración de actualizaciones en Ubuntu Desktop .....	85
2.10. Identificación, instalación y desinstalación de aplicaciones .....	85
2.10.1. Aplicaciones y características de Windows .....	86
2.10.2. Software de Ubuntu .....	86
Resumen .....	87
Ejercicios propuestos .....	88
Actividades de autoevaluación .....	89

<b>3. SISTEMAS OPERATIVOS. GESTIÓN DE ARCHIVOS Y ALMACENAMIENTO</b>	91
Objetivos	91
Mapa conceptual	92
Glosario	92
3.1. Introducción	93
3.2. Sistemas de archivos	93
3.2.1. FAT (File Allocation Table)	95
3.2.2. exFAT	95
3.2.3. NTFS	95
3.2.4. APFS	96
3.2.5. ext4 (Fourth extended file system)	96
3.3. Estructura de directorios en Linux y Microsoft Windows	98
3.3.1. Estructura de directorios en GNU/Linux	99
3.3.2. Estructura de directorios en Microsoft Windows	100
3.4. Gestión de archivos por línea de comandos en Linux	101
3.4.1. Tipos de ficheros	103
3.4.2. Eliminación de ficheros	107
3.4.3. Creación y eliminación de directorios	107
3.4.4. Copia de archivos	108
3.4.5. Renombrado o movimiento de archivos	108
3.4.6. Impresión de archivos	109
3.4.7. Cuenteo de un fichero	110
3.4.8. Ordenación de un fichero	110
3.4.9. Entrada y salidas estándar. Redirecciones	111
3.4.10. Procesamiento de textos	114
3.5. Gestión de archivos por interfaz gráfica en Microsoft Windows	117
3.6. Gestión de almacenamiento por línea de comandos en Linux	118
3.6.1. Montaje y desmontaje	119
3.6.2. Particionar	124
3.6.3. Formatear	127
3.6.4. Desfragmentación	128
3.6.5. Chequeo	130
3.6.6. RAID	132
3.7. Gestión de almacenamiento por interfaz gráfica en Microsoft Windows	137
3.8. Búsqueda de información por línea de comandos en Linux	141
3.8.1. Criterios de búsqueda	141
3.9. Búsqueda de información por interfaz gráfica en Microsoft Windows	144
Resumen	145
Ejercicios propuestos	146
Actividades de autoevaluación	148
<b>4. SISTEMAS OPERATIVOS. GESTIÓN DE USUARIOS Y PROCESOS</b>	151
Objetivos	151
Mapa conceptual	152
Glosario	152
4.1. Introducción	153
4.2. Gestión de usuarios por línea de comandos en Linux	153
4.2.1. Configuración de usuarios y grupos	154
4.2.2. Comandos de gestión de usuarios	157

4.2.3. Usuarios y grupos predeterminados .....	159
4.2.4. Seguridad de cuentas de usuarios y contraseñas .....	161
4.2.5. Acceso a recursos y permisos locales .....	165
4.2.6. Modificación de permisos .....	168
4.2.7. Permisos por defecto .....	170
4.2.8. Configuración de perfiles .....	172
4.3. Gestión de usuarios por interfaz gráfica en Windows .....	175
4.4. Gestión de procesos por línea de comandos en Linux	
4.4.1. Procesos y servicios .....	176
4.4.2. Identificación y administración .....	179
4.5. Gestión de procesos por interfaz gráfica en Windows .....	184
4.6. Automatización de tareas en Linux .....	185
4.7. Monitorización y gestión del sistema. Evaluación de prestaciones .....	187
4.8. Aplicaciones para el mantenimiento y optimización del sistema .....	188
Resumen .....	189
Ejercicios propuestos .....	190
Actividades de autoevaluación .....	192
<b>5. SISTEMAS INFORMÁTICOS EN RED. CONFIGURACIÓN Y EXPLOTACIÓN .....</b>	<b>195</b>
Objetivos .....	195
Mapa conceptual .....	196
Glosario .....	196
5.1. Introducción .....	197
5.2. Protocolos principales de red .....	198
5.2.1. Protocolo Ethernet .....	200
5.2.2. Protocolo Wi-Fi .....	201
5.2.3. Protocolo IPv4 e IPv6 .....	201
5.2.4. Protocolo TCP y UDP .....	204
5.3. Configuración del protocolo TCP/IP .....	205
5.3.1. Estática .....	205
5.3.2. Dinámica .....	206
5.4. Interconexión de redes. Componentes .....	206
5.4.1. Switch .....	207
5.4.2. Router. Tablas de enrutamiento .....	207
5.4.3. Topología física y lógica. Mapas .....	208
5.4.4. Dominios de colisión y difusión .....	211
5.5. Tipos de redes .....	211
5.6. Acceso a redes WAN. Tecnologías .....	212
5.6.1. Conexiones WAN privadas .....	212
5.6.2. Conexiones WAN públicas .....	213
5.7. Redes cableadas .....	214
5.7.1. Tipos y características .....	214
5.7.2. Dispositivos de interconexión .....	218
5.7.3. Adaptadores .....	218
5.8. Redes inalámbricas .....	219
5.8.1. Tipos y características .....	219
5.8.2. Dispositivos de interconexión .....	221
5.8.3. Adaptadores .....	222
5.9. Ficheros de configuración de red .....	223
5.10. Monitorización y verificación de una red mediante comandos .....	225

5.10.1. Gestión de puertos .....	298
5.11. Resolución de problemas .....	299
5.12. Seguridad en las comunicaciones .....	299
5.12.1. Políticas de seguridad .....	299
5.12.2. Tipos de ataques .....	299
5.12.3. Mecanismos de seguridad en las comunicaciones .....	299
Resumen .....	299
Ejercicios propuestos .....	299
Actividades de autoevaluación .....	299
<b>6. GESTIÓN DE RECURSOS EN RED DE UN SISTEMA INFORMÁTICO</b> .....	299
Objetivos .....	299
Mapa conceptual .....	299
Glosario .....	299
6.1. Introducción .....	299
6.2. Permisos .....	299
6.2.1. Permisos de red y locales .....	299
6.2.2. Compartir archivos o carpetas .....	299
6.2.3. Herencia .....	299
6.2.4. ACL .....	299
6.3. Derechos de usuarios .....	299
6.3.1. Directivas de seguridad. Objetos y ámbito de directivas .....	299
6.3.2. Plantillas .....	299
6.4. Requisitos de seguridad del sistema y de los datos. Seguridad a nivel de usuarios y de equipos .....	299
6.5. Servidores .....	299
6.5.1. Servidor de ficheros .....	299
6.5.2. Servidor de impresión .....	299
6.5.3. Servidor de aplicaciones .....	299
6.6. Conexión remota. Herramientas .....	299
6.7. Herramientas de seguridad .....	299
6.7.1. Cifrado .....	299
6.7.2. Administración y análisis .....	299
6.7.3. Cortafuegos .....	299
6.7.4. Sistemas de detección de intrusión .....	299
6.7.5. OpenSSH .....	299
Resumen .....	299
Ejercicios propuestos .....	299
Actividades de autoevaluación .....	299
<b>7. APLICACIONES INFORMÁTICAS</b> .....	299
Objetivos .....	299
Mapa conceptual .....	299
Glosario .....	299
7.1. Introducción .....	299
7.2. Tipos de software .....	299
7.2.1. Clasificación por licencia .....	299
7.2.2. Clasificación por propósito .....	299

7.3. Requisitos mínimos y recomendados .....	281
7.4. Herramientas ofimáticas .....	282
7.4.1. Procesadores de texto .....	283
7.4.2. Hojas de cálculo .....	284
7.4.3. Software de presentación .....	284
7.4.4. Sistemas gestores de bases de datos .....	285
7.5. Herramientas de Internet .....	286
7.5.1. Correo electrónico .....	286
7.5.2. Mensajería instantánea .....	288
7.5.3. Transferencia de ficheros .....	289
7.5.4. Computación y almacenamiento en la nube .....	291
7.6. Software antimalware .....	292
7.7. Clonación y copias de seguridad .....	294
7.7.1. Clonaciones .....	294
7.7.2. Copias de seguridad .....	295
7.7.3. Recuperación de datos .....	300
7.8. Documentación técnica .....	301
7.8.1. Elaboración de documentación .....	301
7.8.2. Métodos de búsqueda de documentación técnica en Internet .....	302
Resumen .....	303
Ejercicios propuestos .....	304
Actividades de autoevaluación .....	305

## CONTENIDOS DIGITALES



- 1.1. Carpetas compartidas en Oracle VM VirtualBox
- 2.1. El origen de los sistemas operativos
- 2.2. Sistema operativo MINIX
- 2.3. El origen de las distribuciones GNU/Linux
- 2.4. Entorno de la ventana de la máquina virtual
- 2.5. Estándar MBR
- 3.1. Esquemas de participación con sistemas de archivos FAT, NTFS y ext4
- 3.2. Administrador de medios virtuales de Oracle VM VirtualBox
- 5.1. Configuración de red en distribuciones anteriores a Ubuntu 17.10
- 5.2. Sintaxis de ifconfig en GNU/Linux
- 5.3. Ejemplos de nmap
- 5.4. Modos de red de Oracle VM VirtualBox



# Presentación

Este libro desarrolla los contenidos del módulo profesional de Sistemas Informáticos de los títulos de Técnico Superior en Desarrollo de Aplicaciones Web y Técnico Superior en Desarrollo de Aplicaciones Multiplataforma de la familia profesional de Informática y Comunicaciones, según el Real Decreto 686/2010, de 20 de mayo, y el Real Decreto 450/2010, de 16 de abril respectivamente.

El libro va dirigido a aquellos perfiles profesionales que ejercen su actividad en empresas o entidades públicas o privadas, tanto por cuenta ajena como propia, desempeñando su trabajo en el área de desarrollo de aplicaciones informáticas como programadores de aplicaciones web o multiplataforma, así como administradores de sistemas informáticos a diferente nivel.

Los sistemas informáticos actuales deben ser administrados y explotados convenientemente, para lo que se necesita un enfoque teórico y práctico adecuado que responda a las necesidades del perfil. Con este objetivo, se desarrollan los siguientes bloques de contenidos:

- Explotación de sistemas microinformáticos.
- Instalación de sistemas operativos.
- Gestión de la información.
- Configuración de sistemas operativos.
- Conexión de sistemas en red.
- Gestión de recursos en una red.
- Explotación de aplicaciones informáticas de propósito general.

El contenido y estructura de los capítulos han sido elaborados de forma clara, incidiendo en aspectos teórico-prácticos variados y reales, a la vez que interrelacionados entre sí, con el objetivo de adecuarse a acciones profesionales. Para ello, se trabajan sistemas operativos libres y propietarios de manera paralela a través de sus interfaces gráfica y textual, pero haciendo hincapié en esta última por su gran potencia y flexibilidad.

PRESENTACIÓN

Con un enfoque muy práctico, la obra incluye numerosos ejemplos y ejercicios resueltos, así como actividades y ejercicios propuestos para afianzar los contenidos. Se aporta también una serie de recursos digitales, disponibles en la plataforma de Editorial Síntesis ([www.sintesis.com](http://www.sintesis.com)).

Gracias a este módulo profesional, el estudiante será capaz de configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos; así como de aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.

El módulo profesional *0483. Sistemas Informáticos* desarrolla la unidad de competencia *UC0223\_3: Configurar y explotar sistemas informáticos* del Catálogo Nacional de Cualificaciones Profesionales. Además, dicha unidad de competencia forma parte de la configuración de los siguientes certificados de profesionalidad:

- ✓ Programación en lenguajes estructurados de aplicaciones de gestión (IFCD0111).
- ✓ Programación con lenguajes orientados a objetos y bases de datos relacionales (IFCD0112).
- ✓ Administración de bases de datos (IFCT0310).

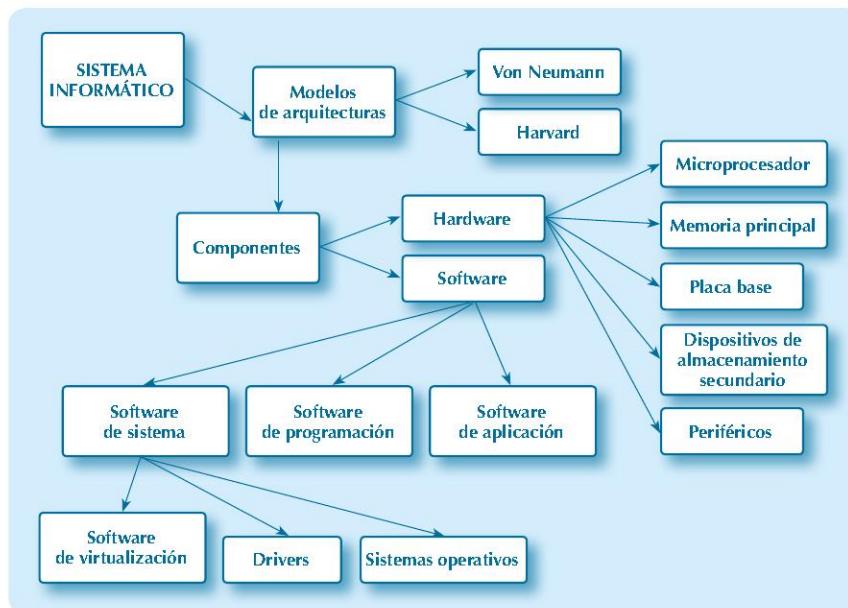
# 1

## Fundamentos de los sistemas informáticos y las máquinas virtuales

### Objetivos

- ✓ Aprender cuáles son y cómo actúan las unidades funcionales de un sistema informático.
- ✓ Conocer las funciones de los principales componentes físicos de un sistema informático.
- ✓ Reconocer los componentes físicos de un sistema informático y mecanismos de interconexión.
- ✓ Verificar el proceso de puesta en marcha de un equipo.
- ✓ Clasificar, instalar y configurar diferentes dispositivos periféricos.
- ✓ Conocer el concepto de máquina virtual y sus ventajas.
- ✓ Operar las máquinas respetando las normas de seguridad y las recomendaciones ergonómicas.

### Mapa conceptual



### Glosario

**Arquitectura.** Término genérico que hace referencia al diseño de sistemas informáticos basándose en el tamaño de los registros del procesador de 64 o 32 bits.

**Chipset.** Circuito integrado y encapsulado principal de la placa base que conecta y gestiona los componentes de la misma.

**CPU.** Parte del microprocesador que se encarga de la ejecución de las instrucciones y que contiene principalmente: unidad de control, unidad aritmético-lógica y registros.

**Hardware.** Parte física de un sistema informático.

**Máquina virtual.** Computadora no real, configurada e instalada en un sistema informático.

**Memoria.** Conjunto de medios o componentes de almacenamiento encargados de alojar de manera temporal o permanente instrucciones o datos.

**Microprocesador.** Circuito integrado y encapsulado que constituye el centro neurálgico de procesamiento del sistema que incluye una o varias unidades centrales de procesamiento.

**Motherboard o placa base.** Circuito impreso principal de todo sistema informático que conecta los componentes hardware del sistema.

**Sistema informático.** Máquina que acepta unos datos de entrada, los procesa y genera unos resultados.

**Sistema operativo.** Software principal de un sistema informático que actúa de interfaz con el usuario y gestiona los recursos hardware y software.

**Software.** Parte lógica o intangible del sistema informático.

**Virtualización.** Proceso de abstracción de un sistema informático.

## 1.1. Introducción

En este capítulo se ha tratado de componer una visión global de los sistemas informáticos en general. Los conceptos que se presentan han dado lugar a una gran literatura al respecto que, además, se actualiza a diario y, por tanto, no se pretende profundizar en ellos.

En primer lugar, se van a estudiar los componentes fundamentales de todo sistema informático. Los dos ejes básicos de un sistema informático son el hardware (sus componentes físicos) y el software (los programas o las aplicaciones).

Comenzaremos por los modelos que originaron la computación actual y, posteriormente, nos centraremos en los componentes físicos que encontramos en cualquier equipo hoy en día. Para un mejor entendimiento de dichos componentes nos basaremos en los sistemas de sobremesa, ya que sus características se hacen extensibles a cualquier sistema informático (tabletas, portátiles o smartphones).

Analizaremos los tipos de software y su relación con el hardware, distinguiendo sus tipos y ahondando en el software más importante de un equipo: el sistema operativo.

Conociendo los distintos elementos del sistema informático y sus funciones, estaremos preparados para entender y verificar su proceso de arranque.

Más tarde introduciremos el concepto de máquina virtual, conociendo sus principales usos y aplicaciones más utilizadas para su desarrollo. Las máquinas virtuales nos van a permitir, a través de un software y durante todo el curso, realizar instalaciones de sistemas operativos, establecer multitud de configuraciones hardware y software, hacer pruebas, etc.

Por último, abordaremos las principales normas de seguridad y prevención en materia de riesgos laborales, con especial atención a las recomendaciones ergonómicas que cualquier trabajador debe cuidar.

## 1.2. Arquitectura de un sistema informático. Modelos

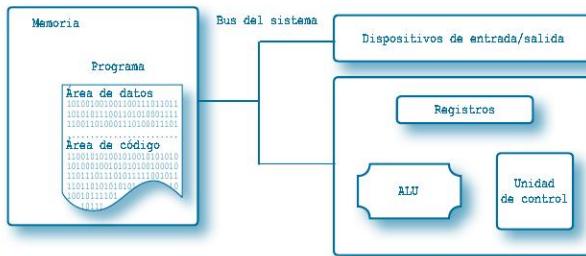
Se entiende por *sistema informático* una máquina que acepta unos datos de entrada, los procesa y genera unos resultados. En todo sistema informático se distinguen dos partes claramente diferenciadas y necesarias:

- Hardware: conjunto de elementos físicamente tangibles.
  - Software: parte intangible formada por instrucciones o datos que pueden ser interpretados o procesados.

Los sistemas informáticos actuales, ya sean computadores personales, grandes supercomputadores o smartphones, tienen como base las arquitecturas de *Von Neumann* y *Harvard*.

El modelo de Von Neumann consta de las siguientes partes:

- a) Unidad de procesamiento: se encarga de la ejecución e interpretación de instrucciones y datos formada por unidad aritmético lógica (ALU), unidad de control y registros de almacenamiento.
  - b) Memoria: almacena instrucciones y datos.
  - c) Dispositivos de entrada/salida: elementos que actúan de interfaz con el resto de partes.



**Figura 1.1**  
Diagrama del modelo de Von Neumann.

En el modelo de Von Neumann las diferentes unidades funcionales se interconectan mediante buses de comunicación o buses del sistema. Estos pueden ser:

- ✓ Buses de instrucciones: líneas de comunicación que transmiten instrucciones.
  - ✓ Buses de datos: líneas de comunicación que transmiten únicamente datos.
  - ✓ Buses de direcciones: líneas de comunicación empleadas para acceder a las distintas memorias, indicando una dirección de acceso de lectura o escritura.

El modelo Harvard mejoró la arquitectura de Von Neumann, ya que el acceso a datos e instrucciones se realiza simultáneamente al encontrarse en caminos distintos. Sin embargo, en el modelo de Von Neumann, la memoria almacena instrucciones (código) y datos de manera conjunta, como se puede ver en la figura 1.1.

Basándose en los modelos de Von Neumann y Harvard, en los sistemas informáticos actuales se encuentran las unidades funcionales que veremos en los apartados 1.3 y 1.4.

## Actividad propuesta 1.1



Busca en Internet los primeros computadores con programas almacenados y su relación con el modelo de Von Neumann.

### 1.3. Componentes hardware de un sistema informático

La parte física de un equipo o hardware consta de multitud de componentes. Los más importantes son: el microprocesador, la placa base, la memoria principal, los dispositivos de almacenamiento secundario, la batería o fuente de alimentación y los periféricos.

#### 1.3.1. Microprocesador

El microprocesador es un circuito integrado encapsulado de altísimo nivel de integración en los componentes que aloja. El microprocesador contiene una o más *unidades centrales de proceso* (CPU). Este es el centro neurálgico de procesamiento del sistema. Las partes más importantes de una CPU son las siguientes:

- a) *Unidad de control (UC)*: encargada del procesamiento, interpretación y ejecución de instrucciones y datos. Envía al resto de componentes señales de control, estado o situación para la correcta automatización de las diferentes funciones del sistema de manera sincronizada.
- b) *Unidad aritmético lógica (UAL)*: componente encargado de realizar cálculos aritméticos y lógicos.
- c) *Registros*: memorias temporales de poca capacidad y alta velocidad.

**Recurso web**

En este vídeo de Global Foundries puedes ver cómo se fabrica un microchip:



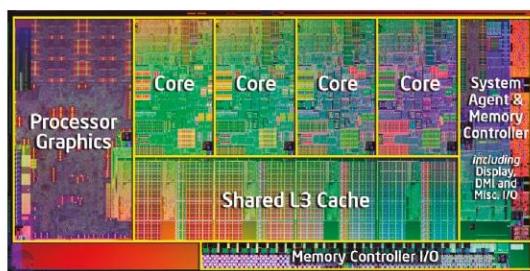
www

Los microprocesadores actuales alojan la CPU, además de otros muchos otros elementos. Detallamos los más importantes:

- *Núcleo*: estructura que aloja las unidades funcionales de una CPU. En la actualidad, es común que los microprocesadores contengan más de un núcleo. Cada núcleo es capaz de ejecutar una instrucción, sincronizándose con el resto para realizar varias tareas simultáneamente.
- *Memorias caché*: memorias temporales, extremadamente rápidas y cercanas al núcleo. Atendiendo a su cercanía con este, suelen encontrarse tres niveles:
  - L1 o de nivel 1: normalmente situada dentro de cada núcleo.
  - L2 o de nivel 2: suele estar situada fuera de los núcleos, pero compartida entre varios. Actúa de intermediaria de los niveles L1 y L3.
  - L3 o de nivel 3: suele estar situada fuera de los núcleos, pero compartida por todos ellos. Este nivel recibe o entrega instrucciones y datos a o desde los módulos de memoria.

- *Controlador de memoria:* la memoria RAM es gestionada por este componente.
- *Controlador gráfico:* hace referencia a la capacidad de computación de cálculo para gráficos. No todos los procesadores integran esta característica, puesto que las tarjetas gráficas dedicadas a este propósito poseen mayor rendimiento.

En la figura 1.2 se aprecia la estructura interna de un microprocesador donde se distinguen: procesador gráfico, cuatro núcleos, memoria cache L3 (por tanto, dispone de caché L2 y L1 no representadas), controlador de memoria y varios controladores de comunicación de entrada/salida.



**Figura 1.2**  
Estructura interna  
de un microprocesador.  
*Fuente:* <https://superuser.com>



#### PARA SABER MÁS

En computadores actuales, el modelo Harvard se aplica en la memoria caché cuando ésta separa instrucciones y datos.

En cuanto a las características más importantes de los procesadores, destacamos:

1. *Velocidad o frecuencia:* medida en gigahercios (GHz), hace referencia al número de ciclos que tienen que transcurrir para ejecutar una instrucción o parte de ella en cada CPU. A mayor frecuencia, mayor velocidad de procesamiento.
2. *Número de hilos:* los procesadores pueden ejecutar, al mismo tiempo, hilos de procesamiento, es decir, tareas como parte de un mismo proceso. Este concepto se refiere al número de hilos con los que los núcleos del procesador son capaces de trabajar en paralelo.
3. *Nivel de integración:* hace referencia a la medida de nanómetros (nm) empleados para la fabricación del procesador, aplicando técnicas litográficas. Cuanto menor sea esta cantidad, mayor nivel de integración tendrá al poder incluir en el mismo espacio mayor número de componentes.
4. *Consumo:* medido en vatios (W), depende del voltaje e intensidad que necesite el procesador.
5. *Potencia de disipación térmica (TDP):* a diferencia de la característica anterior, esta hace referencia a vatios térmicos, con objeto de buscar una solución de refrigeración al procesador. Los equipos móviles se distinguen claramente por su bajo TDP.



### Actividad propuesta 1.2

Accede a la página web de algunos de los mayores fabricantes de procesadores, como Intel o AMD. Sobre un procesador dado al azar, analiza las características estudiadas: número de núcleos, memoria caché (tamaños), frecuencia, nivel de integración, consumo, etc.

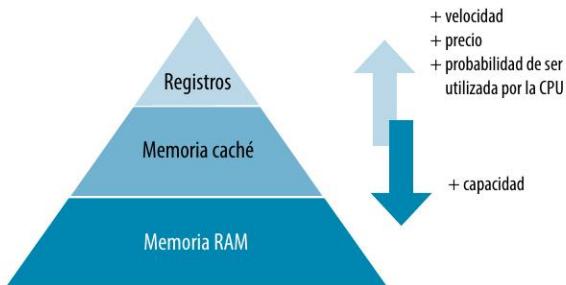
Posteriormente, analiza dos procesadores, pero de diferentes propósitos, como, por ejemplo, para equipos portátiles y para servidores. Analiza sus principales diferencias.

#### 1.3.2. Memoria principal

La memoria de almacenamiento principal se encuentra conectada a la CPU, a la cual abastece almacenando instrucciones o datos de forma temporal, es decir, cuando carece de energía, su contenido desaparece.

La memoria principal engloba varios tipos de memoria: registros, memoria caché y memoria RAM. La siguiente imagen representa la distribución y características de la memoria principal.

**Figura 1.3**  
Diagrama piramidal de la distribución y características de la memoria principal.



Por tanto, la memoria principal está constituida por:

- Registros*: estructuras de almacenamiento pertenecientes al núcleo de la CPU de muy poca capacidad, pero cuyo acceso y escritura es extremadamente rápido. La mayoría se encuentran en la UC y la UAL. El tamaño de los registros define la arquitectura, siendo de 32 bits o 64 bits.
- Memoria caché*: memoria intermedia entre los registros y la memoria RAM, que se encuentra en los núcleos o en el microprocesador. Cuanto mayor es su capacidad, mayor capacidad de cómputo tendrá el microprocesador, ya que disminuirán las veces que ésta tenga que recargarse accediendo a la memoria RAM y volcar nuevos datos o instrucciones. Como se ha comentado con anterioridad, suelen existir tres niveles (L1, L2 y L3), donde se alojan de manera compartida o separada las instrucciones y los datos. Es común que se encuentren separados en algún nivel para aumentar la velocidad de procesamiento, normalmente en L2 y L1.
- Memoria RAM*: memoria externa al microprocesador que se agrupa en forma de módulos de memoria instalados en la placa base. Sus principales características son:

- Capacidad: tamaño especificado en gigabytes (GB).
- Velocidad: frecuencia de trabajo interna de cada módulo. Se mide en gigahercios (GHz).
- Voltaje: tensión necesaria para su funcionamiento (V).
- Latencias: especifica los tiempos de acceso a los datos de los chips del módulo de memoria. Cuanto menor sean las latencias, más velocidad tendrá el módulo en localizar y disponer de los datos. Se mide en ciclos de reloj, por ejemplo: CL21.
- Número de canales de comunicación con el procesador: el número de canales entre la memoria y el procesador para transferir información simultáneamente. Los módulos deben estar desarrollados con tecnología multicanal. Para ello, es necesario emplear parejas o cuartetos de módulos, respectivamente. Esto hará que se incremente la velocidad de transferencia al trabajar el procesador en paralelo con varios módulos.
- Tipo de módulo: los chips de memoria se encapsulan en módulos DIMM o SO-DIMM, según sean para equipos de sobremesa o portátiles, respectivamente, con diferente dimensión.
- Tecnología: los módulos de memoria actuales emplean una tecnología de tipo SDRAM DDR4. Esto hace mención a que son memorias de acceso aleatorio dinámico (DRAM), empleando doble recarga en su versión 4 (DDR4). La figura 1.4 muestra una comparativa entre módulos DIMM DDR4 y DDR3. La muesca, situada entre los contactos metálicos, en la parte inferior de cada módulo, se encuentra en posiciones distintas para evitar errores en la colocación de los módulos sobre los zócalos de memoria. Los módulos de memoria con tecnología SDRAM-DDR4 ofrecen mejoras con respecto a sus predecesoras SDRAM-DDR3: menos voltaje, mayor frecuencia, aumentan la densidad de los chips de memoria y presentan un mayor ahorro energético.



**Figura 1.4**  
Módulos DIMM DDR4 y DDR3.



#### TOMA NOTA

Si debemos elegir entre capacidad de memoria RAM y su velocidad, lo primero es lo más recomendable, en general. Debemos cubrir un mínimo de capacidad según el sistema operativo y las aplicaciones que se van a ejecutar; a partir de ahí, debemos plantearnos si el aumento de frecuencia resulta rentable económico.

No obstante, la agilidad de un equipo no siempre se soluciona aumentando el tamaño de la memoria RAM, puesto que existen otros factores, como el almacenamiento secundario, que pueden lastrar su rendimiento al actuar como cuello de botella.



### Actividad propuesta 1.3

Accede a la página web de algún fabricante de memorias RAM: Kingston, Crucial o Corsair, y analiza las características estudiadas para un modelo específico. Posteriormente, compara dos modelos distintos de un mismo fabricante, analizando sus diferencias.

#### 1.3.3. Placa base

También llamada *motherboard*. La placa base es el circuito impreso principal de todo sistema informático, que conecta todos los componentes hardware directa o indirectamente. Se puede considerar la pieza fundamental (junto con la CPU), ya que determina la potencia de cálculo o procesamiento, la capacidad de expansión, el almacenamiento, el tipo de alimentación o el tipo de caja.

Las placas base se rigen por los llamados *factores de forma*. Estos son estándares contemplados a nivel mundial, que determinan, entre otros aspectos, las medidas de dicha placa base, la disposición y lugar donde se alojan sus componentes (zócalos, conectores, buses de expansión), la potencia, etc. Así, los diversos fabricantes de hardware pueden trabajar de manera independiente para unas especificaciones de placa base bien definidas.

Existen multitud de factores de forma, aunque los más utilizados son:

- ✓ *ATX*. Contemplan una disposición de sus componentes, que mejoran sustancialmente a sus antecesoras las AT y XT en cuanto a la refrigeración, principalmente. Es el más empleado actualmente. Presenta variantes como Micro-ATX o Mini-ATX, que reducen las dimensiones de las placas base y están orientadas a equipos menos potentes, reducido consumo y escasa capacidad de expansión.
- ✓ *Variantes ITX*. Orientados a equipos de muy bajo consumo y reducidas dimensiones. Prácticamente todos sus componentes están integrados en la misma placa base, consiguiendo reducir las dimensiones, pero afectando a su capacidad de expansión. Gracias a su poca potencia, pueden carecer de componentes activos de refrigeración (como ventiladores). Ejemplos: Mini-ITX, Nano-ITX y Pico-ITX.

Los factores de forma más comunes son ATX, Micro-ATX y Mini-ITX, que son compatibles entre ellos. No obstante, existen otros muchos factores de forma que se adecuan a las características o necesidades del sistema informático objeto de producción. También, multitud de fabricantes disponen de factores de forma propios (algunos abiertos) que permiten desarrollar sus propios productos.



### Actividad propuesta 1.4

Busca en Internet dos modelos distintos de placas base con factores de forma para los tipos ATX, Micro-ATX y Mini-ITX. Analiza sus dimensiones y su capacidad de expansión.

Los principales componentes que encontramos en una placa base son los siguientes.

### A) *Chipset*

Con este nombre se conoce al principal circuito integrado y encapsulado (microchip) en la placa base, que resulta fácilmente distinguible por su tamaño y localización. En las placas base actuales para equipos de sobremesa existen de muy variadas características, dependiendo de la potencia y prestaciones.

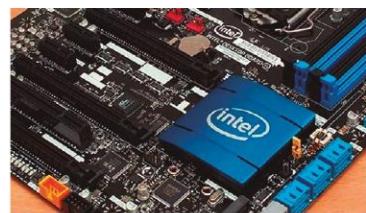
Su labor es la de gestionar todos los componentes de la placa base, dotándolos de sincronismo a través de diferentes buses. Por ello, directa o indirectamente, el chipset siempre interviene en cualquier operación. Tanto es así, que este determina el procesador que se puede instalar en la placa base, la memoria RAM, la cantidad de buses disponibles para ranuras de expansión, el arranque del sistema, la cantidad y tipos de conectores internos y externos, la capacidad de overclocking, etc.

Debido a la gran capacidad de integración de los chips, así como la reducción de consumo asociada a la movilidad de los sistemas actuales, hoy nos encontramos con un chipset cuyas funciones son:

- ✓ Coordinar la asociación entre los componentes de gran capacidad de transferencia de información o procesamiento, como el procesador, memoria o buses PCI Express de gráficos.
- ✓ Actuar de concentrador de componentes de entrada y salida, así como de dispositivos de baja velocidad.

En las placas base ATX, el chipset se sitúan al sureste del zócalo del procesador y son fácilmente distinguibles al ser chips grandes y disponer de un gran disipador.

Al trabajar con una placa base o antes de adquirirla, es importante acceder a la información del fabricante del chipset para conocer sus características. En el diagrama de la figura 1.6 se pueden analizar las comunicaciones con los distintos componentes. En este caso, vemos que este puede gestionar hasta 10 puertos USB 3.1, 6 puertos SATA 6Gbps, 24 lanes PCI Express 3.0; se comunica con procesadores Intel de octava generación (mediante un bus DMI 3.0) e integra multitud de tecnologías, como Intel Optane. Por otro lado, el procesador controla hasta 4 ranuras de memoria DIMM y 16 lanes PCI Express 3.0 para tarjetas gráficas.

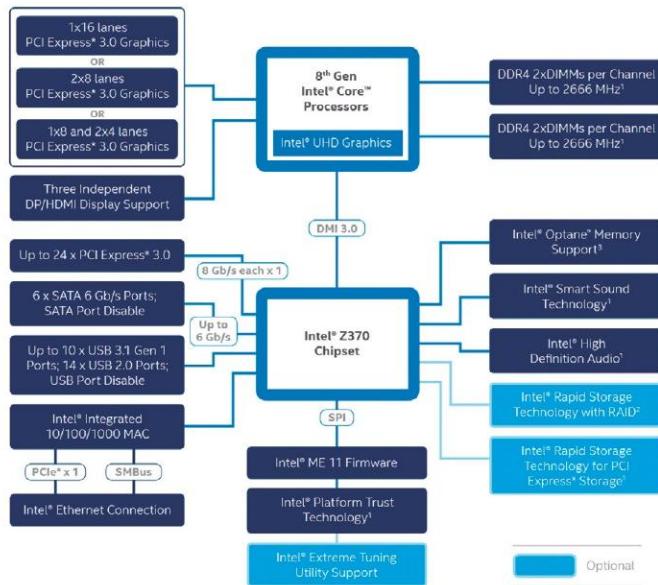


**Figura 1.5**  
Chipset cubierto por un disipador Intel.

#### Actividad propuesta 1.5



Accede a la página web de Intel, en la sección de chipsets. Sobre un chipset dado para un equipo de sobremesa o servidor, analiza sus características con especial atención a las limitaciones sobre los tipos de procesadores, memoria RAM, tipo y número de puertos, y buses de comunicación.



**Figura 1.6**  
Diagrama del chipset Intel Z370.  
Fuente: [www.intel.es](http://www.intel.es)

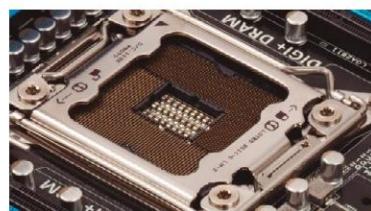
## B) Zócalo del microprocesador

El *socket* o *zócalo* del microprocesador es el lugar donde se instala este. Existen, principalmente, dos tipos:

- *ZIF o PGA (Pin Grid Array)*: consiste en una estructura de plástico con pequeños agujeros, donde se insertan las patillas del microprocesador. Este se coloca en el socket sin ejercer presión, ya que dispone de una palanca para encajarlo sin fuerza.



**Figura 1.7**  
Zócalo ZIF.



**Figura 1.8**  
Zócalo LGA.

- **LGA (Land Grid Array)**: dispone de una base con contactos que se comunican con la placa base, sobre la que cierra una estructura de metal con forma de ventana. El procesador dispone de contactos y no patillas, por lo que establece la comunicación por presión gracias a dicha estructura. La instalación en este socket es sencilla, con mucho menos riesgo de dañar el microprocesador. Permite mayor cantidad de contactos.

### C) Ranuras de memoria RAM

Las ranuras de memoria son espacios destinados a alojar los módulos de memoria RAM. Actualmente, los más utilizados en equipos de sobremesa son ranuras para módulos DIMM SDRAM-DDR4 con 288 pines. Todas las ranuras disponen de una marca para alinear el módulo correctamente, así como retenedores laterales para aumentar la sujeción de este a la placa base.

Las placas base disponen de tecnología de doble, triple o cuádruple canal (llamadas *Dual Channel*, *Triple Channel* o *Quad Channel*, respectivamente). De esta manera, se consigue acceder a varios módulos simultáneamente, mejorando la velocidad de acceso a la memoria RAM por parte del procesador. Esto gracias a duplicar, triplicar o cuaduplicar el canal de 64 bits del *single channel* por defecto.

Las placas base presentan las ranuras de memoria RAM asociadas con colores (parejas, tríos o cuartetos) para hacer uso de esta característica.



**Figura 1.9**  
Placa base ASUS con ranuras para módulos de memoria RAM con tecnología DDR4 y posible configuración en Dual Channel.

#### TEN EN CUENTA

- ✓ El fabricante del procesador, donde se encuentran los controladores de memoria, determinan las características que deben cumplir los módulos de memoria RAM para hacer efectiva la tecnología multicanal. No obstante, en caso de emplear distintos módulos, el controlador de memoria se ajusta a las velocidades, latencias o capacidades más bajas de todos ellos. Sin embargo, emplear módulos con las mismas características permite un mayor rendimiento, por lo que es recomendable adquirir kits de módulos de memoria preparados para ello.

### D) Ranuras de expansión

Las ranuras de expansión son los módulos encargados de alojar las tarjetas de expansión para ampliar las características del equipo. Según el ancho de banda y la velocidad de transmisión, encontramos varios tipos de buses de expansión, los cuales emplean diferentes ranuras. El bus más empleado

es el *PCI Express*, o *PCIe*, que se implementa hasta con 16 líneas (*lanes*) de datos. Cada línea dispone de un ancho de banda de 2 GB/s en su versión 4.0. Las ranuras de expansión PCI Express más comunes según sus lanes y su ancho de banda son:

- ✓ PCIe x1 v4.0: 2 GB/s.
- ✓ PCIe x4 v4.0: 16 GB/s.
- ✓ PCIe x16 v4.0: 32 GB/s.

Las distintas versiones de PCI Express son compatibles entre sí. Se emplean para alojar tarjetas gráficas, tarjetas de sonido, dispositivos de almacenamiento secundario, tarjetas adaptadoras de red, etc.

Aunque se destacan las ranuras de expansión PCI Express porque son las más ampliamente utilizadas por los fabricantes de placas base, existen otros tipos, como las antiguas PCI, ya en desuso.



**Figura 1.10**  
Placa base con ranuras  
PCIe x16, PCIe x1, PCIe x4 y  
PCIe x16, de arriba abajo.

La relación entre CPU, RAM y almacenamiento decide el rendimiento.  
Puedes saber más leyendo este artículo de Xataka.



## E) BIOS

También llamado *ROM BIOS*. Es un chip que se encuentra físicamente visible en la placa base y se encarga de varias tareas:

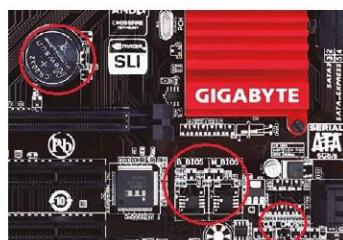
- Comprobar el sistema y lanzar su arranque.
- Realizar funciones básicas de entrada/salida con el sistema operativo funcionando.
- Configurar el equipo a través de una aplicación llamada *BIOS Setup Utility*.

Muchas placas base, por seguridad, disponen de dos BIOS. En caso de que una de ellas se corrompa, la otra se activa, impidiendo que el equipo deje de funcionar.

Además, la BIOS tiene asociada una memoria RAM-CMOS que almacena de manera temporal los datos de la configuración del sistema. Estos datos aparecen cuando accedemos al BIOS Setup Utility: fecha y hora del sistema, medios de arranque, periféricos, buses, overclocking, chipsets, etc.

Al ser la RAM-CMOS una memoria volátil, la placa base tiene una pila que la alimenta e impide que desaparezca su configuración. Si la pila pierde su carga, es necesario ajustar algunos de estos valores para que el sistema arranque correctamente.

Las placas base también disponen de unos pines o botones para resetear la memoria RAM-CMOS, si se desea volver a ajustar la configuración del sistema a la versión de fábrica. En la figura 1.11 se aprecia, de izquierda a derecha: la pila, el sistema de doble BIOS (Backup Bios y Main Bios) y los pines de reseteo de la memoria BIOS RAM-CMOS.



**Figura 1.11**  
Placa base Gigabyte GA-Z97X-SLI.

#### F) Conectores internos

Algunos de los conectores más importantes son:

- ✓ *Conector SATA*: empleado para la transferencia de datos entre el chipset y los dispositivos de almacenamiento secundario. Es el conector más ampliamente utilizado para conectar discos duros. Este conector sustituyó al antiguo IDE, que aún podemos encontrar en placas base antiguas, o en actuales por compatibilidad.
- ✓ *Conector M.2*: se usa especialmente para almacenamiento (discos duros SSD) o conectividad en equipos de reducidas dimensiones, aprovechando el espacio en la placa base. Este trabaja con buses SATA o PCI Express, empleando distintos conectores para cada uno de ellos.
- ✓ *Conectores de ventiladores*: los ventiladores que refrigeran el procesador, la caja o incluso algunos chipsets son alimentados a través de estos conectores.



**Figura 1.12**  
Conectores SATA III.



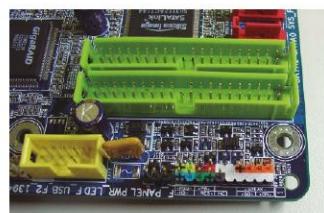
**Figura 1.13**  
Conectores M.2 para discos SSD y tarjeta de red Wi-Fi.



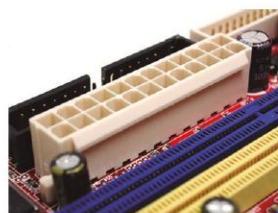
**Figura 1.14**  
Conectores para ventiladores de cuatro pines en la zona superior y uno de tres pines a la derecha.

- ✓ *Conectores USB*: encargados de conectar, a través de un cable, los conectores USB del frontal de la caja de los equipos.
- ✓ *Conectores del panel frontal*: la caja del equipo suele disponer de varios botones y luces led que se conectan a estos conectores para transmitir las acciones. Se suelen presentar por colores, detallando en la placa base la correspondencia con cada conector. Los más empleados son:
  - Botón de encendido.
  - Botón de reset.

- Led de encendido.
  - Led de uso de disco duro.
- ✓ *Conectores de alimentación:* nutren de energía eléctrica a la placa base y a todos sus componentes. Es habitual encontrar un conector de 20 o 24 pines que suministra alimentación a la placa base, y otro de 4 u 8 pines que alimenta específicamente al procesador (el cual se encuentra cercano a este).



**Figura 1.15**  
Conectores SATA (rojo), conectores IDE (verde), conector USB interno (amarillo) y conectores del panel frontal (multicolor).



**Figura 1.16**  
Conector de alimentación de 24 pines.

### G) Conectores externos

La conexión entre los periféricos del sistema con el propio equipo se realiza, principalmente, en equipos de sobremesa, a través de conectores de comunicación externos anclados al lateral oeste de la placa base. Estos conectores emplean diferentes buses de comunicación hacia el chipset.



PARA SABER MÁS

Los conectores son interfaces de comunicación, pero la tecnología de transferencia la protagonizan los buses y protocolos desarrollados para cada conector, soliendo tener el mismo nombre. De ahí que existan conectores compatibles, como, por ejemplo, USB-C, Displayport y Thunderbolt 3.

Los principales conectores externos son:

- *eSATA:* utilizado para conectar dispositivos de almacenamiento externo.
- *Thunderbolt:* empleado para conectar periféricos de almacenamiento o para transmitir vídeo a periféricos. Emplea tecnología óptica.



**Figura 1.17**  
Conector  
eSATA.



**Figura 1.18**  
Conector  
Thunderbolt 2.



**Figura 1.19**  
Conector  
Thunderbolt 3.



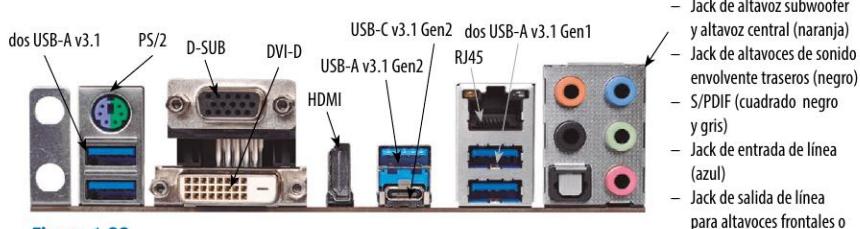
**Figura 1.20**  
Conector  
VGA.

- **USB:** conector empleado para conectar periféricos, como ratón, teclado, impresora, discos duros externos, smartphones, etc. Existen varias versiones de este conector.



**Figura 1.21**  
Diferentes tipos de conectores USB.

- **Conectores de vídeo:** transmiten señales de video a monitores. Los más empleados son D-SUB (VGA), DVI, Displayport y HDMI (estos dos últimos también pueden transmitir audio).
- **Conector Ethernet LAN:** también llamado RJ45, empleado para comunicarse por cable de par trenzado en una red de computadores.
- **Conectores de audio Jack y S/PDIF:** capaces de transmitir sonido analógico y digital, respectivamente.
- **Conectores PS/2:** utilizados para conectar teclados y ratones.



**Figura 1.22**  
Conectores externos de una placa base ASROCK.

### 1.3.4. Dispositivos de almacenamiento secundario

Se emplean para almacenar la información de manera permanente. Hemos de distinguir los dispositivos de los medios, ya que los primeros alojan a los segundos, los cuales contienen la información. Pueden estar juntos (como los discos duros mecánicos) o separados (una tarjeta

Flash SD necesita un lector de tarjetas SD para leer o escribir en ella). Los principales dispositivos o medios de almacenamiento no volátil se clasifican en:

#### A) *Medios de almacenamiento Flash*

Casi todos estos tipos de medios emplean tecnología Flash NAND, haciendo referencia a las puertas lógicas que almacenan los bits. Por ejemplo:

- ✓ Disco duro SSD (Solid State Drive): dispositivo de estado sólido, llamado así en contraposición a los discos duros magnéticos que presentan partes móviles.
- ✓ Tarjetas de memoria: aunque existe multitud de tipos y con distintas capacidades, las más utilizadas son las SD y CompactFlash, en sus diferentes formatos. Los dispositivos donde se utilizan suelen ser portátiles, es decir, cámaras de fotos, móviles, videoconsolas, etc.



Figura 1.23  
Disco duro SSD.



Figura 1.24  
Tarjeta CompactFlash.

#### Recurso web

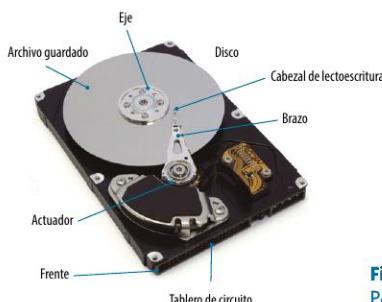
www

En este vídeo de Micron Technology se puede ver el proceso de fabricación de un disco duro de estado sólido.



#### B) *Dispositivos de almacenamiento magnético*

- *Disco duro mecánico*: está formado por un conjunto de discos de material rígido apilados sobre el mismo eje rotatorio. Cada disco consta de dos caras con pistas concéntricas en cada una, susceptibles de ser magnetizadas y alterar así su estado, para almacenar ceros o unos. Entre cada cara se encuentran los cabezales de lectura y escritura que actúan sobre las pistas.
- *Cintas*: medio de almacenamiento formado por una banda de plástico flexible que contiene pistas aptas de ser magnetizadas y que se recoge sobre sí para distribuirlo o almacenarlo de forma segura. Estos medios de almacenamiento se utilizan en centros de proceso de datos como formas de *backup*, ya que son muy económicas y de gran capacidad de almacenamiento, aunque lentas.



**Figura 1.25**  
Partes de un disco duro mecánico.

### C) Medios de almacenamiento óptico

Los medios ópticos más comunes son CD (Compact Disk), DVD (Digital Versatile Disk) y Blu-ray. Estos emplean diferente tecnología láser para grabar o leer en la superficie de los discos, almacenando información en forma de crestas y surcos. La capacidad también es un aspecto importante entre ellos, ya que cada uno puede tener:

- ✓ CD: hasta 700 MB.
- ✓ DVD: hasta 17 GB.
- ✓ Blu-ray: hasta 128 GB.

Para leer, grabar o regrabar información necesitan unidades ópticas hábiles a una determinada velocidad, representada por "x".

Estos dispositivos están en clara decadencia debido al auge del almacenamiento en la nube, discos duros externos (muy compactos y portables) y almacenamiento Flash.

Los medios de almacenamiento secundario más empleados en la pequeña y mediana empresa, así como en el hogar, son los medios Flash y discos duros mecánicos. Mientras sigue bajando el precio de los discos duros SSD, es más económico el coste por byte de los mecánicos. Sin embargo, las ventajas de cualquier dispositivo Flash son claramente diferenciadoras: menos consumo de energía, más ligeros, gran velocidad en operaciones de lectura y escritura, y más resistentes, al no incorporar partes móviles. De esta manera, los dispositivos magnéticos son aún una buena solución desde el punto de vista económico y de *backup*.

#### Actividad propuesta 1.6



Busca en Internet un modelo de cinta de almacenamiento y compara su coste por bit con respecto a los discos duros mecánicos y SSD.

#### 1.3.5. Fuente de alimentación

La alimentación eléctrica en cualquier sistema informático es fundamental, ya que de él dependen todos los componentes del equipo. Esta energía puede ser suministrada a través de baterías

(en dispositivos portables) o mediante una fuente de alimentación como un elemento más del sistema.

La fuente de alimentación es la encargada de proporcionar energía a la placa base, así como a todos los elementos que la rodean. Tiene tres objetivos, principalmente:

1. Suministrar energía a todos los componentes.
2. Actuar de barrera o protección ante alteraciones (ruidos o picos) de la red eléctrica externa.
3. Facilitar la extracción del flujo de aire caliente del sistema en equipos de sobremesa.

De esta manera, la fuente de alimentación transforma la tensión de entrada de 230 voltios a valores inferiores, rectifica la corriente alterna en continua, filtra la señal y la estabiliza.

En fuentes de alimentación para placas ATX los voltajes de salida son de 3,3 V, 5 V, 12 V y -12 V. Además, según el componente a alimentar nos encontramos con distintos conectores y cables de alimentación. En las siguientes imágenes podemos ver algunos.



**Figura 1.26**  
Conector  
de alimentación  
de la placa base  
de 20+4 pines.



**Figura 1.27**  
Conector  
de alimentación  
SATA.



**Figura 1.28**  
Conector  
de alimentación de  
tarjetas de expansión  
de 6+2 pines.



**Figura 1.29**  
Fuente de  
alimentación ATX  
RS Integrator 500.

### 1.3.6. Periféricos

Los dispositivos a través de los cuales los usuarios interactúan con el sistema informático se denominan *periféricos*.

Tradicionalmente, estos se han clasificado en:

- a) *Dispositivos de entrada*: permiten introducir información al sistema. Ejemplos: ratón, teclado, micrófono, etc.
- b) *Dispositivos de salida*: únicamente ofrecen al usuario información. Ejemplos: pantalla, impresora, altavoces, etc.
- c) *Dispositivos de entrada y salida*: realizan ambas tareas. Como, por ejemplo, una pantalla táctil. Dentro de este tipo, podemos encontrar:
  - Dispositivos de almacenamiento: permiten almacenar y recuperar la información. Ejemplos: disco duro, unidad de DVD, etc.
  - Comunicación: permiten la comunicación entre computadoras o elementos de interconexión de un sistema en red, como la tarjeta de red Ethernet o Wi-Fi.

En este apartado, hay que citar también los *adaptadores*, que permiten a los periféricos ser utilizados empleando otra conexión diferente a la utilizada por el sistema informático al que se van a asociar. Prácticamente todos los conectores disponen de una gran variedad de adaptadores. Por ejemplo, PS/2 a USB, USB-A a USB-C, HDMI/VGA a Displayport, etc.

#### 1.4. Controladores de dispositivos. Instalación de drivers

Todos los dispositivos hardware del sistema informático necesitan ser reconocidos por el sistema operativo para poder operar con total funcionalidad. Los componentes hardware disponen de *controladores* (chips o circuitos integrados), que se encargan de gestionar y coordinar el funcionamiento del dispositivo y, además, establecer un “diálogo” con la estructura de orden superior del sistema informático (por ejemplo, un disco duro se debe comunicar con el chipset de la placa base y el procesador).

Por tanto, para establecer una comunicación fluida y profunda entre el sistema operativo y cualquier dispositivo, se debe instalar un componente software asociado a cada uno de estos controladores. A este software se le denomina *driver* y es específico de cada uno de ellos, de manera que el sistema operativo podrá indicarle un conjunto de acciones o tareas y el dispositivo reconocerá cada una de ellas. De igual modo, el dispositivo le indicará al sistema operativo la finalización de una tarea, así como cualquier información de estado, monitorización o error.

Actualmente, los sistemas operativos disponen de una gran variedad de drivers preinstalados, que no hace necesaria su instalación en el momento de la conexión. No obstante, es recomendable su instalación, especialmente cuando no es muy común o dispone de características avanzadas (como una tarjeta gráfica).

La instalación de los componentes de un equipo o sus periféricos, hoy en día, es muy simple: el usuario conecta el componente al computador mediante la interfaz o el puerto correspondiente, y el sistema operativo lo reconocerá y configurará oportunamente para su correcto uso.

Sin embargo, para aprovechar todo el rendimiento de los componentes y periféricos de un sistema informático, debemos instalar los drivers más importantes: placa base y chipset, adaptadores gráficos y adaptadores de red.

La instalación puede ser expreso o a través del administrador de dispositivos que provea el sistema operativo. En el primer caso, el procedimiento es el siguiente:

1. Leer el manual de instalación del fabricante. Este paso es el más importante y el que resuelve la mayoría de los problemas en el proceso de instalación. Se debe prestar atención a las características del periférico y seguir los pasos descritos.
2. Antes de comenzar la instalación es preciso conocer las especificaciones de nuestro equipo: arquitectura de nuestro procesador (x64 o x86) y modelo de procesador, sistema operativo y su versión, espacio libre en disco duro y cantidad de memoria RAM instalada.
3. Conectar el componente o periférico al conector correspondiente. En algunos casos, el fabricante puede indicarnos que se conecte después de instalar su driver.
4. Una vez que sabemos los requerimientos del driver que se va a instalar y comprobamos que se ajustan a las especificaciones de nuestro equipo, procedemos a instalar el driver. Para ello, ejecutamos el archivo que inicia la instalación del driver o su asistente de instalación. Llegados a este paso, se supone que disponemos de dicho software; en

caso contrario, deberíamos descargarlo en la página oficial del fabricante, obteniendo la última versión para las especificaciones de nuestro equipo.

- Una vez termine el proceso de instalación, es recomendable reiniciar el equipo.

#### SABÍAS QUE...

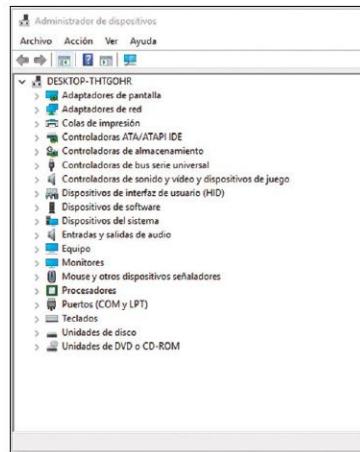
El concepto de *controlador* como elemento hardware difiere de *driver* como elemento software. El origen de la confusión procede de la traducción del inglés de driver como controlador.

### 1.4.1. Administración de dispositivos en Microsoft Windows

En Microsoft Windows, podemos administrar los dispositivos desde el 'Administrador de dispositivos', al que se puede acceder con el botón secundario del ratón sobre el botón de inicio de Microsoft Windows.

El 'Administrador de dispositivos' indica dispositivos que no se encuentren bien configurados o sin drivers mediante un ícono de advertencia. En tal caso, hemos de instalar o actualizar su driver.

Para instalar, desinstalar, actualizar o ver los detalles de un driver a través del 'Administrador de dispositivos', accedemos al dispositivo en cuestión y, en la pestaña 'Controlador', aparecen diferentes acciones. En caso de pulsar sobre 'instalar' o 'actualizar' un driver, aparecerá otra ventana, que nos dará a elegir entre 'Buscar el software de controlador actualizado automáticamente' (dando la opción de buscar una versión actualizada en Windows Update) o 'Buscar software de controlador en el equipo' (donde debemos indicar la ruta del driver o seleccionar entre una lista de drivers preinstalados).



**Figura 1.30**  
Administrador de dispositivos.

### 1.4.2. Administración de dispositivos en Ubuntu Desktop

Aunque los sistemas Linux suelen reconocer la mayoría de dispositivos, podemos acceder a la administración de estos mediante la orden *lshw*. Existen otras aplicaciones que muestran gráficamente los detalles de los dispositivos.

Para utilizar *lshw*, abrimos un terminal o interfaz de comandos a través del 'Tablero' o *Dash* ('Mostrar aplicaciones' y buscamos la aplicación 'Terminal') y ejecutamos dicha orden, tras lo que aparece una ventana con el listado de componentes hardware.



**Figura 1.31**  
Tablero de Ubuntu.



**Figura 1.32**  
Aplicación Terminal en Ubuntu.

Ubuntu instala los dispositivos con drivers genéricos de código abierto preinstalados o buscándolos en línea, de forma transparente al usuario. No obstante, da la posibilidad de instalar otros drivers de dispositivos, en lugar de los libres.

Para su administración, se puede acceder a 'Software y actualizaciones', en la pestaña 'Controladores adicionales'. Aparecerá un listado de dispositivos y los posibles drivers para instalar. Dependiendo del dispositivo, también dará la opción de no usarlo. En el listado de drivers, aparecerá la distinción 'privativo' u 'open source', dependiendo de la licencia de dicho software. Ubuntu recomienda emplear drivers *open source*.

En el caso de que Ubuntu no detecte un dispositivo, el fabricante del mismo debe aportar el driver para el sistema operativo y la arquitectura.

```
lubuntu@lubuntu:~$ sudo lshw
*-virtualbox
  description: Projector
  vendor: Intel Corporation
  product: Intel(R) HD Graphics
  version: 6.14.0.30000
  serial: 000c000000000000
  width: 64 bits
  capabilities: transform
  configuration: height=800
  businfo: PCI\VEN_8086\DEV_9900\SUB_0000\PROG_ID_0000
  product: VirtualBox
  manufacturer: Oracle Corporation
  version: 6.1.2
  serial: 6
  businfo: 0000:00:0d.0
  configuration: latency=0
lubuntu@lubuntu:~$
```

**Figura 1.33**  
Ejecución de *lshw* en modo administrador (anteponemos *sudo*).

## 1.5. Componentes software de un sistema informático

Se denomina *software*, *programa* o *aplicación* al conjunto de instrucciones que, al ser ejecutadas, llevan a cabo una o varias tareas relacionadas con un fin común. Ejemplos de software son: procesadores de texto, hojas de cálculo, juegos, aplicaciones bancarias, navegadores de Internet, exploradores de archivos, sistemas operativos, etc.

La relación entre software y hardware es tan estrecha que uno depende del otro, y viceversa. Cuanto más ajustado sea este margen, el sistema informático funcionará claramente mejor y será más eficiente.

Cuando ejecutamos un programa, el procesador es el encargado de decodificar e interpretar las instrucciones siguiendo un proceso cíclico y rutinario:

1. Un dispositivo de almacenamiento secundario almacena las instrucciones y datos necesarios para lanzar el software.

2. Antes de ejecutar cualquier instrucción, esta se traslada a las distintas estancias de memoria principal hasta alcanzar la última de ellas: los registros de la CPU.
3. Las unidades de control ejecutan una a una las instrucciones en sus registros, generando una serie de señales de control, estado o situación al resto de componentes, según la instrucción ejecutada.

### 1.5.1. Tipos de software

El software y las aplicaciones se pueden catalogar en diferentes tipos, según su uso:

- a) Software de sistema. Hace referencia a cualquier programa que está en contacto con el hardware y actúa de intermediario, gestor o administrador entre el usuario y el hardware. El más conocido es el sistema operativo, aunque todos los dispositivos disponen de algún software que gestiona y actúa de interfaz entre el hardware y los usuarios. Otros ejemplos son: BIOS, sistemas operativos, firmwares de equipos, drivers de dispositivos, aplicaciones de diagnóstico, software de virtualización o software de optimización de componentes.
- b) Conjunto de software necesario para el diseño, desarrollo o implementación de software de sistema o de aplicación. En este se incluyen editores, compiladores, depuradores de código y entornos de desarrollo integrados (IDE).
- c) Software de aplicación. Orientado a realizar tareas concretas y normalmente para un uso cotidiano por parte de un usuario final. Como, por ejemplo: aplicaciones ofimáticas, juegos, aplicaciones de diseño asistido por computador (CAD).

### 1.5.2. El sistema operativo

Por *sistemas operativos* podemos entender:

- Software que actúa de interfaz entre el usuario y el hardware.
- Software que se encarga de gestionar eficazmente los recursos hardware y software del sistema de forma transparente para el usuario.
- Conjunto de programas encargados de facilitar la interacción del usuario con la máquina, aprovechando todos sus recursos.

El sistema operativo no es un software aislado y único que se encarga de una gran cantidad de tareas, sino un núcleo y un conjunto de módulos que interactúan entre sí para realizar las tareas descritas en las definiciones anteriores.

Este software se encuentra en continuo desarrollo, ya que se actualiza para evitar problemas de seguridad y adaptarse a la evolución de los recursos hardware y software.

Existe una gran variedad de sistemas operativos. Cada uno tiene unas particularidades que lo hacen idóneo para según qué actividades.

Como ejemplos de sistemas operativos, destacamos Microsoft Windows de Microsoft, macOS de Apple, Android de Google o diferentes distribuciones de Linux.

## 1.6. Proceso de arranque de un sistema informático. POST

Todos los equipos informáticos se inician de manera muy similar. Al pulsar el botón de arranque del panel frontal, la fuente de alimentación transforma, rectifica, filtra y estabiliza la corriente externa a los distintos valores necesarios. Así se distribuyen principalmente los siguientes voltajes:

- ✓ 3,3 V para los componentes electrónicos de más baja potencia, transformándose este, a su vez, en otros inferiores, ya en la placa base.
- ✓ 5 V para algunos componentes electrónicos y mecánicos de poca potencia.
- ✓ 12 V para dispositivos que emplean motores o que necesiten transformarse en valores menores.

Por tanto, al inicio, el procesador comienza a ejecutar instrucciones de la ROM BIOS (estas se han almacenado previamente en la memoria principal). Una de las primeras tareas de la BIOS es el testeo del sistema, para continuar con el arranque del mismo. A este proceso se le conoce como *POST (power on self test)*.

Durante este proceso POST, que comienza antes de que aparezca imagen alguna por pantalla, se testean los componentes fundamentales (procesador, memoria, fuente de alimentación y placa base) para detectar algún problema o error. Si encontrara algún problema, la BIOS efectúa señales indicando el tipo de error y, en función de su gravedad, continuará o no el arranque del sistema. Estas señales pueden ser:

- a) *Sonoros*: a través del altavoz de la placa base, mediante pitidos largos o cortos de diferente frecuencia, señala el error.
- b) *Visuales*: algunas placas base incorporan displays que indican un código representativo del error. También se pueden emplear tarjetas POST, conectadas a slots de expansión para mostrar estos errores.

Según el fabricante y modelo de la BIOS, los avisos sonoros o por códigos visuales tienen una correspondencia con el error detectado que se debe consultar en la guía del fabricante de la BIOS.

Más tarde, la BIOS configura e inicializa los componentes hardware, atendiendo a los valores de la RAM-CMOS de la BIOS (los cuales se pueden configurar accediendo al BIOS Setup Utility):

1. Se inicia el adaptador gráfico, mostrando por pantalla algunas comprobaciones o configuraciones que se van realizando en los siguientes pasos.
2. Testea, inicializa y establece algunos valores del adaptador gráfico, procesador, memoria RAM, dispositivos de almacenamiento secundario, etc.
3. Muestra mensajes para acceder al BIOS Setup Utility, actualizar la BIOS o acceder al menú rápido de selección de medios de arranque.
4. Se activan buses de la placa base: SATA, USB, etc.
5. Se activan otros dispositivos Plug and Play.
6. Se activan otras BIOS de componentes instalados en la placa base.
7. Se muestra un resumen de los dispositivos detectados y los recursos asignados.

## TEN EN CUENTA

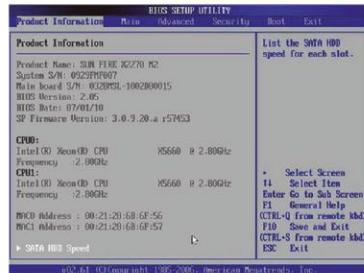
- ✓ En esta segunda etapa, si se detecta algún problema en alguno de los pasos de inicialización, configuración o paso de testigo al primer medio de arranque, se mostrará el correspondiente mensaje por pantalla.

Una vez terminado todo el proceso POST, inicialización y configuración de componentes, la BIOS le pasa el testigo al primer medio de almacenamiento (así configurado en la BIOS Setup Utility) para que comience la carga del sistema operativo.

En la figura 1.34 se muestra una captura del proceso de configuración e inicialización, donde se detalla por líneas y en orden: los logos de la BIOS y del fabricante del equipo, la marca de la BIOS, la fecha de la versión de la BIOS, la marca y modelo del equipo (Servidor Sun Fire X2270), el tipo de procesador, frecuencia y número, las teclas de acceso al BIOS Setup Utility (F2), arranque por red (F12) y menú de opciones de arranque (F8), inicializa y muestra la versión del firmware del BMC y, por último, inicializa otros controladores (USB).



**Figura 1.34**  
Proceso de configuración e inicialización.



**Figura 1.35**  
Aspecto de un BIOS Setup Utility.



**Figura 1.36**  
Aspecto de un UEFI BIOS Setup Utility.

## 1.7. Máquinas virtuales

Actualmente, la virtualización es una herramienta que presenta tantas ventajas que sería impensable un futuro informático sin ella. En empresas medianas y grandes compañías su uso es muy común.

**RECUERDA**

- ✓ Cuando se habla de virtualización en informática, estamos hablando de la abstracción de los recursos hardware de la computadora.

### 1.7.1. Concepto y usos

Llamamos *máquina virtual* a una computadora no real, instalada y configurada en un sistema informático mediante un software que permite simular su funcionamiento autónomo.

El sistema informático al que se abstraen sus recursos para poder instalar máquinas virtuales se denomina *host* o *anfitrión*. Igualmente, al sistema operativo instalado en él se le llama *sistema operativo host* o *sistema operativo anfitrión*. Sobre ellos se podrán instalar las máquinas virtuales con *sistemas operativos guest* o *sistemas operativos invitados*.

Las máquinas virtuales se emplean principalmente para:

- Realizar pruebas. Se pueden probar sistemas informáticos, software y configuraciones sin que un fallo importante en ellos afecte a la máquina real.
- Portabilidad. Al ser software, estos sistemas virtualizados se pueden trasladar muy fácilmente y con una rápida implementación entre máquinas anfitrionas.
- Ahorro de costes. El coste de una máquina virtual es nulo, si pensamos que en un equipo anfitrión podemos instalar multitud de máquinas virtuales con diferentes configuraciones.
- Copias de seguridad. Al tener un entorno virtualizado, es decir, un software instalado y correctamente configurado, podemos hacer copias de seguridad al tratarse de archivos. Por ello, podemos tener grandes sistemas clonados ante posibles fallos que pueden ser restaurados fácilmente.
- Centralización de servicios. Un equipo puede estar configurado para albergar multitud de máquinas virtuales con diferentes servicios, facilitando su mantenimiento, ampliación y actualización de hardware, así como simplificar los accesos y la seguridad.

### 1.7.2. Software de virtualización

Para llevar a cabo la virtualización se necesita un software de abstracción de los recursos hardware de una máquina anfitriona o software de virtualización, llamado *hipervisor* o *VMM (virtual machine monitor)*.

**TEN EN CUENTA**

- ✓ El software de virtualización debe gestionar los recursos hardware del equipo anfitrión entre el conjunto de máquinas virtuales creadas sobre él. Por tanto, los recursos hardware reales determinan el rendimiento de las máquinas virtuales.

Los hipervisores pueden ser nativos (sobre el propio hardware del equipo) o alojados (sobre el sistema operativo).

El software de virtualización permite crear varias máquinas virtuales con diferentes recursos hardware, y hacer uso de ellas simultáneamente. Al ser sistemas independientes, podemos instalar sistemas operativos y cualquier otro software. Estas máquinas virtuales se podrán configurar para que se comuniquen entre ellas e incluso con el equipo anfitrión.

### A) Tipos

Existe una gran variedad de software de virtualización de distintos desarrolladores:

- ✓ *VMWare*. Presenta multitud de productos comerciales. Es la compañía con mayor solera en entornos empresariales.
- ✓ *Microsoft Hyper-V Server*. Las versiones más avanzadas de Wicrosoft Windows incorporan este software; en algunas está instalado como tal (versiones Server) y en otras como característica opcional (versiones Pro y Enterprise). Suele emplearse aprovechando el sistema operativo anfitrión.
- ✓ *Oracle VM VirtualBox*. Utilizado por muchas empresas y particulares. El hecho de ser un producto open source bajo los términos de GNU GPLv2 y soportar una gran cantidad de sistemas operativos anfitriones e invitados hace que sea ideal para trabajar con él.
- ✓ *Citrix XenServer*. Suele emplearse en empresas o entornos domésticos. Dispone de una versión gratuita. Al ser open source, también tiene muchos adeptos.
- ✓ *Qemu*. Es uno de los más empleados en sistemas operativos Linux, destacando por su gran rendimiento en estos sistemas anfitriones.
- ✓ *Parallels*. Solución de virtualización que proveen los sistemas Macintosh.

### B) Características

Algunas de las características y funciones más importantes que presentan las plataformas de virtualización y que determinan su elección son:

- Plataforma. Variedad de sistemas operativos anfitriones y arquitecturas (32 o 64 bits) sobre los que se pueda instalar.
- Sistemas operativos guest. Hace referencia a qué sistemas operativos se pueden instalar en el host.
- Licencia. El tipo de licencia puede determinar la adquisición del software.
- Portabilidad. Las opciones que ofrezca para exportar una máquina virtual o un disco duro virtual.
- Compatibilidad entre máquinas virtuales de distinto software de virtualización.
- Soporte de puertos de comunicación del host, como USB, lectores de tarjetas, etc.
- Soporte para gráficos 3D.
- Creación y gestión de instantáneas o puntos de restauración (estados de configuración de una máquina virtual en un momento dado).
- Actualizaciones del software de virtualización, soporte técnico y foros de ayuda.

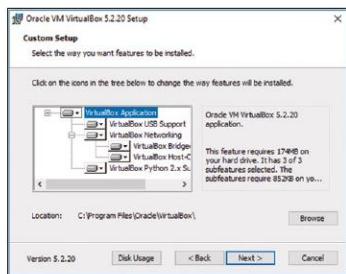
## 1.8. Oracle VM VirtualBox

Vamos a emplear Oracle VM VirtualBox como software de virtualización para trabajar con la instalación y explotación de sistemas operativos por disponer de un producto open source bajo los términos de GNU GPLv2 y ser ampliamente usado a nivel global.

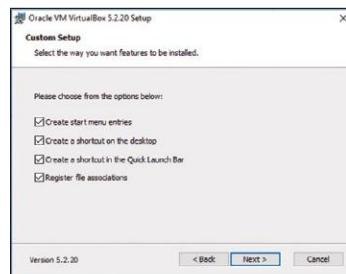
La descarga de Oracle VM VirtualBox (en adelante VirtualBox) se realiza desde la propia página web de la compañía: [bit.ly/3aAbjn4](http://bit.ly/3aAbjn4). En la página de descargas, seleccionamos la plataforma adecuada para el sistema donde se vaya a instalar, y comenzará la descarga. Una vez terminada, su proceso de instalación es muy sencillo, siguiendo los pasos del asistente.

### 1.8.1. Proceso de instalación de Oracle VM VirtualBox

Seleccionamos las características de la instalación (por defecto, no las modificamos) e indicamos la ruta de instalación de VirtualBox. Más tarde, indicará si deseamos crear entradas en el menú de inicio de Windows, un acceso directo en el escritorio, un acceso en la barra de inicio rápido y registrar asociaciones de archivos (para vincular la apertura de archivos de VirtualBox con esta aplicación).



**Figura 1.37**  
Personalización de la instalación.



**Figura 1.38**  
Opciones para crear tras la instalación.

Posteriormente, avisa que el adaptador de red se reiniciará para instalar las características de red de VirtualBox y, por tanto, se desconectará de la red. Al aceptar, comienza la copia de los archivos, pudiendo preguntar si damos nuestro consentimiento para instalar algunos dispositivos virtuales asociados. Por último, indica que el proceso de instalación ha terminado.

### Actividad propuesta 1.7



Descarga Oracle VM VirtualBox de la página oficial e instálalo.

Sobre la instalación base, se pueden añadir nuevas funcionalidades o paquetes, que varían según la versión de VirtualBox. Estos amplían la funcionalidad de todas las máquinas virtuales y su operatividad.

Además, existen otros paquetes opcionales, aunque sí muy recomendables (llamados *Guest Additions*), que se instalan sobre el sistema operativo guest. Mejoran el rendimiento de los sistemas operativos invitados y añaden otras características.

### 1.8.2. Entorno de Oracle VM VirtualBox

La aplicación, recién instalada, presenta un entorno básico.

1. Una barra principal con tres opciones:
  - Archivo: permite administrar la virtualización en general: preferencias, importaciones y exportaciones de máquinas virtuales, la red, los medios de almacenamiento, etc.
  - Máquina: dispone de multitud de opciones para administrar las máquinas virtuales.
  - Ayuda: con accesos a ayudas, destacando el manual de usuario de VirtualBox.
2. En la parte superior y debajo de la barra principal, existe una cinta de botones que permitirán realizar acciones básicas sobre las máquinas virtuales: crear una nueva máquina virtual, configurar una existente, descartarla o iniciarla.
3. En el lado izquierdo, se encuentra un panel que mostrará el listado de máquinas virtuales.
4. En la parte superior derecha, se encuentran dos botones con los que se seleccionan las vistas del entorno: ‘Herramientas de máquina’ y ‘Herramientas globales’.



Figura 1.39  
Entorno de VirtualBox.

### 1.8.3. Creación de una máquina virtual en Oracle VM VirtualBox

Antes de instalar un sistema operativo, debemos crear un entorno virtual hardware adecuado para aquél, conocido como *máquina virtual (MV)*.

Para comenzar el proceso de creación de una nueva máquina virtual, pulsamos en el botón *Nueva* de la cinta de botones superior izquierda. Entonces, aparecen una serie de opciones para configurar para crear la máquina virtual (aunque se podrán modificar con posterioridad, una vez creada). Por defecto, aparece el modo experto, pero se puede crear en modo guiado (botón en la parte inferior). Se han de establecer las siguientes opciones:

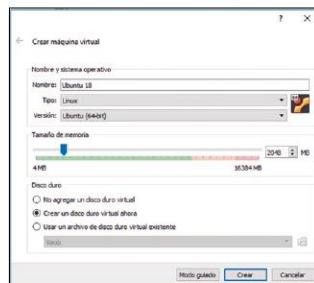
a) Nombre y sistema operativo:

- Nombre: este nombre figurará en la lista de máquinas virtuales. Para que no se den confusiones, se deben evitar nombres cortos o poco representativos de su contenido o funcionalidad.
- Tipo y versión: tipo de sistemas operativos que se van a instalar. Están agrupados y es recomendable asignar el adecuado, ya que VirtualBox habilitará ciertas características que podría necesitar el sistema operativo invitado.

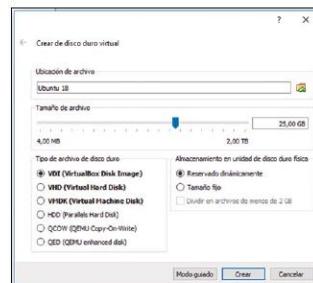
b) Tamaño de memoria. Este aspecto es crucial para la ligereza del sistema operativo anfitrión e invitado, más aún si tenemos en cuenta que varias máquinas virtuales se pueden ejecutar a la vez. Hemos de seleccionar la cantidad de memoria RAM que queremos disponer en la máquina virtual. Esta cantidad se retirará del sistema operativo anfitrión durante la ejecución de la máquina virtual, por lo que hemos de ser conscientes de dicha limitación. La barra donde aparece la cantidad de memoria RAM asignada nos advierte, según la cantidad de RAM del sistema, del margen asignado: aceptable (verde), intermedio (naranja) y crítico (rosa).

c) Disco duro. Selección del medio de almacenamiento virtual para la máquina virtual:

- No agregar un disco duro virtual. Si no deseamos crearlo porque vayamos a añadirlo posteriormente o porque no se necesite.
- Crear un disco duro virtual ahora. En caso de realizar una instalación nueva, se necesita un disco duro para almacenar el sistema operativo. Esta opción es la más común en nuevas instalaciones.
- Usar un disco duro virtual existente. Si ya disponemos de un disco duro virtual y deseemos añadirlo a la máquina virtual. Es otra opción si tenemos un disco duro con un sistema operativo instalado.



**Figura 1.40**  
Opciones  
de creación.



**Figura 1.41**  
Configuración  
del disco duro virtual.

- Seleccionamos las opciones de la imagen superior y, al pulsar en 'Crear', aparece otro conjunto de opciones sobre la creación del disco duro virtual:
  - Ubicación de archivo: indica el nombre y la ruta de almacenamiento. Al igual que el nombre de la máquina virtual, debemos establecer un nombre representativo de su contenido.

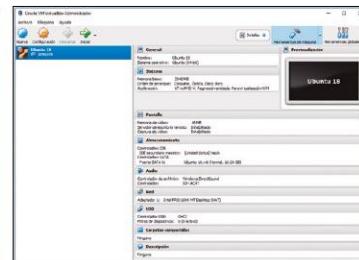
- Tamaño de archivo: cantidad de memoria definida para el disco duro virtual. Según el tipo de disco duro virtual, el almacenamiento en la unidad física de disco duro puede ser:
  - o Reservado dinámicamente: permite una gestión de almacenamiento más flexible, puesto que no reserva todo el tamaño del archivo, sino que va ocupándolo incrementalmente conforme se vayan haciendo uso de nuevos sectores.
  - o Tamaño fijo: reserva el espacio establecido en 'Tamaño del archivo'. Para la creación del archivo virtual, puede tomar un tiempo considerable dependiendo del tamaño de la imagen y el rendimiento de escritura del disco duro, pero suele ser más rápido que el dinámico una vez creado.
  - o La opción 'Dividir en archivos de menos de 2GB' se puede emplear si se prevé un almacenamiento del archivo imagen en sistemas de archivos que no pueden gestionar archivos grandes. Esta opción solo se habilita con el tipo de archivo *VMDK (Virtual Machine Disk)*.

d) Tipo de archivo de disco duro. Se pueden emplear herramientas para cambiar entre formatos de discos virtuales e incluso modificar las extensiones de los mismos. Entre varios formatos de discos duros virtuales, se pueden seleccionar:

- *VDI (VirtualBox Disk Image)*: formato propio de VirtualBox para discos duros virtuales. Es la opción por defecto y aconsejada por VirtualBox al suponer mayor rendimiento en su plataforma.
- *VHD (Virtual Hard Disk)*: formato empleado por Microsoft.
- *VMDK (Virtual Machine Disk)*: formato abierto que suelen utilizar otros softwares de virtualización, como VMWare.
- *HDD (Parallels Hard Disk)*: formato utilizado por el software de virtualización Parallels.
- *QCOW (QEMU Copy-On-Write)*: formato de QEMU.
- *QED (QEMU enhanced disk)*: empleado por compatibilidad con el formato QED de QEMU.



**Figura 1.42**  
Entorno de VirtualBox con una MV.



**Figura 1.43**  
Detalles de una MV.

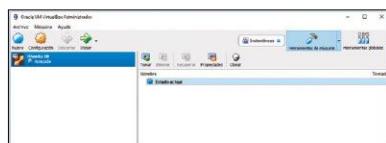
Una vez creada la máquina virtual, aparece el entorno con la nueva máquina en el listado del panel izquierdo. También se han habilitado los botones 'Configuración' e 'Iniciar'. Y, en el cuerpo principal, aparecen ahora las dos opciones de 'Herramientas de máquina': 'Detalles' e 'Instantáneas'.

En 'Detalles' podemos inspeccionar todas las propiedades de la máquina virtual recién creada y acceder a la 'Configuración' si pulsamos en la cabecera de cada una. Además, es recomendable realizar instantáneas o *snapshots*, es decir, guardar el estado de una máquina virtual.

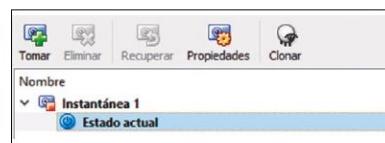
#### 1.8.4. Creación de instantáneas

En 'Instantáneas', aparece el árbol de instantáneas asociado a la máquina virtual. Una instantánea es el estado de configuración de una máquina virtual en un momento dado, es decir, una forma de almacenar todas sus propiedades (toda la configuración global, incluyendo el estado de los discos duros virtuales). De esta manera, se pueden crear múltiples instantáneas para recuperarlas posteriormente. Su principal utilidad es la de copia de seguridad ante situaciones cambiantes de software o hardware, pruebas, etc.

Por tanto, es recomendable crear una instantánea (*snapshot*) como backup del sistema sin instalación del sistema operativo. Cuando se instale, realizaremos otra.



**Figura 1.44**  
Opciones en instantáneas  
de una MV en VirtualBox.



**Figura 1.45**  
Instantáneas  
de una MV.

Para realizar la instantánea, pulsamos en 'Tomar' y aparecerá una ventana para que indiquemos el nombre de la instantánea y la descripción de la misma.

#### Actividad propuesta 1.8



Crea una máquina virtual con el nombre *Prueba*, estableciendo una configuración con memoria RAM de 2GB y un disco duro de tamaño fijo de 20 GB. Crea una instantánea y, a continuación, elimina la máquina virtual.



#### Recurso digital 1.1

Carpetas compartidas en Oracle VM VirtualBox.

#### 1.9. Normas de seguridad y prevención de riesgos laborales

Cuando trabajamos con un sistema informático, debemos adoptar unas recomendaciones ergonómicas y de seguridad básicas de cara a prevenir o minimizar cualquier riesgo laboral.

En la Unión Europea y en España existe una normativa específica en cada sector laboral que regula las normas de seguridad y de prevención de riesgos laborales.

En España, la norma más amplia al respecto es la Ley 31/1995, de prevención de riesgos laborales, que establece la seguridad para el trabajador cuando realiza sus actividades laborales. Esta ley señala que “los trabajadores tienen derecho a una protección eficaz en materia de seguridad y salud en el trabajo”.

Tanto el empresario como el trabajador tienen una serie de derechos y obligaciones en materia de riesgos laborales. Algunos de los derechos son: información a los trabajadores, evaluación de riesgos en el puesto de trabajo, formación o planes de emergencia ante riesgos graves.

Para prevenir los riesgos laborales se debe:

1. Adoptar un plan de prevención de riesgos laborales.
2. Evaluar los riesgos.
3. Planificar y ejecutar la actividad preventiva.

Es importante destacar como obligaciones del trabajador:

- ✓ Usar adecuadamente, de acuerdo con su naturaleza y los riesgos previsibles, las máquinas, aparatos, herramientas, sustancias peligrosas, equipos de transporte y, en general, cualesquier otros medios con los que desarrollen su actividad.
- ✓ Utilizar correctamente los medios y equipos de protección facilitados por el empresario, de acuerdo con las instrucciones recibidas de este.
- ✓ No poner fuera de funcionamiento y utilizar correctamente los dispositivos de seguridad existentes o que se instalen en los medios relacionados con su actividad o en los lugares de trabajo en los que esta tenga lugar.
- ✓ Contribuir al cumplimiento de las obligaciones establecidas por la autoridad competente con el fin de proteger la seguridad y la salud de los trabajadores en el trabajo.
- ✓ Cooperar con el empresario para que este pueda garantizar unas condiciones de trabajo que sean seguras y no entrañen riesgos para la seguridad y la salud de los trabajadores.

En lo que a nuestro tema de estudio se refiere, destacamos las siguientes normas de seguridad y prevención de riesgos laborales respecto a los equipos con pantallas de visualización:

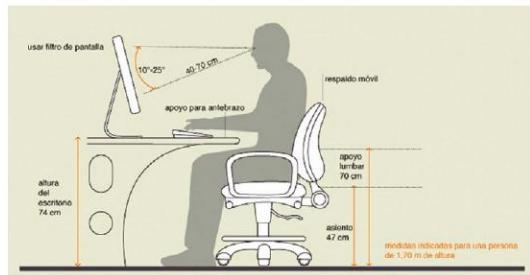
a) Pantalla:

- Los caracteres deben estar bien definidos y configurados de forma clara.
- El usuario deberá poder ajustar fácilmente la luminosidad y el contraste entre los caracteres y el fondo de la pantalla, y adaptarlos fácilmente al entorno.
- La pantalla deberá ser orientable e inclinable a voluntad, con facilidad para adaptarse a las necesidades del usuario.
- La pantalla no deberá tener reflejos ni reverberaciones que puedan molestar.

b) Teclado:

- El teclado deberá ser inclinable e independiente de la pantalla para permitir que el trabajador adopte una postura cómoda que no provoque cansancio en los brazos o las manos.
- Tendrá que haber espacio suficiente delante del teclado para que el usuario pueda apoyar los brazos y las manos.

- c) Mesa o superficie de trabajo:
- Debe tener dimensiones suficientes y permitir una colocación flexible de la pantalla, del teclado, de los documentos y del material accesorio.
  - El espacio debe ser suficiente para permitir a los usuarios una posición cómoda.
- d) Asiento del trabajo:
- Deberá ser estable, regulable en altura, proporcionando al usuario libertad de movimiento y procurándole una postura confortable.
- e) Entorno:
- Debe existir un espacio de dimensiones suficiente para adoptar cambios de postura y movimientos de trabajo.
  - La iluminación debe ser adecuada, evitando deslumbramientos y reflejos.
  - El ruido no debe perturbar la atención del trabajador.
  - Las condiciones atmosféricas de temperatura y humedad, deben ser adecuadas para el desarrollo del trabajo



**Figura 1.46**  
Ejemplo de uso de un entorno ergonómico.  
Fuente: <https://biwo.es/>

Existen muchas situaciones que pueden causarnos lesiones puntuales o generar problemas crónicos. Por ello, es importante adoptar una serie de recomendaciones posturales ergonómicas y condiciones ambientalmente correctas.

