Engineer

Sistema operativo

software que gestiona hardware para los demás programas

Qué es

Un sistema operativo es el software que se inicia primero en un ordenador y permanece en ejecución por debajo de todo lo demás, repartiendo procesador, memoria, almacenamiento y red entre programas y dando a todos las mismas formas de leer archivos, utilizar dispositivos y acceder a la red. Suele abreviarse como SO, u OS en inglés. Windows, macOS, Android, iOS y las distribuciones Linux como Debian y Ubuntu son sistemas operativos.

El glosario del NIST recoge dos descripciones que encajan entre sí. Una lo llama la aplicación de control principal que hace funcionar el ordenador: «el primer programa que se carga al encender el ordenador», cuyo componente principal, el núcleo, permanece en memoria en todo momento y que «establece los estándares para todos los programas de aplicación», como el servidor web. La otra, procedente de la guía del NIST sobre seguridad de tecnología operativa, lo llama «una colección integrada de rutinas de servicio para supervisar la secuencia de ejecución de los programas», que puede realizar control de entrada y salida, planificación de recursos y gestión de datos y que proporciona a los programas de aplicación las órdenes fundamentales para controlar el ordenador.

El núcleo es la parte central, pero un sistema operativo es más que su núcleo. A su alrededor están las bibliotecas de sistema a las que llaman los programas, el intérprete de órdenes y las herramientas habituales de línea de comandos, un supervisor que inicia los servicios durante el arranque y los reinicia, un gestor de paquetes que instala y actualiza software y un amplio conjunto de configuración predeterminada que decide qué está activado, dónde van los registros y quién puede iniciar sesión. Una distribución Linux reúne todo eso alrededor del núcleo Linux, por lo que dos sistemas con el mismo núcleo pueden comportarse de formas muy distintas.

La mayor parte de lo que un sistema operativo ofrece a un programa es una breve lista de abstracciones que representan al hardware. POSIX, el estándar en el que se basan los sistemas de tipo Unix, las define con precisión. Un proceso es un espacio de direcciones donde se ejecutan uno o más hilos, junto con los recursos del sistema que esos hilos necesitan. Un archivo es un objeto en el que se puede escribir, del que se puede leer o ambas cosas, con permisos de acceso y un tipo: archivo normal, directorio, enlace simbólico, FIFO, socket o un archivo especial de caracteres o de bloques que representa un dispositivo. Un archivo especial de bloques da acceso a un dispositivo de una forma que oculta sus características de hardware. El propio dispositivo se define como «un periférico de ordenador o un objeto que se presenta como tal a la aplicación». El propósito declarado de POSIX es hacer portables las aplicaciones a nivel de código fuente, de modo que un programa escrito para esas interfaces pueda compilarse y ejecutarse en otro sistema conforme al estándar.

La memoria recibe el mismo tratamiento. La documentación del núcleo Linux parte de que la memoria física es un recurso limitado y describe la memoria virtual como la capa que abstrae sus detalles para el software de aplicación. Cada programa ve un espacio de direcciones propio, y el sistema operativo decide en cada momento qué partes de qué programas se mantienen en la memoria real y cuáles se comprimen, se escriben en disco o se descartan para volver a leerlas más tarde.

Utilice el concepto en una decisión

Aplique «¿Qué es un sistema operativo?» a una decisión real. Podemos ayudarle a relacionar el concepto con el sistema que está construyendo o comprando. El cuidado diario de servidores, correo, cuentas y actualizaciones para que los servicios de los que depende todo el mundo sigan funcionando sin sorpresas.

Explore nuestros servicios Póngase en contacto →

Compruebe los supuestos detrás del término

Utilice la explicación y los trabajos enlazados para formular preguntas sobre su proyecto. Podemos examinar los supuestos con usted antes de elegir una herramienta. 10 servidores PostgreSQL y MS SQL en Ubuntu — 10 servidores de bases de datos desplegados y mantenidos

10 servidores PostgreSQL y MS SQL en Ubuntu Póngase en contacto →

Por qué importa

Elegir un sistema operativo resuelve muchas cosas de golpe: qué software se ejecutará, qué herramientas conoce ya quien lo opera, quién suministra las correcciones de seguridad y durante cuánto tiempo y cuánto costará un contrato de soporte, si lo hay. Cambiarlo después significa trasladar cada programa y cada hábito construido sobre él, de modo que la elección suele durar más que el hardware para el que se tomó inicialmente.

También es la principal frontera de confianza dentro de una máquina. Usuarios, grupos, permisos de archivos y separación entre procesos son funciones del sistema operativo, y deciden hasta dónde puede llegar un fallo en un programa. A una máquina donde todos los servicios se ejecutan con plenos permisos solo le queda una barrera: el perímetro de la red.

Sus valores predeterminados son decisiones que tomó otra persona. Una instalación nueva trae servicios activados, archivos en ubicaciones acordadas y ajustes ya elegidos. La mayoría son sensatos; unos pocos no encajarán con el trabajo y nadie indica cuáles. Operar bien una máquina incluye saber qué valores predeterminados se han aceptado.

Y decide hasta dónde viaja el código. Un programa que respeta las interfaces portables pasa a otro sistema con una recompilación. Un programa o script que depende de las extensiones de un sistema solo llega hasta donde lleguen esas extensiones.

