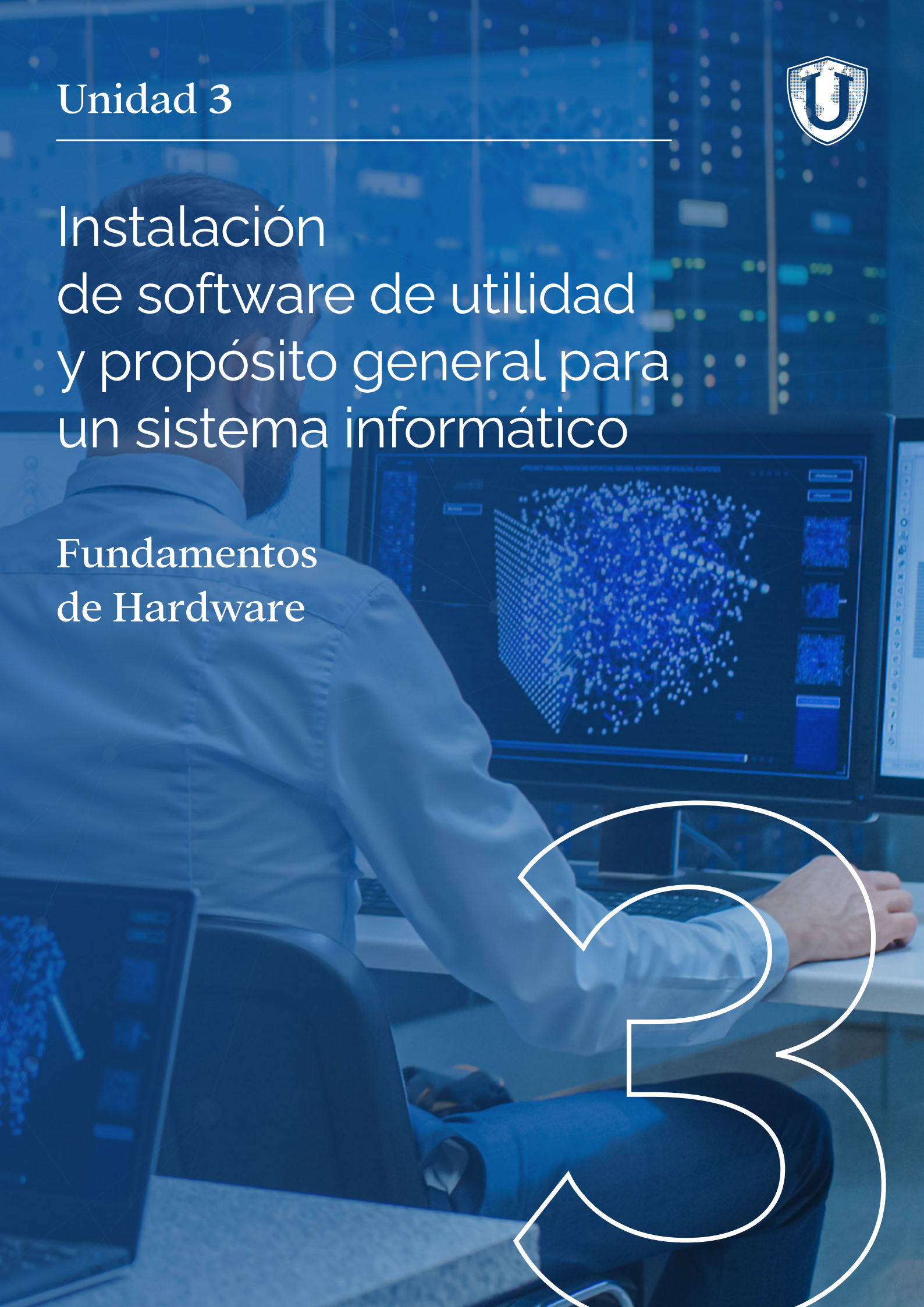
Unidad 3

Instalación de software de utilidad y propósito general para un sistema informático

Fundamentos de Hardware



Índice

Introducción

Los sistemas operativos están intrínsecamente ligados al hardware del sistema informático, por lo que es esencial comprender cómo se puede organizar y optimizar esta relación.

Los sistemas operativos nos proporcionan diversas herramientas y utilidades que nos ayudan a gestionar adecuadamente los recursos del sistema y a organizar sus aplicaciones y procesos de manera eficiente, minimizando el consumo de recursos. Estas herramientas y utilidades varían según la plataforma y el tipo de sistema operativo.

En esta unidad, exploraremos cómo funcionan los distintos tipos de sistemas operativos y cómo influyen en la eficiencia y la gestión del trabajo. Además, abordaremos el software de utilidad y de propósito general, que nos permite mejorar el rendimiento y la administración de nuestros sistemas informáticos.

Importante: A lo largo de esta unidad, es fundamental prestar atención a las mejores prácticas y consejos para mantener nuestros sistemas seguros y optimizados, ya que esto nos permitirá garantizar un rendimiento óptimo y proteger nuestra información.

Al finalizar esta unidad

- Conoceremos los distintos tipos de malware que pueden afectar a un sistema informático y cuáles son sus características.
- Comprenderemos el funcionamiento de los antivirus y cómo protegen nuestros sistemas.
- Analizaremos los distintos entornos operativos y cómo influyen en la gestión de recursos del sistema.
- Manejaremos utilidades de administración, mantenimiento y reparación para mejorar el rendimiento de un sistema informático.
- Seremos capaces de identificar los distintos tipos de aplicaciones existentes en el mercado y cómo pueden satisfacer nuestras necesidades.
- Entenderemos el funcionamiento interno de un compresor de archivos y cómo nos ayuda a optimizar el almacenamiento de datos.

3.1. Entornos operativos

El **software** y las **aplicaciones** pueden clasificarse en función de su uso en las siguientes categorías:

- **Software de sistema**: Es cualquier programa que interactúa directamente con el **hardware**, actuando como intermediario entre el **usuario** y los **componentes físicos** del equipo. El **sistema operativo** es el software de sistema más conocido, aunque otros dispositivos también incluyen **firmware**, que cumple funciones similares de control y gestión del hardware.
- **Herramientas de desarrollo**: conjunto de programas utilizados para el diseño, la implementación y desarrollo de software de sistema o de aplicaciones. Estas herramientas incluyen:
  - Editores.
  - Compiladores.
  - Depuradores de código.
  - Entornos de desarrollo integrados (IDE).
- **Software de aplicación:** tipo de software orientado a realizar tareas concretas y de uso cotidiano por parte del usuario como **aplicaciones ofimáticas, navegadores web, juegos y software de diseño.**

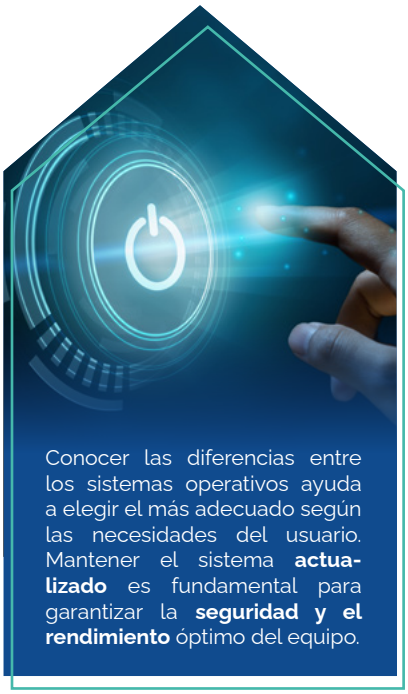
El sistema operativo

Podemos definir el sistema operativo como un conjunto de módulos que interactúan entre sí para realizar una gran cantidad de tareas. Está en continua evolución, ya que se actualiza para corregir problemas de seguridad y adaptarse a la evolución de los recursos hardware y software. Algunas de las características más importantes de los sistemas operativos son las siguientes:

- Actúa como **interfaz entre el usuario y el hardware**.
- Gestiona los recursos de hardware y software del sistema de forma imperceptible para el usuario.
- Facilita la interacción del usuario con la máquina, aprovechando todos los recursos del equipo.