Por tanto, cuando nos sentamos para trabajar con un ordenador, es conveniente tomar una postura adecuada: regular la altura del asiento, la mesa, el apoyo lumbar y el respaldo, apoyar el antebrazo cuando escribimos con el teclado o manejamos el ratón, regular la inclinación y altura de la pantalla, etc.

Además, al utilizar dispositivos eléctricos, debemos adoptar una serie de medidas de prevención básicas:

1. Leer los manuales de instrucciones de uso de todos los componentes eléctricos.
2. Mantener los componentes eléctricos en buen estado.
3. Desconectar los componentes de la red eléctrica cuando no vayan a ser utilizados.
4. Disponer de una instalación eléctrica adecuada para nuestros sistemas y que permita evitar accidentes ante corrientes excesivas o derivaciones.

5. Manejar correctamente y con los medios necesarios los dispositivos sensitivos a descargas electrostáticas.
6. Evitar manipular los componentes con las manos mojadas o húmedas.
7. Cuando se acceda al interior de los dispositivos eléctricos, estos deben estar desconectados de la red eléctrica.
8. No desplazar equipos no portables cuando están en funcionamiento.

## TOMA NOTA



Las recomendaciones cuando estamos sentados frente a un teclado y pantalla son, principalmente:

- El teclado ha de estar situado como mínimo a 10 cm de distancia desde el borde de la mesa.
- El ratón debe estar cerca del teclado.
- La pantalla debe estar a una distancia mínima de 40 cm.
- La silla debe permitir tener un apoyo completo lumbar y ser regulable.
- Mantener una postura erguida, con las rodillas a la altura de la pelvis y los brazos apoyados.

## Resumen

- En la actualidad, un sistema informático puede ser desde un supercomputador con cientos de procesadores y en red o un reloj de pulsera wearable con procesador y memoria principal. Es decir, los hay de muchos tipos, con diferentes características y propósitos diferentes.
- Todos ellos tienen en común las partes fundamentales de las que todo sistema informático, basado en las arquitecturas de Von Neumann y Harvard, dispone: memoria principal, procesador (con registros, unidad de control, unidad aritmético-lógica), buses del sistema y periféricos.
- En este capítulo se han estudiado los elementos hardware de un computador de sobremesa, ya que es ideal para el entendimiento de otros sistemas. Por tanto, cuando se implementa o se fabrica cualquier sistema informático, estos lo hacen con componentes similares, adaptándose a las características del sistema informático en cuestión. Los principales componentes hardware son: microprocesador, memoria principal, placa base y fuente de alimentación. Solo con estos elementos ya podría funcionar, aunque lo normal es que se apoye en dispositivos de almacenamiento secundario para disponer de manera permanente de programas y datos, así como de periféricos para poder comunicarnos con él.
- Para que el hardware haga su trabajo necesita de un conjunto de instrucciones y datos que todo sistema informático debe disponer, es decir, el software. Los hay de muchos tipos, aunque el más cercano al hardware es el software de sistema. Ejemplo de ello

es el sistema operativo, el cual organiza, gestiona, administra todos los recursos de la computadora y hace de interfaz entre el usuario y la máquina. Esta labor es muy compleja y se irá estudiando en posteriores capítulos.

- Tanto el hardware como el software de un sistema informático se pueden virtualizar, es decir, se pueden abstraer unos recursos hardware de una máquina anfitriona a través de un software de virtualización y trabajar sobre ello para instalar un software invitado como un sistema operativo o cualquier otro tipo de aplicación. La virtualización presenta multitud de ventajas, permitiéndonos el estudio de sistemas operativos y su conectividad de manera muy sencilla.
- Por último, se han estudiado una serie de normas de seguridad y recomendaciones ergonómicas para evitar lesiones o enfermedades. Estas deben ser aplicadas en clase, en casa o en el trabajo, ya que son fundamentales para el correcto desempeño de cualquier actividad relacionada con la materia.



## Ejercicios propuestos

1. Accede a la página web del fabricante MSI y localiza la placa base MSI X220 RAIDERS (<https://es.msi.com/Motherboard/X299-RAIDER/Specification>). Realiza los siguientes ejercicios (si es necesario, descarga el manual de usuario):
  - a) Descarga una imagen de la placa base y otra de los conectores externos del panel trasero. Señala aquellos elementos estudiados, indicando su nombre técnico.
  - b) ¿Qué factor de forma tiene la placa base?
  - c) Procesador: ¿Qué tipo de socket de procesador tiene? ¿Con qué procesadores es compatible la placa base?
  - d) Memoria RAM: ¿Qué módulos de memoria soporta? ¿Qué cantidad máxima de memoria puede instalarse? ¿Dispone de tecnología multicanal?
  - e) ¿Qué chipset monta la placa base? Indica las características de dicho chipset mediante un diagrama o describiéndolo.
  - f) Capacidad de expansión: ¿De cuántas ranuras de expansión dispone y de qué tipo?
  - g) Conectores internos: ¿De qué conectores internos dispone y cuál es su número?
  - h) ¿La placa base permite resestar la memoria BIOS RAM-CMOS? ¿Cómo?
  - i) ¿Cuántos conectores de alimentación posee? ¿De qué tipo?
  - j) Indica el procedimiento para descargar los drivers de la placa base.
2. Realiza una comparativa teórica de las velocidades de transferencia de datos de los buses USB, eSATA y Thunderbolt en sus versiones más actuales.
3. Descarga la aplicación CPU-Z de la página web oficial: <https://www.cpuid.com/>. Instálala y ejecútala. Anota la información del equipo:
  - Procesador: nombre comercial, número de núcleos, frecuencias, tecnología de integración, voltaje, conjunto de instrucciones, cachés y número de hilos.

- Placa base y chipset: fabricante y modelo de placa base, tipo de chipset, fabricante y modelo de BIOS.
  - Memoria: tipos, tamaños, latencias y canales.
  - Información en tiempo real del estado de todos los componentes: frecuencias, voltajes y temperaturas.
4. Sobre el mismo equipo y una vez recogida la información del punto anterior, descarga el manual de la placa base de la página oficial del fabricante y localiza los pines de reseteo de la memoria BIOS RAM-CMOS. Accede al BIOS Setup Utility y anota la prioridad del orden de arranque de los dispositivos de almacenamiento secundario. Cambia dichos parámetros a otros cualesquier. Apaga el equipo. Desconecta los cables de alimentación del equipo y de los dispositivos periféricos. Abre el equipo y resetea la BIOS RAM-CMOS. Vuelve a conectar los cables del equipo y comprueba que se ha reseadoo la BIOS RAM-CMOS a sus valores originales de fábrica, volviendo a entrar en la BIOS Setup Utility. Modifica los valores de la BIOS RAM-CMOS a los valores previos al resedeo.
5. Busca al menos tres ejemplos para cada tipo de software.
6. Vamos a analizar la segunda parte del proceso POST de nuestro equipo. Para ello, cuando aparezca una imagen como la de la figura 1.34 y, antes de que pase a la siguiente pantalla, analiza todas las líneas. Para detener la imagen y el proceso de arranque del equipo, pulsa la tecla Pausa en el teclado.
- Explica cada una de las líneas.
  - Accede al BIOS Setup Utility:
    - Obtén información de la temperatura del procesador, su voltaje y la velocidad de los ventiladores.
    - Anota el orden de los medios de arranque. Pon en primer lugar un medio de arranque que no disponga de sistema operativo. Guarda los cambios y reinicia el sistema. ¿Qué ocurre? Restáuralo a su orden anterior.
7. Pon dos ejemplos, buscando en Internet, de dos hipervisores nativos y otros dos alojados.
8. Cita diez ejemplos de distribuciones GNU/Linux.
9. Realiza el proceso de descarga e instalación de la última versión de Oracle VM VirtualBox. A continuación:
- Crea una máquina virtual llamada *Ubuntu MV* para la futura instalación de Ubuntu Desktop con 4 GB de memoria RAM y un disco duro reservado dinámicamente de 250 GB.
  - Crea una máquina virtual llamada *Windows MV* para la futura instalación de Microsoft Windows 10 con 4 GB de memoria RAM y un disco duro reservado dinámicamente de 300 GB.
  - Crea una instantánea de cada una con el nombre *MV Limpia*.
  - Accede a la configuración de *Windows MV* y añade un segundo disco duro reservado dinámicamente de 200 GB. En caso de duda, consulta la ayuda de *VirtualBox*.
10. Descarga una versión de prueba de VMware Workstation Pro desde <https://www.vmware.com/es/products/workstation-pro.html>. Instala la aplicación y crea una máquina virtual de prueba.

## ACTIVIDADES DE AUTOEVALUACIÓN

1. ¿En qué modelo de arquitectura el acceso a datos e instrucciones se realiza simultáneamente?:  
 a) Von Neumann.  
 b) Harvard.  
 c) Von Neumann y Harvard.
2. ¿Cuál de los siguientes periféricos es de entrada y salida?:  
 a) Pantalla.  
 b) Altavoces.  
 c) Tarjeta de red.
3. La fuente de alimentación se encarga de transformar, rectificar, filtrar y estabilizar la corriente externa. Sus voltajes de salida son aproximadamente de:  
 a) 3,3V, 5V y 15V.  
 b) 3V, 5V y 15V.  
 c) 3,3V, 5V y 12V.
4. El controlador de un dispositivo:  
 a) Es un componente hardware que se encarga de gestionar y coordinar el funcionamiento del dispositivo.  
 b) Es un componente software que debe ser instalado en el sistema operativo para el correcto funcionamiento.  
 c) Es una parte del sistema operativo.
5. Las normas de seguridad y prevención de riesgos laborales respecto a los equipos con pantallas de visualización establecen que:  
 a) La pantalla debe estar a una distancia mínima de 40 cm y la parte superior de la misma a la altura de los ojos.  
 b) La silla debe ser estable y no regulable para evitar posturas desfavorables.  
 c) La mesa de trabajo ha de estar bien organizada, aunque no permita una postura confortable.
6. ¿Qué tipo de archivo de disco duro virtual es propio de Oracle VM VirtualBox?:  
 a) VDI.  
 b) VHD.  
 c) HDD.
7. El conector M.2 es empleado principalmente para:  
 a) Transferir datos de discos duros externos.  
 b) Alimentar el procesador.  
 c) Conectividad y almacenamiento.

8. En un equipo con tecnología Dual Channel empleando módulos RAM diferentes:
- a) No puede hacer uso de dicha tecnología.
  - b) El sistema no arrancará al producirse un error POST.
  - c) El controlador de memoria se ajusta a las velocidades, latencias o capacidades más bajas.
9. Los discos duros SSD frente a los discos duros mecánicos:
- a) Son más resistentes.
  - b) Son más lentos en operaciones de lectura.
  - c) Son más pesados.
10. La gestión de drivers de dispositivos:
- a) En Ubuntu Desktop se puede realizar únicamente por línea de comandos.
  - b) En Microsoft Windows se puede realizar a través del "Administrador de dispositivos" y en Ubuntu Desktop a través de "Software y actualizaciones".
  - c) En Ubuntu Desktop no es posible debido a limitaciones de los fabricantes.

**SOLUCIONES:**

- |  |  |   |
|--|--|---|
| 1. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> | 5. <b>a</b> <input checked="" type="checkbox"/> <b>b</b> <input type="checkbox"/> <b>c</b> |   |
| 2. <b>a</b> <input checked="" type="checkbox"/> <b>b</b> <input type="checkbox"/> <b>c</b> | 6. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> | 9. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b>  |
| 3. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> | 7. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> | 10. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> |
| 4. <b>a</b> <input checked="" type="checkbox"/> <b>b</b> <input type="checkbox"/> <b>c</b> | 8. <b>a</b> <input type="checkbox"/> <b>b</b> <input checked="" type="checkbox"/> <b>c</b> |   |

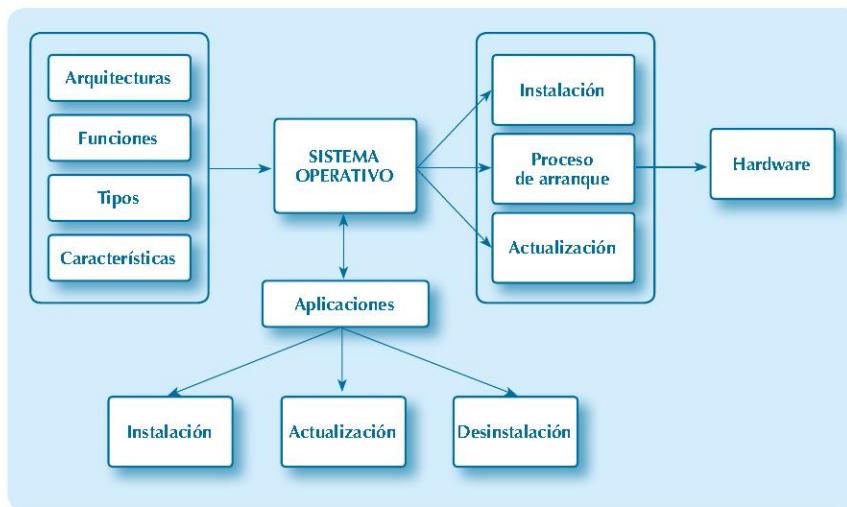
# 2

## Sistemas operativos. Introducción

### Objetivos

- ✓ Analizar las características, funciones y arquitectura de sistemas operativos a partir de los orígenes de estos.
- ✓ Comparar sistemas operativos según sus características, arquitecturas, requisitos y campos de aplicación.
- ✓ Conocer el procedimiento de instalación y actualización de sistemas operativos libres y propietarios.
- ✓ Entender y gestionar los procesos de arranque de sistemas operativos libres y propietarios.
- ✓ Comprender y realizar procedimientos asociados a la gestión de aplicaciones sobre sistemas operativos: instalación, desinstalación y actualización.
- ✓ Utilizar máquinas virtuales para instalar y probar sistemas operativos.

### Mapa conceptual



### Glosario

- BOOTMGR.** Gestor de arranque de la familia Microsoft Windows NT.
- GPT.** Tabla de particiones GUID de los sistemas con estándar UEFI.
- GRUB.** Gestor de arranque propio de los sistemas operativos GNU/Linux.
- HAL.** Parte del núcleo del sistema operativo que abstrae la parte hardware del sistema para poder trabajar, independientemente de la máquina donde sea instalada.
- MBR.** Esquema de particionamiento del estándar BIOS.
- Microkernel.** Tipología de sistema operativo cuyo objetivo es restringir el uso del procesamiento en modo núcleo, facilitando su evolución y mantenimiento.
- Multiprogramación.** Técnica de procesamiento que consiste en cargar varios programas en la memoria del computador para incrementar el uso de la CPU.
- Núcleo o kernel.** Subconjunto software del sistema operativo que por su importancia en la gestión del sistema no puede abandonar la memoria principal.
- Sistema operativo en tiempo real.** Sistema operativo adecuado a procesos que se ejecutan en unos plazos concretos y con un comportamiento predecible.
- UEFI.** Estándar que define la EFI (Extensible Firmware Interfaz), es decir, una interfaz a medio camino entre el sistema operativo y el firmware, mejorando el estándar BIOS.

## 2.1. Introducción

Cuando trabajamos con un computador con sistema operativo, ya sea un móvil, una tableta o un supercomputador, el usuario no se tiene que preocupar de las direcciones de memoria RAM usadas, de la gestión de las interrupciones, de la interfaz gráfica o cómo trabajan internamente los dispositivos de almacenamiento no volátiles.

Los sistemas operativos actuales están compuestos por un conjunto de software muy avanzado que trata de facilitar el empleo del dispositivo al usuario lo máximo posible e intentan menoscabar lo menos posible los recursos hardware.

En este capítulo se abordarán las principales funciones, características y arquitecturas de los sistemas operativos. Además, estos se clasificarán partiendo de conceptos que se han venido desarrollando desde el origen de los sistemas operativos hasta la actualidad, y han determinando su arquitectura.

Profundizaremos en los procedimientos de instalación de los sistemas operativos Microsoft Windows y Ubuntu Desktop sobre máquinas virtuales en Oracle VM VirtualBox, para más tarde estudiar los procesos de arranque y su actualización. Por último, trataremos la gestión de las aplicaciones sobre dichos sistemas operativos.

## 2.2. Funciones y características

Las funciones básicas de un sistema operativo son:

1. Actuar de *interfaz* entre el usuario y el hardware de manera transparente para el primero. Debe ofrecer soporte a los usuarios para que sus acciones se transmitan con facilidad. Los usuarios no tienen por qué ser especialistas de software o hardware para usarlo.
2. *Gestionar* los recursos software y hardware del equipo. El uso eficiente de los recursos es primordial puesto que son limitados. Dependiendo del fin y las tareas encomendadas al sistema informático, la eficiencia puede redirigirse a acciones diferentes. Por ejemplo, la eficiencia buscada en un equipo de sobremesa en nuestro hogar es diferente a la eficiencia de un sistema que gestione un conjunto de alarmas en tiempo real.

El sistema operativo es un software con características particulares, ya que debe administrar todos los recursos del sistema entre los usuarios y el resto de software. Por tanto, las características fundamentales que debe soportar cualquier sistema operativo genérico son:

- ✓ *Adaptabilidad*: se debe acomodar a dos situaciones que evolucionan en paralelo, nuevo software y nuevo hardware. El sistema operativo debe ser capaz de reacondicionarse (normalmente mediante actualizaciones) para hacer uso de nuevas características o mejoras, tanto en componentes físicos como software.
- ✓ *Facilidad de uso*: teniendo como referente el fin al que se empleará el sistema informático, la facilidad de manejo ha de ser primordial. Normalmente, una mayor comodidad implica mayor gasto de recursos (como por ejemplo un sistema gráfico de ventanas). Por ello, existen sistemas operativos que ganan en eficiencia a costa de restringir su manejabilidad.
- ✓ *Eficiencia*: los recursos (procesadores y núcleos, RAM, acceso a discos, red o cola de impresión) son limitados. El sistema operativo debe atender todas las peticiones de usuarios, programas y el propio sistema operativo para facilitar el acceso a los recursos. Ello debe

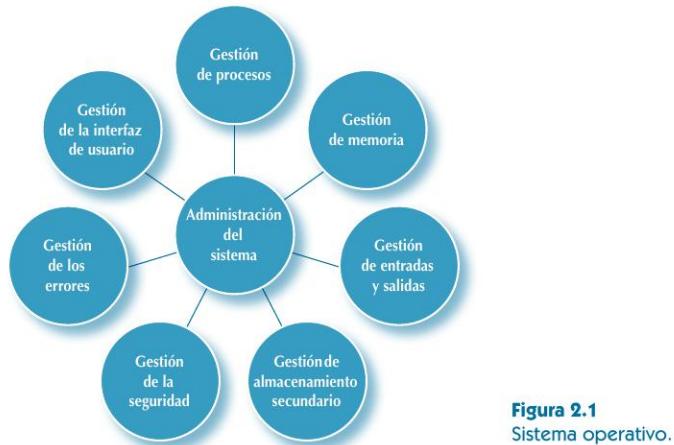
de hacerse barajando la importancia de cada solicitud y de quién desea hacer uso de los recursos. Esta tarea es muy compleja y crítica, ya que repercutirá en todo el sistema.

#### TEN EN CUENTA

- ✓ El propio sistema operativo es software y, por tanto, también consume recursos, que normalmente son muchos, si lo comparamos con la mayoría del software utilizado por un usuario común.

El sistema operativo debe administrar de forma eficiente los recursos, atendiendo al objetivo de dicho sistema operativo. Los más solicitados son:

- Memoria RAM. La parte del sistema operativo que siempre reside en memoria RAM se denomina *núcleo* o *kernel*. Es un subconjunto software del propio sistema operativo que por su importancia en la gestión del sistema no puede abandonar la memoria principal. El resto de módulos del sistema operativo se irá cargando y descargando desde los dispositivos de almacenamiento secundario a la memoria principal, dependiendo de la arquitectura del sistema operativo. El espacio restante de memoria RAM se debe gestionar eficientemente para albergar el resto de software y los datos que maneje este.
- Procesador. Aunque disponga de varios núcleos y, por tanto, pueda ejecutar varios procesos a la vez, existe multitud de software que desea ejecutarse.
- Adaptadores de red. Múltiples aplicaciones hacen uso de la red simultáneamente, debiendo administrar las conexiones de red entre aplicaciones, procesos y usuarios.
- Medios de almacenamiento. El acceso a discos duros puede representar un cuello de botella importante.
- Colas de impresión. Pueden existir más de una petición de impresión a una misma impresora, por lo que se debe gestionar la cola de trabajos de impresión adecuadamente.



**Figura 2.1**  
Sistema operativo.

La administración del sistema por parte del sistema operativo se divide en:

- a) *Gestión de procesos.* El procesador, como recurso fundamental del sistema, ha de repartir su tiempo entre los diferentes procesos que deseen ejecutarse. El sistema operativo debe organizar el paso de estos procesos por el procesador (o procesadores) y sus núcleos, de tal manera que los tiempos de ejecución de las diferentes tareas sigan los objetivos del sistema operativo. Por tanto, el sistema operativo debe gestionar:

- La asignación de procesos a varios procesadores (si dispone de varios).
- El uso de la *multiprogramación* sobre procesadores individuales y sus núcleos.
- La ejecución de una aplicación o proceso en cuanto a su sincronización con otros procesos o hilos.

Estos objetivos son definidos por políticas de planificación con orientaciones diferentes:

- *Planificación orientada a los usuarios (orientada a las entradas y salidas):* intenta agilizar las acciones de procesos como accesos a discos, señales de pantallas táctiles o accesos a Internet. Prima el tiempo de respuesta a los usuarios.
- *Planificación orientada al sistema (orientada a procesos de cálculo):* su objetivo es la eficiencia y el rendimiento de procesamiento. Un ejemplo de ello es lo que ocurre cuando se intenta acaparar el procesador durante mucho tiempo para resolver cálculos aritméticos o lógicos intensos.

- b) *Gestión de memoria.* Íntimamente ligado a la gestión de procesos se encuentra la de memoria. Por gestión de memoria se entiende la planificación y gestión global de la memoria principal con extensión a la memoria secundaria. Hoy en día los sistemas disponen de memoria RAM suficiente para albergar el sistema operativo y mucho más software. Pero también se debe planificar cómo actuar en caso de necesitar mayor espacio de memoria empleando el almacenamiento permanente. El sistema operativo amplía virtualmente la memoria RAM, tomando prestado del disco duro espacio como si fuese una extensión de la primera (a este concepto se denomina *memoria virtual*). Toda la transferencia de información entre memorias requiere una planificación vital para ahorrar tiempo y no lastrar la eficiencia del sistema.

- c) *Gestión de entradas y salidas.* Acciones como tocar una pantalla táctil, imprimir un documento, acceder a un fichero del disco duro o navegar por Internet requieren que el sistema operativo necesite administrar dichos recursos, ofreciendo soluciones rápidas y de la forma menos costosa posible. Cada dispositivo de E/S tiene una forma peculiar de interaccionar con el sistema operativo, y este ha de gestionarlo estableciendo un diálogo claro y fluido.

- d) *Gestión de almacenamiento secundario.* Los discos duros son dispositivos de E/S por sí mismos, pero la gestión de los archivos y directorios como elementos atómicos en ellos es fundamental. La estructura organizativa de los archivos y su gestión viene determinada por los sistemas de archivos.

- e) *Gestión de la seguridad.* Se deben evitar actuaciones originadas por errores software, errores hardware o por actuaciones maliciosas de usuarios, ya sean intencionadas o no, dando lugar a inconsistencias en el sistema. Por ello, el sistema debe garantizar:

- El servicio y la disponibilidad de sus recursos.
- La confidencialidad, protección e integridad del sistema y los datos.

- El control de accesos.
  - La autenticidad en las acciones.
- f) *Gestión de los errores.* Es un elemento fundamental en todo sistema operativo. El control de la totalidad de las acciones que puedan derivarse del software de terceros, el hardware y el propio sistema operativo es prácticamente imposible. Por ello, el sistema operativo debe gestionar todo tipo de errores de la manera más liviana posible, informando al usuario y salvaguardando de forma prioritaria la seguridad del sistema y los datos.
- g) *Gestión de la interfaz de usuario.* Todas las acciones encomendadas al sistema operativo tratadas hasta ahora no tendrían sentido sin una interfaz que permita una clara manejanbilidad del sistema. Por tanto, los sistemas operativos con interfaz gráfica o textual deben ofrecer un soporte que permita una fluida comunicación, así como realizar todas las acciones necesarias para la gestión, administración o explotación del mismo.



### Recurso digital 2.1

El origen de los sistemas operativos.

### 2.3. Tipos de sistemas operativos

Los objetivos de los sistemas operativos marcan la eficiencia en el uso al que se destine el sistema. Se pueden diferenciar tipologías de sistemas operativos con objetivos antagónicos entre sí, aunque en la práctica podamos encontrar versiones intermedias muy variadas.

Existen distintos puntos de vista para catalogar los sistemas operativos:

- a) Atendiendo al número de procesos que se pueden ejecutar concurrentemente:
  - *Monotarea o monoprogramado:* un proceso únicamente puede ser ejecutado por un usuario. Esto quiere decir que un usuario solo puede estar ejecutando un programa, además del propio sistema operativo.
  - *Multitarea o multiprogramado:* un usuario puede ejecutar varios procesos simultáneamente. De esta manera, pueden existir varios programas en memoria susceptibles de ser ejecutados.
- b) Atendiendo al número de usuarios que pueden ser atendidos por el sistema operativo simultáneamente:
  - *Monousuario:* solo pueden atender a un usuario. El usuario goza de todos los recursos, a menos que el sistema operativo los acapare.
  - *Multiusuario:* pueden atender a más de un usuario concurrentemente. Por tanto, los recursos del sistema deben ser gestionados para todos ellos.

## TOMA NOTA



Los sistemas operativos multiusuario son multitarea, puesto que tratan con diferentes procesos asociados a varios usuarios. Por tanto, un sistema operativo multiusuario y monotarea, puede tratar con varios usuarios simultáneamente, pero con un único proceso por usuario.

Es de reseñar que pueden existir sistemas multiusuario y monotarea, así como multitarea y monousuario.

- c) Atendiendo al tipo de procesamiento: el sistema operativo ha de estar preparado para ejecutar procesos con diferentes finalidades y requisitos. Los sistemas operativos intentan optimizar sus recursos, independientemente de los procesos que atiendan. Sin embargo, los procesos, según su forma de ejecutarse, pueden ser:
- De tiempo real: requieren unos plazos en su ejecución o tiempos de respuesta.
  - Interactivos: requieren de la participación del usuario.
  - Por lotes, batch o no interactivos: se suministra un conjunto de tareas al sistema operativo con características similares, y este se encarga de ejecutarlas en serie y sin la intervención del usuario. En caso de producirse un error en una tarea del lote, el resto de tareas no se podrá ejecutar. Ejemplos: realización de facturas agrupadas, tareas de cómputo en investigación, envío de mensajes con informes o resúmenes en cadenas de producción, etc.

## TEN EN CUENTA

- ✓ Por tanto, y de manera general, los procesos que "no son propios" de dicho sistema operativo son penalizados. Es decir, si en un sistema operativo de tendencia interactiva se lanza un conjunto de tareas interactivas y batch, las segundas serán penalizadas, en cuanto a sus tiempos de ejecución.

De esta manera, existen sistemas operativos más orientados a uno u otro tipo de proceso, puesto que la eficiencia de estos se planifica desde el diseño de los mismos:

- *Sistemas operativos en tiempo real*: donde se deben cumplir escrupulosamente los plazos de ejecución de los procesos y, además, deben tener un comportamiento predecible. Ejemplos: en aviación, instrumentación médica, sistemas de alertas en una central nuclear, etc.
- *Sistemas operativos interactivos o de tiempo compartido*: orientados a la participación continua del usuario, los cuales hacen uso de los programas antes comentados, tales como un procesador de textos o un editor de imágenes. Son sistemas de propósito general en los que, a diferencia de los sistemas de tiempo real, no priman los tiempos de respuesta en la ejecución de procesos. En esta clasificación se encuentran los más conocidos por nosotros como las diferentes versiones de escritorio y de red de Microsoft Windows o de Apple (Mac OS), así como distribuciones Linux, como Ubuntu.

d) Atendiendo al sistema de interfaz empleado:

- *Textuales*: emplean un repertorio de comandos que se introducen en el sistema de forma escrita a través de un terminal de órdenes. Aunque, se necesitan mayores conocimientos de sintaxis y manejo del sistema operativo, las acciones pueden llegar a ser muy potentes desde un punto de vista de explotación del sistema operativo.
- *Gráficos*: usan un conjunto de ventanas, botones y desplegables gráficos donde se representan los diferentes volúmenes, unidades y sistemas de ficheros de forma muy intuitiva. Además, los programas lanzados presentan una vista gráfica. El manejo se realiza con un dispositivo de entrada/salida, como un ratón, y destaca por su fácil utilización. Este sistema emplea muchos más recursos que el textual a nivel de procesador, memoria e incluso, en algunos casos, se necesita de manera casi obligada un adaptador gráfico. Por tanto, en sistemas operativos donde se busca ahorrar todo tipo de recursos en favor de atender a peticiones de usuarios y procesos, la interfaz gráfica se desprecia.

e) Atendiendo a la forma de ofrecer los servicios:

- *Sistemas operativos cliente o de escritorio*. Se encargan de realizar el procesamiento de la información, la gestión de los procesos, de la memoria, dispositivos de E/S de una sola computadora. Esta computadora suele estar conectada en red, pero el usuario es consciente de sus accesos externos. En un entorno corporativo, se pueden emplear prácticamente para compartir archivos en red. Por tanto, este tipo de sistema operativo es el normalmente empleado en un hogar o pequeña oficina, así como en entornos empresariales en el ámbito de un servicio de directorio en una red distribuida.
- *Sistemas operativos en red*. Se encargan de gestionar la red, los usuarios y los recursos de una red de computadoras en general, de forma centralizada mediante un servidor o varios como réplicas o extensiones del primero. Es en el servidor donde se instala este sistema operativo. El resto de equipos de la red (con sistemas operativos cliente) se conectan al servidor (de forma consciente) formando parte del sistema e interactuando con él. Su principal objetivo es el intercambio de información centralizada. Sin embargo, el servidor puede resultar un cuello de botella si cae o si se deteriora la transferencia de información. Destacan por su seguridad y robustez en la administración general del sistema y la gestión de la información que gestionan frente a los sistemas operativos de escritorio.
- *Sistemas operativos distribuidos*. A diferencia de los anteriores, actúan varios computadores de manera transparente al usuario, de forma que da la sensación que este interactúa solo con uno de ellos. Por tanto, permiten emplear los recursos de varias computadoras en paralelo.

**RECUERDA**

- ✓ Hemos de diferenciar entre el tipo de sistema operativo en sí y la clasificación hardware del equipo donde se instale. Es decir, los sistemas operativos de escritorio, en red o distribuidos pueden trabajar con equipos de tipo micro-computadores, mainframes, supercomputadores, etc.

 PARA SABER MÁS

Los sistemas operativos distribuidos presentan muchas ventajas, aunque destacan por su:

- Escalabilidad: es relativamente sencillo ampliar la potencia de cálculo y los recursos del sistema, puesto que se pueden añadir más computadoras.
- Confiabilidad: en caso de que una computadora falle, el resto puede hacerse cargo de las tareas que se van a realizar.

Debido a la complejidad en el diseño e implementación (principalmente por el concepto de transparencia) de los sistemas operativos distribuidos, estos no se han popularizado y desarrollado como tales. Sin embargo, muchas de sus ideas se han aplicado a los sistemas operativos de escritorio y en red. Existen pocos ejemplos en la actualidad, destacando *Plan 9* y *Amoeba*. En cualquier caso, se consideran herramientas de estudio e investigación.



- 2.1.** Accede a las páginas web de QNX <https://blackberry.qnx.com/> y LynxOS <http://www.lynx.com/>. Lee ambas páginas y comenta qué usos tienen estos sistemas operativos.
- 2.2.** Busca en Internet dos versiones de sistemas operativos únicamente textuales y explica por qué no presentan interfaz gráfica. Busca dos versiones gráficas de sistemas operativos.

## 2.4. Arquitecturas de los sistemas operativos

La arquitectura de los sistemas operativos ha ido evolucionando de la mano del desarrollo hardware de los sistemas informáticos. Ambas partes no pueden funcionar de forma aislada y dependen la una de la otra.

A lo largo de los años se han sucedido varias tipologías de arquitecturas en el desarrollo de los sistemas operativos, cada una con sus ventajas e inconvenientes y estando orientadas a propósitos diferentes. Si bien es cierto que la evolución de los propios sistemas operativos ha tomado ideas de arquitecturas o modelos anteriores para fusionarlos y hacerlos propios en beneficio de nuevos sistemas operativos.

### 2.4.1. Sistemas con capas o anillos

Presentan una estructura interna llamada *jerárquica, en niveles o en capas*. Se puede decir que están formados por un conjunto de anillos concéntricos que representan servicios o funciones diferentes. Cada capa solo se puede comunicar con la capa inmediata inferior o superior para solicitar servicios o resolver peticiones, respectivamente. Su principal ventaja es el uso de una

estructura bien definida que facilita la corrección de errores, pero resulta lento y complejo al definir las capas. Ejemplo de ello son los sistemas operativos THE y MULTICS, ambos en desuso.



**Figura 2.2**  
Arquitectura genérica  
de un sistema operativo.

En su estructura podemos distinguir las siguientes partes:

- a) Núcleo o kernel: capa que interactúa directamente con el hardware y está formada por los componentes esenciales del sistema operativo debido a su relevancia y frecuencia de uso. Se encuentra cargado permanentemente en memoria principal. Una parte del núcleo se encarga de abstraer la parte hardware del sistema para que el sistema operativo trabaje independientemente de la máquina donde sea instalada. A esta parte se le llama HAL (Hardware Abstraction Layer).
- b) Servicios: formada por un conjunto de funciones básicas que dan soporte a la capa superior para que interactúe con el núcleo. En esta capa se incluye de manera más o menos diferenciada las siguientes funciones:
  - Gestión de procesos.
  - Gestión de memoria.
  - Gestión de la E/S.
  - Gestión de almacenamiento secundario.
- c) Interfaz: constituida principalmente por un intérprete de órdenes cuya función es traducir y trasladar las acciones deseadas por un usuario a las capas inferiores. En este mismo nivel, aunque de manera diferenciada, se pueden catalogar los “Programas de usuario”, es decir, cualquier aplicación o software que instalamos en nuestro equipo y que nos permite realizar tareas concretas.



#### SABÍAS QUE...

En 1972 se reescribió UNIX en C y se puso a disposición de organizaciones, compañías, universidades y el gobierno de EE. UU. Esto provocó que su uso y desarrollo creciera enormemente, surgiendo así multitud de versiones inspiradas en UNIX. Uno de los ejemplos más significativos fue el desarrollo de la Universidad de California en Berkeley (EE. UU.), llamado *Berkeley Software Distribution (BSD)*. A partir del cual se desarrollarían sistemas operativos como NetBSD, FreeBSD, Mac OS X o SunOS.

En este enlace a Wikipedia puedes consultar un gráfico con el desarrollo de sistemas operativos basados en UNIX:





### Actividad propuesta 2.3

Averigua en Internet la relación de Ken Thompson y Dennis Ritchie con los sistemas operativos MULTICS, UNICS y UNIX.

#### 2.4.2. Sistemas monolíticos

Su nombre procede de los sistemas que tenían una única estructura, es decir, un gran programa dividido en rutinas (subprogramas), en la que todas ellas tenían los mismos privilegios (ejecutándose en modo supervisor) y se podían llamar unas a otras. Se ejecutaba en un espacio de direcciones de memoria principal único y compartido por las diferentes rutinas. Por ello, es sencillo su diseño y, sobre todo, su rendimiento o velocidad. Ejemplos de ello fueron los sistemas operativos DOS y las primeras versiones de UNIX.

A día de hoy, los sistemas operativos basados en sistemas monolíticos han mejorado, dejando atrás sus mayores inconvenientes: difícil evolución y resolución de errores y baja estabilidad. Un ejemplo de sistema operativo monolítico es Ubuntu.

#### 2.4.3. Microkernel

Su principal propósito es el de liberar al núcleo del máximo de su funcionalidad. Se pretende restringir el uso del modo supervisor (o modo núcleo) y facilitar la evolución y el mantenimiento del sistema operativo. De esta manera, el kernel se encargaría básicamente de:

- ✓ La gestión de la memoria.
- ✓ Gestiones prioritarias de procesos e hilos.
- ✓ Control básico de la comunicación entre el resto de procesos o servicios.

El resto de servicios quedarían fuera del núcleo, ahora ejecutándose en modo usuario, como, por ejemplo, la gestión de archivos, los protocolos de comunicaciones o los drivers de dispositivos.

La idea es que un proceso cliente, como, por ejemplo, una aplicación de usuario cualquiera, desea obtener servicio de un proceso servidor del sistema operativo. Para ello, la primera envía un mensaje a la segunda a través del micronúcleo, y el micronúcleo es el que se encarga de la comunicación y gestión necesaria para que todos los clientes sean atendidos con eficiencia por los diferentes servidores. De esta manera, tanto clientes como servidores se ejecutan en modo usuario, y una pequeña parte de todo el proceso (la más crítica), en modo núcleo.

Con esto se mejora:

- La seguridad del sistema operativo, al ejecutarse la mayoría de los procesos en modo usuario.
- La estabilidad.
- La actualización del sistema operativo.

Sin embargo, uno de los principales defectos de esta arquitectura es la posible sobrecarga en la gestión de procesos que ocasiona un deterioro en el rendimiento del sistema. Un ejemplo de sistema operativo microkernel es MINIX.



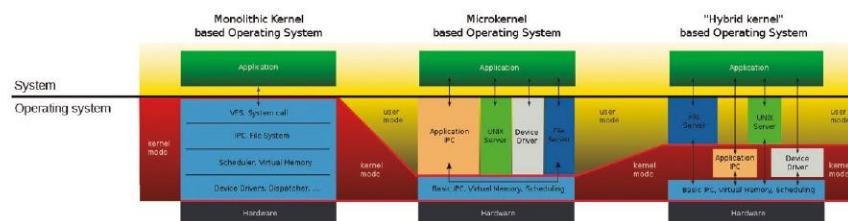
## Recurso digital 2.2

El sistema operativo MINIX.

### 2.4.4. Kernel híbrido

Se considera una evolución que aúna las arquitecturas monolítica y microkernel, persiguiendo las ventajas de ambas. Consiste en un diseño microkernel, pero con una implementación monolítica, que consigue una gran estabilidad y un significativo rendimiento (como ventajas de ambos modelos, respectivamente).

A diferencia de los sistemas microkernel, los sistemas híbridos añadirían en su espacio kernel los drivers de dispositivos y todo lo relativo a la comunicación entre procesos, como servicios fundamentales para ejecutar en modo supervisor.



**Figura 2.3**

Comparativa entre sistemas operativos monolíticos, microkernel e híbridos.

### 2.4.5. Arquitecturas de sistemas operativos actuales

No se pueden clasificar los sistemas operativos más empleados actualmente (MAC OS, Windows o Linux) en una arquitectura claramente definida, aunque sí se puede decir que sean tendentes a una de ellas. Por ejemplo:

- ✓ Mac OS: híbrido.
- ✓ Windows de la familia NT: híbrido.
- ✓ Ubuntu: monolítico.

www

## Recurso web

Para saber más, se recomienda la lectura del artículo “Cómo es el kernel de Windows y cuáles son sus diferencias con el de Linux” (Genbeta).



## 2.5. Versiones de los sistemas operativos más utilizados

Los sistemas operativos, al igual que cualquier otro tipo de software, están asociados a una licencia (las cuales serán estudiadas en el capítulo 7). Los sistemas operativos comerciales más utilizados disponen de versiones o distribuciones para las siguientes plataformas, principalmente: equipos de escritorio, servidores y dispositivos móviles.

### 2.5.1. Sistemas operativos de Microsoft

En el caso de los sistemas operativos de la compañía Microsoft, nos encontramos con multitud de versiones cuyo mercado se centra principalmente en las siguientes plataformas:

1. *Para equipos de escritorio: Microsoft Windows 10.* Incluye multitud de ediciones orientadas a diferentes ámbitos que, según las orientaciones, dispone de mayores prestaciones. Destacamos:
  - Home: para equipos de sobremesa, tabletas o portátiles de poca potencia, para uso básico (multimedia y conectividad).
  - Pro: orientado a fines de negocio para empresas o profesionales.
  - Enterprise: también orientado a fines de negocio, pero de mayor volumen.
  - IoT: ideada para dispositivos relacionados con el Internet de las Cosas.
  - Education: dirigido a un entorno académico.
  - Pro for Workstations: para equipos muy potentes con grandes cargas de trabajo intensivo o tareas críticas.
2. *Para equipos de tipo servidor: Microsoft Windows Server 2019.* Entre las que destacamos las siguientes ediciones:
  - Datacenter: para entornos en la nube o centros de datos altamente virtualizados.
  - Standard: para ambientes poco virtualizados.
  - Essentials: para pequeños negocios con un número limitado de usuarios y dispositivos.

### 2.5.2. Sistemas operativos GNU/Linux

Las distribuciones de sistemas operativos GNU/Linux son muy variadas, existiendo multitud de versiones en cada una de ellas. Algunas de las distribuciones más empleadas para servidores son:

- Red Hat Enterprise Linux.
- Ubuntu Server.
- CentOS.
- SUSE Linux Enterprise Server.
- Debian.
- FreeBSD.

En el caso de distribuciones para equipos de sobremesa o portátiles, son enormes las variedades según el uso del equipo (genérico, seguridad, juegos, ligereza, orientado a la nube). Algunas de las distribuciones más empleadas son:

- ✓ *Ubuntu* y *Mint*: muy genéricos y versátiles, de gran facilidad de uso.
- ✓ *Arch Linux*: distribución personalizable para usuarios avanzados.
- ✓ *Kali Linux* y *Tails*: orientados a la seguridad y la privacidad.
- ✓ *Chromium OS*: versión liberada de Chrome OS (sistema operativo en la nube de Google) con licencia BSD.
- ✓ *Manjaro*: procedente de Arch Linux, está destinado a la facilidad de uso mediante un modelo de actualización continua.
- ✓ *Android*: opción archiconocida para smartphones.

### Recurso digital 2.3

El origen de las distribuciones GNU/Linux.

#### 2.5.3. Sistemas operativos de Apple

Por otro lado, la empresa Apple Inc. desarrolla sistemas operativos para portátiles, equipos de sobremesa, servidores, móviles y otros dispositivos conectados para hardware específico. Sus versiones más utilizadas son:

- a) *macOS*: sistema operativo de escritorio y equipos portátiles.
- b) *iOS*: para sus smartphones.

#### Actividades propuestas



- 2.4.** La página web <https://distrowatch.com/> aglutina mucha información y permite comparar multitud de distribuciones GNU/Linux, BSD y Solaris. ¿Cuáles son las diez distribuciones más populares actualmente, según el ranking ofrecido por esta página? Establece una comparativa, indicando: tipo de sistema operativo, en qué sistema operativo está basado y para qué plataformas o entornos (seguridad, sobremesa, servidores).
- 2.5.** El bajo coste, la filosofía de software abierto para desarrollar cualquier proyecto y la robustez de los sistemas operativos GNU/Linux (estabilidad, flexibilidad al optimizar los recursos y seguridad) hacen de estos los dominadores en supercomputadoras. No obstante, también se emplean para otros entornos, como servidores, desktop o dispositivos móviles.

En la web <https://www.top500.org/> encontramos una lista actualizada de los quinientos sistemas de computación más potentes. Realiza una tabla con los cinco primeros, donde se indique:

- Localización: ciudad y país.
- Fabricante.
- El número de núcleos.
- El número de operaciones máximas en TFlops.
- El sistema operativo empleado.

## 2.6. Instalación de un sistema operativo

El sistema operativo, como tal, no deja de ser un software que ha de instalarse en el equipo donde se desee ser explotado. Su instalación, a diferencia del software de aplicación, es crítica, puesto que se sustenta en el hardware del sistema, así como en el firmware del mismo. Del buen estado del sistema operativo, en simbiosis con el hardware, dependerá el resto del software del sistema.

### 2.6.1. Requisitos

Cada sistema operativo establece unos *requisitos mínimos* para poder ejecutar el sistema operativo y usarse. Junto con los requerimientos mínimos, los propietarios suelen establecer unos *requisitos recomendables*, más en sintonía con la eficiencia y la ligereza del sistema en condiciones de cierto estrés.

Los requisitos mínimos para la instalación de Windows 10 Pro y Enterprise versión 1809 son:

**CUADRO 2.1**  
**Requisitos mínimos de Windows 10 Pro y Enterprise**

Procesador necesario	Procesador a 1 GHz o más rápido
Memoria necesaria	1 GB de RAM para 32 bits; 2 GB para 64 bits
Espacio en disco duro necesario	Hasta 20 GB disponibles
Tarjeta de vídeo necesaria	Resolución de pantalla de 800 x 600 o superior. Procesador de gráficos DirectX 9 con controlador WDDM
Conectividad necesaria	Acceso a Internet



### Actividad propuesta 2.6

Investiga y compara los requisitos mínimos de instalación para la última versión de Ubuntu Server y una edición de Windows Server.

Los requisitos recomendables para Ubuntu 18.10 LTS son:

**CUADRO 2.2**

**Requisitos recomendados de Ubuntu 18.10 LTS**

Procesador	Procesador a 2 GHz dual core o superior
Memoria	2 GB
Espacio en disco duro	25 GB disponibles
Tarjeta de vídeo	Resolución de pantalla de 1024 x 768
Conectividad	Recomendable tener acceso a Internet

### 2.6.2. Planificación y consideraciones previas

Antes de la instalación del sistema operativo, se debe seguir una planificación del proceso:

1. Hemos de asegurarnos que se cumplen los requisitos de instalación establecidos por el fabricante del sistema operativo: procesador, cantidad de memoria RAM, espacio libre en disco duro, sistema gráfico, conectividad con Internet, etc.
2. Realizar una copia de seguridad de los datos de las particiones de aquellos discos duros implicados en el proceso de instalación.
3. Diseñar el proceso de arranque del equipo si queda conformado por más de un sistema operativo. En tal caso, hemos de decidir dónde instalar el gestor de arranque del sistema.
4. Mantener, en la medida de lo posible, una alimentación eléctrica redundante. En el caso de equipos portátiles, estos se deben conectar a la corriente por si agotan la carga de la batería. Además, si disponemos de sistemas de alimentación ininterrumpida (SAI), debemos hacer uso de ellos.
5. Recopilar todos los componentes software pre y post instalación:
  - Imagen del sistema operativo que se va a instalar en algún medio de almacenamiento.
  - Drivers de todos los componentes hardware del sistema: placa base y chipset, periféricos, tarjetas adaptadoras, etc.
  - Aplicaciones para instalar tras la instalación.
  - Cambiar el orden de arranque (BOOT) del sistema a través del BIOS Setup Utility o del menú rápido de arranque.
6. Planificar y establecer la instalación del sistema operativo con un esquema de particionamiento GPT (UEFI) o MBR (BIOS tradicional o heredado). Para ello, debemos configurar el modo de arranque en el BIOS Setup Utility, indicando 'UEFI' o 'Heredado'. El modo de arranque, así como el medio de instalación y los discos donde se desee instalar el sistema operativo, deben encontrarse con el mismo esquema de particionamiento para evitar problemas durante la instalación.

A continuación, vamos a estudiar los procesos de instalación de Ubuntu Desktop y Microsoft Windows 10 Pro a través de imágenes de instalación .iso descargadas previamente desde las páginas oficiales de los respectivos sistemas operativos.

Bios tradicional>>Particionado  
MBR(tradicional)  
Bios UEFI>> Particionado GPT  
Legacy >> tradicionado



### Actividad propuesta 2.7

Crea una unidad flash USB arrancable Ubuntu sobre Microsoft Windows, siguiendo los pasos dados en los tutoriales oficiales de Ubuntu.



### 2.6.3. Proceso de instalación de Ubuntu Desktop en Oracle VM VirtualBox

Ya conocemos el entorno de trabajo de Oracle VM VirtualBox, donde suponemos que hemos creado una máquina virtual para la instalación de Ubuntu Desktop, tal y como se ha estudiado en el capítulo anterior. El siguiente paso es instalar el sistema operativo para poder hacer uso de un sistema totalmente virtualizado. En nuestro caso, descargamos de la página oficial de Ubuntu la última versión en formato .iso.

#### A) Adecuación de los medios de almacenamiento virtuales

Al pulsar el botón *Iniciar* de la máquina virtual antes creada, arranca el sistema mostrando una ventana que solicita que seleccionemos un disco de inicio donde se encuentre la imagen del sistema operativo que se va a instalar. Cuando la máquina virtual se apague, este expulsará el disco virtual de instalación, resultando temporal la selección de dicho disco virtual.

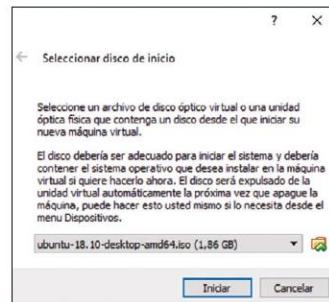
Otra forma de configurar el medio de instalación es a través de la configuración de los medios de arranque (antes de iniciar la máquina virtual) en ‘Almacenamiento’ desde ‘Configuración’.

Se debe añadir la unidad óptica pulsando en el icono de ‘Aregar unidad óptica’ para el ‘Controlador SATA o IDE’. Y, a continuación, aparece una nueva ventana donde se nos preguntará si deseamos seleccionar:

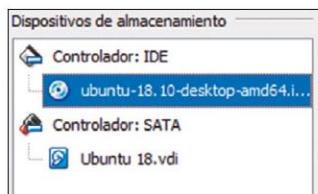
- Un disco óptico virtual. Despues indicaría mos la imagen del sistema operativo para instalar.
- Dejar vacío. Posteriormente, podríamos añadir el disco virtual pulsando en el disco ‘Vacio’, y en los ‘Atributos de la unidad óptica’, seleccionar la ruta para acceder a la imagen del sistema operativo que se va a instalar.

En cualquier caso, la configuración de los dispositivos de almacenamiento podría quedar como muestra la figura 2.5.

Cuando termine la instalación, debemos extraer el disco de instalación para evitar que se inicie la máquina virtual en sucesivas ocasiones con dicho disco.



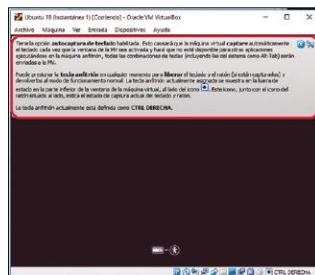
**Figura 2.4**  
Selección de disco de instalación.



**Figura 2.5**  
Dispositivos de almacenamiento.

### PARA SABER MÁS

Ya configurados los medios de arranque, al iniciar la máquina virtual, aparecen mensajes relativos a la captura del teclado en la misma. Estos indican que cuando tengamos seleccionada la máquina virtual, las acciones con el teclado serán enviadas a la máquina virtual y no a la máquina anfitriona. Si quisieramos volver a la máquina anfitriona, deberíamos pulsar la tecla 'Control' derecha del teclado (tecla Host por defecto en VirtualBox).



**Figura 2.6**  
Aviso de autocaptura de teclado.



**Figura 2.7**  
Aviso de integración del ratón.

Más tarde, cuando carga el sistema de instalación, aparece otro mensaje indicando que el sistema operativo invitado soporta integración del ratón. Así, cuando el puntero del ratón se encuentre sobre la ventana de la máquina virtual, todas sus acciones se enviarán a dicha máquina, y cuando salga de ella, el sistema operativo anfitrión será el receptor.

### B) Proceso de instalación

Al igual que una máquina real sigue el orden de arranque establecido en la BIOS, la máquina virtual sigue el orden de arranque establecido en la configuración de la máquina virtual. Al estar en la unidad óptica el disco de instalación de Ubuntu, el sistema lo toma como el medio de arranque del sistema, y comienza el proceso de carga de los archivos de instalación.

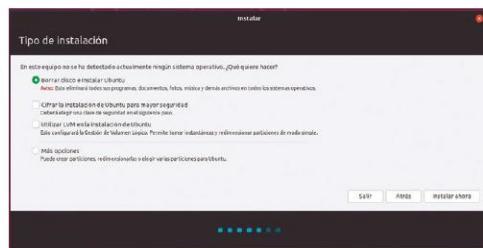
Ya cargados los archivos, el proceso de instalación comienza dando a elegir entre multitud de idiomas y si deseamos probar o instalar el sistema operativo. Probar un sistema operativo es una opción recomendable en caso de no estar seguros del rendimiento del equipo o si deseamos probar sus características. En nuestro caso, seleccionamos la opción de ‘Instalar Ubuntu’.

En la siguiente pantalla, debemos seleccionar la disposición de las teclas de nuestro teclado. Si no estuvieramos seguros, podríamos pulsar en ‘Detectar la distribución del teclado’, y el programa de instalación solicitará que pulsemos una serie de teclas para identificarlo.

A continuación, nos solicita que indiquemos una serie de aspectos de actualización e instalación de aplicaciones. Es recomendable realizar una ‘Instalación normal’, si nuestro sistema dispone de suficiente memoria RAM y espacio en el disco duro. Además, se recomienda ‘Descargar actualizaciones al instalar Ubuntu’. Y, en caso de emplear una tarjeta de red o gráfica poco habitual o usemos multitud de formatos multimedia, seleccionar ‘Instalar programas de terceros para hardware de gráficos y de Wi-Fi y formatos multimedia adicionales’.



**Figura 2.8**  
Elección de idioma.



**Figura 2.9**  
Elección del tipo de instalación.

Al continuar con el proceso de instalación, llegamos a la pantalla donde se requieren más conocimientos técnicos. Las opciones son las siguientes:

1. *Borrar disco e instalar Ubuntu*: borrará el disco objeto de instalación del sistema operativo.
2. *Cifrar la instalación de Ubuntu para mayor seguridad*: debemos valorar si nos decantamos por la seguridad del sistema cifrando la instalación, ya que repercutirá en el rendimiento.
3. *Utilizar LVM en la instalación de Ubuntu*: la gestión de volumen lógico (*logical volume manager*) permite una administración flexible de las particiones al ser lógicas y, por tanto, no físicas. Al igual que una imagen virtual de un disco duro, se pueden hacer instantáneas y redimensionarlas con suma facilidad.
4. *Más opciones*: emplearíamos una gestión manual de los discos duros del sistema en la instalación de Ubuntu.

Si seleccionamos esta última opción, se mostrará una pantalla que permite gestionar la instalación del sistema con una distribución de particiones a nuestro gusto.

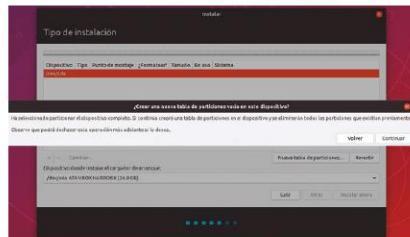
### C) Creación de particiones en el proceso de instalación de Ubuntu

Si se ha seleccionado ‘Más opciones’ en ‘Tipo de instalación’, en nuestro caso de estudio, nos encontramos con un único disco duro, */dev/sda* (nomenclatura de identificación de medios

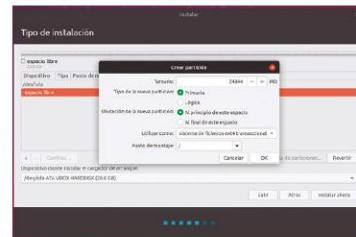
de almacenamiento en Linux), que no se encuentra particionado, por lo que debemos crear una ‘Nueva tabla de particiones’, pulsando en dicho botón.

Una vez creada, esta nos permitirá establecer el esquema de particionamiento que definamos a continuación sobre el espacio libre. Sobre él, añadimos dos particiones (pulsando en “+”):

- ✓ Una partición primaria para el sistema raíz (“/”) de todo el espacio menos 2 GB (que reservaremos para el área de intercambio).
- ✓ Una partición lógica para el área de intercambio (Swap) de 2 GB. Esta es recomendable, aunque no obligatoria. El tamaño y la creación de esta partición puede variar a nuestro criterio, dependiendo de la memoria RAM del equipo, del uso que le demos al sistema, así como del sistema operativo Linux utilizado. Esta área se emplea para aumentar el grado de multiprogramación, es decir, aumentar el número de procesos en memoria RAM y el tamaño de estos. A este concepto se conoce como *memoria virtual*.

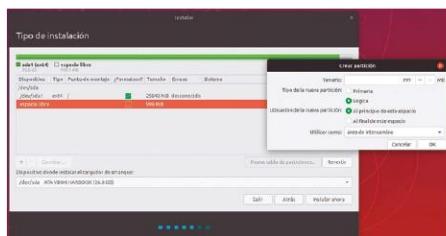


**Figura 2.10**  
Creación de la tabla  
de particiones.

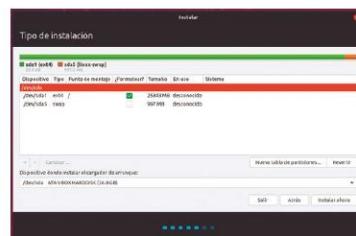


**Figura 2.11**  
Creación de la partición  
del sistema raíz.

Por último, seleccionamos dónde se instalará el lanzador del sistema operativo o cargador de arranque. En nuestro caso, en el único disco duro que disponemos, /dev/sda.



**Figura 2.12**  
Lugar donde instalar el cargador  
del sistema operativo.



**Figura 2.13**  
Creación del área  
de intercambio.

#### D) Finalización del proceso de instalación

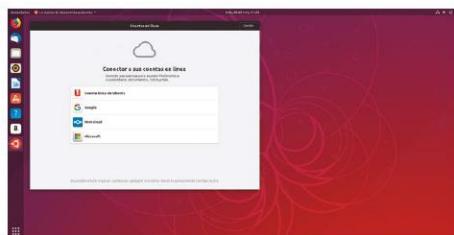
Al continuar (tanto si en el paso anterior indicamos ‘Borrar disco e instalar Ubuntu’ como si indicamos ‘Más opciones’), mostrará el resumen de las opciones seleccionadas. Y al volver a continuar, debemos indicar la franja horaria.

Por último, el proceso de instalación solicita que introduzcamos el nombre de un usuario, el nombre de la máquina, el *login* de usuario y una contraseña (figura 2.14). Como siempre, es recomendable utilizar una contraseña fuerte (con mayúsculas, minúsculas, números y caracteres especiales) para evitar accesos indebidos.

Así pues, comienza el proceso de copia y configuración de los archivos del sistema operativo.



**Figura 2.14**  
Solicitud de credenciales.



**Figura 2.15**  
Escritorio de Ubuntu.

Cuando termina el proceso de copia, muestra un mensaje de terminación de la instalación. Al pulsar en ‘Reiniciar ahora’, el sistema se reinicia, indicando que extraigamos el disco de instalación de Ubuntu, si no lo realiza automáticamente. En tal caso, podemos extraerlo desde el ícono de unidades ópticas de la barra de estado. Y, una vez reiniciado, se inicia el sistema operativo, llegando a la pantalla de autenticación de usuarios. Al introducir la contraseña de usuario, Ubuntu carga su interfaz de escritorio (figura 2.15).



#### Actividad propuesta 2.8

Realiza una instalación de Ubuntu Desktop en una máquina virtual en Oracle VM VirtualBox.

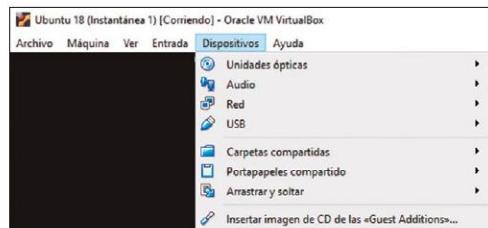
#### E) Instalación de las Guest Additions

Tras la instalación del sistema operativo invitado, es muy recomendable la instalación de las Guest Additions en él. Las Guest Additions son un conjunto de drivers y aplicaciones de sistema que optimizan el sistema operativo invitado para un mejor rendimiento y usabilidad.

Para ello, una vez cargado el entorno gráfico del usuario, insertamos el medio óptico a través de la barra de menú principal: 'Insertar imagen de CD de las Guest Additions' en 'Dispositivos'.

Se lanzará automáticamente un script autoejecutable que nos preguntará previamente si deseamos ejecutarlo.

Al autenticarnos, comenzará el proceso de instalación. Al finalizar, solicitará que pulsemos 'Return'.



**Figura 2.16**  
Opción de instalación de las Guest Additions.

**Recurso digital 2.4**

Entorno de la ventana de la máquina virtual.

#### 2.6.4. Proceso de instalación de Microsoft Windows 10 Pro en Oracle VM VirtualBox

Para instalar Microsoft Windows 10 Pro en Oracle VM VirtualBox es necesario disponer de una máquina virtual ya creada. Además, se supone que disponemos de una imagen .iso de instalación de Microsoft Windows 10 Pro, que hemos asociado a un disco óptico virtual y situado esta en primera posición del orden de arranque de la máquina virtual.

**RECUERDA**

- ✓ La instalación de Microsoft Windows también se puede iniciar sobre un sistema operativo Windows ya instalado y que se esté ejecutando con normalidad. En este caso, bastaría con introducir el medio que contenga la imagen de la instalación y ejecutarla.

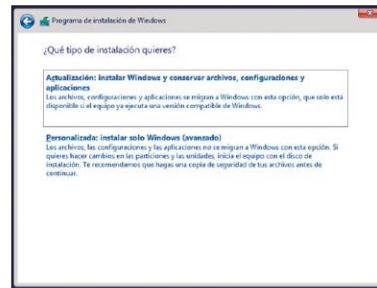
Al iniciar la máquina virtual, comienza cargando los archivos de instalación en memoria principal, mostrando una primera pantalla de selección de idiomas, formatos de fecha y moneda, así como el tipo de teclado. Más tarde nos da la opción de 'Instalar Windows' para comenzar el proceso de instalación. Además, en esta misma pantalla aparece la opción 'Reparar el equipo',

que incluye una serie de herramientas para recuperar el equipo en caso de inconsistencia del mismo.

En el siguiente paso, el asistente de instalación muestra la pantalla de activación del producto, donde se debería introducir la clave, estando formada por cinco grupos de cinco caracteres alfanuméricos. Si estamos utilizando una licencia de evaluación, este paso no se muestra.



**Figura 2.17**  
Inicio del proceso de instalación.



**Figura 2.18**  
Tipo de instalación.

Además, si disponemos de una imagen de instalación múltiple, ofrece la posibilidad de elegir entre varias versiones de Microsoft Windows.

Posteriormente, aparecen los ‘Términos de licencia y avisos aplicables’, cuya lectura es recomendable. Una vez aceptados dichos términos, preguntará sobre el tipo de instalación que deseamos.

La instalación puede ser:

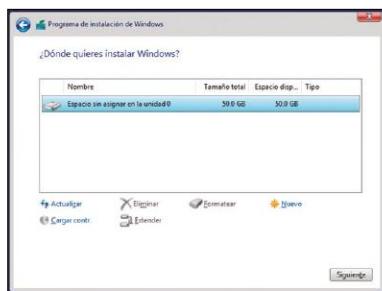
- Actualización: permite instalar Windows sobre una instalación anterior, migrando los archivos, configuraciones y aplicaciones.
- Personalizada: da la opción de crear o modificar las particiones con objeto de instalar Windows y editar las particiones en los diferentes medios de almacenamiento.

Si optamos por la opción personalizada, hemos de crear particiones pulsando en ‘Nuevo’. Se necesita al menos una partición (si no la creamos, la creará el asistente por nosotros). Antes de ser creada, indicará que se va a crear otra más para incorporar en esta archivos del sistema relacionados con el gestor de arranque del sistema operativo.

Al pulsar en ‘Siguiente’, muestra un resumen de las particiones y, si volvemos a pulsar en ‘Siguiente’, comienza la transferencia de archivos del medio de instalación a la partición objeto de instalación, se instalan las características y se actualiza.

Cuando termina de instalar Windows, este se reinicia. En este punto, la instalación ha concluido, por lo que ya podemos retirar la imagen .iso de la unidad óptica o establecer el disco duro objeto de la instalación de Windows como el primer medio en el orden de arranque.

La primera vez que se inicie tras la instalación, el asistente comenzará a configurar el sistema, preparando e inicializando servicios y dispositivos y después presentando al asistente ‘Cor-tana’, el cual nos solicita la región y la distribución del teclado.



**Figura 2.19**  
Creación de particiones.



**Figura 2.20**  
Escritorio de Windows.

Una vez configurado el sistema operativo, solicita iniciar sesión con una cuenta de Microsoft. Introducimos las credenciales, si disponemos de una cuenta de Microsoft y, en caso contrario, nos dará la opción de crear una cuenta local.

En el proceso de creación de la cuenta local, el asistente solicita un nombre de usuario y contraseña. En este proceso nos dará la opción de crear una cuenta en línea de Microsoft.

Como parte del proceso de seguridad, solicita que introduzcamos preguntas y respuestas de seguridad para usar en caso de olvidar la contraseña.

A continuación, pregunta si deseamos utilizar Cortana como asistente digital, así como una serie de aspectos relacionados con datos de diagnóstico, reconocimiento de voz y escritura, privacidad, etc. Y al finalizar todo el proceso de configuración, se inicia Windows.

### Actividad propuesta 2.9



Realiza una instalación de Microsoft Windows 10 Pro en una máquina virtual en Oracle VM VirtualBox.

## 2.7. Instalaciones desatendidas

En muchas ocasiones, no resulta práctico la instalación del sistema operativo que requiera la intervención del usuario, como, por ejemplo:

- Un administrador de sistemas que necesite instalar en multitud de equipos el mismo sistema operativo con una configuración idéntica.
- Un usuario sin experiencia que no tenga por qué conocer el proceso de instalación y posterior configuración de un sistema operativo.

Por tanto, los sistemas operativos, a través de herramientas específicas, pueden simplificar su instalación reduciendo al mínimo la intervención del usuario. Genéricamente, estas herramientas crean un archivo de respuestas que interviene cuando el proceso de instalación así lo requiere, en lugar de un usuario interactivamente.

### 2.7.1. Instalación desatendida de Windows 10

La herramienta que propone Microsoft para crear un archivo de respuestas para instalaciones desatendidas es *Windows System Image Manager (Windows SIM)*. Esta herramienta viene incluida en Windows ADK (Windows Assessment and Deployment Kit), un conjunto de herramientas orientado a comprobar la calidad y el rendimiento del sistema.

Podemos descargar Windows ADK desde <https://docs.microsoft.com/es-es/windows-hardware/get-started/adk-install>. Al proceder a la instalación de este paquete, preguntará qué herramientas deseamos instalar. Podemos encontrar Windows SIM en 'Herramientas de implementación'.

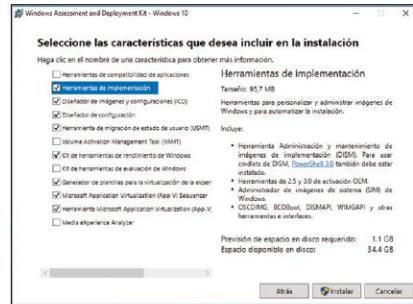
Una vez instalado, podemos buscar la aplicación ‘Administrador de imágenes de sistema de Windows’ a través del cuadro de búsqueda de Microsoft Windows.

Esta aplicación es la más potente y completa a la hora de crear imágenes desatendidas de instalación para Windows, pero requiere tener unos conocimientos profundos en cuanto a los componentes de Windows.

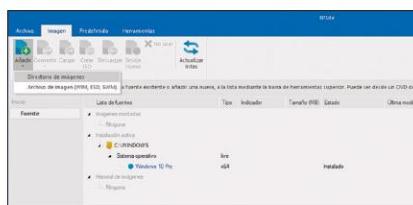
Existen otras aplicaciones que no necesitan tantos conocimientos y cuya gestión es mucho más intuitiva. Una de las más conocidas es *NTLite* (<https://www.ntlite.com/>), cuya versión gratuita es suficiente para crear una imagen desatendida. Al instalar la aplicación, solicita que indiquemos el tipo de licencia. Si elegimos ‘Gratis’ (limitado, no comercial), indica que tiene ciertas funciones y limitaciones. A continuación, se abrirá la aplicación.

Más tarde, debemos cargar una imagen sobre la que crear la imagen desatendida. En el cuerpo principal aparece la instalación activa, es decir, el sistema operativo que tenemos instalado actualmente. En nuestro caso, vamos a cargar una imagen diferente. Para ello pulsamos en el botón ‘Añadir’ y ‘Directorio de imágenes’.

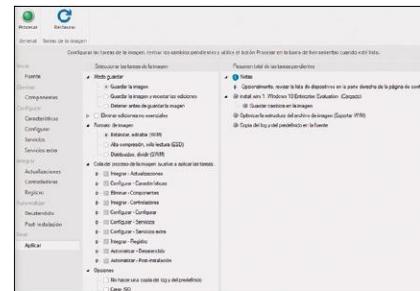
Previamente hemos tenido que descomprimir el contenido .iso de una imagen en un directorio. Al cargar dicho directorio, el historial de imágenes se actualizará.



**Figura 2.21**



## Figura 2.22 NTLite



### Configuración de tareas de la imagen en NTLite.

Ahora, pulsamos con el botón secundario del ratón sobre el sistema operativo que deseamos crear la imagen y en ‘Cargar’. Mostrará en verde el sistema operativo cargado y en el menú de la izquierda aparecerán las opciones para configurar.

A continuación, hemos de configurar todas las opciones a nuestro gusto y, para ello, podemos hacer uso de las guías de ayuda del propio programa.

La última de las opciones es ‘Aplicar’. Tras estar seguros de la configuración realizada, en este último paso podemos ver en el panel de la derecha un resumen de todas las modificaciones hechas, y en el panel de la izquierda las características de la imagen para crear.

Ahora marcamos la opción ‘Crear ISO’ (así como otras tareas de la imagen si deseamos) y luego pulsamos en el botón ‘Procesar’. Comenzará el proceso de creación de la imagen, tras preguntarnos el nombre del *.iso*. Por último, hemos de probar la imagen desatendida en una máquina virtual antes de instalarlo en una máquina física.

### 2.7.2. Instalación desatendida de Ubuntu

Para Ubuntu, existen diferentes herramientas de creación de imágenes desatendidas. No obstante, Ubuntu recomienda dos métodos de creación del archivo de respuestas necesarios para realizar la instalación desatendida:

1. Mediante la edición de un archivo de texto *preseed*. Es el método más complejo y está orientado a usuarios experimentados. La guía de instalación por este método la podemos encontrar en <https://help.ubuntu.com/lts/installation-guide/amd64/apb.html>. En este enlace también se encuentra un archivo ejemplo de respuestas, así como la ayuda de los parámetros de configuración de dicho archivo.
2. Mediante la aplicación ‘Kickstart’. Esta aplicación permitirá crear el archivo de respuestas de manera sencilla sin conocer los comandos de configuración de cada una de las opciones de preinstalación, instalación y postinstalación.

#### Actividad propuesta 2.10



Investiga a través de tutoriales oficiales de Ubuntu el proceso de creación de imágenes desatendidas y realiza una. Prueba la imagen desatendida de Ubuntu en una máquina virtual.

## 2.8. Proceso de arranque del sistema operativo. Gestores de arranque

En el capítulo anterior se estudió cómo el sistema inicia el arranque a través de la BIOS. Una vez terminado el proceso POST, inicialización y configuración de componentes, la BIOS le pasa el testigo al primer medio de almacenamiento establecido en el orden de arranque de la BIOS Setup Utility. Cuando ha encontrado un gestor de arranque, comienza la carga del sistema operativo.

### 2.8.1. Conceptos previos: esquemas de particiones

Los discos duros se deben *particionar* para organizar la información convenientemente, poder almacenar programas, datos e instalar sistemas operativos. Cuando adquirimos un disco duro y lo instalamos en nuestro equipo, normalmente se encuentra particionado. No obstante, podemos realizar particiones libremente, con programas específicos del sistema operativo o externos a él.

Para almacenar información en una partición, esta debe tener un sistema de archivos en uso. En cada partición se puede instalar un único sistema de archivos. A este proceso se le conoce tradicionalmente como *formatear*. Los sistemas de archivos confieren características al tratamiento de los datos contenidos en ellos, como, por ejemplo: *ext4*, *FAT32*, *NTFS*, *APFS* y *exFat*.

Por tanto, llamamos partición a una división del espacio de almacenamiento de forma contigua de un disco duro. Las particiones se utilizan para:

- ✓ Organización de la información: podemos estructurar la información contenida en las particiones de manera coherente.
- ✓ Eficiencia: en discos duros mecánicos, el rendimiento de la cabeza lectora mejora al tener un recorrido inferior al total.
- ✓ Instalación de sistemas operativos: los sistemas operativos obligan a instalar un sistema operativo en su propia partición (con un sistema de archivos compatible).
- ✓ Seguridad: al ofrecer espacios físicos distintos, el riesgo que corre una partición (malware, errores del sistema, errores físicos en sectores) no tiene por qué afectar al resto de particiones.

Los discos duros de nuestro equipo pueden estructurar sus particiones atendiendo a dos estándares:

#### A) MBR (Master Boot Record o Registro de Arranque Maestro)

Es el esquema de particionamiento de los sistemas con estándar BIOS. Es más antiguo, pero aún se sigue empleando por su compatibilidad con sistemas operativos.



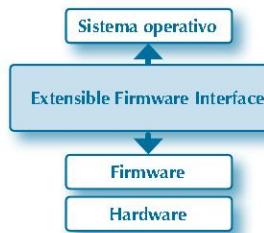
#### B) GPT (GUID Partition Table o tabla de particiones GUID)

Esquema de particionamiento de los sistemas con estándar UEFI (Unified Extensible Firmware Interface). A cada partición se le asigna un identificador global único (GUID). Este estándar mejora el estándar BIOS tradicional y solventa multitud de limitaciones, al ser más flexible, potente y fácil de usar.

- Puede contener hasta 128 particiones con un sistema operativo Microsoft Windows, eliminando las limitaciones de cuatro particiones primarias.
- Soporta discos de tamaño muy superior al esquema MBR, que limitaba en 2,2 TB su capacidad máxima, despreciando el resto.
- Permiten trabajar en modos de 32 y 64 bits.
- Inicio del sistema con mayor rapidez.
- Es mucho más seguro que los sistemas BIOS:
  - Carga el núcleo de los sistemas operativos comprobando su originalidad.
  - GPT realiza copias de la estructura de la tabla de particiones.
- Se puede conectar directamente con Internet.
- Dispone de una interfaz mucho más amigable.
- Consta de un gestor de arranque propio, es decir, no está vinculado a ningún sistema operativo.

El sistema UEFI se encuentra a medio camino entre un sistema BIOS (firmware) y el sistema operativo, por lo que ofrece así muchas ventajas gracias a su flexibilidad.

Por otro lado, la denominación BIOS se puede hacer extensible a sistemas con estándar BIOS o estándar UEFI cuando haga mención al sistema de arranque en sentido genérico.



**Figura 2.24**  
EFI en un sistema informático.

#### PARA SABER MÁS

##### Estructura de tabla de particiones GUID

Los discos con formato GPT deben tener una partición de sistema EFI (ESP). Esta partición ha de estar formateada en FAT32 y suele ser la primera. En ella se encuentran, principalmente, los cargadores de arranque de los sistemas operativos instalados, firmwares UEFI que inician sistemas operativos, imágenes kernel u otras utilidades.

Un disco con esquema GPT tiene una primera parte “Protective MBR”, que evita que programas de gestión de particiones antiguos (que no reconocen las particiones GPT) puedan dañar su estructura y los datos contenidos en sus particiones. Estos programas solo interpretan dicha parte, de tal manera que, si intentan gestionar un disco ya particionado GPT, lo verán como una única partición que ocupa la unidad (Protective MBR contiene una única entrada a una partición de 2 TB existente).

Los sistemas UEFI pueden establecer el modo de arranque. En general, se pueden establecer dos modos:

1. *Heredado o Legacy BIOS*: establece la compatibilidad hacia atrás con discos con esquemas MBR. En este modo, no se hacen efectivas las ventajas del estándar UEFI. Durante la instalación del sistema operativo, la configuración de las particiones se hará con esquema MBR.

EXAMEN



2. **UEFI:** es el modo que recomienda la mayoría de sistemas operativos, dadas sus ventajas. Durante la instalación del sistema operativo en modo UEFI, se crean por defecto esquemas de particiones GPT.

**TEN EN CUENTA**

- ✓ Debe existir una consonancia entre el modo de arranque del equipo y el esquema de particionamiento del disco de arranque (el primero en el orden de arranque). Los discos con sistemas operativos instalados en esquemas de particionamiento GPT y MBR deben ser iniciados en modo UEFI y Heredado, respectivamente. De lo contrario, el sistema UEFI o BIOS no reconocerá el sistema operativo al encontrarse en un disco con sistema de particionamiento no reconocido.

**Actividad resuelta 2.1**

Averiguar el esquema de particionamiento de un volumen en Microsoft Windows.

**SOLUCIÓN**

Podemos conocer el esquema de partición de los discos en Microsoft Windows 10. Para ello, ejecutamos el 'Administrador de discos' (`diskmgmt.msc`). Aparecerán las unidades y seleccionamos con el botón secundario sobre la unidad en propiedades. Se abrirá una nueva ventana con las propiedades de la unidad y seleccionamos la pestaña 'Hardware'. Volvemos a seleccionar el disco correspondiente a la unidad y pulsamos en el botón 'Propiedades'. Aparecerá la información de dicho disco. En la pestaña 'Volúmenes' pulsamos en el botón 'Rellenar' y el apartado 'Estilo de partición' mostrará si es del tipo MBR o GPT.

Otra manera de conocer el estilo de partición de un disco en Windows es ejecutando el comando `msinfo32`. En el apartado 'Modo de BIOS' aparecerá 'Heredado' o 'UEFI'.

Cuando se realiza la instalación de Windows en un equipo con arranque UEFI, el esquema predeterminado de particiones es el siguiente:

- a) Partición del sistema (System): partición del sistema EFI o ESP. Se encuentra en el disco duro de arranque, siendo la primera que inicia.
- b) Partición reservada de Microsoft (MSR): partición reservada para la gestión de las particiones, que no puede almacenar datos de usuarios.
- c) Partición de Windows (Windows): partición que alojará el sistema operativo.
- d) Partición de recuperación (Recovery): partición de herramientas de recuperación.

**Figura 2.25**

Esquema de particiones por defecto en una instalación Windows con arranque UEFI.

Y cuando la instalación se realiza en un equipo con arranque 'Heredado', el esquema ideal que plantea Microsoft es el siguiente:

- a) Partición del sistema (System): ha de estar configurada como la partición activa del disco de arranque.
- b) Partición de Windows (Windows): partición que aloja el sistema operativo.
- c) Partición de recuperación (Recovery): almacena las herramientas del entorno de recuperación de Windows.

**Figura 2.26**

Esquema de particiones por defecto en una instalación Windows con arranque Heredado.

**TOMA NOTA**

Siempre es recomendable emplear un espacio de almacenamiento destinado a guardar datos por parte de los usuarios, diferente a las particiones definidas por defecto por Windows durante su instalación de manera predeterminada. Por tanto, es el administrador del sistema el encargado de crearla. Además, la partición de recuperación no es imprescindible o se puede emplear junto con la partición de Windows, dependiendo del modo de arranque seleccionado y de la planificación de las particiones que el administrador del sistema prevea.

### 2.8.2. Gestor de arranque de Windows

El gestor de arranque de Windows es *BOOTMGR (Windows Boot Manager)*. *BOOTMGR* hace uso de un almacén de datos de configuración de arranque (*BCD, Boot Configuration Data*). Y para realizar modificaciones en la configuración del *BCD*, se emplea el programa *bcdeedit.exe*. Por tanto, si se desean realizar cambios, debemos familiarizarnos con las opciones, en línea de comandos, de este programa. No obstante, para una configuración simple se puede emplear el programa *msconfig*, y también existen aplicaciones como *EasyBCD* (<https://neosmart.net/EasyBCD/>) que permiten una edición gráfica de *BCD* de manera más sencilla e intuitiva.

Windows Boot Manager se encarga de:

- Cargar las aplicaciones de arranque de Windows: el cargador del sistema operativo (OS Loader), el cargador de reanudación (Resume Loader) tras una hibernación o el test de memoria.
- Mostrar el menú de selección por el usuario, donde se muestran las opciones de arranque (pueden aparecer varios sistemas operativos reconocidos por *BOOTMGR*).
- Localizar el cargador del sistema operativo (OS Loader), seleccionado en la opción anterior.
- Cargar el cargador del sistema operativo y transferirle el control.

La localización de BOOTMGR y el resto de archivos que intervienen durante el arranque se localizan en rutas diferentes, según el esquema de partición empleado (GPT o MBR). Sin embargo, el proceso de arranque hasta cargar el sistema operativo es el siguiente: BOOTMGR carga el cargador del sistema operativo seleccionado por el usuario (WINLOAD), y este último carga, a su vez, el núcleo del sistema operativo (NTOSKRNL).

### 2.8.3. Gestor de arranque de Linux

Linux normalmente emplea el gestor de arranque *GRUB 2*, una evolución reescrita de su antecesor *GRUB (GNU Grand Unified Bootloader)* y que ahora se conoce como *GRUB Legacy*. GRUB 2 es muy potente y flexible, puede lanzar la mayoría de los sistemas operativos.

Los archivos y directorios que intervienen en el funcionamiento y gestión de GRUB 2 en Ubuntu son:

- a) El archivo de configuración principal de GRUB 2 normalmente se encuentra en el directorio `/boot/grub/` y es `grub.cfg`. Este es producto de varios scripts y no debe ser modificado directamente. Las modificaciones tienen efecto en este fichero cuando se hace uso de la orden `/usr/sbin/update-grub` o mediante la actualización del kernel.
- b) La configuración del menú gráfico durante el arranque se gestiona mediante el archivo `/etc/default/grub`. Se puede modificar el tiempo de espera del menú, la selección por defecto de este, establecimiento de password, etc., y se edita con privilegios de `root`.
- c) Además, en el directorio `/etc/grub.d/` se encuentra un conjunto de scripts ejecutables numerados. La lectura de estos scripts se realiza en el mismo orden de numeración. Aunque se pueden agregar scripts en el directorio `/etc/grub.d/`, por defecto, en Ubuntu se encuentran:
  - `00_header`: contiene y carga la información básica del GRUB desde `/etc/default/grub`.
  - `05_debian_theme`: establece la configuración de la imagen de fondo, el color del texto, etc.
  - `10_linux`: se encarga de localizar el kernel de Linux.
  - `20_memtest86+`: localiza y añade al menú el programa de testeo de memoria `/boot/memtest86+.bin`.
  - `30_os-prober`: busca otros sistemas operativos instalados en el disco y los añade al menú.
  - `40_custom`: archivo donde el usuario puede agregar entradas nuevas.

Cuando la BIOS pasa el testigo al gestor de arranque GRUB 2, este ejecuta el archivo de configuración `/boot/grub/grub.cfg` y se muestra el menú gráfico de selección de opciones de arranque. Por defecto, se mantendrá unos segundos hasta que el usuario seleccione una opción o, en caso contrario, arrancará la opción por defecto.

Si se ha seleccionado un sistema operativo de tipo *Linux*, se lanzará su kernel `/boot/vmlinuz-<versión>`. En el caso de seleccionar otro sistema operativo “no soportado” por GRUB, se transferirá el control al gestor de arranque propio del otro sistema operativo para que inicie la carga del núcleo correspondiente. A este proceso se le conoce como *arranque en cadena*.

## 2.9. Actualización del sistema operativo

Como cualquier otro software, el sistema operativo ha de actualizarse y, además, esto supone una premisa básica para la correcta funcionalidad y seguridad del sistema. Los sistemas operativos actuales presentan avisos sobre nuevas versiones o actualizaciones disponibles cuando estas se liberan.

### 2.9.1. Administración de actualizaciones en Windows

Por defecto, Windows descarga e instala las actualizaciones para asegurarse de que el dispositivo se encuentra protegido y de que funcione eficazmente. La mayoría de estas actualizaciones requieren que se reinicie el sistema. Microsoft Windows las clasifica en:

- ✓ Críticas e importantes: asociadas a la seguridad, privacidad y estabilidad del sistema.
- ✓ Recomendadas: aunque no son fundamentales, permiten mejorar el rendimiento o ciertas prestaciones.
- ✓ Opcionales: relacionadas con drivers de dispositivos y software de Microsoft.

**Además, Microsoft libera de forma gratuita actualizaciones en bloque llamadas *Service Pack*. Se trata de paquetes de actualizaciones anteriores, así como correcciones y revisiones del sistema.**

La configuración de las actualizaciones se realiza a través de Windows Update en ‘Actualización y seguridad’, dentro de ‘Configuración’. Windows Update avisará si existen nuevas actualizaciones importantes a través de la barra de tareas. Además, al acceder a Windows Update, se indicará el tipo concreto de actualización.

En ‘Opciones avanzadas’, dentro de Windows Update, podemos seleccionar diversas opciones de actualización:

- a) Ofrecer actualizaciones para otros productos de Microsoft.
- b) Descargar automáticamente actualizaciones.
- c) Ver más opciones sobre el reinicio cuando el sistema vaya a reiniciar.
- d) Pausar las actualizaciones. Esta opción detendrá las actualizaciones durante un periodo de tiempo; pasado el cual, se deben instalar las actualizaciones para poder volver a pausar las actualizaciones.

Además, se puede ver el historial de actualizaciones instaladas desde la opción en ‘Ver historial de actualizaciones’, dentro de Windows Update. Desde esta ventana también se pueden desinstalar actualizaciones.

Cuando Windows descarga las actualizaciones, nos avisa de que debe reiniciarse para poder instalarlas. Incluye las opciones de actualización al apagar o reiniciar el equipo. Al arrancar el



**Figura 2.27**

Ejemplo de aviso de actualizaciones en la barra de tareas y en la propia ventana de Windows Update.

sistema, muestra el porcentaje de actualización antes de iniciar la selección de usuarios y cargar el escritorio.



### Actividad propuesta 2.11

Accede a Windows Update y lista todas las actualizaciones del sistema. Desinstala la última de ellas e intenta actualizar el sistema e instalarla de nuevo.

