

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 desee 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 interactuar 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 manejabilidad 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 aviónica, 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 desplegados 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 microcomputadores, mainframes, supercomputadores, etc.



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 computadores.
- 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.



Actividades propuestas

- 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.

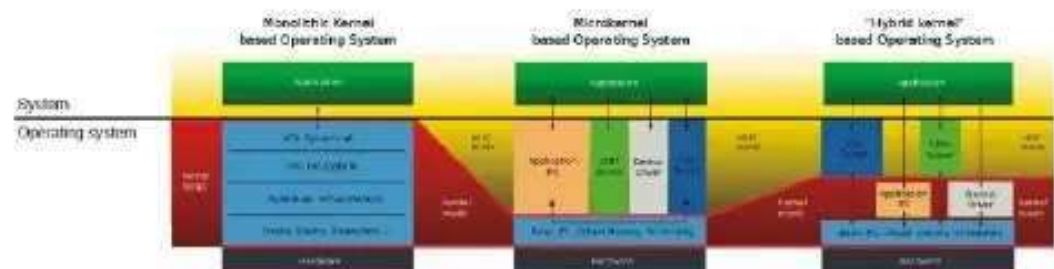


Figura 2.3

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 un 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/s> 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.

o 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.



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 ventada 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 'Agregar 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. Después indicariamos la imagen del sistema operativo para instalar.
- Dejar vacío. Posteriormente, podríamos añadir el disco virtual pulsando en el disco 'Vacío', 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.

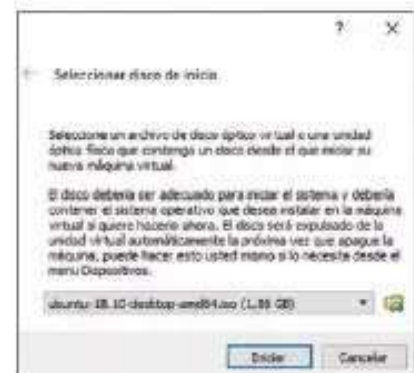


Figura 2.4

Selección de disco de instalación.

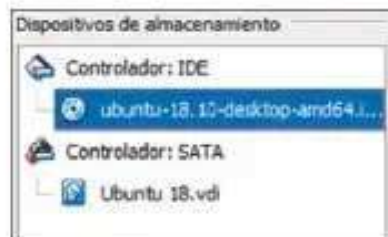


Figura 2.5
Dispositivos
de almacenamiento.



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 quisiéramos 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 estuviéramos 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
Solitud 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 icono 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 Virtual-Box.

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'.

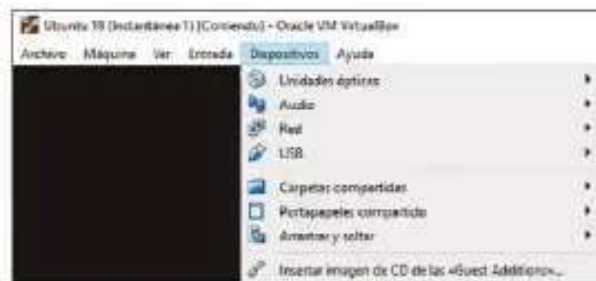


Figura 2.16
Opción de instalación
de las Guest Additions.



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 'Cortana', el cual nos solicita la región y la distribución del teclado.

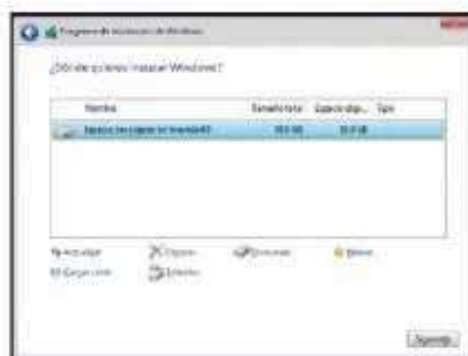


Figura 2.19
Creación de particiones.



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.

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/ls/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 continua 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.



Recurso digital 2.5

Estándar MBR.

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.



Figura 2.24
EFI en un sistema informático.

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.



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.

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 *bcdedit.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 *nisconfig*, 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 opciones 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 icono 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.