Existe una gran variedad de sistemas operativos. Cada uno tiene características especiales y atiende a necesidades diferentes:

- **Windows** (Microsoft).
- **macOS** (Apple).
- **Linux** (diferentes distribuciones).
- **Android** (Google).



Conocer las diferencias entre los sistemas operativos ayuda a elegir el más adecuado según las necesidades del usuario. Mantener el sistema **actualizado** es fundamental para garantizar la **seguridad y el rendimiento** óptimo del equipo.

3.1.1. Firmware

El **firmware** es el software integrado en un hardware específico. A diferencia de un sistema operativo, su función es realizar tareas específicas y optimizar el hardware.

Nos encontramos dispositivos como **ChromeCast** o **Fire Stick TV** que incluyen firmware diseñado para controlar su funcionamiento y la interfaz del usuario. A menudo este software puede actualizarse para corregir errores y mejorar la seguridad.

3.1.2. Estructura de un sistema operativo móvil

Los sistemas operativos móviles mantienen una estructura en capas, donde cada una cumple una función específica y está en comunicación con los demás:

- **Interfaz de usuario y aplicaciones nativas.** incluye aplicaciones preinstaladas en el sistema, que no pueden eliminarse en la mayoría de los casos. Funcionan mediante ***API*** *(Application System Interface)* para interactuar con otras aplicaciones.
- **Interfaz de aplicaciones (API).** permite que las aplicaciones instaladas puedan **acceder a funciones** como notificaciones, llamadas, cámara o ubicación. Es fundamental para que las apps se integren con el resto del sistema.
- **Librerías/middleware. componentes intermedios** que proporcionan acceso a funciones esenciales como gráficos, bases de datos, reproducción multimedia o conexión a redes. Ayudan a que las aplicaciones funcionen correctamente sin necesidad de programar desde cero estas funciones.
- **Núcleo o Kernel.** es el **corazón del sistema operativo**. Se encarga de **gestionar los recursos** del sistema (memoria, procesos, entrada/salida, etc.) y garantizar que todas las capas superiores funcionen correctamente.
  - Android utiliza un **núcleo basado en Linux**.
  - iOS emplea un núcleo derivado de **macOS**, llamado **XNU**.

Estructura de un sistema operativo móvil
Interfaz de usuario y aplicaciones nativas
Interfaz de aplicaciones
Librerías de middleware
Kernel

3.2. Tipos de aplicaciones

El término **software** se utiliza tanto para referirse al **sistema operativo** como a las **aplicaciones** desarrolladas para interactuar con dicho sistema. Dentro de este último grupo, distinguimos entre aplicaciones orientadas a **realizar funciones específicas** y aplicaciones destinadas a **desarrollar nuevo software**.

En sus inicios, las aplicaciones eran **programas compilados** que se ejecutaban en un equipo específico. Sin embargo, con el avance de la tecnología, han surgido diferentes tipos de aplicaciones adaptadas a diversos entornos:

- **Aplicaciones web**. Son aplicaciones alojadas en un servidor web y se ejecutan directamente en el navegador del cliente. Parte del código se ejecuta en el servidor (por ejemplo, PHP) y otra parte en el navegador del cliente (por ejemplo, JavaScript). La mayor ventaja de estas aplicaciones es que suelen ser multiplataforma.
- **Apps móviles**. Aplicaciones diseñadas para ejecutarse en dispositivos móviles. Se dividen en dos categorías:
  - **Nativas:** desarrolladas específicamente para un sistema operativo (iOS o Android).
  - **Híbridas:** aplicaciones que utilizan un *WebView* para renderizar su contenido desde una página web, integrando elementos nativos del sistema operativo.
- **Widgets.** Pequeñas aplicaciones que ofrecen funciones complementarias en dispositivos móviles y sistemas operativos como Windows o macOS. Permiten visualizar información en tiempo real y realizar acciones rápidas sin necesidad de abrir una aplicación completa.
- **Plugins.** Similares a los widgets en algunos aspectos, pero estos se ejecutan sobre otros programas o aplicaciones directamente con la intención de aumentar su funcionalidad. También se les conoce como add-ons. Los plugins pueden extender las capacidades de un programa al agregar características adicionales, mejorar su rendimiento o proporcionar compatibilidad con nuevos formatos

NOTA

WebView: componente que permite a una aplicación mostrar contenido web dentro de su propia interfaz, sin necesidad de abrir un navegador externo.

3.2.1. Software de propósito general

Como indica su propio nombre, las aplicaciones de propósito general son aquellas que tienen funciones más generales y de uso muy común. Estas se contraponen a otro tipo de aplicaciones que solo pueden desarrollar una función para un puesto o sitio específico.

Las aplicaciones más comunes de este tipo de software son:

1. Aplicaciones de oficina u ofimática:
   - Hojas de cálculo
   - Editores de presentaciones
   - Visores de fotos
   - Gestores de bases de datos
   - Procesadores de texto
   - Visores de documentos
2. Productividad y negocios:
   - Agendas de contactos
   - Calculadoras
   - Utilidades de contabilidad
3. Juegos
4. Navegadores
5. Aplicaciones P2P (peer-to-peer)
6. Educación:
   - Enciclopedias
   - Diccionarios
   - Software educativo y de aprendizaje
7. Multimedia:
   - Reproductores de audio y video
   - Creación y edición de video
   - Editores de imágenes y diseño gráfico
8. Antimalware y soluciones de seguridad
9. Imagen y diseño:
   - Editores de imágenes
   - Herramientas de diseño gráfico
   - Modelado y animación 3D
10. Programación:
    - Entornos de desarrollo integrados (IDE)
    - Compiladores y depuradores
    - Herramientas de gestión de versiones y colaboración

3.2.2. Instalación y prueba de aplicaciones

Aunque hoy en día muchas aplicaciones son aplicaciones web que no necesitan de una instalación, sigue habiendo un cierto grupo que no poseen aplicación web (la mayoría poseen ambas).

Para las aplicaciones que deben ser instaladas sí o sí, los fabricantes de estas y sus desarrolladores han creado ciertos repositorios o tiendas donde almacenar las últimas versiones de estas con el fin de evitar así el malware.

Para poder descargar estas aplicaciones, habrá que cumplir con ciertos requisitos, como pagar una tasa o cumplir con características concretas; estos requisitos los marca el fabricante o desarrollador.

Las dos mayores tiendas de aplicaciones móviles son:

AppStore

Es la tienda de Apple en la cual se pueden descargar aplicaciones para todos los dispositivos Apple, y son aplicaciones en su mayoría específicas. Para poder publicar aplicaciones en esta tienda, es necesario ser un desarrollador únicamente de Apple y, además, las apps pasarán un exhaustivo control de calidad

Google Play

Se trata de la tienda de aplicaciones creada por Google para dispositivos Android.

Formas de instalación de aplicaciones según el sistema operativo:

1. Copia directa

La copia directa es una opción común para instalar aplicaciones en algunos sistemas operativos como macOS. Los programas utilizan librerías del sistema operativo, lo que permite una instalación sencilla mediante la copia del programa en la carpeta 'Aplicaciones'. Esta instalación tiene la ventaja de que el desarrollador del sistema sabe que la instalación de estas aplicaciones no va a alterar el sistema. Además, no suelen necesitar privilegios de administración para su instalación.