#### 2.9.2. Administración de actualizaciones en Ubuntu Desktop

A diferencia de Microsoft Windows, Ubuntu Desktop actualiza tanto el software del sistema (incluidos los drivers) como las aplicaciones instaladas. La administración de las actualizaciones del sistema se hace a través de ‘Software y actualizaciones’, en la pestaña ‘Actualizaciones’.

En esta ventana se distinguen tres tipos de actualizaciones:

1. De seguridad: Ubuntu recomienda tener siempre activas.
2. Recomendadas: los desarrolladores de paquetes aconsejan tener activas.
3. Sin asistencia técnica: actualizaciones de paquetes no soportados por Ubuntu.

Además, se puede seleccionar la frecuencia de comprobación de actualizaciones y las acciones para realizar cuando se detectan actualizaciones de seguridad o de otro tipo. También permite que habilitemos notificaciones de versiones nuevas de Ubuntu. Cuando existe una nueva actualización, esta se notifica en el escritorio de Ubuntu e incluso en la pantalla de selección de usuario.



**Figura 2.28**  
Software y actualizaciones  
en Ubuntu.



**Figura 2.29**  
Ejemplo de aviso de actualización  
en la pantalla de selección.

#### 2.10. Identificación, instalación y desinstalación de aplicaciones

Windows centraliza la gestión de las aplicaciones mediante ‘Aplicaciones y características’, a través de ‘Aplicaciones’, desde ‘Configuración’. Por su parte, Ubuntu Desktop emplea el ‘Software de Ubuntu’ como la plataforma sobre la que se sustenta la administración de los programas.

### 2.10.1. Aplicaciones y características de Windows

En ‘Aplicaciones y características’ aparece un listado con las aplicaciones instaladas. Entre ellas, existen aplicaciones nativas de Microsoft Windows que no pueden ser desinstaladas.

Además, Microsoft Windows dispone de un conjunto de opcionales llamadas ‘Características’, algunas de las cuales se encuentran activadas y otras no. Podemos acceder a las ‘Características de Windows’ desde ‘Aplicaciones y características’, en el apartado ‘Opciones de configuración relacionadas’, pulsando sobre ‘Programas y características’. Al igual que antes, aparece el listado con las aplicaciones instaladas y en el menú de la izquierda se muestra la opción ‘Activar o desactivar las características de Windows’. Si pulsamos en él, aparecerá el listado de características activadas o no.

Microsoft Windows permite ejecutar cualquier programa si disponemos de los privilegios oportunos como usuarios para ello. Normalmente, la instalación se resume en ejecutar (doble clic por defecto) el ejecutable del programa para instalar, que lanza un asistente que guiará durante el proceso de instalación y configuración.

El proceso de desinstalación resulta igual de simple: pulsamos sobre una aplicación del listado de aplicaciones de ‘Aplicaciones y características’, y nuevamente pulsamos sobre el botón ‘Desinstalar’. Si es una aplicación relacionada con una característica, debemos desinstalarla.

### 2.10.2. Software de Ubuntu

Se puede acceder a él a través del Dock (o barra lateral que por defecto se instala en la izquierda) mediante su ícono de enlace al ‘Software de Ubuntu’ o por el buscador de aplicaciones.

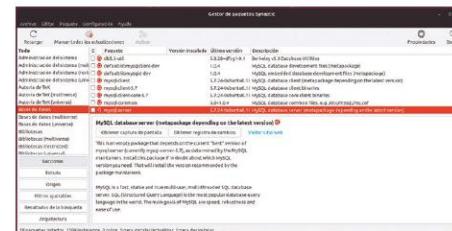
En el ‘Software de Ubuntu’ se puede buscar software por categorías o realizar una búsqueda. Además, en la parte superior podemos encontrar el software instalado y las actualizaciones.

Al igual que las actualizaciones del sistema, Ubuntu Desktop toma la configuración de las aplicaciones a través de ‘Software y actualizaciones’, en la pestaña de ‘Software de Ubuntu’ y ‘Otro software’.

En ‘Software de Ubuntu’ encontramos los tipos de repositorios (almacenes de paquetes software y aplicaciones): *main*, *universo*, *restricted* y *multiverse*, que siempre se recomienda tener activos para la detección e instalación de nuevo software, así como sus actualizaciones. Para los desarrolladores, también se recomienda activar ‘Código fuente’.



**Figura 2.30**  
Centro de Software de Ubuntu de usuario.



**Figura 2.31**  
Gestor de paquetes  
Synaptic.

Además del ‘Software de Ubuntu’, Canonical (empresa que produce, da servicio y comercializa Ubuntu) dispone de socios (empresas como Skype, Adobe o Sun). A través de la pestaña ‘Otro software, en Software y actualizaciones’, podemos habilitar la incorporación y actualización de software relacionado con dichas empresas.

**TOMA NOTA**

Una aplicación con entorno gráfico ampliamente utilizada para la gestión de paquetes es *Synaptic*, la cual podemos descargar desde el propio centro de ‘Software de Ubuntu’. Resulta una herramienta muy potente y versátil.

La interfaz textual también ofrece muchas opciones para la gestión de paquetes en Linux. Una de las herramientas más utilizadas en línea de texto es *apt* (*Advanced Packaging Tool*), de hecho, multitud de gestores de software se basan en él (como Synaptic). Su uso en el terminal se realiza con privilegios de administrador. Las tareas más importantes son:

- *sudo apt update*: actualización de paquetes disponibles: actualiza el índice de repositorios de nuestro sistema.
- *sudo apt install nombre\_paquete*: instalación de paquetes: se pueden dejar espacios entre paquetes si se desean instalar varios.
- *sudo apt remove nombre\_paquete*: desinstalar paquetes.
- *sudo apt upgrade*: actualización del sistema.

**Actividad propuesta 2.12**

Descarga e instala Synaptic Package Manager. Comenta las diferencias entre el centro de Software de Ubuntu y Synaptic. Realiza un breve estudio de la estructura de Synaptic y la forma de instalar, actualizar y desinstalar paquetes.

**Resumen**

- Los sistemas operativos actuales son una evolución de sistemas informáticos originados en los años 40, que establecieron los fundamentos de la computación moderna. Así, conceptos como *multiusuario*, *tiempo compartido* o *multitarea* se emplean hoy en día en sistemas operativos como Microsoft Windows o Ubuntu Desktop.
- La principal función de un sistema operativo es la de gestionar:
  - El procesador.
  - La memoria.
  - Las entradas y salidas.
  - El almacenamiento secundario.
  - La seguridad.

- Los errores.
  - Las interfaces de usuario.
- Además, estos han de ser: adaptables (con capacidad de evolucionar para adecuarse a un nuevo software y hardware), eficientes ante la competencia por los recursos y de fácil uso a través de interfaces textuales o gráficas.
  - La catalogación de los sistemas operativos permite estudiar las características y peculiaridades de estos, como, por ejemplo, distinguiendo entre sistemas operativos atendiendo al procesamiento: en *tiempo real* frente a *tiempo compartido*.
  - Además, las arquitecturas nos adentran en el diseño de los propios sistemas operativos y cómo afecta a su desarrollo, eficiencia, estabilidad o rendimiento (distinguiendo *anillo*, *monolítico*, *microkernel* o *híbrido*).
  - El proceso de instalación de los sistemas operativos, posterior a una planificación previa y teniendo en cuenta unos requisitos mínimos y recomendados, consiste en una práctica obligatoria para entender la función de los mismos. Los procedimientos de instalación los hemos desarrollado mediante Microsoft Windows y Ubuntu Desktop sobre Oracle VM VirtualBox.
  - El estudio mediante máquinas virtuales nos ha permitido profundizar en los gestores de arranque y su configuración, necesarios para la carga de los sistemas operativos. Por último, hemos estudiado las actualizaciones de los sistemas operativos y la gestión de aplicaciones.



## Ejercicios propuestos

1. Busca en Internet ejemplos de sistemas operativos atendiendo a:
  - El número de procesos que se pueden ejecutar concurrentemente.
  - El número de usuarios atendidos simultáneamente.
  - El tipo de procesamiento.
  - El sistema de interfaz empleado.
  - La forma de ofrecer servicios.
2. Busca en Internet ejemplos de sistemas operativos para cada tipo de arquitectura.
3. Crea una unidad flash USB arrancable Windows 10 mediante la utilidad Rufus (<https://rufus.ie/>) sobre Microsoft Windows. Para ello, descarga previamente la imagen .iso desde la página oficial de Microsoft o directamente a través de la propia aplicación Rufus. Lee los tutoriales de ayuda, si fuese necesario, prestando atención al esquema de partición y al sistema destino. Comprueba en un equipo físico que se inicia el proceso de instalación. ¿Qué ocurre si no se ajusta con el sistema destino? Compruébalo modificándolo a través del BIOS Setup Utility.
4. Crea una máquina virtual en Oracle VM VirtualBox para realizar una instalación de Microsoft Windows. Ten en cuenta los requisitos mínimos y recomendados. Documenta y detalla todos los pasos del proceso de instalación.

5. Crea una máquina virtual en Oracle VM VirtualBox para realizar una instalación de la última versión LTS de Ubuntu Desktop. Ten en cuenta los requisitos mínimos y recomendados. Crea manualmente y durante el proceso de instalación las particiones que sean necesarias. Documenta y detalla todos los pasos del proceso de instalación.
6. A través de la aplicación NTLite, crea una instalación desatendida donde se añada la característica *Internet Information Services*. Comprueba la correcta generación del *.iso* en una máquina virtual.
7. Explica el esquema de particiones predeterminado tras la instalación de Microsoft Windows del ejercicio 4. Para ello, ejecuta el 'Administrador de discos', detallando el contenido de cada partición.
8. Conociendo los archivos de configuración del gestor de arranque de GNU/Linux (GRUB 2) y a partir del ejercicio 5, modifica el gestor de arranque de manera que aparezca un menú gráfico que ofrezca entre la carga de Ubuntu Desktop y el programa de testeo de memoria *memtest86+.bin* con una imagen personalizada de fondo y 23 segundos de tiempo de espera. Accede a tutoriales oficiales de GRUB 2 en <https://help.ubuntu.com/community/Grub2> como apoyo.
9. En Microsoft Windows 10:
  - a) Descarga un navegador web (que no tengas instalado desde su página oficial) e instálalo.
  - b) Localiza el nuevo navegador web en 'Aplicaciones y características' de Windows. Ahora desinstálalo.
  - c) Desactiva Internet Explorer (característica de Windows) y vuelve a activarlo.
  - d) Lista las actualizaciones de Microsoft Windows. ¿Existe alguna pendiente?
10. A través de un terminal de texto en Ubuntu Desktop:
  - a) Actualiza los paquetes disponibles.
  - b) Instala el programa Synaptic.
  - c) A través de Synaptic, instala un navegador web que no tengas instalado en Ubuntu Desktop.
  - d) Desinstala el programa Synaptic.
  - e) Actualiza el sistema.

### ACTIVIDADES DE AUTOEVALUACIÓN

1. Los sistemas operativos que pueden atender a más de un usuario concurrentemente se denominan:
  - a) Multitarea.
  - b) Monotarea.
  - c) Multiusuario.

2. El gestor de arranque de la familia Microsoft Windows NT es:

- a) GRUB 2.
- b) NTOSKRNL.
- c) BOOTMGR.

3. Los sistemas UEFI pueden gestionar esquemas de particiones:

- a) Solo GPT.
- b) Solo MBR.
- c) GPT y MBR.

4. El estándar UEFI:

- a) Solo permite trabajar en modo de 32 bits.
- b) Limita en 2,2 TB la capacidad máxima gestionable de discos.
- c) Es más flexible que el estándar BIOS.

5. Ubuntu Desktop es un sistema operativo:

- a) Distribuido.
- b) De tiempo compartido.
- c) Monotarea.

6. Un equipo con modo de arranque UEFI:

- a) Creará particiones MBR por defecto.
- b) No reconocerá un disco de arranque MBR.
- c) No podrá crear una tabla de particiones MBR a través de gestores de disco.

7. GRUB 2:

- a) No puede enlazar con sistemas operativos de tipo Windows.
- b) Es configurable directamente y prioritariamente a través de la edición del fichero /boot/grub/grub.cfg.
- c) Sus modificaciones tienen efecto al ejecutar /usr/sbin/update-grub.

8. Las actualizaciones de Microsoft Windows son gestionadas por:

- a) Software y actualizaciones.
- b) Windows Update.
- c) Aplicaciones y características.

9. Por defecto, la administración de programas en Ubuntu es llevada a cabo por:

- a) Software de Ubuntu.
- b) Synaptic.
- c) Dash.

10. La siguiente ejecución actualizaría el índice de repositorios de Ubuntu:

- a) sudo apt install index.
- b) sudo apt upgrade.
- c) sudo apt update.

#### SOLUCIONES:

1.  a  b  c

2.  a  b  c

3.  a  b  c

4.  a  b  c

5.  a  b  c

6.  a  b  c

7.  a  b  c

8.  a  b  c

9.  a  b  c

10.  a  b  c

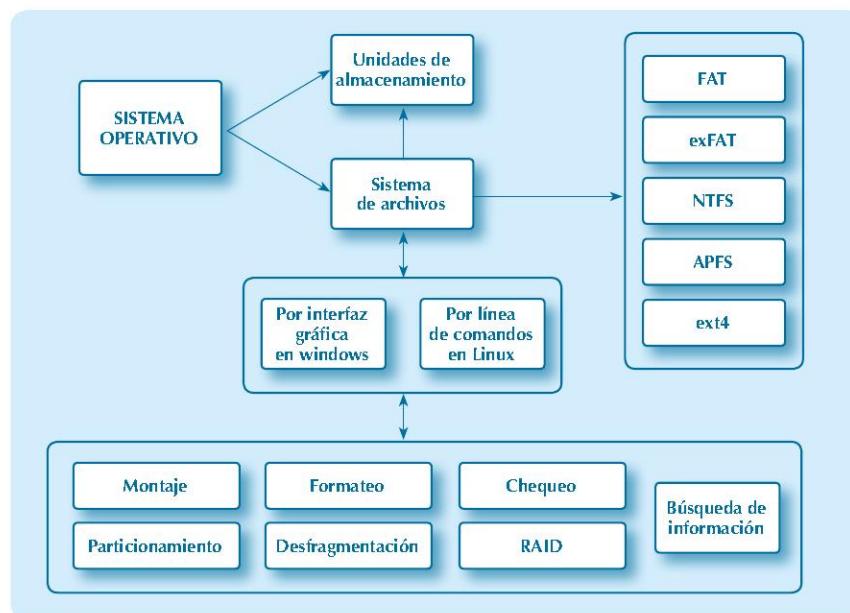
# 3

## Sistemas operativos. Gestión de archivos y almacenamiento

### Objetivos

- ✓ Conocer y comparar los sistemas de archivos más empleados.
- ✓ Reconocer la estructura y funciones de los directorios de los sistemas operativos más empleados.
- ✓ Crear e identificar diferentes tipos de particiones y unidades lógicas.
- ✓ Instalar y evaluar utilidades relacionadas con la gestión del almacenamiento e información.
- ✓ Operar con distintas herramientas, por comandos o entorno gráfico, para localizar archivos e información en el sistema de archivos.

### Mapa conceptual



### Glosario

**Archivo.** La unidad lógica mínima que contiene información.

**Desfragmentación.** Proceso de unión de bloques de datos de un mismo archivo, evitando la disgregación o esparcimiento de estos entre sí en el medio de almacenamiento.

**Directorio.** Contenedores lógicos de ficheros o directorios.

**Enlace simbólico.** Tipo de archivo en sistemas de archivos ext4, entre otros, que refieren a otros archivos o directorios del mismo u otro sistema de archivos.

**Estructura de directorios de un sistema operativo.** Conjunto de directorios que constituyen la organización y despliegan los archivos y subdirectorios del propio sistema operativo.

**Formateo.** Proceso por el que se instala un sistema de archivos en una partición.

**i-nodo.** Estructura de datos, en sistemas de archivos ext4, entre otros, que almacenan meta-information del archivo que representan.

**Journaling.** Registro del sistema de archivos que evita la inconsistencia de ficheros y facilita la recuperación de datos en los medios de almacenamiento.

**Partición.** División interna del dispositivo de almacenamiento que permite organizar la información y facilitar la gestión del almacenamiento.

**RAID.** Configuración de un grupo de discos independientes para aumentar la integridad, la capacidad de almacenamiento, la velocidad de transferencia o disminuir el riesgo a fallos.

### 3.1. Introducción

Al igual que la gestión de procesos y la gestión de usuarios, la gestión de los archivos es uno de los pilares fundamentales de cualquier sistema operativo.

Los sistemas operativos son capaces de gestionar la información contenida en los medios de almacenamiento gracias a los sistemas de archivos. Estos proveen la manera de almacenar la información, así como mecanismos que permitan realizar operaciones sobre ella.

Existen multitud de sistemas de archivos que confieren diferentes características al espacio de almacenamiento y repercuten en la seguridad de los datos, su rendimiento o su gestión. En este tema estudiaremos diferentes sistemas operativos, centrándonos en los más usados: *ext4* y *NTFS*.

Los sistemas operativos proveen herramientas para la gestión del almacenamiento, ya sea por línea de comandos o por interfaz gráfica, para realizar particiones de los medios de almacenamiento, formatear, montar y desmontar los sistemas de archivos, desfragmentar, chequear los sistemas de archivos, buscar información e incluso crear diferentes esquemas RAID.

Para trabajar la gestión de archivos y almacenamiento, a lo largo del tema trabajaremos con los sistemas operativos *Microsoft Windows* a través de su interfaz gráfica y con *Ubuntu* mediante su interfaz textual. Sobre este último recaerá la mayoría del peso de los aspectos que se van a estudiar, ya que consiste en una herramienta fundamental para usuarios expertos o administradores de sistemas operativos.

### 3.2. Sistemas de archivos

Los sistemas de archivos emplean el *archivo (fichero)* como la herramienta fundamental de abstracción lógica de la información. Se dice que un archivo es, por tanto, la unidad lógica mínima de almacenamiento que contiene información. Es decir, los archivos se emplean para evitar que el usuario conozca la estructura interna y las propiedades características de los medios de almacenamiento, facilitando la gestión y organización por su parte.

Otro elemento empleado por los sistemas de archivos son los *directorios (carpetas)*. Estos son ficheros que actúan de contenedores lógicos de ficheros o directorios. Independientemente del sistema de archivos donde se defina el directorio, este almacena información relativa a la localización física de la información y los atributos propios de cada archivo o directorio que contenga.

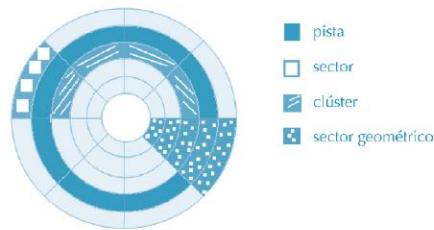
Los sistemas de archivos tienen como objetivo:

- Acceder a la información de los ficheros.
- Crear, eliminar y modificar ficheros.
- Acceder a los ficheros mediante diferentes protocolos de comunicación en red u otros ficheros.

- Facilitar el acceso multiusuario.
- Facilitar el acceso a multitud de medios de almacenamiento.
- Realizar copias de seguridad.
- Utilizar herramientas de recuperación de información.
- Priorizar la eficiencia y la seguridad de acceso a la información.
- Maximizar el rendimiento en las operaciones sobre los archivos.
- Permitir la monitorización y contabilidad sobre ficheros.
- Administrar el espacio de almacenamiento, gestionar la asignación del espacio libre y el espacio ocupado de los archivos.

Para administrar el espacio libre (susceptible de ser asignado) y el espacio ocupado, se han de definir espacios de asignación. Estas son las unidades físicas mínimas de almacenamiento gestionadas por los sistemas de archivos. Los sistemas de archivos definen el tamaño del espacio de asignación (también llamado *unidad de asignación* o *clúster*) durante la instalación del propio sistema de archivos (formateo). Este espacio determina el tamaño mínimo que ocupará un archivo en el medio de almacenamiento. A nivel físico (hardware), el dispositivo administra sectores, sin embargo, el sistema de archivos gestiona clústeres.

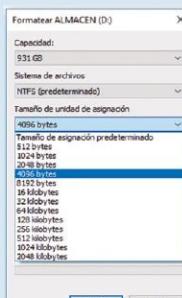
La planificación de este espacio es muy importante, siendo ideal un equilibrio entre el promedio del tamaño de los archivos que vaya a alojar el sistema de archivos (evitando así que se desperdicie espacio interno al clúster, también conocido como *fragmentación interna*) y el tamaño del volumen (facilitando la administración del sistema de archivos). Si el usuario no define el tamaño del clúster, el sistema operativo lo asignará automáticamente, atendiendo al tamaño del disco, el tipo de sistema de archivos y el esquema de particionamiento (*MBR* o *UEFI*).



**Figura 3.1**  
Esquema físico de un disco duro mecánico.

#### RECUERDA

- ✓ En Microsoft Windows, al formatear una unidad (pulsando el botón secundario del ratón sobre ella), se puede indicar el tamaño de unidad de asignación.



### 3.2.1. FAT (File Allocation Table)

Es un sistema de archivos creado para el sistema operativo MS-DOS. La administración del espacio del almacenamiento es sencilla, por lo que se convierte en un sistema de archivos muy extendido en la mayoría de los sistemas operativos. De tal manera, que se suele emplear en dispositivos utilizados para intercambiar datos en computadoras con varios sistemas operativos, videoconsolas, televisores, etc.

Las principales limitaciones de FAT32 son:

- ✓ Imposibilidad de gestionar particiones superiores a 8 TB (32 GB en Microsoft Windows) y archivos de más de 4 GB.
- ✓ Bajo rendimiento.
- ✓ Inseguro: no permite encriptación, sus atributos y permisos son limitados y no permite *journaling*.

#### TOMA NOTA



*Journaling* es un registro o diario del sistema de archivos. Los sistemas de archivos que hacen uso de este sistema se denominan transaccionales. Consiste en registrar una serie de acciones previas a la operación sobre el sistema de archivos que se vaya a realizar. De tal manera, que si se produce un error durante alguna operación (copiado, borrado, creación, etc.), el sistema puede deshacer los cambios y volver a la situación original. Así se evitan muchas situaciones de inconsistencia de ficheros, facilitando la recuperación de datos y chequeos de medios de almacenamiento.

### 3.2.2. exFAT

Resultado de una evolución del sistema de archivos FAT32, que elimina sus principales limitaciones. Se pueden tratar archivos de hasta 16 EB y mantiene la ligereza frente a sistemas de archivos más avanzados, como NTFS y APFS. Aunque sigue resultando inseguro, es ideal para medios de almacenamiento FLASH portables con gran capacidad y compatibles entre sistemas operativos.

### 3.2.3. NTFS

Se considera el sistema de archivos estándar de Microsoft Windows. Sus mejoras son considerables con respecto a FAT32, primando la seguridad y la confiabilidad. Sus principales ventajas son:

todas

- Emplea *journaling*. Favoreciendo una pronta recuperación ante errores inesperados.
- Permite cifrado y compresión.
- Reduce significativamente la fragmentación y aumenta la velocidad de búsqueda de archivos con respecto a FAT32.
- Puede llegar a gestionar volúmenes de hasta 16 EB y archivos de hasta 16 TB.
- Emplea Unicode para el nombre de archivos, con hasta 255 caracteres.

### 3.2.4. APFS

Sistema de archivos empleado por Apple Inc. para sus medios de almacenamiento, que supone una versión mejorada de su predecesora HFS+. Sus características son similares a NTFS y ext4, por lo que permite administrar archivos y volúmenes de hasta 8 EB. Permite encriptación y está optimizado para almacenamiento Flash.

 **Recurso digital 3.1**

Esquemas de partición con sistemas de archivos FAT, NTFS y ext4.

### 3.2.5. ext4 (Fourth extended file system)

Sistema de archivos predeterminado para sistemas operativos de tipo Linux en su cuarta versión. Incluye journaling, maneja archivos de hasta 16 TB y volúmenes de hasta 1 EB. Supera a sus antecesores ext2 y ext3, ya que:

- ✓ Mejora el rendimiento.
- ✓ Reduce la fragmentación.
- ✓ Permite trabajar con ficheros de mayor tamaño gracias al uso de *extents*.

A diferencia de NTFS, ext4 no emplea extensiones como parte del nombre de los archivos. En Microsoft Windows, un nombre de archivo se divide en *<nombre>.<extensión>*, donde *extensión* es un conjunto de caracteres (normalmente tres o cuatro) que se asocian con programas para que el sistema operativo reconozca la manera de ejecutar el archivo.

Ejemplos de extensiones muy usadas son:

- *.c*: fichero fuente en lenguaje de programación C.
- *.txt*: fichero de texto ASCII.
- *.tar*: fichero resultado de utilizar el comando tar.
- *.html*: fichero de lenguaje de marcas de hipertexto.

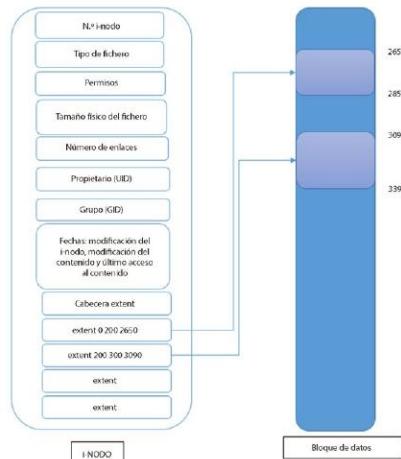
No obstante, muchos nombres de archivos en Linux incorporan sufijos separados por un punto en el nombre del fichero por convención, pero no como requisito establecido por el sistema de archivos.

Una partición con sistema de archivos *ext4* se divide en *grupos de bloques*. Cada grupo de bloques se divide, a su vez, en las siguientes partes:

- a) *Superbloque*: contiene la información más relevante del grupo de bloques.
- b) *Descriptores de grupos*: almacena la información más importante del resto de bloques.
- c) *Bitmap de bloques de datos*: contiene un mapa de bits donde se representa cada clúster, así como su estado (libre u ocupado).

- d) *Bitmap de i-nodos*: mapa de bits representando a cada *i-nodo*, que además indica su estado (libre u ocupado).
- e) *Tabla de i-nodos*: tabla que contiene una entrada por cada *i-nodo*. El *i-nodo* almacena la información propia de cada archivo.
- f) *Bloques de datos*: clústers con información. Cada bloque de datos está asociado con un archivo.

La estructura fundamental es el *i-nodo* o *nodo índice*. Este almacena toda la meta-information asociada al archivo que representa: tipo de archivo, propietario, tamaño, fechas, número de bloques de datos, localización de los bloques de datos, etc.



**Figura 3.3**  
Estructura de i-nodo en ext4.

TEN EN CUENTA

- ✓ Es importante la compatibilidad entre sistemas de archivos y sistemas operativos. Los sistemas de archivos ext4 y APFS no son reconocidos por Microsoft Windows, sin embargo, se puede emplear NTFS en la mayoría de distribuciones Linux. Por otro lado, exFAT se considera ideal para medios extraíbles de tipo FLASH por su compatibilidad entre la mayoría de sistemas operativos.

En ext4, el i-nodo emplea extents, los cuales intervienen con ficheros grandes. Estos emplean un conjunto de clústers contiguos (en lugar de estar separados) y se gestionan de manera simple, almacenando: el inicio del bloque dentro del archivo, el número de bloques almacenados y el inicio del número de bloque físico en disco.

Un i-nodo está compuesto por:

- ✓ Un identificador único de i-nodo o número de i-nodo. Único en el sistema de archivos.
- ✓ Tipo de fichero: regular, enlace simbólico, directorio o dispositivo.
- ✓ Permisos de lectura, escritura y ejecución para el propietario, grupo y otros usuarios.
- ✓ Tamaño del fichero (en bytes).
- ✓ Número de enlaces (duros). Hace referencia al número de veces que el i-nodo es referenciado en el árbol de directorios. Cuando se elimina un fichero de un directorio, decrementa en uno este número y se elimina la entrada correspondiente en dicho directorio. Así, ha de llegar a cero este número para que se elimine el i-nodo.
- ✓ Identificador del propietario del archivo (UID). Número identificativo único que representa a un usuario.

- ✓ **Identificador del grupo (GID)**. Número identificativo único que representa a un grupo de usuarios.
- ✓ **Fechas** de última modificación de la meta-information del i-nodo (*ctime*), última modificación de su contenido (*mtime*) y último acceso a su contenido (*atime*).
- ✓ **Cabecera extent**. Contiene información asociada a los extents que lo siguen: número válido de entradas que siguen a la cabecera, máximo número de entradas que siguen a la cabecera y la profundidad del extent actual (0 apunta a bloques de datos y en otro caso hasta 4 apuntará a nodos extents intermedios que actúan como indirecciones). Si el archivo está más fragmentado que los cuatro extents del i-nodo, estos apuntarán a nodos índices del árbol de extent hasta llegar a hojas extent.
- ✓ Cuatro nodos hoja extent que contienen el inicio del bloque dentro del archivo, el número de bloques almacenados y el inicio del número de bloque físico en disco.

### 3.3. Estructura de directorios en Linux y Microsoft Windows

Cuando los sistemas operativos GNU/Linux y Microsoft Windows son instalados en medios de almacenamiento, crean un conjunto de directorios donde se despliegan los archivos y subdirectorios del propio sistema operativo en una estructura en forma de árbol invertido del sistema de archivos. Esto es, un directorio del que cuelgan un conjunto de subdirectorios y, dentro de estos últimos, se alojan otros subdirectorios, y así sucesivamente.

#### RECUERDA

- ✓ En ambos, el directorio principal del que cuelgan el resto de directorios es el **directorio raíz**, simbolizado en Microsoft Windows por "\\" y en Linux por "/".

De esta manera, para hacer referencia a la localización de un directorio dentro de la estructura arbórea, se emplea el término *ruta, camino o path*. Así, se localiza fácilmente un archivo o directorio. Por ejemplo:

- En Linux podemos hacer referencia a la ruta de un directorio como */home/usuario 2/* donde "/" es el directorio raíz, "home" es un subdirectorio del directorio raíz y "usuario 2" es un subdirectorio del directorio *home*.
- En Microsoft Windows se puede indicar un directorio como *C:\Documents and Settings*. En Microsoft Windows el directorio raíz se antecede por una letra que simboliza una unidad o volumen. "C:\\" indica el directorio raíz de la unidad C.Y "*Documents and Settings\*" es un subdirectorio del directorio raíz.

El *directorio de trabajo actual* es el directorio donde se encuentra situado actualmente, al ir navegando por los directorios y subdirectorios de una unidad, dentro del árbol de directorios. El directorio de trabajo actual se simboliza por "...". En Linux podemos mostrar la ruta de trabajo actual mediante el comando *pwd* (print working directory).

Por *directorio padre* de un directorio se conoce al directorio que se encuentra por encima de él en la estructura jerárquica del árbol de directorios. Se simboliza por "...". De la misma manera,

un *directorio hijo* de un directorio es el directorio que se encuentra por debajo del primero en la estructura jerárquica.

La ruta de un archivo o directorio se puede indicar de dos maneras:

1. Ruta absoluta: ruta completa indicada desde el directorio raíz.
2. Ruta relativa: ruta indicada desde el directorio de trabajo actual. Se suele hacer uso de “..” y “.” para desplazarse.

El comando *cd* (change directory) permite cambiar de directorio. Su sintaxis es la siguiente: *cd [directorio]*, si no se indica directorio, cambiará al directorio home del usuario y equivaldría a ejecutar *cd ~*.



### Actividad resuelta 3.1

Nos encontramos en */home/luis* como directorio actual y queremos acceder al directorio raíz. ¿Cómo se puede indicar el cambio de directorio mediante ruta absoluta y relativa?

#### SOLUCIÓN

Ruta Absoluta: *cd /*  
Ruta relativa: *cd ../../..*

#### 3.3.1. Estructura de directorios en GNU/Linux

La mayoría de los sistemas operativos GNU/Linux siguen el estándar *FHS (Filesystem Hierarchy Standard)* que define el contenido y las funciones de los directorios en la estructura jerárquica del árbol de directorios. Los directorios más importantes son:

- ✓ */*: Directorio raíz o *root*. Todos los directorios y subdirectorios parten de este directorio raíz.
- ✓ */bin*: contiene los archivos binarios (ejecutables) a nivel de usuario.
- ✓ */boot*: almacena los archivos ejecutables y de configuración necesarios para el arranque del sistema.
- ✓ */dev*: se encuentran los componentes del sistema y todos los dispositivos de almacenamiento representados por archivos (memorias FLASH, particiones, DVD, etc.). A través de este directorio podemos acceder a la información propia del medio de almacenamiento.
- ✓ */etc*: almacena los archivos de configuración globales del sistema y que afectan a todos los usuarios.
- ✓ */home*: directorio que aloja los directorios de los diferentes usuarios del sistema, a excepción del usuario root (el cual emplea */root*).
- ✓ */lib*: contiene archivos muy importantes para el sistema operativo, como librerías y módulos del kernel.
- ✓ */media*: directorio que se emplea para montar dispositivos como discos duros o medios removibles como CD o DVD.

- ✓ /mnt: también utilizado para albergar puntos de montaje pero, en este caso, temporales, como, por ejemplo, un sistema de archivo en red o carpetas compartidas en máquinas virtuales.
- ✓ /proc: directorio empleado por el sistema para guardar información relativa a los procesos y al kernel del sistema. No contiene archivos físicos, sino que se generan sobre la marcha archivos virtuales.
- ✓ /sys: al igual que /proc, almacenan archivos virtuales relativos al kernel del sistema, así como información de drivers y dispositivos.
- ✓ /sbin: almacena ejecutables que se suelen emplear para tareas administrativas por el superusuario.
- ✓ /tmp: las aplicaciones emplean esta carpeta para almacenar archivos temporales.
- ✓ /usr: almacena archivos de solo lectura de la mayoría de las aplicaciones y utilidades instaladas en el sistema.
- ✓ /opt: contiene el resto de aplicaciones no almacenadas en /usr. Normalmente, son aquellas que no son parte de los paquetes instalados con la distribución Linux objeto de uso.
- ✓ /srv: directorio encargado de alojar datos, scripts y carpetas para servidores instalados en nuestro sistema (servidores web, ftp, repositorios, etc.).
- ✓ /var: directorio considerado como registro del sistema. En él se incluye información del sistema, logs, información de caché, etc.

### Actividad propuesta 3.1



Busca imágenes en Internet con la estructura de directorios de los sistemas operativos Microsoft Windows y Ubuntu (en sus versiones más actuales). Compáralos. El 'Escritorio', en ambos, es una carpeta que contiene ficheros o directorios. Para un usuario cualquiera, ¿cuál es la ruta del 'Escritorio' en ambos sistemas operativos?

### 3.3.2. Estructura de directorios en Microsoft Windows

El directorio raíz de una partición con Microsoft Windows instalado dispone del siguiente árbol de directorios:

- \Archivos de programa (Program Files): en sistemas de 32 bits, se encuentran todos los programas instalados. Sin embargo, en sistemas de 64 bits se almacenarán las aplicaciones de 64 bits.
- \Archivos de programa (x86): esta carpeta se encuentra en sistemas de 64 bits y contiene las aplicaciones instaladas de 32 bits.
- \PerfLogs: por defecto, está vacía, pero puede contener registros de rendimiento del sistema.
- \ProgramData: carpeta oculta que contiene datos de programas genéricos para todos los usuarios del sistema.
- \Usuarios (Users): carpeta que contiene subcarpetas por cada usuario del sistema, así como la carpeta \Acceso público (\Public) y \Default (carpeta oculta):
  - \Acceso público: carpeta compartida por todos los usuarios del sistema donde se definen aspectos comunes a ellos. Por defecto, está compartida en red.

- \Default: contiene el perfil base sobre el que se crean nuevos perfiles en el sistema. Así, al crear un nuevo usuario, su carpeta situada en C:\Usuarios, contendrá el perfil y la estructura definida en \Default.
- \[nombreUsuario]: contiene un conjunto de carpetas que definen el perfil del usuario (conjunto de valores de configuración del entorno: escritorio, aplicaciones, impresoras, conexiones de red, etc.), así como la carpeta oculta AppData. Esta última carpeta contiene los datos de las aplicaciones asociadas al usuario en sí (a diferencia de \ProgramData). En ella se encuentran tres subcarpetas: Roaming, que aloja perfiles de configuración de aplicaciones que puedan sincronizarse entre equipos; Local y LocalLow, que almacenan el resto de archivos empleados por las aplicaciones.

• Windows: contiene el grueso de la instalación del sistema operativo. En esta carpeta destacan las subcarpetas:

- \System32: contiene archivos DLL de 32 bits o 64 bits, dependiendo de si la versión de Microsoft Windows es de 32 bits o 64 bits respectivamente.
- \SysWOW64: solo en las versiones de 64 bits para almacenar archivos DLL de 32 bits.
- \WinSxS: conocido como el almácen de componentes de Microsoft Windows, que contiene archivos utilizados para la instalación, las actualizaciones del sistema, los Service Packs o las características de Microsoft Microsoft Windows.



#### SABÍAS QUE...

Las bibliotecas de vínculos dinámicos (DLL) son archivos que contienen código ejecutable y datos. Microsoft Windows los emplea frecuentemente, ya que aumenta la modularidad en las aplicaciones, ahorra recursos del sistema y simplifica la instalación y la ejecución de las aplicaciones.



#### Actividades propuestas

- 3.2.** Navega por la estructura de directorios de Ubuntu y lista las carpetas aquí estudiadas. Para mayor detalle, puedes hacer uso del comando `man hier` en un terminal de Linux, el cual especificará la utilidad de cada carpeta.
- 3.3.** Navega por la estructura de directorios de Microsoft Windows y lista las carpetas aquí estudiadas, observando su contenido.

### 3.4. Gestión de archivos por línea de comandos en Linux

En entornos domésticos o de oficina se emplea la interfaz gráfica para el manejo y gestión de archivos. No obstante, los administradores de sistemas suelen hacer uso de la interfaz por línea de comandos, ya que su versatilidad y potencia de uso la convierte en una herramienta ideal.

Los comandos de Linux siguen una sintaxis:

```
comando [opciones] [argumentos]
```

Cada comando varía el tipo de opciones y argumentos que pueda utilizar, e incluso puede no utilizarse ninguno en caso de ser opcionales. Las opciones pueden ser cortas (una sola letra) o largas (una palabra), antecedidas por uno o dos guiones. Se pueden emplear ambos tipos de opciones con un mismo comando, aunque las opciones cortas pueden unirse. Tanto los comandos como las opciones y los argumentos que se usen son sensibles a mayúsculas y minúsculas.

Uno de los comandos que más se emplea es *ls* (list).

```
ls [opciones] [ficheros]
```

Permite listar el contenido de un directorio e información de archivos. Sus opciones más utilizadas son las siguientes:

- ✓ l: muestra en formato largo.
- ✓ t: ordena por fecha de modificación.
- ✓ r: invierte el orden de salida.
- ✓ R: lista recursivamente el contenido de cada directorio.
- ✓ i: muestra el número de i-node.
- ✓ a: muestra los archivos ocultos. En Linux los archivos ocultos son aquellos que empiezan con “.”. Si no indicamos esta opción, el comando *ls* no listará los archivos ocultos.
- ✓ h: muestra el tamaño de cada fichero en K, M, G, etc.
- ✓ size: muestra el tamaño de cada fichero en bloques.
- ✓ S: lista los archivos ordenados por tamaño.

Por defecto y sin argumentos, lista el directorio actual ordenado alfabéticamente. Si se indica más de uno, se listará el contenido de cada uno de ellos.

Cuando se emplea la opción “-l”, aparecen clasificadas en columnas la información propia de cada fichero listado:

1. La primera columna es la lista de control de acceso o *máscara de permisos*. A su vez, se divide en dos partes:
  - a) El primer carácter puede ser: “d” indicando un directorio, “l” un enlace simbólico, “-“ un archivo regular, “c” un dispositivo de tipo carácter, o “b” un dispositivo de tipo bloque.
  - b) El resto de caracteres son los permisos asociados al usuario, al grupo y a otros, tomados en grupos de tres.
2. La segunda columna establece el número de enlaces duros asociados al archivo.
- 3 y 4. La tercera y cuarta columna indican el propietario y el grupo, respectivamente.
5. La quinta columna hace mención al tamaño que ocupa en bytes.
6. La sexta columna se refiere a la fecha de la última modificación de su contenido (*mtime*) o de su creación (si no se ha modificado).

7. La séptima y última columna se refiere al propio nombre del fichero. El nombre del fichero no reside en el i-nodo, sino en el directorio.

**TOMA NOTA**

En Linux, los nombres de los ficheros pueden tener una extensión entre 1 y 255 caracteres, pero, además, debemos conocer que:

- No se puede emplear el carácter "/".
- No es recomendable emplear los caracteres: =, ^, ~, ', ", `\*, -, ?, [, ], (,), !, &, ~, <, >. Esto se debe a que tienen un significado especial en el terminal. No obstante, podemos emplearlos, pero entrecambiando el nombre del fichero.
- Un archivo puede contener espacios. Para indicar al terminal que es un espacio en lugar del siguiente argumento, se entrecilla el nombre completo del fichero. También se puede anteceder cada espacio (se dice que se "escapa") con una barra invertida "\\" o se usan apóstrofes simples.
- Los nombres no tienen por qué usar extensiones. No obstante, es recomendable para muchos de ellos emplearlas para identificar fácilmente el tipo de información que contienen.
- Si queremos ocultar un archivo, se antecede con un punto "..".

Cualquier archivo o directorio se localiza e identifica dentro del árbol de directorios gracias a su ruta. Por ello, no pueden existir dos archivos con el mismo nombre en un mismo directorio.

Además, todos los directorios disponen, al menos, de dos entradas ocultas: " ." el directorio actual y " .." el directorio padre. Estas se definen automáticamente cuando se crea un directorio. El directorio actual hace referencia a él mismo y se emplea muy a menudo cuando se trabaja con rutas relativas, y también lo hacen por defecto multitud de comandos. También el directorio padre se utiliza necesariamente para poder moverse por el árbol de directorios.

**Actividades propuestas**

**3.4.** Ejecuta y explica las siguientes instrucciones: `ls /home /usr, ls -l /home, ls -R /home, ls -lra`.

**3.5.** Ejecuta y explica las siguientes instrucciones: `cd ., cd .. y cd ./..`

### 3.4.1. Tipos de ficheros

Los tipos de ficheros en Linux son muy importantes. Cualquier elemento físico (discos, impresoras, etc.) o lógico (directorio, enlace, etc.) se representa en Linux mediante un archivo, lo que facilita y estandariza la gestión de todos ellos. Se distinguen cuatro tipos elementales de ficheros:

- a) *Regulares*. Son ficheros ordinarios que contienen información de diversa naturaleza. Dentro de ellos se incluyen los ficheros ejecutables como un tipo de fichero regular que tiene activo algún permiso de ejecución y contienen código ejecutable.

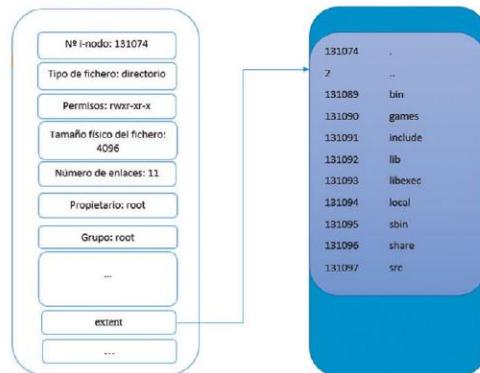
- b) *Directarios.* Almacenan en su bloque de datos el número de i-nodo y el nombre de los archivos que contiene. Cuando son listados con el comando *ls* muestra la información accediendo a la estructura interna de cada i-nodo.

### Ejemplo

En la siguiente imagen se observa cómo el directorio actual “.” (asociado al directorio /usr) tiene 131074 como número de i-nodo con 11 enlaces duros. Y el directorio padre del directorio actual (/usr) es el directorio raíz que tiene como número de i-nodo 2 con 24 enlaces duros. Su bloque de datos contiene los números de i-nodo y los nombres de los archivos que contiene.

```
luis@luis-VirtualBox:~$ ls -lai /usr
total 100
131074 drwxr-xr-x 11 root root 4096 oct 18 00:28 .
2 drwxr-xr-x 24 root root 4096 dic 23 11:15 ..
131089 drwxr-xr-x 2 root root 45056 dic 12 22:59 bin
131090 drwxr-xr-x 2 root root 4096 oct 18 00:27 games
131091 drwxr-xr-x 8 root root 4096 nov 10 11:28 include
131092 drwxr-xr-x 127 root root 4096 nov 15 10:09 lib
131093 drwxr-xr-x 2 root root 4096 oct 18 00:28 libexec
131094 drwxr-xr-x 10 root root 4096 oct 18 00:23 local
131095 drwxr-xr-x 2 root root 12288 dic 12 22:59 sbin
131096 drwxr-xr-x 258 root root 12288 nov 14 23:48 share
131097 drwxr-xr-x 8 root root 4096 dic 23 11:14 src
```

**Figura 3.4**  
Ejemplo de salida  
del comando *ls*  
con opciones –lai.



**Figura 3.5**  
Estructura de i-nodo  
de ejemplo.

- c) *Enlaces.* Existen dos tipos:

- *Enlaces duros.* Son asociaciones de nombres de ficheros a i-nodos. Es decir, se trata de reutilizar un i-nodo asignándole nombres distintos y localizándose en diferentes directorios o en el mismo. Los enlaces duros se establecen sobre ficheros regulares y no sobre directorios. Además, pueden hacer referencia solo dentro del mismo sistema de archivos. Para crear enlaces duros, se emplea el comando *ln*, y su sintaxis es la siguiente: *ln fichero fichero\_enlace*.

- *Enlaces simbólicos o enlaces blandos.* El directorio que contiene un enlace simbólico almacena un i-nodo (diferente al i-nodo con el archivo que enlaza) y un nombre de fichero. Para acceder al fichero con el que enlaza, el i-nodo del enlace simbólico almacena en un campo la ruta de acceso al archivo destino, si esta es inferior a 60 bytes y, si es superior, se almacenará en el bloque de información de un *extent*. De esta manera, pueden referenciar archivos de otros sistemas de archivos, particiones y equipos en red. Para crear enlaces simbólicos, se debe incluir la opción “*-s*” con *ln: ln -s fichero fichero\_enlace*.



### Actividad resuelta 3.2

*Crea un archivo llamado notas.txt sobre el que crearemos, posteriormente, un enlace duro llamado notas.txt.bck y un enlace simbólico notas.txt.s\_bck.*

#### SOLUCIÓN

Para crear un archivo nuevo sin contenido, podemos emplear el comando *touch*, siendo su sintaxis: *touch nuevo\_fichero*

El comando *touch* también permite actualizar la fecha de acceso y modificación del fichero, si existe, sin necesidad de modificarlo o acceder a su contenido.

Primero creamos el fichero *notas.txt*: *touch notas.txt*.

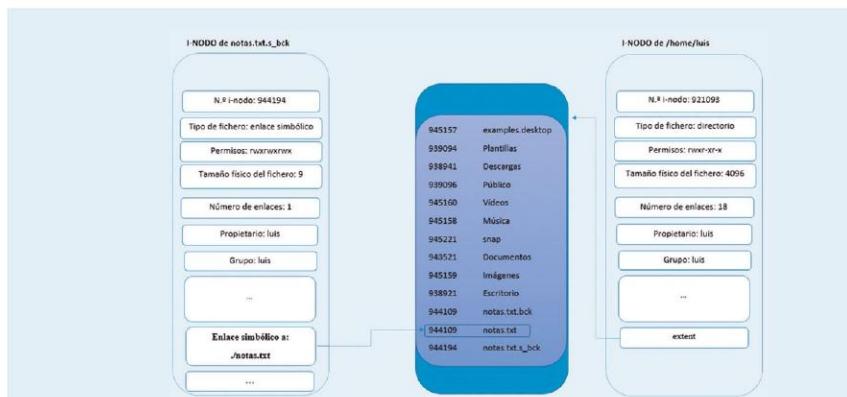
A continuación, creamos el enlace duro *notas.txt.bck* asociado al archivo *notas.txt*: *ln notas.txt notas.txt.bck*

Y, por último, creamos el enlace simbólico *notas.txt.s\_bck* y listamos el contenido del directorio actual en formato largo, mostrando los números de i-nodo, como aparece en la siguiente imagen:

```
luis@luis-VirtualBox:~$ ln -s notas.txt notas.txt.s_bck
luis@luis-VirtualBox:~$ ls -ltr
total 48
045157 -rw-r--r-- 1 luis luis 0 9:00 nov 10 11:27 examples.desktop
939041 drwxr-xr-x 2 luis luis 4096 nov 10 11:30 Plantillas
939041 drwxr-xr-x 2 luis luis 4096 nov 10 11:30 Documentos
939096 drwxr-xr-x 2 luis luis 4096 nov 10 11:36 Últimas
945160 drwxr-xr-x 2 luis luis 4096 nov 10 11:36 Videos
945150 drwxr-xr-x 2 luis luis 4096 nov 10 11:36 Música
945221 drwxr-xr-x 3 luis luis 4096 nov 10 11:47 snap
943531 drwxr-xr-x 2 luis luis 4096 nov 14 23:43 Documentos
945159 drwxr-xr-x 2 luis luis 4096 nov 15 08:10 Imágenes
938921 drwxr-xr-x 3 luis luis 4096 nov 17 13:20 Escritorio
944109 -rw-r--r-- 2 luis luis 0 dic 23 12:24 notas.txt.bck
944109 -rw-r--r-- 2 luis luis 0 dic 23 12:24 notas.txt
944194 lrwxrwxrwx 1 luis luis 9 dic 23 12:26 notas.txt_s_bck -> notas.txt
```

**Figura 3.6**  
Creación de enlace simbólico.

Podemos observar cómo el número de i-nodo de *notas.txt* y *notas.txt.bck* es igual. No así el de *notas.txt.s\_bck*. Además, este último indica que es simbólico, ya que “apunta” mediante los caracteres “->” a *notas.txt* y además consta el carácter “/” en su máscara de permisos. En la siguiente imagen se detalla esquemáticamente la relación entre i-nodo, carpeta y enlace simbólico.



**Figura 3.7**  
Relación entre i-nodo, carpeta y enlace simbólico.

Los enlaces duros, una vez creados, no se distinguen del archivo original, puesto que están asociados al mismo i-nodo. Por tanto, la eliminación del archivo original, manteniendo un enlace duro asociado, no afecta en nada al segundo. No así ocurre con los enlaces simbólicos, puesto que si eliminamos el archivo original (al no disponer de enlaces duros), el enlace blando se considera “roto”, no pudiendo acceder a la información.

- d) Dispositivos. Son archivos que representan a dispositivos físicos. En el directorio /dev se localiza la mayoría de ellos. Se distinguen dos tipos de archivos de dispositivos:
1. Dispositivos por caracteres: suelen ser aquellos que no disponen de sistema de archivos, como impresoras, teclados, terminales de texto, etc. Transfieren los datos carácter a carácter.
  2. Dispositivos por bloques: almacenan la información en bloques de datos físicamente, como, por ejemplo: discos duros, cintas magnéticas, unidades flash, etc.

Además, existen dispositivos virtuales, los cuales tienen un tratamiento especial, como /dev/null, también conocido como *cubeta de bits*, puesto que se suele emplear para eliminar o descartar toda información que sea enviada a él.

La propia orden ls dispone de la opción “`- -color`”, que se encuentra activada por defecto en Ubuntu y permite discriminar el tipo de archivo según el color:

- Blanco: archivo regular.
- Verde: archivo ejecutable.
- Azul: directorio.
- Cian: enlace simbólico.
- Rojo: enlace roto.

### 3.4.2. Eliminación de ficheros

Se pueden eliminar archivos mediante la orden *rm*. Su sintaxis es la siguiente:

```
rm [-irf] lista_de_ficheros
```

Sus opciones más utilizadas son las siguientes:

- ✓ i: solicita confirmación antes de realizar la acción.
- ✓ r o R: eliminación recursiva sobre directorios.
- ✓ f: fuerza la eliminación aun estando protegido el archivo contra escritura.

Ejemplo: *rm notas.txt notas.txt.bck*

Además, también se pueden emplear caracteres comodín, como “\*” o “?”. Estos caracteres permiten generar patrones de identificación de ficheros. “\*” hace referencia a cualquier cadena de caracteres y “?” solo referencia un único carácter cualquiera. El intérprete de comandos, antes de ejecutar la orden donde se incluyan los caracteres comodines, realiza una acción llamada “expansión de comodines”, donde traduce estos comodines a todas las posibles combinaciones de caracteres, según el patrón aportado. De esta manera, se pueden referir varios archivos a la vez.



#### Actividades propuestas

**3.6.** Empleando caracteres comodín, realiza las siguientes acciones por línea de comandos:

- Eliminar todos los archivos del directorio actual.
- Elimina, solicitando confirmación, todos los ficheros que comiencen por “doc”.
- Elimina todos los archivos que comiencen por “script” y tengan un carácter más como nombre de fichero.

**3.7.** En un mismo directorio, crea un archivo llamado *origen1*. Crear un enlace duro sobre *origen1* llamado *origen\_d*. Crear un enlace simbólico sobre *origen1* llamado *origen\_s*. Lista el contenido del directorio en formato largo mostrando los números de i-nodo. Elimina el archivo *origen1*. Vuelve a listar el contenido del directorio en formato largo, mostrando los números de i-nodo. ¿Qué ha ocurrido? Eliminar el archivo *origen\_d*. Volver a listar el contenido del directorio en formato largo, mostrando los números de i-nodo. ¿Qué ha ocurrido?

### 3.4.3. Creación y eliminación de directorios

Se pueden crear directorios mediante la orden *mkdir*. Su sintaxis es la siguiente:

```
mkdir lista_de_directorios
```

La eliminación de directorios se puede efectuar mediante la orden *rm -r*, si dispone de contenido, o empleando el comando *rmdir*, si el directorio se encuentra vacío.

```
rm lista_de_directorios
```

Ejemplos:

- *mkdir scripts*: crea el directorio.
- *rmdir scripts*: elimina el directorio vacío *scripts*.
- *rm -ri ./trabajo*: elimina, pidiendo confirmación, todos los archivos del directorio *trabajo* y el propio directorio.

#### 3.4.4. Copia de archivos

El comando utilizado para copiar la información de un archivo es *cp*. Su sintaxis es:

```
cp [-irR] lista_archivos_origen destino
```

Por defecto se sobrescribirá en el destino la lista de archivos de origen, en caso de equivalencia en los nombres. Sus opciones más utilizadas son:

- ✓ i: solicita confirmación antes de sobrescribir un archivo con el mismo nombre en el destino.
- ✓ r o R: copia de forma recursiva en directorios.

Ejemplos:

- *cp -i /etc/passwd ./passwd.bck*: copia el archivo *passwd*, solicitando confirmación, a otra ubicación (desde /etc, al directorio actual) cambiando además de nombre.
- *cp -r ./backup/ ./backup/*: copia el directorio actual a *./backup/* directorios completos.

#### 3.4.5. Renombrado o movimiento de archivos

Se emplea la orden *mv* para mover o renombrar archivos. Su funcionamiento es similar a la orden *cp*, salvo que los archivos de origen desaparecen:

```
mv [-iu] lista_archivos_origen destino
```

Sus opciones más utilizadas son:

- ✓ i: se emplea para asegurarse que en el destino no exista un archivo con el mismo nombre, ya que se sobreescritaría.
- ✓ u: solo mueve archivos o directorios con el mismo nombre entre origen y destino si estos son más actuales en el origen.

Ejemplo: *mv passwd.bck ./backup/*

### 3.4.6. Impresión de archivos

La interfaz de comandos dispone de multitud de comandos para poder inspeccionar el contenido de los ficheros. Para que este pueda ser correctamente interpretado, ha de encontrarse en formato ASCII. En caso contrario, aparecerá un conjunto de caracteres no legibles.

Los principales comandos empleados para mostrar el contenido de ficheros en la interfaz textual son:

```
cat [lista_archivos] [ { >|>>|<|<<} archivo]
```

Concatena la lista de archivos, mostrándolos por pantalla. Su salida se puede redireccionar a un archivo mediante:

- > sobrescribe si existe el archivo o, en caso contrario, lo crea.
- >> añade al contenido de un fichero existente o, en caso contrario, lo crea.

Por tanto, se emplea la orden *cat* para crear archivos de escaso contenido de forma rápida. Para ello, se emplea el comando *cat*, seguido de “>” o “>>” y de un archivo. De esta manera, el intérprete de comandos espera que introduzcamos texto hasta que pulsemos la combinación de teclas CTRL+D. En la siguiente imagen, podemos ver cómo hemos creado dos archivos de texto para, más tarde, fusionarlos en uno solo.



```

luis@luis-VirtualBox:~$ cat > texto1.txt
linea1 del texto1
luis@luis-VirtualBox:~$ cat > texto2.txt
linea2 del texto2
luis@luis-VirtualBox:~$ cat texto1.txt texto2.txt > fusion_textos.txt
luis@luis-VirtualBox:~$ cat fusion_textos.txt
linea1 del texto1
linea2 del texto2

```

**Figura 3.8**  
Usos de *cat*.

Otros comandos muy utilizados son:

```
more [lista_archivos]
less [lista_archivos]
```

Permiten visualizar el contenido de ficheros por páginas. A diferencia de *cat*, se puede navegar entre páginas. El comando *less* es más versátil y potente que *more*, resultando ideal para ficheros de gran tamaño.

Existen otras órdenes que ayudan a obtener el principio o el final de archivos:

```
head [-n] lista_ficheros
tail [-n] lista_ficheros
```

Se emplean para visualizar las *n* primeras líneas (*head*) o últimas (*tail*) de uno o varios ficheros de texto. Por defecto, mostrará diez líneas.

Ejemplos:

- ✓ *head -5 /etc/passwd* : muestra las cinco primeras líneas del fichero */etc/passwd*.
- ✓ *tail -3 /etc/passwd* : muestra las tres últimas líneas del fichero */etc/passwd*.

**TOMA NOTA**

Navegación entre páginas para comandos *more* y *less*:

- Tecla barra espaciadora: pasa a la siguiente página.
- Tecla enter: pasa a la siguiente línea.
- Tecla q: sale de la visualización.
- /texto: busca el texto o expresión regular.
- n: busca la siguiente coincidencia del texto o expresión regular.
- :n: pasa al siguiente fichero.
- :p: retrocede al fichero anterior.

### 3.4.7. Cuenteo de un fichero

La orden *wc* permite mostrar, por defecto, el número de líneas, palabras y número de bytes de un fichero. Su sintaxis es la siguiente:

```
wc [-lwCL] lista_ficheros
```

Podemos seleccionar algunos contadores de manera aislada o conjunta:

- l: número de líneas.
- w: número de palabras.
- c: número de bytes.
- L: longitud de la línea más larga.

Ejemplos:

- ✓ *wc texto1.txt texto2.txt fusion\_textos.txt* : muestra las líneas, palabras y número de bytes de cada uno de los ficheros indicados.
- ✓ *wc -l texto1.txt* : solo imprime por pantalla el número de líneas de *texto1.txt*.

**SABÍAS QUE...**

El fichero */etc/passwd* almacena las cuentas de los usuarios del sistema. Cada fila correspondiente con un usuario, consta de siete campos delimitados por ":". Cada columna hace referencia, por este orden, a: usuario, password, UID, GID, descripción, home y shell.

### 3.4.8. Ordenación de un fichero

La orden *sort* permite mostrar, de forma ordenada, las líneas de un fichero. Por defecto, ordena siguiendo el orden establecido por los caracteres en ASCII, donde las letras minúsculas tienen un orden mayor que las mayúsculas.

```
sort [-fnru] [-t <delimitador>] [-k <num_campo>] lista_archivos
```

Las opciones más importantes son:

- f: ignora las mayúsculas y las minúsculas.
- r: invierte el orden.
- n: ordena numéricamente en lugar de alfabéticamente.
- u: elimina entradas repetidas.
- t *delimitador*: indica un delimitador o separador de campos.
- k: indica el número de campo por el que se va a realizar la ordenación. Por defecto, toma los espacios, tabuladores y carácter de fin de línea como delimitadores de campo.



### Actividad resuelta 3.3

Genera un archivo de texto llamado ordenacion.txt con las siguientes palabras (una por línea) y en el siguiente orden: coche, moto, Coche, Moto, 1, 100, 2, 200, Autobus, autobus.

#### SOLUCIÓN

- Muestra por pantalla de forma ordenada alfabéticamente el fichero, sin distinguir mayúsculas y minúsculas, y eliminando líneas repetidas: `sort -fu ordenacion.txt`
- Ordena numéricamente el fichero `ordenacion.txt`: `sort -n ordenacion.txt`. Puedes apreciar la diferencia con la ordenación del ejemplo anterior.
- Ordena inversamente el fichero `/etc/passwd` por nombre de usuario (primer campo), estableciendo como delimitador ":" `sort -t: -k1 -r /etc/passwd`
- Ordena por el noveno campo correspondiente a los nombres de ficheros la salida listada en formato largo del directorio actual: `ls -l | sort -k9`
- Ordena numéricamente por el tamaño de los ficheros la salida listada del directorio actual en formato largo: `ls -l | sort -k5n`

### 3.4.9. Entrada y salidas estándar. Redirecciones

En general, todos los comandos y, en consecuencia, los procesos que se generan a partir de ellos reciben la información mediante un flujo de entrada de información y los resultados son enviados mediante un flujo de salida. El sistema operativo asigna automáticamente a cada comando entradas y salidas estándar asociadas a los flujos de entrada y salida, respectivamente.

Por defecto, se asignan tres ficheros: *entrada estándar (stdin)*, *salida estándar (stdout)* y *salida de errores estándar (stderr)*. Además, estos flujos de entrada o salida se identifican por un número o descriptor de fichero: 0, 1 y 2 para *stdin*, *stdout* y *stderr*, respectivamente.

La entrada estándar nutre al comando de información para su ejecución, la salida estándar transmite el resultado y, si durante la ejecución de un comando sobreviene un error o un aviso, este se enviará a la salida de errores estándar.

Normalmente, la entrada estándar está asociada con el teclado y la salida estándar y la salida de errores estándar con la pantalla. No obstante, no siempre es así, ya que muchos comandos pueden tomar la entrada o salida estándar de otros ficheros. Además, el comportamiento normal del flujo de un comando se puede “alterar” mediante operadores que redireccionan su flujo a otros comandos, dispositivos o ficheros.

#### A) Redirecciones de la salida estándar

Podemos emplear el operador “>” para volcar la salida de una orden sobre un fichero en lugar de a la salida estándar:

```
orden > fichero
```

Si el fichero no existe, se crea un nuevo y, si existe, sobrescribe su contenido. También se puede utilizar “1>”, puesto que “1” hace referencia al descriptor de fichero.

También se puede emplear el operador “>>” para realizar la misma acción, pero, a diferencia del operador “>”, añade su contenido al fichero sin sobrescribirlo.

```
orden >> fichero
```

Ejemplos:

- ✓ Ejecutar `cat /etc/passwd`. La entrada estándar es el fichero `/etc/passwd` y la salida estándar y de errores es la pantalla.
- ✓ Ejecutar `cat > notas.txt`. La entrada estándar es el teclado y la salida estándar se redirecciona al fichero `notas.txt`.
- ✓ Ejecutar `ls /usr >> notas.txt`. La entrada estándar es el resultado de la ejecución del comando `ls /usr`, el cual no se muestra por pantalla, ya que se redirecciona al fichero `notas.txt`, donde se añade al contenido existente en este.

#### B) Redirecciones de la entrada estándar

También podemos redireccionar un fichero como entrada de una orden, en lugar de la entrada estándar. Para ello, se emplea el operador “<”:

```
orden < fichero
```

Existe una redirección particular que permite introducir texto hasta que se encuentre una línea únicamente con el delimitador establecido.

```
<<delimitador
```

```
Luis@luis-VirtualBox:~$ cat <<END
> capítulo1
> capítulo2
> END
capítulo1
capítulo2
```

**Figura 3.9**  
Uso de `cat` con <<.

Ejemplos:

- Ejecutar `cat < notas.txt`. La orden `cat` no recibe ningún fichero como argumento, por tanto, toma el fichero `notas.txt` como entrada al ser redireccionado a la entrada de la orden.
- Ejecutar `cat <<END`. La orden `cat` toma el teclado como entrada estándar hasta que se encuentre la cadena “END” en una línea.

### C) Redirección de la salida de error estándar

Durante la ejecución de un comando, tanto la salida estándar como la salida de error estándar pueden tener o no actividad. Al ser flujos diferentes, la salida de error estándar también se puede redireccionar empleando el operador `2>`:

```
orden 2> fichero
```

Del mismo modo, se puede emplear el operador “`2>>`” para añadir contenido al fichero. Ejemplos:

- ✓ Ejecutar `ls s` (dado un archivo s inexistente). El archivo “s” no existe en el directorio actual. Por tanto, al listarla, el terminal ofrece un mensaje de error por la salida estándar. Si ejecutamos la misma orden redireccionando la salida de error estándar al fichero `error.txt`, se almacenará en este el mensaje de error y no será mostrado por pantalla: `ls s 2> error.txt`.
- ✓ Ejecutar `ls -R /usr > salida.txt 2> salida_err.txt`. El comando “`ls -R /usr`” almacena su salida en el fichero `salida.txt` y los errores o avisos se almacenarán en `salida_err.txt`.
- ✓ Ejecutar `ls -R / 1>listado.txt 2>errores.txt`. El listado del directorio raíz de manera recursiva se redirige al archivo `listado.txt`, mientras que todos los errores generados (principalmente, porque no se poseen permisos de acceso a determinados directorios) se envían al fichero `errores.txt`.

### D) Redirección de la salida estándar y la salida de error estándar al mismo destino

Ambas salidas se pueden redireccionar al mismo destino mediante el comando `&>` o `&>>`. El primero sustituye y el segundo añade contenido al fichero.

```
orden &> fichero  
orden &>> fichero
```

Ejemplo: `ls -R / &> listado_y_errores.txt`

### E) Redirección de la salida estándar y la salida de error estándar de una orden con la entrada estándar de otra orden

Es muy común emplear el operador “tubería” o “pipe”, “|” para concatenar salidas estándar con entradas estándar entre órdenes.

`ordenA | ordenB`

Al igual que antes, con el operador `|&` podemos redireccionar la salida estándar y la salida de error estándar de una orden a la siguiente.

`ordenA | & ordenB`

Además, podemos emplear el operador “tubería” a la vez que enviamos la información de la salida estándar a la salida estándar y a un archivo mediante la orden `tee`:

`ordenA | tee fichero | ordenB`

Ejemplos:

- Ejecutar `ls -1 /usr | wc -l`. La orden `ls -1 /usr` lista en una columna una línea por cada fichero o directorio. Su salida estándar se redirecciona a la entrada estándar de `wc -l` que a su vez envía a su salida estándar (pantalla) el resultado: 9. Hemos obtenido el número de ficheros o directorios de `/usr`.
- Ejecutar `ls -1 /usr | tee listado | wc -l`. La salida estándar de `ls -1 /usr` se envía a la orden `tee` que almacena en el fichero `listado` el resultado de la orden anterior, a la vez que envía esta a la siguiente orden.

#### F) Redirección de la salida de error a la salida estándar

Se suele emplear el operador `2>&1` para que la salida de error se dirija al mismo lugar que la salida estándar.

Ejemplo: `ls -R / > listado_y_errores.txt 2>&1`

#### 3.4.10. Procesamiento de textos

Existen dos comandos ampliamente utilizados para extraer información sobre textos generados por otros comandos, ficheros o cadenas de caracteres en general. Estos son `cut` y `grep`.

El comando `cut` se emplea para obtener información a partir de la división de un fichero o cadena de caracteres en columnas. Estas columnas se pueden establecer por caracteres o por campos delimitados por un delimitador de campo.

Su sintaxis es la siguiente:

`cut -c<lista_caracteres> | -f<lista_columnas> [<-d delimitador>] fichero_texto`

Donde:

- ✓ `-c <lista_caracteres>`: corta por caracteres especificados por `lista_caracteres`.
- ✓ `-f <lista_columnas> [<-d delimitador>]`: corta por campos establecidos por `lista_columnas`. Por defecto, los delimitadores son: espacio, tabuladores o espacio fin de línea, a menos que se especifique otro mediante la opción `-d`. No se pueden ejecutar conjuntamente las opciones `-c` y `-f`.

Ejemplos:

- Obtener el primer campo del fichero `/etc/passwd`: `cut -f1 -d: /etc/passwd`. El comando `cut` toma la entrada estándar del fichero `/etc/passwd`. Los campos se establecen por el delimitador ":" gracias a la opción "-d". La opción `-f1` hace referencia al primer campo. Si quisieramos añadir más campos, bastaría con emplear comas para campos independientes o guiones para intervalos de campos, como: `cut -f1,4-7 -d: /etc/passwd`.
- Obtener los caracteres 1º, 2º, 3º, 4º y 20º del fichero `/etc/passwd`:

```
cut -c1-4,20 /etc/passwd
```

Por otro lado, el comando `grep` localiza un patrón en uno o varios ficheros, mostrando las líneas donde se encuentra. Su sintaxis es la siguiente:

```
grep [-nvlicw] patrón fichero_texto [fichero_texto ...]
```

Las opciones más empleadas son:

- ✓ l: solo muestra los ficheros que contienen el patrón especificado.
- ✓ i: elimina la distinción entre mayúsculas y minúsculas.
- ✓ c: muestra el número de líneas totales que cumplen con el patrón para cada fichero.
- ✓ w: localiza el patrón como palabra y no como parte de una cadena de texto.
- ✓ n: imprime el número de línea del patrón localizado.
- ✓ v: busca líneas que no contengan el patrón especificado.

Además, grep permite emplear expresiones regulares básicas (patrones que definen un conjunto de cadenas de texto). Las más utilizadas son las que se muestran en el cuadro 3.1.

**CUADRO 3.1**  
**Expresiones regulares**

Símbolo	Significado	Ejemplos
.	Cualquier carácter, excepto el carácter fin de línea	Cas.
*	Cero o más repeticiones del carácter que le precede	C*
[lista]	Coincide con uno de los caracteres presentes en la lista	[aCgh]
	Se puede indicar la negación de la coincidencia de un patrón	[^aCgh]
	Se pueden indicar rangos de caracteres, si se incluyen guiones y estos caracteres se especifican de mayor a menor	[0-9] [^0-9]
^	Comienzo de línea	^C
\$	Fin de línea	a\$

Los símbolos que tienen un significado especial se pueden emplear si se anteceden con el carácter "\ " en los patrones. Como, por ejemplo "\\*" o "\^".

El comando *grep* puede utilizarse con la opción “*-E*”. De esta forma, su uso sería equivalente al comando *egrep*. Este último permite usar expresiones regulares extendidas, pudiendo formar patrones más complejos y potentes. Es recomendable la lectura del manual de ayuda de *grep* o *egrep*: *man grep*.

### Actividad resuelta 3.4



- Localizar aquellas líneas que contengan el patrón “root” en el fichero */etc/passwd*:

```
grep root /etc/passwd
```

- Crear un fichero de texto para trabajar con él. Este fichero se va a titular *Pirata.txt* y contendrá el siguiente texto:

```
Con diez cañones por banda,  
viento en popa a toda vela,  
no corta el mar, sino vuela,  
un velero bergantín:  
bajel pirata que llaman  
por su bravura el Temido,  
en todo mar conocido  
del uno al otro confín.
```

**Figura 3.10**  
Contenido de un fichero de texto.

- Listar aquellas líneas que tengan el patrón “el mar”, mostrando además el número de línea donde se encuentren dentro del fichero:

```
grep -n "el mar" Pirata.txt
```

En este caso, se ha de entrecollar el patrón, ya que contiene un espacio. Si no se entrecollará, resultaría un error al tomar dos argumentos en lugar de uno, siendo interpretado “mar” como un archivo.

- Mostrar aquellas líneas que tienen una “u” seguida de cualquier otro carácter:

```
grep u. Pirata.txt
```

- Mostrar aquellas líneas con palabras con “u” seguida de cualquier otro carácter:

```
grep -w u. Pirata.txt
```

- Mostrar aquellas líneas que comienzan por “en”:

```
grep ^en Pirata.txt
```

- Mostrar aquellas líneas que terminan en “,”:

```
grep ,\$ Pirata.txt
```

- Mostrar aquellas líneas que comiencen por “C”, seguidas de cero o más repeticiones de cualquier carácter y terminen en “,”:

```
grep ^C.*\$ Pirata.txt
```

- Mostrar aquellas líneas que contengan cadenas de caracteres que no contienen una letra mayúscula, seguida del carácter “a” y del carácter “,”:

```
grep [^A-Z][a], Pirata.txt
```

- Mostrar aquellas líneas que no contengan la palabra “en”:

```
grep -v -w en Pirata.txt
```

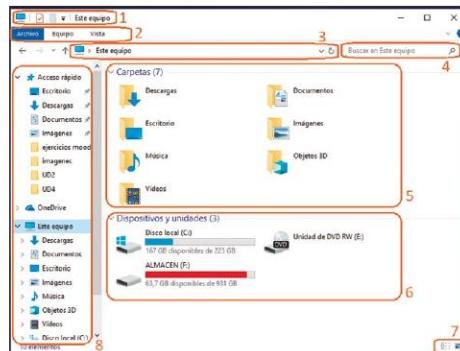


### 3.5. Gestión de archivos por interfaz gráfica en Microsoft Windows

El 'Explorador de archivos' de Microsoft Windows facilita la gestión de los archivos sin necesidad de conocer la administración interna del sistema. Este dispone de accesos rápidos a carpetas muy utilizadas: 'Escritorio', 'Imágenes' o 'Descargas', así como a las unidades conectadas. En la figura 3.11 se detallan sus partes:

1. Barra de herramientas de acceso rápido.
2. Cinta de opciones.
3. Barra de direcciones.
4. Cuadro de búsqueda.
5. Lista de archivos: carpetas de usuario.
6. Lista de archivos: dispositivos y unidades.
7. Iconos de vista de los archivos.
8. Panel de navegación.

Su estructura es bastante intuitiva, lo que permite realizar cualquier operación sobre los elementos de diversas maneras.



**Figura 3.11**  
Explorador de archivos.

El propio 'Explorador de archivos' se puede configurar mediante la 'Barra de herramientas' de acceso rápido (pulsando en el botón con forma de triángulo), habilitando o deshabilitando algunas opciones.

El 'Explorador de archivos' se adapta a la navegación de la estructura de carpetas. De esa manera, dependiendo si nos encontramos en una unidad, carpeta o archivo, el entorno se va adaptando, así como las operaciones que podemos hacer con cada uno.

La cinta de opciones nos resulta de mucha utilidad, puesto que contiene en modo ícono los comandos que podemos realizar sobre cada elemento sobre el que nos encontremos: copiar, pegar, mover a otro lugar, copiar a otro lugar, eliminar, cambiar el nombre, crear una nueva carpeta o un acceso directo, ver sus propiedades, etc. Todo ello en la pestaña 'Inicio'.

Además, permite de forma muy sencilla compartir de diversas maneras un elemento en la pestaña 'Compartir'.

**TOMA NOTA**

Todas estas opciones se pueden realizar a través del menú contextual si nos situamos sobre un elemento y pulsamos el botón secundario del ratón.

También podemos modificar la manera de organizar los archivos y directorios en la pestaña 'Vista'. Por ejemplo, con la vista 'Detalles' de archivos, disponemos de más información propia de cada elemento. Incluso podemos añadir o quitar propiedades si nos situamos en la barra de detalles con el botón secundario del ratón.

En la propia pestaña 'Vista', el botón 'Opciones' permite configurar multitud de características. Entre ellas, destacan las opciones de configuración avanzada de la pestaña 'Ver', donde podemos mostrar archivos, carpetas o unidades ocultas, ocultar archivos protegidos del sistema operativo u ocultar extensiones de los archivos. Algunas de estas opciones se encuentran también en la cinta de opciones.

Además, en la pestaña 'Herramientas de unidad' (accesible cuando se selecciona una unidad) se gestionan las unidades de almacenamiento, pudiendo cifrarlas, desfragmentarlas, liberar espacio o formatearlas.

Para una mayor agilidad en la gestión de archivos se suelen hacer uso de teclas combinadas junto con el ratón o combinaciones de teclas rápidas.

**CUADRO 3.2**  
Combinaciones en Windows

Combinaciones	Descripción
Ctrl+x	Cortar el elemento seleccionado
Ctrl+c	Copiar el elemento seleccionado
Ctrl+v	Pegar el elemento seleccionado
Ctrl+z	Deshacer una acción
F2	Modificar el nombre del elemento
Ctrl+e	Seleccionar todos los elementos
Ctrl+d	Eliminar el elemento seleccionado y enviarlo a la papelera de reciclaje
Ctrl+click ratón sobre elementos	Seleccionar distintos elementos

### 3.6. Gestión de almacenamiento por línea de comandos en Linux

La estructura de directorios de Linux incluye los diferentes discos y particiones que el sistema operativo es capaz de gestionar. Linux puede tratar con una partición de disco cuando esta contiene un sistema de archivos y se anexa a su árbol de directorios mediante un directorio común, al que se denomina *punto de montaje*. Este directorio es uno más entre el conjunto de directorios, al que se le otorga la capacidad de acceder a través de él a un *subsistema de archivos*.

**RECUEDE**

- ✓ Ya sabemos que la partición es una división física del disco duro, que permite organizar la información, incrementar la eficiencia de acceso a los datos en el disco, aumentar la seguridad e instalar diferentes sistemas de archivos, así como sistemas operativos.

Los dispositivos de almacenamiento se administran a través del directorio del sistema `/dev` (al igual que el resto de dispositivos físicos del equipo), el cual contiene archivos que se agrupan por tipos, atendiendo al comienzo del nombre. Los más importantes en cuanto a la gestión de almacenamiento son los siguientes:

- `/dev/hd*`: interfaz para unidades de disco duro IDE.
- `/dev/sd*`: interfaz para discos SCSI, SATA y unidades con conexión USB (unidades FLASH o discos duros externos).
- `/dev/tty*`: consolas o terminales físicos. Para cambiar entre consolas, se ha de utilizar la combinación de teclas CTRL+ALT+F1..F6. Si queremos volver a la consola gráfica o entorno gráfico se emplea la combinación de teclas CTRL+ALT+F7.
- `/dev/ttys*`: puertos serie.
- `/dev/sr*` y `/dev/scd*`: interfaz para unidades CD o DVD.

De tal manera, que la sintaxis para determinar la identificación de cada dispositivo o partición es la siguiente:

`/dev/<id_dispositivo><letra_orden><numero_partición>.`

Ejemplos:

- ✓ `/dev/hda`: primer disco duro IDE.
- ✓ `/dev/sdb`: segundo disco duro SCSI, SATA o con conexión USB.
- ✓ `/dev/sdc2`: segunda partición del tercer disco duro SCSI, SATA o con conexión USB.
- ✓ `/dev/ttys0`: primer puerto serie.

En discos con sistema de particiones MBR, a las particiones primarias se le asignan los números del 1 al 4 y, las lógicas, desde el 5 en adelante.

### 3.6.1. Montaje y desmontaje

La orden que realiza el montaje de un sistema de archivos es `mount`. Gracias a ella, se monta cualquier sistema de archivos reconocible por el núcleo de Linux en un punto de montaje del sistema. Todos los sistemas de archivos se montan directamente por nosotros o indirectamente durante el arranque del sistema, a excepción del sistema de archivos raíz “`/`”, que se asocia a un punto de montaje compilado en el propio kernel y que monta la partición especificada durante la instalación. La sintaxis de la orden `mount` es la siguiente:

```
mount [-avwrt] [tipo] [dispositivo] [punto_de_montaje]
```

Las opciones más empleadas son:

- **-a:** monta los sistemas de archivos presentes en `/etc/fstab`, salvo que se indique el parámetro `noauto`, que impediría el montaje por esta opción.
- **-v:** muestra información del proceso de montaje.
- **-w:** monta el sistema de archivos con permisos de lectura y escritura.
- **-r:** monta el sistema de archivos con permisos de solo lectura.
- **-t <tipo>:** indica el tipo de sistema de archivos para montar. Este puede ser, entre otros: `ext`, `ext2`, `ext3`, `ext4`, `ntfs`, `nfs`, `iso9660`, `msdos`, `vfat`, `hfs`, `hfsplus` o `smbfs`.

El sistema mantiene actualizada una lista de sistemas de archivos montados a través del archivo `/proc/self/mounts` (en versiones anteriores se empleaba `/etc/mtab`, que actualmente es un enlace simbólico al anterior). Este se interpreta cuando ejecutamos `mount` sin argumentos y se actualiza al montar nuevos sistemas de archivos o al desmontar sistemas de archivos existentes. Por tanto, la orden `mount` sin modificadores y argumentos lista los sistemas de archivos montados actualmente en el sistema.

```
luis@luis-VirtualBox:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,noexec,relatime,size=998080k,nr_inodes=249520,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,noexec,relatime,size=204116k,mode=755)
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro)
```

**Figura 3.12**  
Ejemplo de ejecución de `mount`.

Por defecto, Ubuntu monta automáticamente un sistema de archivos. Esto se da, por ejemplo, cuando Linux detecta que un pendrive ha sido conectado en un puerto USB o cuando se inserta un nuevo disco DVD en la unidad lectora. Esta opción se puede deshabilitar, aunque realmente resulta cómoda.

Las unidades Flash USB u otros dispositivos portables son gestionados por el gestor de archivos de Linux que, en el caso de *Ubuntu 18.10*, es *Nautilus*, y se encuentra en el entorno de escritorio de *GNOME*. Este gestor automatiza su conexión.

Si realizamos un montaje, hemos de indicar el tipo de sistema de archivos, la partición y el punto de montaje donde se situará el nuevo sistema de archivos (este último ha de existir). No obstante, se han de seguir una serie de pasos cuando deseamos montar un sistema de archivos:

1. Detectar el nombre asignado por el sistema a la partición objeto de montaje. Una manera de conocerlo es realizar un listado de las unidades o particiones recientemente creadas. Si sabemos que es un disco SCSI, SATA o con conexión USB, podemos ejecutar `ls -ltr /dev/sd*`. Otras formas de averiguarlo, son mediante los comandos:

- `lsblk`: lista la información de los dispositivos por bloques: nombres, tipos, puntos de montaje, tamaños, etc. Si ejecutamos `lsblk -fs`, mostrará el sistema de archivos.
- `lshw -C disk`: muestra información detallada de los discos conectados al sistema.
- `fdisk -l`: muestra información de los dispositivos o particiones que figuran en `/proc/partitions`, el cual registra los dispositivos conectados.
- `lsusb`: muestra información de los buses USB y los dispositivos conectados a ellos. Esto permite recabar información sobre los tipos de dispositivos USB conectados: `lsusb -tv`.
- `dmesg`: lista el contenido del buffer de mensajes del núcleo de Linux. Este comando muestra gran cantidad de información. En nuestro caso, podemos averiguar el

último dispositivo conectado ejecutando este comando justo después de conectar un dispositivo, mostrando una serie de mensajes relativos a la conexión de un nuevo dispositivo USB y su posterior configuración. En caso de que el sistema tenga mucha actividad, podríamos filtrar, por ejemplo, con `dmesg | grep sd`.

2. Crear el punto de montaje, si este no lo está.
3. Montar el sistema de archivos en el punto de montaje. Por defecto, solo el usuario `root` tiene permisos para ejecutar este comando.

```
luis@luis-VirtualBox:~$ sudo mount -t vfat /dev/sdb1 /home/luis/pendrive
luis@luis-VirtualBox:~$ cd /home/luis/pendrive
luis@luis-VirtualBox:~/pendrive$
```

**Figura 3.13**  
Ejemplo de montaje de sistema de archivos.

Dado el ejemplo de la imagen anterior y a partir de ese momento, el sistema de archivos del pendrive cuelga del directorio `/home/luis/pendrive` y toda acción sobre él afectará al propio sistema de archivos del dispositivo. La opción `-t` resulta optativa, puesto que `mount` intenta averiguar el tipo de sistema de archivos.



### Actividades propuestas

- 3.8.** Investigar otros dos gestores de archivos empleados en GNU/Linux que estén asociados a otros entornos de escritorio.
- 3.9.** `mount` emplea el comando `blkid` y el archivo `/proc/filesystems` para intentar montar una partición con un sistema de archivos no indicado. Utilizando el comando `blkid -k` y accediendo al contenido del archivo `/proc/filesystems`, realiza un listado de todos los sistemas de archivos que Ubuntu puede manejar.

Para dejar de usar completamente o extraer un dispositivo con sistema de archivos, este se debe desmontar. Para el desmontaje se emplea el comando `umount`, el cual puede completarse si no existen directorios en uso del sistema de archivos objeto a desmontar o procesos lanzados desde él. Para ejecutar el desmontaje se puede indicar la partición o el punto de montaje:

```
umount <dispositivo> o umount <punto_de_montaje>
```

Para el ejemplo anterior son igualmente válidos:

```
umount /dev/sdb1
umount /home/Luis/pendrive
```

Los sistemas operativos GNU/Linux automatizan el proceso de montaje de particiones gracias al archivo editable `/etc/fstab`. Generalmente, los discos duros internos y unidades de CD/DVD son los que se especifican en este archivo. De manera que las particiones que se detallen en él se montarán durante el arranque del sistema operativo.

Una partición que no figure en este archivo solo podrá ser montada y desmontada por un usuario administrador o `root`.

El fichero /etc/fstab se estructura por columnas separadas por espacios de la siguiente manera.

- ✓ File system: partición.
- ✓ Mount point: punto de montaje.
- ✓ Type: tipo de sistema de archivos que contiene la partición. En caso de indicar "auto", detecta el sistema de archivos automáticamente.
- ✓ Options: opciones de montaje. Son semejantes a las del comando mount. Las opciones más comunes son:
  - auto: monta el sistema de archivos durante el arranque. Este es el valor por defecto.
  - noauto: el sistema de archivos se montará solo manualmente.
  - ro: monta el sistema de archivos en modo solo lectura.
  - rw: monta el sistema de archivos en modo lectura-escritura.
  - user: permite a cualquier usuario montar el sistema de archivos.
  - users: cualquier usuario del grupo *users* puede montar el sistema de archivos.
  - nouser: solo el usuario *root* puede montar el sistema de archivos.
  - defaults: establece una serie de opciones de montaje predeterminadas.
  - errors=VALOR, establece una acción en caso de que en el sistema de archivos se produzca un error. Si VALOR es:
    - continue: el sistema sigue funcionando.
    - remount-ro: el sistema se reinicia en modo de solo lectura.
    - panic: se apaga el sistema.
- ✓ Dump: habilita o deshabilita la copia de seguridad mediante el comando *dump*. Normalmente no se encuentra instalado este programa, por lo que la opción más común es 0 (deshabilitado). Con valor 1, *dump* hace una copia de seguridad del sistema de archivos.
- ✓ Pass: establece el orden en el que se comprueban los sistemas de archivos. Con un valor 0 no se comprueba el sistema de archivos, 1 se comprueba en primer lugar y 2 se comprueba en segundo lugar. Normalmente, el sistema de archivos raíz ha de comprobarse en primer lugar y el resto en segundo lugar.