Cómo se aplica en la práctica

En un portátil, los informes de diagnóstico de una aplicación macOS registraron 111 GB de páginas comprimidas en una máquina de 36 GB el 2026-09-01. Una muestra separada de la huella de memoria registró un crecimiento de unos 41 MB por segundo durante un bucle de reintentos de conexión que emitía eventos del núcleo de la aplicación; son observaciones separadas, no una medición continua única. Se cambió la ruta de eventos para limitar los búferes, filtrar las suscripciones por tipo de evento y fijar un presupuesto de frecuencia para el registro. Una comparación de ejecuciones completas de pruebas registró que las líneas de registro del núcleo de la aplicación bajaron de 610.996 a 1.411. Las pruebas protegen el diseño acotado, pero esa comparación no es un benchmark de producción ni una cifra de memoria medida después de la corrección. Otros fallos por tiempo de espera agotado durante el cierre necesitaron su propia corrección.

En el host de la empresa, un rol de automatización llevaba meses escribiendo sus ajustes SSH en cada ejecución cuando un informe de acceso leyó la configuración efectiva resuelta por el propio demonio y encontró que dos ajustes no coincidían. El relato del refuerzo de SSH con validación en tres etapas explica la causa: el sistema operativo distribuye sus propios valores predeterminados sin comentar, por delante del lugar donde queda el archivo de configuración adicional, y para

esos dos ajustes prevalece la primera aparición. La solución fue un prefijo de nombre de archivo que se ordena antes que el del proveedor, y ahora se comprueban 24 directivas contra lo que devuelve el demonio.

El mismo host se encontró con otra decisión del sistema al configurar su servicio de bloqueo de intrusos. La fuente de registros se establece en detect porque el sistema operativo no proporciona un registro de autenticación tradicional, y una elección incorrecta falla en silencio al vigilar un archivo que nunca aparece.

Dónde surgen los problemas

Un script arrastra las costumbres del sistema donde se escribió. Un script de shell escrito en un Mac y ejecutado en un servidor Linux, o al revés, encuentra los mismos nombres de órdenes con opciones diferentes. El `sed` de POSIX acepta `-E` y `-n`; editar un archivo directamente con `-i` es una extensión, y las dos versiones habituales la interpretan de manera distinta:

```
sed -i 's/old/new/' notes.txt     # GNU sed, on most Linux systems
sed -i '' 's/old/new/' notes.txt  # BSD sed, on macOS
```

Cada forma falla en el otro sistema. Limítese a lo que define POSIX cuando sea suficiente, ejecute el script en el sistema que lo ejecutará en producción e instale la misma herramienta en ambos cuando merezca la pena utilizar una extensión.

Dos máquinas con el mismo sistema operativo difieren. «Debian» nombra una familia de versiones, y dos hosts descritos como sistemas Debian pueden estar separados por años en sus versiones de distribución, núcleo y paquetes, tener paquetes distintos instalados y cambios locales diferentes. Compruebe qué sistema ejecuta realmente cada máquina antes de compararlas:

```
cat /etc/os-release
uname -srm
```

La primera orden muestra el nombre y la versión de la distribución; la segunda, el nombre y la versión del núcleo y el tipo de procesador de la máquina. Construya cada host a partir de la misma descripción escrita y mantenga la versión en el inventario.

Los servicios se ejecutan como administrador. Un servicio que se ejecuta como root en un sistema de tipo Unix puede leer cualquier archivo y cambiar cualquier ajuste, así que cualquier fallo en él es un fallo en toda la máquina. Dé a cada servicio su propio usuario sin privilegios, con permiso de escritura solo donde escriba, y conceda permisos adicionales uno a uno cuando el servicio demuestre necesitarlos.

Una versión sigue en uso después de terminar su soporte de seguridad. Todas las versiones dejan de recibir correcciones según un calendario publicado. Por ejemplo, el equipo de seguridad de Debian da soporte a cada distribución estable durante tres años desde el día de su publicación. Un host que permanece en una versión antigua sigue funcionando y precisamente por eso se acaba olvidando. Registre la fecha de fin de soporte de cada host junto a sus otras fechas de renovación y planifique la actualización con suficiente antelación.

Lecturas adicionales

- Sistema operativo: glosario NIST CSRC — ambas definiciones, desde la aplicación de control principal hasta la colección de rutinas de servicio.
- POSIX.1-2024, definiciones básicas: definiciones — formulación exacta de proceso, archivo, dispositivo y los demás conceptos que componen un sistema operativo.
- POSIX.1-2024: sed — opciones que acepta todo `sed` conforme al estándar, como medida de lo que puede utilizar un script portable.
- Documentación del núcleo Linux: introducción a los conceptos — memoria virtual, caché de páginas y recuperación de memoria, tal como los describe el propio núcleo.
- Preguntas frecuentes sobre seguridad de Debian — cómo gestiona Debian las correcciones de seguridad y cuánto tiempo recibe soporte una versión estable.

Glosario

Elija según sus restricciones reales

Describa la ruta de despliegue, los riesgos operativos y quiénes se encargan de las guardias. Podemos definir el alcance de un cambio con comprobaciones, una vía de recuperación y responsabilidades claras.

Vea cuánto cuesta Póngase en contacto →

Haga práctica la terminología

Pregunte cómo se aplica «¿Qué es un sistema operativo?» a su organización. Basta un ejemplo concreto de su sistema actual para iniciar la conversación.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/glossary/operating-system/