2. Instalación mediante instalador

La instalación mediante instalador es la forma más común de instalar programas en Windows. Después de la instalación, el programa se puede eliminar a través del Panel de control. El mayor problema de esta opción es que el propio sistema Windows almacena toda la información relativa a instalaciones y desinstalaciones en el llamado Registro del sistema, una gran base de datos con los registros de todo el sistema.

Esto puede requerir limpiezas periódicas del registro y, en ocasiones, del propio sistema. Aunque con el tiempo los registros se llenan menos, sigue siendo un inconveniente para los usuarios que utilizan este sistema a gran escala.

3. Instalación mediante un gestor de paquetes

La instalación mediante un gestor de paquetes es la opción más utilizada en los sistemas Linux. Las aplicaciones se almacenan en paquetes que el sistema instala y administra cuando lo solicitemos. Estos paquetes se almacenan en repositorios, algunos de los cuales son comunes del sistema y otros más específicos para ciertas aplicaciones de terceros. Dos sistemas de paquetes ampliamente utilizados son:

- APT (Advanced Packaging Tool): es la herramienta utilizada principalmente por el proyecto Debian y Ubuntu.
- RPM (Red Hat Package Manager): creada por la distribución Red Hat, también es característica en otras distribuciones como Mandriva y openSUSE, entre otras.

NOTA

Antes de instalar una aplicación, es recomendable leer las opiniones de otros usuarios y revisar la calificación otorgada a la aplicación. Además, siempre es conveniente instalar aplicaciones de fuentes confiables y mantener actualizado el software de seguridad en nuestro dispositivo.

3.2.3. Comparación de aplicaciones. Evaluación y rendimiento

Cuando se desarrolla un software, es crucial evaluar y comparar su rendimiento a lo largo del proceso de desarrollo. Esto permite a los desarrolladores y al equipo de pruebas asegurar que el software cumpla con las expectativas y funcione correctamente. La evaluación del rendimiento es un componente esencial para garantizar la calidad del software y garantizar que el producto final sea eficiente y efectivo.

Durante el desarrollo de software, se realizan varios tipos de pruebas para evaluar y mejorar el rendimiento de la aplicación, tales como:

- **Pruebas de unidad:** Estas pruebas se enfocan en componentes individuales del software para asegurar que cada uno funcione correctamente y según lo previsto.
- **Pruebas de integración:** Las pruebas de integración se realizan para garantizar que todos los componentes del software funcionen correctamente en conjunto.
- **Pruebas de sistema:** Estas pruebas evalúan el rendimiento del software en su totalidad, incluyendo la interacción con el sistema operativo, hardware y otros componentes externos.
- **Pruebas de rendimiento:** Estas pruebas se centran en evaluar la velocidad de procesamiento, la capacidad de respuesta, la estabilidad y la escalabilidad del software en diferentes condiciones de carga y uso. Algunos tipos de pruebas de rendimiento incluyen pruebas de carga, pruebas de estrés y pruebas de capacidad.
- **Pruebas de usabilidad:** Estas pruebas evalúan la facilidad de uso y la experiencia del usuario al interactuar con el software, garantizando que la interfaz de usuario y la funcionalidad cumplan con las expectativas de los usuarios finales.
- **Pruebas de seguridad:** Las pruebas de seguridad evalúan la capacidad del software para protegerse de ataques, vulnerabilidades y amenazas, garantizando que los datos y la información del usuario estén protegidos y seguros.

El proceso de evaluación y mejora del rendimiento es un aspecto crucial en el desarrollo de software de alta calidad. Al realizar pruebas exhaustivas y continuas, los desarrolladores y los equipos de pruebas pueden identificar y abordar problemas de rendimiento, garantizando que el software funcione de manera óptima y cumpla con los estándares de calidad necesarios para una experiencia de usuario satisfactoria.

3.3. Compresión y descompresión de archivos

La **compresión** de archivos es una **técnica fundamental** en la gestión de la información digital, cuyo objetivo es reducir el tamaño de los archivos para facilitar su almacenamiento, transmisión y manipulación. La **descompresión**, por su parte, es el **proceso inverso** que permite recuperar los datos en su forma original o cercana a la original, según el tipo de compresión empleado.

La elección del método de compresión adecuado depende del tipo de datos, del uso previsto y de la necesidad de mantener la integridad de la información. Existen dos grandes tipos de compresión: con pérdida y sin pérdida.

3.3.1. Tipos de algoritmos de compresión

Compresión con pérdida (lossy)

Los algoritmos de compresión con pérdida eliminan información que se considera prescindible o redundante para reducir el tamaño del archivo. Este proceso no es reversible: al descomprimir, no se obtiene una copia exacta del archivo original.

Este tipo de compresión es especialmente útil en contenidos multimedia (imágenes, audio y vídeo), donde la pérdida de calidad puede ser imperceptible para el usuario final y se compensa con una notable disminución del tamaño.

Ejemplos de algoritmos con pérdida:

- **JPEG (Joint Photographic Experts Group):** formato de imagen que reduce significativamente el tamaño eliminando detalles poco perceptibles.
- **MP3 (MPEG Audio Layer 3):** algoritmo de compresión de audio que elimina frecuencias no detectables por el oído humano.
- **H.264/H.265:** algoritmos de compresión de video que optimizan la calidad visual reduciendo el tamaño de los archivos.

Los **algoritmos con pérdida** no deben utilizarse en archivos donde la fidelidad de los datos sea crítica (por ejemplo, archivos de respaldo o documentos legales).

Compresión sin pérdida (lossless)

A diferencia de la compresión con pérdida, estos algoritmos garantizan que los datos descomprimidos sean idénticos a los originales. Se utilizan para archivos que requieren una integridad total de la información, como documentos de texto, bases de datos y archivos ejecutables.

Algunos de los algoritmos más comunes son:

- **Huffman:** basado en la codificación de símbolos más frecuentes con secuencias de bits más cortas. Es eficiente para datos con distribución estadística conocida.
- **Lempel-Ziv-Welch (LZW):** almacena secuencias de datos repetidas en forma de diccionario. Es utilizado en formatos como **GIF** y **TIFF**.
- **Deflate:** combinación de los algoritmos LZ77 y Huffman. Es la base de formatos ampliamente utilizados como **ZIP** y **PNG**.

Un archivo de texto de 1 MB puede reducirse hasta un 60-70 % con compresión sin pérdida, mientras que una imagen JPEG puede reducirse un 90 % con compresión con pérdida, aunque con ligera pérdida de calidad visual.

3.3.2. Formatos y herramientas de compresión

Además de los algoritmos, existen diversos formatos y herramientas que permiten la compresión y descompresión de archivos. Los más utilizados incluyen:

- ZIP:
  - Basado en el algoritmo Deflate.
  - Soportado nativamente por la mayoría de sistemas operativos (Windows, macOS, Linux).
  - Permite compresión de múltiples archivos y carpetas en un solo archivo comprimido.
- RAR:
  - Propietario, desarrollado por el software WinRAR.
  - Ofrece tasas de compresión más altas y opciones avanzadas como recuperación de errores.
  - Requiere software específico para su creación y descompresión.
- 7z:
  - Formato de código abierto que permite elegir entre varios algoritmos (por ejemplo, LZMA o LZMA2).
  - Muy eficiente en compresión de grandes volúmenes de datos.
  - Compatible con herramientas multiplataforma.
- TAR:
  - No aplica compresión por sí mismo; se utiliza para empaquetar archivos en sistemas Unix/Linux.
  - Comúnmente se combina con algoritmos como GZIP o BZIP2 para generar archivos comprimidos: .tar.gz, .tar.bz2.