#### PARA SABER MÁS

##### *Desmontaje de sistemas de archivos*

Los dispositivos de almacenamiento disponen de una memoria caché de tamaño variable que actúa como unidad temporal volátil o buffer cuya principal ventaja es la velocidad de escritura o lectura y el sincronismo con el procesador, chipset y memoria RAM. En el proceso de escritura, la caché almacena aquellas modificaciones que aún no han sido realmente actualizadas en la propia memoria permanente del dispositivo. Cuando la memoria caché se llena, se vuelca su contenido al medio de almacenamiento permanente. Este proceso se efectúa en intervalos de tiempo altamente frecuentes.

Una vez terminadas las acciones sobre el sistema de archivos, este se ha de desmontar para separar el sistema de archivos del punto de montaje y evitar pérdidas de información de aquellos datos que se encuentren en caché y que aún no hayan sido volcados al almacenamiento permanente del medio, indicándole que realicen así este traspase de información.

Un ejemplo de ello se representa en la figura 3.14, donde las líneas que comienzan por el carácter “#” son comentarios y no se interpretan. Nos encontramos con una partición donde se encuentra el sistema raíz en *ext4* y un archivo de intercambio para swap.

**Figura 3.14**  
Contenido  
de /etc/fstab.

```
luts@luts-VirtualBox:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUIDs as a more robust way to name devices
#
#  <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=f10a3274-bd7c-4914-bc61-185190ff66db /          ext4    errors=remount-ro 0      1
/swapfile none            swap    sw              0      0
```

Como ya sabemos, los dispositivos y particiones se identifican con un nombre descriptivo por el tipo de dispositivo, como, por ejemplo, */dev/sda1* o */dev/sr0*. Esta nomenclatura puede dar lugar a equívocos, ya que se asocia a la conexión de estos con los puertos físicos y cableado en la placa base o al orden de almacenamiento en la BIOS. Como podemos extraer un dispositivo interno físicamente y conectarlo en otro puerto, se podría alterar la asociación de */etc/fstab* con un dispositivo. Por tanto, resulta conveniente asignar a los dispositivos por bloques, un nombre que lo identifique inequívocamente (pero evitando asignar dos etiquetas iguales a dos particiones diferentes). Esto se puede realizar:

1. Mediante una etiqueta. Se puede asignar una etiqueta a una partición mediante el programa *GParted* (cuando la partición se encuentre desmontada) o mediante otros comandos, dependiendo del sistema de archivos.

- *NTFS: ntfslabel <partición> <etiqueta>*
- *FAT: fatlabel <partición> <etiqueta>*
- *ext2/3/4: e2label <partición> <etiqueta> y tune2fs -L <etiqueta> <partición>*
- *swap: swaplabel -L <etiqueta> <partición>*

Ejemplo: sudo e2label /dev/sda1 sistemaraiz

2. Mediante un UUID. El UUID o identificador único universal se asigna a la partición cuando esta es formateada. Por tanto, no tenemos que asignarla nosotros. Sin embargo, si se vuelve a formatear una partición o se modifican sus características (como su tamaño), cambiará su UUID.

Podemos ver las etiquetas y los UUID de las particiones del sistema mediante *lsblk -fs*.

Otras formas de mostrar las etiquetas y los UUID son listando los directorios */dev/disk/by-label/* y */dev/disk/by-uuid/*, respectivamente, en sistemas MBR o */dev/disk/by-partlabel/* y */dev/disk/by-partuid/* para sistemas UEFI con GPT.

Ejemplos:

```
ls -l /dev/disk/by-label
ls -l /dev/disk/by-uuid
```

La identificación de etiquetas o UUID en el archivo */etc/fstab* es sencilla, basta con indicar *LABEL=<nombre-etiqueta>* o *UUID=<código\_UUID>* en la columna *file system*. Para ello, editamos el archivo */etc/fstab* con privilegios de *root* y añadimos las modificaciones necesarias.

### Actividad propuesta 3.10



Empleando la ayuda de `mount (man mount)`, averigua el significado de cada una de las opciones preestablecidas mediante la opción `defaults`.

#### 3.6.2. Particionar

Existen varias herramientas para crear y manipular particiones en Linux, teniendo en cuenta que la mayoría de tareas de administración con particiones necesitan privilegios de `root`.

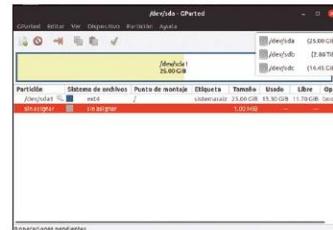
Las más conocidas son `fdisk`, `gdisk`, `parted` y `GParted`. Todas trabajan con esquemas de particionamiento MBR y GPT, a excepción de `gdisk`, que lo hace solo con GPT. Las tres primeras se manejan a través de la interfaz de texto mediante un menú interactivo. Tanto `fdisk` como `gdisk` (`GPT fdisk`) presentan un conjunto de opciones muy similares, sin embargo, `parted` dispone de otras opciones y permite, además, formatear las particiones.

Ubuntu con escritorio GNOME, incorpora la herramienta de partición de discos GNOME-disks (Discos de GNOME) a la que podemos acceder como “Discos”. Dentro de ella, se pueden seleccionar cualesquiera de las particiones en el listado de la izquierda; y a la derecha aparecerá información al respecto. Además, permite desmontar, eliminar y establecer opciones adicionales de la partición de forma sencilla:

- Formatear.
- Editar la partición.
- Redimensionarla.
- Comprobar y reparar el sistema de archivos.
- Crear y restaurar una imagen.
- Probar el rendimiento de la partición.



**Figura 3.15**  
Discos de GNOME.



**Figura 3.16**  
GParted.

La herramienta ‘GParted’ es un clásico entre los gestores de particiones en Linux, aunque no viene instalada en Ubuntu por defecto. Puede trabajar con MBR, GPT y permite formatear las particiones. Podemos descargar la aplicación a través de ‘Software de Ubuntu’.

Al igual que GNOME-disks, GParted presenta un entorno muy intuitivo. A la derecha se selecciona el disco y en el cuadro central aparece un esquema gráfico con las dimensiones de las particiones y el espacio ocupado en cada una de ellas. Marcando una partición o en espacio no particionado, podemos realizar sobre ella multitud de acciones, como:

- ✓ Crear nuevas particiones en espacio no particionado.
- ✓ Eliminar o crear nuevas particiones.
- ✓ Redimensionar o mover particiones.
- ✓ Copiar y pegar particiones en otros discos.
- ✓ Formatear.
- ✓ Asignar un nombre.

Los discos deben de disponer de una tabla de particiones para poder gestionar las particiones que en él se establezcan. Si el disco está “limpio”, se ha de crear previamente la tabla de particiones.

En el menú ‘Dispositivo’, podemos seleccionar la opción ‘Crear tabla de particiones’ y, a continuación, seleccionar el tipo de tabla de particiones, siendo las más conocidas *msdos* (*MBR*) o *gpt*.

Para crear una nueva partición, marcamos el espacio sin asignar o no particionado, y en el menú ‘Partición’ seleccionamos ‘Nueva’. Es entonces donde se establecen todos los parámetros de la partición, como el tamaño, el sistema de archivos para implantar, la etiqueta o el tipo de partición (primaria, extendida o lógica) que, en el caso de GPT, solo puede ser primaria (puesto que, como ya sabemos, no existen particiones lógicas o extendidas).

Además, se pueden redimensionar las particiones, reduciendo su tamaño desde el principio de la misma o por el final, e incluso, si hay espacio sin asignar en el disco, también se pueden ampliar.



### Actividad propuesta 3.11

Accede a la herramienta ‘Discos de GNOME’, observa la información que muestra y navega por las distintas opciones que ofrece para realizar sobre las distintas unidades.

Instala ‘GParted’ y, a partir de un nuevo disco, crea la tabla de particiones, crea nuevas particiones en GPT, instala diferentes sistemas de archivos, etíquelos y redimensiónalos.

En cuanto a las herramientas por línea de comandos, la más característica es *fdisk*. Ya conocemos la utilidad *fdisk -l*, sin embargo, esta herramienta dispone de una gran cantidad de opciones para manejar las particiones de los discos del sistema. Para ejecutar *fdisk* hemos de indicar el disco o partición sobre el que queremos trabajar: *fdisk <disco | particion>*.

Al comienzo de la ejecución del programa nos puede realizar diversos avisos, como, por ejemplo, si detecta que el disco tiene más de 2 TB, conviene emplear un esquema de particionamiento GPT.

```

luis@luis-VirtualBox:~$ sudo fdisk /dev/sdc
Bienvenido a fdisk (utilidad Linux 2.32).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0x9ac
cd118.

Orden (m para obtener ayuda): g
Se ha creado una nueva etiqueta de disco GPT (GUID: 11980E17-E492-584A-B785-A964
54622883).

Orden (m para obtener ayuda): n
Número de partición (1-128, valor predeterminado 1):
Primero sector (2048-20971486, valor predeterminado 2048):
Último sector, +sectores o +tamaño(K,M,G,T,P) (2048-20971486, valor predeterminado 20971486):

Crea una nueva partición 1 de tipo 'Linux filesystem' y de tamaño 10 GiB.

Orden (m para obtener ayuda): p
Disco /dev/sdc: 10 GiB, 10737418240 bytes, 20971520 sectores
Unidad: 512 bytes/página, 16 páginas/sector
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de t/s (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: gpt
Identificador del disco: 11980E17-E492-584A-B785-A96454622883

Dispositivo Contenido Final Sectores Tamaño Tipo
/dev/sdc1          2048 20971486 20969439   10G Sistema de ficheros de Linux

Orden (m para obtener ayuda): w
Se ha modificado la tabla de particiones.
Llamando a ioctl() para volver a leer la tabla de particiones.
Se están sincronizando los discos.

```

Figura 3.17. Uso de *fdisk*.

Su funcionamiento consiste en introducir la tecla adecuada según el menú del programa. Al pulsar “m” podemos ver un listado de todas las acciones.

Es muy importante guardar y salir antes de terminar con la ejecución del programa; de lo contrario, no tendrán efecto las acciones sobre la tabla de particiones.

En la imagen 3.17 se crea una tabla de particiones nueva GPT, con una partición que ocupa todo el espacio del disco con *fdisk*.

En cuanto al programa *gDisk* (*GPT fdisk*), emplea un conjunto de acciones muy similares a *fdisk* pero hemos de tener precaución, ya que las modificaciones que se realicen sobre discos MBR serán modificadas a GPT y, por tanto, pueden causar errores inesperados en BIOS MBR o UEFI en modo heredado. Por otro lado, durante la creación de una partición en *gDisk* nos solicita que indiquemos mediante un código en hexadecimal el tipo de partición. Esta información es una recomendación para que sea reconocida la partición. En *fdisk* también se puede establecer.

### Actividad resuelta 3.5



*Crear una tabla de particiones GPT en un disco mediante gdisk, estableciendo dos particiones que ocupen espacios similares en cuanto a tamaño.*

#### SOLUCIÓN

Lanzamos el programa pasándole como argumento un disco flash USB de 14.4 GB: *sudo gdisk /dev/sdc*, mostrando la tabla de particiones actual del disco. La primera opción es crear una tabla de particiones nueva “o”.

A continuación, creamos una nueva partición “n”, y el programa nos preguntará el número de partición, siendo la primera (opción por defecto). Luego preguntará por el sector de comienzo de la partición, e indicaremos la opción por defecto para que no exista espacio sin particionar. Más tarde, hemos de indicar el último sector o tamaño de la partición que en nuestro caso indicamos 7GB (+7G). Por último, pregunta el tipo de partición ,que también estableceremos la opción por defecto.

```
Command (? for help): n
Partition number (1-12, default 1): 
First sector (34-30299586, default = 2048) or (+)-size[KMGTP]: +7G
Last sector (2048-30299486, default = 30299486) or (+-)size[KMGTP]: +7G
Disk identifier (GUID) (L to show codes, Enter = 03000): 
Hex code or GUID (L to show codes, Enter = 03000): 
Changed type of partition to 'Linux filesystem'

Command (? for help): P
Disk /dev/sdc: 30299520 sectors, 14.4 GiB
Model: USB DISK 2.0
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): F040D010-702E-47f0-A465-6E8F8C1C43FA
Partition table holds up to 128 entries
All partitions will be aligned on 2048-sector boundaries
First usable sector is 34, last usable sector is 30299486
Partitions will be aligned on 2048-sector boundaries
Total free space is 15619389 sectors (7.4 GiB)

Number  Start (sector)  End (sector)  Size            Code  Name
 1          2048        14682111   7.0 GiB         0300  Linux filesystem
```

**Figura 3.18**

*Creación de una primera partición con gdisk.*

Después, mostramos la tabla de particiones para comprobar las acciones tomadas.

Ahora repetimos los pasos anteriores, como indica la figura 3.20, pero sin indicar el tamaño de la partición, con objeto de ocupar el resto de espacio en disco.

```

Command (? for help): n
Partition number (1-128, default: 2):
First sector (34-38299488, default: 14682112) or (+)-size(MGiB):
Last sector (14682112-38299486, default = 38299486) or (+)-size(MGiB):
Current type is 'Linux filesystem'
New GUID (L to show list, Enter = 0x80):
Hex value of GUID (L to show list, Enter = 0x800):
Changed type of partition to 'Linux filesystem'

Command (? for help): p
Disk /dev/sdc: 28299488 sectors, 14.4 GiB
Model: USB DISK 2.0
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): F04D8010-702E-4760-A465-6EBFBC1C43FA
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 38299486
Partitions will be aligned on 2048-sector boundaries
Total free space is 2814 sectors (1407.0 KiB)

Number  Start (sector)   End (sector)  Size    Code  Name
 1        2048          14682111    7.0 GiB  8300  Linux filesystem
 2       14682112         38299486    7.4 GiB  8300  Linux filesystem

```

**Figura 3.19**  
Creación de una primera partición con gdisk.

Y, por último, hacemos efectivas las modificaciones hechas con "w".

```

Command (? for help): w
Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!
Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/sdc.
The operation has completed successfully.

```

**Figura 3.20**

Guardar los cambios con gdisk.

### 3.6.3. Formatear

Ya sabemos que muchas herramientas de particionamiento, como *GNOME-disks*, *GParted* y *parted*, permiten formatear las particiones creadas. Pero existen otras en modo texto que se encargan únicamente de realizar esa función.

El comando *mkfs* presenta la siguiente sintaxis:

```
mkfs [-t tipo_sistema_archivos] [opciones] dispositivo
```

Donde *dispositivo* es normalmente una partición, y las opciones más comunes son:

- *t tipo\_sistema\_archivos*: tipo de sistema de archivos que se desea implantar como *ext2*, *ext3*, *ext4*, *vfat* (*msdos*) o *ntfs*, entre otros.
- *opciones*: se pueden especificar otras muchas opciones, como etiquetas, formato rápido, tamaño del cluster, etc. Estas opciones varían según el constructor, al que llama *mkfs*. Por tanto, para mayor detalle, hemos de familiarizarnos con la ayuda (*man*) de estos comandos.

Para formatear una partición, el dispositivo ha de estar desmontado.

Ejemplo: *mkfs -t ntfs /dev/sdc1*

*mkfs* es un *front-end* de otros programas que implantan sistemas de archivos. Es decir, *mkfs* llama al programa constructor del sistema de archivos, como son *mkfs.ext2*, *mkfs.ext3*, *mkfs.ext4*, *mkfs.vfat*, *mkfs.ntfs*, etc., dependiendo del sistema de archivos para implantar. Y algunos de estos son enlaces simbólicos a otros, como *mke2fs*, *mknntfs* o *mkfs.fat*. Podemos analizarlo si ejecutamos *ls -l /sbin/mk\**.

Por tanto, las siguientes acciones son equivalentes:

```
sudo mkfs.ntfs /dev/sdc1  
sudo mknntfs /dev/sdc1  
sudo mkfs -t ntfs /dev/sdc1
```

Además de los sistemas de archivos antes descritos (propios del núcleo), podemos incorporar otros que sean reconocidos por GNU/Linux. Como muestra de ello, el sistema de archivos exFAT no se incorpora dentro del núcleo, por lo que si deseamos hacer uso de él para montar, leer, escribir o formatear un sistema de archivos de este tipo, podemos instalar el paquete asociado: *sudo apt install exfat-utils*.

Esta orden instalará los paquetes *exfat-utils* y *exfat-fuse*. El término *FUSE* (Filesystem in Userspace) hace mención a que el nuevo sistema de archivos se implementará y será gestionado aparte de los sistemas de archivos propios del núcleo.

Ejemplo: *sudo mkfs -t exfat -n pendriveExFAT /dev/sdd1*

A partir de este momento, ya podríamos hacer uso de este sistema de archivos cuando sea montado.

### 3.6.4. Desfragmentación

La fragmentación de un sistema de archivos se entiende como la disgregación o el esparcimiento de datos, relacionados entre sí, en el medio de almacenamiento. En principio, este hecho no presenta problema alguno, puesto que el sistema de archivos conoce y gestiona los bloques de datos de cada archivo. Sin embargo, los discos duros mecánicos se ven penalizados por esta característica, ya que el cabezal de lectura y escritura ha de oscilar continuamente para seguir los bloques de un archivo, deteriorándose los tiempos de lectura y escritura de dichos bloques.

Cuando existe mucha fragmentación en un disco, se generan multitud de huecos donde se pueden almacenar, o no, bloques de archivos distintos. Al proceso de unión de los bloques de datos de un mismo archivo se le denomina *desfragmentación*.

La fragmentación y sus características dependen de cada sistema de archivos. En sistemas de archivos *NTFS* y *FAT* la desfragmentación es común para mejorar su rendimiento.

Los sistemas de archivos Linux, en general, no necesitan desfragmentarse, puesto que emplean sistemas inteligentes de asignación de bloques a ficheros, de manera que raramente estos se han de desfragmentar. Normalmente, dejan un espacio en bloques considerable entre ficheros para futuros crecimientos, evitando que el crecimiento de un fichero suponga la división de bloques en otra parte del disco y que, si se reduce un fichero, pueda ocupar el espacio libre una parte de otro fichero, evitando más fragmentación.

Sin embargo, sistemas de archivos Linux con dispositivos con poco espacio en disco y mucho movimiento, creación y eliminación frecuente de ficheros es posible que se deban

desfragmentar. Por ello, el sistema de archivos *ext4* puede emplear la utilidad *e4defrag* para este propósito.

El comando *e4defrag* presenta la siguiente sintaxis:

e4defrag [-cv] objetivo

Donde *objetivo* puede ser un archivo, un directorio o un dispositivo con sistema de archivos *ext4* que emplea *extents*. Las opciones más comunes son:

- ✓ *c*: muestra un recuento de la fragmentación actual con respecto a la fragmentación ideal. Si se combina con la opción *v*, lo muestra para cada archivo. Con esta opción no se hace efectiva la fragmentación.
  - ✓ *v*: muestra el recuento para cada archivo antes y después de la desfragmentación.

## Ejemplo de uso de e4defrag

La figura 3.21 muestra una estimación de la desfragmentación:

```
[root@ahora ~]# sudo wiperflag -c
wiperflag 1.0.0 (18-Aug-2018)
-Fragmented files:
  1. /home/ahora/backup/cursor/0/ahora/ubuntu-18.04.1-desktop-amd64/filesystem.squashfs
    now/best      512K/ext
  2. /home/ahora/.local/share/Trash/files/ahora/ubuntu-18.04.1-desktop-amd64/casper/filesystem.squash
    19/1          97944 KB
  3. /home/ahora/.local/share/Trash/files/ahora/ubuntu-18.04.1-desktop-amd64/casper/filesystem.squash
    17/1          104644 KB
  4. /home/ahora/.local/share/ahora/ahora-18.04.1-desktop-amd64/ahora-18.04.1-desktop-amd64/ahora-18.04.1-de
    1/1           13 KB
  5. /home/ahora/salida.sst
    1/1           3892 KB
  6. /home/ahora/.config/user-dirs.locale
    1/1           4 KB

Total Best extents      $130/5072
Average size per extent       835 KB
```

**Figura 3.21**  
Estimación  
de desfragmentación.

La figura 3.22 muestra la ejecución de desfragmentación en el directorio actual:

```
luis@luis-VirtualBox:~$ sudo e4defrag -v
```

**Figura 3.22**

La figura 3.23 muestra la parte final del proceso de desfragmentación, ofreciendo un resumen

```
[Fragmented Files]          new/best    size/ext
1. /home/lukas/.IceAuthority      2/3        4 KB
2. /home/lukas/.cache/mozilla/firefox/lighthung/default/cache2/entries/0865f652073E10728375D3057B1B88C228A47      2/3        4 KB
3. /home/lukas/.cache/mozilla/firefox/lighthung/default/cache2/entries/17375F26E05D923B4570AC278078779895      2/3        4 KB
4. /home/lukas/.cache/mozilla/firefox/lighthung/default/cache2/entries/AB81A0187E7455D29C77C885D9EBC95C8C989B95      2/3        4 KB
5. /home/lukas/.bash_history      2/3        4 KB
6. /home/lukas/.bash_history      2/3        4 KB

[total: best extents           525/5072
[total: fragmentation extent 185 KB
[fragmentation score          0
[0 no problem: 35-55 a little bit fragmented: 56- needs defrag]
This directory (.) does not need defragmentation.
Done.
```

**Figura 3.23**  
Parte final del proceso  
de desfragmentación.

**Actividad propuesta 3.12**

Desfragmenta una unidad con sistema de archivos `ext4`.

### 3.6.5. Chequeo

El componente más importante en cualquier sistema informático son los datos. Estos tienen un valor incalculable y, si se pierden, sin tener una copia de los mismos, los resultados pueden ser catastróficos.

Los discos duros, a lo largo de su vida útil, suelen ser fuente de diversos problemas, originados internamente o por agentes externos. Nos referimos a:

- a) Malware.
- b) Fallos en componentes electrónicos del disco duro (normalmente en su placa de circuito impreso por humedad, condensación o suministro eléctrico).
- c) Fluctuaciones de tensión en el suministro de energía. Ya sean por picos de tensión, bajas de esta, fluctuaciones periódicas o cortes de luz.
- d) Daños físicos, como caídas o simplemente elementos deteriorados por el uso. Especialmente en discos mecánicos, ya que estos son más propensos al desgaste o problemas debidos por alguna de sus partes móviles (motor, cabezal de lectura-escritura, etc.).
- e) Errores firmware o de actualización de drivers.

La gran mayoría de discos duros emplea la tecnología *S.M.A.R.T.* (*Self Monitoring Analysis and Reporting Technology*) con capacidad para detectar e informar de errores o fallos.

Cuando se combina esta tecnología junto con software compatible, ya sea la propia BIOS del equipo u otro software de terceros instalado en el sistema operativo, obtenemos información muy valiosa sobre el estado actual del disco, e incluso puede avisar de un fallo inminente.



### Recursos web

Herramientas compatibles con la tecnología SMART:

- GSsmartControl para Linux: <https://gsmartcontrol.sourceforge.io/>
- CrystalDiskInfo para Microsoft Windows: <https://crystalmark.info/en/software/crystaldiskinfo/>

La monitorización del sistema de archivos se ha de realizar continuamente o chequearse periódicamente dependiendo de la importancia de los datos que almacene el disco, por lo que es muy recomendable instalar una aplicación de monitorización del estado de los discos duros.

En cuanto a utilidades propias de Linux, este dispone principalmente de los comandos *fsck* y *e2fsck*. Al igual que ocurre con *mksfs*, *fsck* es un *front-end* de otros programas que chequean sistemas de archivos, como *e2fsck* que permite chequear sistemas de archivos de la familia *ext*. Por tanto, este último es el apropiado y el que se ejecuta para chequear sistemas *ext2*, *ext3* o *ext4*. Podemos ejecutar *ls -l /sbin/\*fsck\** para listar los ejecutables relacionados con el chequeo.

El comando *fsck* presenta la siguiente sintaxis:

```
fsck [-A] [-V] [-t tipo_sistema_archivos] [-a] [-r] [sistema_de_archivos]
```

Donde las opciones más comunes son:

- A: Chequea todos los sistemas de archivos establecidos en */etc/fstab*. Esta opción se lanza durante el arranque de Linux. Con esta opción no se puede emplear el argumento *sistema\_de\_archivos*.
- V (verbose): detalla las acciones realizadas por *fsck*.
- t *tipo\_sistema\_archivos*: tipo de sistema de archivos que se desea chequear como *ext2*, *ext3*, *ext4*, *fat* (*msdos*) o *ntfs*, entre otros.
- a: repara sin pedir confirmación.
- r: pide confirmación antes de reparar los daños.

Ejemplo: *sudo fsck -t ext4 /dev/sdb1*

El comando *e2fsck* presenta la siguiente sintaxis:

```
e2fsck [-pcnyvD] sistema_archivos
```

Donde las opciones más comunes son:

- ✓ p: repara el sistema de archivos automáticamente. No es compatible con las opciones *-n* o *-y*.
- ✓ c: hace uso del programa *badblocks*, que busca bloques defectuosos. En caso de localizar alguno, lo añade a una lista de bloques defectuosos y evita así su uso.
- ✓ y: responde afirmativamente a todas las respuestas que *e2fsck* pudiera plantear.
- ✓ n: responde negativamente a todas las respuestas que *e2fsck* pudiera plantear.
- ✓ v: detalla las acciones realizadas por *e2fsck*.
- ✓ D: optimiza los directorios para un mejor acceso a los datos.

Ejemplo: *sudo e2fsck /dev/sdb1 -c*

Por tanto, las anteriores acciones son equivalentes a *sudo fsck.ext4 /dev/sdb1*.

Además de los sistemas de archivos propios del núcleo, podemos incorporar otros que sean reconocidos por GNU/Linux y así, pues, chequearlos con la utilidad correspondiente (como, por ejemplo, la utilidad *exfatfsck*).

Es altamente recomendable desmontar un sistema de archivos antes de proceder a su chequeo, así aseguramos la integridad de los datos, al no estar usándose los archivos asociados. Además, cuando se hayan efectuado cambios mediante estos programas de chequeo, es aconsejable reiniciar el sistema (*shutdown -r now*).

**RECUERDA**

- ✓ El fichero `/etc/fstab` automatiza el chequeo de los sistemas de archivos durante el arranque mediante `fsck`, estableciéndose su prioridad en su columna `pass`. Por ello, es importante establecer a 1 el valor `pass` para el sistema de archivos raíz de Linux, ya que, de lo contrario, no podremos hacer el chequeo con garantías una vez el sistema se haya iniciado.

### 3.6.6. RAID

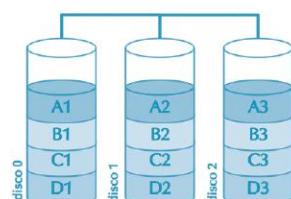
Cuando hablamos de seguridad en la información y eficiencia en la accesibilidad a los datos en los discos duros, debemos detenernos en el concepto de RAID de forma obligatoria. Es una de las maneras más eficaces de evitar la pérdida de datos y aumentar el rendimiento en tareas de lectura o escritura en discos duros.

RAID (Redundant Array of Independent Disks) consiste en establecer un modo de trabajo, nivel o configuración de un grupo de discos para aumentar la integridad, la capacidad total de almacenamiento, la velocidad de transferencia o disminuir el riesgo a fallos. Para establecer esta configuración, se puede realizar mediante software (propio o no del sistema operativo) o mediante hardware específico (tarjeta controladora o chipset de la placa base). La forma de realizar un nivel RAID es distribuyendo o redundando los datos entre varios discos de diferentes maneras.

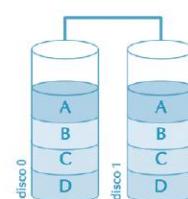
#### A) Tipos de RAID

Los niveles RAID más empleados son los siguientes:

- RAID 0. Se encarga de dividir o distribuir los datos entre dos o más discos sin duplicar la información, es decir, *no existe redundancia de datos*. Por este motivo, no se considera una configuración propia RAID, pero aumenta la velocidad de lectura y escritura.
- RAID 1. Establece una copia exacta entre dos o más discos. Esto permite aumentar la fiabilidad de los datos al quedar estos duplicados en tantos discos como se desee. Además, aumenta la velocidad de lectura.

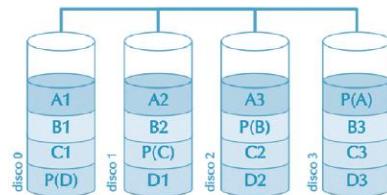


**Figura 3.24**  
Esquema RAID 0.



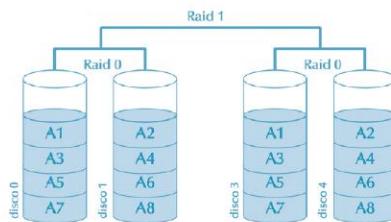
**Figura 3.25**  
Esquema RAID 1.

- RAID 5. Al igual que RAID 0, realiza una distribución de los bloques de datos y genera información de paridad que se distribuye en todos los discos (al menos tres). Los bloques de paridad permiten reconstruir un disco en caso de fallo. Para ello, han de realizar cálculos de los datos, generando dicha paridad, también llamada *código de detección de error* o *CRC*. De este modo, no se desaprovecha tanto espacio redundante, como RAID 1 y, además, mejora la velocidad de lectura, si bien las escrituras son más costosas al deber generar códigos CRC.

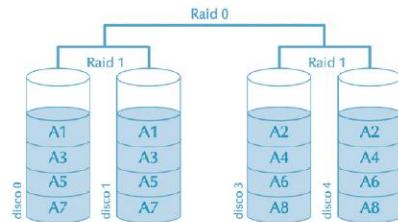


**Figura 3.26**  
Esquema RAID 5.

- Combinaciones RAID: RAID 1+0, RAID 0+1 y RAID 5+0. También se pueden establecer combinaciones de niveles RAID anidando estos y aprovechando las ventajas de varias configuraciones. Así, destacamos los siguientes niveles anidados:



**Figura 3.27**  
Esquema RAID 10.



**Figura 3.28**  
Esquema RAID 01.

SABÍAS QUE...  
Es frecuente emplear el término *JBOD* o *RAID Lineal* al método de combinar diferentes discos físicos en uno solo lógico. *JBOD*, por tanto, no presenta redundancia ni mejora el rendimiento del conjunto, sin embargo, el tamaño global es la suma de todos ellos.

## B) Administración de RAID

En Linux, la gestión y administración de RAID se realiza mediante el paquete *mdadm* (*Multiple Device Administrator*), el cual podemos instalar mediante: `sudo apt install mdadm`.

Antes de proceder, podemos comprobar si existe algún dispositivo RAID (multidispositivo) en el sistema, para lo que visualizamos el archivo `/proc/mdstat` mediante `cat /proc/mdstat`. Y en

su resultado podemos comprobar en la línea *Personalities* los tipos de niveles RAID soportados por el núcleo Linux actualmente.

La creación del RAID se puede hacer sobre los dispositivos o sobre particiones, y no necesariamente del mismo tamaño. En caso de que empleen diferente tamaño, *mdadm* avisará si este es más del 1% entre cualquiera de ellos y tomará el tamaño más pequeño. De cualquier modo, lo normal es emplear varios discos con la misma capacidad. Los comandos más comunes empleados para la gestión de RAID en Linux son los siguientes:

1. Creación de RAID.

```
mdadm --create /dev/mdX --level=Y --raid-devices=Z dispositivos
```

Donde:

- ✓ *create /dev/mdX* indica la creación del multidispositivo, siendo X un número.
- ✓ *level=Y* es el nivel RAID para aplicar, pudiendo ser Y:
  - *linear* para RAID lineal.
  - *raid0, 0 o stripe* para RAID0.
  - *mirror, raid1 o 1* para RAID1.
  - *raid4 o 4* para RAID4.
  - *raid5 o 5* para RAID5.
  - *raid6 o 6* para RAID6.
  - *raid10 o 10* para RAID10.
- ✓ *raid-devices=Z dispositivos*, donde Z indica el número de dispositivos asociados al RAID y cada uno de ellos separado por espacios (*/dev/sdX /dev/sdY ...*).

2. Establecer un disco como defectuoso de un RAID:

```
mdadm /dev/mdX --fail /dev/sdY
```

3. Eliminar un disco de un RAID:

```
mdadm /dev/mdX --remove /dev/sdY
```

4. Añadir un disco a un RAID:

```
mdadm /dev/mdX --add /dev/sdY
```

5. Comprobar el estado de todos los multidispositivos:

```
cat /proc/mdstat
```

6. Obtener información de todos los multidispositivos:

```
mdadm --detail --scan
```

7. Obtener información de un multidispositivo:

```
mdadm --detail /dev/mdX y mdadm --detail /dev/mdX --scan
```

8. Examinar el estado de un dispositivo asociado a un RAID:

```
mdadm --examine /dev/sdX
```

9. Detener un RAID:

```
mdadm --stop /dev/mdX
```

10. Eliminar el superbloque de un dispositivo sobreescribiendo ceros:

```
mdadm --zero-superblock /dev/sdY
```



### Actividad resuelta 3.6

#### Crear un RAID5

##### SOLUCIÓN

Antes de comenzar, hemos de disponer de al menos tres dispositivos y, en nuestro caso, hemos creado una partición que ocupa la totalidad de cada uno: `/dev/sdc1`, `/dev/sdd1` y `/dev/sde1`, todos ellos de 10 G.

A continuación, creamos el dispositivo RAID, indicando el nombre del multidispositivo (`/dev/mdX` donde X es un número que empieza en 0 y ha de estar libre), el número de dispositivos que lo constituirán y las unidades de almacenamiento:

```
sudo mdadm -- create /dev/md0 --level=5 --raid-devices=3 /dev/sdc1 /dev/sdd1 /dev/sde1
```

Podemos visualizar la información del RAID mediante: `mdadm -detail -scan`, y la construcción del mismo mediante `mdadm -detail /dev/md0` y `cat /proc/mdstat`. Con `mdadm` se informa del estado de construcción en la línea `Rebuild Status`. Una vez construido, todos los dispositivos aparecerán activos y en sincronía con el resto, como aparece en la siguiente imagen.

```
[luis@luis-VirtualBox:~]$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0] [raid1] [raid10]
md0 : active raid5 sde1[3] sdd1[1] sdc1[0]
      20951040 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
```

**Figura 3.29**  
Estado de los multidispositivos.

Una vez completada la construcción del RAID, si volvemos a analizar el estado de los multidispositivos del sistema, comprobamos la existencia de uno nuevo. Ahora debe aparecer `md0` activo con nivel RAID5 formado por los dispositivos `sde1`, `sdd1` y `sdc1`.

Y para poder usar `md0`, solo debemos implantar un sistema de archivos al multidispositivo y montarlo en un directorio existente con permisos suficientes para hacer uso de él:

```
sudo mkfs.ext4 /dev/md0
sudo mount /dev/md0 /media/Luis/RAID
```

**Actividad resuelta 3.7**

*Montar al inicio del sistema un multidispositivo.*

**SOLUCIÓN**

Al ser *md0* un dispositivo más, podemos hacer que permanezca ante reinicios. Los pasos serían los siguientes:

- Ubuntu modifica el nombre de los multidispositivos cuando reinicia el sistema. Para asignar un nombre multidispositivo a un grupo de discos en RAID, debemos añadir una línea en el archivo de configuración */etc/mdadm/mdadm.conf* ejecutando el script */usr/share/mdadm/mkconf* o añadiendo la línea directamente mediante *mdadm -detail /dev/md0 -- scan >> /etc/mdadm/mdadm.conf*, y después actualizamos el *initramfs* (sistema de archivos RAM de inicio de Linux) mediante el comando *update-initramfs -u*:

```
sudo /usr/share/mdadm/mkconf | sudo tee /etc/mdadm/mdadm.conf
sudo update-initramfs -u
```

- Como ya sabemos, debemos añadir una línea en */etc/fstab*, indicando que monte durante el inicio del sistema el multidispositivo *md0* (en este caso).

**Actividad resuelta 3.8**

*Simulación de fallos en RAID5*

**SOLUCIÓN**

Siguiendo con el ejemplo práctico anterior, vamos a simular fallos en discos de sistema RAID5 en el multidispositivo *md0*. Supongamos que el sistema, mediante algún software que genera avisos S.M.A.R.T., indica que el disco */dev/sdc* no funciona bien y que disponemos de otro disco con las mismas características que el resto del RAID.

Primero, marcamos como defectuoso el disco */dev/sdc* y comprobamos que su estado es "*[F]*" *faulty*:

```
sudo mdadm /dev/md0 --fail /dev/sdc1
cat /proc/mdstat
```

A continuación, quitamos el disco del RAID, restando aún dos discos más, y lo comprobamos:

```
sudo mdadm /dev/md0 --remove /dev/sdc1
cat /proc/mdstat
```

Por último, añadimos un nuevo disco */dev/sdg1*, reconstruyendo automáticamente el RAID5 y comprobamos:

```
sudo mdadm /dev/md0 --add /dev/sdg1  
cat /proc/mdstat
```

Si añadiésemos otro disco en RAID5 (sumando cuatro discos), sin estar ninguno de los existentes defectuoso, quedaría en modo repuesto (*spare*) ante futuros fallos de otro disco, es decir, el sistema lo sincronizaría automáticamente, si fallase uno de los otros. Esto se debe a que RAID5 emplea tres dispositivos, como mínimo.

Si deseamos eliminar un multidispositivo RAID y así evitar que aparezcan en modo inactivo o que algunos dispositivos no se puedan usar por estar asociados a otros RAID no activos, debemos:

1. Desmontar el dispositivo si está en uso.
2. Detener el multidispositivo RAID.

```
sudo mdadm --stop /dev/md0
```

3. Borrar el superbloque de cada dispositivo que constituía el RAID:

```
sudo mdadm --zero-superblock /dev/sde1  
sudo mdadm --zero-superblock /dev/sdd1  
sudo mdadm --zero-superblock /dev/sdg1
```

4. En caso de que estuviera asociado al arranque del sistema, actualizar */etc/fstab* eliminando la línea asociada y actualizar *initramfs*.

### 3.7. Gestión de almacenamiento por interfaz gráfica en Microsoft Windows

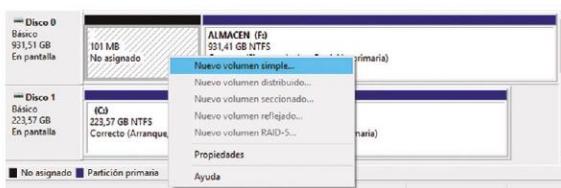
En Microsoft Windows la gestión del almacenamiento se realiza principalmente mediante el 'Administrador de discos'. Se puede acceder a través del 'Panel de control', 'Herramientas administrativas' y 'Administración de equipos'.

En él figuran detalladamente cada uno de los discos instalados en el sistema y sus particiones. De tal manera que podemos estudiar el tamaño total, capacidad, espacio libre y usado. Para cada disco o partición podemos realizar una serie de acciones asociadas: crear particiones, formatear, eliminar particiones, cambiar las etiquetas o letras de la unidad, etc.

Otra utilidad de Microsoft Windows es 'Almacenamiento'. Se puede acceder dentro de Sistema en 'Configuración'. En ella, podemos ver el espacio libre de cada unidad y, además, al acceder a cada una de ellas, se estudia el 'Uso de almacenamiento', donde el sistema realiza un análisis de los tipos de archivos ordenados por el espacio que ocupan. A su vez, por cada tipo, resultan ordenados por tamaño. También se puede acceder a cada archivo de los que lo conforman o realizar sobre ellos actividades propias de cada tipo, como, por ejemplo: desinstalar una aplicación, administrar la restauración del sistema, acceder a un archivo en concreto o administrar las carpetas de descargas.

Además, el 'Explorador de archivos' de Microsoft Windows facilita la gestión de los archivos sin necesidad de conocer la gestión interna del sistema. Desde este, es muy sencillo realizar diferentes acciones sobre las unidades, como formatear, desfragmentar o chequear unidades o particiones.

Para particionar un disco en Microsoft Windows se realiza desde el ‘Administrador de discos’ y seleccionando el espacio no particionado del disco con el botón secundario del ratón se marca ‘Nuevo volumen simple…’.



**Figura 3.30**  
Creación de partición  
en Microsoft Windows.

Sucesivamente, nos solicitará que introduzcamos el tamaño de la partición, la letra para asignarla a dicha unidad y si deseamos formatearla, indicaremos el tipo de sistema de archivos, el tamaño del clúster y la etiqueta identificativa.

Se puede dar la circunstancia, por diversos motivos, que Microsoft Windows no reconozca una unidad o una partición y, por tanto, el ‘Explorador de archivos’ no la mostrará. En tales casos, desde el ‘Administrador de discos’ debemos inicializar el disco (situándonos sobre el recuadro del disco e ‘Inicializar disco’) o asignarle una letra a una unidad pulsando en ‘Cambiar la letra y rutas de acceso de unidad’.

Estando creada una partición, se puede formatear desde el ‘Administrador de discos’ o el propio ‘Explorador de archivos’ mediante el menú contextual de la unidad.

Y desde la opción ‘Propiedades’ (pestaña ‘Herramientas’) del menú contextual del ‘Explorador de archivos’ sobre una unidad, podemos chequearla y desfragmentarla.



#### TOMA NOTA

Todas estas opciones también están presentes en la cinta de opciones del ‘Explorador de archivos’, en la pestaña ‘Herramientas de unidad’.

Por otro lado, con Microsoft Windows, podemos hacer uso del ‘Administrador de discos’ para añadir a los discos otras características y funcionalidades, convirtiéndolos en ‘discos dinámicos’. Con este concepto, Microsoft Windows permite crear distintos tipos de volúmenes, y algunos de ellos de tipo RAID, a saber:

- Volumen distribuido: consiste en unir diferentes espacios de diferentes discos bajo una misma unidad lógica.
- Volumen reflejado: conocido también como RAID-1. Genera redundancia de datos.
- Volumen seccionado: también llamado RAID-0. Genera un gran rendimiento, pero un fallo en cualquiera de sus discos hará que el volumen falle.
- Volumen RAID-5: permite distribuir los datos entre tres o más discos con tolerancia a fallos de forma distribuida. Es accesible desde versiones de Windows Server.
- Volumen simple: una única unidad en un disco.

Hemos de tener en consideración que la conversión de discos básicos a dinámicos no implica la pérdida de datos, sin embargo, de dinámicos a básicos sí. El procedimiento es muy sencillo: desde el 'Administrador de discos' hemos de convertir los discos a dinámicos, si estos no lo son, y han de estar inicializados.

Pulsando con el botón secundario del ratón sobre uno de los discos, nos dará a elegir entre las distintas posibilidades de discos dinámicos (puede ser que alguna de ellas aparezca sombreada, debido a que el sistema detecta imposibilidad en la acción). Posteriormente, comenzará el asistente para la creación de un nuevo volumen del tipo seleccionado, resultando un proceso muy sencillo e intuitivo en el que solo tenemos que seguir los pasos.

### Actividad resuelta 3.9

*Crear un volumen reflejado en Microsoft Windows 10 desde el 'Administrador de discos'.*

#### SOLUCIÓN

Suponemos que hemos instalado tres discos con espacio no asignado, tal y como aparece en la siguiente imagen. Inicializamos los discos y los convertimos a dinámicos.

Seleccionamos la opción 'Nuevo volumen reflejado' y se desplegará el asistente de creación en el que seguimos los pasos empleando todo el espacio de cada disco (se puede emplear menos), seleccionamos una letra a la unidad, implantamos el sistema de archivos NTFS e indicamos una etiqueta a la unidad (en nuestro caso, RAID1).

Una vez terminada la configuración con el asistente, aparecerá un resumen de las características del nuevo volumen y, tras pulsar en 'Finalizar', el sistema creará el nuevo volumen, que se reflejará como una unidad más.

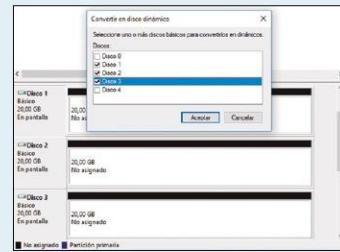


Figura 3.31  
Creación de discos dinámicos.

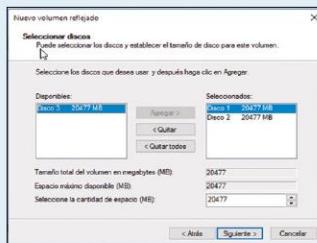


Figura 3.32  
Selección de discos en el asistente para nuevo volumen reflejado.

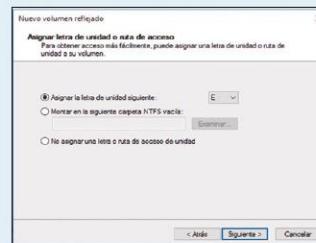
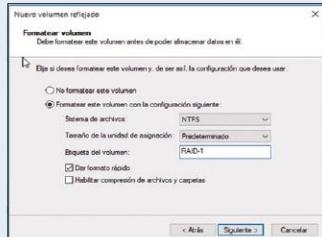
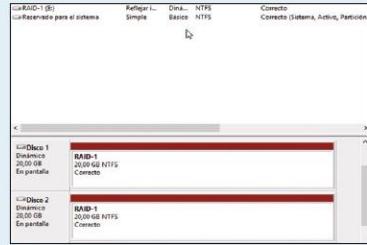


Figura 3.33  
Asignación de letra en el asistente para nuevo volumen reflejado.



**Figura 3.34**  
Selección de la configuración para formatear el nuevo volumen reflejado.

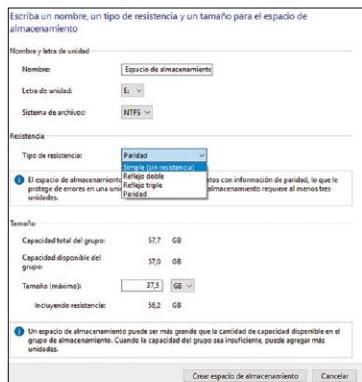


**Figura 3.35**  
Nuevo volumen reflejado.

Por otro lado, Microsoft Windows dispone de otra herramienta llamada ‘Espacios de almacenamiento’, a la que podemos acceder a través de ‘Almacenamiento’ y ‘Administrar espacios de almacenamiento’, o mediante el ‘Panel de control’.

‘Espacios de almacenamiento’ permite crear unidades virtuales agrupando dos o más unidades en un grupo de almacenamiento. Esto facilita, sin apenas conocimientos y mediante una gestión muy sencilla, la administración de espacios simples, similar a RAID0 (ya que no protegen los datos, pero aumentan el rendimiento), espacios de reflejo (doble o triple), similar a RAID1, y espacios de paridad, similar a RAID5.

Dentro de ‘Espacios de almacenamiento’, su creación es muy sencilla. Basta con acceder a ‘Crear un nuevo grupo y espacios de almacenamiento’. Posteriormente, seleccionamos la etiqueta, la unidad, el sistema de archivos para implantar, el tipo de espacio de almacenamiento y su tamaño.



**Figura 3.36**  
Creación de un espacio de almacenamiento.



**Figura 3.37**  
Administración de un espacio de almacenamiento.

Una vez creado, se puede administrar el espacio de almacenamiento agregando unidades, cambiando el nombre del grupo, etc.

En estas condiciones, el sistema avisa cuando una unidad física se encuentra defectuosa. En ese caso, se debe agregar una nueva unidad física para solventar el problema.

### 3.8. Búsqueda de información por línea de comandos en Linux

Los sistemas de archivos han de ofrecer herramientas para localizar archivos por diferentes criterios (fecha de creación, fecha de modificación, tamaño, nombre, etc.) y búsqueda de información en el contenido de estos.

Uno de los comandos más empleados en Linux es sin duda *find*. Este comando permite multitud de opciones para localizar cualquier fichero en el sistema de archivos. Su sintaxis es la siguiente:

```
find [ruta] [criterio] [acción]
```

El comando *find* realiza una búsqueda sobre la ruta dada (pueden ser varias). Si no se especifica ruta, la búsqueda se realiza sobre el directorio actual. El resultado y los errores se envían a las salidas estándares por defecto. En caso de no indicar criterio alguno, no se hará ningún tipo de filtro. Además, la acción permite tratar los ficheros encontrados mediante alguna acción.

#### 3.8.1. Criterios de búsqueda

Los criterios más importantes de búsqueda son los que se exponen a continuación:

##### A) Por nombre de fichero

Están las siguientes opciones:

- ✓ *name patrón*: permite buscar ficheros permitiendo la búsqueda con patrones.
- ✓ *iname patrón*: actúa del mismo modo que *name*, pero sin distinguir mayúsculas de minúsculas.

Ejemplos:

- *find . -name texto.txt*. Busca en el directorio actual recursivamente archivos con el patrón “texto.txt”.
- *find . -name 't\*.txt'*. Busca en el directorio actual recursivamente archivos que comiencen por “t” y terminen en “.txt”.
- *find . -iname texto.txt*. Busca en el directorio actual recursivamente archivos con el patrón “texto.txt” sin distinguir mayúsculas de minúsculas.

### B) Nivel de profundidad en subdirectorios

Por defecto, *find* busca recursivamente en directorios a partir de la ruta especificada, pero podemos limitar el nivel de profundidad máximo (hasta donde llega la búsqueda) y mínimo (desde dónde empieza la búsqueda).

- ✓ *maxdepth n*: especifica hasta qué subdirectorios se realiza la búsqueda. La ruta especificada se encuentra en el nivel 1, un subdirectorio dentro de este en el nivel 2 y así sucesivamente.
- ✓ *mindepth n*: se especifica desde qué nivel comienza la búsqueda. Si se indica el nivel 2, buscará desde los subdirectorios de la ruta especificada recursivamente.

Ejemplos:

- *find . -maxdepth 1 -name texto.txt*. Limita la búsqueda de ficheros con nombre “texto.txt” en el directorio actual sin entrar en subdirectorios.
- *find . -maxdepth 2 -name texto.txt*. Limita la búsqueda de ficheros con nombre “texto.txt” al directorio actual y subdirectorios.
- *find . -mindepth 2 -name texto.txt*. Comienza la búsqueda de ficheros con nombre “texto.txt” desde los subdirectorios del directorio actual.
- *find . -maxdepth 2 -mindepth 2 -name texto.txt*. Limita la búsqueda de ficheros con nombre “texto.txt” únicamente a subdirectorios del directorio actual.

### C) Tiempos de acceso, modificación y cambio

Podemos hacer búsquedas atendiendo a la fecha de última modificación del *i-nodo* (*c*), última modificación de su contenido (*m*) y último acceso a su contenido (*a*). Para cualquiera de ellos, se puede especificar el tiempo en minutos (min) o en días (time). De tal manera que podemos especificar las siguientes opciones: *cmin*, *mmin*, *amin*, *ctime*, *mtime* y *atime*. Los valores numéricos que acompañan a las opciones pueden ser: *+n* indicando mayor que, *-n* indicando menor que y *n* indicando igualdad.

Ejemplos:

- ✓ *find . -amin -1*. Localiza archivos que se accedieron hace menos de un minuto.
- ✓ *find . -mtime -1*. Localiza archivos que se modificaron hace menos de un día.

### D) Comparación de ficheros

También se pueden localizar ficheros comparándolos con otro fichero. A saber:

- *newer fichero*: busca ficheros que se modificaron más recientemente que *fichero*.
- *anewer fichero*: busca ficheros accedidos más recientemente que *fichero* fue modificado.
- *cnewer fichero*: busca ficheros en los que el estado del *i-nodo* se modificó más recientemente que *fichero* fue modificado.

Ejemplo:

*find . -anewer notas.txt*. Busca ficheros cuya fecha de acceso fue más reciente que la modificación de *notas.txt*.

### E) Tamaños

Podemos especificar comparaciones con tamaños. Para lo que podemos emplear la siguiente opción: `-size [+|-]n[bckMG]`. Donde cada parámetro indica:

- ✓ +n indica mayor que n, -n indica menor que n y n indica igualdad, siendo n un valor numérico.
- ✓ b bloques de 512 bytes.
- ✓ c bytes.
- ✓ k kilobytes.
- ✓ M megabytes.
- ✓ G gigabytes.

Ejemplo:

`find . -size +1M`. Busca archivos mayores de 1 megabyte.

### F) Tipo de fichero

Se pueden realizar búsquedas por el tipo de fichero mediante la opción `-type` con alguno de los siguientes modificadores: l (enlace simbólico), d (directorio), f (fichero regular), b (dispositivo de tipo bloque) y c (dispositivo de tipo carácter).

Ejemplo:

`find ./nivel2 -type d`. Busca directorios a partir del subdirectorio “nivel2”.

### G) Permisos

Las búsquedas se pueden efectuar sobre los permisos mediante la opción `perm [-/] permisos`. Se pueden establecer los permisos en octal o de manera simbólica. El signo “-” indica que el fichero debe contener al menos los permisos dados y “/” indica que debe tener alguno de los permisos que se dan. Si no se indica ninguno de estos dos signos, los permisos deben de ser idénticos a los especificados.

Ejemplos:

La figura 3.38 muestra varios ejemplos en los que se puede observar cómo para los permisos dados en el subdirectorio `nivel2`, los archivos encontrados con la opción `-perm 644` son aquellos que tienen exactamente esos permisos (`rw-r- - r--`). Con `-perm -644` se suman los directorios `nivel2` y `nivel3`, puesto que disponen de los permisos anteriores más dos de ejecución (`rwxr-xr-x`). Y `-perm /644` son los mismos que los anteriores, puesto que todos ellos tienen al menos `rw-` en usuario, `r` en grupo o `r` en otros.

```
luis@luis-VirtualBox:~$ ls -l ./nivel2
total 4
drwxr-xr-x 2 luis luis 4096 dic 28 10:22 nivel3
-rw-r--r-- 1 luis luis 0 dic 28 10:22 texto.txt
-rw-r--r-- 1 luis luis 0 dic 28 10:22 Texto.txt
luis@luis-VirtualBox:~$ find ./nivel2 -perm 644
./nivel2/nivel3/texto.txt
./nivel2/nivel3/Texto.txt
./nivel2/texto.txt
./nivel2/Texto.txt
luis@luis-VirtualBox:~$ find ./nivel2 -perm -644
./nivel2
./nivel2/nivel3
./nivel2/nivel3/texto.txt
./nivel2/nivel3/Texto.txt
./nivel2/texto.txt
./nivel2/Texto.txt
luis@luis-VirtualBox:~$ find ./nivel2 -perm /644
./nivel2
./nivel2/nivel3
./nivel2/nivel3/texto.txt
./nivel2/nivel3/Texto.txt
./nivel2/texto.txt
./nivel2/Texto.txt
```

**Figura 3.38**  
Uso de `find` con la opción `-perm`.

### H) Otras opciones de búsqueda

- user usuario. Localiza archivos por un usuario dado.
- inum inodo. Busca ficheros por número de inodo.
- uid UID. Busca ficheros por el UID de usuario.
- gid GID. Busca ficheros por GID.

También podemos combinar opciones de búsqueda mediante las siguientes opciones:

- ✓ criterio1 *–and* criterio2: hace que se cumplan dos criterios. Es similar a no indicar nada entre criterios.
- ✓ criterio1 *–a* criterio2: idéntico al anterior.
- ✓ criterio1 *–or* criterio2: hace que se cumpla un criterio u otro.
- ✓ criterio1 *–o* criterio2: idéntico al anterior.
- ✓ *-not* criterio: niega el criterio.
- ✓ *!criterio*: idéntico al anterior.

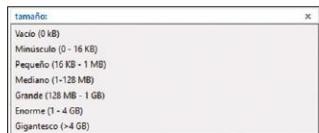
Además, se puede indicar mediante paréntesis la preferencia entre criterios.

Ejemplos:

- *find ./nivel2 –perm /644 –type d*. Los criterios *–perm /644* y *–type d* se han de cumplir a la vez.
- *find ./nivel2 –perm /644 –o –perm 755*. Los criterios para obtener un resultado pueden ser uno u otro.
- *find . –size +1M –a \ -user root –o –user luis \*. El criterio *–size +1M* es obligatorio y que se dé alguno de los otros dos criterios: que el usuario sea *root* o *luis*. Como se puede observar, los paréntesis se deben “escapar” con el carácter “\”.

## 3.9. Búsqueda de información por interfaz gráfica en Microsoft Windows

El cuadro de búsqueda del ‘Explorador de archivos’ es una herramienta muy potente para realizar búsquedas desde la unidad o carpeta actual por diferentes criterios para archivos o carpetas: fecha, modificación, creación, tamaño (vacío, minúsculo, pequeño, mediano, grande, enorme, gigantesco o tamaño exacto), clase (carpeta, vínculo, película, imagen, etc.), extensión, carpeta o archivo. Además, se pueden combinar diferentes criterios mediante operadores lógicos: NOT, OR o &. La cinta de opciones de ‘Herramientas de búsqueda’ se habilita al situarse sobre el cuadro de búsqueda.



**Figura 3.39**  
Cuadro de búsqueda por tamaño.



**Figura 3.40**  
Cinta de opciones de búsqueda.



### Actividades propuestas

**3.13.** Realiza búsquedas con *find* en Linux, atendiendo a las diferentes opciones que hemos estudiado.

**3.14.** Realiza búsquedas a través del 'Explorador de archivos' de Microsoft Windows mediante el cuadro de búsqueda y la cinta de opciones de 'Herramientas de búsqueda'.



### Recurso digital 3.2

Administrador de medios virtuales de Oracle VM VirutalBox.

### Resumen

- Los sistemas operativos modernos ofrecen herramientas para la gestión de archivos y almacenamiento. En este capítulo hemos trabajado tanto desde el punto de vista gráfico (con Microsoft Windows) como textual (con Ubuntu), aunque se ha hecho hincapié en este último, dada su gran potencia y versatilidad.
- Primero se ha estudiado el concepto de sistema de archivos, así como los ejemplos más característicos: FAT, exFAT, NTFS, ext4 y APFS. Más tarde se ha dado a conocer la estructura de directorios de Linux y Microsoft Windows.
- Los comandos más importantes tratados para la gestión de archivos en Ubuntu han sido:

ls	cd	touch	pwd	ln	rm	mkdir	rmdir
cp	mv	cat	more	less	head	tail	wc
sort	cut	grep					

- Por otro lado, hemos estudiado la gestión de almacenamiento mediante los comandos:

mount	lsblk	fdisk	lsusb
umount	gdisk	parted	GParted
mkfs	e4defrag	fsck	e2fsck
mdadm			

- Además, se han estudiado archivos o directorios importantes, tanto en la gestión de almacenamiento como en la configuración general del sistema Ubuntu, como:

/etc/passwd	/dev/null	/dev	/proc/self/mounts
/etc/fstab	/proc/mdstat		

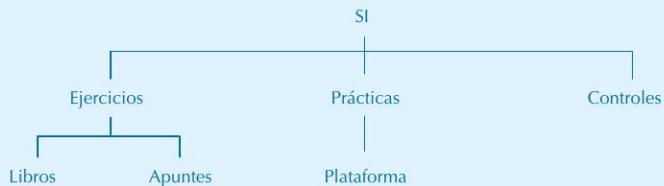
- Desde el punto de vista de la administración gráfica con Microsoft Windows, para la gestión de archivos y almacenamiento hemos trabajado con el 'Explorador de archivos' y el 'Administrador de discos', respectivamente. No obstante, siempre es posible una administración por línea de comandos en Microsoft Windows, aunque no se haya estudiado en este capítulo.

- Por último, se han abordado herramientas para localizar archivos por diferentes criterios, así como búsqueda de información dentro de estos, como *find* en Ubuntu y a través las herramientas de búsqueda del ‘Explorador de archivos’ en Microsoft Windows.



## Ejercicios propuestos

1. ¿Cuáles son los objetivos de los sistemas de archivos?
2. Gestión de archivos en Ubuntu. En Ubuntu, crea la siguiente estructura de directorios a partir del directorio home del usuario:



- Copia el archivo */etc/passwd* en el directorio *Plataforma* (estando situado en el directorio */etc*).
- Copia los archivos que contengan la letra *c* del directorio */bin* al directorio *Ejercicios*.
- Copia todos los archivos que empiecen por “m” o por “n” del directorio */bin* al directorio *Prácticas*.
- Mueve un solo archivo que empiece por la letra “m” del directorio *Prácticas* al directorio *Libros*.
- Borra el archivo *mkdir* con confirmación del directorio *Prácticas*.
- Renombra el archivo *mount* de *Prácticas* por *montar*.
- Crear un enlace simbólico de *montar* llamado *e\_montar*. Crea dos enlaces duros de correo llamados *montar\_duro1* y *montar\_duro2*. ¿Cuántos enlaces duros contiene *montar*? Compara los números de i-node de *montar*, *montar\_duro1* y *montar\_duro2* y justificalo. Borra *montar*. ¿Se ha convertido *e\_montar* en un enlace roto? ¿Por qué? ¿Y si eliminamos *montar\_duro1* y *montar\_duro2*?
- Copia toda la información que contiene el directorio *Prácticas* al directorio *Apuntes*.
- Crear un archivo de texto de dos líneas con el comando *cat* en el directorio *SI*. Muestra el número de palabras y líneas de dicho archivo de texto.

3. Procesamiento de textos en Ubuntu:

- Concatena los archivos */etc/passwd*, */etc/shadow* y */etc/fstab* en un solo archivo llamado *concatenado*. Todo ello en una sola instrucción.
- Muestra el número de usuarios que disponen del shell “bash” como intérprete de comandos. Emplea el fichero */etc/passwd*.
- Listar de manera inversamente ordenada solo los grupos primarios de aquellos usuarios cuyo UID comienza por 1. Emplea el fichero */etc/passwd*.

**4. Búsqueda de información en Ubuntu:**

- a) Encuentra los archivos ocultos de tu directorio de trabajo.
- b) Busca en todo el sistema los ficheros de tu usuario. Evita mostrar los mensajes de error.
- c) Busca todos los archivos que comiencen por "a" en múltiples rutas de forma conjunta: en tu *home* y en */dev*.
- d) Busca todos los archivos que comiencen por "ca" pero que no terminen con ".php". Evita mostrar los mensajes de error.
- e) Encuentra todos los archivos modificados en la última hora. Evita mostrar los mensajes de error.
- f) Busca todos los ficheros que tengan como usuario vuestro usuario que empiecen por "e" y que tenga más de 1K. Evita mostrar los mensajes de error.
- g) Busca los ficheros en */etc* que tengan permiso de lectura sin entrar en subdirectorios. Evita mostrar los mensajes de error.
- h) Crea un fichero llamado *fichero1*. Después, crea 2 ficheros llamados *fichero2* y *fichero3*. Encuentra aquellos ficheros que se hayan creados posteriormente a *fichero1*.
- i) Modifica *fichero2*, *fichero3* y *fichero1* por ese orden, con el contenido que deseas. Busca los ficheros que se hayan modificado más recientemente a la modificación de *fichero2*.
- j) Entra en *fichero3*. Sal. Busca los ficheros cuyo acceso sea más reciente.

**5. Búsqueda de información en Microsoft Windows:**

- a) Busca en todo el equipo aquellas imágenes entre 1 y 128 MB creadas el mes pasado.
- b) Busca en el directorio actual aquellos archivos con extensión *.txt* y creados hoy.
- c) Busca en todas las subcarpetas aquellos directorios que tengan como nombre *datos* o *copia*.

**6. Particiones y volúmenes en Ubuntu.** Para el siguiente ejercicio, a partir de una máquina virtual con Ubuntu, añade un nuevo disco duro de 30 GB.

- a) Describe los pasos y comandos para crear y poder utilizar un disco duro con la siguiente estructura GPT y tamaño de sus particiones:



Donde cada partición debe ser montada a partir del directorio para crear \$HOME/particiones.

- b) Hacer que la partición *Datos* y *Backup* se monte automáticamente al iniciarse el sistema en modo de solo lectura.

- c) Chequea la partición *Backup* y desfragménatala.
- d) Disponemos de un archivo *helloween.mp3* en el directorio *musica* de nuestro pendrive con sistema de archivos NTFS que está sin montar y queremos copiarlo al directorio *documentos* de nuestro sistema dentro de nuestro *home*. Al finalizar, hay que desmontarlo. Indica todas las acciones para realizar todo el proceso.
- 7. Gestión de almacenamiento en Microsoft Windows.** A partir de una máquina virtual con Microsoft Windows, añade un nuevo disco duro de 30 GB. A través del 'Administrador de discos', crea tres particiones de 10 GB cada una.
- 8. Gestión de archivos en Microsoft Windows:**
- A través del 'Explorador de archivos' y continuando el ejercicio anterior, formatea una de las particiones de la unidad con sistema de archivos NTFS, etiquétala con nombre *Datos* y asigna un tamaño de la unidad de asignación de 4096.
  - Formatea otra de las particiones de la unidad con sistema de archivos *FAT32* y etiquétala como *Compartida*.
- 9. RAID en Ubuntu.** Añade a la máquina virtual de Ubuntu tres discos de 20 GB cada uno:
- Crea un sistema RAID1 con dos de ellos.
  - Haz que dicho RAID1 sea permanente ante reinicios.
  - Simula el fallo de uno de los discos, márcalo como defectuoso, elimínalo del RAID1 y asocia otro disco para que se sincronice al RAID1.
- 10. 'Discos dinámicos y espacios de almacenamiento' en Microsoft Windows.** Añade a la máquina virtual de Microsoft Windows tres discos de 20 GB cada uno:
- Crea un sistema RAID1 con dos de ellos a través del 'Administrador de discos'.
  - Comprueba la nueva unidad a través del 'Explorador de archivos'. Deshaz el RAID.
  - Crea un espacio de reflejo con los tres discos mediante 'Espacios de almacenamiento'.

### ACTIVIDADES DE AUTOEVALUACIÓN

1. El sistema de archivos exFAT:

- a) Es más ligero que NTFS y APFS.
- b) Es más seguro que NTFS.
- c) Solo permite gestionar archivos de hasta 4 GB.

2. La eliminación definitiva de un fichero en *ext4* se produce cuando:
- a) El fichero es suprimido de la papelera de reciclaje.
  - b) Su número de enlaces duros es 0.
  - c) Su número de enlaces simbólicos es 0.
3. En Ubuntu, el directorio */etc* almacena:
- a) Los archivos binarios (ejecutables) a nivel de usuario.
  - b) Archivos como librerías y módulos del kernel.
  - c) Archivos de configuración globales del sistema que afectan a todos los usuarios.
4. En Microsoft Windows, ¿qué carpeta contiene el perfil base sobre el que se crean nuevos perfiles?:
- a) C:\Usuarios\Acceso público.
  - b) C:\Usuarios\Default.
  - c) C:\Usuarios\System32.
5. En Ubuntu, un archivo con máscara de permisos *lrw-rw-rw-*:
- a) Es un archivo regular.
  - b) Es un directorio.
  - c) Es un enlace simbólico.
6. En Ubuntu, ¿cuál de las siguientes denominaciones hace referencia a la tercera partición primaria del segundo disco duro tipo SATA de nuestro ordenador?:
- a) sdc.
  - b) hdb3.
  - c) sdb3.
7. En Ubuntu, ¿a través de qué archivo podemos visualizar los sistemas de archivos que están accesibles en nuestro sistema?:
- a) /proc/self/mounts.
  - b) /dev/ttyS1.
  - c) /etc/passwd.
8. Cuando se ejecuta en un terminal *cat /etc/passwd | grep ^luis;* en Ubuntu, obtendremos:
- a) Aquellas columnas del fichero /etc/passwd que comienzan por "luis:".
  - b) Aquellas líneas del fichero /etc/passwd que comienzan por "luis:".
  - c) Aquellas líneas del fichero /etc/passwd que finalicen por "luis:".
9. En Ubuntu, cuando se ejecuta en un terminal *sort -t: -k1 -r /etc/passwd;*
- a) Ordena inversamente el fichero /etc/passwd por login, estableciendo delimitador ":".
  - b) Ordena el fichero /etc/passwd por login, estableciendo delimitador ":".
  - c) Ordena el fichero k1 por nombre de usuario, estableciendo delimitador ":".
10. Microsoft Windows permite crear distintos tipos de volúmenes y algunos de ellos de tipo RAID, como un volumen seccionado que equivale a:
- a) RAID 1.
  - b) RAID 5.
  - c) RAID 0.

**SOLUCIONES:**1.  a  b  c2.  a  b  c3.  a  b  c4.  a  b  c5.  a  b  c6.  a  b  c7.  a  b  c8.  a  b  c9.  a  b  c10.  a  b  c

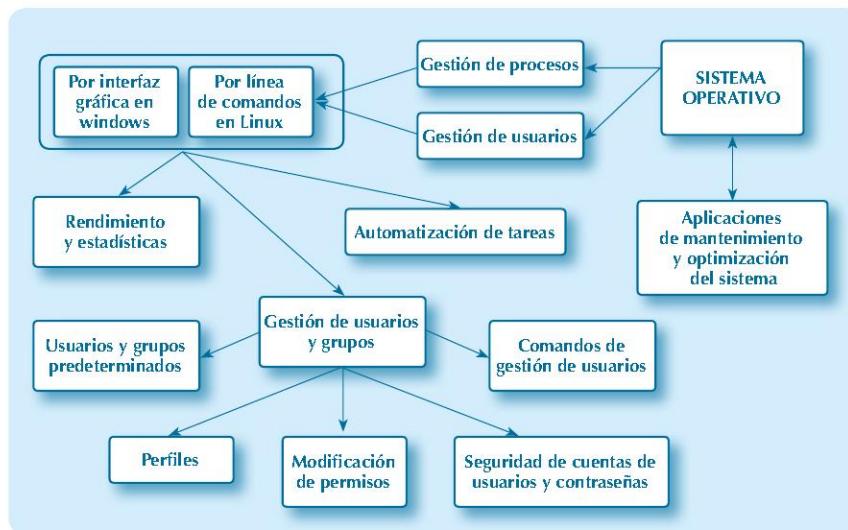
# 4

## Sistemas operativos. Gestión de usuarios y procesos

### Objetivos

- ✓ Descubrir los fundamentos de gestión de usuarios y gestión de procesos.
- ✓ Crear cuentas de usuario locales y grupos.
- ✓ Asegurar el acceso al sistema mediante directivas de cuenta y de contraseñas.
- ✓ Proteger el acceso a la información mediante permisos locales.
- ✓ Conocer diferentes mecanismos para la gestión de procesos.
- ✓ Emplear comandos para realizar tareas básicas de configuración y monitorización del sistema.
- ✓ Conocer diferentes herramientas para el mantenimiento y optimización del sistema.
- ✓ Saber operar con software de automatización de tareas.

### Mapa conceptual



### Glosario

- Background.** Ejecución de procesos en segundo plano.
- Cuantum.** Espacio de tiempo asignado a cada proceso para ocupar la CPU.
- GID.** Número de identificación de grupo único en sistemas Linux.
- Linux-PAM.** Sistema centralizado de autenticación de usuarios en Linux.
- Login.** Nombre que se emplea para acceder al sistema operativo.
- Máscara de permisos.** Privilegios de los que dispone el propietario, el grupo y el resto de usuarios sobre un objeto del sistema de archivos en sistemas Linux.
- Modo kernel de ejecución.** Capacidad de ejecución en modo privilegiado sobre el sistema operativo.
- PCB.** Bloque de control de proceso.
- PID.** Identificador de proceso.
- Proceso.** Instancia de un programa en ejecución.
- Prompt.** Línea de petición de órdenes en el intérprete de comandos.
- Shell.** Intérprete de comandos que actúa de interfaz entre el sistema operativo y los usuarios.
- Superusuario.** Usuario con mayor privilegio sobre un sistema Linux.
- UID.** Número de identificación de usuario único en sistemas Linux.

## 4.1. Introducción

Todo sistema operativo ha de ofrecer el soporte necesario para que los usuarios puedan operar en un ambiente amigable, interactuando con el sistema y las aplicaciones instaladas. Esto ha de realizarse en un ambiente seguro, donde el sistema operativo ofrezca herramientas para gestionar usuarios a diferentes niveles: usuario común y usuario administrador.

El usuario debe conocer algunas características propias del sistema operativo donde opera con objeto de facilitar su desempeño en el mismo. Ello incluye aspectos de configuración de su cuenta de usuario, gestión de sus propios procesos, administración de dispositivos, etc.

Las labores de los administradores del sistema informático van más allá de las de los usuarios comunes, ya que su conocimiento sobre el sistema ha de ser mayor, así como el manejo de las herramientas que brinda el sistema operativo para su administración.

Por otro lado, la gestión de procesos es el eje vertebral del sistema operativo, es decir, por él pasan el conjunto de las acciones que realiza todo el software (ya sea del propio sistema operativo o no) y, gracias a él, se dan la mano el resto de secciones del sistema operativo: gestión de usuarios, sistema de archivos, gestión de memoria, seguridad, etc.

El sistema operativo gestiona la ejecución de los procesos, de tal manera que los tiempos de ejecución de las diferentes tareas sigan los objetivos del propio sistema operativo bajo la supervisión de los administradores, respetando la eficiencia y la seguridad del sistema.

Los usuarios administradores se encargan de la supervisión y gestión de los procesos del sistema, intentando orientar el rendimiento del mismo hacia una política equitativa o justificada en beneficio de procesos por lotes o interactivos. Además, la gestión de usuarios por parte de los administradores determina gran parte de la seguridad del sistema.

En este capítulo abordaremos tanto la gestión de usuarios como la gestión de procesos, trabajando con los sistemas operativos Ubuntu y Microsoft Windows. Trabajaremos con diferentes herramientas propias de estos sistemas operativos para automatizar tareas y monitorizar el sistema.

Por último, conoceremos diferentes tipos de aplicaciones que ayudan a un mantenimiento adecuado del sistema, así como a optimizarlo, mejorando así su eficiencia, tiempos de respuesta, accesos a recursos, rendimiento del hardware y la productividad de los medios de almacenamiento.

A lo largo del capítulo se enfatiza en la necesidad de adquirir destrezas de interacción por línea de comandos, ya que su uso y manejabilidad es un objetivo para conseguir como usuario o administrador de un sistema operativo. Las herramientas de gestión propias de los sistemas operativos mediante comandos, sobre los que tenemos que conocer su sintaxis, deben ser manejadas por los usuarios, siempre con apoyo de los propios comandos. No obstante, no podemos olvidar las herramientas gráficas para la administración del sistema, más extendidas en su uso por los usuarios comunes, pero que no dejan de ser menos importantes.

## 4.2. Gestión de usuarios por línea de comandos en Linux

Los sistemas GNU/Linux gestionan los usuarios mediante archivos de configuración. Sobre estos archivos, los usuarios comunes no gozan de privilegios, por lo que son los administradores o el usuario *root* los únicos que pueden editarlos. En algunos casos, la edición de los archivos de configuración puede ser directa y, en otros, según el archivo que se va a tratar, es recomendable emplear comandos concretos para evitar errores sintácticos o de formato.

#### 4.2.1. Configuración de usuarios y grupos

Los usuarios y grupos en Linux se gestionan a través de los archivos `/etc/passwd` y `/etc/group`, principalmente, además de otros muchos ficheros, como `/etc/sudoers`, `/etc/shadow`, etc.

Debemos conocer la estructura de todos estos archivos, así como la forma de modificar su contenido para gestionar los usuarios.

Como ya sabemos, `/etc/passwd` almacena las cuentas de los usuarios del sistema. Cada fila se corresponde con un usuario y consta de siete campos delimitados por “:”, en el siguiente orden:

1. *Login* de usuario o nombre que se emplea para acceder al sistema.
2. *Passwrd* requerida para autenticarse en el sistema. Aparece una “x” indicando que la contraseña se encuentra encriptada en el fichero de configuración `/etc/shadow`.
3. *UID* o número de identificación de usuario único (User IDentification). El 0 se corresponde con el *superusuario*; del 1 al 99 para cuentas predeterminadas y de la 100 a 999 para cuentas administrativas del sistema, por tanto, los nuevos usuarios serán asociados a partir del 1000.
4. *GID* o número de identificación del grupo principal del usuario (Group IDentification).
5. *Información personal del usuario*, donde suelen incluirse datos de localización del usuario, como teléfono, oficina, etc.
6. *Home* o *directorio de trabajo*, es decir, el directorio inicial del usuario, cuando se conecta al sistema..
7. *Shell* o intérprete de comandos empleado por el usuario cuando inicia el sistema.

<code>slice:</code>	<code>x:</code>	<code>1002:</code>	<code>1002:</code>	<code>Usuario Slice,,,: </code>	<code>/home/slice:</code>	<code>/bin/bash</code>
					Carpet personal	Shell

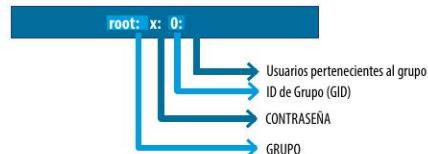
Información del usuario  
ID de grupo (GID)  
ID de usuario (UID)  
Contraseña  
Nombre de usuario

**Figura 4.1**  
Estructura del fichero `/etc/passwd`.

Por usuario administrador en Linux entendemos aquel que tiene capacidad de gestión en el sistema, sin ser necesariamente el *superusuario* (`root`). Esta capacidad puede ser desarrollada si dispone de privilegios gracias al comando `sudo` o si se encuentra en grupos de usuarios con privilegios sobre determinados archivos o comandos de gestión.

Los grupos en Linux son muy empleados, ya que facilitan la administración de privilegios en el sistema. Por ejemplo, se emplean cuando se desea que algunos usuarios tengan permisos sobre archivos o carpetas (lectura o edición), sin ser los propietarios de los mismos.

El fichero de configuración de grupos `/etc/group` centraliza la gestión de grupos en el sistema.



**Figura 4.2**  
Estructura del fichero `/etc/group`.

Cada fila se corresponde con un grupo y consta de cuatro campos delimitados por “:”, en el siguiente orden:

1. Nombre del grupo: nombre del grupo asociado al identificador del grupo.
2. Contraseña: no se suele utilizar, apareciendo una “x”, e indica que la contraseña se encuentra encriptada en el fichero de configuración */etc/gshadow*.
3. Identificador de grupo: GID o número de identificación del grupo único (Group IDentification).
4. Lista de usuarios: usuarios pertenecientes al grupo identificado como grupo secundario. Un usuario puede pertenecer a varios grupos.

**RECUERDA**

- ✓ Cada usuario ha de pertenecer a un grupo principal (cuarto campo del fichero */etc/passwd*), pero, además, puede pertenecer a varios grupos secundarios, especificándose en */etc/group*.

Los usuarios disponen de *login* y *UID* únicos y necesarios para identificarse en el sistema y poder operar en él. Existe un usuario de especial relevancia por su capacidad de gestión y administración sobre los recursos del sistema, conocido como *superusuario*, cuyo *login* es *root*. El *superusuario* dispone de tal control sobre el sistema que, en principio, no está “habilitado”, por lo que evita acciones perjudiciales de manera inconsciente.

El *prompt*, o la línea de petición de órdenes en el intérprete de comandos, varía según el usuario activo en él. En Ubuntu, cuando el *superusuario* se encuentra activo en el intérprete de comandos a la espera de introducir órdenes, su indicativo de petición en el *prompt* es el símbolo “#”, mientras que para el resto de usuarios es “\$”. No obstante, es posible su modificación mediante la edición de las variables *PS1* y *PS2*.

**TOMA NOTA**



Los ficheros de configuración son estructuras críticas, dada su importancia en el sistema. Todos estos ficheros pueden ser editados directamente con privilegios de *superusuario*, pero esto conlleva un riesgo innecesario, puesto que podemos tener un resultado catastrófico si erramos en su edición. Por tanto, lo conveniente es usar comandos del sistema para gestionar de manera adecuada los ficheros de configuración.

En la siguiente secuencia de órdenes, se ha “habilitado” al usuario *root* (estableciéndole una contraseña), para después cambiar al propio usuario *root*. Obsérvese cómo el identificador para *luis* es “\$” y para *root* “#”.

```

luis@luis-VirtualBox:~$ sudo passwd root
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
passwd: contraseña actualizada correctamente
luis@luis-VirtualBox:~$ su root
Contraseña:
root@luis-VirtualBox:/home/luis#

```

**Figura 4.3**  
Habilitación del superusuario.

El *superusuario* puede editar el fichero `/etc/passwd`: modificar los valores de los campos existentes y añadir o eliminar filas. El campo contraseña, al estar encriptada, debe ser modificada mediante el comando `passwd`.

La contraseña del usuario *root* se encuentra bloqueada por defecto en Ubuntu. No obstante, se pueden realizar acciones en su nombre sin ser él realmente quien las ejecute. Pero ¿cómo se puede gestionar el sistema en Linux si no está “habilitado” el usuario *root* por defecto? Es aquí donde entra en escena el comando `sudo` (tan empleado hasta ahora por nosotros). Este comando permite ejecutar comandos en nombre de otros usuarios, siempre que tanto el usuario como el comando que se va a ejecutar estén permitidos gracias al archivo de configuración `/etc/sudoers`.

Cuando ejecutamos `sudo`, este solicita la contraseña del usuario que lo ejecuta. Queda almacenada durante unos minutos y, pasado este tiempo, si se vuelve a emplear, deberá introducirse de nuevo. El comando `sudo` permite ejecutar comandos con privilegios de *superusuario*, pero también es posible ejecutar comandos de otros usuarios mediante la siguiente sintaxis:

```
sudo -u usuario comando_de_usuario
```

### Actividades propuestas



- 4.1.** Muestra el contenido del fichero `/etc/passwd`, mediante `cat /etc/passwd`. Analiza los campos de cada fila.
- 4.2.** Muestra el contenido del fichero `/etc/group`, mediante `cat /etc/group`. Analiza los campos de cada fila.
- 4.3.** Analiza el contenido del fichero `/etc/sudoers`. Averigua cómo se puede editar su contenido. ¿Es necesario editarlo para que un usuario pueda realizar acciones en nombre de otros usuarios?

Por tanto, gracias a `sudo`, se actúa temporalmente con privilegios de otros usuarios. En cuanto termina la acción del comando que se va a ejecutar con `sudo`, el usuario vuelve a cobrar sus propios privilegios.

### RECUERDA

- ✓ Habilitar al usuario *root* es generalmente innecesario, ya que la gran mayoría de acciones que necesita hacer un usuario como usuario administrador de Ubuntu se pueden hacer mediante `sudo`. De esta manera, se evita correr riesgos debido a usuarios que conozcan la contraseña del usuario *root* o accesos indebidos que intenten hacerse con el control del sistema. Por tanto, establecer una política administrativa adecuada del sistema mediante el comando `sudo` o la inserción en grupos de usuarios con capacidades administrativas es mucho más conveniente que trabajar con *root* directamente.

Otra forma de ejecutar acciones de otro usuario es cambiando de usuario directamente mediante el comando `su`, para lo que debemos conocer sus credenciales. El comando `su` permite ejecutar una sesión del intérprete de comandos como otro usuario, que resulta equivalente a iniciar la sesión en el sistema con otro usuario. Su sintaxis es:

```
su [-] [usuario]
```

Donde “–” es opcional y permite cargar las preferencias del usuario al iniciar su sesión (home de usuario, variables por defecto, etc.). Si no se especifica el usuario que va a cambiar su sesión, por defecto será *root*. Para salir de la sesión de usuario con *su*, debemos escribir “*exit*”.

#### 4.2.2. Comandos de gestión de usuarios

Para añadir un nuevo usuario al sistema, se emplea el comando *useradd*. Su sintaxis es:

```
useradd [-g grupo] [-G grupo[, grupo ...]] [-d directorio_trabajo [-m]]  
[-p contraseña_encriptada] [-s shell] login
```

Este comando crea una nueva entrada en */etc/passwd* y permite copiar los archivos del directorio */etc/skel* al directorio de usuario. El directorio */etc/skel* contiene los archivos de configuración por defecto que se añaden al directorio de trabajo de un usuario cuando este es creado con las opciones adecuadas. Las opciones más utilizadas son:

- *g grupo*: asignación al grupo principal. Todos los usuarios están adscritos, al menos, a un grupo principal y, en caso de pertenecer a más de uno, el resto serán grupos secundarios. Estos grupos han de existir. En caso de no especificar esta opción, se creará por defecto un grupo con su mismo nombre.
- *G grupos*: lista de grupos secundarios separados por comas y sin espacios.
- *d directorio de trabajo*: establece un directorio existente como directorio de trabajo para dicho usuario. Si no se especifica esta opción, se tomará por defecto */home/login de usuario*.
- *p contraseña encriptada*: contraseña del usuario. Si no se especifica, el usuario no podrá acceder al sistema.
- *m*: crea el directorio de trabajo si no existe o no se especifica. Se copian los archivos de configuración de */etc/skel*.
- *s shell*: establece un intérprete de comandos al usuario. Por defecto, se emplea */bin/bash*, aunque existen otros, como */bin/csh* o */bin/sh*.

#### Ejemplos

```
sudo useradd -m maria
```

Crea un usuario “maria” con grupo principal *maria*, estableciéndose su directorio en */home/maria* y copiando en él los archivos de configuración por defecto de */etc/skel*.

```
sudo useradd -g ventas -d /home/German -m -s /bin/bash german
```

Crea un usuario “german” con grupo principal “ventas” (ya existente), asignándole como directorio de trabajo */home/German* donde se copian los archivos de configuración por defecto de */etc/skel*. Además, se asigna a este usuario el shell */bin/bash*.

Las características de la cuenta de los usuarios del sistema pueden ser modificadas mediante el comando *usermod*. Su sintaxis es la siguiente:

```
usermod [-c comentario] [-g grupo] [-G grupo[, grupo ...]] [-d directorio_
trabajo [-m]]
[-p contraseña_encriptada] [-e fecha] [-f dias] [-l nuevoLogin] [-L] [-U]
[-s shell] login
```

Donde las opciones más utilizadas son:

- ✓ *c comentario*: establece valores asociados al quinto campo del fichero */etc/passwd*.
- ✓ *d directorio*: asigna un nuevo directorio de trabajo. Si se emplea junto con el modificador “*-m*”, todo el contenido del antiguo directorio de trabajo se moverá al nuevo.
- ✓ *g grupo*: asignación al nuevo grupo principal.
- ✓ *G grupos*: lista de grupos secundarios separados por comas y sin espacios. Se eliminará su asociación con los grupos secundarios anteriores, a menos que se indique con el modificador “*-a*” que se añaden los grupos nuevos a los anteriores.
- ✓ *l nuevoLogin*: se modifica el login anterior al nuevo login aportado.
- ✓ *s shell*: se modifica el anterior shell al nuevo shell aportado.

Para eliminar un usuario del sistema, se emplea el comando *userdel*. Su sintaxis es:

```
userdel [-r] login
```

Donde el modificador “*-r*” permite eliminar la carpeta home del usuario.

Los usuarios disponen de información adicional que se almacena en el quinto campo del fichero */etc/passwd*. Dependiendo de la distribución de GNU/Linux, esta información puede variar. Para actualizarla, se emplea el comando *chfn*.