NOTA

En entornos Linux, es habitual encontrar comandos como tar -czvf archivo.tar.gz directorio/, que combinan empaquetado y compresión en una sola instrucción.

3.4. Utilidades para el mantenimiento de y reparación de los sistemas informáticos

El mantenimiento preventivo y correctivo de los sistemas informáticos es una competencia esencial para cualquier técnico de sistemas. Existen numerosas herramientas diseñadas para diagnosticar, reparar y optimizar el funcionamiento de equipos, tanto a nivel físico como lógico. En esta sección se presentan algunas de las utilidades más representativas en diferentes sistemas operativos.

3.4.1. Recuperación del arranque (cargador)

En numerosas ocasiones, podemos enfrentarnos a situaciones en las que un equipo no arranca, y sospechamos que el problema puede estar relacionado con el cargador de arranque, independientemente del tipo que sea.

Rescatux es una distribución GNU/Linux basada en Debian, orientada a la recuperación de sistemas. Proporciona un entorno gráfico accesible para usuarios técnicos y permite reparar múltiples elementos relacionados con el arranque y la configuración del sistema.

Funciones principales de Rescatux:

- Reparación del cargador de arranque GRUB (usado en sistemas Linux).
- Restauración del **MBR (Master Boot Record)** en sistemas Windows y Linux.
- Restablecimiento de contraseñas de usuario y administrador.
- Diagnóstico y reparación de sistemas de archivos dañados.
- Administración de particiones.
- Comprobación de hardware básico y resolución de conflictos de arranque.

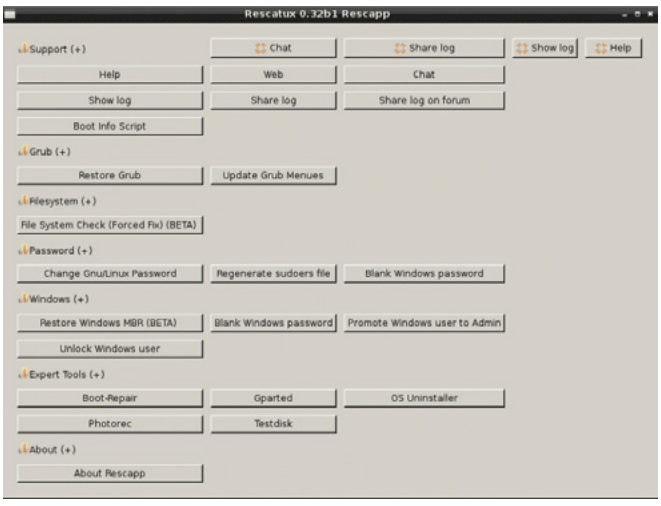


Imagen 1. Panel de control de Rescatux.

NOTA

Antes de utilizar herramientas de reparación, se recomienda realizar una copia de seguridad de los datos importantes. Algunas operaciones, como la restauración del MBR o la reasignación de particiones, pueden provocar pérdidas de datos si no se ejecutan correctamente.

PREGUNTA

¿Qué herramienta basada en Debian permite restaurar el MBR y reparar el cargador de arranque de un sistema Linux?

3.4.2. Utilidades para la recuperación de ficheros

La pérdida accidental de datos puede deberse a diversos factores: eliminación no intencionada, corrupción del sistema de archivos, fallos de hardware, ataques de malware, entre otros. La recuperación de datos es posible si el área del disco donde se almacenaba el archivo aún no ha sido sobrescrita.

Fundamento técnico: *file carving*

El *file carving* es una técnica forense que consiste en analizar sectores del disco en busca de firmas o patrones que identifiquen archivos eliminados, sin depender de la estructura lógica del sistema de archivos. Esta técnica es especialmente útil cuando dicha estructura está **dañada** o **ausente**.

Cuando se elimina un archivo, el sistema operativo no lo borra físicamente del disco duro; en su lugar, marca el espacio que ocupaba como disponible para su reutilización. Esto significa que, en algunos casos, aún es posible recuperar el contenido utilizando técnicas como el "file carving" o herramientas forenses de archivos, siempre que los datos no hayan sido sobrescritos o dañados.

Una de las utilidades más populares para la recuperación de archivos en Windows es **Recuva**. Esta herramienta cuenta con una versión portátil, lo que significa que no es necesario instalarla en el sistema para utilizarla.

Herramienta destacada para Windows: Recuva

Recuva es una utilidad de recuperación de datos desarrollada por Piriform, compatible con sistemas Windows. Su versión portátil permite ejecutarse desde un medio externo sin necesidad de instalación, lo que evita sobrescribir datos en el disco afectado.

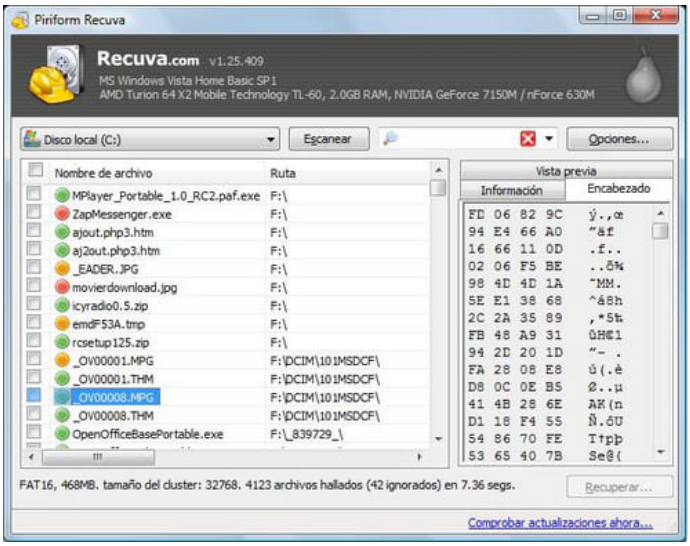


Imagen 2. Recuva Portable.

Características de Recuva

1. Análisis profundo de discos y unidades extraíbles.
2. Clasificación de archivos recuperables por colores:
   - **Verde:** archivo totalmente recuperable.
   - **Amarillo:** recuperación parcial; el archivo puede estar dañado.
   - **Rojo:** archivo irrecuperable.
3. Interfaz intuitiva, adecuada para entornos técnicos y no técnicos.

NOTA

Cuando se detecte una pérdida de datos, evitar el uso del disco afectado y operar desde otro sistema o desde un entorno en vivo para maximizar la probabilidad de recuperación.

PREGUNTA

¿Cuál es una herramienta de recuperación de datos portátil y gratuita para sistemas Windows?

3.4.3. Multiherramienta para Mac OS-X Onyx

Para llevar a cabo el mantenimiento de un equipo con Mac OS-X, una de las herramientas más utilizadas es OnyX. Se trata de una multiherramienta que, en primer lugar, verifica que la estructura del sistema de archivos sea correcta y, a continuación, permite administrar el sistema de manera eficiente.

OnyX ofrece una amplia gama de funciones, incluyendo la posibilidad de reorganizar los índices del sistema, eliminar archivos temporales y cachés, configurar parámetros ocultos del Finder, Dock, Safari y otras aplicaciones, y muchas otras tareas de administración del equipo. La instalación de OnyX se realiza mediante Copia Directa, lo que facilita su implementación en el sistema.

Funciones principales de OnyX:

1. Verificación de la estructura del sistema de archivos.
2. Eliminación de archivos temporales, logs, cachés y registros del sistema.
3. Reindexación de Spotlight y reconstrucción de bases de datos del sistema.
4. Configuración de parámetros ocultos del sistema y personalización de interfaz (Finder, Dock, Safari, etc.).
5. Automatización de tareas de mantenimiento preventivo.

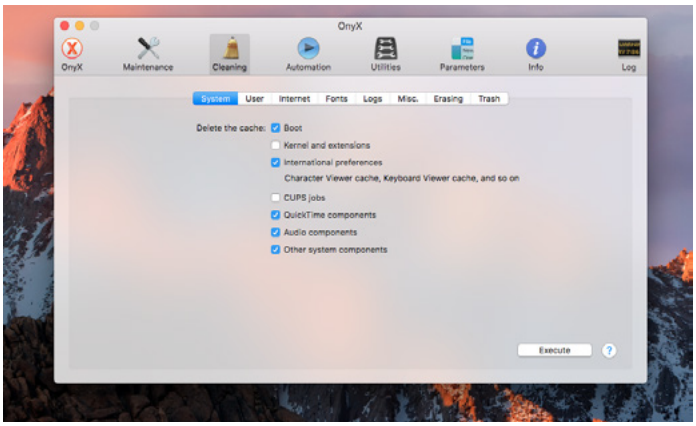


Imagen 3. Onyx.

NOTA

Aunque OnyX es una herramienta segura, cualquier cambio avanzado en los parámetros del sistema debe ejecutarse con conocimiento previo. Un uso incorrecto puede afectar la estabilidad del sistema operativo..

3.5. Malware y antivirus

El término *malware* (del inglés *malicious software*) hace referencia a cualquier programa o código informático diseñado para infiltrarse, dañar o realizar actividades no autorizadas en un sistema informático sin el consentimiento del usuario. A diferencia de la noción tradicional centrada exclusivamente en los virus, el malware actual abarca una amplia variedad de amenazas, muchas de ellas sofisticadas y difíciles de detectar.

La defensa frente al malware es una responsabilidad fundamental del técnico en sistemas informáticos, especialmente en entornos donde los usuarios finales tienen conocimientos limitados en ciberseguridad.

3.5.1. Malware

A continuación, se describirán los tipos más conocidos de malware, cómo funciona cada uno y el daño que pueden causar, con la intención de poder identificarlos y detenerlos de manera más sencilla.

NOTA

Aunque OnyX es una herramienta segura, cualquier cambio avanzado en los parámetros del sistema debe ejecutarse con conocimiento previo. Un uso incorrecto puede afectar la estabilidad del sistema operativo..

Los virus

Los **virus informáticos** son programas diseñados para infectar archivos legítimos y replicarse automáticamente. Su objetivo habitual es el sabotaje del sistema, provocando daños que pueden ir desde el deterioro del rendimiento hasta la inutilización completa del equipo.

- **Modo de acción habitual:** Infección del cargador de arranque o de archivos del sistema.

Ejemplos históricos:

- **ILOVEYOU (2000):** se distribuía por correo electrónico como archivo adjunto de texto.
- **Melissa (1999):** se propagaba a través de documentos Word e infectó redes corporativas de alto nivel.

NOTA

No abrir correos ni enlaces de remitentes desconocidos. Ante sospechas, escalar al **personal de ciberseguridad.**

Los troyanos (trojan horses)

Los **troyanos** no dañan directamente el sistema, pero permiten el acceso no autorizado a través de una puerta trasera. Se utilizan habitualmente para robar credenciales, espiar al usuario o instalar otros tipos de malware.

- **Funcionamiento:** se camuflan como software legítimo y actúan de forma silenciosa.
- **Uso frecuente:** instalar spyware o ransomware, formar parte de botnets, controlar el sistema remotamente.

Imagen 4. Partes de un troyano.

Los troyanos suelen ser personalizables, lo que significa que el atacante puede indicar qué acciones específicas debe llevar a cabo el troyano y controlarlo desde su ubicación de origen.

NOTA

La detección de un troyano debe acompañarse siempre de una búsqueda exhaustiva de malware asociado (por ejemplo, spyware o rootkits).

Botnets

Una botnet es una red de robots informáticos llamados bots que son autónomos y siguen una serie de órdenes de acuerdo con el administrador de la botnet. Actualmente, los bots se utilizan para varios fines maliciosos, como:

- **Realizar ataques DDoS:** para denegar el servicio, ya que, al encontrarse dispersos, los ataques hacia un servidor o red son difíciles de rastrear, lo que provoca la caída del servicio atacado.
- **Enviar spam:** posiblemente el mayor uso para estos bots, ya que se pueden enviar multitud de mensajes desde diferentes ubicaciones y, nuevamente, resulta muy difícil filtrar de dónde provienen los ataques.

Si sospechas que tu equipo ha sido infectado y se ha convertido en un bot, debes tener en cuenta lo siguiente:

- El equipo no se actualiza.
- La conexión a Internet del equipo es lenta, pero si se realiza una prueba de velocidad, todo indica que funciona correctamente.
- El equipo se ralentiza al realizar cualquier tarea.
- El equipo se encuentra trabajando cuando no debería.
- Si se abre el administrador de tareas y hay ejecuciones desconocidas.
- Si hay picos de tráfico de Internet en el monitoreo de la red.

Si detectas alguna de estas características en un equipo, deberías utilizar algún servicio antibotnet que monitoree la conexión a Internet. También puedes instalar plugins en los navegadores que te alerten en caso de que el equipo esté infectado.

NOTA

Un bot no es malware en sí mismo, pero puede formar parte de una estructura maliciosa. Su función original es la automatización.

keyloggers

Los **keyloggers** registran cada pulsación del teclado para capturar datos confidenciales como contraseñas o información bancaria.

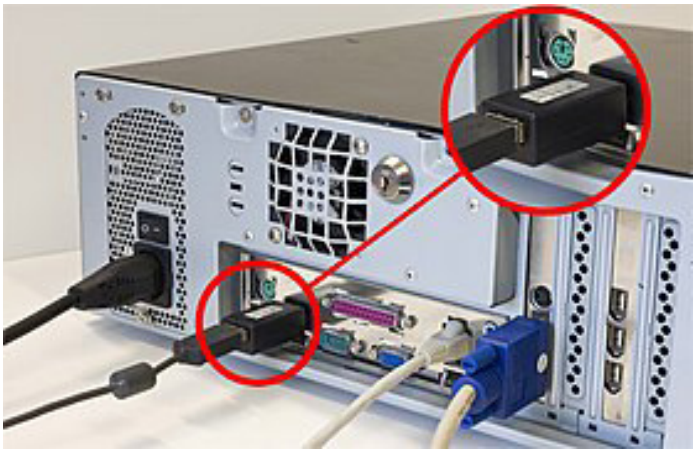


Imagen 5. Keylogger Hardware. Fuente: commons.wikimedia.org

Tipos:

- **Software:** detectables por la mayoría de los antivirus.
- **Hardware:** más difíciles de detectar, algunos transmiten la información por vía inalámbrica.

El spyware

El **spyware** recopila información del usuario sin su consentimiento. Afecta negativamente al rendimiento del sistema y consume recursos de red..

Síntomas frecuentes:

- Cambia la página de inicio.
- Cambia el motor de búsqueda.
- Aparecen múltiples anuncios en el navegador.
- Aparecen nuevos botones en la barra de herramientas.
- Hay procesos sospechosos y desconocidos en el administrador de tareas, y aunque se eliminen, vuelven a aparecer.
- Se desinstala un programa que creemos malicioso y, por sí solo, se vuelve a instalar.

NOTA

Para luchar contra un sypware se usan los antispywares, pero no están muy recomendados debido a que solo detecta elimina el programa, pero detectarlo es tarea nuestra.

Los *spywares* van asociados normalmente de manera muy estrecha a los troyanos, pues estos últimos muchas veces se introducen para poder instalar el *spyware*.

Esto quiere decir que, si detectamos un troyano, revisemos el equipo en busca de un *spyware*.