#### Actividades propuestas



- 4.4.** Crea dos usuarios con los archivos de configuración del */etc/skel*. Elimina un usuario, manteniendo sus archivos. Elimina totalmente el otro usuario. Comprueba las acciones.
- 4.5.** Muestra el contenido del directorio */etc/skel*, mediante *ls -la /etc/skel*. Presta atención a cada fichero.

Como ya sabemos, Ubuntu es multiusuario, por lo que pueden existir varios usuarios trabajando en el sistema en paralelo. Para comprobar los usuarios conectados en el sistema, se utiliza el comando *who*. Su sintaxis es:

```
who [am i] [-u] [-H] [-q]
```

Donde:

- *am i*: muestra el usuario actual. Esto tiene mucho sentido, ya que podemos trabajar en varios terminales, o cambiando de identidad (mediante *su*, por ejemplo), llegando a perder la noción del usuario con el que estamos actuando en un preciso momento.
- *u*: muestra información de los usuarios conectados: login, terminal, fecha y hora de conexión, tiempo de inactividad e identificador del proceso shell de usuario.

- *H*: imprime cabeceras.
- *q*: muestra solamente los logins y el número de usuarios conectados.

Para añadir un nuevo grupo al sistema, empleamos el comando *groupadd*:

```
groupadd [-g GID] nombre_grupo
```

Donde:

- *g GID*: asigna un identificador de grupo al nombre de grupo. Por defecto, hemos de indicar un valor igual o superior a 1000, y no se deben repetir.
- También podemos eliminar un grupo del sistema mediante *groupdel*:

```
groupdel nombre_grupo
```

La eliminación de un grupo está supeditada la eliminación de las cuentas asociadas al grupo como primario.

Además, podemos modificar las características de los grupos con *groupmod*:

```
groupmod [-g GID] [-n nuevo_nombre_grupo] nombre_grupo
```

Donde:

- *g GID*: indica el nuevo identificador de grupo. Al realizar esta modificación, hemos de prestar atención, puesto que los ficheros con el antiguo *GID* deben ser asignados manualmente al nuevo *GID*.
- *n nuevo\_nombre\_grupo*: especifica el nuevo nombre para el grupo.

Podemos averiguar a qué grupos pertenecemos mediante el comando *groups [usuario]*. Al ejecutar este comando, el primer grupo es el principal y los sucesivos son los secundarios. No obstante, el comando *id [usuario]* muestra la información más detallada, especificando, además, el *UID* y los *GID* de los grupos.

Para añadir un usuario a un grupo, se emplea el comando *adduser* con la siguiente sintaxis:

```
adduser [login_usuario] grupo
```

Para eliminar un usuario de un grupo, se utiliza el comando *deluser* con la siguiente sintaxis:

```
deluser [login_usuario] grupo
```

#### 4.2.3. Usuarios y grupos predeterminados

En Linux, además del usuario predeterminado *root*, existe una serie de grupos por defecto cuya misión es otorgar permisos para así facilitar la administración del sistema. Estos se pueden ver en */etc/group*. Algunos de ellos son los que se muestran en el cuadro 4.1:

**CUADRO 4.1**  
Grupos predeterminados

Grupos	Descripción
adm	Grupo de administración que permite accesos a archivos de registro y comandos como <i>sudo</i> y <i>su</i>
users	Grupo de usuarios estándar
nobody	Sin privilegios
root	Administración sin restricciones sobre todo el sistema
tty	Aporta privilegios sobre algunos dispositivos, como /dev/tty
lpadmin	Confiere privilegios sobre dispositivos de puerto paralelo

Podemos modificar el propietario de un archivo, así como el grupo al que pertenece, usando el comando *chown* (*change owner*), cuya sintaxis es la siguiente:

```
chown [-R] [-h] nuevo_propietario[.nuevo_grupo] fichero
```

Donde:

- ✓ R: modo recursivo para aplicar los cambios al contenido de directorios.
- ✓ h: afecta al enlace simbólico, en lugar del archivo referenciado, ya que sin esta opción solo afectaría al archivo referenciado.

El propietario del archivo se sustituirá por *nuevo\_propietario* y el grupo por *nuevo\_grupo*. Tanto el propietario como el grupo han de existir en el sistema y se pueden modificar uno u otro por separado. El separador entre el propietario y el grupo es “.” o “:”, dependiendo de la distribución de Linux, aunque en sistemas GNU lo más común es “.”.

**TEN EN CUENTA**

- ✓ Solo puede cambiar el propietario de un archivo el usuario *root* (o mediante *sudo*). Además, para modificar el grupo de un fichero debemos ser *root* o cualquier otro usuario, siempre que pertenezca al grupo al que se desea modificar el grupo del fichero. Todo ello se debe a una cuestión de seguridad, ya que sin estas restricciones se podría asignar la propiedad de un fichero a otro usuario sin su consentimiento o su asignación a un grupo nuevo de usuarios (también sin el consentimiento de ellos).

También podemos modificar el grupo de un archivo únicamente mediante el comando *chgrp*, que presenta la siguiente sintaxis:

```
chgrp [-R] nuevo_grupo ficheros
```

Donde la opción más empleada es R que afecta al contenido de directorios de forma recursiva.

Y ficheros puede ser uno o varios ficheros o directorios, afectando el cambio de grupo a todos ellos.



#### Restricciones de chown con diferentes usuarios

Se puede apreciar cómo, por seguridad, no se permite al usuario *luis* asignar la propiedad del fichero *propietario.txt* a *maria*, sin embargo, sí permite el cambio del grupo (al pertenecer *luis* al grupo de *maria*). Sin embargo, con *sudo* no existe tal limitación. Por ello, se debería limitar la capacidad de acción de *chown* en sistemas críticos mediante *sudo* y su fichero de configuración */etc/sudoers*.

```

luis@luis-VirtualBox:~$ id luis
uid=1000(luis) gid=1000(luis) grupos=1000(luis),4(adm),24(cdrom),27(sudo),30(dlp),46(plugdev),118(lpadmin),129(sambashare),1064(maria),1005(carla)
luis@luis-VirtualBox:~$ chown maria propietario.txt
chown: cambiando el propietario de 'propietario.txt': Operación no permitida
luis@luis-VirtualBox:~$ sudo chown maria propietario.txt
luis@luis-VirtualBox:~$ ls -l propietario.txt
-rw-rw-r-- 1 maria maria 0 abr 19 21:35 propietario.txt

```

**Figura 4.4**  
Uso de chown con diferentes usuarios.



#### Actividad propuesta 4.6

Crea un usuario con un *GID* propio. Crea varios archivos con dicho usuario. Modifica el *GID* del usuario. Asigna los ficheros anteriores al nuevo *GID*.

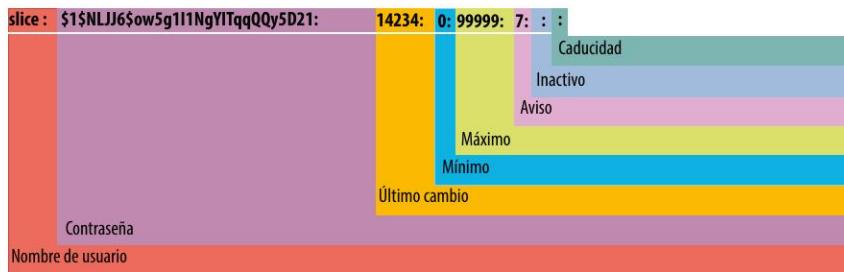
#### 4.2.4. Seguridad de cuentas de usuarios y contraseñas

Linux emplea un *sistema centralizado de autenticación* de usuarios para las distintas aplicaciones o comandos que intenten acceder al sistema, llamado *Linux-PAM*. Linux-PAM facilita la administración de la seguridad del sistema de forma versátil para cada aplicación, por tanto, el comportamiento de autenticación puede variar según el comando empleado.

Como ya sabemos, la seguridad de las cuentas de usuarios se basa en las contraseñas, y estas se gestionan gracias al fichero de configuración */etc/shadow*, del cual Linux-PAM hace uso, y que contiene las cuentas de los usuarios del sistema. Cada fila se corresponde con un usuario (al igual que */etc/passwd*), y consta de ocho campos delimitados por “:”, a saber:

1. Login de usuario.
2. Password encriptada. Dependiendo del algoritmo de cifrado, puede variar la extensión de la contraseña encriptada. Su comienzo especifica este: \$1\$ (MD5), \$5\$ (SHA-256), \$6\$ (SHA-512), etc. Si la contraseña comienza por “!” indica que se encuentra bloqueada.
3. Días transcurridos desde el 1/1/1970, cuando la contraseña fue cambiada por última vez.
4. Número de días, como mínimo, que han de pasar para que el usuario pueda cambiar la contraseña.
5. Número máximo de días que la contraseña es válida.
6. Número de días que el sistema avisa antes de que caduque la contraseña.

7. Número de días después de que la contraseña caduque para deshabilitar la cuenta.
8. Días transcurridos desde el 1/1/1970, cuando la cuenta será deshabilitada.



**Figura 4.5**  
Estructura del fichero /etc/shadow.

Cuando se crea un usuario con *useradd*, no se le asigna una contraseña por defecto y tampoco se le solicita al usuario en cuestión, por lo que este no puede acceder al sistema. Por tanto, un administrador del sistema, el usuario *root* o aquel que disponga de privilegios mediante *sudo*, podrá asignar contraseñas a usuarios.

Por tanto, podemos asignar una contraseña mediante:

- Comando *passwd*. Permite modificar una contraseña ya existente de un usuario o crear una por primera vez. Esta contraseña se encripta y se almacena en el fichero /etc/shadow. Un usuario solo puede modificar su contraseña, mientras que el usuario *root* puede modificar la contraseña de otros usuarios mediante:

```
passwd [usuario]
```

- Comando *openssl passwd*. Se trata de una herramienta criptográfica que permite generar contraseñas hash. El comando que se va a emplear es:

```
openssl passwd [opciones] "contraseña_a_encriptar"
```

#### Actividad propuesta 4.7



Modifica la contraseña de un usuario mediante: *sudo passwd usuario*.

Las opciones más empleadas son:

- 1. Genera una contraseña basada en un algoritmo de encriptación en MD5.
- 5. Genera una contraseña basada en un algoritmo de encriptación en SHA-256.
- 6. Genera una contraseña basada en un algoritmo de encriptación en SHA-512.

Esto generará una contraseña encriptada, como vemos en el ejemplo de la figura 4.6.

```
luis@luis-VirtualBox:~$ openssl passwd -1 oso12  
$1$dob/0niR$Lt/DVLgPAAEmvU5Jxetzf/
```

**Figura 4.6**  
Uso de openssl.

Los comandos *useradd* o *usermod* pueden establecer la contraseña de un usuario, pero ha de estar encriptada. Para ello, podemos trasladarla tal cual, ejecutando previamente *openssl passwd* o, más versátil aún, podemos lanzarlo dentro de *useradd* o *usermod*, gracias a la sustitución de comandos mediante \${comando}. Esto hará que se resuelva el comando dentro del paréntesis antes de ejecutar la línea entera. Por ejemplo:

```
sudo useradd -m -p $(openssl passwd -1 oso12) carla
```

TOMA NOTA



*Recomendaciones de uso de contraseñas*

La gestión de contraseñas de usuarios es crucial para la seguridad del sistema. Se pueden emplear multitud de herramientas que nos ayudan a establecer políticas de gestión de contraseñas, pero, como siempre, el sentido común es nuestro gran aliado. Por tanto, recomendamos seguir estos sencillos consejos:

- Emplear contraseñas únicas para cada usuario y cuenta o equipo. Es decir, si disponemos de varias cuentas de acceso para diferentes gestores de correo electrónico, diferentes equipos (móviles, equipos de sobremesa), banca online, etc., todas han de ser únicas.
- Emplear contraseñas largas con al menos ocho caracteres, que incluyan letras mayúsculas, letras minúsculas, números y caracteres especiales.
- Evitar contraseñas previsibles, como fechas especiales asociadas a nosotros o a alguien cercano, nombre de mascotas, apodos o más sencillas aún, como "1234" o que incluyan nombres, apellidos, login de usuario, etc.
- Cambiar la contraseña frecuentemente.

Los administradores de Linux pueden gestionar y establecer políticas de caducidad de contraseñas mediante los comandos *passwd* y *chage*, modificando así valores del archivo de configuración */etc/shadow*.

El comando *chage* presenta la siguiente sintaxis:

```
chage [opciones] [login]
```

Donde las opciones más usuales son:

- ✓ d, -- lastday FECHA/DÍAS: establece la fecha o número de días desde 1/1/1970, cuando la contraseña fue modificada por última vez. Un valor igual a 0 obliga a la expiración de la contraseña y, por tanto, a su actualización.

- ✓ E, -- expiredate *FECHA*: establece la fecha (en número de días desde 1/1/1970) a partir de la cual la cuenta del usuario caduca y, por tanto, no será accesible. A -1 indica que no existe tal limitación y a 1 la cuenta se deshabilita de inmediato.
- ✓ I, -- inactive *DIAS*: establece el número de días de inactividad después de que una contraseña haya expirado antes de que la cuenta se bloquee. Para poder desbloquear la cuenta, el usuario deberá ponerse en contacto con un administrador o el usuario *root*. A -1 indica que no existe tal limitación y a 0 se deshabilitará la cuenta en cuanto expire la contraseña.
- ✓ -l, -- list: muestra la información de caducidad de la contraseña.
- ✓ m, -- mindays *DIAS*: establece el número mínimo de días para poder cambiar la contraseña. A 0 indica que no existe tal limitación.
- ✓ M, -- maxdays *DIAS*: establece el número máximo de días para poder cambiar la contraseña, es decir, para que caduque, desde que se cambió la contraseña por última vez. Transcurrido este número máximo de días, el sistema instará al usuario para que cambie la contraseña, antes de poder usar su cuenta. A -1 indica que no existe tal limitación.
- ✓ W, - – warndays *DIAS*: establece el número de días de aviso previos a que la contraseña caduque.

#### TEN EN CUENTA

- ✓ Al bloquear o dejar en blanco una contraseña, no se llega a deshabilitar la cuenta del usuario. Para ello, hemos de emplear *chage* y *usermod* (con la opción “-- expiredate 1”). De lo contrario, el usuario podría acceder al sistema por otros medios, como, por ejemplo, *ssh*. Para comprobar el estado de la cuenta o contraseña de usuario podemos ejecutar:

```
chage -l usuario
passwd -- status usuario
```

El formato por defecto de fecha es *YYYY-MM-DD*. Por otro lado, en caso de no emplear opciones, *chage* se ejecutará en modo interactivo, solicitando valores para cada campo.

El comando *passwd*, además de establecer la contraseña para un usuario, permite modificar otros valores relativos a esta. Su sintaxis es:

```
passwd [opciones] [login]
```

Donde las opciones más empleadas son:

- d, -- delete: deja en blanco la contraseña del usuario.
- e, -- expire: hace expirar la contraseña de usuario, haciendo que en la siguiente conexión el sistema la solicite.
- i, -- inactive *DIAS*: establece el número de días de inactividad después de que una contraseña haya expirado, antes de que la cuenta se bloquee. A -1 indica que no existe tal limitación y a 0 se deshabilitará la cuenta en cuanto expire la contraseña.
- l, -- lock: bloquea la contraseña de la cuenta de usuario.
- u, -- unlock: desbloquea la contraseña de la cuenta de usuario.

- x, -- maxdays *DIAS*: establece el número máximo de días para poder cambiar la contraseña, es decir, para que caduque, desde que se cambió la contraseña por última vez. Transcurrido este número máximo de días, el sistema instará al usuario para que cambie la contraseña, antes de poder usar su cuenta. A -1 indica que no existe tal limitación.
- w, -- warndays *DIAS*: establece el número de días de aviso previos a que la contraseña caduque.

Además, el comando *usermod* dispone de varias opciones que también permiten modificar ciertos valores relacionados con la seguridad de la cuenta o la contraseña de los usuarios, ya tratados en los anteriores comandos:

- ✓ e, -- expiredate *FECHA*: establece la fecha (en número de días desde 1/1/1970) a partir de la cual la cuenta del usuario caduca y, por tanto, no será accesible. A -1 indica que no existe tal limitación y a 1 la cuenta se deshabilita de inmediato.
- ✓ f, -- inactive *DIAS*: establece el número de días después de que una contraseña haya expirado antes de que la cuenta se deshabilite. A -1 indica que no existe tal limitación y a 0 se deshabilitará la cuenta en cuanto expire la contraseña.
- ✓ L, -- lock: bloquea la contraseña de la cuenta de usuario.
- ✓ U, -- unlock: desbloquea la contraseña de la cuenta de usuario.
- ✓ p, -- password *CONTRASEÑA ENcriptada*: asigna una contraseña ya encriptada a un usuario.
- ✓ f *DIAS*: establece el número de días que transcurrirán entre la fecha de expiración de la contraseña y su eliminación definitiva. A 0 indica que la cuenta se deshabilitará justo cuando expire la contraseña. Y a -1 no se habilita esta característica (por defecto).

#### RECUERDA

- ✓ Habilitar la cuenta *root* puede ser peligroso y, por tanto, no es aconsejable. Realizar una buena política de seguridad sobre el usuario *root*, mediante la edición de los archivos de configuración de aquellos comandos que accedan al sistema, aquellos que ejecuten comandos en su nombre y, sobre todo, de Linux-PAM, es la solución idónea.

#### 4.2.5. Acceso a recursos y permisos locales

Los archivos son recursos del sistema operativo y, por tanto, este ha de disponer de herramientas para discriminar qué usuarios y grupos pueden acceder a aquellos y con qué operaciones.

Como ya sabemos, el *i-nodo* de los archivos almacena el usuario propietario del archivo (*UID*), su grupo propietario (*GID*) y la máscara de permisos, que presenta la siguiente estructura:

Los permisos de usuario son los privilegios que tendrá un usuario como propietario de un archivo

- **RWXRWXRWX**



**Figura 4.7**  
Máscara de permisos.

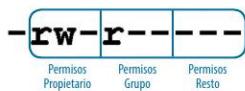
(primer grupo de tres bits). Los permisos de grupo son los permisos que tienen los usuarios si pertenecen al grupo del fichero (segundo grupo de tres bits). Y, en otro caso, que el usuario no sea el propietario ni pertenezca al grupo del fichero, entonces disfrutará de los privilegios para otros (tercer grupo de tres bits).

Cada grupo de tres bits representa un conjunto de permisos de lectura (r), escritura (w) y ejecución (x), los cuales se pueden visualizar mediante `ls -l`. Cuando un archivo no presenta permiso, se representa por “-”. Estos permisos se interpretan de diferente manera si se aplican sobre ficheros o directorios:

**CUADRO 4.2**

**Permisos sobre archivos y carpetas**

Permisos	Archivos	Carpetas
Permiso de lectura (r)	Puede ser leído o visualizado.	Se puede visualizar su contenido, mostrando los archivos o carpetas que contenga.
Permiso de escritura (w)	Pueden modificar su contenido, sus permisos, el propietario y el grupo.	Permite modificar el contenido, creando o eliminando archivos o carpetas en ella.
Permiso de ejecución (x)	Permite ejecutarlo.	Permite acceder a ella.



Máscara de permisos de un archivo regular en el que el propietario puede leer y escribir, los usuarios del grupo pueden leer y el resto de usuarios no disponen de privilegios.



Máscara de permisos de un archivo regular en el que el propietario puede leer y escribir, los usuarios del grupo pueden leer y el resto de usuarios pueden leer y escribir.



Máscara de permisos de un archivo regular en el que el propietario no dispone de privilegios, los usuarios del grupo pueden leer y ejecutar, y el resto de usuarios pueden leer, escribir y ejecutar.



Máscara de permisos de un directorio en el que el propietario puede leer su contenido, modificarlo y acceder a él, los usuarios del grupo pueden leer su contenido y el resto de usuarios no disponen de privilegios.

**Figura 4.8**  
Ejemplos de máscaras de permisos.

*A priori*, solo los privilegios se asocian al propietario, a los usuarios del grupo del fichero y al resto de usuarios del sistema. Si se desea aplicar permisos especiales sobre un archivo o carpeta a otro conjunto de usuarios, no se puede establecer, a menos que se empleen las *listas de control de acceso (ACL)*.

Existen unos bits llamados “raros” que se asocian a unos *modos especiales*, a saber:

- a) *Set-uid*. Este bit hace referencia a la propiedad que un archivo o carpeta puede adoptar cuando un usuario (que no sea el propietario) lo ejecuta, ya que tomaría el UID del propietario. Si el propietario es el *superusuario*, entonces otro usuario podría ejecutar tal archivo como si fuese *root*. Esta propiedad solo es posible durante la ejecución, evitando problema alguno de seguridad. El bit *set-uid* no tiene efecto si se aplica sobre directorios o si el propietario no tiene permiso de ejecución. Cuando está activo el *set-uid*, se simboliza con “s” en lugar de “x” en los bits de la máscara de permisos del propietario. Si no es ejecutable el archivo y dispone de *set-uid*, se representará por “S”. Ejemplo de ello es el archivo ejecutable *passwd*, pues permite ser ejecutado por cualquier usuario, ya que tiene activo el *set-uid*. De esta manera, un usuario puede ejecutar dicho comando y así modificar su contraseña actuando durante su ejecución como *root* (para así modificar los archivos */etc/passwd* y */etc/shadow*). Igualmente, ocurre con muchos otros comandos como *su*.
  - b) *Set-gid*. Presenta unas características diferentes, según se aplique a archivos o directorios.
    - Cuando está activo en archivos, su uso es similar al *set-uid*. Permite que, al ejecutarse un fichero, se realice con el *GID* de su grupo y, por tanto, con los privilegios del *GID* del grupo durante la ejecución. Esto hace que los usuarios de un grupo puedan trabajar con los ficheros de ese mismo grupo.
    - Cuando está activo en directorios, los archivos y subdirectorios creados en su interior serán forzados a pertenecer al grupo del directorio y no al grupo del usuario que lo haya creado.
- Cuando está activo el *set-gid*, se simboliza con “s” en lugar de “x” en los bits de la máscara de permisos del grupo. Si no es ejecutable el archivo o el directorio y dispone de *set-gid*, se representará por “S”. No obstante, no tiene efecto si el grupo no tiene permiso de ejecución.
- c) *Sticky-bit* o *bit de permanencia*. Este bit permite establecer características especiales sobre directorios, ya que sobre archivos no tiene efecto actualmente en sistemas Linux. En directorios, permite que solo el propietario del archivo creado dentro del directorio con *sticky-bit* activo o el propietario del directorio con *sticky-bit* activo pueda eliminar o modificar su contenido. Es decir, si suponemos que un usuario (que no sea el propietario del directorio con *sticky-bit* activo) dispone de permisos de lectura, escritura y ejecución, no podría eliminar o modificar su contenido. Cuando está activo el *sticky-bit*, se simboliza con “t”, en lugar de “x”, en los bits de la máscara de permisos del resto de usuarios. Si no es ejecutable el archivo o el directorio y sí dispone de *sticky-bit*, se representará por “T”. No obstante, no tiene efecto si el resto de usuarios no tiene permiso de ejecución. Un ejemplo de ello es lo que ocurre en directorios como */tmp* y */var/tmp*, donde cualquier usuario puede acceder a ellos y crear archivos, pero solo los usuarios que los crean pueden eliminarlos o modificarlos.

**Actividades propuestas**

- 4.8.** Ejecuta `ls -l /usr/bin/passwd`. Observa quién es el propietario y si se encuentra activo el set-uid en dicho archivo.
- 4.9.** Ejecuta `ls -ld /tmp`. Observa quién es el propietario, el grupo y si se encuentra activo el sticky-bit del directorio.

**4.2.6. Modificación de permisos**

Para modificar los permisos de un archivo, se utiliza el comando `chmod` (change mode) siendo propietarios del archivo o superusuario. Su sintaxis es la siguiente:

```
chmod [-R] permisos archivos
```

La opción `-R` es recursiva para un directorio. Donde *permisos* puede especificarse en modo octal o simbólico. Detallamos el procedimiento para cada uno de ellos:

**A) Octal**

Hemos de activar o desactivar cada bit sobre los permisos en binario mediante un 1 o un 0 respectivamente. Por ejemplo, si quisieramos habilitar un archivo con máscara de permisos “`rwxrw-r --`” deberíamos aplicar el siguiente código en binario: 111110100. Posteriormente, debemos pasar el código en binario con la máscara de permisos a octal. En nuestro caso, la cadena binaria anterior se corresponde con 764 en octal.

Además, para habilitar los modos especiales sobre la máscara de permisos hemos de aplicar el siguiente procedimiento:

- Sticky-bit: sumar 1000 en octal al resultado anterior.
- Set-gid: sumar 2000 en octal al resultado.
- Set-uid: sumar 4000 en octal al resultado anterior.

El resultado obtenido constituirá la máscara de permisos como argumento de `chmod` sobre un archivo determinado. Como, por ejemplo: `chmod 5776 prueba.txt`, quedando una máscara de permisos `rwsrwxrwT`.

**Actividad resuelta 4.1**

Crear un archivo `prueba.txt` y aplicar las siguientes máscaras de permisos en octal, sobre `prueba.txt`, comprobando su resultado: “`rwsrwxrwT`”, “`rwxrwxrw-`” y “`rw-r-srw-`”.

SOLUCIÓN

```
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rw-r--r-- 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod 5776 prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
rwsrwxrwT 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod 776 prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rwxrwxrw- 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod 2656 prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rw-r-srw- 1 luis luis 0 abr 15 17:27 prueba.txt
```

**Figura 4.9**  
Ejemplo de `chmod` en modo octal.

La máscara de permisos "rwsrwxrwT" para aplicar con chmod es 5776.  
La máscara de permisos "rwxrwxrw-" para aplicar con chmod es 0776 o 776.  
La máscara de permisos "rw-r-srw-" para aplicar con chmod es 2656.

## B) Simbólico

Es una manera más sencilla de aplicar una máscara de permisos, pero, sobre todo, de modificarla con relación a su valor actual. Su sintaxis es la siguiente:

chmod modo\_simbolico[,modo\_simbolico] archivos

El *modo simbólico* se puede repetir más de una vez, separándose por comas. Su formato es:

[u] [g] [o] {+ | = | -} [r] [w] [x] [s] [t]

Donde el formato que se va a aplicar se ha de especificar en tres grupos:

- ✓ Destinatarios de los permisos para modificar:

“u”: propietario.  
“g”: grupo.  
“o”: otros (el resto de usuarios).  
“a”: propietario, grupo y otros.

- ✓ Tipo de modificación:

“+”: se añaden al valor actual.  
“=”: establece los permisos especificados y anula el resto.  
“-”: se quitan al valor actual.

- ✓ Permisos:

“r”: lectura.  
“w”: escritura.  
“x”: ejecución.  
“s”: set-uid o set-gid, dependiendo de su aplicación al propietario o al grupo.  
“t”: sticky-bit.

Por ejemplo, si quisieramos deshabilitar para el grupo y otros el permiso de lectura:

```
chmod go-r prueba.txt.
```

**Actividad resuelta 4.2**

Sobre el archivo prueba.txt anterior, aplicar las siguientes modificaciones sobre la máscara de permisos comprobando sus resultados.

**SOLUCIÓN**

- Deshabilitar para el grupo y otros el permiso de lectura.
- Habilitar el permiso de ejecución para el propietario y deshabilitar el *set-gid*.
- Conceder permisos de lectura y escritura para el usuario y el grupo, anulando el resto de permisos. Para el resto de usuarios se mantienen sus permisos.

```
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rw-r-srw- 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod go-r prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rw--s-w- 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod u+x,g-s prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rwx--x-w- 1 luis luis 0 abr 15 17:27 prueba.txt
luis@luis-VirtualBox:~$ chmod ug=rw prueba.txt
luis@luis-VirtualBox:~$ ls -l prueba.txt
-rw-rw--w- 1 luis luis 0 abr 15 17:27 prueba.txt
```

**Figura 4.10**  
Ejemplo de chmod en modo simbólico.

#### 4.2.7. Permisos por defecto

Los permisos originales de un archivo y un directorio son 0666 y 0777 en octal, respectivamente. Sin embargo, sobre estos permisos se aplica una máscara de permisos, dando como resultado los permisos que posee un archivo o directorio cuando son creados. Su finalidad es aportar seguridad a los archivos y directorios al crearlos, siendo el valor, por defecto, de la máscara de permisos 0022 o 0002 (aunque depende de la distribución de Linux y el usuario).

Para ver o modificar la máscara de permisos, empleamos el comando *umask* (user mask) con la siguiente sintaxis:

```
umask [máscara]
```

Donde *máscara* ha de ser el valor de la máscara de permisos en octal (aunque se puede emplear la opción “-S” para usar la notación simbólica). Si no se especifica, mostrará el valor de la máscara de permisos.

El procedimiento de aplicación de la máscara de permisos es el siguiente:

1. Convertimos la máscara de permisos de octal a binario.
2. Aplicamos el operador NOT sobre la máscara en binario.
3. Realizamos la operación AND entre los permisos originales de archivos o directorios y la máscara.

Independientemente de la máscara que se aplique, el resultado no activa los bits de ejecución para los archivos, puesto que los permisos originales de estos (0666) no tienen habilitado el bit de ejecución.



### Actividad resuelta 4.3

*Obtener los permisos de los archivos y directorios recién creados aplicando una máscara de permisos 0022.*

#### SOLUCIÓN

1. Convertimos la máscara de permisos de octal a binario.  $0022_{10} = 000\ 000\ 010\ 010_{12}$
2. Aplicamos el operador NOT a la cadena binaria anterior.  
 $\text{NOT } 000\ 000\ 010\ 010_{12} = 111\ 111\ 101\ 101_{12}$
3. Realizamos la operación AND lógica entre los permisos originales de archivos (0666) o directorios (0777) y la cadena binaria anterior.

Para Archivos:

$$\begin{array}{rcl} 0666_{12} & = & 000\ 110\ 110\ 110_{12} \\ \text{AND} & & \underline{111\ 111\ 101\ 101_{12}} \\ & = & 000\ 110\ 100\ 100_{12} = 0644_{10} \end{array}$$

Para Directorios

$$\begin{array}{rcl} 0777_{12} & = & 000\ 111\ 111\ 111_{12} \\ \text{AND} & & \underline{111\ 111\ 101\ 101_{12}} \\ & = & 000\ 111\ 101\ 101_{12} = 0755_{10} \end{array}$$

La modificación de la máscara de permisos mediante *umask* es temporal y solo afecta al proceso asociado al *shell* o a los procesos generados por este. Sin embargo, podemos establecer permanentemente el valor de la máscara de permisos mediante los archivos de configuración globales, como */etc/profile* (afectan a todos los usuarios y solo se podría hacer con permisos de *superusuario*), o locales como *~/.bashrc* (solo afecta a la cuenta de usuario).



### Actividad resuelta 4.4

*Crear un archivo prueba\_umask.txt y un directorio prueba\_umask. Modificar la máscara de permisos a 0002. Crear un nuevo archivo prueba\_umask2.txt y un nuevo directorio prueba\_umask2 volviendo a comprobar sus permisos.*

#### SOLUCIÓN

```
[luis@luis-VirtualBox:~$ umask
0022
luis@luis-VirtualBox:~$ touch prueba_umask.txt ; mkdir prueba_umask
luis@luis-VirtualBox:~$ umask 0002
luis@luis-VirtualBox:~$ touch prueba_umask2.txt ; mkdir prueba_umask2
luis@luis-VirtualBox:~$ ls -ld prueba_umask*
drwxr-xr-x 2 luis luis 4096 abr 15 21:23 prueba_umask
drwxrwxr-x 2 luis luis 4096 abr 15 21:24 prueba_umask2
-rw-rw-r-- 1 luis luis    0 abr 15 21:24 prueba_umask2.txt
-rw-r--r-- 1 luis luis    0 abr 15 21:23 prueba_umask.txt
```

Figura 4.11

Ejemplo de modificación de la máscara de permisos con *umask*.

#### 4.2.8. Configuración de perfiles

Los perfiles de usuario albergan la configuración del entorno de trabajo de los usuarios en el sistema operativo. La configuración afecta desde que el usuario inicia sesión hasta que este se desconecta, gracias a archivos globales o locales. Los archivos de configuración globales afectan a todos los usuarios y necesitan permisos de *superusuario* para su edición. Sin embargo, los archivos de configuración locales (situados en el directorio home de cada usuario) solo afectan al propio usuario y este los puede editar.

En Ubuntu, los archivos de configuración globales y locales, por defecto, son:

**CUADRO 4.3**  
Archivos de configuración

Tipos	Archivos	Descripción
Archivos globales	/etc/skel	Directorio que contiene la plantilla de creación de perfiles de usuarios.
	/etc/profile	Configuración genérica de perfiles cuando se inicia sesión en el sistema como <i>login shell</i> .
	/etc/bash.bashrc	Configuración genérica de perfiles cuando se inicia sesión con Shell Bash <i>interactivo</i> (ya sea <i>login</i> o <i>non-login</i> ).
Archivos locales	~/.bashrc	Configuración local de usuario que se ejecuta cuando este inicia sesión con Shell Bash como <i>non-login shell</i> .
	~/.bash_logout	Configuración local de usuario que se ejecuta cuando este termina sesión con Shell Bash
	~/.profile	Configuración local de usuario cuando este inicia sesión en el sistema con un shell de inicio de sesión como <i>login shell</i> .

El intérprete de comandos lanza unos u otros archivos de configuración, dependiendo de cómo inicie el usuario la sesión. Hemos de diferenciar la ejecución de un shell en diferentes modos:

- *Login shell*: aquel que cuando se inicia el shell autentica al usuario solicitando usuario y contraseña. Como, por ejemplo, cuando abrimos un nuevo terminal virtual (CTR-L+ALT+F1).
- *Non-login shell*: aquel que no solicita autenticación del usuario, como, por ejemplo, cuando abrimos un terminal en *Gnome* o cuando lanzamos el comando *bash* dentro de un *login shell*.
- *Shell interactivo*: aquel que permite interactuar con el terminal escribiendo comandos, interrumpirlos, etc., al estar asociado a un terminal. Ejemplos de ello son los terminales virtuales o los terminales *Gnome*.
- *Shell no interactivo*: no está asociado a un terminal, como suele ser un *subshell*, que normalmente se ejecuta en procesos automáticos, como, por ejemplo, cuando se ejecuta un script.

En sesiones interactivas, cuando un usuario inicia sesión como *login shell*, se leerán, en primer lugar, los ficheros */etc/profile*, */etc/bash.bashrc* y, después, el archivo de configuración del home de usuario *~/.profile*. Por el contrario, un terminal interactivo abierto sin autenticación (*non-login shell*) leerá el archivo */etc/bash.bashrc* y, después, *~/.bashrc*.

Aunque una configuración diferenciada *non-login shell* y *login shell* aporta flexibilidad al sistema para diferentes accesos, la mayoría de distribuciones de Linux aúnan ambas, ya que la mayoría

de las veces se desea la misma configuración. Por ello, los archivos de configuración *non-login shell* como `~/.bashrc`, pueden definir variables que estén disponibles en los entornos *login shell*.

La edición de estos archivos de configuración supone un conocimiento profundo del funcionamiento del sistema operativo y de *shell scripting* (programación shell). Y se emplean principalmente para:

- ✓ Asignar o personalizar variables.
- ✓ Ejecutar comandos al entrar o salir del sistema o del intérprete de comandos.
- ✓ Asignar el *PATH* que indica las rutas de búsqueda para ejecutar archivos.
- ✓ Personalizar el *prompt*.

El shell se puede configurar mediante variables, alias y opciones.

### A) Variables

Una *variable* es un identificador que almacena una cadena de caracteres. Son empleadas por el sistema, programas o el propio Shell, almacenando valores de configuración de cada uno de ellos. Una variable se define mediante:

```
VARIABLE=VALOR
```

Una variable puede tener múltiples valores, separándose por ":";

```
VARIABLE=valor1:valor2:...
```

Si el valor de una variable contiene espacios, se han de poner dobles comillas:

```
VARIABLE="valor con espacios"
```

Para ver el contenido de una variable, se antepone el símbolo "\$" al nombre de la variable:

```
$VARIABLE
```

En caso de ver el contenido de una variable no definida, esta devolverá una cadena vacía. Ejemplo: `echo $HOME` (el comando `echo` permite mostrar por pantalla).

Se distinguen dos tipos de variables:

1. Variables globales o de entorno: son reconocidas y pueden ser utilizadas por procesos hijos del shell. Podemos ver una lista de variables de entorno mediante los comandos `env` y `printenv`. Las variables de entorno más comunes son:

**CUADRO 4.4**  
**Variables globales**

Variables globales	Descripción
SHELL	Shell por defecto
USER	Usuario actual
PWD	Directorio de trabajo actual
OLDPWD	Directorio de trabajo previo
PATH	Conjunto de rutas de directorios que contienen ejecutables. El usuario no se tendrá que preocupar por indicar la ruta completa de los archivos ejecutables contenidos en esta variable
HOME	Directorio home del usuario

2. Variables locales o de shell: solo pueden ser interpretadas por el shell. Cuando se define una variable en un shell, esta es local al shell y, por tanto, solo se reconocerá dentro de él. Es decir, si ejecutamos un programa en un shell, el programa no reconocerá las variables locales. Para ver una lista de las variables locales, se emplea el comando *set*, no obstante, en esta lista también aparecerán las variables de entorno. Las variables locales más comunes son:

**CUADRO 4.5**  
Variables locales

Variables locales	Descripción
HOSTNAME	Nombre del equipo
IFS	Valores de separación en la línea de comandos
PS1	Valor del prompt
UID	UID del usuario actual

Para establecer una variable en global, hemos de exportarla empleando: *export VARIABLE*. De esta manera, los procesos hijos podrán utilizarla. Sin embargo, la definición de una variable global en los procesos hijos no afectarán al padre, solo a los nietos y subsecuentes por seguridad.

Para convertir una variable global a local se emplea: *export -n VARIABLE*.

Para eliminar una variable local o de entorno por completo, se emplea: *unset VARIABLE*.

Cuando estamos trabajando en un terminal y definimos una variable local o de entorno, al cerrar el terminal desaparecerá. Para hacer permanente una variable de shell o de entorno, esta se ha de definir en alguno de los archivos de configuración de perfiles, normalmente en *~/.bashrc*.

### Actividades propuestas



- 4.10.** Crea una variable local PRUEBA con valor "1". Comprueba que es una variable local. Conviértela en global y compruébalo. Elimínala.
- 4.11.** Crea una variable local PRUEBA con valor "2". Comprueba que es una variable local lanzando el programa *bash* (este programa es el propio intérprete de comandos que, al ser lanzado desde un shell, crea un proceso hijo para su ejecución). Vuelve al proceso padre de *bash* (*exit*) y convierte la variable PRUEBA en global. Comprueba que es una variable global lanzando el programa *bash*. Dentro de *bash*, crea la variable de entorno PRUEBA\_HIJO con valor "3". Vuelve al proceso padre y comprueba que no es reconocida. Elimina la variable PRUEBA.
- 4.12.** Define una variable de entorno PRUEBA con valor "OK" de forma permanente en el archivo de configuración *non-login shell* *~/.bashrc* y comprueba que podemos ver su contenido desde un terminal *login shell*.
- 4.12.** Busca información en Internet para editar las variables PS1 y PS2 mediante secuencias de escape.

### B) Alias

Por otro lado, los alias permiten ejecutar comandos de manera personalizada. De esta manera, facilitan la ejecución de comandos “propios”, ahorrando tiempo en la escritura. Los alias actúan como variables locales, por lo que estos desaparecerán cuando termine la ejecución del shell, a menos que sean definidos en los archivos de configuración como `~/.bashrc`. Podemos ver los alias definidos con el comando `alias`.

- Para crear un alias, se emplea: `alias nombre_alias='comandos'`
- Para eliminar un alias, se emplea: `unalias nombre_alias`



#### Actividad propuesta 4.13

Crea un alias con nombre “li” que ejecute “ls -lra”. Comprueba la creación del alias y bórralo.

### 4.3. Gestión de usuarios por interfaz gráfica en Windows

En Microsoft Windows la gestión gráfica de usuarios se realiza desde ‘Cuentas’ en ‘Configuración’. Podemos añadir nuevos usuarios en ‘Agregar otra persona a este equipo’, donde solicitará la forma de iniciar sesión. Pulsamos en ‘No tengo los datos de inicio de sesión de esta persona’, si no conocemos o no deseamos acceder por email o teléfono. A continuación, solicitará los datos para crear una cuenta Microsoft. Si no deseamos crearla o se desea acceder mediante credenciales tradicionales, pulsamos en ‘Agregar un usuario sin cuenta Microsoft’. Por último, introducimos el usuario y la contraseña. Desde cuentas podemos ver el conjunto de usuarios, con la posibilidad de cambiar su tipo de cuenta (administrador o estándar) o eliminar la cuenta.

Otra manera de gestionar usuario en Windows es a través de la opción ‘Cuentas de usuario’ del ‘Panel de control’, donde podemos gestionar la cuenta actual o ‘Administrar otra cuenta’. Si pulsamos en ‘Administrar otra cuenta’, podremos añadir nuevos usuarios, gestionarlos o eliminarlos.

Por otro lado, y para una administración más profunda, podemos acceder a ‘Administración de equipos’ de la opción de ‘Herramientas administrativas’ del ‘Panel de control’, desde donde podemos realizar toda la administración de usuarios y grupos del sistema a través de ‘Usuarios y grupos locales’.

Por defecto, Windows crea varias cuentas administrativas que no se encuentran habilitadas por seguridad, aunque a través de esta ventana podemos activarlas. Como son:

- ✓ *Administrador*: cuenta con los privilegios más altos del sistema, que permite realizar cualquier acción, similar a *root* en Linux.
- ✓ *Invitado*: cuenta destinada a aquellos usuarios que acceden al sistema esporádicamente, sin apenas privilegios.

Además, desde *Grupos* (dentro de ‘Usuarios y grupos locales’), podemos ver los usuarios que pertenecen a los diferentes grupos del sistema y crear otros nuevos. Los usuarios de tipo *Administrador* pertenecen al grupo *Administradores*, permiten acceder a la mayoría de las configuraciones del equipo. Los usuarios de tipo estándar pertenecen al grupo *Usuarios*, los cuales no pueden realizar acciones que afecten a otros usuarios.

#### Actividad propuesta 4.14



Accede al ‘Administrador de equipos’ de Microsoft Windows y muestra todos los usuarios (deshabilitados o no) y grupos del sistema, leyendo detenidamente su descripción. ¿Qué usuarios se encuentran deshabilitados por defecto?

## 4.4. Gestión de procesos por línea de comandos en Linux

Los sistemas operativos más utilizados, como Microsoft Windows, macOS y GNU/Linux, son multitarea y multiusuario. Por tanto, el sistema operativo trabaja de forma concurrente con todas las tareas y usuarios al mismo tiempo. En realidad, el sistema operativo asigna pequeños espacios de tiempo a cada tarea y usuario, para así atenderlos y crear la sensación de trabajo con todos ellos a la vez.

Las instancias de los programas en ejecución, también llamados *tareas* o *procesos*, son administrados por el sistema operativo como un recurso más. Dependiendo de los privilegios de los usuarios sobre el sistema, estos podrán modificar la planificación de procesos gracias a comandos específicos.

### 4.4.1. Procesos y servicios

El sistema operativo gestiona todos los procesos mediante operaciones de creación, comunicación, compartición y finalización de procesos. El módulo del sistema operativo encargado de realizar estas tareas es el *planificador de procesos*.

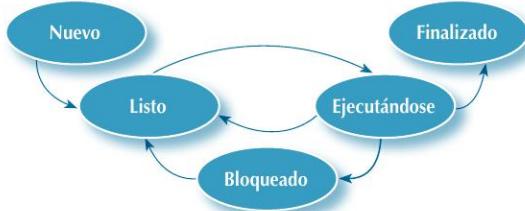
Los procesos pueden pasar por distintos estados. El planificador de procesos se encarga de establecer el estado de cada proceso y de modificarlo, atendiendo a un *algoritmo de planificación*. La mayoría de los algoritmos de planificación asignan un tiempo de ejecución o *quantum* a aquellos procesos que están ocupando la CPU (se están ejecutando). Pasado el *quantum*, se genera una interrupción de reloj, haciendo que el sistema operativo recupere el control. A continuación, el planificador toma el siguiente proceso de la cola de listos, según su algoritmo de planificación. Con esto se consigue favorecer la multitarea y evitar que se apoderen de la CPU aquellos procesos que necesitan ocupar mucho tiempo de CPU.

## TEN EN CUENTA

- ✓ Es de gran importancia conocer la política de planificación de procesos del sistema operativo, ya que establece su prioridad ante el tipo de proceso, beneficiándolo o perjudicándolo. No obstante, no existe una política de planificación ideal u óptima para cualquier proceso, ya que depende de las características de estos.

El ciclo de vida de un proceso establece los posibles estados, a saber:

- Nuevo: se crean las estructuras de datos para la gestión del nuevo proceso y se le asigna un espacio en memoria. La estructura más representativa es el *bloque de control de proceso* (PCB), donde se almacena la información necesaria para la gestión del mismo.
- Listo: el planificador de procesos determina que el proceso está preparado para pasar a ejecución. Los procesos en estado “listo” pueden ser aquellos que:
  - Se acaban de crear.
  - Han vuelto a la cola de listos al resolverse la causa de su bloqueo.
  - Han terminado el *quantum* de tiempo asignado para ejecutarse.
- Ejecutándose: el proceso se encuentra en el procesador ejecutándose, ocupando tiempo de procesamiento en la CPU. Es el planificador de procesos el que determina qué proceso de los presentes en la cola de listos es el siguiente para ser ejecutado, según el algoritmo de planificación.
- Bloqueado: si un proceso, que se encuentra en ejecutándose, se encuentra a la espera de un suceso, como el acceso a un recurso o la información derivada de una E/S (archivo, intervención del usuario, etc.), este se bloquea hasta que se disponga de dicha información.
- Finalizado: el proceso ha terminado de ejecutar las instrucciones. Se liberan todos los recursos asociados (archivos, buffers y referencias de memoria) y elimina las estructuras de datos para su gestión, como el PCB del proceso.



**Figura 4.12**  
Ciclo de vida de un proceso.

*A priori*, no se conoce el tiempo de ejecución de cada proceso, si se va a bloquear y durante cuánto tiempo o su permanencia en la cola de listos. Por tanto, el tiempo de ejecución y el estado de un proceso en un momento dado es impredecible.

Cuando un proceso cambia de estado o se produce un cambio de contexto (un proceso entra en ejecución mientras otro se retira de ella modificando la información contenida en las estructuras de datos de gestión del sistema operativo y actualizando los valores de la CPU con los datos del nuevo proceso), toda la información del estado del proceso se ha de actualizar en su PCB.

El procesador puede ejecutar instrucciones de dos modos:

- Modo usuario: normalmente es el empleado por los programas de usuario y las actividades no críticas del sistema operativo. El sistema operativo puede definir varios modos de usuario, diferenciados por privilegios.
- Modo kernel, núcleo o privilegiado: es el empleado principalmente por el núcleo del sistema operativo para ejecutar las instrucciones contenidas en él. En este modo se puede obtener el control total del procesador y, por tanto, del sistema. Tareas como la creación, terminación y sincronización de procesos, gestión y reserva del espacio de memoria, la gestión de las interrupciones, gestión de accesos al hardware, etc., se ejecutan en modo kernel. Si un programa, ejecutándose en modo usuario, desea realizar una operación crítica, debe solicitar dicha operación al sistema operativo, y que este la acepte y ejecute en modo kernel. La ejecución del modo núcleo, dependerá de la arquitectura del sistema operativo (monolítica, micronúcleo o híbrida).

Por otro lado, la ejecución de los procesos puede realizarse de varios modos:

1. Por lotes, de trabajos o batch: se lanza un conjunto de tareas para realizar por el sistema y este ejecuta todas ellas, una detrás de otra, sin intervención del usuario. Por ejemplo: la automatización de las copias de seguridad de un sistema, el lanzamiento de una serie de trabajos a la cola de impresión o la renderización de varias imágenes por lotes.
2. Interactivo: a diferencia de los anteriores, estos solicitan constantemente las acciones del usuario para su continuidad. El usuario realiza una acción mediante la ejecución de un comando o acción dentro de un programa y espera a que finalice para realizar otra acción o proceso interactivo. El uso de un programa de diseño por ordenador (CAD), el trabajo con una hoja de cálculo, un videojuego o navegar por Internet con un navegador web emplean un procesamiento interactivo.



#### SABÍAS QUE...

Existen procesos llamados *servicios* o *demonios*, que se caracterizan generalmente por comenzar automáticamente cuando se inicia el sistema y ejecutarse en segundo plano (el usuario no espera que finalice para interactuar con el sistema).

#### 4.4.2. Identificación y administración

Los procesos disponen de un identificador único llamado PID (IDentificador de Proceso). El PCB de cada proceso almacena información esencial, como:

- ✓ Identificación del proceso (PID).
- ✓ Identificación del proceso padre (PPID), es decir, el PID del proceso que lo creó.
- ✓ Usuario propietario.
- ✓ Valores del estado del procesador en el momento de producirse el cambio de contexto.
- ✓ Estado.
- ✓ Valores de referencia de memoria RAM.
- ✓ Ficheros abiertos.
- ✓ Buffers de memoria utilizados.

##### RECUERDA

- ✓ El *superusuario* o aquellos usuarios con potestad para administrar el sistema son los únicos que disponen de privilegios para administrar los procesos del sistema operativo y, por tanto, son los responsables de mantener un control del conjunto de procesos. No obstante, cada usuario puede gestionar sus propios procesos.

Gracias al comando *ps* podemos obtener información de los procesos del sistema. La potencia de este comando es enorme y, dada su gran variedad de opciones, recomendamos encarecidamente la lectura de su ayuda (*man ps*). Su sintaxis es la siguiente:

```
ps [modificadores]
```

Este comando puede emplear modificadores u opciones de versiones UNIX (precedidas por “-”), versión BSD (sin ser precedidas por “-”) y opciones largas precedidas con dos guiones propias de GNU. Dependiendo de estas opciones empleadas para un mismo modificador, la salida puede variar. Los modificadores o las agrupaciones más usados son los siguientes:

- Obtener información de todos los procesos del sistema

```
ps aux  
ps aux -- sort cputime (se ordenan por tiempo consumido de CPU)  
ps -ef
```

- Imprimir información junto a un árbol de procesos

```
ps axjf
```

- Seleccionar procesos por usuario, PID, terminal, etc.

```
ps -U luis -u luis u (muestra los procesos del usuario luis)  
ps -- pid 8943 (información del proceso con PID 8943)  
ps -t /dev/pts/2 (información de los procesos del terminal virtual "/dev/pts/2")
```

El comando *ps* puede mostrar la siguiente información establecida en su cabecera:

1. UID del usuario propietario del proceso.
2. PID del proceso.
3. PPID del proceso.
4. Índice de utilización reciente del procesador (columna titulada “C” y empleada para calcular la prioridad entre procesos).
5. Tiempo de inicio del proceso (Start Time o STIME).
6. Terminal de lanzamiento (TTY).
7. Tiempo de CPU consumido (TIME).
8. Orden de ejecución (COMMAND).
9. Tamaño del proceso en la memoria virtual en KB (VSZ).
10. Tamaño de la memoria residente del proceso (en memoria física) en KB (RSS).
11. Porcentaje de uso de CPU (%CPU).
12. Porcentaje de uso de memoria (%MEM).
13. Estado del procesador (STAT o S).

#### **CUADRO 4.6**

#### **Estados del procesador (STAT o S)**

Estado	Descripción
R	Ejecutándose o listo para ser ejecutado (Runnable)
S	Bloqueado o durmiendo (Sleeping)
T	Parado (Trace)
Z	Zombi (proceso que ha muerto, pero el proceso padre no ha reconocido su muerte)
I	Inactivo en creación (Idle)
N	Con prioridad menor de lo normal (NICE)
<	Con prioridad mayor de lo normal
+	Se encuentra en el grupo de procesos en primer plano
s	Proceso líder de sesión
l	Es un proceso multihilo (un mismo proceso con diferentes tareas que se pueden ejecutar en paralelo, evitando así el cambio de contexto)

Un uso muy extendido es filtrar el resultado de *ps*, concatenando el comando *grep*. De tal manera, que podemos realizar búsquedas de cadenas de texto sobre la salida. Como:

```
ps aux | grep bash
```

Además, para mostrar la estructura arborescente de procesos podemos utilizar el comando *pstree*.

Por otro lado, si en lugar de una instantánea de los procesos actuales del sistema (mediante el comando *ps*), necesitamos una actualización continua de la información de los procesos del

sistema, empleamos el comando *top*. Este comando permite visualizar el estado del sistema, mostrando información en tiempo real de los procesos. Para salir de *top*, pulsamos la tecla “q”.

El comando *top* muestra información genérica del sistema antes del detalle por proceso. Por líneas, la información es la siguiente:

- ✓ Línea 1.<sup>a</sup>: hora actual, tiempo del sistema encendido, número de usuarios y carga media en intervalos de 1, 5 y 15 minutos, respectivamente.
- ✓ Línea 2.<sup>a</sup>: número de tareas, número de procesos en estado, ejecutándose o listos (R), bloqueados o hibernando (S), parados (T) y zombis (Z) respectivamente.
- ✓ Línea 3.<sup>a</sup>: tiempos de CPU de usuario, del kernel del sistema, etc.
- ✓ Línea 4.<sup>a</sup>: tamaño en MB de memoria física en total, libre, usada y utilizada por buffer.
- ✓ Línea 5.<sup>a</sup>: tamaño en MB de memoria virtual total, libre usada y disponible.

El resto de líneas muestran la información de los procesos, identificando las columnas mediante cabeceras similares a las del comando *ps*.

También podemos modificar el tiempo de actualización de la información mostrada en segundos (por defecto, tres segundos), monitorizar determinados procesos por PID o usuarios:

- Mostrar los procesos del usuario *luis* actualizando la información cada dos segundos: *top -d 2 -u luis*
- Mostrar información del proceso con PID 8933: *top -p 8933*



### Actividades propuestas

- 4.15.** Práctica con las distintas opciones de *ps* aquí estudiadas para recoger diferente información de los procesos del sistema. Analiza la información según la cabecera.
- 4.16.** Estudia los procesos en tiempo real del sistema. Filtra la información en tiempo real de los procesos por usuarios o por PID. Analiza la información según la cabecera en cada caso.

#### A) Primer y segundo plano

Cuando un usuario ejecuta un comando desde el *shell*, este crea una *subshell* que ejecuta dicho comando. El usuario ha de esperar para recuperar el control del intérprete de comandos durante el tiempo que pasa desde que el programa es lanzado hasta que termina, volviendo a mostrar el *prompt*. Los comandos que se ejecutan de esta forma, se dice que lo hacen en primer plano o en *foreground*.

Existe una alternativa para evitar que el propio usuario tenga que esperar a la terminación de una tarea para poder continuar con la ejecución de otras nuevas, denominada *ejecución en segundo plano* o *background*. Consiste en añadir al final de la línea de mandatos que ejecutar en el *shell*, el símbolo “&”. De esta manera, el *prompt* se devolverá inmediatamente, sin esperar la terminación de la tarea recién lanzada.

Para probar la ejecución en *foreground* y en *background*, lo haremos con el comando *yes*, el cual imprime indefinidamente el carácter “y”. Para evitar que se llene la pantalla, redirigimos la salida a la *cubeta de bits* (*/dev/null*): *yes > /dev/null*.

Si lo lanzamos en primer plano, no devuelve el *prompt* y hemos de teclear *CTRL+C* para matar el proceso, recuperando así el control del terminal.

Si lo lanzamos en segundo plano, devuelve el prompt al añadir el símbolo “&” al final de la línea, indicando la ejecución en *background*. Se puede observar cómo devuelve el número de tarea (entre corchetes) y el PID del proceso.

**Figura 4.13**  
Ejecución en segundo plano.

```
luis@luis-VirtualBox:~$ yes > /dev/null &
[1] 9957
luis@luis-VirtualBox:~$
```

Con el comando *jobs* podemos identificar las tareas que se hallan en segundo plano. Este identifica la tarea más reciente con el símbolo “+” y el segundo más reciente con “-”.

Además, se pueden pasar procesos de *foreground* a *background*, y viceversa, mediante los siguientes comandos:

- *fg [%][tarea]* pasa una tarea a primer plano. Sin argumentos, la tarea más reciente.
- *bg [%] [tarea]* pasa una tarea a segundo plano. Sin argumentos, la tarea más reciente.

Para pasar una tarea de primer plano a segundo plano, hemos de detenerla mediante la combinación de teclas *CTRL+Z* y, a continuación, lanzarla en *background* con *bg*.

Para pasar una tarea de segundo plano a primer plano, ejecutamos el comando *fg*.

**Figura 4.14**  
Paso de proceso de primer a segundo plano.

```
luis@luis-VirtualBox:~$ yes > /dev/null
^Z
[2]+  Detenido                  yes > /dev/null
luis@luis-VirtualBox:~$ jobs
[1]-  Ejecutando                yes > /dev/null &
[2]+  Detenido                  yes > /dev/null
luis@luis-VirtualBox:~$ bg %2
[2]+ yes > /dev/null &
luis@luis-VirtualBox:~$ jobs
[1]-  Ejecutando                yes > /dev/null &
[2]+  Ejecutando                yes > /dev/null &
luis@luis-VirtualBox:~$
```

**Figura 4.15**  
Paso de proceso de segundo a primer plano.

#### Actividad propuesta 4.17



Empleando el comando *sleep* (el cual bloquea la ejecución durante un tiempo expresado por defecto en segundos, como, por ejemplo: *sleep 12*), suspende la devolución del *prompt* del shell durante 50 segundos. Pasa esa tarea a segundo plano y después a primer plano, antes de que finalice.



### B) Prioridad de las órdenes

El algoritmo de planificación de Linux que determina el orden de ejecución entre los procesos en la cola de listos, emplea una mezcla de algoritmos como *Round Robin*, *FIFO*, *prioridades*, etc. La prioridad real de cada proceso se calcula de manera compleja mediante varios factores, la cual se puede alterar relativamente mediante un índice denominado *nice*.

El valor *nice* de un proceso oscila entre -20 (máxima prioridad) y 19 (menor prioridad), y puede ser modificada por el propietario del proceso o el *superusuario*. Por defecto, un usuario solo puede disminuir la prioridad de sus procesos cuando los lanza. Es una manera de ser "amable" con el sistema, beneficiando al resto de procesos.

Cuando se ejecuta un proceso, este lo hace predeterminadamente con una prioridad relativa 0 (valor *nice*). Podemos comprobarlo con la orden *ps -l*, donde lo indican las columnas NI (NIce) y PRJ (por defecto con valor 80).

Podemos lanzar procesos, modificando su prioridad relativa con *nice*, siendo su sintaxis:

```
sudo nice [-[n]] {+|-} num_nice orden
```

Una vez lanzado un proceso, podemos modificar su prioridad relativa mediante la orden *renice*. Sin embargo, un usuario común no puede aumentar la prioridad, solo decrementarla. Su sintaxis es:

```
sudo renice prioridad [[-p] PID's]] [[-u] usuarios]
```

Donde se puede modificar la prioridad por PID (por defecto), por usuarios o ambas.



#### Actividad propuesta 4.18

Lanza el proceso *yes > /dev/null* en segundo plano con prioridad +10 con un usuario común. Intenta aumentar la prioridad, ¿es posible? Realiza el mismo procedimiento como *superusuario*.

### C) Envío de señales

Los procesos reciben señales para ser controlados desde el propio sistema operativo y desde el exterior. Un usuario también puede enviar señales a los procesos mediante la orden *kill*. Su sintaxis es:

```
kill -señal PID
```

Donde las señales más utilizadas son:

- 2 o SIGINT: interrumpe un proceso, similar a CTRL+C. Esta señal puede ser manejada por el propio proceso, aunque no es lo habitual, terminando su ejecución.
- 9 o SIGKILL: mata un proceso.
- 15 o SIGTERM: mata un proceso, aunque esta señal puede ser ignorada en determinados casos, por lo que SIGKILL sería más efectiva.
- 18 o SIGCONT: continúa la ejecución de un proceso. Similar a CTRL+Z la segunda vez que se la lanza para reanudar un proceso.
- 19 o SIGSTOP: pausa la ejecución de un proceso. Similar a CTRL+Z la primera vez que es lanzada para detener un proceso.

```
luis@luis-VirtualBox:~$ sleep 8000 &
[1] 12532
luis@luis-VirtualBox:~$ yes > /dev/null &
[2] 12533
luis@luis-VirtualBox:~$ ps -l -pid 12533 12532
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN TTY      TIME CMD
0 S 1000 12532 12374 0  80  0 - 4164 hrtime pts/2          0:00 sleep 8000
0 R 1000 12533 12374 32  80  0 - 4164 -                pts/z        1:52 yes
luis@luis-VirtualBox:~$ kill -2 12532
luis@luis-VirtualBox:~$ kill -SIGSTOP 12533
[1]+  Detenido  yes > /dev/null
luis@luis-VirtualBox:~$ ps -l -pid 12533 12532
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN TTY      TIME CMD
0 S 1000 12533 12374 31  80  0 - 4164 do_sig pts/2        2:06 yes
[2]+  Detenido  yes > /dev/null
luis@luis-VirtualBox:~$ kill -SIGCONT 12533
luis@luis-VirtualBox:~$ ps -l -pid 12533 12532
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN TTY      TIME CMD
0 R 1000 12532 12374 29  80  0 - 4164 hrtime -          2:07 yes
luis@luis-VirtualBox:~$ kill -9 12533
luis@luis-VirtualBox:~$ ps -l -pid 12533 12532
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN TTY      TIME CMD
[2]+  Terminado (killed)  yes > /dev/null
luis@luis-VirtualBox:~$ ps -l -pid 12533 12532
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN TTY      TIME CMD
luis@luis-VirtualBox:~$
```

**Figura 4.17**  
Envío de señales a procesos.

Son muchas las utilidades del envío de señales por parte de los usuarios, como, por ejemplo, detener procesos que consuman muchos recursos, interrumpir procesos que bloqueen otras tareas, matar procesos, etc. La combinación de teclas CTRL+C solo se puede enviar a procesos en primer plano, sin embargo, las señales mediante *kill* se pueden enviar desde diferentes orígenes, como otros terminales. En el ejemplo de la figura 4.17 se observa cómo se lanzan señales, tanto su valor numérico como textual, y cómo cambia su estado.

#### Actividad propuesta 4.19



Lanza varios procesos similares a los del ejemplo anterior, tratando de enviarles diferentes señales de terminación, pausa y continuación, comprobando su estado tras cada una.

## 4.5. Gestión de procesos por interfaz gráfica en Windows

En Microsoft Windows la planificación de procesos se basa en la utilización de colas múltiples por prioridades. Estas colas emplean diferentes algoritmos de planificación; uno de ellos es *Round-Robin*.

Para la gestión de procesos en Windows, se emplea el ‘Administrador de tareas’. En la pestaña ‘Procesos’, podemos analizar todos los procesos en ejecución del sistema agrupados por ‘Aplicaciones’, ‘Procesos en segundo plano’ y ‘Procesos de Windows’.

Pulsando en el botón derecho del ratón sobre una aplicación, podemos ver los detalles o finalizar su ejecución, entre otras opciones.

Además, la pestaña “Detalles” pormenoriza cada uno de ellos y ofrece muchas más opciones como establecer su prioridad o establecer una afinidad para con CPU.



## Actividad propuesta 4.20

Accede al ‘Administrador de tareas’ de Microsoft Windows y observa las aplicaciones en ejecución. Observa los recursos consumidos por todos los procesos. ¿Cuál es el proceso que consume más memoria? Abre un procesador de textos y finaliza su ejecución.

## 4.6. Automatización de tareas en Linux

La automatización de tareas en Linux se lleva a cabo mediante la ejecución de tareas planificadas para un momento puntual o para el lanzamiento recurrente de tareas. Ambas acciones son diferentes, ya que la primera no implica repetición y la segunda sí.

Para lanzar una tarea o conjunto de ellas de forma aislada en el tiempo se puede emplear el comando *at*. Podemos instalar *at* mediante: *sudo apt install at*. Este permite lanzar órdenes en una fecha y a una hora concreta. Su sintaxis es la siguiente:

```
at hora [fecha] [-f fichero]
```

El comando *at* facilita la introducción de tareas, devolviendo el *prompt PS2*. Una vez indicadas, se teclea *CTRL+D* para guardar la tarea. En su lugar, se pueden indicar las tareas que se van a lanzar mediante un fichero con el modificador *-f*. Es recomendable leer la ayuda (*man at*) para estudiar las distintas opciones. Un ejemplo es el que se muestra en la figura 4.18.

La orden *at -l* muestra los procesos listos para ejecutarse mediante *at*. De todos ellos, se puede eliminar la planificación de ejecución de cualesquiera de ellos mediante: *at -d num\_tarea*. Y, además, podemos ver el desglose de subtareas de una tarea programada con: *at -c num\_tarea*.

Una alternativa al comando *at* para trabajos por lotes es *batch*. Con *batch*, es el propio núcleo el que decide el momento adecuado para la ejecución de las tareas. Los trabajos se lanzarán de manera inmediata, pero en caso de que la carga del sistema sea excesiva (por defecto, igual o superior a 1,5), retrasará la ejecución de las tareas. Podemos ver la carga media del sistema en la primera línea del comando *top* (*load average*). Por tanto, el uso de *batch* es similar a *at*, siendo más adecuada para una gestión razonable de la carga de trabajos sobre el sistema para procesos que necesitan ocupar durante largos períodos de tiempo la CPU y con pocas E/S.

La forma de uso más común con *batch* es: *batch < tareas*, donde *tareas* incluye un conjunto de órdenes en un archivo de texto. Tal y como se muestra en el ejemplo de la figura 4.19.

```
luis@luis-VirtualBox:~$ at 17:00 05/15/2019
Warning: commands will be executed using /bin/sh
at> echo "Reunión de empresa" > ~/reunión.txt
at> <EOF>
job 7 at Wed May 15 17:00:00 2019
luis@luis-VirtualBox:~$ at now + 5 minutes
Warning: commands will be executed using /bin/sh
at> echo "Llamar al jefe" > /dev/pts/1
at> <EOF>
job 8 at Fri May 10 23:57:00 2019
luis@luis-VirtualBox:~$ at -l
7    Wed May 15 17:00:00 2019 a luis
6    Fri May 14 23:43:00 2900 a luis
8    Fri May 10 23:57:00 2019 a luis
4    Sat May 11 10:30:00 2019 a luis
```

**Figura 4.18**  
Ejemplos de planificación de tareas con *at*.

```
luis@luis-VirtualBox:~$ cat > tareas
echo "Contactar con el jefe" > /dev/pts/0
touch reunión.txt
luis@luis-VirtualBox:~$ batch < tareas
warning: command will be executed using /bin/sh
job 14 at Wed May 15 08:53:00 2019
luis@luis-VirtualBox:~$ 
luis@luis-VirtualBox:~$ Contactar con el jefe
ls -l reunión.txt
-rw-r--r-- 1 luis luis 0 may 15 08:53 reunión.txt
luis@luis-VirtualBox:~$
```

**Figura 4.19**  
Ejemplos de planificación de tareas con *batch*.

En Linux también se pueden planificar tareas de manera periódica o recurrente gracias a *cron*. Esta utilidad es un demonio que permite lanzar órdenes o scripts definidos en un archivo *crontab* (*cron table*) cuya sintaxis interna ha de establecerse de forma estructurada y donde se especifica su patrón de repetición.

El archivo */etc/crontab* se define para el conjunto de acciones planificadas periódicamente sobre todo el sistema sobre el que el *superusuario* es el único capaz de editarla y el resto de usuarios solo pueden leerla. No obstante, cada usuario puede tener su propio archivo *crontab*.

Para la planificación recurrente con *cron* podemos:

1. Comprobar si *cron* está instalado mediante *dpkg -l cron*.
2. Verificar la actividad del demonio *cron* mediante: *systemctl status cron*. En caso negativo, podemos iniciarla mediante *systemctl start cron*.
3. Reiniciar el demonio mediante *systemctl restart cron*.

Los ficheros *crontab* han de especificar las órdenes y su periodicidad mediante los siguientes valores separados por espacios:

Minutos [0..59] Hora [0..23] Día [1..31] Mes [1..12] Dia de la semana [0..6] orden

Si un campo se ignora, se especifica mediante “\*”, indicando cualquier valor válido. Se pueden especificar listas mediante comas y sin espacios, mediante sus valores mínimos y máximos separados por “-”, y también mediante un valor de inicio y un valor incremental, separados por “/”.

### Ejemplo

*Ejemplos de ejecuciones periódicas con crontab:*

- *03 \* \* \* \* orden*  
Ejecuta *orden* en el minuto 3 de cada hora y cada día de todos los meses y para todos los días de la semana.
- *00 \* \* \* 0 orden*  
Ejecuta *orden* cada hora en punto, de todos los días del mes que sean domingos.
- *3/3 2/3 4 4 4 orden*  
Ejecuta *orden* cada tres minutos empezando por el minuto 3 de las horas 2,5,8, etc. del día 4 de abril y que sea jueves.
- *30 18 20 1-6 \* orden*  
Ejecuta *orden* a las 18:30 horas del día 20 de cada mes desde enero a junio.
- *45 06 1/2 \* 1,3,5 orden*  
Ejecuta *orden* a las 06:45 horas de los días impares de cada mes, siempre que sean lunes, miércoles o viernes.
- *\*/15 10-14 \* \* 6 orden*  
Ejecuta *orden* cada 15 minutos de 10 a 14 horas todos los sábados.
- *10,30,45 08 \*/2 \* \* orden*  
Ejecuta *orden* a las 08:10, 08:30 y 08:45 cada dos días y cada mes.

---

Para editar un fichero *crontab*, lo más conveniente es mediante la orden *crontab -e*. Ello abrirá un editor donde podremos definir las órdenes ajustándolas a la sintaxis *crontab* para el usuario

en cuestión. También podemos visualizar el contenido del archivo *contrab* del usuario mediante *crontab -l* o eliminar la orden con *crontab -r*.

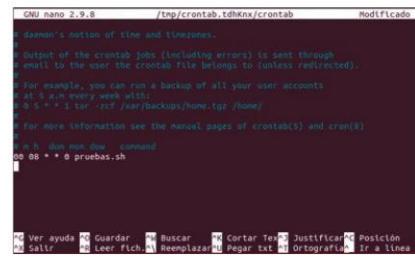
```
luis@luis-VirtualBox:~$ crontab -e
crontabs installing new crontab
luis@luis-VirtualBox:~$ crontab -l
# Edit this file to introduce tasks to be run by cron.

# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task

# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezone.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).

# For example, you can run a backup of all your user accounts
# at 5 a.m. every week with:
# 0 5 * * * tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8).
#
# m h dom mon dow   command
# 0 0 * * * pruebas.sh
luis@luis-VirtualBox:~$ crontab -r
luis@luis-VirtualBox:~$ crontab -l
no crontab for luis
```

**Figura 4.20**  
Ejemplo de órdenes con crontab.



**Figura 4.21**  
Edición del archivo predefinido crontab con crontab -e.

No obstante, podemos crear nuestro propio archivo *crontab* y lanzarlo, redirigiéndolo al comando *crontab*. Una vez editado el archivo *crontab*, hemos de actualizar el demonio *cron* para que tome los últimos cambios.

```
luis@luis-VirtualBox:~$ cat > luis.cron
00 08 * * 0 pruebas.sh
luis@luis-VirtualBox:~$ crontab < luis.cron
luis@luis-VirtualBox:~$ crontab -l
00 08 * * 0 pruebas.sh
```

**Figura 4.22**  
Ejemplo de ejecución de un nuevo archivo crontab.

## 4.7. Monitorización y gestión del sistema. Evaluación de prestaciones

Los administradores del sistema, entre sus labores, deben obtener información del rendimiento del sistema para así detectar o anteponerse a futuros problemas, a la vez que establecer mejoras en el desempeño del mismo. Para ello, tanto el sistema operativo como programas de terceros ofrecen información que ayuda a realizar estas acciones.

Ya conocemos algunos comandos que ofrecen información muy valiosa sobre el rendimiento del sistema, como *top* o *ps*. Además, se emplean otros programas muy utilizados.

El comando *uptime* muestra la hora del sistema, el tiempo del sistema en activo, el número de usuarios y la carga del sistema en intervalos de 1, 5 y 15 minutos, respectivamente. Equivale a la primera línea de la orden *top*.

También podemos obtener información sobre el espacio libre y usado de la memoria real y física con *free*.

Si deseamos más detalle sobre los usos de memoria, la herramienta *vmstat* devuelve información de la memoria RAM, memoria virtual, intercambios entre memoria RAM y disco, interrupciones y el procesador. Su sintaxis es: *vmstat [tiempo [actualizaciones]]*, donde:

- ✓ tiempo: es el tiempo en segundos transcurrido entre dos actualizaciones.
- ✓ actualizaciones: es el número de muestras. Como, por ejemplo: *vmstat 3 5*.

También es importante el uso de *df*, que muestra el porcentaje de uso de las unidades de almacenamiento del sistema.

Con respecto a los usuarios, el comando *w* muestra quién está conectado y qué está haciendo (parecido a *who*).

Por otro lado, en Microsoft Windows, a través del 'Administrador de tareas', podemos hacer un estudio preciso del rendimiento del sistema en la pestaña 'Rendimiento'. Podemos analizar el uso de CPU, cantidad de procesos y subprocesos activos, estado de la memoria RAM (en uso, disponible, espacio paginado), usos de discos, tráfico de los adaptadores de red, etc.

También permite analizar los porcentajes de CPU, memoria, disco y red empleados por cada usuario y un desglose por aplicaciones de cada usuario en la pestaña 'Usuarios'.

Además, podemos estudiar las aplicaciones en segundo plano y los servicios de Windows en las pestañas 'Detalles' y 'Servicios'.

#### Actividades propuestas



- 4.21.** Practica con todas las herramientas de rendimiento y estadísticas especificadas en este apartado. En caso de duda, lee los manuales de ayuda.
- 4.22.** Practica y analiza las pestañas 'Rendimiento', 'Usuarios', 'Detalles' y 'Servicios' del 'Administrador de tareas' de Microsoft Windows 10.

## 4.8. Aplicaciones para el mantenimiento y optimización del sistema

Existen aplicaciones propias del sistema operativo o de terceros que aumentan la funcionalidad, mejoran la manejabilidad o incrementan la seguridad del sistema. Existen, por tanto, multitud de herramientas que ayudan al mantenimiento y la optimización del sistema. Podemos clasificarlas en:

- a) Aplicaciones de actualización y control de drivers. Aunque el propio sistema operativo se encarga de actualizar los drivers del hardware del sistema, a veces se pueden dar situaciones de incompatibilidad entre una actualización del sistema operativo y un driver, dejando en estado inconsistente el hardware. Por tanto, puede ser adecuado una aplicación que advierta sobre actualizaciones de los drivers del sistema. Algunas de ellas son *Drivers Cloud* y *Driver Booster*.
- b) Aplicaciones de sincronización, copias de seguridad e imágenes del sistema. Existe una gran variedad de aplicaciones que ayudan a automatizar el proceso de copias de seguridad. Muchas de ellas disponen de una larga experiencia y, por tanto, tienen un gran número de seguidores, como *EaseUS Todo Backup Free* (realizar backups), *Clonezilla* (gran fiabilidad para realizar imágenes del sistema), *FreeFileSync* (sincronizar elementos).
- c) Antivirus. Es por todos conocida la importancia de disponer de aplicaciones antimalware de terceros, con una gran experiencia en la detección y eliminación de todo tipo de software malicioso. Empresas como *Avira*, *Panda*, *Bitdefender*, *Kaspersky*, *AVG* o *Avast* ofrecen gran variedad de software antispyware, firewall, antimalware, antipop-ups, etc.
- d) Optimización del sistema. Una explotación eficiente del sistema operativo implica un mantenimiento cuidado con la descarga de archivos, instalación y desinstalación de programas, control de carpetas temporales, etc. Algunos programas que facilitan estas tareas son *CCleaner* (Microsoft Windows) y *Stacer* (Linux).



### Actividad propuesta 4.23

Instala, prueba y estudia al menos una aplicación correspondiente a cada apartado de esta clasificación.

## Resumen

- Los sistemas operativos modernos ofrecen herramientas para la administración de procesos y usuarios. En este capítulo hemos trabajado tanto desde el punto de vista gráfico (con Microsoft Windows) como textual (con Ubuntu), aunque se ha hecho hincapié en este último, dada su gran potencia y versatilidad.
- Los comandos más importantes tratados para la gestión de usuarios en Ubuntu han sido:

```
useradd      userdel  chfn    who     su      sudo   groupadd  groupdel  
groupmod    groups   adduser  deluser usermod  passwd  chage
```

- Estas órdenes permiten editar archivos de configuración, tales como:

```
/etc/passwd      /etc/skel        /etc/group        /etc/shadow
```

- Por otro lado, hemos estudiado la edición de permisos mediante los comandos:

```
chmod          umask        chown        chgrp
```

- Además, hemos trabajado con un conjunto de herramientas que ayudan a administrar la gestión de procesos en Ubuntu:

```
kill           nice       renice      ps       pstree      fg  
top            jobs       at         batch     crontab     bg
```

- Desde el punto de vista de la administración gráfica con Microsoft Windows 10, para la gestión de usuarios y procesos hemos trabajado a través de 'Cuentas de usuario' y del 'Administrador de tareas', respectivamente. No obstante, siempre es posible una administración por línea de comandos en Windows.
- Durante todo el capítulo se ha destacado la distinción entre usuarios comunes y usuarios administradores con privilegios para gestionar el sistema. Las figuras del *superusuario* en Ubuntu (*root*) y el *Administrador* en Windows se encuentran deshabilitadas por seguridad.
- Por último, hemos destacado la importancia de algunas aplicaciones que aumentan la funcionalidad del sistema, mejoran su manejabilidad y la seguridad.



## Ejercicios propuestos

### 1. Gestión de usuarios en Linux:

- a) Crea los grupos *SI1* y *SI2*.
- b) Crea el usuario *usuario0* cuyo home sea */home/usuario\_cero*, copiando los archivos de configuración de */etc/skel*. Comprueba a cuántos grupos pertenece.
- c) Localiza la línea de dicho usuario en el fichero */etc/passwd* para comprobar sus datos.
- d) Elimina el usuario *usuario0* y su carpeta *home*.
- e) Crea los usuarios *usuario1* y *usuario2*. Ambos deben pertenecer únicamente al grupo *SI1* como principal. Comprueba su pertenencia a dicho grupo.
- f) Crea los usuarios *usuario3* y *usuario4*, perteneciendo como grupo principal a *SI2*.
- g) Modifica el *shell* por defecto del usuario3 a */bin/sh*.
- h) Intenta eliminar *SI1*.

### 2. Seguridad en cuentas de usuarios en Linux:

- a) Suministra la contraseña *sistEmas\_%20* a todos los usuarios antes creados.
- b) Deshabilita la cuenta de *usuario3* y bloquea la contraseña de *usuario4*. Compruébalo.
- c) Habilita la cuenta de *usuario3* y desbloquea la contraseña de *usuario4*. Compruébalo.
- d) Establece 20 días como máximo y 10 días como mínimo para cambiar la contraseña de *usuario2*. Compruébalo.
- e) Establece una fecha para *usuario4*, a partir de la cual la cuenta caducará y será inaccesible. Compruébalo. Elimina la expiración de la cuenta y compruébalo.

### 3. Gestión de recursos y permisos en Linux y Windows:

En Linux:

- a) En tu home, crea la carpeta *dirPerm* y, dentro de ella, un archivo llamado *permisos*.
- b) Emplea la notación octal para modificar los permisos de *dirPerm* a *rwxr-----*. Y la notación simbólica para deshabilitar para el grupo el permiso de lectura sobre el archivo *permisos*. Compruébalo.
- c) Emplea la notación octal para modificar los permisos de *dirPerm* a *rwxrwxrw-*. Y la notación simbólica para habilitar todos los permisos para el propietario y el grupo, y deshabilitar todos los permisos al resto de usuarios sobre el archivo *permisos*. Compruébalo.
- d) Cambia el propietario y grupo de *dirPerm* a *usuario1* y *SI1*, respectivamente, afectando a su contenido.

En Microsoft Windows:

- a) Crea dos usuarios, *usuario1* y *usuario2*, y dos grupos de usuarios, *Administrativos* e *Informáticos*. Asigna cada usuario a un grupo. Compruébalo.
- b) Establece contraseñas a ambos usuarios.
- c) Habilita el usuario Administrador.
- d) Deshabilita el usuario *usuario1*. Elimina *usuario2*.
- e) Monitoriza los procesos y estudia el rendimiento del sistema.

**4. Acceso de usuarios en Linux:**

- a) Desde el terminal, ejecuta una sesión como *usuario1*. Lista el contenido de su home. Muestra el valor de la variable PWD. Sal de dicha sesión.
- b) Accede desde varios terminales con diferentes usuarios. Desde cualquiera de ellos, muestra los usuarios que se encuentran conectados.

**5. Permisos especiales en Linux:**

- a) Crea un directorio llamado *dirCompartido* en el que todos los usuarios del grupo *SI2* puedan escribir para guardar sus archivos o subcarpetas. Ahora modifica los permisos del directorio para que solo el propietario de cada objeto pueda borrar sus propios archivos.
- b) Ahora modifica los permisos de *dirCompartido* para que los archivos y subdirectorios creados en su interior sean forzados a pertenecer al grupo del directorio y no al grupo del usuario que lo haya creado.

**6. Máscara de permisos en Linux:**

- a) Para *usuario1*, modifica la máscara de permisos por defecto para que los ficheros creados tengan todos los permisos activos. Comprueba si es posible.
- b) Configura la máscara de permisos por defecto para que en los nuevos ficheros el propietario tenga permiso de lectura solo, y los demás no tengan ningún permiso. Sal de la sesión de *usuario1* y comprueba si se ha guardado la máscara de permisos. ¿Podríamos hacerla permanente?

**7. Superusuario en Linux:**

¿Por qué se encuentra la cuenta root deshabilitada por defecto? ¿Cómo se podría habilitar?

**8. Alias y variables en Linux:**

- a) Muestra, a través de variables el directorio de trabajo actual, el nombre del equipo, el shell y el login del usuario actual.
- b) Crea un alias que permita crear nuevos ficheros con la cadena *cfch*. Compruébalo y elimínalo.
- c) Crea una variable global llamada *SISTEMAS* con valor *SI\_20*. Compruébalo.
- d) Haz permanente la variable global *SISTEMAS*.

**9. Procesos en Linux:**

- a) Muestra todos los procesos del sistema.
- b) Muestra los procesos del usuario actual.
- c) Desde un terminal, lanza el proceso *yes > /dev/null*. Pásalo a segundo plano. Elimina el proceso.
- d) Desde un terminal, lanza el proceso *yes > /dev/null*. Desde otro terminal, localiza el proceso. Comprueba su estado tras el envío de cada señal.
- e) Lanza el proceso *sleep 500* en segundo plano y con mínima prioridad. Intenta subir la prioridad. Sube la prioridad al máximo del proceso con *root*. Disminuye la prioridad del proceso. Comprueba su prioridad tras cada modificación.

**10. Automatización de tareas en Linux:**

- a) Crea el archivo de texto *planificado.txt* dentro de 2 minutos.
- b) Crea los siguientes archivos de texto:
  - planificado [hora]\_[fecha].txt, todos los lunes y jueves a las 15 horas.*
  - planificado2 [hora]\_[fecha].txt, todos los viernes de los meses pares a las 4 am.*
- c) Monitorización del sistema en Linux. Realiza un estudio detallado del sistema con los comandos *uptime, top, free, vmstat, df y w*.

**ACTIVIDADES DE AUTOEVALUACIÓN**

1. En sistemas GNU/Linux, un usuario administrador es:

- a) El que corresponde con el usuario *root*.
- b) El *superusuario*.
- c) Aquel que tiene suficientes privilegios para ejecutar determinados archivos o comandos con capacidad de gestión sobre el sistema.

2. ¿Es recomendable habilitar el *superusuario* en sistemas Linux?:

- a) Sí, siempre, para tener un control total sobre el sistema.
- b) No, por seguridad, evitando correr riesgos innecesarios y supliendo sus acciones mediante el comando *sudo* o la incorporación de usuarios en grupos administradores.
- c) Sí, aunque no se vaya a usar, ya que permite disponer de él en cualquier momento.

3. ¿Qué comandos pueden deshabilitar una cuenta de usuario?:

- a) *chage* y *useradd*.
- b) *chage* y *usermod*.
- c) *su* y *chage*.

4. ¿Qué permiso especial sobre la máscara de permisos de un directorio permite que solo el propietario de un archivo dentro de él o el propietario de dicho directorio puedan eliminar o modificar su contenido?:

- a) Set-uid.
- b) Set-gid.
- c) Sticky-bit.

5. ¿Qué comando permite listar variables locales?:

- a) *env*.
- b) *printenv*.
- c) *set*.

6. En cuanto a la gestión de procesos:

- a) *A priori*, se conoce el tiempo de ejecución de cada proceso, sus bloqueos y permanencia en los distintos estados.
- b) El PCB almacena la información necesaria para la gestión de un proceso.
- c) El procesador puede ejecutar instrucciones en dos modos distintos: modo kernel y modo privilegiado.

7. El procesador suele ejecutar las instrucciones de una aplicación ofimática en:

- a) Modo Usuario.
- b) Modo Kernel.
- c) Ambos modos.

8. ¿En qué estado se encuentra un proceso, si el comando *ps* indica, en la columna STAT asociada al mismo, el valor *T*?:

- a) Parado.
- b) Zombi.
- c) Bloqueado.

9. ¿Es posible aumentar la prioridad de un proceso lanzado por un usuario estándar?:

- a) Sí.
- b) No.
- c) Sí, siempre que el proceso se encuentre en segundo plano.

10. ¿Qué comando es el más adecuado, atendiendo a las necesidades del sistema, para el lanzamiento planificado de tareas de forma no recurrente?:

- a) crontab.
- b) at.
- c) batch.