El adware

El **adware** muestra publicidad no solicitada en el sistema del usuario. Aunque no suele causar daños graves, puede ser intrusivo.

> **Consideración legal:** Algunos programas gratuitos integran adware con consentimiento explícito del usuario.

Imagen 6. El adware

Las cookies

Las **cookies** son pequeños archivos utilizados legítimamente por los navegadores para almacenar preferencias. Sin embargo, algunas se usan con fines de rastreo abusivo o comercialización de datos.

NOTA

Revisar y gestionar cookies en la configuración del navegador. Usar navegadores con políticas de privacidad reforzadas.

Imagen 7. Cookies

Backdoors

Las **puertas traseras** son accesos ocultos que permiten al atacante ingresar al sistema sin autentificación. Suelen ser instaladas por troyanos u otros programas maliciosos.

> **Uso común:** acceso remoto persistente y silencioso, especialmente en ataques avanzados (APT).

Ransomware

El ransomware es, posiblemente, el tipo de malware **más dañino**. Se utiliza para cifrar todo el sistema y quitar el acceso a los usuarios, ya que no poseen la clave correspondiente.

Este cifrado se suele llevar a cabo para exigir una recompensa económica, que suele ser de un precio muy elevado.

Una solución para este tipo de malware es llevar un sistema de copias de seguridad actualizado y poder restaurar el sistema desde cero.

NOTA

En el año 2019, concretamente en noviembre, Everis y la Cadena Ser sufrieron un ataque con un ransomware llamado Ryuk y sus consecuencias fueron devastadoras.

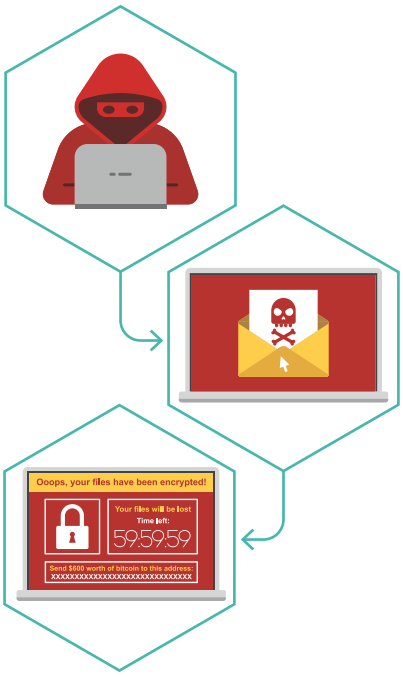


Imagen 8. Encriptación mediante Ransomware

El web bug, tracking pixel, tracking bug, pixel tag o web beacons

Los cinco realizan prácticamente la misma función, que es la de monitorizar la actividad de red de un usuario, por eso son considerados malware.

Son fáciles de detectar, ya que suelen descargar imágenes de un tamaño muy pequeño pero que están presentes en sitios web.

Los exploit

Los exploit son un tipo de malware que se basa en explotar los fallos o vulnerabilidades de seguridad de un sistema.

Estos exploit también se usan en muchos campos de la ciberseguridad para detectar fallas y poder corregirlas.

NOTA

Para poder protegernos correctamente de los *exploit*, una buena práctica es tener instaladas todas las actualizaciones y parches de seguridad disponibles para el sistema.

Los rootkits

Los rootkits son un tipo de herramienta que se usa para obtener el control del sistema con la intención de encontrar ciertas vulnerabilidades o crear exploits con los que acceder al sistema de manera completa.

Leapfrogs o ranas

Como su propio nombre indica, es un tipo de malware que salta de un dispositivo a otro replicándose en cada uno de ellos. Para poder pasar de un equipo a otro, se basan en el descubrimiento de contraseñas y los mensajes de correo electrónico.

Bulos, jokes o hoaxes

Este tipo de programa malicioso suele tener la intención de engañar al usuario que usa el equipo para que realice alguna acción sobre el equipo, generalmente instalar algún tipo de software con la intención de recopilar información o introducirse en el equipo.

Basura, escoria o *scumware*

El scumware es un tipo de malware que no permite ser desinstalado de manera normal, ya que se reinstala solo otra vez. Hay que destacar que su intención no es tan dañina, ya que suele afectar solo en temas de publicidad vía navegador web.

NOTA

Muchos de los *softwares* libres que se instalan de páginas no fiables suelen llevar este tipo de *malware* con ellos.

El spam

El **spam** es el envío masivo de correos no solicitados, habitualmente con fines comerciales o engañosos. Aunque no es estrictamente malware, puede ser vehículo de otros tipos.

Imagen 9. Carpeta de Spam que tienen por defecto algunos clientes de correo electrónico.

IMPORTANTE

Es esencial utilizar un antivirus y mantenerlo actualizado para protegerse de los distintos tipos de malware, incluidos los mencionados anteriormente.

PREGUNTA

¿Cuál es la función principal de los web bugs o tracking pixels?

¿Qué es un rootkit y cuál es su propósito?

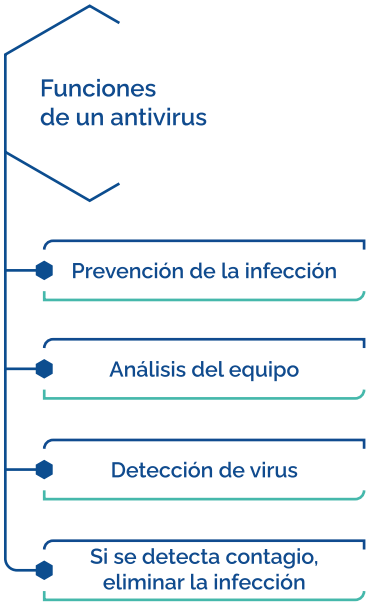
¿Cuál es la diferencia entre un hoax y un scumware?

3.5.2. Los antivirus o antimalware

Los **antivirus** o **soluciones antimalware** son herramientas fundamentales para proteger los sistemas frente a programas maliciosos. Su uso es especialmente importante en entornos donde los usuarios tienen escasos conocimientos técnicos y, por tanto, son más vulnerables a amenazas informáticas.

Un buen antivirus no solo detecta y elimina malware, sino que también previene infecciones y analiza el comportamiento del sistema para detectar amenazas emergentes.

- **Que consuma la mínima memoria posible.** Si el antivirus consume una elevada carga de memoria nos ralentizará el equipo, por lo que su efecto será contraproducente.
- **Que consuma el mínimo de CPU.** Nos sucede lo mismo que con la memoria.
- **Que se actualice de manera frecuente.** Con esto no nos referimos a una subida de versión, que también, sino a actualizaciones en sus bases de datos frente a posibles ataques o infecciones.



NOTA

Dependiendo del nivel de seguridad que necesitemos elegiremos un tipo u otro de antivirus.

3.5.3. Funcionamiento de un antivirus

Los antivirus modernos pueden funcionar generalmente de dos formas:

- **Funcionamiento pasivo (detección por firmas):** la forma más clásica de actuación de un antivirus. Llamada también *técnica de scanning* se basa en que el antivirus tiene una base de datos donde va recopilando información acerca de los distintos tipos de *malware* y sus maneras de actuar para poder detectarlos a tiempo.

  Esta base de datos se actualiza de manera constante y todos los antivirus mandan información a través de internet para mantenerse al día.

  Cada vez que el antivirus identifique que algún *software* está siguiendo un patrón similar al de un virus intentará eliminarlo y en caso de no poder pondrá todo lo infectado en cuarentena y avisará al usuario final.

  El mayor problema de este tipo de funcionamiento es que no se evita el contagio previo a la detección del virus.

- **De forma activa o técnicas heurísticas**: es la forma más moderna de trabajar de un antivirus. Se basa en el intento de prevenir la entrada de *malwares* en el equipo monitorizando los procesos del sistema constantemente con la intención de encontrar movimientos sospechosos.