#### SOLUCIONES:

1.  a  b  c

2.  a  b  c

3.  a  b  c

4.  a  b  c

5.  a  b  c

6.  a  b  c

7.  a  b  c

8.  a  b  c

9.  a  b  c

10.  a  b  c

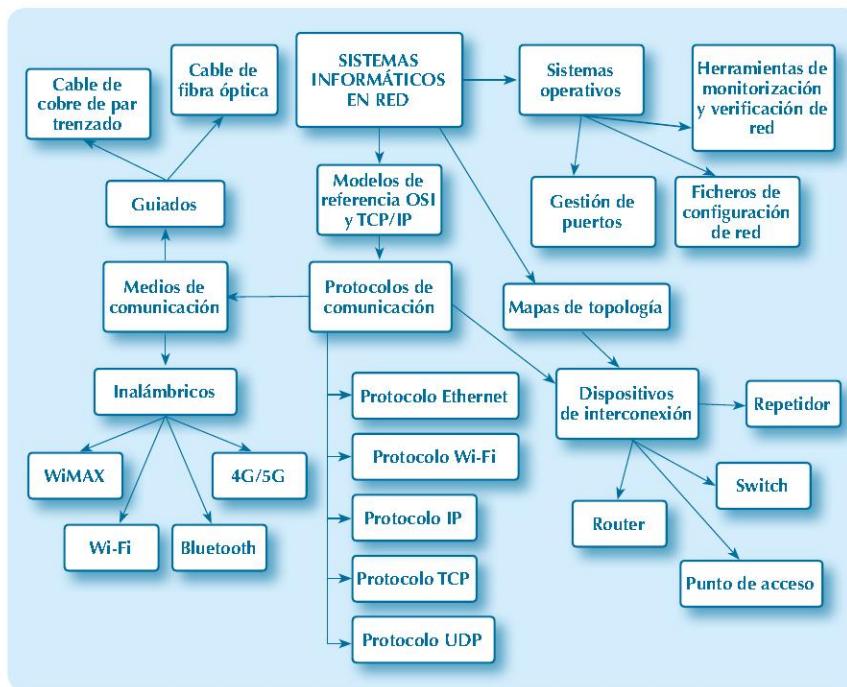
# 5

## Sistemas informáticos en red. Configuración y explotación

### Objetivos

- ✓ Identificar los tipos de redes, sus componentes y sistemas de interconexión.
- ✓ Interpretar mapas físicos y lógicos de una red informática.
- ✓ Conocer los modelos de referencia OSI y TCP/IP y su descomposición en varios niveles para entender su aplicación en los protocolos de red más importantes.
- ✓ Descubrir los protocolos de red más importantes.
- ✓ Saber configurar el protocolo TCP/IP y el acceso a redes de área extensa.
- ✓ Emplear comandos y herramientas para monitorizar y verificar el funcionamiento de la red.
- ✓ Aplicar distintas configuraciones para establecer redes de área local cableadas e inalámbricas.
- ✓ Poder gestionar puertos de comunicaciones.
- ✓ Proteger la transmisión de datos mediante protocolos seguros de comunicaciones.

### Mapa conceptual



### Glosario

**Dirección IP.** Dirección lógica asociada a una interfaz de red para su uso con el protocolo IP.

**Dirección MAC.** Dirección física asociada a una interfaz de red.

**FTTH.** Tecnología de conexión que emplea fibra óptica desde la red troncal hasta el hogar del cliente.

**IANA (Internet Assigned Numbers Authority).** Entidad encargada de gestionar a nivel mundial la asignación de direcciones IP públicas.

**IEEE 802.11.** Familia de estándares Wi-Fi.

**IEEE 802.15.** Grupo de trabajo de redes WPAN, como Zigbee (IEEE 802.15.4) y Bluetooth (IEEE 802.15.1).

**IEEE 802.16.** Familia de estándares WiMAX.

**IP.** Protocolo de Internet con dos versiones en uso (IPv4 e IPv6).

**Protocolo UDP.** Protocolo de datagramas de usuario orientado a enviar segmentos entre aplicaciones de manera rápida, sin importar su confiabilidad.

**Protocolo TCP.** Protocolo de control de transmisión que garantiza que todos los segmentos lleguen al destino.

**Puerta de enlace predeterminada (Gateway por defecto).** Elemento de conexión intermedio a través del cual se accede a una red remota.

**Socket.** Combinación de una dirección IP y un puerto lógico.

**TIA/EIA-568-B.** Norma que regula, entre otros aspectos, los mapas de conexión de los cables de cobre de par trenzado con el conector RJ-45.

**VPN.** Red privada virtual.

## 5.1. Introducción

Los sistemas informáticos se comunican y comparten información gracias a los sistemas en red. Estos sistemas informáticos en red se fundamentan en modelos de referencia que establecen las características y especificaciones necesarias para poder comunicarse entre diferentes entidades con objeto de intercambiar información. Los modelos de referencia más utilizados son el *modelo de referencia OSI* y el *modelo de referencia TCP/IP*.

Estos modelos descomponen sus funciones en varios niveles para definir protocolos y estándares, reducir la complejidad, controlar los flujos de comunicación y facilitar su evolución. De esta manera, el modelo OSI consta de las siguientes capas: aplicación, presentación, sesión, transporte, red, enlace y físico.

Por su parte, el modelo TCP/IP constituye el estándar abierto de Internet, por lo que nos detendremos en algunos de sus protocolos más característicos, como IP, Ethernet, Wi-Fi, TCP y UDP. Una vez conocidos sus fundamentos teóricos, estudiaremos la configuración del protocolo TCP/IP en tarjetas de red inalámbricas y cableadas, así como los puntos de acceso Wi-Fi.

Los dispositivos intermedios de conexión de redes más importantes correspondientes a la capa de enlace de datos y capa de red del modelo OSI son los switches y los routers, respectivamente. Profundizando en estas capas, abordaremos conceptos como tablas de enrutamiento, dominios de colisión y dominios de difusión.

En cuanto a la capa física, estudiaremos los tipos de medios de comunicación guiados más utilizados: cable de cobre y cable de fibra óptica. Y también los medios de comunicación inalámbricos más extendidos: familia Wi-Fi, WiMAX, 4G, 5G, Zigbee y Bluetooth.

Por otro lado, nos detendremos en estudiar los tipos de redes según su tamaño y, en particular, en los tipos de tecnologías de conexión WAN.

En cuanto al tratamiento de la red por parte de los sistemas operativos, estos disponen de multitud de comandos y ficheros que permiten su configuración y monitorización. Su manejo es necesario para abordar la configuración integral del sistema operativo. Además, se indicarán los problemas y fallos más comunes, para lo que se hacen necesarios distintos planes de mantenimiento.

Por último, se darán a conocer mecanismos de seguridad en las comunicaciones y diferentes políticas que deben diseñarse para minimizar el riesgo que pueda ocasionar cualquier software malicioso en los sistemas.

## 5.2. Protocolos principales de red

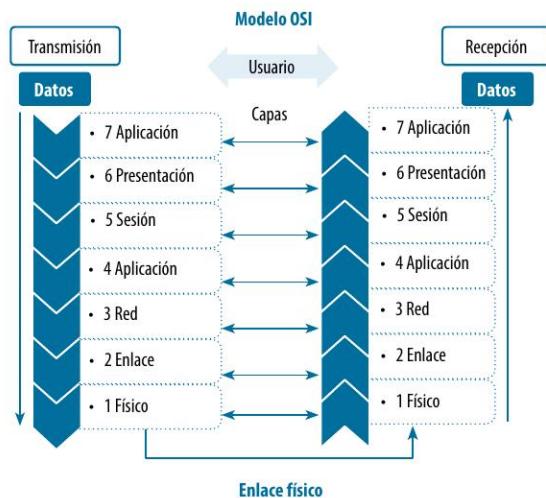
Los sistemas informáticos actuales se pueden considerar sistemas en red. Prácticamente no existe distinción hoy en día, ya que cualquier sistema informático con un sistema operativo que trabaje con un hardware específico de red y conectado con otros elementos de red con objeto de compartir información, forma parte de una red de comunicaciones.

Los sistemas informáticos en red se basan en modelos de referencia, que establecen las características y especificaciones necesarias para poder comunicarse entre diferentes entidades e intercambiar información. Estos modelos de referencia utilizan arquitecturas de red diferentes que descomponen sus funciones en varios niveles para definir protocolos y estándares, reducir la complejidad, controlar los flujos de comunicación y facilitar su evolución.

Los modelos de referencia más utilizados son el modelo de referencia OSI y el modelo de referencia TCP/IP.

El *modelo de referencia OSI* determina las funciones de comunicación de manera clara, dividiéndose en siete niveles. Cada nivel se corresponde con una capa que se comunica con su inmediata superior e inferior, de tal manera que el proceso de comunicación entre un emisor y un receptor sigue el recorrido que muestra la figura 5.1.

Cada capa aporta una traza con metainformación necesaria para su interpretación en el receptor. A este proceso se le denomina encapsulamiento, en el que cada capa añade a los datos de la capa superior información asociada al protocolo que representa, constituyendo unidades de paquetes de datos (PDU). Así, cuando el flujo de bits llega al receptor, deberá liberarse del encapsulamiento en la capa correspondiente hasta llegar a la más alta.



**Figura 5.1**  
Propiedades de un archivo.  
Pestaña seguridad.

Las capas y su función son las siguientes:

- *Aplicación*. Actúa de interfaz entre el usuario y las propias aplicaciones: navegadores web, aplicaciones de transferencia de ficheros, correo electrónico, terminales de red, exploradores de archivos, etc.

- *Presentación.* Determina el formato de la información para transferir entre las aplicaciones emisora y receptora. Codifica los datos, pudiendo comprimirlos o cifrarlos.
- *Sesión.* Define los mecanismos para establecer, mantener y controlar el diálogo entre las aplicaciones emisora y receptora. Las tres capas más altas no tienen un nombre concreto para sus PDU, por lo que se llaman *datos* en las tres.
- *Transporte.* Prepara y controla el flujo de datos entre emisor y receptor. Encapsula en *segmentos* los datos de la capa de sesión.
- *Red.* Encargada de seleccionar la ruta entre el emisor y receptor. Encapsula los segmentos de datos en *paquetes*.
- *Enlace de datos.* Establece mecanismos de detección y corrección de errores en la transmisión de datos. Encapsula los paquetes en *tramas*.
- *Física.* Determina las especificaciones mecánicas, eléctricas y funcionales que establece y mantiene el enlace físico de transmisión. La trama, constituida por *bits*, se traduce en señales eléctricas, electromagnéticas o pulsos de luz, hasta que llegan al receptor, donde se vuelven a convertir en ceros y unos.

Las cuatro capas inferiores se encargan del transporte y el control del flujo de datos, mientras que las tres superiores están relacionadas con las aplicaciones (el host).

A diferencia del modelo OSI, el *modelo TCP/IP* no es solo un modelo conceptual y genérico, sino que constituye el estándar abierto de Internet. El modelo TCP/IP se adapta al modelo OSI, o viceversa, de tal manera que existe una correspondencia entre las capas de ambos:

- a) La capa aplicación del modelo TCP/IP se corresponde con las capas aplicación, presentación y sesión del modelo OSI.
- b) La capa transporte del modelo TCP/IP se asocia con la capa de idéntico nombre del modelo OSI.
- c) La capa internet del modelo TCP/IP se corresponde con la capa red del modelo OSI.
- d) La capa acceso a la red del modelo TCP/IP se asocia con las capas enlace de datos y física del modelo OSI.

**CUADRO 5.1**  
**Correspondencia entre los modelos OSI y TCP/IP**

Modelo OSI	Modelo TCP/IP
7. Aplicación	a) Aplicación
6. Presentación	
5. Sesión	
4. Transporte	b) Transporte
3. Red	c) Internet
2. Enlace de datos	d) Acceso a red
1. Física	

El nombre del modelo TCP/IP hace referencia a los protocolos más importantes empleados en el modelo: TCP e IP. Cada capa del modelo tiene asociados multitud de protocolos. Muchos de ellos los conocemos por sus siglas y otros son más desconocidos. En el cuadro 5.2 se muestran algunos de ellos y, a continuación, se analizan algunos.

**CUADRO 5.2**  
Protocolos destacados del modelo TCP/IP

Protocolo	Utilidad	Capa
HTTP (Hypertext Transfer Protocol)	Web	APLICACIÓN
HTTPS (Hypertext Transfer Protocol Secure)		
SMTP (Simple Mail Transfer Protocol)	Correo electrónico	
POP3 (Post Office Protocol 3)		
IMAP (Internet Message Access Protocol)		
DHCP (Dynamic Host Configuration Protocol)	Obtención de direcciones IP	
DNS (Domain Name System)	Traducción de nombres de dominio a direcciones IP	
FTP (File Transfer Protocol)	Transferencia de archivos	
FTPS (File Transfer Protocol Secure)		
TLS (Transport Layer Security)	Encriptación	
SSL (Secure Sockets Layer)		
UDP (User Datagram Protocol)	Conexión y envío de información entre hosts	TRANSPORTE
TCP (Transmission Control Protocol)		
IP (Internet Protocol)	Enrutamiento de paquetes	INTERNET
NAT (Network Address Translation)	Traducción de direcciones IP privadas a públicas	
ARP (Address Resolution Protocol)	Correspondencia entre direcciones MAC e IP	ACCESO A LA RED
RARP (Reverse Address Resolution Protocol)		
ETHERNET	Transmisión por cableado	
WLAN (Wireless Local Area Network)	Transmisión por Wi-Fi	
FDDI (Fiber Distributed Data Interface)	Transmisión por fibra óptica	

### 5.2.1. Protocolo Ethernet

Establece una forma de conexión y transmisión de datos por cable donde se especifican las características del cableado y su señalización, así como el formato de las tramas de datos. Está asociado a la capa física del modelo OSI.

Esta tecnología emplea un mecanismo llamado CSMA/CD (acceso múltiple por detección de portadora y detección de colisiones) en un medio compartido por varios hosts. El host que desee transmitir ha de escuchar el medio previamente. Si está ocupado el canal, espera un tiempo antes de volver a intentarlo. Si dos hosts transmiten a la vez, se produciría una colisión y ambos detendrían la transmisión.

La principal ventaja de Ethernet es su bajo coste, flexibilidad y facilidad en su implementación y seguridad ante accesos no permitidos. Por ello, es la más empleada en redes de área local (LAN).

Ethernet se corresponde con el estándar *IEEE 802.3*, el cual se divide en multitud de versiones con diferente ancho de banda para cable coaxial, cable de par trenzado y cable de fibra óptica.

### 5.2.2. Protocolo Wi-Fi

La tecnología Wi-Fi define un conjunto de especificaciones para redes de área local inalámbricas, asociándose a la capa física del modelo OSI. La familia IEEE 802.11 establece multitud de estándares de transmisión de datos por radiofrecuencia en las bandas ISM con fines no comerciales.



#### SABÍAS QUE...

Las bandas ISM son aquellas que emplean ondas electromagnéticas de uso libre con fines industriales, científicos o médicos.

La familia de protocolos Wi-Fi emplean el mecanismo CSMA/CA (acceso múltiple por detección de portadora y prevención de colisiones) que, a diferencia de CSMA/CD, antes de transmitir, envía una notificación sobre su intención de hacerlo y, si recibe autorización, lo hace. Por tanto, se reduce considerablemente la probabilidad de colisiones en el medio.

Su principal ventaja es la facilidad en su instalación y la movilidad. Sin embargo, sus principales inconvenientes son la inseguridad al ser el medio de transmisión abierto y la saturación de los canales, donde se sitúan las bandas de 2,4 GHz y 5 GHz, creando interferencias y, por lo tanto, aumentando la latencia en las comunicaciones.

Los estándares Wi-Fi mejoran con el paso del tiempo a sus antecesores, siendo los más utilizados los siguientes.

**CUADRO 5.3**  
Estándares de la familia Wi-Fi

Estándar	Banda	Ancho de banda máximo
802.11a	5 GHz	54 Mbps
802.11b	2,4 GHz	11 Mbps
802.11g	2,4 GHz	54 Mbps
802.11n (Wi-Fi 4)	2,4 GHz y 5 GHz	600 Mbps
802.11ac (Wi-Fi 5)	5 GHz	7 Gbps
802.11ax (Wi-Fi 6)	2,4 GHz y 5 GHz	11 Gbps

### 5.2.3. Protocolo IPv4 e IPv6

El *protocolo IP*, el más conocido del modelo TCP/IP, se encarga del enrutamiento o encañamamiento de paquetes de datos. Es decir, decide la ruta más adecuada para transportar los paquetes desde el origen al destino, pasando por diferentes nodos intermedios. Además, utilizará

el direccionamiento a hosts (asignación de direcciones IP a interfaces de red) para poder enrutar los paquetes.

El protocolo IP no garantiza si un paquete llega a su destino y en qué orden, por lo que no es fiable, sin embargo, esta labor la pueden realizar otros protocolos de capas superiores como el *protocolo TCP*.

La dirección IP o dirección lógica se asigna a cada controlador o interfaz de red de un equipo que utilice el protocolo IP, como, por ejemplo, una tarjeta Wi-Fi o una tarjeta Ethernet. Las direcciones IP son necesarias para enviar y recibir paquetes, identificando de forma única cada dispositivo de red. Por tanto, no se pueden repetir dos direcciones IP en una misma red, ya que daría lugar a conflictos de red, ocasionando errores en la recepción o envío de datos.

Actualmente, se emplea el protocolo IP en sus versiones IPv4 e IPv6.

#### A) Protocolo IPv4

La versión IPv4 utiliza 32 bits, desglosados en 4 bloques de 8 bits separados por puntos. De tal manera que cada bloque representa un número comprendido entre 0 y 255.

Además, el protocolo establece que se necesita una *máscara de red*, con el mismo formato que una dirección IP, asociada a la dirección IP. De esta manera, se identifica la red a la que pertenece la dirección IP.

Intrínsecamente, la dirección IP se divide en una porción de red y una porción de host. La importancia de la máscara de red radica en que esta determina qué bits de la dirección IP se corresponden con la red a la que pertenece y qué bits especifica el host dentro de dicha red.

Un ejemplo de dirección IP y máscara de red asociada se muestra en las figuras 5.2 y 5.3:



**Figura 5.2**  
Ejemplo de dirección IP.



**Figura 5.3**  
Ejemplo de máscara de red.

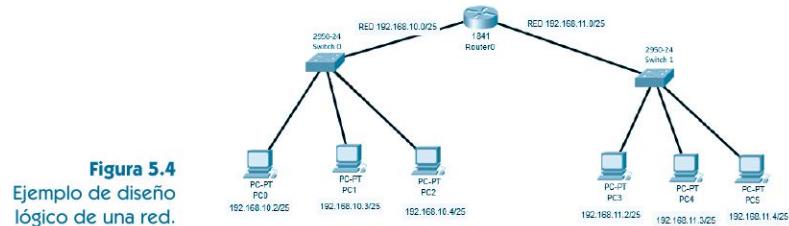
Los bits de la dirección IP que se encuentran en la misma posición que los bits 1 de la máscara de red (de izquierda a derecha) representan la porción de red. Y los bits de la dirección IP que se encuentran en la misma posición que los bits 0 de la máscara de red (de izquierda a derecha) representan la porción de host. Por ello, en el ejemplo de las figuras 5.2 y 5.3, los tres primeros octetos binarios (de izquierda a derecha) de la dirección IP se corresponden con la dirección de red.

La máscara de red también se puede representar como sufijo, es decir, se indica la dirección IP y el número de unos que contiene la máscara de red intercalando el carácter "/" (conocida como notación CIDR). Por ejemplo, 192.168.0.1/24 equivale a indicar la dirección IP 192.168.0.1 con máscara de red 255.255.255.0.

El administrador de la red debe establecer las diferentes redes y el rango de direcciones IP dentro de cada una. Por tanto, antes de asignar la dirección IP y la máscara de red de cada adaptador o interfaz de red se ha de diseñar la organización lógica de la red mediante un mapa de topología lógica donde se especifiquen los dispositivos y el esquema de direccionamiento IP.

Dentro del rango de direcciones IP de cada red se diferencian varios tipos de direcciones, a saber:

- ✓ *Dirección de red:* especifica la red. Se identifica por la primera dirección del rango de direcciones de la red, es decir, todos los bits de la porción de hosts se encuentran a 0. Resulta equivalente a realizar una operación AND bit a bit entre la dirección IP y la máscara de red. Siguiendo con el ejemplo anterior, la dirección de red es: 192.168.0.0.
- ✓ *Dirección de broadcast:* empleada para enviar paquetes a todos los hosts de la red a la vez. Se identifica por la última dirección del rango de direcciones de la red, es decir, todos los bits de la porción de hosts se encuentran a 1. Siguiendo con el ejemplo anterior, la dirección de broadcast es: 192.168.0.255.
- ✓ *Direcciones de hosts:* direcciones susceptibles de asignarse a hosts dentro de una red. Son aquellas comprendidas entre la dirección de red y la dirección de broadcast. Siguiendo con el ejemplo anterior, la dirección de host mínima es: 192.168.0.1 y la dirección de host máxima es: 192.168.0.254.



**Figura 5.4**  
Ejemplo de diseño lógico de una red.

Las direcciones IP se pueden catalogar en:

- Públicas:* para su uso con Internet y únicas a nivel mundial. Existen entidades que gestionan la asignación de direcciones IP públicas, como IANA (Assigned Numbers Authority) y los registros regionales de Internet, como el RIPE NCC en Europa.
- Privadas:* designadas para redes con un acceso restringido o nulo con Internet. Solo los siguientes bloques de direcciones IP se pueden asignar a redes privadas y no son asignables para Internet:
  - 10.0.0.0/8
  - 172.16.0.0/12
  - 192.168.0.0/16

Las redes privadas con acceso a Internet (como oficinas u hogares) disponen de un *router* que sí tiene acceso a Internet gracias al proveedor de servicios de Internet (compañía que da de alta el servicio). Este router es el que traduce direcciones IP privadas a públicas, y viceversa, gracias al protocolo NAT.

Además de la dirección lógica o dirección IP, los adaptadores de red disponen de una dirección física llamada *dirección MAC*, que está asociada a cada interfaz de red por el fabricante del producto. Esta dirección es única a nivel mundial y está formada por 48 bits que se representan

de manera hexadecimal con un formato del tipo: XX-XX-XX-XX-XX-XX. Es empleada por la capa de enlace de datos del modelo OSI y gracias a ella el protocolo Ethernet establece el origen y el destino de cada trama.

### B) Protocolo IPv6

El protocolo IP en su versión 6 (*IPv6*) emplea 128 bits y se representa en hexadecimal en bloques de dos bytes con un formato del tipo: 3D4A:1AD1:1FF0:43D1:A1BB:234C:4455:-FF00.



#### TOMA NOTA

El protocolo *IPv6* presenta una serie de ventajas:

- Aumenta la seguridad en la comunicación.
- Mejora el tratamiento de los paquetes.
- Incrementa el número de direcciones IP asignables, de tal manera que prácticamente sean inagotables.
- Permite implantar el Internet de Todo (IoE).

Las direcciones *IPv6* se pueden abbreviar mediante las siguientes reglas:

1. El bloque 0000 se reduce a 0.
2. Dos o más bloques consecutivos con valor 0 se reducen a ":" en una ocasión para una misma dirección.
3. Los ceros a la izquierda se pueden descartar.
4. Ejemplo: la dirección 5560:0088:0000:0000:F103:31AA:75AC:0000 equivale a 5560:88::F103:31AA:75AC:0.

Además, una dirección *IPv4*, como, por ejemplo, 192.168.1.5, puede escribirse en notación *IPv6* como 0:0:192.168.1.5, o también ::192.168.1.5.

#### 5.2.4. Protocolo TCP y UDP

En la capa de transporte del modelo OSI los protocolos más empleados son el *protocolo de control de transmisión (TCP)* y el *protocolo de datagramas de usuario (UDP)*. Ambos se encargan de establecer comunicaciones entre aplicaciones de host de origen y de host de destino, enviando y recibiendo datos entre ellas sin importar las capas inferiores: medios de transmisión, rutas de los datos, congestiones, tipos de hosts, etc.

Con objeto de mantener conversaciones entre aplicaciones de origen y destino, ambos protocolos deben segmentar los datos en origen (dividiéndose en partes manejables llamados *segmentos*) y reconstruyéndolos en el destino. Además, a las aplicaciones que participan se les asigna un número de puerto exclusivo en cada host.

Los protocolos TCP y UDP se diferencian en la forma en que se transfieren los segmentos entre host (ver apartado 5.10.1):

- Protocolo TCP garantiza que todos los segmentos lleguen al destino. Para ello, realiza un seguimiento de todos los datos transmitidos y recibidos (el receptor envía un acuse de recibo). En caso de no recibir un segmento, este se vuelve a enviar. Por tanto, el protocolo TCP es confiable.
- Protocolo UDP envía segmentos entre aplicaciones de manera rápida sin importar su confiabilidad, ya que la pérdida de algunos segmentos no compromete la comunicación entre las aplicaciones.

El protocolo TCP es confiable, pero más lento que UDP, al realizar todo el proceso de seguimiento, acuse de recibo y retransmisión. Los protocolos FTP y HTTP emplean TCP, mientras que las aplicaciones de *streaming* de vídeo y audio suelen emplear UDP.

### 5.3. Configuración del protocolo TCP/IP

La asignación de una dirección IP a un adaptador de red puede realizarse de dos maneras: estática y dinámica.

#### 5.3.1. Estática

Se emplea una dirección IP fija para un host (no cambia con el paso del tiempo), resultando ideal para servidores de Internet o que deban mantener la dirección IP para ofrecer servicios de impresión, HTTP, FTP, etc. La asignación de una dirección IP estática se puede realizar manualmente, por el administrador del sistema, a través de la configuración del adaptador de red.

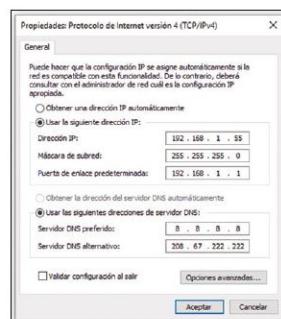
Para configurar un adaptador de red Ethernet de manera estática en Microsoft Windows, lo hacemos a través de las 'Conexiones de red', pudiendo acceder a través de 'Centro de redes y recursos compartidos' y 'Cambiar opciones del adaptador'. Abrimos el adaptador Ethernet objeto de configuración y accedemos a sus propiedades. Seleccionamos el 'Protocolo de Internet versión 4 (TCP/IPv4)' y pulsamos en 'Propiedades'.

El administrador de la red ha de indicarnos los datos de los campos para llenar. Para lo que debemos habilitar la opción de 'Usar la siguiente dirección IP':

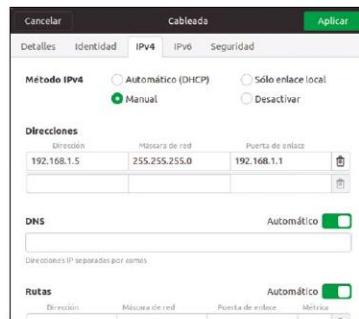
- ✓ Dirección IP.
- ✓ Máscara de subred.
- ✓ Puerta de enlace predeterminada o *Gateway por defecto*: normalmente es el router a través del cual accedemos a Internet. También puede ser otro host que interconecte redes distintas.

También debemos habilitar la opción 'Usar las siguientes direcciones de servidor DNS':

- a) Servidor DNS preferido: dirección IP de un servidor DNS que traduce direcciones de dominio a direcciones IP. Este servidor es el encargado, por ejemplo, de traducir el dominio www.sintesis.com a su dirección IP 193.70.32.179. Ejemplos de servidor DNS son *Google Public DNS* con IP 8.8.8.8 y 8.8.4.4 y *OpenDNS* con IP 208.67.222.222 y 208.67.220.220.
- b) Servidor DNS alternativo: es recomendable emplear otro servidor de DNS.



**Figura 5.5**  
Asignación estática de IPv4  
en Microsoft Windows.



**Figura 5.6**  
Asignación estática de IPv4  
en Ubuntu.

Para configurar un adaptador de red Ethernet de manera estática en Ubuntu, lo hacemos a través de 'Red', desde 'Configuración'. Al pulsar en la rueda dentada del adaptador, accedemos a su configuración. En la pestaña IPv4 podemos introducir los datos de manera estática, seleccionando la opción 'Manual'.

### 5.3.2. Dinámica

Con este mecanismo, la dirección IP cambia con el paso del tiempo. La mayoría de los equipos emplean este método gracias al *protocolo DHCP*. Dicho protocolo emplea un servidor DHCP, que provee la configuración necesaria a los clientes DHCP (hosts con esta configuración del adaptador de red habilitada) para poder comunicarse en red: dirección IP, máscara de red, servidor DNS, puerta de enlace, etc. Los *routers SoHo* (*Small office Home office*), es decir, los que solemos tener en casas u oficinas, habilitan este servidor por defecto y, por tanto, no nos tenemos que preocupar de su configuración.

Los servidores DHCP permiten establecer el rango de direcciones asignables por este protocolo. El resto de direcciones IP se reservan para direccionamiento estático. En la siguiente imagen se aprecia la configuración del servidor DHCP de un router SoHo, donde se indica el rango de direcciones IP asignadas por DHCP (dirección IP de inicio y fin), direcciones de servidores DNS, Gateway por defecto, etc. Todos estos datos se trasladan a la configuración TCP/IP de los clientes DHCP.

Por tanto, la configuración de las propiedades del Protocolo de Internet versión 4 (TCP/IPv4) se ha de mantener en automático para obtener una dirección IP dinámica.



**Figura 5.7**  
Configuración de un servidor DHCP.

## 5.4. Interconexión de redes. Componentes

Los sistemas informáticos en red emplean dispositivos intermedios que conectan distintas redes y host entre sí. Estos elementos se clasifican según la capa del modelo OSI sobre la que actúan. A saber:

**CUADRO 5.4**  
Dispositivos de interconexión por capas

Capa	Dispositivo	Función
Física	Repetidor	Regenera la señal entre dos puntos de una red. Existen inalámbricos o cableados.
	Hub	Replica la información entrante por uno de sus puertos al resto de puertos.
Enlace de datos	Switch	Conecta la información entrante por uno de sus puertos al puerto de destino únicamente.
	Punto de acceso	Extiende la red cableada mediante un medio inalámbrico. Pertenece a las capas 1 y 2 del modelo OSI.
Red	Router	Conecta redes diferentes.

### 5.4.1. Switch

Trabaja en la capa de enlace de datos del modelo OSI y tiene como función conectar varios segmentos de una misma red o, lo que es equivalente, dividir una red en subredes.

El switch, a diferencia del hub, evita que colisionen paquetes de datos en el medio de transmisión. Para ello, cuando un paquete es recibido por uno de sus puertos, solo lo retransmite al puerto de destino y no a todos los restantes.

**Figura 5.8**  
Símbolo de representación de switch.



### 5.4.2. Router. Tablas de enrutamiento

El router pertenece a la capa de red del modelo OSI y se encarga de conectar diferentes redes. Son dispositivos que disponen de su propio sistema operativo, CPU, RAM y ROM.

**Figura 5.9**  
Símbolo de representación de router.



Existen dos tipos de routers:

- Rackable o empresarial: empleados para la conectividad en armarios o racks donde se necesitan unas prestaciones superiores.
- Routers SoHo: son los que suelen suministrar los proveedores de acceso a Internet. Estos permiten conectar la red local de nuestra casa con Internet. Integran otros dispositivos, como: switch, punto de acceso Wi-Fi y firewall.



**Figura 5.10**  
Router SoHo Netgear.

Tanto los routers como los hosts utilizan las *tablas de enruteamiento* para encaminar los paquetes a otros dispositivos de una red local o remota. Cuando dos hosts se encuentran en una misma red local, en su comunicación no interviene el router. Sin embargo, cuando la comunicación es remota (redes distintas) entre un host origen y otro destino, sí son necesarios.

Los hosts emplean tablas de enruteamiento que almacenan las direcciones de hosts a los que puede enviar paquetes, que pueden ser:

- ✓ A él mismo. Se suele utilizar para realizar pruebas (direcciones IP 127.0.0.0/8).
- ✓ A un host local.
- ✓ A un host remoto. Cuando no encuentra una coincidencia en la tabla de enruteamiento, se entiende que es un host remoto y utiliza la dirección por defecto 0.0.0.0, que indica la puerta de enlace que se hará cargo del paquete.

Podemos observar la tabla de enruteamiento de un host en Microsoft Windows con el comando *netstat -r* y en GNU/Linux con *ip route show*.

Los routers emplean tablas de enruteamiento más complejas para poder enviar paquetes a redes diferentes, localizando la ruta más conveniente. Estas tablas se almacenan en la memoria de los routers. En ellas se indican las redes de destino, la métrica (valor asociado a cada destino que discrimina un mejor o peor encaminamiento) y la interfaz de salida para alcanzar la red de destino.

Los routers disponen de tres tipos de entradas en sus tablas de enruteamiento:

1. Conexiones locales. Conectadas directamente por alguna interfaz del router.
2. Conexiones estáticas. Establecidas manualmente por el administrador de la red.
3. Conexiones dinámicas. Entradas que han sido aprendidas mediante algún algoritmo de enruteamiento. Estos algoritmos son utilizados por los routers para comunicarse e intercambiar entradas entre ellos. La mayoría de las entradas son de este tipo.

#### 5.4.3. Topología física y lógica. Mapas

El diseño de una red de computadores se realiza mediante mapas que establecen la organización física o lógica de los dispositivos o componentes implicados. De esta manera, se estudia la organización de cara a su implementación y mejorar su eficiencia, según los objetivos que se pretendan conseguir. Así pues, distinguimos entre:

### A) Topología física

Ilustra la organización de los componentes y conexiones físicas entre elementos de red. Dentro de las topologías físicas, distinguimos los siguientes tipos:

✓ Inalámbricas:

- Distribuida: se emplean puntos de acceso para que los clientes se conecten a la red y se puedan mover libremente, saltando de uno a otro de forma transparente para ellos.
- Centralizada: se utilizan puntos de acceso sin capacidad de gestión, ya que se conectan varios de ellos a switches WLAN. Estos son los encargados de realizar el control y la gestión de la red Wi-Fi.



**Figura 5.11**  
Topologías cableadas.

✓ Cableadas:

- Redes de área extensa (WAN):
  - Punto a punto: dos equipos se comunican directamente.
  - Estrella: un equipo central interconecta todos los dispositivos.
  - Malla: todos los equipos están interconectados entre sí parcialmente o totalmente.
- Redes de área local (LAN):
  - Estrella.
  - Estrella extendida: estrellas unidas entre sí.
  - Bus: un medio totalmente compartido al cual se conectan distintos equipos.
  - Anillo: un medio compartido cerrado donde se conectan los equipos.

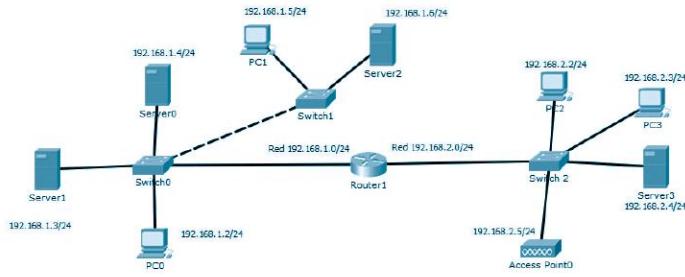
### B) Topología lógica

Para cada elemento de red se establece su configuración para la comunicación y acceso al medio. Dentro de las topologías lógicas, distinguimos los siguientes tipos:

- En redes WAN: se considera una conexión punto a punto entre dos equipos.

- En redes LAN: tenemos un medio compartido, con lo cual se necesita un conjunto de reglas para controlar el acceso. Para ello, existen dos métodos:
  - Acceso por contienda: cuando un equipo desea enviar al medio una trama hacia otro equipo, este escucha el medio y, si está libre, la envía. Si durante la transmisión se produce una colisión (dos tramas se interceptan al ser enviadas a la vez), la trama se descarta y se espera un tiempo para volver a enviarla. Empleado en redes Ethernet y Wi-Fi a través de los mecanismos CSMA/CD y CSMA/CA.
  - Acceso controlado: se establece un turno para poder enviar una trama. Cuando le toca el turno a un equipo, este puede enviar la trama. Si no lo hace, debe esperar a que le toque su turno. Este tipo de método es común en redes físicas en anillo, como Token Ring o FDDI.

A continuación, en la figura 5.12 se muestra un mapa lógico en el que intervienen diferentes elementos de red. Podemos distinguir los nombres de cada equipo de red, su dirección IP y líneas de conexión con otros dispositivos.



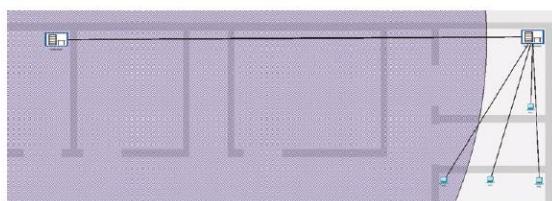
**Figura 5.12**  
Ejemplo de mapa lógico.

En la figura 5.12 podemos distinguir dos redes distintas, las cuales se comunican a través del router. Los switches comunican segmentos de una misma red.

**RECUERDA**

- ✓ En un mapa lógico, la dirección IP de los elementos es fundamental para la compresión de la organización.

En la figura 5.13 aparece un ejemplo de mapa físico que se corresponde con el anterior mapa lógico. Distinguimos las estaciones, los computadores de la red, los armarios de distribución y el rango de cobertura Wi-Fi en color púrpura el punto de acceso. Los componentes de red, como servidores, switches y routers se encuentran en los racks.



**Figura 5.13**  
Ejemplo de mapa físico.

#### 5.4.4. Dominios de colisión y difusión

Los dispositivos de capa 2 o superiores, como los routers y switches, dividen los dominios de colisión (áreas donde pueden colisionar paquetes). Además, los dispositivos de capa 3 o superiores, como los routers, dividen los dominios de difusión (áreas donde se reciben tramas de broadcast).

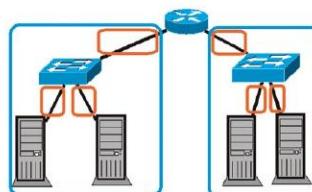
La segmentación de una red en dominios de colisión y dominios de difusión mejora la eficiencia de la red y aumenta el ancho de banda.

#### Ejemplos

En la figura 5.14 se distinguen cinco dominios de colisión, uno para cada dispositivo conectado al switch. Y en la figura 5.15 se muestran dos dominios de difusión (en azul) que, a su vez, disponen de tres dominios de colisión cada uno (en naranja).



**Figura 5.14**  
Dominios de colisión.



**Figura 5.15**  
Dominios de colisión.

#### 5.5. Tipos de redes

Las redes se pueden clasificar atendiendo a diferentes criterios:

- Según su tamaño:
  - Redes de área personal (PAN): su ámbito de acción es el entorno del propio usuario. Emplean tecnologías como Bluetooth, Zigbee o NFC.
  - Redes de área local (LAN): de poco alcance, que pueden abarcar un hogar, una oficina, una empresa o un edificio. Una red LAN inalámbrica se conoce como WLAN (Wireless LAN).

- Redes de área metropolitana (MAN): redes de extensión intermedia entre LAN y WAN, que suelen estar constituidas por varias redes LAN. Ejemplo de ello son las conexiones entre poblaciones próximas o el entorno de un campus universitario. Una red MAN inalámbrica se conoce como WMAN (Wireless MAN).
- Redes de área extensa (WAN): redes de larga distancia que conectan redes LAN o WAN. Las redes WAN pueden conectar ciudades lejanas e incluso continentes.

b) Según su transmisión:

- Redes punto a punto: se transmite la información desde un host origen a un host destino a través de un medio.
- Redes multipunto: permiten transmitir la información desde un host a múltiples destinos compartiendo el mismo medio.

c) Según su función:

- Redes entre iguales: los hosts interconectados ofrecen y acceden por igual a los servicios.
- Redes cliente-servidor: unos hosts ofrecen servicios y recursos (servidores) y otros acceden a ellos (clientes).

d) Según los medios empleados:

- Inalámbricas: emplean ondas electromagnéticas para la transmisión de información por el aire. Existen muchos estándares inalámbricos, siendo los más conocidos Bluetooth y Wi-Fi.
- Cableada: utilizan algún medio físico para transmitir señales portadoras de información. Los medios más empleados son el cable de par trenzado de cobre y el cable de fibra óptica de vidrio o plástico.
- Mixtas: utilizan ambos medios.

## 5.6. Acceso a redes WAN. Tecnologías

Cuando se necesita comunicar varias redes LAN entre sí a largas distancias, entran en acción las redes WAN. Las redes LAN son propiedad de particulares que, cuando deciden conectarse a otra red LAN inalcanzable geográficamente por dicho propietario o a Internet, deben suscribirse a un proveedor de servicios de red o un proveedor de Internet (ISP).

Las redes de área extensa requieren estándares y tecnologías diferentes a las redes LAN debido a las grandes distancias con las que trabajan. Principalmente, estas tecnologías se centran en las capas de red, enlace de datos y física del modelo OSI.

Las principales tecnologías WAN se agrupan en diferentes tipos de conexión:

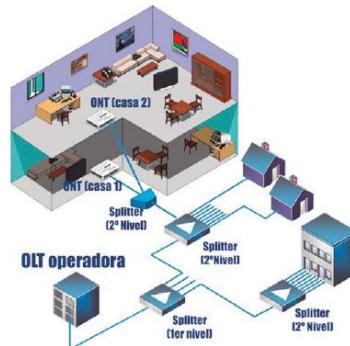
### 5.6.1. Conexiones WAN privadas

Entre las conexiones privadas, se pueden encontrar diversos tipos:

- ✓ Comutación de circuitos: requieren que se establezca un circuito (canal) dedicado entre los nodos y terminales antes de que se comuniquen los usuarios. Ejemplo de ello es la red

telefónica commutada tradicional, donde el canal es compartido por varias conversaciones gracias a la multiplexación por división temporal (TDM), la cual reparte el tiempo de las conexiones por turnos. Ejemplos de tecnologías de este tipo son PSTN e ISDN (RDSI).

- ✓ Conmutación de paquetes: divide los datos para transmitir en paquetes a través de una red compartida. A diferencia de la conmutación de circuitos, no es necesario que se establezca un circuito previamente. Además, la red compartida facilita la comunicación entre multitud de pares de nodos a través del mismo canal. Por ello, este tipo de conexión WAN resulta más económica que la conmutación de circuitos, aunque sus latencias son superiores. No obstante, las tecnologías actuales de este tipo permiten comunicaciones de voz y vídeo. Ejemplos son Frame Relay, x.25 y ATM.
- ✓ Dedicada: se emplea para una conexión directa y permanente entre dos nodos de la red WAN del proveedor de servicios (conectando diferentes localizaciones del cliente entre un origen y un destino remoto). Su coste es muy elevado, pero se reducen tiempos de latencia, siendo ideal para voz sobre IP (VoIP) y vídeo sobre IP.



**Figura 5.16**  
Elementos de conexión FTTH.  
Fuente: <http://fibraopticahastaelhogarecuador.blogspot.com/>

### 5.6.2. Conexiones WAN públicas

A continuación, se detallan las conexiones WAN públicas más usadas:

- DSL (Digital Subscriber Line): es una familia de tecnologías, como SDSL, ADSL, VDSL y HDSL, en sus diferentes versiones. Permiten acceder a Internet mediante cables de cobre de par trenzado de la red telefónica con un ancho de banda aceptable.
- FTTH o fibra hasta el hogar: alcanza velocidades muy superiores a la familia DSL, empleando fibra óptica desde la red troncal hasta los clientes. Utilizan un conjunto de equipos con tecnología GPON (Gigabit-capable Passive Optical Network), a saber:
  - OLT (Optical Line Termination): dispositivo activo del que parten las fibras a los diferentes usuarios.
  - Divisor óptico o splitter: divide la señal óptica entrante en partes iguales de menor potencia a diferentes ramas o usuarios. Existen splitters de diferentes niveles de división, según la envergadura de la red troncal, y de distribución.
  - ONT (Optical Network Terminal): convierte las señales ópticas en señales eléctricas, y viceversa. Se integrada en los routers SoHo actualmente.
- HFC o híbrido fibra-coaxial: emplea fibra óptica en la red troncal y cable coaxial en su red de distribución hasta los hogares.

- Inalámbricas: existen diferentes tecnologías que utilizan ondas electromagnéticas para la transmisión de datos. Principalmente, se diferencian en la longitud de onda empleada y su frecuencia, por lo que son muy empleadas en redes WAN:
  - WiMAX: permiten un alcance alrededor de 60 km, pudiendo alcanzar 1 Gbps. Es ideal para zonas que no dispongan de cobertura por cable.
  - LTE-A (4G) y 5G: permiten gran movilidad de los terminales inalámbricos, llegando a alcanzar varios Gbps.

**TOMA NOTA**

Internet se emplea como una alternativa muy económica al uso de conexiones WAN privadas. Ejemplo de ello es la tecnología VPN (Virtual Private Network), ya que permite establecer una conexión segura a través de una red pública (como Internet) entre redes privadas.

## 5.7. Redes cableadas

Las redes de comunicación cableadas son aquellas que emplean algún medio de transmisión guiado, como cables de cobre (coaxial o par trenzado) o de fibra óptica. Aunque la instalación de los medios guiados es mucho más compleja que los inalámbricos, presentan multitud de ventajas, como su seguridad o un gran ancho de banda sostenido.

### 5.7.1. Tipos y características

Los medios de transmisión cableados más utilizados son el cable de cobre de par trenzado y la fibra óptica, los cuales pasamos a detallar.

#### A) Cable de cobre de par trenzado

Formado externamente por una cubierta de PVC, que dispone en su interior de ocho cables de cobre aislados y entrelazados, identificados por el color individual de su cubierta. Los cables están entrelazados por pares de la siguiente manera:

- ✓ Azul - blanco/azul
- ✓ Naranja - blanco/naranja
- ✓ Verde - blanco/verde
- ✓ Marrón - blanco/marrón



**Figura 5.17**  
Cable UTP (sin blindaje).

Podemos encontrar protegido el cable contra interferencias electromagnéticas externas mediante diferentes blindajes en los pares o en el cable.

**CUADRO 5.5**  
**Tipos de blindajes en cable de cobre de par trenzado**

U/FTP	Pantalla de aluminio en los pares	
F/FTP	Pantalla de aluminio en los pares y en el cable	
S/FTP	Pantalla de aluminio en los pares y malla de aluminio en el cable	
F/UTP	Pantalla de aluminio en el cable	
SF/UTP	Pantalla y malla de aluminio en el cable	

Además, los cables emplean conectores de tipo RJ-45 para su conexión en tarjetas adaptadoras de red, routers, switches, etc. Este conector presenta la siguiente forma.

**Figura 5.18**  
**Conector RJ45 para cables UTP.**



**Figura 5.19**  
**Conector RJ45 apantallado para cables con blindaje.**



La terminación de los cables en el conector, es decir, el orden en el que han de ser engastados a él, está regulada por la norma TIA/EIA-568-B, la cual establece dos tipos: T-568A y T-568B. El orden de cada terminación es el siguiente:

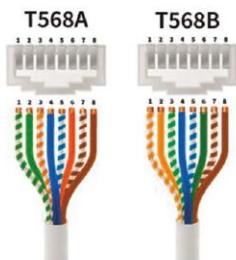
- T-568A:

1. Blanco/verde.
2. Verde.
3. Blanco/naranja.
4. Azul.
5. Blanco/azul.
6. Naranja.
7. Blanco/marrón.
8. Marrón.

- T-568B:

1. Blanco/naranja.
2. Naranja.
3. Blanco/verde.
4. Azul.
5. Blanco/azul.
6. Verde.
7. Blanco/marrón.
8. Marrón.

El cable con la misma terminación en ambos extremos se denomina directo, y con distinta terminación, cruzado.



**Figura 5.20**  
Código de colores  
para terminaciones  
T-568A y T-568B.

Además, el estándar TIA/EIA-568-B determina varias categorías de cable de par trenzado, según sus características eléctricas. Esto detalla aspectos, como su frecuencia de funcionamiento y la velocidad máxima. Las más utilizadas son: Cat5e, Cat6, Cat6e, Cat7 y Cat7e. A mayor categoría, mayor es su frecuencia y ancho de banda.

El cable de cobre de par trenzado destaca por su facilidad en la instalación, prestando un gran ancho de banda a un bajo coste, tanto en el propio medio como en los dispositivos de interconexión.

### Actividad propuesta 5.1



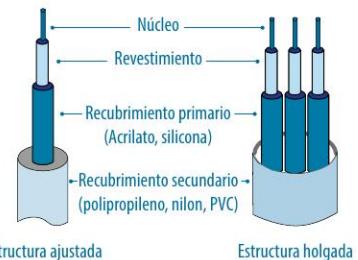
Busca información en Internet sobre las categorías Cat5e, Cat6, Cat6e, Cat7 y Cat7e, indicando su ancho de banda y la distancia máxima.

### B) Cable de fibra óptica

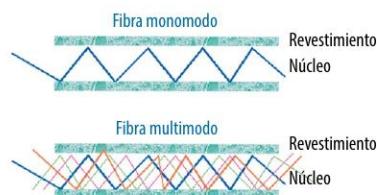
El cable de fibra óptica está formado por uno o más hilos de fibra de vidrio o plástico, recubierto por varias capas de diferentes materiales para dotarle de protección y rigidez.

Podemos clasificar los cables de fibra óptica en diferentes tipos:

- ✓ Según su estructura interna:
  - Estructura holgada: los hilos de fibra óptica se encuentran con cierta libertad en tubos dentro del cable de fibra óptica. Es utilizada principalmente en redes de área local (LAN) y metropolitanas (MAN).
  - Estructura ajustada: los hilos de fibra óptica no presentan libertad de movimiento debido a su recubrimiento secundario, por lo que solo existe un hilo de fibra por tubo. Normalmente se emplea para redes metropolitanas (MAN) y de área extensa (WAN).
- ✓ Según el modo de transmisión:
  - Monomodo (SM): se emite un único haz de luz por el interior del hilo. Es empleado principalmente para largas distancias. Los estándares monomodo más utilizados son OS1 y OS2.
  - Multimodo (MM): transmite varios haces de luz con diferentes trayectorias. Se suele utilizar para distancias cortas (entre manzanas de edificios o en el interior de estos). Los estándares más empleados son OM1, OM2, OM3, OM4 y OM5.



**Figura 5.21**  
Estructura de cables de fibra óptica.



**Figura 5.22**  
Transmisión en fibra óptica.

Algunos de los conectores más empleados para cables de fibra óptica son: FC, ST, LC, SC, MT-RJ y MPO.



**Figura 5.23**  
Conector LC.



**Figura 5.24**  
Conector SC.



**Figura 5.25**  
Conector MT-RJ.



**Figura 5.26**  
Conector MPO.

#### TEN EN CUENTA

- ✓ A diferencia de los cables de cobre de par trenzado, los cables de fibra óptica son extremadamente seguros en su transmisión y son capaces de sostener velocidades muy altas a larga distancia. Sin embargo, en su despliegue e instalación se requieren dispositivos caros (como fusionadoras de fibra óptica), así como elementos de interconexión de coste superior a los de cobre de par trenzado.



#### Actividad propuesta 5.2

Busca en Internet versiones de Ethernet para cables de tipo coaxial, par trenzado y fibra óptica. Especifica su ancho de banda.

### 5.7.2. Dispositivos de interconexión

El estándar TIA/EIA-568-B, que define el diseño e implementación del cableado en un edificio o entre varios, establece una topología de red en estrella. En la topología en estrella, los nodos y hosts están conectados a un nodo central que commuta y controla el flujo de datos entre todos ellos.

Los elementos de electrónica de red empleados para conectar cables de par trenzado como nodo central son principalmente los switches y routers. Estos dispositivos se instalan en armarios de distribución o *racks*, que alojan multitud de elementos, dependiendo de la envergadura de la infraestructura de red.

Los racks suelen contener:

- Dispositivos de electrónica de red: switches, routers, etc.
- Paneles de parcheo: elementos de conexión de cables que facilitan la conexión, organización y estructura del cableado en el rack. En el otro extremo del cable, la terminación es la *toma de usuario* en las *áreas de trabajo*.
- Otros elementos: regletas eléctricas, bandejas, organizadores de cables, etc.

También existen dispositivos de interconexión y adaptadores de red de fibra óptica y mixtos (para cables de par trenzado y fibra óptica).



**Figura 5.27**  
Detalle de un rack.



**Figura 5.28**  
Roseta con cuatro tomas.

### Actividad propuesta 5.3



Busca en Internet el significado de los siguientes estándares: 100BaseT, 100BaseFX y 10GBASE-T.

### 5.7.3. Adaptadores

Las tarjetas de red o adaptadores de red, también llamados *NIC (Network Interface Controller)*, son necesarios para que los hosts puedan conectarse a una red. Existen diferentes tipos, atendiendo a las características de estas:

- a) Medio de transmisión: cable de cobre de par trenzado, fibra óptica, etc.
- b) Conectividad con el host: integrada, PCIe, USB, etc.
- c) Modo de transmisión: full dúplex o half dúplex, según pueda emitir y recibir datos de forma simultánea o no, respectivamente.
- d) Velocidad de conexión: 10Mbps, 100Mbps, 1Gbps, 10Gbps, etc.
- e) Wake On LAN: característica que permite al adaptador encender el host de forma remota.



**Figura 5.29**  
Tarjeta de red con puerto RJ-45.

## 5.8. Redes inalámbricas

Las redes inalámbricas aportan ventajas con respecto a las redes cableadas, como la movilidad, la flexibilidad y la facilidad de instalación. Estas emplean ondas electromagnéticas para transmitir datos, cuya capacidad de transmisión depende, principalmente, de:

- ✓ Longitud de onda: distancia entre dos crestas o valles consecutivos de una onda. Se mide en metros.
- ✓ Frecuencia: número de veces que se repite una onda en un segundo. Se mide en hertzios (Hz).

El espectro electromagnético representa un amplio rango de ondas electromagnéticas, según su longitud de onda.

### 5.8.1. Tipos y características

Las redes inalámbricas más usadas son las redes Wi-Fi, WiMAX, los sistemas de comunicación móviles 4G y 5G, así como otras redes WPAN, como Bluetooth o Zigbee.

Cada una dispone de unas características que las hacen más apropiadas para según qué aplicaciones o usos. Pero dichas características pueden reducir su capacidad de transmisión debido a factores, como colapso de la banda de trabajo de la red inalámbrica, fuentes electromagnéticas externas, número de usuarios conectados, posicionamiento de las antenas y de los receptores, uso en interiores o exteriores, etc.

#### A) Wi-Fi

Su principal objetivo es la transmisión de datos a gran velocidad en una red local. Wi-Fi se basa en el conjunto de estándares IEEE 802.11, entre los que destacamos: IEEE 802.11b, IEEE 802.11g, IEEE 802.11n e IEEE 802.11ac (Wi-Fi 5).

Estos estándares trabajan en las bandas de 2,4 GHz y 5 GHz, pudiendo alcanzar un ancho de banda teórico de 10 Gbps con un alcance de 1 Km, aproximadamente. Cada estándar posee un rango de acción y ancho de banda diferente que está asociado a su frecuencia. A menor frecuencia, mayor es su alcance y menor ancho de banda.

Estas redes necesitan puntos de acceso Wi-Fi para conectar los diferentes terminales con NIC inalámbricas como teléfonos inteligentes, computadores, Smart TV, etc.



#### Actividad propuesta 5.4

Busca en Internet información acerca de los estándares Wi-Fi estudiados, así como 802.11ad, 802.11af y 802.11ax. Crea una tabla donde se describa su banda de trabajo (frecuencia), alcance y ancho de banda teórico máximo.

### B) WiMAX

Establece una red de comunicación de alta velocidad para redes MAN con alcance de decenas de kilómetros. Se fundamenta en la familia de estándares IEEE 802.16, llegando a alcanzar 1 Gbps.

Su infraestructura es parecida a los sistemas de comunicación móvil 4G y 5G. Requieren estaciones base con dispositivos electrónicos para emitir señales microondas y receptores WiMAX.



#### SABÍAS QUE...

Debido a sus características y el coste de instalación, su aplicación se centra en dotar de acceso a Internet y telefónico a áreas geográficas poco densas o lejanas, donde el coste del despliegue de cable de fibra óptica o de cobre resultaría costoso.

### C) Sistemas de comunicación móvil 4G y 5G

Las siglas 4G y 5G hacen mención a las actuales generaciones de tecnologías de comunicación móvil para redes WMAN y WWAN.

El estándar *LTE-Advanced* detalla las características técnicas de la cuarta generación con un ancho de banda de hasta 1 Gbps.

El estándar *5G NR* recoge los aspectos técnicos de la generación 5G, pudiendo alcanzar 20 Gbps. Su consumo es muy inferior a su antecesor y presenta mayor capacidad, siendo ideal para aplicaciones en tiempo real y el desarrollo del *IoT* (*Internet de las Cosas*).

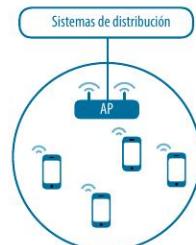
### D) Otras redes WPAN

Además, se suelen utilizar redes de área personal inalámbricas para una comunicación entre dispositivos de forma directa, sin utilizar dispositivos intermedios. Los más empleados son:

- Zigbee: definido por el estándar IEEE 802.15.4, que se fundamenta en su bajo consumo y baja tasa de transferencia de datos. Sus principales usos son aplicaciones de control y monitorización a muy bajo coste.
- Bluetooth: recogido en el estándar IEEE 802.15.1, pretende facilitar transmisión de datos y voz entre dispositivos cercanos, así como la sincronización, eliminando su conexión por medio de cables. Su empleo en condiciones ideales puede superar los 200m y un ancho de banda de varias decenas de bps.

Tanto Zigbee como Bluetooth, en sus diferentes versiones, se orientan cada vez más al *IoT*, gracias a la reducción del consumo, bajo coste y mayor rango de acción.

Además, existen otros estándares, como NFC, que se consideran de corto alcance y permiten una comunicación de datos entre dos dispositivos a pocos centímetros de distancia.



**Figura 5.30**  
BSS.



## Actividades propuestas

- 5.5. Realiza una tabla comparativa de los estándares WPAN estudiados, indicando: ancho de banda, alcance, principales usos y ejemplos de aplicaciones reales.
- 5.6. Busca en Internet tres aplicaciones, utilidades o avances que se puedan realizar gracias a la tecnología 5G.

### 5.8.2. Dispositivos de interconexión

Cada tipo de red inalámbrica requiere dispositivos de interconexión adecuados a las características de transmisión definidas por el estándar de dicha red. Se crea así la infraestructura de red inalámbrica necesaria en cada caso.

De esta manera, en redes WiMAX, 4G o 5G, las estaciones base están provistas de equipos de telecomunicaciones y antenas para aportar la cobertura necesaria a los usuarios de una zona.

En el caso de la tecnología Wi-Fi, normalmente se utilizan puntos de acceso Wi-Fi (PA) para conectarse y ofrecer los servicios necesarios a los distintos dispositivos, creando así la red inalámbrica. No obstante, se pueden utilizar diferentes topologías de red:

- a) Modo ad hoc (IBSS): dos clientes se conectan directamente sin emplear ningún dispositivo de infraestructura.
- b) Modo infraestructura: los clientes se conectan mediante un dispositivo de infraestructura (normalmente puntos de acceso inalámbricos). Estos puntos de acceso Wi-Fi se conectan al sistema de distribución (normalmente switches o routers). Diferencia dos tipos:
  - Conjunto de servicios básicos (BSS): existe un único punto de acceso que ofrece unos servicios básicos para que los clientes se puedan comunicar en la zona de cobertura de dicho punto de acceso. Si los clientes se salen de la zona de cobertura, no se podrán comunicar.
  - Conjunto de servicios extendidos (ESS): varios puntos de acceso se conectan mediante un sistema de distribución (de manera cableada o inalámbrica). Gracias a ello, se amplía la zona de cobertura y los clientes pueden circular entre puntos de acceso sin perder la conexión.

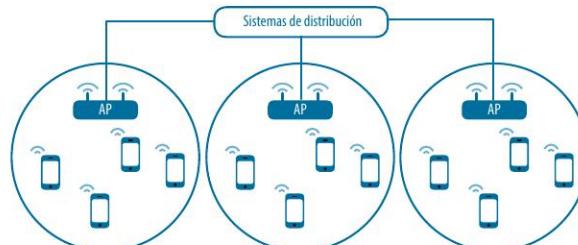


Figura 5.31  
ESS.

En general, los estándares Bluetooth y NFC trabajan en modo ad hoc, mientras que Zigbee puede trabajar en ambos modos de infraestructura.

**Actividad resuelta 5.1**

*Instala y configura un punto de acceso.*

**SOLUCIÓN**

Como ya sabemos, los routers SoHo integran puntos de acceso. No obstante, se pueden instalar nuevos puntos de acceso a routers SoHo siguiendo los pasos de instalación y configuración de dichos dispositivos aportados por el fabricante.



**Figura 5.32**

Esquema de un punto de acceso Wi-Fi conectado a un router SoHo.

Una vez hechas las conexiones, accedemos a la interfaz de configuración del punto de acceso, ya sea independiente o integrado en un router SoHo.

La configuración se realiza mediante un navegador web a través de la dirección y credenciales aportadas por el fabricante (normalmente <http://192.168.1.1>, usuario *admin* y contraseña *admin*).

Tras acceder, establecemos los parámetros de la red Wi-Fi:

- SSID: nombre de la red.
- Método y contraseña de autenticación: normalmente seleccionaremos WPA2 o WPA3 con una contraseña robusta (letras mayúsculas y minúsculas, números y caracteres especiales).

### 5.8.3. Adaptadores

Los equipos que deseen conectarse a una red inalámbrica necesitan adaptadores de red inalámbricos del tipo de red a la que va a conectarse. Muchos de ellos integran estos adaptadores en el propio hardware de los dispositivos, como los smartphones (que integran adaptadores Wi-Fi, 4G, 5G, Bluetooth, NFC, etc.), los portátiles y placas base de computadores de sobremesa que integran (adaptadores Ethernet, Bluetooth o Wi-Fi). En otros casos, podemos adquirir adaptadores inalámbricos externos o internos, que deben instalarse en alguna ranura de expansión interna o puerto de conexión externo.

Las características más importantes de los adaptadores de red Wi-Fi son:

- ✓ Estándares Wi-Fi soportados: los más comunes son IEEE 802.11 a/b/g/n/ac.
- ✓ Bandas de trabajo (frecuencias).
- ✓ Velocidad de transferencia medida en Mbps o Gbps.

- ✓ Conectividad con el host: integrada, M.2, PCIe, USB, etc.
- ✓ Antenas: número de antenas y características.
- ✓ Seguridad: protocolos de seguridad como WEB, WPA, WPA2, WPA3, etc.

Existen adaptadores que integran otros tipos de adaptadores, como Wi-Fi y Bluetooth, o Wi-Fi y Ethernet.



### Actividad resuelta 5.2

*Compartir una conexión a Internet con otros dispositivos mediante una tarjeta de red Wi-Fi. Para este ejercicio disponemos de un equipo con Microsoft Windows con conexiones Ethernet o datos móviles a Internet y un adaptador de red Wi-Fi.*

#### SOLUCIÓN

Microsoft Windows dispone de la opción 'Zona con cobertura inalámbrica móvil', que se activa desde 'Configuración' y 'Red e Internet'. Entre sus opciones, se puede indicar la conexión desde la que se desea compartir la conexión a Internet y editar el nuevo nombre de la red y la contraseña de acceso Wi-Fi.

A continuación, y desde el otro equipo, buscamos las redes Wi-Fi disponibles y accedemos a la nueva con la contraseña antes configurada.

## 5.9. Ficheros de configuración de red

Ubuntu emplea la herramienta NetPlan para gestionar y administrar la configuración de red. Desde Ubuntu 17.10, se emplea NetPlan con la intención de sustituir la configuración clásica anterior (a través del archivo `/etc/network/interfaces`).

Antes de realizar la configuración de la red, podemos conocer las interfaces de red identificadas por el sistema (para su posterior configuración) mediante los comandos:

```
ip a  
sudo lshw -class network
```

El directorio `/etc/netplan/` alberga los archivos de configuración de NetPlan. Para las distribuciones Ubuntu Desktop, encontramos en dicho directorio los archivos: `01-network-manager-all.yaml` que establece la primera configuración, `02-network-manager-all.yaml` para la segunda (si se dispone), etc. De tal manera que se aplican estas configuraciones en el mismo orden numérico que el comienzo de su nombre. La configuración de estos archivos ha de realizarse con privilegios de administrador y debemos seguir la siguiente sintaxis, respetando los caracteres espacio:

```
Network:  
  Version: 2  
  Renderer: NetworkManager/networkd  
  ethernets:  
    Nombre_dispositivo:  
      dhcp4: yes/no
```

```
addresses: [DIRECCION_IP/MÁSCARA_DE_RED]
gateway4: GATEWAY
nameservers:
    addresses: [NOMBRE_1, NOMBRE_2]
```

Donde:

- *Renderer*: nombre del gestor de red. *NetworkManager* es usado en sistemas de escritorio y *netwrokd* en servidores.
- *Nombre dispositivo*: se sustituye por el nombre de la interfaz para configurar.
- *dhcp4*: se indican los valores *yes* o *no*, si se configura por DHCP o con direccionamiento estático, respectivamente.
- *addresses*: se indica la dirección IP con notación prefijo.
- *gateway4*: señala la puerta de enlace.
- *nameservers*: indica las direcciones IP de los servidores DNS, siguiendo el formato indicado.

### Ejemplo

Veamos ejemplos de configuraciones:

- Configuración estática de una interfaz ethernet.
- Configuración dinámica de una interfaz ethernet.

```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: no
      addresses: [192.168.1.8/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.8.4]
```

**Figura 5.33**  
Ejemplo de configuración estática.

```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: yes
```

**Figura 5.34**  
Ejemplo de configuración dinámica.

Para establecer los cambios, utilizamos el comando *netplan apply*. Y, por último, comprobamos los cambios en las interfaces con *ip address show*. Veamos el proceso completo con el ejemplo de la figura 5.35.

Para llevar a cabo una configuración de los servidores DNS de manera temporal, podemos modificar el archivo */etc/resolv.conf* directamente. Pero esta acción no es recomendable, ya que NetPlan lo configura y actualiza dinámicamente (a través de *systemd-resolve*).

```
luis@luis-VirtualBox:~$ cat /etc/netplan/01-network-manager-all.yaml
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: no
      addresses: [192.168.1.6/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.8.4]

luis@luis-VirtualBox:~$ sudo netplan apply
luis@luis-VirtualBox:~$ ip address show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
    link/ether 08:00:27:70:6e:a8 brd ffffff:ffff:ffff:ff
    inet 192.168.1.6/24 brd 192.168.1.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe70:6ea/64 scope link
            valid_lft forever preferred_lft forever
```

**Figura 5.35**  
Ejemplo de configuración de red.

El orden de los mecanismos de resolución de nombres en los sistemas GNU/Linux viene establecido en el fichero `/etc/nsswitch.conf`. Este archivo es editable por el administrador del sistema, pudiendo modificar los mecanismos o su orden de aplicación.

La siguiente línea del fichero `/etc/nsswitch.conf` establece la resolución de nombres, donde, de izquierda a derecha, se indica el orden:

```
hosts:      files mdns4_minimal [NOTFOUND=return] dns
```

**Figura 5.36**

Línea del fichero `/etc/nsswitch.conf`, donde se especifica la resolución de nombres.

Donde:

- ✓ `files`: fichero `/etc/hosts`
- ✓ `mdns4_minimal [NOTFOUND=return]`: utilizar el protocolo mDNS (para los nombres acabados en `.local`).
- ✓ `dns`: fichero `/etc/resolv.conf`

El archivo `/etc/hosts` contiene entradas con asignaciones entre direcciones IP y nombres de hosts. Por defecto y según la definición del fichero `/etc/nsswitch.conf`, el archivo `/etc/hosts` tiene prioridad sobre la configuración DNS del equipo, por lo que si se intenta resolver una dirección IP de un host coincidente con una entrada del archivo `/etc/hosts`, no se resolverá a través de DNS.

**Recurso digital 5.1**

Configuración de red en distribuciones anteriores a Ubuntu 17.10.

En Microsoft Windows el archivo de configuración `hosts`: se encuentra en `C:\Windows\system32\drivers\etc\` y se encarga de mantener un listado de asociaciones entre direcciones IP y dominios. Para agilizar la traducción de resolución de nombres de dominio, el orden en Microsoft Windows es el siguiente:

- Memoria caché del navegador web.
- Archivo hosts.
- Servidores DNS.

Gracias a ello, al resolver `localhost` en un navegador web, se accede a la dirección IP 127.0.0.1 (definida esta asociación en el archivo hosts). Para editar este archivo, se debe realizar con privilegios de administrador, donde, por defecto, únicamente se define la interfaz de `loopback` (interfaz virtual de pruebas).

## 5.10. Monitorización y verificación de una red mediante comandos

Ya hemos estudiado cómo podemos establecer la configuración de los adaptadores de red de manera permanente. No obstante, se pueden realizar modificaciones de las interfaces temporalmente empleando otros comandos.

Existen multitud de comandos que ayudan a monitorizar y verificar el correcto uso de la red.

Los principales comandos que permite monitorizar, mostrar información y configurar el entorno de red de un host en GNU/Linux son *ip* y *ss*.

El comando *ip* es muy potente. Las acciones de configuración de red más usuales son:

- ✓ Listar las interfaces activas e inactivas: *ip a*
- ✓ Deshabilitar una interfaz: *ip link set <interfaz> down*
- ✓ Habilitar una interfaz: *ip link set <interfaz> up*
- ✓ Configurar una interfaz. *ip addr add <dir\_IP/mascara> dev <interfaz>*
- ✓ Eliminar una dirección IP: *ip addr del <dir\_IP/mascara> dev <interfaz>*
- ✓ Mostrar la tabla de enrutamiento: *ip route show*
- ✓ Borrar una puerta de enlace predeterminada: *ip route del 0.0.0.0/0 vía dir\_IP dev <interfaz>*
- ✓ Añadir una puerta de enlace predeterminada: *ip route add 0.0.0.0/0 vía dir\_IP dev <interfaz>*
- ✓ Mostrar la tabla ARP: *ip neighbour show*

#### RECUERDA

- ✓ Es muy recomendable la lectura de la ayuda del comando *ip* (*man ip*) para mayor detalle.

```
ultralinux-VirtualBox:~$ sudo ip addr add 192.168.1.6/24 dev enp0s3
ultralinux-VirtualBox:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:00:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.6/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
        valid_lft 86019sec preferred_lft 85919sec
    inet6 fe80::5f76:f18c:7585:5db4/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

**Figura 5.37**  
Configurar una interfaz con el comando *ip*.

Otro comando muy utilizado y que también nos puede ayudar a identificar y mostrar multitud de detalles de las interfaces de red es *lshw*.

Para mostrar las asociaciones entre las direcciones físicas (MAC) y direcciones IP del segmento de red local en un equipo, se utilizan los comandos *arp* (en Microsoft Windows) e *ip neighbor show* (en GNU/Linux). Estas asociaciones se almacenan en una tabla ARP (también llamada *caché ARP*) y son necesarias para incluir las direcciones MAC en las tramas de la capa de enlace de datos. Su sintaxis es:

*arp [opciones]*

Además, permite modificar las entradas de la tabla, dependiendo de las opciones aplicadas al comando.

Con ello, podemos detectar a qué equipos de su red se ha conectado un host gracias a las direcciones físicas de su tabla ARP.

```
C:\Users\Luis>arp -a
Interfaz: 10.0.2.15 --- 0x5
          Dirección de Internet   Dirección física     Tipo
10.0.2.1   52-54-00-12-35-02  dinámico
10.0.2.255 ff-ff-ff-ff-ff-ff  estático
224.0.0.22  01-00-5e-00-00-16  estático
224.0.0.252 01-00-5e-00-00-fc  estático
224.0.0.253 01-00-5e-00-00-fd  estático
239.255.255.250 01-00-5e-7f-ff-ff  estático
255.255.255.255 ff-ff-ff-ff-ff-ff  estático
```

**Figura 5.38**  
Ejecución del comando arp –a  
en Microsoft Windows.

Los comandos con los que tradicionalmente se han monitorizado las interfaces de red han sido *ipconfig* (en Windows) e *ifconfig* (sustituido por *ip* y *ss* en Linux). Recomendamos emplear la ayuda en Microsoft Windows mediante *ipconfig -h*.



Otro de los comandos más empleados para comprobar una conexión de red es mediante el comando *ping*. Usado tanto en Windows como en Linux, este comando envía paquetes de prueba a un destino especificado y nos informa del tiempo de respuesta, en caso de existir conexión. Su sintaxis es la siguiente:

*ping [opciones] destino*

Donde *destino* es un nombre de dominio o la dirección IP. Para detener la salida por pantalla de los tiempos de respuesta, se utiliza la combinación de teclas *CTRL+C*.

Gracias a él, podemos comprobar si un adaptador de red funciona correctamente o si se tiene acceso a otros equipos dentro de la red local o fuera de ella (Internet, por ejemplo). Con esto, podemos descartar multitud de errores y localizar un posible problema.

```
luis@luis-VirtualBox:~$ ping sintesis.com
PING sintesis.com (193.70.32.179) 56(84) bytes of data.
64 bytes from diana.binpar.com (193.70.32.179): icmp_seq=1 ttl=51 time=43.4 ms
64 bytes from diana.binpar.com (193.70.32.179): icmp_seq=2 ttl=51 time=97.2 ms
64 bytes from diana.binpar.com (193.70.32.179): icmp_seq=3 ttl=51 time=56.0 ms
64 bytes from diana.binpar.com (193.70.32.179): icmp_seq=4 ttl=51 time=40.8 ms
^C
--- sintesis.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 8ms
rtt min/avg/max/ndev = 46.847/59.352/97.178/22.581 ms
```

**Figura 5.39**  
Ejecución del comando ping sobre síntesis.com.



Existen otras muchas aplicaciones que permiten realizar tareas de comprobación, monitorización, escaneo, verificación, auditoría, etc., entre las que destacamos *Wireshark*, *Pandora FMS* y *Tcpdump*.

Un complemento al comando *ping* para diagnosticar fallos de interconexión de redes es el comando *traceroute* o *tracert*, en Ubuntu y Microsoft Windows respectivamente. Este permite conocer la ruta que sigue un paquete en la red (desde un origen IP a un destino IP), comprobando el estado de esta, los routers por los que pasa y localizar un posible fallo de conectividad.

Su sintaxis es la siguiente:

- En Microsoft Windows: *tracert servidor\_destino*
- En Ubuntu: *traceroute servidor\_destino*

Otra herramienta ampliamente utilizada para realizar auditorías y funciones de seguridad de redes es *nmap*. Es un programa de código libre. En Linux se puede instalar a través de la línea de comandos y en Microsoft Windows, descargando e instalándolo desde *nmap.org*. Es una utilidad potente que permite escanear la red para inventariarla y monitorizarla.

`nmap [opciones] objetivos`



### Recurso digital 5.3

Ejemplos de *nmap*.

#### Actividad propuesta 5.7



Descarga *Wireshark* desde [wireshark.org](http://wireshark.org) e instálalo. Trata de realizar un escaneo de paquetes o tramas del tráfico de red.

#### 5.10.1. Gestión de puertos

En los sistemas informáticos en red, el término *puerto* puede hacer mención a:

- a) Puerto físico: entrada o conector de un dispositivo de red al que se conecta un medio de comunicación. Como, por ejemplo, un puerto RJ-45.
- b) Puerto lógico: número que se asocia a la aplicación de origen o destino de una comunicación. Son empleados por la capa de transporte, donde los segmentos especifican:
  - Puerto de origen: número que identifica la aplicación que origina la comunicación en el host.
  - Puerto de destino: número asociado a la aplicación de destino en el host remoto.

Por tanto, en las comunicaciones TCP y UDP los hosts han de indicar en el encabezado de cada segmento de la capa de transporte el número de puerto de origen y el número de puerto de destino.

Existen tres tipos de puertos lógicos asociados a su número:

1. Puertos bien conocidos: números de 0 al 1023. Reservados para aplicaciones y servicios como HTTP (80), FTP (20), HTTPS (443), SMTP (25), etc.

2. Puertos registrados: números del 1024 al 49151. Son puertos empleados por las aplicaciones de usuario cuando conectan con servidores. En este rango se incluyen números de puertos registrados por la IANA para determinadas aplicaciones.
3. Puertos dinámicos, privados o efímeros: números 49152 al 65535. Son utilizados principalmente por aplicaciones de intercambio de archivos punto a punto.

De esta manera, si un usuario desea acceder a una página web, el proceso sería el siguiente. El host cliente indica en el encabezado del segmento de la capa de transporte el puerto de destino bien conocido 80 (ya que es un servicio HTTP) y, como puerto de origen, un número aleatorio a partir del 1024. De esta manera, se pueden establecer simultáneamente multitud de comunicaciones sobre un mismo servidor HTTP. Cuando el servidor se comunica con el cliente, este indica en el encabezado del segmento su puerto origen 80 y puerto destino el correspondiente a la aplicación y comunicación concreta del host cliente.

Como ya sabemos, los segmentos se encapsulan dentro de paquetes de la capa de red. Y en el encabezado de los paquetes de la capa de red se indican las direcciones IP de origen y destino. A la combinación de una dirección IP y un puerto se le denomina *socket*. Por tanto, una comunicación entre dos hosts viene establecida por una pareja de sockets.

Un ejemplo de un socket es 192.168.1.55:80, formado por la dirección IP 192.168.1.55 y puerto 80. Este socket indica que pertenece a un servidor http, al ser un puerto bien conocido.

Los comandos que permiten monitorizar los puertos, sockets o conexiones de un sistema son *netstat* (Windows) y *ss* (GNU/Linux). Sus opciones son similares.

En sistemas GNU/Linux, el comando *ss* viene a sustituir a *netstat*. Son muchas las utilidades de este comando y presenta la siguiente sintaxis:

```
ss [opciones] [filtro]
```

Algunas de las acciones más habituales para realizar con *ss* son:

- ✓ Mostrar información sobre las conexiones asociadas a los sockets *ss -a*.
- ✓ Listar los sockets en escucha de nuestro host: *ss -l*.
- ✓ Información de sockets TCP: *ss -t*.
- ✓ Recoger estadísticas: *ss -s*.

## 5.11. Resolución de problemas

La infraestructura de red y los sistemas informáticos en red requieren un mantenimiento que permita evitar averías o fallos y, si estos ocurren, actuar de manera planificada. El mantenimiento debe abordarse desde tres ámbitos:

- Predictivo: se intenta pronosticar un futuro fallo para lo que se suelen emplear utilidades de diagnóstico.
- Preventivo: se lleva a cabo un *Plan de mantenimiento preventivo*, donde se detallan las acciones, técnicas y procedimientos que realizar, así como su frecuencia
- Correctivo: se repara el objeto del fallo, siguiendo un *Plan de mantenimiento correctivo* que establece el método para diagnosticar y resolver averías.

Ya conocemos herramientas de diagnóstico donde podemos monitorizar y testear el estado y comunicación de los dispositivos de red: *ping*, *ifconfig* (*ipconfig* en Microsoft Windows), *ss* (*netstat* en Microsoft Windows), *lshw*, etc. Además de otras herramientas analizadas en capítulos anteriores, como los monitores de rendimiento y administradores de dispositivos que permiten estudiar el estado de los dispositivos de red y su rendimiento (con el Administrador de tareas de Microsoft Windows o el Monitor del sistema GNOME en Linux).

### Actividad propuesta 5.8



En Microsoft Windows y Ubuntu, accede al Administrador de tareas de Microsoft Windows y al Monitor del sistema, respectivamente, navegando por sus diferentes opciones y familiarizándote con ellas.

También es recomendable apoyarse en aplicaciones especializadas para comprobar otros aspectos más concretos, según las necesidades de mantenimiento o resolución de averías. Ejemplo de ello es *Wireshark*, muy útil para analizar diferentes protocolos y filtrar el tráfico de la red en busca de vulnerabilidades. Así como otras muchas aplicaciones, que ayudan a planificar y resolver posibles problemas en entornos Wi-Fi, como, por ejemplo, *WiFi Analyzer*, *WiFi HeatMap* y *NetSpot*.

Los fallos o averías en el funcionamiento de los sistemas informáticos en red pueden ser muy variados. La experiencia y una planificación adecuada, mediante un *Plan de mantenimiento correctivo* adecuado, ante cualquier incidencia es fundamental para detectar y solventar el problema de la manera más eficiente posible.

Los principales fallos los podemos agrupar en:

#### CUADRO 5.6

#### Fallos de los sistemas informáticos en red más comunes

Fallos	Comprobaciones	
Fallos en hosts		
Fallos en la tarjeta de red	Tarjeta averiada	Probar otra tarjeta de red en el equipo.
	Tarjeta mal instalada	Comprobar la correcta instalación hardware y software, mediante drivers adecuados al sistema operativo.
Fallos en la configuración de la tarjeta de red	Configuración TCP/IP inadecuada	Revisar los valores: dirección IP, máscara de red, Gateway y DNS. En su caso, habilitar la opción DHCP.
	Configuración Wi-Fi inadecuada y baja señal	Comprobar el tipo de autenticación Wi-Fi y la contraseña. Testear la cobertura de la señal inalámbrica, debiendo ser adecuada la finalidad de la red, sin verse mermada por ruido electromagnético o una mala ubicación del punto de acceso o el dispositivo inalámbrico.

[.../...]

**CUADRO 5.6**  
 (cont.)

<b>Fallos en el medio</b>	
Fallos en cableado	Chequear que no se sobrepasa el radio de curvatura máximo y que no se encuentra forzado, aplastado o roto. Cerciorarse que el tipo de cableado es adecuado al ruido electromagnético del entorno. En cables de fibra óptica, la pérdida de señal ha de ser la mínima posible en el proceso de fusión.
Fallos en conectores	Revisar que los conectores y puertos no están forzados y sueltos. Los cables han de estar bien engastados en su interior. Comprobar el mapa de cableado, según los estándares TIA/EIA para cobre de par trenzado.
<b>Fallos en la electrónica de red</b>	
Configuración inadecuada de puntos de acceso Wi-Fi	Revisar la configuración de la autenticación Wi-Fi, filtros MAC, SSID oculto, DHCP, conjunto de direcciones estáticas, etc. Debe estar correctamente conectado con el sistema de distribución.
Problemas en switches	Chequear que el switch está encendido, con conectividad por los indicadores de estado led de cada puerto y a una temperatura de trabajo óptima.

**RECUERDA**

Las herramientas hardware más utilizadas para comprobar los diferentes medios de transmisión de datos son:

- ✓ Certificadora de fibra óptica y cobre. Herramienta muy completa que permite medir la pérdida de potencia de la fibra, detectar distancias exactas a cortes de fibra óptica, comprobación y certificación de redes de cobre y fibra óptica y estudiar otros muchos parámetros.
- ✓ Inspector de fibra óptica. Muestra el estado de conectores de fibra óptica.
- ✓ Analizador de cableado de cobre de par trenzado. Consta de dos módulos, uno transmite un pulso eléctrico por un extremo y el otro módulo permite seguirlo, sin necesidad de desconectar cables, al pasar por encima de estos y emitir un sonido.
- ✓ Analizador de redes inalámbricas. Realiza un estudio del estado de la señal inalámbrica: pruebas de conexión, vulnerabilidades, cobertura, etc.



**Figura 5.40**  
Inspector de fibra óptica.

## 5.12. Seguridad en las comunicaciones

La seguridad de los sistemas informáticos está íntimamente asociada a la seguridad en las comunicaciones, ya que hoy en día uno está integrado en el otro.

Para que las comunicaciones entre sistemas sean seguras, estas han de basarse en cuatro pilares fundamentales:

1. Los accesos a la información, a los sistemas y a los recursos han de ser *confidenciales*, es decir, solo se permite el acceso a aquellos usuarios o procesos autorizados.
2. La información o los recursos han de estar *disponibles* para los usuarios o procesos con permisos.
3. La modificación de la información o recursos debe realizarse por procesos o usuarios autorizados, de esta manera, se garantiza la *integridad* de los mismos.
4. Se debe garantizar la *autenticidad*. Para ello, se tiene que confirmar la identidad del emisor y del receptor:
  - El emisor debe asegurar al receptor que los datos han sido enviados por él.
  - El receptor debe asegurar al emisor que los datos han sido recibidos por él.

Para lograrlo, primero se deben establecer unas políticas de seguridad a partir de planes de contingencia y seguridad.

### 5.12.1. Políticas de seguridad

En cualquier sistema informático, se necesitan establecer políticas o planes de seguridad basados en los elementos que se van a proteger. Para ello, periódicamente se debe realizar un *análisis de riesgos*, donde se evalúen los recursos, la infraestructura de red y los sistemas, estableciendo sus puntos débiles. Basándose en los anteriores, se definirán *planes de contingencia y seguridad* centrados en los pilares de la seguridad en las comunicaciones: *confidencialidad, disponibilidad, integridad y autenticidad*.

En entornos empresariales e instituciones públicas o privadas, se deben difundir las políticas de seguridad y aquellas normas o procedimientos necesarios para que todos los usuarios conozcan los criterios y métodos para abordar la seguridad. Las más destacables son:

- ✓ Políticas de contraseñas.
- ✓ Política de actualizaciones.
- ✓ Política de uso de correo electrónico.
- ✓ Políticas de aplicaciones permitidas.
- ✓ Política de uso de conexiones externas.
- ✓ Políticas de almacenamiento y copias de seguridad.
- ✓ Políticas de uso de portátiles corporativos.
- ✓ Políticas de dispositivos personales.

En ellas se detallan aspectos de seguridad, como:

- Empleo de contraseñas robustas y actualización periódica de las mismas.
- Uso de aplicaciones conocidas y actualizadas.
- Actualización y mantenimiento de cuentas de usuarios.

- No difusión de cuentas y contraseñas a terceros.
- No ejecución de aplicaciones desconocidas o sin verificar procedentes del exterior: email, medios de almacenamiento externos, red externa, etc.
- Actualización y mantenimiento activo del sistema operativo y las aplicaciones.
- Creación y mantenimiento de las copias de seguridad.
- Monitorización de la red.
- Protección antimalware.
- Control de acceso físico a los sistemas y medios de red.
- Configuración segura de las redes inalámbricas.

### 5.12.2. Tipos de ataques

Un ataque informático es una acción ofensiva y deliberada que intenta tomar información, dañar, alterar, desestabilizar o destruir datos, información o sistemas informáticos independientes o en red.

Se distinguen los ataques activos (aquellos que ocasionan cambios en la información o los recursos) de los pasivos (que simplemente monitorizan, registran o acceden a los recursos, sin alterar estos o su información). Los ataques más usuales son:

- a) Reconocimiento y detección de vulnerabilidades en los sistemas. Tratan de obtener información del sistema, sin provocar daño alguno, de vulnerabilidades para su posterior explotación.
- b) Interceptación de información. Tratan de interceptar datos enviados en red, vulnerando la confidencialidad.
- c) Modificación de información. Reenvían mensajes o documentos alterados de manera premeditada que han sido previamente interceptados. De esta manera, se vulnera la integridad, autenticidad y confidencialidad.
- d) Suplantación de identidad. Son muchos los tipos de ataques de este tipo que afectan a la integridad, autenticidad y confidencialidad, como, por ejemplo:
  - Capturas de cuentas de usuario y contraseñas.
  - SMTP Spoofing: envío de emails con remitentes falsos o suplantando su identidad.
  - IP Spoofing: envío de paquetes IP desde un host distinto al que realmente lo ha enviado.
  - DNS Spoofing: direccionamiento erróneo de nombres de dominio.

### 5.12.3. Mecanismos de seguridad en las comunicaciones

Con objeto de proteger las comunicaciones, se pueden emplear un conjunto herramientas variadas. Su uso o aplicación dependerá de las necesidades establecidas en los planes de contingencia y seguridad. Para lograrlo, podemos utilizar un canal seguro de comunicaciones, o que el mensaje en sí sea seguro. Por tanto, las herramientas más empleadas son:

- ✓ Filtros de contenido. Software que se encarga de gestionar, restringir y limitar el acceso a sitios web con el propósito de evitar contenidos maliciosos o de dudosa intención.
- ✓ Redes privadas virtuales o VPN. Consisten en la creación de una extensión de una red local a través de una red pública (como Internet), de tal manera que se pueda establecer

una conexión virtual segura punto a punto. Se pueden realizar desde los propios sistemas operativos o mediante aplicaciones específicas. Son muy amplios sus usos:

- Navegar de manera anónima.
  - Descargas P2P.
  - Obtener un extra de seguridad en las comunicaciones.
  - Teletrabajo.
- ✓ Cortafuegos o firewall. Son herramientas (hardware o software) que controlan el tráfico entrante y saliente. Limitan o bloquean el tráfico externo, normalmente de Internet, para evitar accesos no autorizados a otra red (generalmente privada).
- ✓ Software antimalware.
- ✓ Herramientas de cifrado. Permiten cifrar datos a través de redes inseguras de tal manera que se garantice la confidencialidad, autenticidad e integridad. Consiste en aplicar un algoritmo que transforme un mensaje a partir de una clave. Existen dos tipos de cifrado:
1. Cifrado simétrico: solo se emplea una clave para cifrar y descifrar. Aporta confidencialidad.
  2. Cifrado asimétrico: se emplean dos claves, una privada y otra pública. A partir de la clave pública no se puede averiguar la clave privada. Aporta autenticidad, integridad y no repudio (no se puede negar la recepción o envío de un mensaje). La clave pública cifra el mensaje y la clave privada lo descifra. Únicamente el emisor posee su clave privada y hace llegar su clave pública a aquellos que quieran comunicarse (cifrando el mensaje) con él.
- ✓ Empleo de protocolos seguros, como SSL/TLS, OpenSSL o GnuTLS, HTTPS, SFTP, etc.



#### Recurso digital 5.4

Modos de red en VirtualBox.

### Resumen

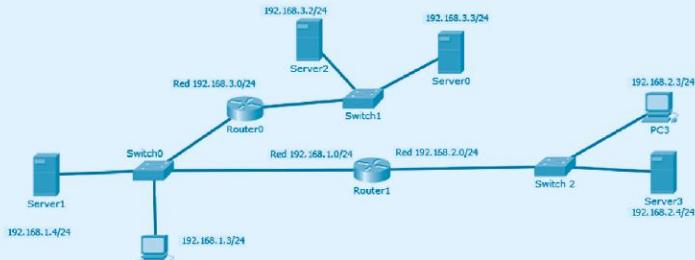
- El modelo de referencia TCP/IP constituye el modelo el estándar abierto de Internet. Existe una correspondencia clara entre capas de este modelo y el modelo de referencia OSI. Hemos estudiado algunos de los protocolos más importantes, como:
  - Protocolo Ethernet definido por el estándar IEEE 802.3.
  - La familia de protocolos IEEE 802.11, que definen los estándares Wi-Fi.
  - Protocolo IPv4 e IPv6, que actúa sobre la capa de red del modelo OSI y establece la forma de asignar direcciones IP (direcciones lógicas) para poder enrutar paquetes entre hosts.

- Los protocolos TCP y UDP se encargan de establecer comunicaciones entre aplicaciones de host de origen y de host de destino, enviando y recibiendo datos entre ellas, sin importar los medios de transmisión, las rutas de los datos, las congestiones, los tipos de hosts, etc. El protocolo TCP es confiable, mientras que el protocolo UDP no lo es, primando la rapidez en la entrega.
- La configuración del protocolo TCP/IP en adaptadores de red Ethernet e inalámbricos puede realizarse de manera estática o dinámica, según las necesidades de gestión del host y el diseño de la red.
- Para la interconexión de redes cableadas, los medios más empleados son el cable de cobre de par trenzado y la fibra óptica. Los dispositivos más empleados son switches y routers, pertenecientes a la capa de enlace de datos y de red, respectivamente, del modelo OSI. Tanto hosts como routers utilizan tablas de enrutamiento para encaminar paquetes a otros dispositivos de redes locales o remotas.
- Las redes inalámbricas aportan ventajas con respecto a las redes cableadas, como la movilidad, la flexibilidad y la facilidad de instalación. Para lograrlo, las redes inalámbricas emplean ondas electromagnéticas para transmitir datos. Las más usadas son las redes Wi-Fi, WiMAX, los sistemas de comunicación móviles 4G y 5G, así como otras redes WPAN, como Bluetooth o Zigbee. Cada una dispone de unas características que las hacen más apropiadas para según qué aplicaciones o usos.
- Además, según el tipo de organización de nuestro sistema informático, dispondremos de varios tipos de redes. Para diseñar estas redes, se emplean mapas de topología física y lógica. Sobre ellos, podremos analizar si la comunicación es la correcta, o mejorarla, para más tarde implementarla.
- La comunicación entre redes LAN remotas se realiza mediante diferentes tipos de conexiones WAN, que se clasifican en privadas (como commutación de circuitos, commutación de paquetes y dedicada) y públicas (como DSL, FTTH, HFC e inalámbricas).
- Los sistemas operativos integran multitud de funcionalidades, herramientas y comandos capaces de gestionar y monitorizar los adaptadores de red y el flujo de datos por estos. Ejemplo de ello son las herramientas *NetPlan*, *ip*, *ss*, *ping* en GNU/Linux, e *ipconfig* y *netstat* en Windows, entre otras muchas.
- Por último, se han indicado los pilares básicos de seguridad en las comunicaciones: confidencialidad, disponibilidad, integridad y autenticidad. Para lograrlos, en cualquier sistema informático se deben establecer unas políticas de seguridad a partir de planes de contingencia y seguridad. Los mecanismos de seguridad más empleados para lograr comunicaciones seguras son filtros de contenidos, VPN, firewalls, herramientas de cifrado y software antimalware.



## Ejercicios propuestos

- 1.** Dado un adaptador de red con una dirección IPv4 192.168.110.21/26. Indica:
  - a) Direcciones mínima y máxima asignables a hosts.
  - b) Dirección de broadcast de la red donde se encuentra.
  - c) Dirección de red donde se encuentra.
  - d) Representación de la dirección del adaptador de red en IPv6.
- 2.** Configura el adaptador de red (Ethernet o Wi-Fi) de tu equipo con una dirección IPv4 estática válida (no usada), que permita la comunicación con otros equipos en red a través de un switch. Justifica los datos de la nueva configuración: dirección IP, máscara de red, puerta de enlace y direcciones DNS. Comprueba la nueva configuración y su comunicación con otros equipos.
- 3.** Muestra:
  - a) La tabla de enrutamiento.
  - b) La tabla ARP.
  - c) Los puertos del sistema.
- 4.** Con nuestro smartphone, conéctate a una red Wi-Fi. Descarga e instala la aplicación WiFi Analyzer. Localiza el canal donde se sitúa la red Wi-Fi, analiza la cobertura Wi-Fi en varias estancias y estudia los canales con menos saturación.
- 5.** Realiza la configuración de un punto de acceso leyendo la guía de instalación y configuración del fabricante. El SSID ha de ser *Slunidad5* con método de autenticación WPA2 o WPA3 y una contraseña robusta.
- 6.** Dado el siguiente mapa lógico de una red de computadores:



**Figura 5.41**  
Ejemplo de diagrama lógico.

- a) Identifica todos sus componentes de red y describe sus funciones en el diseño.
- b) ¿Cuántas redes lógicas existen? ¿Por qué?
- c) Señala los dominios de colisión y difusión.

7. Descarga la aplicación de diseños estructurados *DIA* desde la web oficial <http://dia-installer.de/>. Este programa nos permite realizar multitud de diagramas, empleando hojas y objetos para diferentes propósitos. En nuestro caso, realizaremos un diseño lógico de una red de comunicación empleando las hojas de “Cisco – Red”, “Cisco – Comutador” y “Red”.

Realiza el diseño lógico de una red de computadores que disponga de dos subredes:

- a) Subred 192.168.1.0 para profesores. Esta red dispondrá de 5 equipos para profesores, un servidor y una impresora.
- b) Subred 192.168.2.0 para alumnos. Esta red dispondrá de 20 equipos para alumnos y un servidor.

Emplea el número mínimo de routers y switches para conectar todos los equipos y justifica la asignación de direcciones IP.

Recuerda que cada equipo, servidor o impresora debe disponer de una dirección IP (la añadiremos junto con el icono del objeto, editando un recuadro de texto). No pueden existir dos direcciones IP iguales y se reservan las direcciones más bajas de cada subred a los routers.

8. Modifica el archivo *hosts* en Ubuntu (*/etc/hosts*) o Microsoft Windows (*c:\Windows\system32\drivers\etc\hosts*), asociando una dirección IP con un nombre de dominio, con objeto de comprobar que dicho archivo tiene prioridad sobre la resolución DNS.

9. Disponiendo de un router SoHo con servidor DHCP, accede a su configuración y provee la configuración necesaria a los clientes DHCP, estableciendo un rango de direcciones asignables para diez hosts.

10. Disponiendo de dos puntos de acceso con la función WDS, realiza la configuración de infraestructura ESS entre ellos, ampliando la cobertura de la red Wi-Fi.

La función WDS (Wireless Distribution System) permite realizar dos acciones:

- a) Conectar dos dispositivos para comunicar redes diferentes (denominado *bridge*).
- b) Conectar clientes a la misma red Wi-Fi para extender su cobertura (objetivo de esta práctica).

Para ello, es recomendable:

- Que los puntos de acceso pertenezcan al mismo fabricante.
- Modificar las direcciones IP de los puntos de acceso y que estas sean diferentes dentro de la misma subred.
- Deshabilitar el servidor DHCP de los puntos de acceso secundarios.

## ACTIVIDADES DE AUTOEVALUACIÓN

1. ¿Qué capa del modelo de referencia OSI encapsula los paquetes de datos en tramas?:  
 a) Red.  
 b) Enlace de datos.  
 c) Física.
2. El protocolo IPv6:  
 a) Utiliza 32 bits.  
 b) Utiliza 64 bits.  
 c) Utiliza 128 bits.
3. Al configurar un adaptador de red de forma estática:  
 a) La dirección IP se mantendrá fija, por lo que no cambiará con el paso del tiempo.  
 b) No se podrá asignar una dirección IP pública.  
 c) No requiere especificar la máscara de red.
4. ¿Qué dispositivos de interconexión de redes pertenecen a la capa de nivel físico del modelo de referencia OSI?:  
 a) Switch y router.  
 b) Repetidor y hub.  
 c) Hub y switch.
5. El comando *traceroute* permite:  
 a) Comprobar una conexión de red enviando paquetes de prueba a un destino.  
 b) Conocer la ruta que sigue un paquete en la red.  
 c) Conocer las interfaces de red.
6. ¿Qué comandos permiten monitorizar los puertos de comunicaciones de un sistema?:  
 a) ss y netstat.  
 b) lshw y ss.  
 c) netstat y netplan.
7. Para la configuración de una red de área local:  
 a) Pueden emplearse, al mismo tiempo, direcciones IP estáticas y dinámicas.  
 b) La asignación de direcciones IP dinámicas no siempre requiere un servidor DHCP.  
 c) Solo puede establecerse sobre medios guiados.
8. ¿Cuáles de los siguientes son protocolos seguros de comunicaciones?:  
 a) Netplan, GnuTLS y SSH.  
 b) SFPT, ss, OpenSSL.  
 c) OpenSSL, SFTP, SSL/TLS.

9. La configuración de una red de área local inalámbrica BSS:

- a) Es similar al modo ad hoc.
- b) Puede integrar varios puntos de acceso.
- c) Solo existe un único punto de acceso, que ofrece unos servicios básicos para que los clientes se puedan comunicar en su zona de cobertura.

10. ¿Qué topología de red en edificios define el estándar TIA/EIA-568-B?:

- a) En bus.
- b) En estrella.
- c) En anillo.

**SOLUCIONES:**

1. **a** **b** **c**

2. **a** **b** **c**

3. **a** **b** **c**

4. **a** **b** **c**

5. **a** **b** **c**

6. **a** **b** **c**

7. **a** **b** **c**

8. **a** **b** **c**

9. **a** **b** **c**

10. **a** **b** **c**

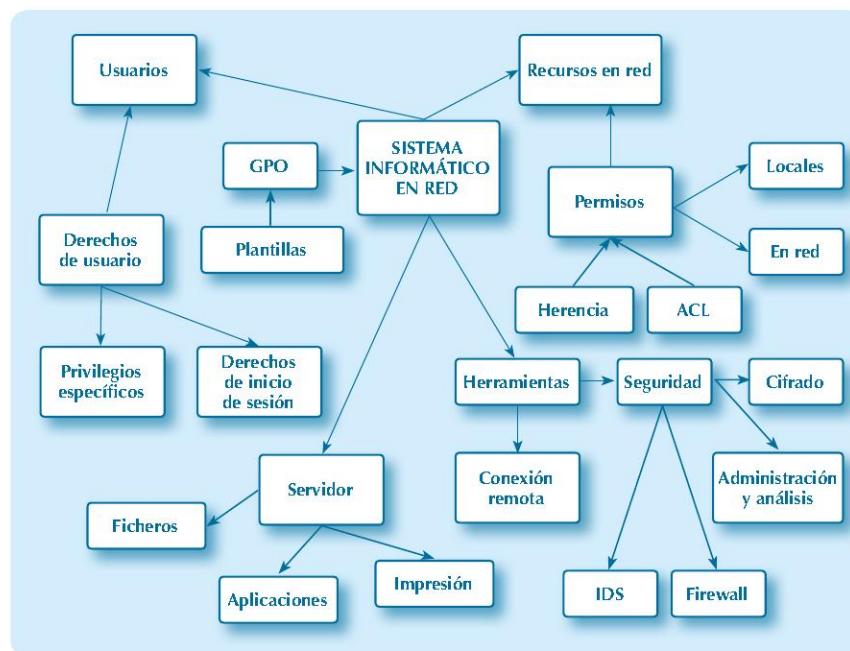
# 6

## Gestión de recursos en red de un sistema informático

### Objetivos

- ✓ Configurar el acceso a recursos locales y en red aplicando permisos locales y en red, herencia y listas de control de acceso.
- ✓ Diferenciar derechos de usuarios de permisos sobre los recursos.
- ✓ Establecer directivas de seguridad sobre usuarios y equipos.
- ✓ Identificar derechos de usuario y directivas de seguridad.
- ✓ Saber implementar y hacer uso de servidores de ficheros, servidores de impresión y servidores de aplicaciones.
- ✓ Acceder a equipos en red o servidores utilizando técnicas de conexión remota.
- ✓ Estudiar los requisitos de seguridad del sistema, usuarios y de los datos, instalando y evaluando utilidades de seguridad básica (herramientas de cifrado, cortafuegos, herramientas de análisis y administración y sistemas de detección de intrusión).

### Mapa conceptual



### Glosario

**ACL.** Listas de control de acceso que permiten flexibilizar los accesos o restricciones de usuarios y grupos de usuarios sobre los recursos.

**Cliente.** Equipo o software que hace uso de los servicios ofrecidos en red por un servidor.

**Cortafuegos (firewall).** Herramienta (hardware o software) que limita o bloquea el tráfico externo, normalmente de Internet, para evitar accesos no autorizados a otra red.

**Derechos de usuarios.** Son privilegios que determinan las acciones que pueden realizar usuarios o grupos de usuarios en un sistema.

**GPO.** Directivas de grupo que definen directivas de seguridad sobre usuarios y equipos.

**Herencia.** Característica de un objeto que permite tomar la misma configuración de permisos que su objeto antecesor.

**Herramienta de conexión remota.** Aplicación que permite acceder a otro equipo a distancia.

**IDS.** Sistema de detección de intrusión.

**MMC.** Consola de administración de Microsoft.

**Permisos en red.** Conjunto de permisos de un recurso compartido aplicados cuando este es compartido y accedido en red.

**Plantilla administrativa.** Configuración sobre el registro del sistema operativo para administrar aplicaciones de terceros o características del propio sistema operativo.

**Recurso compartido en red.** Recurso configurado para su uso o acceso por usuarios del sistema desde otros equipos en red.

**Servidor.** Software o hardware que ofrece servicios sobre recursos en red.

## 6.1. Introducción

Hoy en día y cada vez más, se ofrecen recursos en red con una gestión centralizada para así evitar duplicidades y mejorar el control sobre ellos. En este capítulo se aborda la gestión y explotación de los recursos en red por parte de los usuarios en condiciones de seguridad.

Comenzaremos a trabajar conceptos clave, como *permisos* o *recursos* (ya tratados en capítulos anteriores sobre sistemas GNU/Linux) con Microsoft Windows 10 Pro. Al tratarse de equipos en red, se abordarán términos y procedimientos necesarios, como permisos de red, herencia, ACL, recursos ocultos y compartir recursos, entre otros.

Estudiaremos el entorno Microsoft Windows, donde se han de conocer aspectos propios que determinan el modo en el que un usuario inicia sesión (derechos de inicio de sesión) y los derechos que posee un usuario una vez que este ha accedido al sistema (privilegios específicos). Estos términos son importantes a la hora de comprender el funcionamiento y la configuración de los sistemas operativos modernos en red. Para implementar estos derechos y privilegios, se emplean directivas de seguridad mediante directivas de grupo (GPO).

Para facilitar la aplicación de directivas de seguridad sobre usuarios o equipos de manera selectiva en Microsoft Windows, se emplea una herramienta llamada *consola de administración* de Microsoft o *MMC*, la cual ayudará en las tareas del administrador del sistema.

Más tarde, se estudiarán los procedimientos para configurar servidores, es decir, las herramientas software encargadas de compartir recursos. Dichas herramientas permiten administrar y gestionar el acceso a los recursos compartidos de manera más eficiente. Trataremos con algunos de los servidores más utilizados: servidores de ficheros, de impresión y de aplicaciones.

También emplearemos herramientas de conexión remota, ya que la mayoría de los recursos en red se ofrecen sobre servidores cuyo servicio no se interrumpe, y suelen encontrarse sin acceso físico directo. Por tanto, estos deben estar habilitados para acceder desde otros equipos a través de conexiones remotas. Ello facilita tareas de teletrabajo, soporte técnico, administración y análisis.

Por último, para garantizar la confidencialidad, autenticidad e integridad de los datos en red, se deben emplear herramientas de seguridad complementarias. Ejemplo de ello son las aplicaciones de cifrado sobre archivos o unidades de almacenamiento. También se deben emplear cortafuegos que ayuden a controlar el tráfico entrante y saliente de una red. Además, para detectar posibles intrusiones, se utilizan sistemas de detección de intrusión. Y para monitorizar de manera pormenorizada y centralizada toda la red, hosts y recursos, debemos disponer de herramientas de administración y análisis.

## 6.2. Permisos

Cuando un usuario intenta acceder al sistema, debe ser autenticado. Posteriormente, el usuario accede a los recursos siempre que esté autorizado. Para ello, se emplean los permisos como herramienta de protección de los recursos.

Los objetos son estructuras de datos que representan recursos, como archivos, carpetas, impresoras, procesos, etc., y todos ellos están protegidos contra el uso no autorizado gracias a los permisos. Los permisos definen el tipo de acceso concedido a un usuario o grupo de usuarios sobre un objeto.

En Microsoft Windows, los recursos constan de un propietario que concede o deniega permisos a usuarios o grupos de usuarios (conocidos como *Security Principals*). Los Security Principals se identifican por un valor *SID* (IDentificador de Seguridad único).

El propietario de los objetos es el creador del mismo. Este se puede modificar siempre que se posea el permiso de *Tomar posesión*. Por defecto, los administradores del sistema y el propietario disponen de dicho permiso activo. Por tanto, el propietario tiene el control total sobre el archivo.

Las acciones y permisos que pueden realizar los Security Principals sobre un recurso dependen del tipo de objeto. Ejemplos de permisos son: lectura, escritura, ejecución, modificación o control total. No obstante, los siguientes permisos son comunes a cualquier objeto: modificar, cambiar propietario, lectura y eliminar.

Los permisos básicos son los siguientes:

**CUADRO 6.1**  
Permisos básicos de carpetas y de archivos en Microsoft Windows

Tipos	Permisos	Descripción
Permisos en carpetas	Mostrar el contenido de la carpeta	Posibilita listar el contenido de la carpeta
	Lectura	Permite ver el contenido de la carpeta, permisos, propietario y atributos
	Escritura	Posibilita crear nuevos archivos y subcarpetas, ver el propietario, modificar atributos y permisos
	Lectura y ejecución	Permite navegar por las subcarpetas más los permisos de lectura y mostrar el contenido
	Modificar	Posibilita eliminar la carpeta más los permisos de lectura y ejecución
	Control total	Permite cambiar permisos, eliminar subcarpetas y archivos, tomar posesión y todos los permisos anteriores
	Permisos especiales	Se habilita cuando se activa uno de ellos
Permisos en archivos	Lectura	Permite ver el contenido del archivo, propietarios, permisos y atributos
	Escritura	Posibilita modificar su contenido y sus atributos, así como ver el propietario, permisos y atributos
	Lectura y ejecución	Permite ejecutar el archivo más el permiso de lectura
	Modificar	Posibilita modificar y eliminar el archivo más los permisos de escritura, y lectura y ejecución
	Control total	Permite cambiar permisos, tomar posesión más todos los permisos anteriores
	Permisos especiales	Se habilita cuando se activa uno de ellos

Para acceder a los permisos NTFS de un archivo o carpeta, desde el 'Explorador de Windows' pulsamos sobre 'Propiedades' del menú contextual, al hacer clic con el botón secundario sobre el archivo. Aparecerá una ventana cuyos permisos se detallan y se pueden modificar en la pestaña 'Seguridad'.

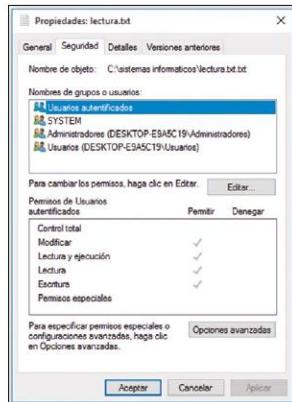


Figura 6.1  
Propiedades de un archivo.  
Pestaña Seguridad.

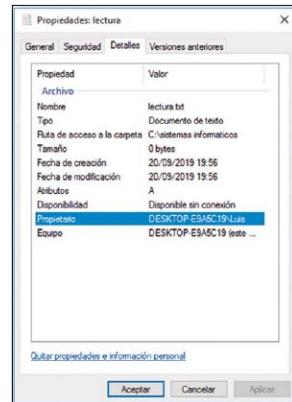


Figura 6.2  
Propiedades de un archivo.  
Pestaña Detalles.

Podemos acceder a las opciones avanzadas de seguridad de un archivo pulsando en 'Opciones avanzadas' de la pestaña 'Seguridad'.

En la figura 6.3 se aprecia la ruta del archivo, su propietario y las entradas de permisos por usuarios o grupos de usuarios. Se pueden ver o editar dichas entradas (si está deshabilitada la herencia).

Como vemos en la figura 6.4, a través de permisos avanzados (se pueden hacer visibles en 'Mostrar permisos avanzados'), se detallan otros permisos que posibilitan acciones más concretas sobre los objetos. Son los llamados *permisos especiales*:

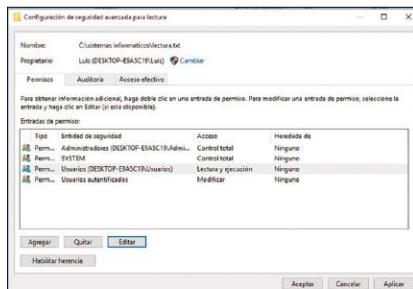


Figura 6.3  
Configuración de seguridad avanzada.

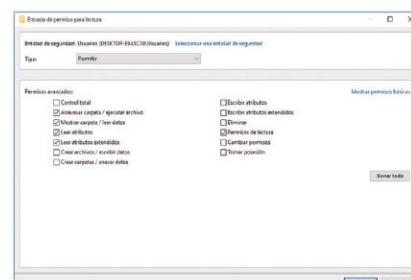


Figura 6.4  
Permisos avanzados.

**CUADRO 6.2****Permisos especiales en Microsoft Windows**

Permiso especial	Descripción
Atravesar carpeta/ ejecutar archivo	Posibilita moverse por carpetas, aunque no se tenga permiso de acceso. En archivos, permite su ejecución
Mostrar carpeta/ leer datos	Permite visualizar los nombres de ficheros y subcarpetas de una carpeta. En archivos, posibilita leer su contenido
Leer atributos	Permite ver los atributos de un archivo o carpeta como lectura y oculto
Leer atributos extendidos	Permite ver los atributos extendidos de un archivo o carpeta. Los atributos extendidos están definidos por los programas y pueden variar según estos
Crear archivos/ escribir datos	En carpetas, permite crear archivos. En archivos, permite modificar su contenido
Crear carpetas/ anexar datos	En carpetas, permite crear carpetas. En archivos, posibilita añadir datos sin modificar los existentes
Escribir atributos	Permite modificar los atributos del archivo o carpeta
Escribir atributos extendidos	Permite modificar los atributos extendidos del archivo o carpeta
Eliminar	Permite eliminar el archivo o la carpeta
Permisos de lectura	Permite leer los permisos del archivo o la carpeta
Cambiar permisos	Permite modificar los permisos del archivo o de la carpeta
Tomar posesión	Permite tomar posesión de un archivo o carpeta

Cuando un usuario forma parte de un grupo de usuarios, este tiene los mismos permisos del grupo sobre un objeto. Por eficiencia del sistema, es recomendable asignar permisos a grupos de usuarios, en lugar de a usuarios independientes, siempre que sea posible.

**TOMA NOTA**

- Se pueden definir los permisos NTFS solo en las unidades con formato NTFS.
- Los permisos no permitidos explícitamente están implícitamente denegados.
- Los permisos se suman cuando un usuario pertenece a distintos grupos.
- La denegación de permisos tiene preferencia sobre la concesión.
- Los permisos sobre ficheros prevalecen sobre los de carpeta.



### Actividad propuesta 6.1

Crea las carpetas y archivos que consideres oportuno para practicar con la concesión o denegación de permisos sobre ellos con varios usuarios. Comprueba las acciones de todos los permisos básicos y especiales.

#### 6.2.1. Permisos de red y locales

Hasta ahora, hemos tratado con permisos locales NTFS en un equipo. Independientemente de estos, cuando trabajamos en red con Microsoft Windows y se comparten archivos o carpetas, se les pueden asignar también permisos de red, aplicándose los más restrictivos en caso de conflicto.

Para aplicar permisos de red en carpetas, debemos seleccionar la pestaña 'Compartir' en las 'Propiedades' de la carpeta en el explorador de Windows. Indicaremos los usuarios o grupos de usuarios con quien compartir y el nivel de permiso sobre estos.

En 'Uso compartido avanzado' se puede modificar el nombre del recurso compartido y otras opciones más concretas, como:

- ✓ Indicar el número máximo de usuarios que pueden acceder simultáneamente al recurso compartido.
- ✓ Establecer un comentario sobre el recurso compartido para su correcta documentación.
- ✓ Asignar permisos a usuarios y grupos de usuarios a través de 'Permisos'.

De forma más sencilla, se pueden compartir archivos a través de la opción 'Compartir con' y 'Usuarios específicos', al pulsar el botón secundario del ratón sobre ellos en el 'Explorador de Windows'. Se deberá indicar con qué usuarios o grupos de usuarios se desea compartir y los permisos.



**Figura 6.5**  
Propiedades de un archivo.  
Pestaña Compartir.



**Figura 6.6**  
Uso compartido avanzado.

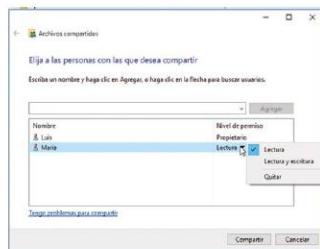


**Figura 6.7**  
Asignación de permisos  
sobre grupos y usuarios.

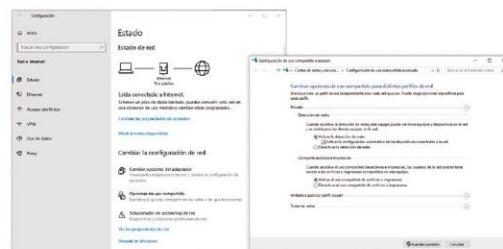
### 6.2.2. Compartir archivos o carpetas

A la hora de compartir archivos o carpetas entre distintos equipos en Microsoft Windows, suponemos que los usuarios disponen siempre de contraseña por seguridad, y debemos asegurarnos:

- ✓ Que los equipos se encuentren en la misma subred lógica establecida por el administrador de la red.
- ✓ Activar la detección de redes y el uso compartido de archivos. Para ello, hemos de acceder a 'Opciones de uso compartido' en 'Red e Internet' dentro de 'Configuración'.
- ✓ Que se encuentran en el mismo grupo de trabajo el equipo con el recurso compartido y el equipo que desea hacer uso de aquél. Comprobados o realizados los puntos anteriores, comprobar que los equipos tienen conexión en ambas direcciones mediante ping (para lo que debemos conocer las direcciones IP).



**Figura 6.8**  
Selección de permisos sobre usuarios en archivos compartidos.



**Figura 6.9**  
Configuración de uso compartido avanzado.

Para acceder a un recurso compartido, podemos hacerlo a través de:

- 'Red' del 'Explorador de Windows'. En 'Red' aparecerán los equipos accesibles desde el equipo actual. Podemos acceder a ellos introduciendo las credenciales, para más tarde acceder a sus recursos de red compartidos.
- Su especificación en formato UNC (convención de nomenclatura universal). Es decir, a través del 'Explorador de Windows' podemos indicar, mediante el siguiente formato, el acceso al recurso:

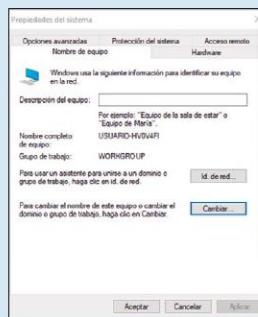
`\nombreEquipo\nombreRecurso`

- Una unidad asignada. Mediante la opción 'Conectar a una unidad de red', al pulsar con el botón secundario del ratón sobre 'Red' del 'Explorador de Windows', se configura el acceso a un equipo y a una ruta determinada. Se conectará al equipo remoto con las credenciales usadas actualmente; si se desean cambiar, se debe marcar 'Conectar usando otras credenciales'. El recurso asignado será visible en el panel 'Carpetas' del 'Explorador de Windows'.

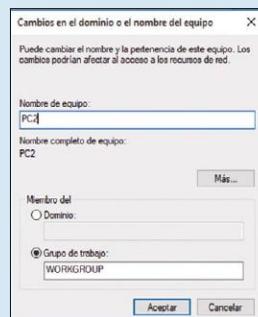
## RECUERDA

- ✓ El grupo de trabajo es un mecanismo empleado por Microsoft Windows para compartir recursos entre equipos de igual a igual. Podemos identificar el nombre del equipo y del grupo de trabajo a través de la información del sistema.

Para cerciorarnos sobre el nombre del grupo de trabajo, debemos acceder a 'Propiedades del sistema' en la pestaña 'Nombre de equipo' y, si es necesario, pulsando en 'Cambiar' para que todos los equipos pertenezcan al mismo grupo de trabajo (por defecto, WORKGROUP). También se puede modificar el nombre del equipo.



**Figura 6.10**  
Propiedades del sistema.



**Figura 6.11**  
Asignación del nombre  
de equipo y grupo de trabajo.

Se pueden ver los recursos compartidos de un equipo especificando `\localhost` en la barra de navegación del 'Explorador de Windows'. No obstante, y para un mayor detalle y gestión de todos los recursos compartidos, podemos acceder a 'Recursos compartidos' (dentro de 'Carpetas compartidas') en el 'Administrador de equipos' dentro de 'Herramientas administrativas' del 'Panel de control'.

De manera automática se comparten:

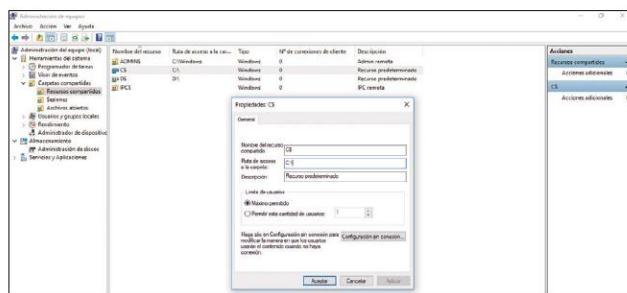
- Las unidades con el formato *letraUnidad\$*, como, por ejemplo, C\$ y D\$.
- El directorio del sistema cuyo nombre de recurso compartido es *ADMIN\$*, siendo normalmente la ruta C:\Windows.
- Agrupación de tuberías de comunicación de procesos mediante el recurso compartido con nombre *IPC\$*.



## Actividades propuestas

- 6.2.** Crea una carpeta compartida en Microsoft Windows y establece los permisos locales y de red necesarios para que un usuario desde otro equipo pueda acceder. Compruébalo accediendo desde otro equipo.
- 6.3.** Oculta el recurso compartido de la actividad anterior y compruébalo accediendo desde otro equipo.

Por seguridad, los recursos compartidos terminados en \$ quedan ocultos, aunque se puede acceder a ellos indicando el nombre del recurso compartido completo, es decir, con la terminación \$. Este es el motivo por el que no aparecen si especificamos \\localhost en el *Explorador de Windows*.



**Figura 6.12**  
Recursos compartidos de un equipo, incluidos los recursos compartidos ocultos.

### 6.2.3. Herencia

Cuando trabajamos con carpetas y subcarpetas, hablamos de *objetos primarios*, o *contenedores*, y objetos secundarios (subcarpetas y archivos dentro de la carpeta principal).

Microsoft Windows permite la herencia de permisos de objetos primarios a secundarios. A los permisos que se heredan de los objetos primarios se les conoce como *permisos heredados*. Siendo el propietario quien controla cómo se heredan los permisos.

Los permisos de carpetas o archivos pueden heredar, implícitamente, permisos de la carpeta que los contienen. No obstante, estos pueden tener, además, permisos explícitos (permisos establecidos directamente sobre ellos). La herencia de permisos es dinámica, por lo que la modificación de un permiso en una carpeta afectará a los archivos y carpetas que contenga.

En 'Opciones de seguridad avanzadas' podemos habilitar o deshabilitar la herencia de un contenedor pulsando sobre el botón 'Deshabilitar herencia'. Por defecto, se encuentra habilitada. Si se deshabilita la herencia, se puede optar por convertir los permisos heredados en explícitos o quitarlos. En cualquier caso, ya no afectarán las modificaciones de permisos del objeto primario sobre el secundario.

Además, la herencia de permisos se puede editar para cada entrada de permiso, seleccionando la entrada y pulsando en 'Editar' e indicando el tipo y a qué objetos les afectarán los permisos del objeto primario (ver figura 6.3).

#### TEN EN CUENTA

- ✓ Una carpeta con permisos explícitos más los permisos heredados de su carpeta contenedora podrán ser heredados por los archivos y carpetas que contenga.  
En caso de conflicto, los permisos explícitos tienen prioridad sobre los heredados.

#### 6.2.4. ACL

El sistema de archivos NTFS implementa los permisos utilizando *listas de control de acceso* (ACL). Estas listas contienen los usuarios, grupos y equipos que tienen acceso permitido al archivo o carpeta y qué tipo de acceso.

Cada objeto tiene asociada una ACL, donde se indican los permisos de usuarios y grupos de usuarios. Para cada usuario o grupo de usuarios con permisos (denegados o concedidos) establecidos sobre un objeto, existe una entrada de control de acceso (ACE) en su ACL.

Como hemos visto en los puntos anteriores, Windows representa gráficamente las ACL de un archivo o carpeta a través de la pestaña 'Seguridad' de 'Propiedades'.

Por otro lado, en GNU/Linux se pueden aplicar ACL para adecuar y hacer más flexible los accesos o restricciones de usuarios y grupos de usuarios sobre los recursos. De esta manera, se pueden aplicar permisos más flexibles a usuarios o grupos de forma diferente del establecimiento de permisos regulares.

**Recurso web**

Para saber más sobre ACL, puedes consultar el siguiente tutorial de la web de Ubuntu:

### 6.3. Derechos de usuarios

Los derechos de usuarios (también conocidos como *privilegios*) determinan las acciones que pueden realizar usuarios o grupos de usuarios en un sistema, ya sea en un equipo o en un dominio (administración centralizada de los recursos de una organización). A diferencia de los permisos, los derechos de usuario se asocian con usuarios y no con objetos.

Los derechos de usuarios se clasifican en:

- a) Derechos de inicio de sesión: determina de qué modo y quién inicia sesión en un sistema.
- b) Privilegios específicos: establecen los derechos de los usuarios una vez que han accedido al sistema, como, por ejemplo, realizar copias de seguridad de archivos y directorios, apagar el sistema, etc.

TOMA NOTA



Los derechos de usuarios prevalecen sobre los permisos de los objetos. Ejemplo de ello puede ser un usuario que no tenga permisos sobre un objeto, pero puede tener derecho para cambiar los permisos sobre dicho objeto.

Al igual que ocurría con los permisos, al asignar derechos de usuarios a grupos de usuarios, se asignarán automáticamente a los usuarios si son miembros de dichos grupos.

Los derechos de usuario se administran mediante la herramienta ‘Directiva de seguridad local’ de ‘Herramientas administrativas’, en el nivel ‘Asignación de derechos de usuario’, dentro de ‘Directivas locales’. O también mediante la herramienta ‘Editor de directivas de grupo local’ de las versiones avanzadas de Microsoft Windows 10, que incluye a ‘Directiva de seguridad local’ dentro de los niveles ‘Configuración de equipo’, ‘Configuración de Windows’ y ‘Configuración de seguridad’.

### 6.3.1. Directivas de seguridad. Objetos y ámbito de directivas

Las directivas de seguridad establecen un conjunto de reglas de seguridad para administrar usuarios y equipos. Se definen mediante objetos de directivas de grupo (GPO). Una GPO contiene parámetros que definen políticas del sistema. De esta manera, se pueden centralizar políticas sobre usuarios y equipos, estableciendo permisos, bloqueos o controles. Ejemplos de ello son:

- ✓ Configurar el entorno gráfico.
- ✓ Automatizar tareas.
- ✓ Evitar que un usuario pueda instalar y desinstalar programas.
- ✓ Fortalecer las contraseñas de inicio de sesión.
- ✓ Evitar el uso de memorias flash USB.

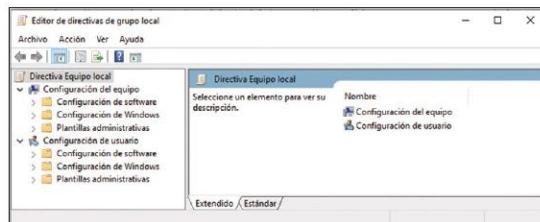
Podemos distinguir dos tipos:

1. GPO locales: utilizadas para los equipos que no forman parte de un dominio. Son las estudiadas en este capítulo.
2. GPO no locales: orientadas al servicio de directorio de Microsoft, llamado *Active Directory*.

Para la gestión de GPO se emplea el ‘Editor de directivas de grupo local’, el cual se puede ejecutar en las versiones avanzadas Microsoft Windows 10 mediante el ‘Editor de directivas de grupo’ (o también ejecutando *gpedit.msc*).

Según los objetos que configuran las GPO locales, estas se dividen en:

- a) Directivas de configuración del equipo: que agrupan las políticas de configuración a nivel de equipo. Las modificaciones se aplican en el arranque del sistema.
- b) Directivas de configuración de usuario: donde se agrupan las políticas de configuración a nivel de usuario del equipo. Las modificaciones se aplican en cada inicio de sesión de usuario.



**Figura 6.13**  
Editor de directivas de grupo local.



### Actividad resuelta 6.1

*Aplicar una GPO para prohibir el acceso a 'Configuración de PC' y 'Panel de control'.*

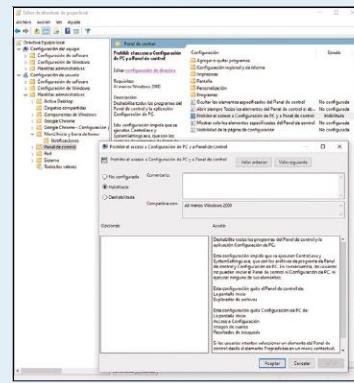
#### SOLUCIÓN

Cuando se desean establecer GPO, hemos de seleccionar el nivel donde se encuentra y, posteriormente, editar la GPO concreta. En este caso, dentro de 'Directiva de seguridad local', la GPO se encuentra en 'Configuración de usuario', 'Plantillas administrativas' y 'Panel de control'. En general, las opciones para aplicar son:

- No configurada: no se ha editado.
- Habilitada: se establece la GPO.
- Deshabilitada: se deshabilita la GPO.

Además, podemos establecer algún comentario sobre la opción tomada, como administradores del sistema. Es de especial relevancia tener en cuenta el apartado 'Compatible con', ya que determina las versiones de Microsoft con las que es efectiva la GPO seleccionada.

A partir de ahora, cuando un usuario intente acceder a la 'Configuración del PC' o al 'Panel de control', aparecerá un mensaje indicando que la acción ha sido cancelada debido a las restricciones especificadas para el equipo.



**Figura 6.14**  
GPO de prohibición de acceso  
a Configuración de PC y  
a Panel de control.

A su vez, cada una se divide en las siguientes partes (aunque con diferentes políticas):

- Configuración de software: permite la configuración del software ya instalado y la instalación automática de un nuevo software.
- Configuración de Windows: relacionado con el entorno de Windows, es decir, la configuración de seguridad (como la asignación de derechos de usuarios), la ejecución de scripts, etc.
- Plantillas administrativas: incluyen políticas basadas en la configuración y ajustes del equipo, como el inicio y apagado, el panel de control, la red, los componentes de Windows, etc. Por tanto, es considerada la categoría más importante por la gran cantidad de directivas que aglutina.

El ámbito de actuación de las GPO en Microsoft Windows 10 es en el propio equipo local. En caso de implementarse *Active Directory*, los ámbitos pueden ser más variados: sitio, dominio o unidad organizativa.

**Actividad resuelta 6.2**

Aplicar una GPO para que las contraseñas deban cumplir los requisitos de complejidad, es decir, ha de contener al menos tres de los siguientes tipos de caracteres: letras mayúsculas, letras minúsculas, dígitos, caracteres no alfanuméricos, otros caracteres Unicode y, además, no pueden contener el valor de la cuenta o el usuario.

**SOLUCIÓN**

Dentro del 'Editor de directiva de grupo', esta GPO se aplica en 'Configuración del equipo', 'Configuración de Windows', 'Configuración de seguridad', 'Directivas de cuenta', 'Directiva de contraseñas' y 'La contraseña debe cumplir los requisitos de complejidad'.

Para llevar un control de las GPO establecidas, podemos listar 'Todos los valores' dentro de 'Plantillas administrativas', tanto en 'Configuración de usuario' como en 'Configuración de equipo'. Y, además, desde el símbolo del sistema podemos ejecutar el comando `gpresult`, con diferentes opciones para mostrar la salida. Ejemplo de ello son las políticas aplicadas sobre un usuario concreto.

```
gpresult /USER usuario /V
```

O las políticas aplicadas al sistema en general:

```
gpresult /r
```

**Actividades propuestas**

- 6.4.** Averigua para qué sirve el servicio de directorio de Microsoft, el *Active Directory*, y sus ventajas frente a varios equipos locales conectados y compartiendo recursos. Indica en qué consisten los diferentes contenedores de un directorio activo: sitio, dominio o unidad organizativa.
- 6.5.** Aplica una GPO para que se oculte algún elemento especificado del panel de control.

### 6.3.2. Plantillas

Las plantillas administrativas permiten definir la configuración del registro de Microsoft Windows para administrar aplicaciones o características del propio sistema operativo, así como otras aplicaciones de terceros (paquete Office, Chrome, Adobe, etc.).

A las plantillas administrativas, ya descritas en el apartado anterior, se pueden agregar o quitar nuevas plantillas de aquellas aplicaciones sobre las que deseemos establecer GPO.



### Actividad resuelta 6.3

*Crear una GPO para que el navegador Mozilla Firefox esté configurado con una determinada página de inicio.*

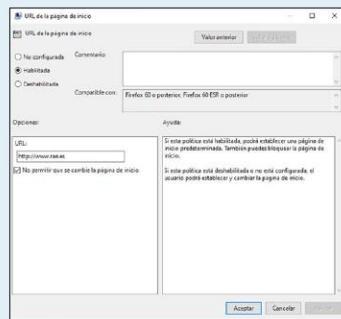
#### SOLUCIÓN

En primer lugar, debemos realizar una copia de seguridad de las plantillas administrativas, las cuales se almacenan en 'C:\Windows\PolicyDefinitions'.

Después, debemos descargar una plantilla administrativa estable de Mozilla Firefox (policy\_templates\_vX.X.zip) desde 'https://github.com/mozilla/policy-templates/releases' y descomprimirla.

A continuación, se añaden los ficheros de la carpeta descomprimida 'firefox.admx' y 'mozilla.admx' a 'C:\Windows\PolicyDefinitions', así como los propios de la subcarpeta es-ES descomprimida a 'C:\Windows\PolicyDefinitions\es-ES'.

Ahora, en el 'Editor de directivas de grupo local', ya podemos establecer las GPO desde 'Configuración de usuario', 'Plantillas administrativas', 'Mozilla', 'Firefox', 'Página de inicio'. Para ello, habilitamos la GPO 'Página de inicio' a su valor de 'Página de inicio' para que se cargue la página que se establezca en la GPO 'URL de la página de inicio'.



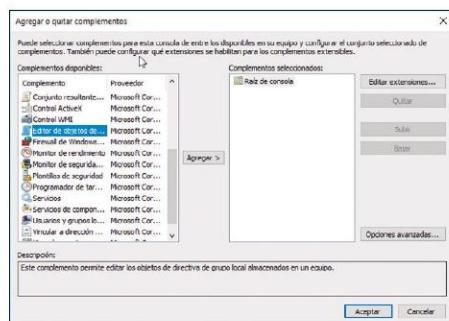
**Figura 6.15**  
GPO para establecer una página de inicio predeterminada.

## 6.4. Requisitos de seguridad del sistema y de los datos. Seguridad a nivel de usuarios y de equipos

Los recursos del sistema informático en red, así como los datos dependientes del él, han de quedar protegidos gracias a unos requisitos de seguridad concretos y definidos en las políticas de seguridad. En Microsoft Windows, estas políticas de seguridad se implementan, en gran medida, a través de los permisos sobre los recursos y las directivas de seguridad aplicadas a usuarios y equipos.

Gracias a la consola de administración de Microsoft (MMC), podemos aplicar directivas de seguridad sobre usuarios o equipos de manera selectiva. También permite gestionar cómodamente las 'Herramientas administrativas' de Microsoft Windows, es decir, podremos crear consolas personalizadas donde se reúnan diferentes herramientas administrativas al gusto del administrador del sistema.

La aplicación de GPO sobre equipos o usuarios de manera selectiva mediante la consola de administración de Microsoft (MMC) se realiza de la siguiente manera: abrimos la consola de administración de Microsoft a través del cuadro de búsqueda de Windows, escribiendo MMC. En el menú 'Archivo', seleccionamos 'Agregar o quitar complemento'. Aparecerá una gran variedad de complementos que pueden ser añadidos a la consola. En nuestro caso, agregaremos el 'Editor de objetos de directiva de grupo'.



**Figura 6.16**  
Complementos de la Consola de Administración de Windows.

Al pulsar en ‘Añadir’, se iniciará un asistente para directivas de grupo y deberemos pulsar en ‘Examinar’ para indicar el tipo de objeto.

A continuación, podremos indicar el equipo o los usuarios sobre los que deseamos aplicar la GPO. Para ello, en la pestaña ‘Equipos’ se puede seleccionar ‘Este equipo’ u ‘Otro equipo’ al pulsar en ‘Examinar’. Si decidimos aplicar la GPO sobre usuarios, debemos seleccionar los usuarios o grupos de usuarios sobre los que aplicarla en la pestaña ‘Usuarios’.

Una vez aceptamos y finalizamos el asistente, terminamos por aceptar la agregación de componentes y aparecerá la nueva configuración de la consola con el complemento. Por último, guardamos la nueva consola.

Gracias a la consola de administración de Microsoft, podemos administrar de forma centralizada, rápida y cómodamente distintas GPO a grupos, usuarios y equipos.

## 6.5. Servidores

Los servidores permiten compartir recursos en red. Un servidor hardware soporta el sistema operativo y el software que comparte los recursos, por lo que debe tener unas prestaciones acordes con los recursos que comparte y a las peticiones de acceso a dichos recursos.

Un servidor ha de cumplir con tres aspectos:

1. Disponibilidad. El sistema ha de estar activo y en funcionamiento el máximo tiempo posible para que los recursos estén accesibles a cualquier cliente.
2. Escalabilidad. Debe posibilitar el crecimiento del sistema al aumentar la demanda de trabajo.
3. Mantenimiento. El sistema ha de estar preparado para realizar tareas de mantenimiento sin mermar la disponibilidad del mismo.

### RECUERDA

- ✓ Los servidores se fundamentan en el *modelo cliente-servidor*, por el que un servidor ofrece sus servicios a diferentes clientes. El servidor queda continuamente a la espera de recibir peticiones por cualquiera de los clientes y, en caso de producirse, deberá dar respuesta a las mismas.

Existen multitud de servidores, según los recursos ofrecidos a los clientes. A continuación, se estudiarán distintos tipos.

### 6.5.1. Servidor de ficheros

Tiene como objetivo gestionar el almacenamiento controlado de ficheros sobre diferentes clientes. La centralización de los ficheros facilita la compartición de los mismos, gestionar los permisos de acceso, registrar las conexiones, realizar copias de seguridad y controlar diferentes versiones de un mismo fichero.

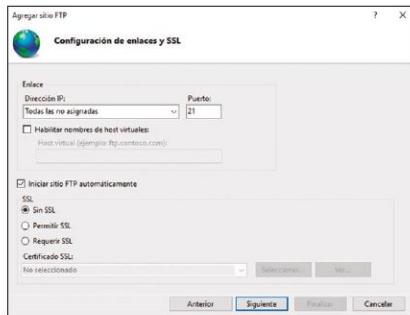
Estos servidores utilizan para transferir ficheros los protocolos FTP (File Transfer Protocol), FTPS (FTP mediante SSL para el cifrado de datos) y SFTP (SSH File Transfer Protocol) de mayor seguridad, principalmente.

#### A) Configuración de un servidor FTP en Windows

Microsoft Windows facilita la instalación de un servidor FTP mediante la activación de sus características. Para ello, debemos acceder a 'Programas' y 'Características' del 'Panel de control'. Pulsamos en 'Activar o desactivar las características de Windows' y expandimos 'Internet Information Services', activando las opciones de 'Servidor FTP' y 'Herramientas de administración web'. Al aceptar, comienza la instalación.

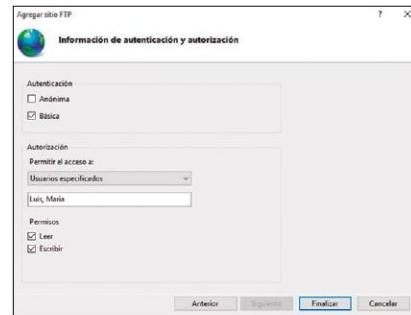
A continuación, se debe configurar el servidor FTP. Por lo que debemos crear un sitio FTP a través del administrador de 'Internet Information Services (ISS)' en 'Herramientas administrativas'.

En el panel izquierdo de conexiones, pulsamos con el botón secundario del ratón sobre 'Sitios' y seleccionamos 'Añadir sitio FTP'. Hemos de indicar un nombre para el sitio FTP y la ruta de una carpeta sobre la que accederán los clientes FTP.



**Figura 6.17**

Información de autenticación y autorización.



**Figura 6.18**

Configuración de enlaces y SSL.

Pulsamos en 'Siguiente' y mantenemos la configuración en el apartado 'Enlace'. Activamos 'Sin SSL' en el apartado 'SSL' (aunque en un entorno corporativo es muy recomendable requerir SSL).

Al pulsar en ‘Siguiente’, seleccionamos la forma de autenticación ‘Básica’ y podemos permitir la autorización de diferentes usuarios o grupos. En nuestro caso, indicamos dos usuarios con permisos de lectura y escritura.

Finalizamos el asistente y aparecerá el nuevo sitio FTP en el panel de conexiones.

#### TEN EN CUENTA

- ✓ Los accesos al servidor FTP pueden ser bloqueados por el firewall del sistema, por lo que se deben permitir accesos a este. Para ello, accedemos a ‘Firewall de Windows’ del ‘Panel de control’ y pulsamos sobre ‘Permitir una aplicación o una característica a través de ‘Firewall de Windows’’. Habilitamos ‘Servidor FTP’.
- Por otro lado, y si vamos a acceder al equipo servidor a través de conexiones externas sobre una red interna, debemos asignar una dirección estática y configurar el router Gateway para permitir dichas conexiones al equipo servidor mediante la redirección de puertos o *Port Forwarding*. El puerto empleado para las conexiones FTP es el 21.

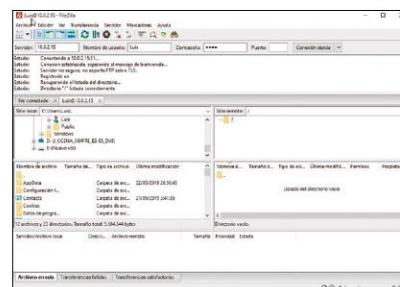
#### B) Cliente FTP

Desde el ‘Explorador de archivos’ de Microsoft Windows se puede acceder a servidores FTP. Para ello, basta con indicar la dirección IP del servidor con el siguiente formato: `ftp://direccionIP`.

A continuación, solicitará las credenciales de acceso de los usuarios que tengan permisos para acceder al servidor FTP.



**Figura 6.19**  
Acceso desde el Explorador de archivos de Microsoft Windows a un servidor FTP.



**Figura 6.20**  
Interfaz de Filezilla Client.

No obstante, es muy común utilizar aplicaciones clientes FTP, ya que aglutan multitud de herramientas que facilitan la transferencia de ficheros por FTP. Una de las más empleadas es Filezilla Client (<https://filezilla-project.org>). Su interfaz es muy sencilla.



## Actividad propuesta 6.6

Instala y configura un servidor FTP en una máquina virtual con Microsoft Windows. Desde otra máquina virtual con Microsoft Windows, accede a la anterior como cliente FTP.

### 6.5.2. Servidor de impresión

Las impresoras son recursos escasos, lentos y bastante solicitados en entornos corporativos, por lo que los servidores de impresión ayudan a gestionar todas las peticiones.

Los servidores de impresión hardware posibilitan la conectividad con impresoras, sin depender de host alguno. De esta manera, estos se conectan directamente a la red, haciendo de enlace con las impresoras. Según la localización de los servidores de impresión, distinguimos:

- ✓ Servidor externo: se encuentran fuera de la impresora, conectándose a través de un puerto de la misma.
- ✓ Servidor interno: situado en el interior de la impresora.

Los servidores de impresión hardware disponen de su propia guía de instalación, la cual se debe seguir para su correcta configuración.

No obstante, Microsoft Windows, en sus versiones avanzadas, dispone de herramientas (como ‘Administración de impresión’ que proporcionan la capacidad de gestionar impresoras de manera centralizada) para actuar como un servidor de impresión.

## TOMA NOTA



Para configurar Microsoft Windows como un servidor de impresión, el equipo debe estar encendido. Cuando se realice la conexión, los usuarios han de estar activos en el sistema y disponer de contraseña. Además, se deben encontrar en la misma red y han de estar habilitadas la detección de redes y la compartición de archivos e impresoras en la ‘Configuración de uso compartido avanzado’.

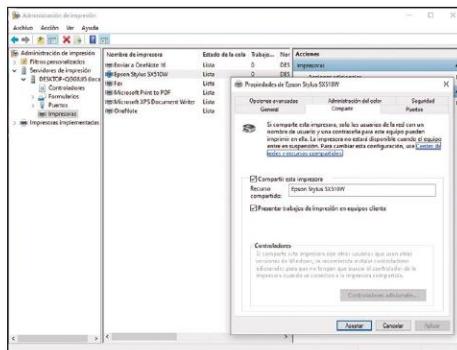
Para llevar la configuración en modo servidor de impresión sobre un equipo en Windows, se debe acceder al ‘Administrador impresión’ en ‘Herramientas administrativas’. Seleccionamos la impresora sobre la que se desea actuar, y se pulsa con el botón secundario del ratón sobre ‘Propiedades’. En la pestaña ‘Compartir’, debemos seleccionar ‘Compartir esta impresora’ e indicar el nombre del recurso compartido. En la pestaña ‘Seguridad’ indicamos los permisos de los usuarios y grupos sobre la impresora.

Además, en la pestaña ‘Opciones avanzadas’ podemos señalar varios aspectos de utilidad como la disponibilidad (pudiendo habilitar un horario concreto) o cuándo comenzar a imprimir.

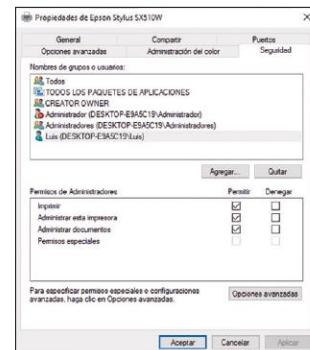
La centralización mediante este servidor de impresión posibilita, entre otras acciones:

- Agregar drivers (según la arquitectura del sistema operativo cliente) al servidor de impresión, para que nuevos clientes puedan emplear la impresora sin preocuparse por este aspecto.

- Agregar más impresoras.
- Modificar la carpeta de cola de impresión.



**Figura 6.21**  
Pestaña Compartir de las Propiedades de una impresora.



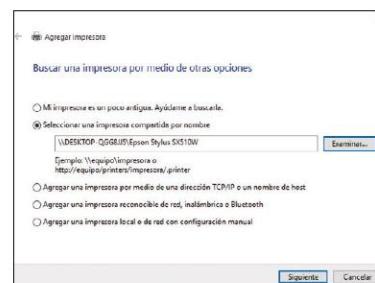
**Figura 6.22**  
Pestaña Seguridad de las Propiedades de una impresora.

### A) Clientes de impresión

En equipos Microsoft Windows, se configuran impresoras en red a través de servidores de impresión mediante 'Impresoras y escáneres' del menú 'Configuración'. Para lo que se pulsa en 'Aregar una impresora o un escáner' y, si no la localiza, pulsamos sobre 'La impresora que deseo no está en la lista'.

En este paso, debemos indicar, por alguna de las opciones que pone a disposición, el acceso al recurso compartido.

Posteriormente, pedirá las credenciales de un usuario con privilegios.



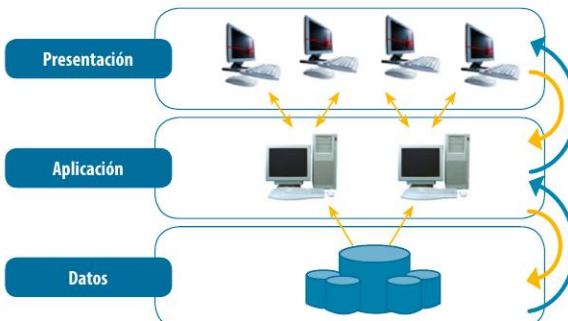
**Figura 6.23**  
Selección de una impresora compartida por nombre.

### 6.5.3. Servidor de aplicaciones

Los servidores de aplicaciones alojan y ejecutan programas a petición de los clientes a través de la red. Se encargan de gestionar la lógica de negocio de la aplicación, abriendo y centralizando sus operaciones. Por ello, los servidores de aplicaciones suelen emplearse cuando se requiere reducir la complejidad y el tamaño de las aplicaciones cliente, dirigir el flujo de datos para aumentar el rendimiento o controlar la seguridad de los datos.

Los servidores de aplicaciones se fundamentan en una arquitectura en tres capas:

1. Nivel 1: presentación o interfaz gráfica (GUI) del lado del cliente, a través de la cual los usuarios se conectan al servidor de aplicaciones (normalmente utilizando un navegador web u otra aplicación cliente específica).
2. Nivel 2: aplicación o servidor de aplicaciones encargado de ejecutar el procesamiento de la información y que actúa de intermediario entre los clientes y la capa de datos.
3. Nivel 3: datos o capa de datos, cuya misión es alojar el conjunto de datos necesarios para procesar las peticiones de los clientes.

**Figura 6.24**

Arquitectura de tres capas.

Los ejemplos más conocidos de servidores de aplicaciones son WebLogic de Oracle, WebSphere de IBM, WildFly, Apache Geronimo o GlassFish.

Microsoft Windows dispone de un conjunto de servicios orientados a servidores web que pueden ser activados a través de las características de Windows. Para ello, accedemos a 'Programas' y 'Características' del 'Panel de control'. Pulsamos en 'Activar o desactivar las características de Windows' y expandimos 'Internet Information Services', activando las opciones de 'Herramientas de administración web' y 'Servicios World Wide Web' (lo cual habilitará solo las características mínimas necesarias). Al aceptar, comienza el proceso de instalación.

Una vez instalado, podemos comprobar la correcta instalación escribiendo *localhost* en la barra de dirección de un navegador, lo cual mostrará la pantalla de presentación de IIS.

Accedemos a su gestión a través del 'Administrador de Internet Information Services (IIS)' en 'Herramientas administrativas'.



### Actividad propuesta 6.7

Instala y configura un servidor de aplicaciones en una máquina virtual con Microsoft Windows. Desde otra máquina virtual con Microsoft Windows, accede a la anterior como cliente web. Accede al 'Administrador de Internet Information Services (IIS)' e indaga sobre las distintas opciones de configuración.

## 6.6. Conexión remota. Herramientas

Las herramientas de conexión remota permiten acceder a equipos a distancia. Estas utilidades son muy empleadas para realizar tareas de soporte técnico, teletrabajo, presentaciones a distancia, etc.

La mayoría de sistemas operativos ofrecen herramientas propias que facilitan conexiones remotas. No obstante, existe multitud de software de terceros que ofrecen esta característica, junto con otras muchas utilidades que les aportan valor añadido. Algunos ejemplos característicos son TeamViewer o Chrome Remote Desktop.

Para poder conectarse remotamente a las versiones avanzadas de Microsoft Windows debemos acceder a 'Propiedades del sistema'. Dentro de él, seleccionamos la pestaña 'Acceso remoto'. En el área de 'Asistencia remota', al seleccionar 'Opciones avanzadas', podemos determinar el tiempo máximo de las conexiones y el permiso de control remoto. Y, en el área de 'Escritorio remoto', hemos de seleccionar 'Permitir las conexiones remotas a este equipo'. Además, se deben autorizar los usuarios que hagan uso del escritorio remoto (por defecto, el grupo de administradores tiene permiso) en 'Seleccionar usuarios'.

Una vez configurado, para acceder al equipo remotamente debemos conocer su dirección IP privada o pública, según la conexión. Por tanto, desde otro equipo Microsoft Windows abrimos la herramienta 'Conexión a escritorio remoto' donde debemos desplegar 'Mostrar opciones' para introducir la dirección IP del equipo y el nombre del usuario con el que efectuar la conexión.



**Figura 6.25**  
Pestaña Acceso remoto  
de Propiedades del sistema.



**Figura 6.26**  
Conexión a escritorio remoto.

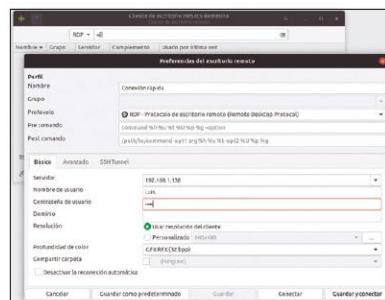


#### SABÍAS QUE...

Para conocer la IP pública basta con ingresar en una de las múltiples páginas web que nos devuelven este servicio, escribiendo en un buscador "cuál es mi IP" o "What is my IP".

Del mismo modo, Ubuntu facilita habilitar las conexiones remotas a través de la opción 'Compartir' en 'Configuración'. En ella, podemos establecer las opciones de acceso al equipo.

Para establecer conexiones desde equipos Ubuntu a otros equipos, se puede utilizar la aplicación cliente de escritorio remoto *Remmina*. Su configuración para acceder a otras máquinas es sencilla, basta con añadir un nuevo perfil de conexión (pulsando en el símbolo más en la esquina superior izquierda) con los datos necesarios: dirección IP del equipo servidor, nombre de usuario y contraseña.



**Figura 6.27**  
Configuración cliente  
de escritorio remoto Remmina.

#### TEN EN CUENTA

- ✓ En caso de un error en el acceso, hemos de asegurarnos que el equipo está encendido, se encuentra en red, habilitado el acceso remoto y habilitadas las reglas de entrada y salida del firewall relativas a la administración y asistencia remota de Windows.

## 6.7. Herramientas de seguridad

Los sistemas informáticos en red deben estar cubiertos ante la gran diversidad de amenazas de la forma más extensa posible. Para dar respuesta especializada al mayor número de amenazas posible, se han de implantar herramientas de seguridad complementarias en los sistemas informáticos. Por ejemplo, un antivirus hace una labor necesaria pero no es infalible ante amenazas sofisticadas de tráfico de red empleando protocolos que, sin embargo, un cortafuegos sí puede detectar. Por tanto, es necesario utilizar un conjunto de herramientas de seguridad adecuado a la estructura y tamaño de la organización. A las tratadas en temas anteriores, añadimos las siguientes.

### 6.7.1. Cifrado

Las herramientas de cifrado permiten garantizar la confidencialidad, autenticidad e integridad de los datos, mediante la conversión de los datos originales a un formato codificado. De esta manera, si otras herramientas de seguridad son superadas por amenazas, el cifrado es una opción muy recomendable.

Microsoft Windows ofrece cifrado EFS (Encrypting File System) sobre el sistema de archivos NTFS en versiones avanzadas. EFS no es compatible con la compresión, no obstante, garantiza que los archivos se cifren de manera individual, vinculándose al usuario que los cifró. Por tanto, los elementos cifrados están disponibles para aquellos usuarios que los cifraron y no para el resto.

Para cifrar archivos o carpetas en Microsoft Windows debemos acceder a las 'Propiedades' de un archivo y en 'Avanzados' de la pestaña 'General', seleccionamos 'Cifrar contenido para proteger datos'.

En caso de cifrar archivos y no su carpeta contendora, Microsoft Windows recomienda cifrar esta última mediante una advertencia. Y, una vez efectuado el cifrado, mediante el

botón ‘Detalles’ en la misma ventana anterior, permite añadir otros usuarios que pueden tener acceso.

Además, también avisará con la recomendación de crear una copia de seguridad de la clave de cifrado EFS.



#### TOMA NOTA

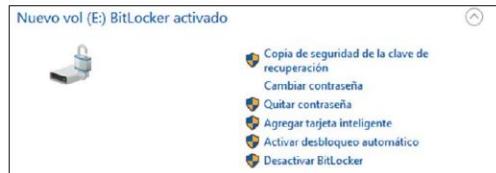
En sistemas de archivos NTFS, al mover archivos o carpetas a una carpeta cifrada, producirá que se cifre lo movido; sin embargo, esto no sucede al contrario.

Los archivos o carpetas cifrados se descifrarán si los movemos a un volumen que no sea NTFS.

Los archivos o carpetas cifradas son susceptibles de ser eliminados o listados por otros usuarios con permisos para ello.

Otra manera de cifrado y solucionar los inconvenientes del cifrado EFS en Microsoft Windows es a través de la herramienta *BitLocker*, que se encarga de asegurar la privacidad y la integridad en volúmenes enteros cifrando su contenido. Es más potente que EFS, ya que es independiente de los usuarios y puede emplear la tecnología de protección de datos TPM (Trusted Platform Module) para encriptar la información, así como una clave o tarjeta inteligente con PIN para su acceso. Se puede emplear solo en las versiones avanzadas de Windows.

Para cifrar volúmenes con BitLocker hemos de acceder a ‘Cifrado de unidad BitLocker’ del ‘Panel de control’ y, en ella, pulsar en ‘Activar BitLocker’ para un volumen. Microsoft Windows debe cumplir con varios requisitos, como que el hardware del sistema sea compatible con TPM (recomendado) o, en su lugar, usar una contraseña, y disponer de dos particiones como las que crea automáticamente Microsoft Windows cuando se instala este.



**Figura 6.28**  
Opciones de unidad con BitLocker activado.

Por otro lado, existen muchas herramientas de cifrado en sistemas operativos GNU/Linux. Una de ellas es *GNU Privacy Guard (GPG)*. GPG emplea encriptación simétrica y asimétrica. Ubuntu integra la aplicación *Seahorse*, que administra claves y contraseñas de cifrado, pudiendo gestionar claves GPG.

Para emplear esta utilidad cifrando carpetas o archivos, primero debemos crear las claves GPG desde ‘Archivo’, ‘Nuevo’ y ‘Clave PGP’.

Para utilizar las herramientas integradas con el explorador de archivos *Nautilus* lanzamos las siguientes instrucciones desde el terminal:

```
sudo apt update
sudo apt install seahorse-nautilus
```

De esta manera, ya podemos emplear la opción ‘Cifrar’ desde el menú contextual al pulsar el botón secundario del ratón sobre los archivos o carpetas. Al cifrarse, se genera un archivo con extensión .gpg, por lo que, por seguridad, se debe eliminar el anterior archivo sin cifrar.

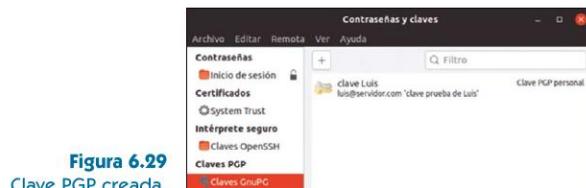


Figura 6.29  
Clave PGP creada.

### 6.7.2. Administración y análisis

El tratamiento, estudio y análisis del tráfico de la red puede resultar tan complejo como grande sea la infraestructura de red empleada, los servicios ofrecidos, el tráfico, el número de componentes y su función, etc.

Por tanto, se necesitan herramientas adecuadas que monitoricen la red de forma unificada, permitiendo:

- Estudiar la red en un momento concreto y a largo plazo.
- Valorar una optimización de la infraestructura de red.
- Detectar tráfico malintencionado.

Un ejemplo de herramienta de administración y análisis es *Pandora FMS* (<https://pandora-fms.com>). Este software está orientado a empresas y se encarga de monitorizar toda la red corporativa, trabajando sobre diferentes sistemas operativos, aplicaciones, protocolos, herramientas de virtualización, bases de datos, etc. Pandora FMS cuenta con una versión de pago (Enterprise) y una versión libre (Opensource).

### 6.7.3. Cortafuegos

Como ya sabemos, los cortafuegos tratan de controlar el tráfico entrante y saliente de una red. Las amenazas externas son filtradas por el cortafuegos (firewall), evitando así accesos no autorizados. Sus principales capacidades de filtrado son:

- ✓ De paquetes por direcciones IP o MAC.
- ✓ De aplicaciones, atendiendo al número de puerto.
- ✓ De direcciones URL.

Además, los firewalls suelen emplear el protocolo NAT para ocultar las direcciones privadas de los dispositivos protegidos. Esto permite que un atacante desde el exterior no conozca las direcciones privadas de una red.

Su implementación es llevada a cabo por hardware específico o software de seguridad, a saber:

- a) Firewall dedicados: son dispositivos hardware específicos cuya única función es analizar el tráfico a gran velocidad.
- b) Firewall integrados: implementados en dispositivos hardware que agrupan varias funciones, como los routers SoHo.
- c) Firewall por software: son aplicaciones software propias del sistema operativo o de terceros. Se distinguen:
  - Firewall de servidor: orientado a sistemas operativos en red con ámbito sobre los equipos clientes que gobierna.
  - Firewall personal: filtra el tráfico entre en equipo y el resto de la red.

Antes de configurar el firewall, se debe determinar la política de seguridad del mismo, distinguiéndose entre:

- Denegar por defecto: prohíbe todo lo que no está autorizado explícitamente.
- Aceptar por defecto: prohíbe cualquier comunicación definida explícitamente.

El firewall trabaja sobre unas reglas que deben predefinirse y que permiten, principalmente:

- ✓ Autorizar una conexión (*allow*).
- ✓ Bloquear una conexión (*deny*).

Los sistemas operativos permiten configurar los firewalls que incorporan mediante un terminal o a través de herramientas gráficas.

El sistema operativo Ubuntu incorpora el firewall *UFW* (firewall sin complicaciones). Funciona sobre *Iptables*, que es una herramienta muy extendida a nivel global e integrada en los Kernel GNU/Linux, encargada de filtrar los paquetes en función de unas reglas establecidas. Una interfaz gráfica para UFW es *Gufw*, cuya instalación debemos llevar a cabo.

Una vez instalada en Ubuntu, abrimos la aplicación desde ‘Sistema’, ‘Administración’ y ‘Configuración de firewall’.

En principio, debemos habilitar el firewall y configurarlo añadiendo reglas. Las reglas se pueden añadir de forma preconfigurada, simple o avanzada, según la pestaña por la que sean añadidas.



**Figura 6.30**  
Configuración de firewall en Ubuntu.

#### RECUERDA

- ✓ Es muy recomendable leer el manual de usuario de Gufw para establecer las reglas en el orden correcto y el tipo adecuado.

Por otro lado, y como ya sabemos, Microsoft Windows dispone del firewall de Windows Defender, que hemos utilizado con anterioridad para realizar conexiones FTP o hacer uso del escritorio remoto. Podemos acceder a través del ‘Panel de control’.

Dependiendo de la red en la que estemos conectados (dominio, pública y red doméstica o de trabajo), el firewall actuará con unas opciones predeterminadas. Para ajustar dichas opciones, activamos o desactivamos el ‘Firewall de Windows’.



**Figura 6.31**  
Firewall de Windows Defender.

También podemos permitir las conexiones de aplicaciones o características de Windows desde ‘Permitir una aplicación o una característica a través de Firewall de Windows’. Y desde ‘Configuración avanzada’ se pueden establecer reglas de entrada y de salida.



#### Actividad resuelta 6.4

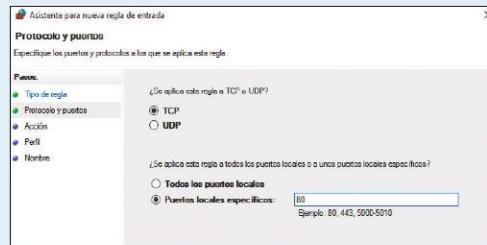
*Se desea establecer una regla para abrir un puerto que escucha solicitudes de entrada HTTP en el puerto 80.*

##### SOLUCIÓN

Desde ‘Configuración avanzada del Firewall de Windows’, pulsamos en ‘Reglas de entrada’ y ‘Nueva regla’ en la ventana de ‘Acciones’. A continuación, se abrirá un asistente que nos guiará en el proceso. Para nuestro caso, indicamos ‘Puerto’ y pulsamos en ‘Siguiente’.

En la siguiente ventana sobre ‘Protocolo y puertos’, seleccionamos ‘TCP’ y el puerto específico 80.

Al pulsar en ‘Siguiente’, seleccionamos los tipos de redes sobre los que deseemos su aplicación. A continuación, indicamos que deseamos permitir la conexión. Indicamos un nombre y, al finalizar, se creará la nueva regla, apareciendo en el conjunto de reglas de entrada.



**Figura 6.32**  
Asistente para nueva regla de entrada del firewall de Windows Defender.

#### 6.7.4. Sistemas de detección de intrusión

Los sistemas de detección de intrusión (IDS) monitorizan actividades o eventos en una red o en un host (de forma aislada), en busca de intentos de acceso sin permiso.

Los IDS se pueden catalogar:

- a) Según el sistema de detección o análisis empleado para detectar intrusiones:
  - Detección mediante firmas. Se coteja el tráfico de la red con patrones de reconocimiento predefinidos o firmas de ataques conocidos.
  - Detección mediante anomalías. En este caso se compara el comportamiento usual de la red en base a puertos, usuarios, conexiones de red, ancho de banda, protocolos, etc., con respecto a alteraciones en estos mismos objetos.
- b) Según el tipo de respuesta ante un ataque:
  - Pasivos: informan a administradores de sistemas o personal responsable del sistema o la red para que tomen las medidas oportunas.
  - Activos: actúan automáticamente, sin intervención humana, pudiendo informar al personal responsable del sistema o la red, recogiendo la máxima información posible sobre el ataque y tratando de parar el ataque o responder a este.
- c) Según la fuente de análisis:
  - Sistemas de detección de intrusiones en red (NIDS): analizan el tráfico de la red en segmentos estratégicos de la misma. Ejemplos característicos de NIDS son *Snort* y *Bro-IDS*.
  - Sistemas de detección de intrusiones de host (HIDS). Analizan la actividad interna de un host mediante los archivos de registro, conexiones de red o programas instalados. Ejemplos de HIDS muy utilizados son *OSSEC* y *Wazuh*.

*Security Onion* es una distribución basada en Ubuntu que aglutina un gran número de herramientas orientadas a la seguridad y análisis forense para la detección de intrusiones malintencionadas. Algunas de las herramientas de las que dispone son *Snort*, *Wireshark* o *Bro-IDS*. La instalación y configuración del sistema operativo y de las herramientas que contiene se realiza de manera muy sencilla, por lo que resulta ideal para probarlas.

#### 6.7.5. OpenSSH

Secure Shell (SSH) y su versión más avanzada y libre OpenSSH (emplea el protocolo OpenSSL) son herramientas que permiten acceder de forma remota y segura a otro equipo, por lo que al trabajar con servidores es muy recomendable su uso.

Para poder administrar un equipo Ubuntu por línea de comandos de forma segura, debemos instalar un servidor OpenSSH:

```
sudo apt install openssh-server
```

Automáticamente, comienza su ejecución, por lo que comprobamos su estado:

```
sudo systemctl status ssh
```

Permitimos que el firewall de Ubuntu (UFW) acepte conexiones a través del puerto SSH.

```
sudo ufw allow ssh
```

Solo bastaría modificar el archivo `/etc/ssh/sshd_config` para configurarlo según las políticas de seguridad implementadas por el administrador del sistema, como el puerto de conexión (deberemos modificarlo para mayor seguridad), usuarios permitidos (por defecto, no existe limitación), etc. Si lo modificamos, debemos reiniciar el servidor: `sudo service ssh restart`.

Ya en marcha, podemos establecer conexiones desde clientes SSH en otros equipos. Una vez dentro, deberemos autenticarnos con las credenciales de un usuario del sistema que esté habilitado para conexiones SSH.

En Microsoft Windows podemos descargar e instalar la aplicación *PuTTY* (<https://www.putty.org/>). A través de la cual introducimos la dirección IP y el puerto de conexión y, al pulsar en 'Open', establece la conexión. Mostrará un aviso preguntando si es seguro el sitio, muestra su clave pública y si se desea guardar para futuras conexiones.

En caso de emplear otro equipo Ubuntu como cliente, ya integran clientes SSH, por lo que simplemente debemos conectarnos por línea de comandos mediante:

```
ssh -p <numero Puerto> <usuario>@<IP_servidor>
```

Al igual que antes, nos pregunta si confiamos en el sitio en el que se va a conectar, ya que no conoce la clave pública. Por último, si deseamos terminar la conexión SSH, escribimos *exit*.

**Figura 6.34**  
Conexión SSH  
desde Ubuntu.