- El mayor problema ahora mismo de este funcionamiento es que esa monitorización tiene que llevarse a cabo lo más eficientemente posible para que no confunda otros procesos o archivos con *malware* cuando no lo son.

3.6. Utilidades

Vamos a describir unas ciertas utilidades que está bien que sepa un administrador de sistemas con la intención de facilitar el día a día en su cometido.

3.6.1. Monitorización del sistema Linux mediante comandos

El uso del terminal de comando por parte de un administrador de sistemas es debido a que da una información muy detallada de manera muy rápida.

Los principales comandos para monitorizar un sistema Linux son:

- **uptime:** presenta la siguiente información:
  - Hora del sistema y el tiempo que lleva encendido.
  - Número de usuarios conectados.
  - Valor medio de la carga en:
    1. El último minuto.
    2. Los últimos 5 minutos.
    3. Los últimos 15 minutos.

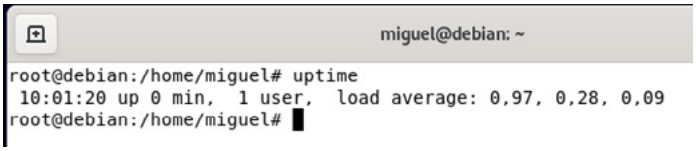


Imagen 10. Comando

- **time programa:** nos permite ver la distribución del tiempo que ha tardado en ejecutar un programa concreto nuestro procesador tanto en modo usuario como en modo supervisor.
- ***top:*** podemos ver con este comando todos los procesos que hay en ejecución y su consumo de memoria en tiempo real.

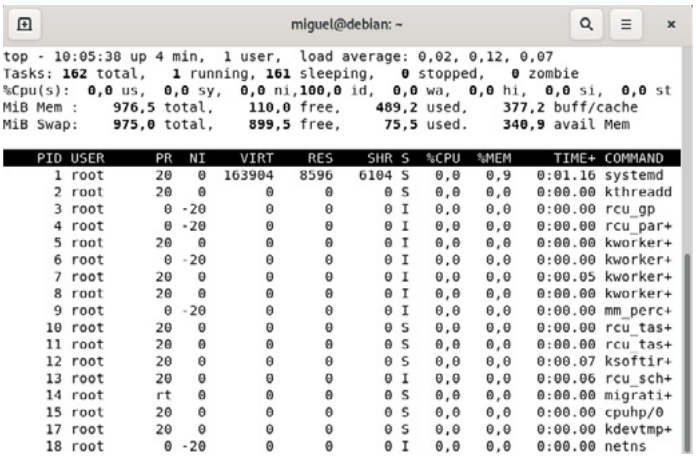


Imagen 11. Salida del comando top.

- **ps**: nos muestra los procesos del sistema que han sido lanzados por el usuario que los invoca.

```
root@debian:/home/miguel# ps
    PID TTY          TIME CMD
   1841 pts/0    00:00:00 su
   1842 pts/0    00:00:00 bash
   2135 pts/0    00:00:00 top
   2177 pts/0    00:00:00 ps
```

Imagen 12. ps invocado por el usuario root.

Además de los comandos mencionados los sistemas Linux tienen implementados en su línea de comandos un conjunto de herramientas llamadas *Sysstat*. Estas herramientas se usan para monitorizar el análisis del rendimiento del equipo.

Algunas de las herramientas que presenta este conjunto son:

- **iostat**. Muestra las estadísticas de entrada/salida de los dispositivos, particiones y sistemas de ficheros en la red.

```
root@debian:/home/miguel# iostat
Linux 5.10.0-11-amd64 (debian)  17/02/22        _x86_64_        (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           2,96    0,05    1,05    0,23    0,00   95,71

Device             tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_w
rtn    kB_dscd
sda              40,94      1181,15       784,67         0,00    1258464     836
036          0
sr0               0,03         0,07         0,00         0,00         72
  0          0
```

Imagen 13. Comando iostat.

- **mpstat**. Nos genera las estadísticas del procesador.

```
root@debian:/home/miguel# mpstat
Linux 5.10.0-11-amd64 (debian)  17/02/22        _x86_64_        (1 CPU)

10:23:02     CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest
  %gnice   %idle
10:23:02     all    2,35    0,04    0,81    0,18    0,00    0,02    0,00    0,00
    0,00   96,59
```

Imagen 14. Comando mpstat.

- **pidstat**. Nos informa de los procesos activos del sistema.

```
miguel@debian: ~
root@debian:/home/miguel# pidstat
Linux 5.10.0-11-amd64 (debian)  17/02/22        _x86_64_        (1 CPU)

10:35:04      UID       PID    %usr %system  %guest   %wait    %CPU   CPU  Comma
nd
10:35:04        0         1    0,03    0,06    0,00    0,04    0,09     0  syste
md
10:35:04        0        12    0,00    0,00    0,00    0,02    0,00     0  ksoft
irqd/0
10:35:04        0        13    0,00    0,00    0,00    0,05    0,00     0  rcu_s
ched
10:35:04        0        14    0,00    0,00    0,00    0,00    0,00     0  migra
tion/0
10:35:04        0        23    0,00    0,01    0,00    0,00    0,01     0  kcomp
actd0
10:35:04        0        25    0,00    0,00    0,00    0,00    0,00     0  khuge
paged
10:35:04        0        48    0,00    0,01    0,00    0,01    0,01     0  kwork
er/0:1H-kblockd
10:35:04        0        49    0,00    0,04    0,00    0,15    0,04     0  kswap
d0
10:35:04        0       105    0,00    0,00    0,00    0,00    0,00     0  scsi_
eh_0
10:35:04        0       106    0,00    0,00    0,00    0,00    0,00     0  scsi_
```

Imagen 15. Comando pidstat.

- **sar**. Monitoriza y recoge información acerca de todas las actividades del sistema que tengan relación la CPU, la memoria, las interrupciones o llamadas al sistema, las interfaces, y las tablas del *kernel*.

Todas estas utilidades nos ofrecen información acerca del sistema como puede ser:

- Las tasas de entrada/salida y transferencias
- La carga de la CPU
- EL uso de memoria
- La paginación y su carga de memoria y fallos
- La velocidad de generación de nuevos procesos
- Lel número de interrupciones
- La cola de ejecución
- La carga del sistema

Normalmente toda esta información se agrupa en 5 grupos: memoria, red, procesadores, CPU y E/S.

3.6.2. Gestión de recursos (memoria, disco, etc.) mediante comando en Linux

Ahora que ya hemos hablado acerca de cómo los comandos como *top* muestran de manera seguida los procesos que se ejecutan en el equipo y su consumo de recursos, tenemos que hablar de tres comandos que también nos van a facilitar la administración del sistema.

Estos son: *df, du y free.*

df [opciones] [directorio]

Este comando viene de *disk free* y se puede ejecutar con o sin opciones adicionales.

Si lo mostramos solo, sin ninguna opción, nos mostrará la información referente al espacio que tenemos libre en el disco, el que tenemos usado y así, con todos los discos que se encuentran montados en el sistema.

Si especificamos un directorio. Nos mostrará la información de espacio usado justo en la ruta donde se encuentra el directorio.

La opción *-h* hace que se nos muestre el resultado en formato legible para el ser humano, no es que el otro no lo sea, es que este es más sencillo porque usa las medidas convencionales.

```
root@debian:~# df -h
S.ficheros     Tamaño Usados  Disp Uso% Montado en
udev              466M      0  466M   0% /dev
tmpfs              98M   1,1M   97M   2% /run
/dev/sda1         6,9G   5,0G  1,6G  77% /
tmpfs             489M      0  489M   0% /dev/shm
tmpfs             5,0M   4,0K  5,0M   1% /run/lock
tmpfs              98M   116K   98M   1% /run/user/1000
root@debian:~# df
S.ficheros     bloques de 1K  Usados Disponibles Uso% Montado en
udev                  476984       0      476984   0% /dev
tmpfs                  99992    1120       98872   2% /run
/dev/sda1            7173040 5166172     1621128  77% /
tmpfs                 499944       0      499944   0% /dev/shm
tmpfs                   5120       4        5116   1% /run/lock
tmpfs                  99988     116       99872   1% /run/user/1000
root@debian:~#
```

Imagen 16. df -h y df.

du [opciones][path]

El comando *du* nos muestra el espacio total del disco que ocupan los ficheros y subdirectorios de la ruta en la que se lance. Esto lo hace siempre que se lance sin opciones, si por el contrario especificamos una ruta, nos lo mostrará de esta ruta.

Este comando también posee la opción *-h*.

```
root@debian:~# du
8       ./.cache/dconf
4       ./.cache/appstream
16      ./.cache
8       ./.dbus/session-bus
12      ./.dbus
4       ./.config/procps
8       ./.config
8       ./.synaptic
60      .
root@debian:~# du -h
8,0K    ./.cache/dconf
4,0K    ./.cache/appstream
16K     ./.cache
8,0K    ./.dbus/session-bus
12K     ./.dbus
4,0K    ./.config/procps
8,0K    ./.config
8,0K    ./.synaptic
60K     .
```

Imagen 17. du y du -h.

free [opciones]

El comando *free* muestra la información acerca de la RAM y el espacio *swap* usado en el momento en que se lanza el comando. Nos permitirá ver el uso de cada una de las dos categorías, así como el total. Tiene las siguientes opciones:

- *-b:* la salida se muestra en bytes.
- *-k:* la salida se muestra en kilobytes.
- *-m*: la salida se muestra en megabytes.

```
root@debian:~# free
               total        used        free      shared  buff/cache   available
Mem:          999888      461956      135912        6548      402020      387976
Swap:         998396      171720      826676
root@debian:~# free -b
               total        used        free      shared  buff/cache   available
Mem:      1023885312   473042944   139173888     6705152   411668480   397287424
Swap:     1022357504   175841280   846516224
root@debian:~# free -m
               total        used        free      shared  buff/cache   available
Mem:             976         451         132           6         392         378
Swap:            974         167         807
root@debian:~# free -k
               total        used        free      shared  buff/cache   available
Mem:          999888      461956      135912        6548      402020      387976
Swap:         998396      171720      826676
```

Imagen 18. Comando free sin opciones y con sus tres opciones.

NOTA

Si nos fijamos en la imagen anterior podremos ver que, si usamos el comando free sin opciones, por defecto muestra la información en kilobytes.

3.6.3. El monitor del sistema en Linux

El 'Monitor del sistema' de Linux, al igual que con Windows nos dará toda la información que hemos ido mostrando antes, aunque algo más básica, mediante una interfaz gráfica.

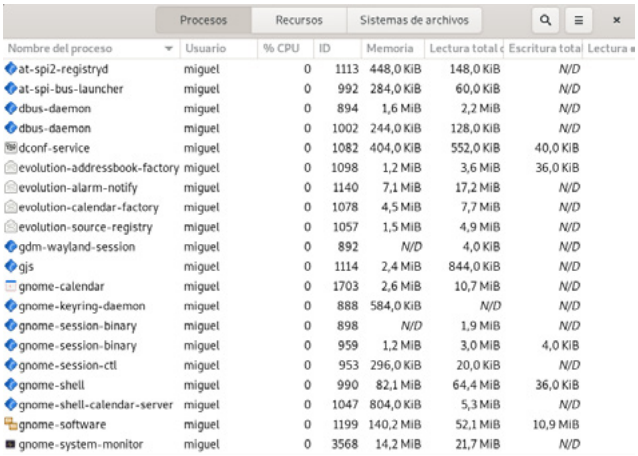


Imagen 19. Monitor del sistema en Debian.

Como se puede observar en la imagen anterior, tiene tres pestañas:

Procesos **Recursos** **Sistemas de archivos**

La pestaña con la que se abre, que es la de 'Procesos' es la que se ve en la imagen de más arriba y nos muestra el ID del proceso y los recursos consumidos.

Si hacemos clic derecho sobre algún proceso podemos:

- Ver sus propiedades
- Ver sus mapas de memoria
- Ver que archivos tiene abierto el proceso
- Modificar la prioridad
- Otras opciones:
  - Detener
  - Continuar
  - Finalizar
  - Matar

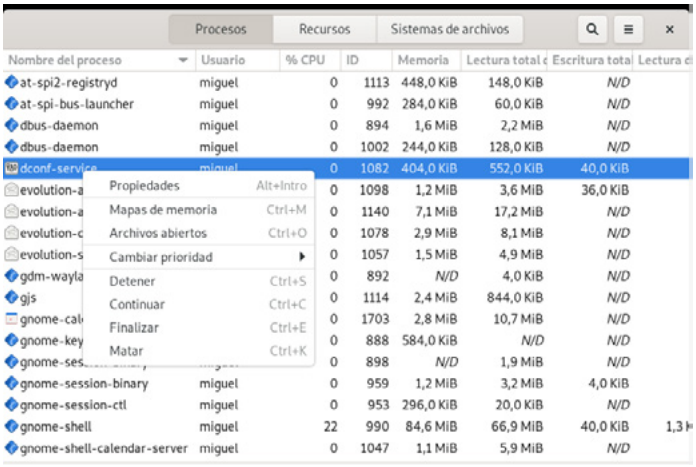


Imagen 20. Opciones sobre procesos.

Si nos dirigimos a la pestaña de 'Recursos', se nos mostrará un panel interactivo sobre el uso de la CPU, la memoria y la red del equipo.

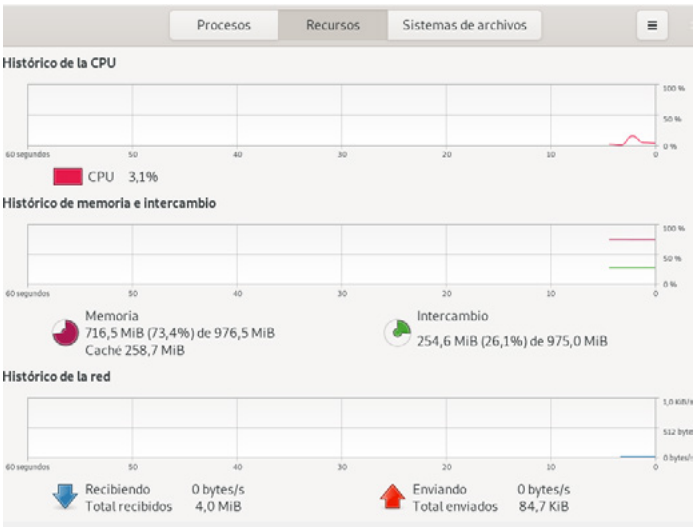


Imagen 21. Pestaña 'Recursos'.

Esta pestaña es importante ya que la información que muestran los gráficos nos mostrará el estado en que se encuentra nuestro sistema en cuanto a recursos principales nos referimos.

Por último, la pestaña 'Sistemas de archivos', nos va a mostrar las particiones del sistema, sus puntos de montaje, los sistemas de archivos que contiene cada una de ellas, su tamaño total, el espacio que queda libre, el espacio que queda disponible y el espacio usado.

Además de todo esto nos mostrará una barra con los porcentajes.

Imagen 22. Pestaña 'Sistemas de archivos'.

www.universae.com