```
luis@luis-VirtualBox:~$ ssh -p 22 luis@192.168.1.140
The authenticity of host '192.168.1.140 (192.168.1.140)' can't be established.
ECDSA key fingerprint is SHA256:5i669mCuIk5cZM9ZWUx0KwkeuftSolUHp4ZYpxNUpw4.
Are you sure you want to continue connecting (yes/no)? yes
```

## Resumen

- Uno de los objetivos principales de los sistemas operativos es la explotación de los recursos, ya sean locales o en red. Cuando los recursos son compartidos en red, se debe tener un conocimiento más profundo de los permisos sobre estos y los derechos de usuario.

- En Microsoft Windows, para hacer uso de los recursos en red, deben compartirse adecuadamente, configurando permisos y derechos de usuario.
- Los permisos son una herramienta de protección de los recursos y definen el tipo de acceso concedido a un usuario o grupo de usuarios. Estos permisos pueden ser locales y en red, aplicándose los más restrictivos en caso de conflicto.
- La herencia facilita la gestión de los permisos en objetos contenidos y las listas de control de acceso o ACL flexibilizan la implementación de los permisos.
- Por otro lado, los derechos de usuario se asocian con usuarios y no con objetos, distinguiéndose entre derechos de inicio de sesión y privilegios específicos. Microsoft Windows los administra mediante la herramienta *Directiva de seguridad local*.
- Además, se deben establecer un conjunto de reglas de seguridad para administrar usuarios y equipos. Su implementación se lleva a cabo en Microsoft Windows mediante directivas de seguridad, que se implementan, a su vez, mediante objetos de directivas de grupo (GPO). De esta manera, se pueden centralizar políticas sobre usuarios y equipos, estableciendo permisos, bloqueos o controles. Ejemplos de ello son:
  - Configurar el entorno gráfico.
  - Automatizar tareas.
  - Evitar que un usuario pueda instalar y desinstalar programas.
  - Fortalecer las contraseñas de inicio de sesión.
  - Evitar el uso de memorias flash USB.
- Por otro lado, los sistemas que ofrecen servicios deben ser exprimidos, sacando el máximo partido a sus prestaciones y recursos ofrecidos. Por lo que el modelo cliente-servidor es el adecuado para que los servidores se enfoquen en administrar, gestionar y centralizar sus recursos ofrecidos. Además, las tareas de administración deben estar configuradas para que se realicen remotamente, gracias a herramientas de conexión remota que permiten realizar tareas de teletrabajo, soporte técnico, etc.
- Y todo esto sin olvidar la seguridad del sistema. Para lo que se han de emplear un conjunto de herramientas, complementarias entre ellas, que cubran la mayor diversidad de amenazas posibles.



## Ejercicios propuestos

Desde una máquina virtual con Microsoft Windows y con la configuración de la tarjeta de red más oportuna entre las máquinas virtuales que intervengan:

### 1. Permisos:

- a) Crea un usuario llamado *Jefe* y los siguientes grupos de usuarios con al menos dos usuarios en cada uno de ellos: *Comerciales*, *Gestores* y *Empleados*.
- b) En una unidad distinta a la del sistema operativo, crea una carpeta llamada *Usuarios*, que contenga otras dos: *Comerciales* y *Gestores*. A las carpetas *Comerciales* y *Gestores* tienen acceso todos los usuarios, pero solo los *Gestores*

podrán modificar, leer y ejecutar sobre la carpeta *Gestores*. De igual forma, los usuarios del grupo *Comerciales* podrán modificar, leer y ejecutar sobre la carpeta *Comerciales*. El usuario *Jefe* será el único que pueda modificar los permisos de todas las carpetas.

- c) Comprueba que todos los permisos se han establecido correctamente, realizando las pruebas que consideres oportunas.

**2. Permisos de red:**

- a) Comparte la carpeta *Usuarios* con el nombre *compUsuarios* para que solo los usuarios *Comerciales* y el usuario *Jefe* tengan acceso a ella desde otro equipo. Accede de dos formas distintas a dicha carpeta.
- b) Modifica este último recurso compartido como oculto.
- c) Accede desde otro equipo al recurso compartido oculto anterior con un usuario con permisos.
- d) Comprueba los recursos compartidos del equipo.

**3. Derechos de usuarios.** Realiza las acciones oportunas para que solo los *Administradores* del sistema y el usuario *Jefe* puedan apagar el equipo.

**4. Directivas de seguridad.** Crea dos GPO para prohibir el acceso a 'Configuración de PC' y 'Panel de control', así como cumplir los requisitos de complejidad de las contraseñas.

**5. Servidores:**

- a) Instala un servidor FTP con acceso a la carpeta *Gestores*, de modo que solo los usuarios del grupo *Gestores* puedan hacer uso del servicio.
- b) Instala el servidor de aplicaciones de Windows en el equipo de modo que aparezca nuestro nombre y apellidos cuando accedamos a él (*localhost*). Para lo que debemos crear una página web muy simple y sustituirla por la predeterminada. Abre el puerto 80 mediante el *Firewall de Windows* para poder escuchar solicitudes de entrada HTTP.

**6. Conexión remota.** Desde otro equipo con Microsoft Windows, accede al equipo mediante *Conexión de escritorio remoto*.

**7. Herramientas de seguridad:**

- a) Cifra los archivos (si no existen, los creas) de la carpeta *Comerciales* con EFS.
- b) Cifra una unidad entera con BitLocker.
- c) Accede a otro equipo Ubuntu mediante una conexión SSH con la aplicación PuTTY (se deberá configurar previamente un servidor SSH en él).

Desde una máquina virtual con Ubuntu y con la configuración de la tarjeta de red más oportuna entre las máquinas virtuales que intervengan:

**8. Conexión remota y herramientas de seguridad.**

- a) Accede al equipo Windows anterior mediante la aplicación Remmina.
- b) Accede a otro equipo Ubuntu mediante SSH por línea de comandos.

Crea máquinas virtuales configuradas en la red de clase (en modo Adaptador puente) y realiza los siguientes ejercicios:

9. Descarga la versión Opensource Appliance CD basado en CentOS de Pandora FMS desde <https://sourceforge.net/projects/pandora/> e instálala en una máquina virtual siguiendo los pasos de instalación. A continuación, debemos:
  - a) Detectar los dispositivos de la red.
  - b) Revisar los sistemas detectados.
  - c) Monitorizar el tráfico de red sobre una interfaz.
10. Descarga e instala el sistema operativo Security Onion (<https://securityonion.net>). Después, haz uso de cualquier herramienta IDS apoyándote en la documentación técnica oficial.

## ACTIVIDADES DE AUTOEVALUACIÓN

1. Los derechos de usuarios se clasifican en:
  - a) Derechos de inicio de sesión y directivas de seguridad.
  - b) Derechos de inicio de sesión y directivas específicas.
  - c) Derechos de inicio de sesión y privilegios específicos.
2. Las directivas de seguridad:
  - a) Se pueden catalogar en locales y especiales.
  - b) Permiten centralizar políticas sobre usuarios y equipos.
  - c) No permiten fortalecer las contraseñas de inicio de sesión de los usuarios.
3. Servidores de ficheros FTP:
  - a) El explorador de archivos de Microsoft Windows puede acceder a servidores FTP.
  - b) Microsoft Windows permite la instalación de un servidor de ficheros FTP solo con aplicaciones de terceros y no como una característica propia.
  - c) Filezilla Client es una aplicación cliente de impresión que puede ser utilizado como cliente FTP.
4. Las capas en las que se fundamentan los servidores de aplicación son:
  - a) Aplicación, datos y capa de datos.
  - b) Interfaz gráfica, presentación y datos.
  - c) Presentación, aplicación y datos.
5. ¿Qué protocolo se emplea para realizar conexiones remotas seguras?:
  - a) Remmina.
  - b) OpenSSL.
  - c) Conexión a escritorio remoto.

6. Los firewalls incorporados en Ubuntu y Microsoft Windows son respectivamente:

- a) Gufw y Firewall de Windows Defender.
- b) Firewall de Windows Defender y Gufw.
- c) UFW y Firewall de Windows Defender.

7. Security Onion:

- a) Es una distribución de Linux basada en Ubuntu, que aglutina diferentes IDS.
- b) Es un firewall.
- c) Es una herramienta de conexión remota segura.

8. En sistemas de archivos NTFS:

- a) Los archivos o carpetas cifrados se descifrarán si los movemos a un volumen que no sea NTFS.
- b) Al mover archivos de una carpeta cifrada a una carpeta no cifrada, provocará que se descifren los archivos movidos.
- c) Los archivos cifrados quedan ocultos al resto de usuarios no propietarios.

9. La herramienta GPG:

- a) Permite cifrar volúmenes enteros.
- b) Permite gestionar claves de cifrado.
- c) Usa tecnología TPM.

10. Pandora FMS:

- a) Es un IDS.
- b) Es un gestor de claves de cifrado.
- c) Es una herramienta de administración y análisis de la red.

#### SOLUCIONES:

1. **a** **b** **c**

2. **a** **b** **c**

3. **a** **b** **c**

4. **a** **b** **c**

5. **a** **b** **c**

6. **a** **b** **c**

7. **a** **b** **c**

8. **a** **b** **c**

9. **a** **b** **c**

10. **a** **b** **c**

# 7

## Aplicaciones informáticas

### Objetivos

- ✓ Saber clasificar el software en función de su licencia y propósito.
- ✓ Diferenciar entre requisitos de software mínimos y recomendados.
- ✓ Elaborar documentación haciendo uso y valorando diferentes aplicaciones ofimáticas de propósito general.
- ✓ Poder utilizar diferentes herramientas de Internet: correo, mensajería instantánea, transferencia de ficheros, búsqueda de documentación técnica.
- ✓ Hacer uso de diferentes utilidades de propósito general, como software antivirus, software de recuperación de datos o aplicaciones de mantenimiento del sistema.

### Mapa conceptual



### Glosario

**Botnet.** Conjunto de sistemas infectados por malware que actúan conjuntamente para realizar acciones malintencionadas como propagación de virus, generación de spam o ataques DDoS.

**Cloud computing.** Conjunto de servicios ofrecidos remotamente sobre una red, normalmente Internet.

**IMAP4.** Protocolo utilizado para acceder y descargar los correos desde los buzones de los servidores de correo, sin ser eliminados, a las aplicaciones de correo cliente.

**POP3.** Protocolo utilizado para acceder y descargar los correos desde los buzones de los servidores de correo a las aplicaciones de correo cliente.

**Redes P2P.** Redes que se basan en la compartición de datos entre equipos de igual a igual.

**Sistemas gestores de bases de datos.** Aplicaciones que tratan, administran y gestionan bases de datos.

**SMTP.** Protocolo empleado para enviar correos entre servidores de correos o desde un cliente de correo a un servidor de correo.

**Software antimalware.** Son aplicaciones que intentan evitar la acción de distintos tipos de amenazas: virus, adware, spyware, ransomware, gusanos, troyanos, etc.

**Software base o de sistema.** Software que actúa de intermediario entre el usuario y el hardware, que administra y gestiona los recursos hardware o software del sistema.

**Software de clonación.** Aplicaciones que permiten copiar los datos contenidos en particiones o discos y almacenarlos en otras particiones o medios, creando una réplica exacta.

**Software propietario.** Software que no cumple con cualquiera de las libertades del software libre, es decir, prohíbe o se debe autorizar su uso, distribución o modificación.

## 7.1. Introducción

Los sistemas informáticos sustentan las aplicaciones informáticas como herramientas para explotar diferentes tareas, como, por ejemplo, procesadores de texto, hojas de cálculo, antivirus o diferente software de mantenimiento.

Estas aplicaciones se pueden clasificar según su licencia o el propósito por el que son utilizadas. Además, para una óptima ejecución de las aplicaciones, estas especifican unos requisitos mínimos y recomendados en cuanto al hardware necesario y el sistema operativo sobre el que se instalarán.

Entre las aplicaciones más conocidas y utilizadas por la gran mayoría de usuarios, se encuentran las aplicaciones ofimáticas, que pretenden ayudar en tareas de oficina, como elaboración de textos, cálculos a partir de tablas de datos, creación de presentaciones o gestión de bases de datos.

Por otro lado, los sistemas informáticos en red emplean herramientas que hacen uso de servicios de Internet muy conocidos, como correo electrónico, mensajería instantánea, transferencia de ficheros y computación, y almacenamiento en la nube.

Además, se deben manejar un conjunto de aplicaciones relacionadas con el software base y orientadas a la optimización, administración o la seguridad del sistema: antimalware, clonación y copias de seguridad del sistema operativo, y la recuperación de datos.

Por último, nos detendremos en una tarea implícita en el mantenimiento de sistemas informáticos empresariales: la búsqueda y creación de documentación técnica de aplicaciones informáticas.

En este tema, desarrollaremos el conjunto de aplicaciones comentadas, intentando obtener una visión global de la explotación de las aplicaciones más prácticas y utilizadas en entornos domésticos y empresariales.

## 7.2. Tipos de software

Al software, como cualquier producto mercantil, se le asocia un contrato entre el vendedor y el comprador, que se establece cuando el segundo lo adquiere. A este contrato se le denomina

licencia, y en ella se establecen una serie de acuerdos para su explotación, instalación, distribución, publicación y estudio.

### 7.2.1. Clasificación por licencia

Los tipos de licencias más empleados son:

- a) Software libre o free software. El usuario dispone de cuatro libertades (definidas por La Fundación por el Software Libre), que son:
  - Libertad para ejecutar con cualquier propósito el programa (*libertad 0*).
  - Libertad para estudiarlo y adaptarlo a sus necesidades (*libertad 1*). Por lo que se debe tener acceso al código fuente.
  - Libertad para distribuir copias (*libertad 2*).
  - Libertad para modificarlo y mejorarlo (*libertad 3*). Así se pone a disposición de todo el público para su beneficio.
- Estas libertades no implican que no se pueda comercializar el mismo. Por ello, el software libre puede ser gratuito o no. Es común la confusión de software gratuito por la traducción del término inglés *free* como gratuito y no como libre, ya que ambas acepciones son traducibles al castellano.
- b) Software propietario o privativo. Cualquier software que no cumpla con cualquiera de las libertades del software libre, es decir, prohíbe o se debe autorizar su uso, distribución o modificación.
- c) Software con copyleft. Software que garantiza una distribución sin restricciones añadidas. Las licencias con copyleft presentan un conjunto de cláusulas para distribuir programas con copyleft. El ejemplo más característico es GNU GPL. Resulta una forma de asegurarse de que las modificaciones o distribuciones de dicho software sean también software libre, como el kernel de Linux.
- d) Software de dominio público. Es todo software que no tiene derechos de autor (sin copyright). Si hablamos de código fuente de dominio público, se puede decir que es software libre sin copyleft, por lo que podría degenerar en privativo.
- e) Software sin copyleft (también llamadas *con licencia permisiva*). El software distribuido o copiado no tiene por qué permanecer con el mismo tipo de licencia que su antecesor (como ocurre con copyleft). Por ello, estas licencias permisivas presentan flexibilidad en la distribución. Ejemplos de licencias de software libre permisivas, donde se puede cambiar la licencia son:
  - Licencia BSD. Existen varias versiones que se diferencian ligeramente. Son ejemplos de sistemas operativos que usan esta licencia FreeBSD y NetBSD.
  - Licencia MIT. Muy similar a las versiones de BSD. Bitcoin emplea esta licencia.
  - Licencia APACHE. El ejemplo más notorio que emplea esta licencia es el sistema operativo Android.
- f) Mozilla Public License (MPL). Se considera una licencia con copyleft débil, ya que está a medio camino entre las licencias GPL y BSD. Esto se debe a que se puede modificar la licencia, pero su código debe permanecer bajo licencia MPL en su formato original. De

esta manera, se intenta distribuir el código, pero sin perder el derecho de sus creaciones. Ejemplo de ello es el navegador Mozilla.

Además, existen diferencias entre las licencias de *software libre* y licencias de *código abierto* (*open source*). Ambas abogan por una mayor libertad para el software, sin embargo, el software libre está más orientado a la ética (habla de libertades) y el software de código abierto es más pragmático (debiéndose cumplir diez términos). No obstante, ambas son parecidas y contrarias al software privativo.



Los términos de distribución del software *open source* son:

1. Distribución libre.
2. Inclusión del código fuente.
3. Permitir modificaciones y trabajos derivados en las mismas condiciones que el software original.
4. Integridad del código fuente del autor, pudiendo requerir que los trabajos derivados tengan distinto nombre o versión.
5. No discriminación a personas o grupos.
6. Sin uso restringido a campo de actividad.
7. Los derechos otorgados a un programa serán válidos para todo el software redistribuido, sin imponer condiciones complementarias.
8. La licencia no debe ser específica para un producto determinado.
9. La licencia no debe poner restricciones a otro producto que se distribuya junto con el software licenciado.
10. La licencia debe ser tecnológicamente neutral.

Por otro lado, distinguimos los términos *shareware* y *freeware* como dos formas de distribución de software para dar a conocer el producto. En la mayoría de software shareware y freeware, el código fuente no está disponible, por lo que no es software libre. Shareware se emplea generalmente para aquel software que, o bien no tiene todas sus funcionalidades habilitadas, o se limita su uso temporalmente. Más tarde, se podrá aportar una cantidad económica para utilizarlo con total normalidad. Sin embargo, el software freeware es gratuito y sin limitaciones temporales.



### Actividades propuestas

- 7.1.** Busca información en Internet, indagando sobre la persona de Richard Stallman y su proyecto GNU.
- 7.2.** Busca en Internet dos ejemplos de software freeware y otros dos de software shareware.
- 7.3.** Localiza en Internet el contrato EULA de Windows 10. Lee detenidamente el apartado 2, "Instalación y derechos de uso" y 3, "Privacidad; consentimiento al uso de datos".

En el caso de los sistemas operativos propietarios (como es el caso de Microsoft Windows 10), se establecen contratos EULA (end-user license agreement) o contrato de licencia de usuario final (CLUF). En estos se establecen los términos por los que el usuario final se compromete, en general, a no copiar y distribuir el software, que el propietario puede recopilar información del sistema o que el usuario solo puede usar el producto sin ser dueño del mismo.

En cuanto a la distribución de sistemas operativos, debemos detenernos en las licencias de distribución que utiliza Microsoft:

- ✓ Licencia OEM: ligada a un equipo físico en concreto. Normalmente, está asociada al fabricante del equipo. Al adquirir un equipo nuevo, este incorpora el sistema operativo que se activa automáticamente al conectarnos a Internet.
- ✓ Licencia retail: son las vendidas a los usuarios, independientemente del hardware de destino. Su principal ventaja es que la licencia tiene validez, aunque cambiemos el hardware del equipo. Esta licencia está destinada a un único equipo.
- ✓ Licencias de volumen: destinadas a medianas o grandes corporaciones de cara a una activación de múltiples equipos con una única licencia, de manera más sencilla y con un beneficio económico considerable.

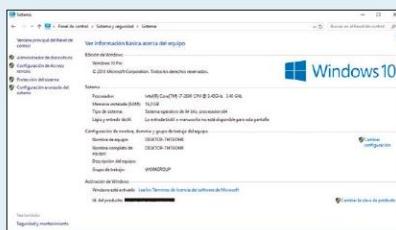
### Actividad resuelta 7.1



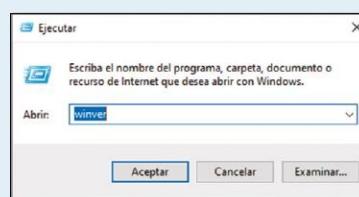
*Identificar el tipo de licencia (retail u OEM) de un equipo con sistema operativo Windows 10.*

#### SOLUCIÓN

Para ello, debemos localizar el identificador del producto, el cual es una agrupación alfanumérica que encontramos en la información del sistema dentro de 'Sistema' del 'Panel de control'. En el apartado 'Activación de Windows', encontramos el identificador que, dependiendo si acaba en OEM o no, es una licencia OEM o retail, respectivamente.



**Figura 7.1**  
Información del sistema en Microsoft Windows.



**Figura 7.2**  
Ejecución del comando `winver`.

También podemos identificar el tipo de licencia ejecutando el comando `winver`. Aparece la versión de Microsoft Windows y, en la parte inferior, muestra la concesión de la licencia a un usuario (retail) o a un fabricante (OEM).

**Recurso web**

Para saber más sobre este tema, puedes leer el artículo “¿Cuál es la diferencia entre software libre y open source?”, de Yúbal FM (Genbeta).

**www**



### 7.2.2. Clasificación por propósito

Atendiendo al uso del software, tradicionalmente este se clasifica en:

- a) Software base o de sistema: que incluye todos aquellos programas, aplicaciones o software (a diferente nivel con respecto a la cercanía con el hardware) que se encargan de actuar de intermediario, gestor o administrador entre el usuario y el hardware. Se incluye aquel software como BIOS, sistemas operativos, firmware de equipos, drivers de dispositivos, aplicaciones de diagnóstico y software mejora de equipos, etc.
- b) Software de desarrollo de aplicaciones: conjunto de software necesario para el diseño, desarrollo o implementación de software de sistema o de aplicación. En este se incluyen editores, compiladores, depuradores de código y entornos de desarrollo integrados (IDE).
- c) Software de aplicación: orientados a realizar tareas concretas, normalmente para un uso cotidiano por parte de un usuario final. Cualquier software que no se corresponda con las anteriores clasificaciones, formaría parte, como, por ejemplo:
  - Software ofimático.
  - Software empresarial.
  - Software de diseño.
  - Navegadores.
  - Software de seguridad.
  - Juegos.

### 7.3. Requisitos mínimos y recomendados

Las aplicaciones establecen unos *requisitos mínimos* para poder ejecutarse y usarse. Estos requisitos mínimos suelen estar alejados de las necesidades reales de los sistemas informáticos, cuando en estos se instalen diferentes aplicaciones, se ejecuten multitud de programas en paralelo, la conectividad sea constante, los accesos a los sistemas de almacenamiento se intensifiquen, etc. Por ello, junto con los requerimientos mínimos, los propietarios de aplicaciones o sistemas operativos suelen establecer unos *requisitos recomendados*, más en sintonía con la eficiencia y la ligereza del sistema en condiciones de cierto estrés.

A continuación, mostramos algunos ejemplos:

**CUADRO 7.1**  
Requisitos mínimos de Windows Server 2019

Procesador	RAM	Disco duro	Adaptador de red
64 bits a 1,4 GHz	512 MB y 2 GB con GUI y de tipo ECC	32 GB de espacio libre	PCI Express 1 Gigabit

**CUADRO 7.2**  
Requisitos mínimos de Firefox 70.0.1

Sistema operativo	RAM	Procesador	Disco duro
Windows 7 o superior	512 MB para sistemas de 32 bits o 2 GB para sistemas de 64 bits	Pentium 4 o nuevos	200 MB de espacio libre
macOS 10.9 o superior	512 MB	Macintosh con Intel x86	200 MB de espacio libre
GNU/Linux	Según distribución		

**CUADRO 7.3**  
Requisitos del videojuego Fortnite: Salvar el mundo

	Adaptador gráfico	Procesador	RAM	Sistema
Requisitos mínimos	Intel HD 4000 sobre PC o Intel Iris Pro 5200 en Mac	Core i3 2,4 GHz	4 GB de RAM	Windows 7, 8, o 10 de 64 bits o Mac OSX Sierra (10.12.6 o superior)
Requisitos recomendados	Nvidia GTX 660 o AMD Radeon HD 7870 con DX11 GPU equivalente	Core i5 2,8 GHz	2 GB de VRAM	Windows 7, 8, o 10 de 64 bits o Mac OSX Sierra (10.13.6 o superior)

#### 7.4. Herramientas ofimáticas

Las aplicaciones ofimáticas son imprescindibles como parte del software de aplicación de cualquier sistema informático. Normalmente, las aplicaciones ofimáticas se encuentran agrupadas en *suites* o *paquetes ofimáticos*, teniendo como objetivo ayudar en las tareas de oficina más habituales:

- Elaboración de textos.
- Cálculos a partir de tablas de datos.
- Creación de presentaciones.
- Gestión de bases de datos.
- Planificación de proyectos.
- Diseño gráfico.

Las suites ofimáticas más usadas son:

- a) *Microsoft Office*: privativo y disponible para Microsoft Windows y Mac OS. Constituida, entre otras, por las aplicaciones:
- Microsoft Word: procesador de textos.
  - Microsoft Excel: hoja de cálculo.
  - Microsoft Access: gestor de bases de datos.
  - Microsoft PowerPoint: editor de presentaciones.
  - Microsoft Outlook: gestor de correo electrónico y agenda.
  - Microsoft Publisher: diseño de publicaciones.
- b) *Apache OpenOffice* y *LibreOffice*. Ambas con licencias de software libre y de código abierto (GNU LGPL y MPL, respectivamente) para Microsoft Windows, Mac OS y GNU/Linux. Comparten la mayoría de características y funcionalidades. Constituidas por las aplicaciones:
- Writer: procesador de textos.
  - Calc: hoja de cálculo.
  - Base: gestor de bases de datos.
  - Impress: editor de presentaciones.
  - Draw: editor gráfico.
  - Math: editor de fórmulas matemáticas.

No obstante, cada vez más, se tiende a trabajar en la nube con aplicaciones ofimáticas como las ofrecidas por Google o suites como *G Suite* (Google) y *Office 365* (Microsoft). Las principales ventajas son:

1. Se pueden compartir documentos y trabajar de forma simultánea sobre estos.
2. Los documentos siempre se encuentran disponibles desde cualquier lugar.
3. Integran otros servicios en la nube, como correo electrónico, videoconferencias, almacenamiento de datos, desarrollo web, etc.
4. La escalabilidad es muy alta.

A continuación, vamos a detallar brevemente las herramientas ofimáticas más usadas, tomando como ejemplo la suite ofimática LibreOffice.



#### Actividad propuesta 7.4

Busca información en Internet acerca del origen de LibreOffice y su relación con Apache OpenOffice.

##### 7.4.1. Procesadores de texto

Son aplicaciones empleadas para trabajar con documentos de texto, a las que se puede dar formato y editar de forma rápida y cómoda. Los procesadores de texto permiten incorporar imágenes, gráficos, tablas, índices, tablas de contenido, macros y muchos otros elementos que lo enriquecen y facilitan su tratamiento.

Los formatos más usados son *.docx* (*Office Open XML*) y *.odt* (*OpenDocument Text*).

El manejo básico de los procesadores de texto pasa por conocer los siguientes aspectos:

- Abrir, cerrar, crear y guardar documentos de texto.
- Conocer la ventana del procesador de texto, junto con las diferentes áreas, barras y sus botones.
- Moverse por el documento de texto.
- Insertar, eliminar, cortar, copiar y pegar texto.
- Establecer formato de página y de fuente.
- Establecer formato de párrafo: numeración y viñetas, espaciados, alineaciones, sangrías y tabulaciones.
- Creación de tablas.
- Generación de tabla de contenidos o sumario.
- Insertar imágenes, formas y otros objetos.
- Revisión ortográfica y añadir comentarios.
- Imprimir documentos aplicando diferentes opciones.

#### 7.4.2. Hojas de cálculo

Las hojas de cálculo son muy utilizadas para la edición y manipulación de datos, empleando fórmulas o funciones para su tratamiento. Una hoja de cálculo está constituida por una tabla en la que las intersecciones entre filas y columnas constituyen las celdas. Estas pueden alojar datos (numéricos o alfanuméricos) y aplicarse cálculos.

El entorno es ideal para el tratamiento de datos, ya que se puede trabajar sobre rangos de celdas, aplicar diferentes formatos, imprimir de forma selectiva, generar gráficos y tablas, ordenar valores, automatizar acciones (macros), introducir formularios, etc.

Los formatos más usados son *.xlsx* (*Office Open XML*) y *.ods* (*OpenDocument Spreadsheet*).

El manejo básico de las hojas de cálculo pasa por conocer los siguientes aspectos:

- ✓ Abrir, cerrar, crear y guardar libros.
- ✓ Conocer la ventana de la aplicación, junto con las diferentes áreas, barras y sus botones.
- ✓ Moverse por un libro.
- ✓ Referenciar una celda y un rango de celdas dentro de un libro.
- ✓ Conocer los tipos de valores.
- ✓ Aplicar formato de celdas y formato de datos.
- ✓ Usar fórmulas y funciones.
- ✓ Generar gráficos.
- ✓ Imprimir hojas de cálculo.

#### 7.4.3. Software de presentación

En charlas, debates, presentaciones, conferencias y un sinnúmero de actuaciones de cara al público es habitual apoyarse en presentaciones para guiar la charla y captar la atención de los presentes.

El software de presentación facilita la creación de documentos mediante diapositivas. En estas se pueden incorporar textos, imágenes, gráficos, enlaces o vídeos. El documento de presentación se potencia al dotar de animación a las diapositivas, añadir temas comunes, incorporar música o sonidos, generar transiciones entre diapositivas, etc.

Los formatos más usados son *.pptx* (*Office Open XML*) y *.odp* (*OpenDocument Presentation*).

El manejo básico del software de presentación pasa por conocer los siguientes aspectos:

- Abrir, cerrar, crear y guardar presentaciones.
- Conocer la ventana de la aplicación, junto con las diferentes áreas, sus barras y sus botones.
- Insertar, eliminar y mover diapositivas.
- Usar plantillas.
- Organizar la disposición de los elementos de una diapositiva, incorporando texto, imágenes, enlaces, tablas y objetos multimedia.
- Configurar la presentación de diapositivas: transiciones y música.
- Opciones de impresión.

#### 7.4.4. Sistemas gestores de bases de datos

Cualquier compañía, empresa u organización trata con una gran cantidad de información, que es necesario almacenar de forma organizada y estructurada para su explotación de forma fácil y eficiente. Toda esta información relacionada entre sí se denomina *base de datos*, y está formada por uno o varios archivos. Las bases de datos se estructuran mediante tablas relacionadas, que son fruto de un esquema o diseño previo, conocido como *modelo entidad-relación*.

La explotación de la base de datos, es decir, su tratamiento, administración y gestión se realiza mediante una aplicación llamada *sistema gestor de bases de datos*. Este sistema permite definir, manipular o consultar datos.

Los formatos más usados son *.accdb* (*Office Open XML*) y *.odb* (*OpenDocument Database*).



SABÍAS QUE...

Los sistemas gestores bases de datos presentan multitud de ventajas frente al almacenamiento estándar de información en el sistema de archivos como son: control de redundancia de datos, integración de los datos, consistencia, independencia entre datos y tratamiento de los mismos, mayor eficiencia, reducción en el almacenamiento, etc.

El manejo básico de los sistemas gestores de bases de datos pasa por conocer los siguientes aspectos:

- ✓ Abrir, cerrar, crear y guardar bases de datos.
- ✓ Conocer la ventana de la aplicación, junto con las diferentes áreas, barras y sus botones.
- ✓ Crear, eliminar y editar tablas.
- ✓ Insertar y editar datos en tablas.
- ✓ Relacionar tablas entre sí.
- ✓ Realizar consultas simples de selección.
- ✓ Crear formularios.
- ✓ Generar informes.
- ✓ Imprimir formularios e informes.



## Recurso web

LibreOffice proporciona una excelente colección de guías en las que se detalla el manejo de la suite. Es recomendable la lectura de los capítulos con los primeros pasos con Writer, Calc, Impress y Base:



## 7.5. Herramientas de Internet

Hoy en día no se entiende Internet sin servicios, como correo electrónico, mensajería instantánea, la nube o aplicaciones de transferencia de ficheros, entre otros. A continuación, analizamos estos.

### 7.5.1. Correo electrónico

El correo electrónico ha desplazado en muchos aspectos al correo postal, aportando eficiencia y rapidez en las comunicaciones a nivel global. No obstante, ambos son necesarios y cada uno ocupa un espacio diferente. El correo electrónico (e-mail) es un servicio de intercambio de mensajes que hace uso de servidores de correo electrónico para su gestión. Este servicio es el más empleado en Internet. Los servidores de correo utilizan diferentes protocolos para enviar o recibir correos:

- a) Protocolo SMTP (Simple Mail Transfer Protocol): empleado para enviar correos entre servidores de correos o desde un cliente de correo a un servidor de correo. Es decir, envía y recibe correos entre buzones.
- b) Protocolo POP3: se utiliza para acceder y descargar los correos desde los buzones de los servidores de correo a las aplicaciones de correo cliente. Por defecto, estos correos se descargan localmente y se eliminan del servidor de correo.
- c) Protocolo IMAP4: también se emplea para acceder a los buzones de correo, pero, a diferencia de POP3, los correos no se eliminan, por lo que pueden ser utilizados desde diferentes máquinas.

Los tipos de gestores de correo, según su acceso, son:

- Basados en aplicación. Son aplicaciones instaladas en un equipo cliente, que se conecta al servidor de correo, pudiendo gestionarlo. Ejemplo de ello son Microsoft Outlook y Mozilla Thunderbird.
- Basados en web (webmail). Son páginas webs a través de las cuales se puede gestionar el correo, como Gmail, Hotmail y Yahoo.

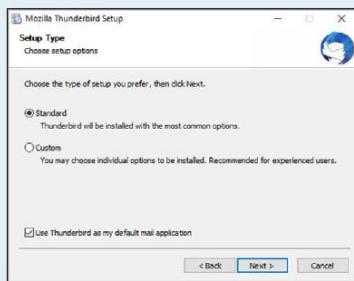


### Actividad resuelta 7.2

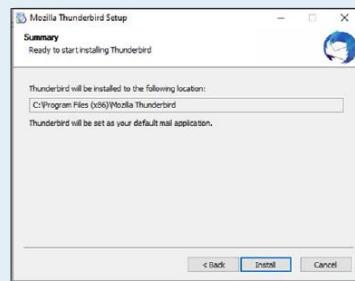
*Configurar la aplicación de correo gratuita Mozilla Thunderbird con una cuenta de Gmail.*

#### SOLUCIÓN

Descargamos Mozilla Thunderbird desde <https://www.thunderbird.net/en-US/download/>. Instalamos la aplicación, indicando el tipo de instalación standard y la ruta por defecto de instalación.



**Figura 7.3**  
Tipo de configuración.



**Figura 7.4**  
Ruta de instalación.

Una vez instalado, el programa es lanzado, solicitando los datos de configuración de una cuenta de correo.



**Figura 7.5**  
Credenciales para la gestión de correos.



**Figura 7.6**  
Acceso a la cuenta de correo.

Tras validar la contraseña, indicamos la configuración IMAP. A continuación, solicita las credenciales de la cuenta de correo en una nueva ventana para permitir a Mozilla Thunderbird gestionar los correos de Gmail.

Al finalizar el proceso, ya podemos gestionar la cuenta de correo con el cliente de correo Mozilla Thunderbird.



**Figura 7.7**  
Configuración de Mozilla Thunderbird.

### 7.5.2. Mensajería instantánea

Las herramientas de mensajería instantánea facilitan la comunicación a través de texto, imágenes, vídeos o sonido en tiempo real, y entre diferentes usuarios a través de una red de comunicación.

La mensajería instantánea, que aún multitud de tecnologías y protocolos, es un concepto amplio, que crece día a día, tanto a nivel empresarial como personal. Suelen ser servicios donde una aplicación cliente instalada en un dispositivo solicita autenticación por parte de un usuario. Una vez conectado el usuario, el sistema de mensajería habilita su lista de contactos, indicando quién se encuentra en línea para iniciar una comunicación.

Los sistemas de mensajería instantánea pueden clasificarse:

- Según el tipo de comunicación:
  - Síncronos: los usuarios han de estar conectados durante la comunicación.
  - Asíncronos: los usuarios pueden acceder al sistema de comunicación, recibiendo los mensajes almacenados durante su tiempo de desconexión.

#### TEN EN CUENTA

- ✓ A nivel empresarial, existen aplicaciones que difieren de las tradicionales por su orientación claramente corporativa, en las que un mayor volumen de comunicación es necesario, y que se permiten centralizar la gestión. Algunas de estas herramientas son Skype Empresarial o WhatsApp Business.

- Según el tipo de información transmitida:

- Por texto.
- Por voz.
- Por vídeo.
- Mixtos.

Algunos ejemplos son: WeChat, Viber, Telegram, Signal, Skype, Facebook Messenger, WhatsApp, Google Mensajes, Google Duo, iMessage, Snapchat o Yahoo! Messenger.

### 7.5.3. Transferencia de ficheros

Además del protocolo FTP, uno de los más extendidos e importantes, existen otros muchos protocolos, servicios y herramientas asociados a la transferencia de ficheros. Como, por ejemplo, a través de aplicaciones específicas, como *WeTransfer*, el almacenamiento en la nube y su compartición, etc.

Sin embargo, son muy empleadas las redes P2P (Peer to Peer). Estas se fundamentan en la compartición de datos entre equipos de igual a igual, sin servidores intermedios. Por ello, también se conocen como *redes entre iguales*, donde cada equipo o nodo de la red puede actuar de cliente y servidor al mismo tiempo.

Sus principales características son:

1. Escalabilidad: pueden incorporarse fácilmente más nodos a la red, aumentando los recursos de esta, su robustez y, por tanto, mejorando su funcionamiento.
2. Anonimato: se garantiza el anonimato a los usuarios y nodos de la red, restringiendo la identificación a la necesaria para gestionar la comunicación.
3. Descentralización: la mayoría de las redes P2P son descentralizadas, aunque existen diferentes grados de descentralización de los recursos:
  - a) Descentralizada: distribuyen los recursos entre todos los nodos. Ejemplos: las empleadas por el protocolo *Gnutella2*, así como *Ares*.
  - b) Centralizada: por un servidor central circula todo el tráfico y administra el funcionamiento de la red, almacenando los contenidos.
  - c) Híbrido: situación intermedia entre las anteriores, en la que existen nodos que actúan de servidores que controlan la estructura de la red, sin almacenar ni distribuir los contenidos. Ejemplos: las empleadas por el protocolo *BitTorrent* y *eDonkey*, entre muchas otras.

Dadas las características de las redes P2P, no solo se emplean para compartir archivos, sino también para realizar cálculos en investigaciones científicas (mediante aplicaciones distribuidas), implementar sistemas de ficheros distribuidos (CFS), telefonía VoIP (*Skype*), transacciones de monedas virtuales (*Bitcoin*), compartir ancho de banda y acceso a Internet (*Open Garden*) o enrutamientos anónimos para redes seguras (*Tor*).



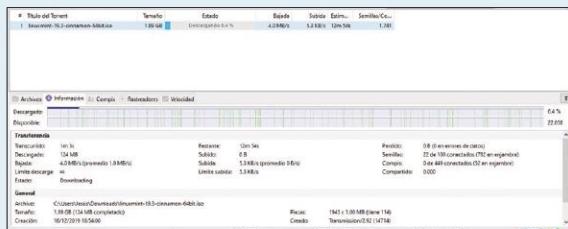
#### Actividad resuelta 7.3

*Descargar Linux Mint por BitTorrent.*

##### SOLUCIÓN

Si no disponemos de ningún cliente BitTorrent, podemos descargar alguno, como, por ejemplo, BitTorrent (aplicación que emplea el mismo nombre que el protocolo) desde <https://www.bittorrent.com/lang/es/>. Posteriormente, lo instalamos a través de su asistente de instalación.

Una vez instalado, accedemos a la página oficial de Linux Mint y seleccionamos el enlace de descarga según el tipo de arquitectura y escritorio (por ejemplo, *Cinnamon 64-bit*). En la siguiente página, seleccionamos el enlace Torrent, que iniciará la descarga del archivo *.torrent*. Al ejecutar el archivo *.torrent* descargado, comenzará el proceso de descarga del sistema operativo Linux Mint, abriendo la aplicación BitTorrent.



**Figura 7.8**  
Descarga de Linux Mint por BitTorrent.

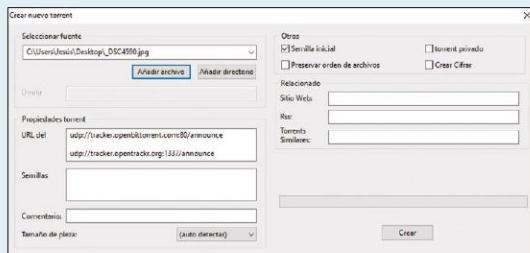
#### Actividad resuelta 7.4



Compartir archivos por BitTorrent generando un *.torrent*.

##### SOLUCIÓN

Si deseamos generar un *.torrent* a partir de archivos propios, también podemos hacerlo mediante la aplicación cliente BitTorrent. Para ello, debemos acceder a 'Crear nuevo Torrent' desde el menú 'Archivo'.



**Figura 7.9**  
Creación de un archivo *.torrent* por BitTorrent.

Hemos de seleccionar el archivo o carpeta para compartir y del cual se generará el *.torrent*. Al seleccionar 'Crear', solicitará el nombre y el lugar donde se almacenará el *.torrent*. Este archivo deberemos compartirlo con aquellas personas que deseen descargar el archivo original, sin cerrar la aplicación. A través de él y desde otro cliente BitTorrent, se podrá descargar el archivo (como hemos hecho en el ejemplo anterior).



### Actividad propuesta 7.5

Emplea servicios en la nube, como OneDrive, Google Drive, Dropbox y WeTransfer para compartir o enviar carpetas o archivos. Indica las principales diferencias entre ellos.

#### 7.5.4. Computación y almacenamiento en la nube

La computación en la nube, también conocida como *cloud computing*, se entiende como un modelo formado por un conjunto de servicios ofrecidos sobre una red (normalmente Internet) que atienden a clientes remotamente.

Se fundamenta en tecnologías que permiten a los clientes conseguir los servicios y la efectividad necesaria en cada uno de ellos bajo demanda. De esta manera, el modelo de computación en la nube presenta multitud de ventajas:

- ✓ Los clientes no tienen la necesidad de instalar aplicaciones para acceder a los servicios.
- ✓ Gran flexibilidad y eficiencia de los recursos al ajustar los servicios a la demanda.
- ✓ Alta disponibilidad desde cualquier localización y dispositivo.
- ✓ Seguridad y protección de datos al estar desvinculados estos de los dispositivos clientes.
- ✓ Reducción del coste de computación por parte de los clientes.

Existen tres modelos de servicios de computación en la nube, según los servicios gestionados por los usuarios a nivel de arquitectura de computación:

- a) Software as a Service (SaaS): utilizado como un modelo en el que una aplicación es alojada como un servicio para usuarios. El usuario solo puede hacer uso de las aplicaciones del proveedor. En cambio, el proveedor de los servicios se encarga de todo el soporte y el mantenimiento de la red, servidores, sistemas operativos, almacenamiento o aspectos de configuración de la aplicación.
- b) Platform as a Service (PaaS): modelo orientado a desarrolladores de aplicaciones que alojan sus programas sobre la plataforma de un proveedor. De esta manera, se proporciona un entorno para desarrollar y desplegar aplicaciones.
- c) Infrastructure as a Service (IaaS): modelo en el que los proveedores ofrecen la infraestructura (red, almacenamiento y servidores) a clientes (normalmente empresas) para que exploren sus necesidades. Sus usuarios suelen ser ingenieros de sistemas.

Por tanto, cada modelo dependerá de quién gestione las capas de la arquitectura de computación (figura 7.10).

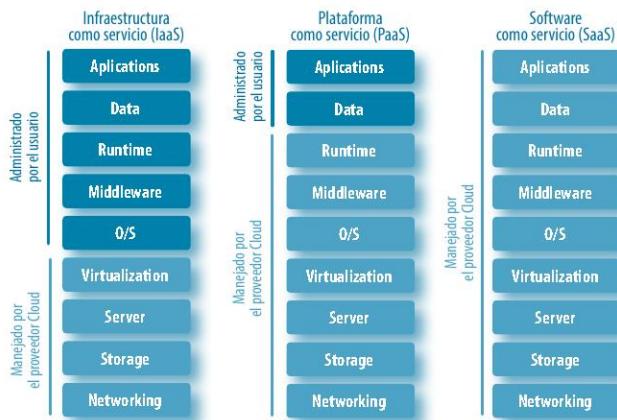
Una tecnología clave para el desarrollo de la computación en la nube es la virtualización, ya que permite escalar fácilmente los recursos físicos informáticos. Es decir, a partir de unos recursos de infraestructura comunes se ofrecen soluciones adaptadas y flexibles a diferentes clientes, servicios y necesidades.

Ejemplos de productos de computación en la nube por modelos de servicio son:

- SaaS: Hotmail, Office 365, OneDrive, Gmail, Google Docs, Dropbox, Drive, iCloud, Facebook, Twitter, etc.
- PaaS: Microsoft SharePoint, Heroku, Google App Engine o Cloud Foundry.

- IaaS: Microsoft Azure, Google Compute Engine (bases de la arquitectura de computación en la nube para Microsoft y Google, respectivamente) y Amazon Web Services.

Sin embargo, no todo son ventajas. La computación en la nube implica la pérdida de control completo sobre los servicios y los datos, ya que dependen de las plataformas e infraestructuras de los proveedores. Es decir, los clientes no pueden usar los servicios sin acceso a la red (Internet) y encontrándose bajo las políticas de privacidad, seguridad, actualización, escalabilidad y flexibilidad de los proveedores.



**Figura 7.10**  
Modelos de servicios de computación en la nube.

### Actividades propuestas



- 7.6.** Busca en Internet un producto SaaS diferente a los ejemplos mencionados en el capítulo.
- 7.7.** Busca en Internet y describe brevemente los servicios o aplicaciones que ofrece uno de los productos PaaS mencionados en el capítulo.
- 7.8.** Busca en Internet productos SaaS o PaaS no privativos, indicando su propósito y el modelo de servicio al que pertenece.

### 7.6. Software antimalware

Por software antimalware se entienden aquellas aplicaciones que intentan evitar la acción de distintos tipos de amenazas: virus, adware, spyware, ransomware, gusanos, troyanos, etc. Aunque se diferencian los antivirus de los antimalware, ambos pretenden detectar y erradicar el software malicioso. Los antivirus suelen integrar multitud de herramientas, como control parental, firewall, gestores de contraseñas, destructores de datos, privacidad en la navegación web, VPN, mejora de rendimiento, backups, etc. Algunos ejemplos de compañías con productos antivirus son: Avira, BitDefender, Avast, McAfee, Bullguard, Norton, Kaspersky, Vipre, Sophos, etc.

Los antivirus son capaces, cada vez más, de detectar cualquier tipo de malware, sin embargo, los antimalware, propiamente dichos, suelen trabajar de manera independiente a los antivirus, actuando de forma no residente. Son capaces de detectar amenazas no conscientes por parte de los antivirus, por lo que se hacen necesarios. Algunos ejemplos de antimalware son: Malwarebytes Anti-Malware, Malicious Software Removal Tool, Secure Hunter, Malware Fox, ComboFix, etc.

Dentro del vocabulario relacionado con ataques informáticos, podemos encontrar:

- ✓ Phishing: suplantación de identidad con objeto de que un usuario aporte datos valiosos al atacante para actuar en nombre del primero. Resulta común el phishing a través de páginas webs y e-mails donde se intenta captar la atención de los usuarios para que aporten datos bancarios, contraseñas u otra información.
- ✓ Sniffing: técnica que trata de capturar el tráfico de la red para, una vez interpretado, obtener información valiosa, como credenciales, datos bancarios, lectura de emails, etc.
- ✓ DoS o denegación de servicio (Denial of Service): tiene lugar cuando se colapsa un sistema, interrumpiendo los servicios ofrecidos, debido la gran cantidad de tráfico generado sobre este. Si la denegación de servicio proviene de varios puntos de conexión simultáneamente, se denomina denegación de servicio distribuido (DDoS).
- ✓ Hijacking: consiste en secuestrar, es decir, sustituir por parte de los atacantes un navegador web, dominios, contenidos web, DNS, URL, etc., de manera que el atacante redirige a su criterio la navegación para obtener datos de los usuarios u obtener algún beneficio.

Vocabulario relacionado con tipos de malware es el siguiente:

- Virus: término genérico que hace referencia al software que trata de alterar el funcionamiento de un sistema informático o la red. Según el método de propagación, se distinguen:
  - Gusano: no requieren la intervención humana para replicarse, empleando la memoria de los equipos para propagarse autónomamente a través de la red a una gran velocidad.
  - Virus (propriamente dicho): se presentan como ficheros ejecutables o adjuntos a estos, propagándose al ser copiados o ejecutados.
- Troyanos: software con apariencia inofensiva o que se hace pasar por una aplicación auténtica, que puede actuar con diferentes propósitos, como el acceso de terceros a través de puertas traseras al equipo para obtener información confidencial.
- Spyware: software encargado de realizar tareas de espionaje sobre el equipo con el fin de recabar información sobre ciertas actuaciones del usuario y enviarlas a terceros para su posterior venta (normalmente para usos publicitarios).
- Adware: programa que muestra publicidad mediante ventanas emergentes, navegadores, barras de herramientas, etc. En algunos casos, resulta ser también spyware.
- Ransomware: programa que bloquea o restringe el acceso a sistemas, datos o información exigiendo un rescate.
- Keylogger: software que registra las pulsaciones del teclado de un equipo y las envía a terceros. Suelen estar vinculados con la captura de información bancaria.
- Botnet: conjunto de sistemas infectados por malware que actúan de manera conjunta para realizar alguna acción malintencionada, como la propagación de virus, generación de spam o ataques DDoS.



### SABÍAS QUE...

El Instituto Nacional de Ciberseguridad de España (INCIBE) es la entidad de referencia para el desarrollo de la ciberseguridad y de la confianza digital de ciudadanos, profesionales, empresas y especialmente para sectores estratégicos (hospitales, centros de almacenamiento, centrales generadoras de energía, centros financieros, centros de transporte, aeropuertos, etc.).

INCIBE aúna diferentes recursos, herramientas y enlaces para reforzar la ciberseguridad, la confianza y la protección de la información y privacidad en los servicios de la sociedad de la información. Su página web es <https://www.incibe.es/>

## 7.7. Clonación y copias de seguridad

Los administradores de sistemas necesitan herramientas específicas que faciliten las siguientes tareas sobre el sistema y los datos:

- ✓ Copia de seguridad de todo el sistema.
- ✓ Aseguramiento de los datos.
- ✓ Recuperación, reinicio y restauración.

Las causas más comunes por las que se necesita la restauración son:

- ✓ Errores irrecuperables de los sistemas operativos.
- ✓ Eliminaciones o accesos malintencionados.
- ✓ Mal funcionamiento de una aplicación.
- ✓ Rotura de un medio de almacenamiento o inconsistencia de los datos almacenados en él.
- ✓ Descuidos personales.
- ✓ Malware.

### 7.7.1. Clonaciones

Las clonaciones permiten copiar los datos contenidos en particiones o discos y almacenarlos en otras particiones o medios, creando una réplica exacta. De esta manera, quedan a salvo todos los datos contenidos en origen, por lo que, si dispone de sistema operativo, aplicaciones instaladas y datos en el origen, todos ellos quedarán duplicados exactamente.

Las herramientas de clonación facilitan la creación de *imágenes*, es decir, permiten crear un archivo o conjunto de archivos que contienen la estructura y el contenido del origen, facilitando así su manejo para la posterior clonación.

Al realizar una clonación, en general, los discos implicados no pueden estar en funcionamiento, por lo que las políticas de copias de seguridad y de *backup* deben recoger estas acciones de manera planificada. Además, al finalizar la clonación de un disco con sistema operativo, se ha de configurar el “nuevo” sistema, adecuándolo al sistema de arranque, la red, los usuarios, licencias, etc., evitando duplicidades que generen errores.

Entre el software libre de clonación más empleado, destaca *GParted* (<https://gparted.org/>), por su versatilidad en la mayoría de las plataformas, y *Clonezilla*, ya que es extremadamente fiable y flexible, aunque con una interfaz más profesional.

## TOMA NOTA



Clonezilla facilita la clonación de particiones o discos. Su página oficial <https://clonezilla.org/>, a través de tutoriales, explica los pasos para la clonación de sus versiones: *Clonezilla live* (una sola máquina), *Clonezilla SE* y *Clonezilla lite server* (versiones para clonar más de cuarenta equipos simultáneamente en red). Las diferentes versiones se pueden descargar desde la propia página. Además de la lectura de estos tutoriales, es recomendable ver vídeos explicativos para un mejor entendimiento del procedimiento de clonación.

### 7.7.2. Copias de seguridad

Las copias de seguridad son acciones habituales que se llevan a cabo por parte de los administradores en la mayoría de los sistemas informáticos, previo diseño de unas políticas de *backup*. No obstante, esta tarea ha de hacerse extensiva a todos los usuarios con sus propios datos.

A diferencia de las clonaciones, las herramientas de copias de seguridad manejan archivos concretos, facilitando su restauración individual. Además, podemos tratar con estos sin necesidad de dejar de usarlos.

Las políticas de backup determinan la frecuencia y el conjunto de archivos sobre los que se efectúan las copias de seguridad. También establecen el soporte de backup y el tipo de copia de seguridad. Los tipos de copias más comunes son:

- a) Copia total: se efectúa una copia de seguridad de todos los archivos seleccionados.
- b) Copia incremental: solo se realiza la copia de seguridad de aquellos archivos que han cambiado desde la última copia de seguridad total o incremental.
- c) Copia diferencial: se hace una copia de seguridad de aquellos archivos que hayan cambiado desde la última copia de seguridad total.



#### Actividad propuesta 7.9

Investiga las ventajas y desventajas de cada tipo de copia de seguridad: total, incremental y diferencial.

##### A) Por línea de comandos en Linux

Para realizar las copias de seguridad en Linux, se emplea el comando *tar*. Este comando permite empaquetar archivos en un contenedor (archivo que contiene otros archivos o carpetas). Las opciones más empleadas son:

- c: crea un contenedor.
- x: extrae o restaura archivos desde el contenedor.
- f <contenedor>: crea o lee desde el contenedor.
- z: comprime o descomprime con *gzip*.
- j: comprime o descomprime con *bzip2*.

- t: lista el contenido del contenedor.
- v: detalla las acciones realizadas.
- C <directorio>: cambia al directorio *directorio* antes de realizar otra acción.

Es recomendable que al comprimir con *gzip* y *bzip2* se añada al nombre del contenedor la cadena *.tgz* y *.tbz*, respectivamente. A continuación, se muestran algunos ejemplos.

Creamos dos archivos y los empaquetamos en un contenedor de nombre *almacen*. Los eliminamos del directorio actual. Vemos el contenido del contenedor. Los extraemos del contenedor al directorio actual.

```
luis@luis-VirtualBox:~$ touch archivo1 archivo2
luis@luis-VirtualBox:~$ tar -cvf almacen archivo1 archivo2
archivo1
archivo2
luis@luis-VirtualBox:~$ rm archivo1 archivo2
luis@luis-VirtualBox:~$ ls -l archivo*
ls: no se puede acceder a 'archivo': No existe el archivo o el directorio
luis@luis-VirtualBox:~$ tar -tf almacen
archivo1
archivo2
luis@luis-VirtualBox:~$ tar -xvf almacen
archivo1
archivo2
luis@luis-VirtualBox:~$ ls -l archivo*
-rw-r--r-- 1 luis luis 0 may 14 21:10 archivo1
-rw-r--r-- 1 luis luis 0 may 14 21:10 archivo2
```

**Figura 7.11**  
Ejemplo de uso  
del comando *tar*.

Comprimimos con *gzip*, en un contenedor llamado *backup.bz*, el directorio */home/luis*. Creamos un directorio *dir\_respaldo* y descomprimimos (con *gzip*) en este el contenido del contenedor. Por último, comprobamos el contenido del directorio *dir\_respaldo*.

```
luis@luis-VirtualBox:~$ tar -czf backup.bz /home/luis/*
tar: Eliminando la '/' inicial de los nombres
tar: Eliminando la '/' inicial de los objetivos de los enlaces
luis@luis-VirtualBox:~$ mkdir dir_respaldo
luis@luis-VirtualBox:~$ tar -xzf backup.gz -C ./dir_respaldo
luis@luis-VirtualBox:~/dir_respaldo$ ls
luis@luis-VirtualBox:~/dir_respaldo$ cd home/
luis@luis-VirtualBox:~/dir_respaldo$ ls
luis
luis@luis-VirtualBox:~/dir_respaldo$ cd luis/
luis@luis-VirtualBox:~/dir_respaldo$ ls
ahora    backup.bz  Escritorio      Música      prueba_umask2    si
almacen  backup.gz  examples.desktop Plantillas  prueba_umask2.txt  tareas
almacen_luis Descargas   hoja        proprietario  prueba_umask.txt  Videos
archivo1  dir_respaldo Imágenes    propietario.txt  público
archivo2  Documentos   luis.cron   prueba_umask  reunión.txt
```

**Figura 7.12**  
Ejemplo de uso  
del comando *tar*  
comprimiendo  
con *gzip*.

En las copias de seguridad es recomendable indicar, como parte del nombre de la copia de seguridad, la fecha y el tipo de copia efectuada. Para indicar la fecha, se emplea el comando *date*. Este puede modificar su salida por pantalla, según el formato indicado, por lo que es recomendable estudiar su ayuda (*man date*) (figura 7.13).

Podemos indicar una fecha dentro de una cadena, si ejecutamos la orden *date* previamente: enmarcándola entre el carácter “`” (ácento grave) o dentro de los paréntesis de *\$()* (figura 7.14).

```
luis@luis-VirtualBox:~$ date
mié may 15 00:19:43 CEST 2019
luis@luis-VirtualBox:~$ date +%d%Y
15052019
luis@luis-VirtualBox:~$ date +%d
15
luis@luis-VirtualBox:~$ date +%m
05
luis@luis-VirtualBox:~$ date +%y
19
luis@luis-VirtualBox:~$ date +%Y
2019
```

**Figura 7.13**  
Ejemplo de uso del comando *date*.

**Figura 7.14**  
Integración del comando *date* con otros comandos.

```
luis@luis-VirtualBox:~$ echo "INCREMENTAL_backup`date +%d%m%Y`.tgz"
INCREMENTAL_backup15052019.tgz
luis@luis-VirtualBox:~$ echo "INCREMENTAL_backup$(date +%d%m%Y).tgz"
INCREMENTAL_backup15052019.tgz
```

Para realizar una copia incremental con *tar*, debemos generar un archivo con metadatos sobre los archivos que puedan cambiar al realizar una copia incremental. Por ello, lo más conveniente es realizar una copia total, generando dicho archivo con la opción *-g*.

```
luis@luis-VirtualBox:~$ tar -czf "TOTAL_DocsLuis_`date +%d_%m_%Y`.tgz" -g backup.minf /home/luis/Documentos/*
tar: Eliminando la '/' inicial de los nombres
tar: Eliminando la '/' inicial de los objetivos de los enlaces
luis@luis-VirtualBox:~$ ls -l TOTAL*
-rw-r--r-- 1 luis luis 36 may 14 23:53 backup.minf
-rw-r--r-- 1 luis luis 978 may 14 23:53 TOTAL_DocsLuis_14_05_2019.tgz
```

**Figura 7.15**  
Generación de copia total y archivo de metadados incremental con *tar*.

A continuación, realizamos la copia incremental, empleando el mismo archivo de metadatos, por lo que el contenedor solo almacenará aquellos archivos modificados con respecto a la información contenida en el archivo de metadatos.

```
luis@luis-VirtualBox:~$ touch ./Documentos/nuevoDocumento.txt
luis@luis-VirtualBox:~$ tar -czf "INC_DocsLuis_`date +%d_%m_%Y`.tgz" -g backup.minf /home/luis/Documentos/*
tar: Eliminando la '/' inicial de los nombres
tar: Eliminando la '/' inicial de los objetivos de los enlaces
luis@luis-VirtualBox:~$ ls -l *.tgz backup*
-rw-r--r-- 1 luis luis 35 may 14 23:50 backup.minf
-rw-r--r-- 1 luis luis 144 may 14 23:50 INC_DocsLuis_14_05_2019.tgz
-rw-r--r-- 1 luis luis 978 may 14 23:53 TOTAL_DocsLuis_14_05_2019.tgz
luis@luis-VirtualBox:~$ tar -tf INC_DocsLuis_14_05_2019.tgz
/home/luis/Documentos/nuevoDocumento.txt
```

**Figura 7.16**  
Ejemplo de copia incremental con *tar*.

En este ejemplo se ha creado un nuevo archivo *nuevoDocumento.txt*, y es el único que se incluye en la copia de seguridad.

Cuando se desee restaurar los archivos backup (el total y los incrementales), estos se deberán de hacer en el mismo orden y empleando la opción *-G* (indica que la reposición será incremental) regenerando en el orden adecuado los archivos afectados.

```
tar -xvf backupTotal.tgz
tar -xvf backupIncl.tgz
tar -xvf backupInc2.tgz
...

```

Con *tar* podemos hacer copias diferenciales con respecto a una fecha mediante la opción *-N fecha*. Gracias a esta opción, *tar* almacenará aquellos archivos cuya fecha *cime* sea más reciente que la fecha indicada con *-N*.

En este caso, almacena aquellos archivos del directorio */home/Luis/Documentos* cuyo *cime* ha cambiado desde el 14 de mayo del 2019.

Para automatizar las copias de seguridad, se combinan los comandos *crontab* y *tar*, de tal manera, que podamos realizar *backups* sobre los directorios objeto de la planificación.

```

luis@luis-VirtualBox:~$ tar -czf "DIF.DocsLuis_date +%d_%m_%Y".tgz /home/luis/Documentos/* -N 14-may-19
tar: Eliminando la '/' inicial de los nombres
tar: Eliminando la '/' inicial de los nombres
luis@luis-VirtualBox:~$ ls -l DIF.DocsLuis_14_05_2019.tgz
luis@luis-VirtualBox:~$ tar -tf DIF.DocsLuis_14_05_2019.tgz
/home/luis/Documentos/almacen
/home/luis/Documentos/propietario
/home/luis/Documentos/prueba_unmask.inf
/home/luis/Documentos/luis
/home/luis/Documentos/luis.cron
/home/luis/Documentos/nuevodocumento.txt
/home/luis/Documentos/propietario
/home/luis/Documentos/propietario.txt
/home/luis/Documentos/prueba_unmask?.txt
/home/luis/Documentos/prueba_unask.txt
/home/luis/Documentos/reunion.txt
/home/luis/Documentos/s1
/home/luis/Documentos/tareas

```

**Figura 7.17**  
Ejemplo de copia diferencial con tar.

### Actividad resuelta 7.5



Realizar una copia de seguridad total de las carpetas /home y /root el día 1 de cada mes de forma comprimida con bzip2 en el dispositivo /mnt/backup. Realizar copias de seguridad incremental de dichas carpetas cada domingo.

#### SOLUCIÓN

Al hacer copias de seguridad de carpetas del sistema, debemos editar con privilegios de superusuario y ejecutar crontab -e con root, añadiendo las siguientes líneas:

```

00 01 1 * * tar -cjf /mnt/backup/backupTotal_`date +%d%m%Y`_home_root.tbz
-g backup`date +%m%Y`.minf /home /root
00 01 * * 6 tar -cjf /mnt/backup/backupInc_`date +%d%m%Y`_home_root.tbz -g
backup`date +%m%Y`.minf /home /root

```

Por último, reiniciamos el demonio cron.

### B) Por interfaz gráfica en Windows

Microsoft Windows 10 ofrece la posibilidad de realizar copias de seguridad de manera fácil mediante ‘Copia de seguridad’ dentro de ‘Actualización y seguridad’ de ‘Configuración’.

De esta manera, podemos agregar una unidad, donde se almacenará la copia de seguridad (se requiere, al menos, de otra unidad distinta a la de instalación de Windows). Una vez añadida, podremos gestionar la copia de seguridad pulsando en ‘Más opciones’.

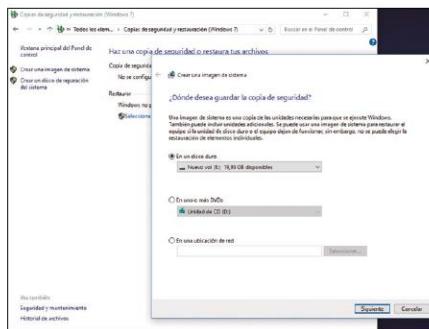


**Figura 7.18**  
Historial de archivos.

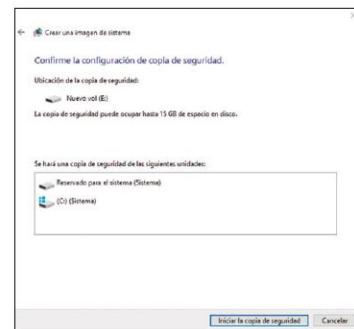
Podremos establecer la frecuencia, la duración, las carpetas (por defecto, las propias de la cuenta del usuario) y también podemos acceder a la configuración avanzada. En esta última, se muestra la misma ventana que 'Historial de archivos' del 'Panel de control', donde se especifican algunas opciones de configuración más concretas.

Además, Microsoft Windows 10 permite realizar una 'Copia de imagen del sistema'. Esta función, heredada de Microsoft Windows 7, permite crear una copia de seguridad de los archivos del sistema operativo, como si de una instantánea se tratara. Esta función es muy útil cuando el sistema operativo entra en un estado inconsistente debido a diferentes errores en el sistema, de tal manera que podemos restaurarlo en un breve espacio de tiempo.

Para ello, podemos acceder desde la opción de 'Copias de seguridad y restauración (Windows 7)' del 'Panel de control'. En ella, podemos pulsar en 'Crear una imagen de sistema', seleccionando un disco duro (una unidad donde no se encuentre ningún espacio reservado o utilizado por el sistema operativo), en DVD o en unidades remotas de red.



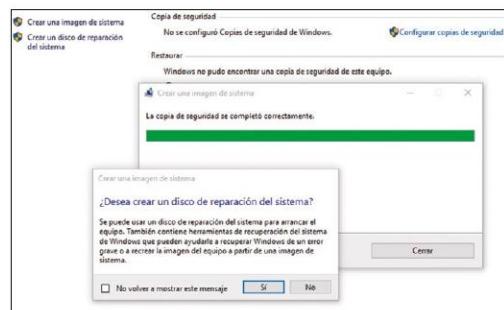
**Figura 7.19**  
Copia de imagen del sistema.



**Figura 7.20**  
Inicio de la copia del sistema.

Al pulsar en 'Siguiente', podremos acceder a 'Iniciar la copia de seguridad'.

Cuando acaba el proceso, nos da la opción de crear un disco de reparación del sistema, a través del cual se accede a un menú con distintas opciones de recuperación (esta opción siempre es recomendable) en caso de iniciar el sistema por este medio.



**Figura 7.21**  
Creación del disco de reparación  
del sistema en Windows 10.

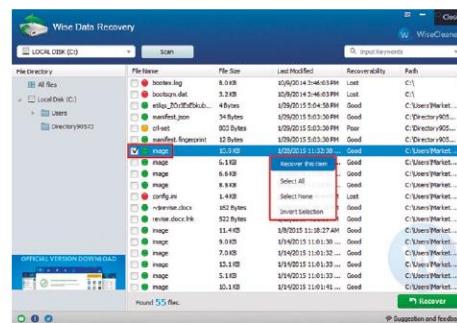
### 7.7.3. Recuperación de datos

Llegado el caso en el que se hayan eliminado los datos fortuita o intencionalmente, juegan un papel crucial las copias de seguridad. No obstante, estas últimas no cubren temporalmente todos los escenarios de generación de copias de seguridad ante una eliminación de datos.

Es en estas situaciones cuando se intenta recuperar los datos. Esto es posible porque, tal y como estudiamos en el capítulo 2, los bloques de datos se encuentran físicamente en los medios de almacenamiento en lugares distintos a su índice (*i-nodo* en *ext4* y *MFT* en *NTFS*). Así, cuando un archivo es borrado, el sistema de archivos marca los bloques de datos como espacio libre. Sin embargo, los datos quedan inalterados y en estado recuperable, a menos que hayan sido sobrescritos.

La recuperación de datos es el último recurso para restaurar los datos, y nunca se debe considerar como un instrumento de generación de copias de seguridad. Por ello, los datos más críticos han de tener un buen respaldo, tanto a nivel de copias de seguridad como RAID. Además, el estado de los discos ha de estar monitorizado continuamente con objeto de sustituirlos antes de un posible fallo.

Existen empresas especializadas en recuperación de datos que emplean técnicas muy sofisticadas a nivel de software y físico. No obstante, el uso de las aplicaciones de recuperación de datos puede ayudar en algunos casos, aunque no sean completamente certeras. Si tenemos la necesidad de recuperar datos, debemos inmediatamente limitar el uso del medio de almacenamiento afectado al mínimo para evitar la sobrescritura de datos y ejecutar el software de recuperación o solicitar ayuda a estas empresas.



**Figura 7.22**  
Wise Data Recovery. Web.

www

### Recursos web

Algunos de los programas más utilizados para recuperar datos son:

- Recuva (<https://www.ccleaner.com/recuva>).
- EaseUS Data Recovery (<https://es.easeus.com/data-recovery-software/data-recovery-wizard-free.html>).
- Disk Drill (<https://www.cleverfiles.com/>).
- Wise Data Recovery (<https://www.wisecleaner.com/wise-data-recovery.html>).

## 7.8. Documentación técnica

En los sistemas informáticos es necesario la elaboración de nueva documentación técnica y la correcta interpretación de la ya existente asociada con la instalación, mantenimiento y explotación de aplicaciones. Por ello, se necesita:

- ✓ Crear documentación técnica asociada al ciclo de vida de las aplicaciones en el sistema, detallando los procesos de instalación, configuración, mantenimiento o desinstalación de aplicaciones.
- ✓ Buscar documentación técnica de los proveedores de aplicaciones, asociada a los programas que se van a instalar, configurar, mantener o desinstalar.

La documentación de la vida del software es un procedimiento primordial para el correcto mantenimiento y explotación del sistema informático. Tradicionalmente, el proceso de documentación se percibe como una tarea tediosa y poco fructífera. Sin embargo, si esta se realiza como un procedimiento inherente a la instalación y mantenimiento de los programas, obtendremos un menor tiempo de respuesta en la resolución de errores. Y también favorecerá un mayor conocimiento de las aplicaciones por el administrador en pro de una mejor eficiencia en su ejecución.

### 7.8.1. Elaboración de documentación

En el proceso de documentación, se deben elaborar y completar un conjunto de modelos o formularios donde se reflejen los aspectos más relevantes en la instalación, mantenimiento, actualización y desinstalación del software en cada equipo:

- a) Sistema operativo:
  - Identificador: código alfanumérico único.
  - Fecha y hora de instalación y actualizaciones.
  - Detalles hardware originales del equipo: localización del equipo, procesador, memoria RAM, discos duros, adaptadores de red y gráficos, periféricos, etc.
  - Sistema operativo: nombre comercial, versión, licencia, clave del producto, usuarios (perfil y login), etc.
  - Actualizaciones: identificador único sobre el identificador del sistema operativo, fecha y hora, modificaciones hardware con respecto al original, descripción de la actualización (detallar la nueva versión o paquete).
- b) Aplicación:
  - Identificador: código alfanumérico único.
  - Nombre y versión: nombre comercial del software y su versión.
  - Fecha y hora de la instalación y actualizaciones.
  - Descripción del software.
  - Identificadores del sistema operativo y de la actualización del mismo sobre el que se instala.
  - Actualizaciones: identificador único sobre el identificador de la aplicación, fecha y hora, descripción de la actualización.
  - Hardware vinculado: dispositivos hardware asociados en su ejecución.
  - Desinstalación: si se ha desinstalado, indicar la fecha y los motivos.

### 7.8.2. Métodos de búsqueda de documentación técnica en Internet

La búsqueda fiable de documentación técnica para realizar tareas de instalación, mantenimiento y explotación de aplicaciones se lleva a cabo a través de las propias páginas proveedoras del software o también mediante buscadores de Internet.

**Recurso web**

Aquí puedes ver documentación técnica del proceso de instalación de LibreOffice.

Los buscadores relacionan páginas web con los contenidos de estas, para así mostrar un listado de todas las páginas asociadas a los parámetros de búsqueda introducidos por los usuarios. Para ello, previamente los buscadores utilizan herramientas llamadas *robots* o *spiders*, que rastrean páginas web, creando una gran base de datos con la información que contienen. Los buscadores de Internet hacen consultas a estas bases de datos mediante  *motores de búsqueda*, mostrando los resultados en un orden, según un algoritmo. Los algoritmos utilizados tienen en cuenta la frecuencia de búsqueda de las páginas, así como los enlaces publicitarios de pago por los que algunas empresas hacen uso para situarse en posiciones prioritarias.

Los buscadores más utilizados de forma general son Google, Bing, Baidu, Yahoo! Search, Yandex o DuckDuckGo.

Existen técnicas de búsqueda que permiten aumentar el éxito de localización de páginas web afines a las palabras clave, ajustando el acierto de los resultados devueltos por el buscador:

- Afinar en las palabras clave de búsqueda, es decir, que estas informen claramente sobre las páginas que contengan resultados asociados a nuestro interés.
- Indicar el mayor número de palabras clave posible y en el orden correcto, es decir, las más relevantes al comienzo.
- Utilizar operadores lógicos.

**CUADRO 7.4**  
Operadores lógicos más comunes

Operador	Descripción
Operador AND (Y)	Para buscar páginas que coincidan con las palabras clave o expresiones a ambos lados del operador. Este operador equivale al espacio entre palabras clave.
Operador OR (O)	Para buscar páginas que coincidan con una o las dos palabras clave que le rodea.
Operador NOT (-)	Para excluir una palabra clave, simplemente antepónéndolo a esta o con el carácter guion.
Operador comillas (" ")	Indica concordancias exactas. Para ello, enmarcamos la expresión de búsqueda entre comillas.

**RECUERDA**

- ✓ Las expresiones con operadores se pueden agrupar con paréntesis y volver a emplear operadores.

El buscador más habitual, Google, presenta sugerencias de búsqueda conforme se está escribiendo. Además, presenta una apariencia en la que, una vez que se introducen las palabras clave y se pulsa 'Buscar en Google', mostrará:

- a) En la parte alta distintos filtros de búsqueda: 'Todo', 'Imágenes', 'Vídeos', 'Noticias', 'Libros', 'Más' (Maps, Shopping, Vuelos, Finance) y otros pulsando en 'Herramientas'. Además, aparecerá el número de resultados aproximados y el tiempo de búsqueda.
- b) En la parte media, los diez primeros resultados ordenados en la primera página. Cada resultado consta de:
  - El título de la página web en color azul.
  - La URL de la página web en color verde y una flecha con la que, al desplegarla, podemos acceder a:
    - *En cache*: para obtener la página que dispone el buscador almacenada, cuando esta fue indexada, y no la actual (aunque pueden coincidir si no ha sido actualizada).
    - *Similares*: que redirecciona a otras páginas con contenido semejante.
  - Porción de texto de la página web, donde se resaltan en negrita las palabras clave. Si una palabra clave no aparece en la página, esta aparecerá tachada al final de esta porción antepuesta por *Falta*.
- c) En la parte baja (tras los resultados), aparecen búsquedas relacionadas con las palabras clave y un índice sobre otras páginas con resultados.

Si se desean realizar búsquedas avanzadas, se puede acceder a través de la parte alta, pulsando en 'Configuración' y 'Búsqueda avanzada'. Mostrará un formulario donde se pueden indicar múltiples opciones para afinar la búsqueda.

**Actividad propuesta 7.10**

Encuentra, mediante el uso de servicios de búsqueda de Internet, la documentación técnica asociada a la placa base de tu equipo.

**Resumen**

- Los sistemas informáticos actuales, hacen uso de aplicaciones informáticas o servicios locales o externos. Cuando estas aplicaciones son ejecutadas en el propio sistema operativo, se debe tener un conocimiento de los tipos que existen, atendiendo a su

licencia, ya que se establecen acuerdos para su explotación, instalación, distribución, publicación o estudio, como, por ejemplo:

- Software libre.
  - Software propietario.
  - Software con copyleft.
  - Software de dominio público.
  - Software sin copyleft.
  - Software open source.
- Además, la mayoría del software utilizado es software de aplicación, es decir, empleados para realizar tareas concretas y cotidianas por parte de un usuario final, como las herramientas ofimáticas (procesadores de texto, hojas de cálculo, software de presentación, etc.). Tanto para este como para cualquier tipo de software, deben conocerse de antemano los requisitos mínimos y recomendados para poder ejecutarlos de manera más eficiente en condiciones de cierto estrés.
  - Por otro lado, las aplicaciones que se ejecutan en el equipo pero que dependen de servicios ofrecidos por Internet, encontramos el correo electrónico, la mensajería instantánea, la transferencia de ficheros y, cómo no, la computación y almacenamiento en la nube. Este último presenta un crecimiento constante gracias al aumento del ancho de banda de acceso a Internet, lo que permite el uso de tecnologías que facilitan a los clientes conseguir los servicios y la potencia necesarios en cada uno de ellos bajo demanda.
  - En cuanto a las aplicaciones relacionadas con el software del sistema, tanto usuarios administradores como usuarios finales han de estar familiarizados con:
    - Software antimalware, los distintos tipos de ataques y amenazas.
    - Herramientas de clonación.
    - Software de copias de seguridad.
    - Aplicaciones de recuperación de datos.
  - Por último, no se debe olvidar que el software en un sistema informático es un proceso vivo que ha de mantenerse y, por tanto, documentarse para el correcto mantenimiento y explotación del sistema informático. La documentación ha de elaborarse con objeto de detallar el ciclo de vida de las aplicaciones en el sistema, así como buscar documentación técnica de los proveedores de dichos programas.



### Ejercicios propuestos

1. Indica tres ejemplos de software libre y software de código abierto. ¿Cuáles son las libertades que definen al software libre? ¿Cuáles son los diez términos que deben cumplir las licencias de código abierto?
2. Instala la suite ofimática *LibreOffice*, previa descarga desde <https://es.libreoffice.org/>.
  - a) Ejecuta *LibreOffice Writer* y practica con formato de página y de fuente, formato de párrafo (numeración y viñetas, espaciados, alineaciones, sangrías y tabulaciones) y creación de tablas.

- b) Ejecuta *LibreOffice Writer* y practica con formato de celdas y formato de datos, referencias a celdas, fórmulas y funciones básicas, y generación de gráficos.
- c) Ejecuta *LibreOffice Impress* y practica con las transiciones, la disposición de los elementos de una diapositiva, incorporando texto, imágenes, enlaces, tablas y objetos multimedia.
3. ¿Cuáles son las ventajas y desventajas de los protocolos POP3 e IMAP4?
4. Busca en Internet, instala y prueba una aplicación de mensajería instantánea que no hayas utilizado. Observa las similitudes y diferencias con respecto a otras que ya conozcas.
5. Prueba dos antivirus y dos antimalware, estudiando los tipos de protecciones, ante qué ataques son efectivos y qué otras utilidades de interés ofrecen.
6. Descarga *Clonezilla live* (<https://www.clonezilla.org/>) y lee su manual de usuario en <https://clonezilla.org/clonezilla-live-doc.php>. Crea una imagen del disco del sistema y, posteriormente, restaurala en otro disco. Además, realiza una clonación de una partición en otro disco.
7. En Ubuntu, configura el sistema de forma automática y periódica para realizar una copia de seguridad total de la carpeta */home* el día 1 de cada mes, de forma comprimida con *gzip* en */mnt/backup*. Además, realiza copias de seguridad incrementales de dichas carpetas cada sábado. Los nombres de la copia de seguridad deberán contener la fecha actual mediante el comando *date*.
8. En Ubuntu, realiza una copia de seguridad diferencial comprimida con *bzip2* de la carpeta */home* con respecto al día de ayer. El nombre de la copia de seguridad deberá contener la fecha actual mediante el comando *date*.
9. Entre los programas de recuperación de datos recomendados, descarga y prueba uno de ellos. Desde una unidad de disco o un pendrive, crea o copia un archivo de texto, una imagen y un vídeo, bórralos e intenta recuperarlos.
10. Con *LibreOffice*, crea un formulario de documentación sobre la instalación del sistema operativo y dos formularios sobre la instalación de un nuevo navegador web y un nuevo visor de PDF. Puedes seguir los aspectos de documentación del apartado 9.1 y aportar otros nuevos.

### ACTIVIDADES DE AUTOEVALUACIÓN

1. Indica qué tipo de licencia software es aquella que prohíbe o debe autorizar el uso, distribución o modificación del software:
- a) Software con copyleft.  
 b) Software libre.  
 c) Software propietario.
2. La principal diferencia entre licencias de software libre y licencia de código abierto es:
- a) El software libre es más pragmático que la licencia de código abierto.  
 b) La licencia de código abierto ha de cumplir cuatro términos y la de software libre diez libertades.

- c) La licencia de software libre está más orientada a la ética y la de código abierto es más pragmática.
3. ¿Qué tipo de licencia de distribución de Microsoft es aquella que es vendida a usuarios, independientemente del hardware de destino, pero destinada a un único equipo?:  
 a) Licencia de volumen.  
 b) Licencia retail.  
 c) Licencia OEM.
4. Los drivers de dispositivos, ¿qué tipo de software son, según su propósito?:  
 a) Software de sistema.  
 b) Software de desarrollo de aplicaciones.  
 c) Software de aplicación.
5. ¿Qué tipo de aplicación ofimática es base y a qué suite pertenece?:  
 a) Hoja de cálculo de Microsoft Office.  
 b) Gestor de base de datos de OpenOffice y LibreOffice.  
 c) Procesador de textos de OpenOffice y LibreOffice.
6. El protocolo SMTP es utilizado para:  
 a) Acceder y descargar los correos desde los buzones de los servidores de correo a las aplicaciones de correo cliente. Se descargan localmente y se eliminan del servidor de correo.  
 b) Acceder a los buzones de correo y descargar los correos sin eliminarlos.  
 c) Enviar correos entre servidores de correos, o desde un cliente de correo a un servidor de correo.
7. Son modelos de servicios de computación en la nube, según los servicios gestionados por los usuarios a nivel de arquitectura de computación:  
 a) SaaS, WaaS, IaaS.  
 b) SaaS, PaaS, LaaS.  
 c) SaaS, PaaS, IaaS.
8. ¿Qué herramientas para el aseguramiento de los datos y su posterior recuperación permiten trabajar de forma concreta sobre archivos individuales sin interrumpir el servicio ofrecido por las particiones o discos que los contienen?:  
 a) Clonaciones.  
 b) Copias de seguridad.  
 c) Ambas.
9. La recuperación de datos de medios de almacenamiento por causas intencionadas o fortuitas:  
 a) Es un instrumento más de generación de copias de seguridad.  
 b) Se considera un último recurso, pero no como copia de seguridad.  
 c) Siempre es una solución completamente segura para restaurar los archivos a su situación original.

10. El proceso de generación y búsqueda de documentación técnica asociada con la instalación, mantenimiento, actualización, explotación y desinstalación de aplicaciones en los sistemas informáticos es necesario para:
- a) El correcto mantenimiento de las aplicaciones, así como para obtener un menor tiempo de respuesta en la resolución de errores.
  - b) La reparación y sustitución de componentes hardware del equipo.
  - c) Obtener un listado de todas aquellas aplicaciones que hayan sido explotadas.

**SOLUCIONES:**1. **a** **b** **c**2. **a** **b** **c**3. **a** **b** **c**4. **a** **b** **c**5. **a** **b** **c**6. **a** **b** **c**7. **a** **b** **c**8. **a** **b** **c**9. **a** **b** **c**10. **a** **b** **c**

# Table of Contents

- Cubierta
- pagina del titulo
- Página de derechos de autor
- Índice
- Presentación
- 1. Fundamentos de los sistemas informáticos y las máquinas virtuales
  - Objetivos
  - Mapa conceptual
  - Glosario
  - 1.1. Introducción
  - 1.2. Arquitectura de un sistema informático. Modelos
  - 1.3. Componentes hardware de un sistema informático
    - 1.3.1. Microprocesador
    - 1.3.2. Memoria principal
    - 1.3.3. Placa base
    - 1.3.4. Dispositivos de almacenamiento secundario
    - 1.3.5. Fuente de alimentación
  - 1.4. Controladores de dispositivos. Instalación de drivers
    - 1.4.1. Administración de dispositivos en Microsoft Windows
    - 1.4.2. Administración de dispositivos en Ubuntu Desktop
  - 1.5. Componentes software de un sistema informático
    - 1.5.1. Tipos de software
    - 1.5.2. El sistema operativo
  - 1.6. Proceso de arranque de un sistema informático. POST
  - 1.7. Máquinas virtuales
    - 1.7.1. Concepto y usos
    - 1.7.2. Software de virtualización
  - 1.8. Oracle VM VirtualBox
    - 1.8.1. Proceso de instalación de Oracle VM VirtualBox
    - 1.8.2. Entorno de Oracle VM VirtualBox
    - 1.8.3. Creación de una máquina virtual en Oracle VM VirtualBox
    - 1.8.4. Creación de instantáneas
  - 1.9. Normas de seguridad y prevención de riesgos laborales

- Resumen
  - Ejercicios propuestos
  - Actividades de autoevaluación
- 2. Sistemas operativos. Introducción
  - Objetivos
  - Mapa conceptual
  - Glosario
  - 2.1. Introducción
  - 2.2. Funciones y características
  - 2.3. Tipos de sistemas operativos
  - 2.4. Arquitecturas de los sistemas operativos
    - 2.4.1. Sistemas con capas o anillos
    - 2.4.2. Sistemas monolíticos
    - 2.4.3. Microkernel
    - 2.4.4. Kernel híbrido
    - 2.4.5. Arquitecturas de sistemas operativos actuales
  - 2.5. Versiones de los sistemas operativos más utilizados
    - 2.5.1. Sistemas operativos de Microsoft
    - 2.5.2. Sistemas operativos GNU/Linux
    - 2.5.3. Sistemas operativos de Apple
  - 2.6. Instalación de un sistema operativo
    - 2.6.1. Requisitos
    - 2.6.2. Planificación y consideraciones previas
    - 2.6.3. Proceso de instalación de Ubuntu Desktop en Oracle VM VirtualBox
    - 2.6.4. Proceso de instalación de Microsoft Windows 10 Pro en Oracle VM VirtualBox
  - 2.7. Instalaciones desatendidas
    - 2.7.1. Instalación desatendida de Windows 10
    - 2.7.2. Instalación desatendida de Ubuntu
  - 2.8. Proceso de arranque del sistema operativo. Gestores de arranque
    - 2.8.1. Conceptos previos: esquemas de particiones
    - 2.8.2. Gestor de arranque de Windows
    - 2.8.3. Gestor de arranque de Linux
  - 2.9. Actualización del sistema operativo
    - 2.9.1. Administración de actualizaciones en Windows

- 2.9.2. Administración de actualizaciones en Ubuntu Desktop
  - 2.10. Identificación, instalación y desinstalación de aplicaciones
    - 2.10.1. Aplicaciones y características de Windows
    - 2.10.2. Software de Ubuntu
  - Resumen
  - Ejercicios propuestos
  - Actividades de autoevaluación
3. Sistemas operativos. Gestión de archivos y almacenamiento
- Objetivos
  - Mapa conceptual
  - Glosario
  - 3.1. Introducción
  - 3.2. Sistemas de archivos
    - 3.2.1. FAT (File Allocation Table)
    - 3.2.2. exFAT
    - 3.2.3. NTFS
    - 3.2.4. APFS
    - 3.2.5. ext4 (Fourth extended file system)
  - 3.3. Estructura de directorios en Linux y Microsoft Windows
    - 3.3.1. Estructura de directorios en GNU/Linux
    - 3.3.2. Estructura de directorios en Microsoft Windows
  - 3.4. Gestión de archivos por línea de comandos en Linux
    - 3.4.1. Tipos de ficheros
    - 3.4.2. Eliminación de ficheros
    - 3.4.3. Creación y eliminación de directorios
    - 3.4.4. Copia de archivos
    - 3.4.5. Renombrado o movimiento de archivos
    - 3.4.6. Impresión de archivos
    - 3.4.7. Cuenteo de un fichero
    - 3.4.8. Ordenación de un fichero
    - 3.4.9. Entrada y salidas estándar. Redirecciones
    - 3.4.10. Procesamiento de textos
  - 3.5. Gestión de archivos por interfaz gráfica en Microsoft Windows
  - 3.6. Gestión de almacenamiento por línea de comandos en Linux
    - 3.6.1. Montaje y desmontaje
    - 3.6.2. Particionar

- 3.6.3. Formatear
  - 3.6.4. Desfragmentación
  - 3.6.5. Chequeo
  - 3.6.6. RAID
  - 3.7. Gestión de almacenamiento por interfaz gráfica en Microsoft Windows
  - 3.8. Búsqueda de información por línea de comandos en Linux
    - 3.8.1. Criterios de búsqueda
  - 3.9. Búsqueda de información por interfaz gráfica en Microsoft Windows
  - Resumen
  - Ejercicios propuestos
  - Actividades de autoevaluación
4. Sistemas operativos. Gestión de usuarios y procesos
- Objetivos
  - Mapa conceptual
  - Glosario
  - 4.1. Introducción
  - 4.2. Gestión de usuarios por línea de comandos en Linux
    - 4.2.1. Configuración de usuarios y grupos
    - 4.2.2. Comandos de gestión de usuarios
    - 4.2.3. Usuarios y grupos predeterminados
    - 4.2.4. Seguridad de cuentas de usuarios y contraseñas
    - 4.2.5. Acceso a recursos y permisos locales
    - 4.2.6. Modificación de permisos
    - 4.2.7. Permisos por defecto
    - 4.2.8. Configuración de perfiles
  - 4.3. Gestión de usuarios por interfaz gráfica en Windows
  - 4.4. Gestión de procesos por línea de comandos en Linux
    - 4.4.1. Procesos y servicios
    - 4.4.2. Identificación y administración
  - 4.5. Gestión de procesos por interfaz gráfica en Windows
  - 4.6. Automatización de tareas en Linux
  - 4.7. Monitorización y gestión del sistema. Evaluación de prestaciones
  - 4.8. Aplicaciones para el mantenimiento y optimización del sistema

- Resumen
  - Ejercicios propuestos
  - Actividades de autoevaluación
- 5. Sistemas informáticos en red. Configuración y explotación
  - Objetivos
  - Mapa conceptual
  - Glosario
  - 5.1. Introducción
  - 5.2. Protocolos principales de red
    - 5.2.1. Protocolo Ethernet
    - 5.2.2. Protocolo Wi-Fi
    - 5.2.3. Protocolo IPv4 e IPv6
    - 5.2.4. Protocolo TCP y UDP
  - 5.3. Configuración del protocolo TCP/IP
    - 5.3.1. Estática
    - 5.3.2. Dinámica
  - 5.4. Interconexión de redes. Componentes
    - 5.4.1. Switch
    - 5.4.2. Router. Tablas de enrutamiento
    - 5.4.3. Topología física y lógica. Mapas
    - 5.4.4. Dominios de colisión y difusión
  - 5.5. Tipos de redes
  - 5.6. Acceso a redes WAN. Tecnologías
    - 5.6.1. Conexiones WAN privadas
    - 5.6.2. Conexiones WAN públicas
  - 5.7. Redes cableadas
    - 5.7.1. Tipos y características
    - 5.7.2. Dispositivos de interconexión
    - 5.7.3. Adaptadores
  - 5.8. Redes inalámbricas
    - 5.8.1. Tipos y características
    - 5.8.2. Dispositivos de interconexión
    - 5.8.3. Adaptadores
  - 5.9. Ficheros de configuración de red
  - 5.10. Monitorización y verificación de una red mediante comandos
    - 5.10.1. Gestión de puertos

- 5.11. Resolución de problemas
- 5.12. Seguridad en las comunicaciones
  - 5.12.1. Políticas de seguridad
  - 5.12.2. Tipos de ataques
  - 5.12.3. Mecanismos de seguridad en las comunicaciones

Resumen

Ejercicios propuestos

Actividades de autoevaluación

## 6. Gestión de recursos en red de un sistema informático

Objetivos

Mapa conceptual

Glosario

### 6.1. Introducción

### 6.2. Permisos

6.2.1. Permisos de red y locales

6.2.2. Compartir archivos o carpetas

6.2.3. Herencia

6.2.4. ACL

### 6.3. Derechos de usuarios

6.3.1. Directivas de seguridad. Objetos y ámbito de directivas

6.3.2. Plantillas

### 6.4. Requisitos de seguridad del sistema y de los datos. Seguridad a nivel de usuarios y de equipos

### 6.5. Servidores

6.5.1. Servidor de ficheros

6.5.2. Servidor de impresión

6.5.3. Servidor de aplicaciones

### 6.6. Conexión remota. Herramientas

### 6.7. Herramientas de seguridad

6.7.1. Cifrado

6.7.2. Administración y análisis

6.7.3. Cortafuegos

6.7.4. Sistemas de detección de intrusión

6.7.5. OpenSSH

Resumen

Ejercicios propuestos

## Actividades de autoevaluación

### 7. Aplicaciones informáticas

Objetivos

Mapa conceptual

Glosario

7.1. Introducción

7.2. Tipos de software

    7.2.1. Clasificación por licencia

    7.2.2. Clasificación por propósito

7.3. Requisitos mínimos y recomendados

7.4. Herramientas ofimáticas

    7.4.1. Procesadores de texto

    7.4.2. Hojas de cálculo

    7.4.3. Software de presentación

    7.4.4. Sistemas gestores de bases de datos

7.5. Herramientas de Internet

    7.5.1. Correo electrónico

    7.5.2. Mensajería instantánea

    7.5.3. Transferencia de ficheros

7.5.4. Computación y almacenamiento en la nube

7.6. Software antimalware

7.7. Clonación y copias de seguridad

    7.7.1. Clonaciones

    7.7.2. Copias de seguridad

    7.7.3. Recuperación de datos

7.8. Documentación técnica

    7.8.1. Elaboración de documentación

    7.8.2. Métodos de búsqueda de documentación técnica en Internet

Resumen

Ejercicios propuestos

Actividades de autoevaluación