

Introducción a la Administración de Sistemas Unix/Linux

Miguel Ortuño
Escuela Técnica Superior de Ingeniería de Telecomunicación
Universidad Rey Juan Carlos

Septiembre de 2022



© 2022 Miguel Angel Ortuño Pérez.
Algunos derechos reservados. Este documento se distribuye bajo la
licencia *Atribución-CompartirIgual 4.0 Internacional* de Creative
Commons, disponible en
<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

- 1 **Introducción a Linux**
 - Un poco de historia de Unix y Linux
 - ¿GNU/Linux ó Linux?
- 2 **Linux es Software Libre**
 - Inconvenientes del software libre para el usuario
 - Ventajas del software libre para el usuario
- 3 **El Sistema Operativo**
 - Procesos
 - Servicios principales de un sistema UNIX
 - Interfaces de usuario

Un poco de historia de Unix y Linux

- UNIX surgió en 1969 en los Laboratorios Bell (Ken Thomson, Dennis Ritchie)
- Dos grandes vertientes
 - BSD: SunOS, NetBSD, OpenBSD, Mac OS
 - System V: Solaris, Iris, Aix, Linux (año 1991)
Distribuciones Linux
 - Slackware
 - Gentoo
 - Suse
 - RedHat y derivados: Fedora, Mandriva (Mandrake)
 - Debian y derivados: Ubuntu, knoppix, GnuLiNex, guadalinux

Partes de un sistema operativo

- Kernel (Núcleo): elemento más importante. Permite que las aplicaciones accedan al hardware. Es responsable de la gestión de recursos, seguridad, etc
- Procesos de usuario: distintos programas ejecutándose concurrentemente en un sistema
- La interacción entre el núcleo y los procesos se hace mediante llamadas al sistema (*system calls*)

La shell es un interfaz de usuario en modo texto. Es una aplicación como otra cualquiera

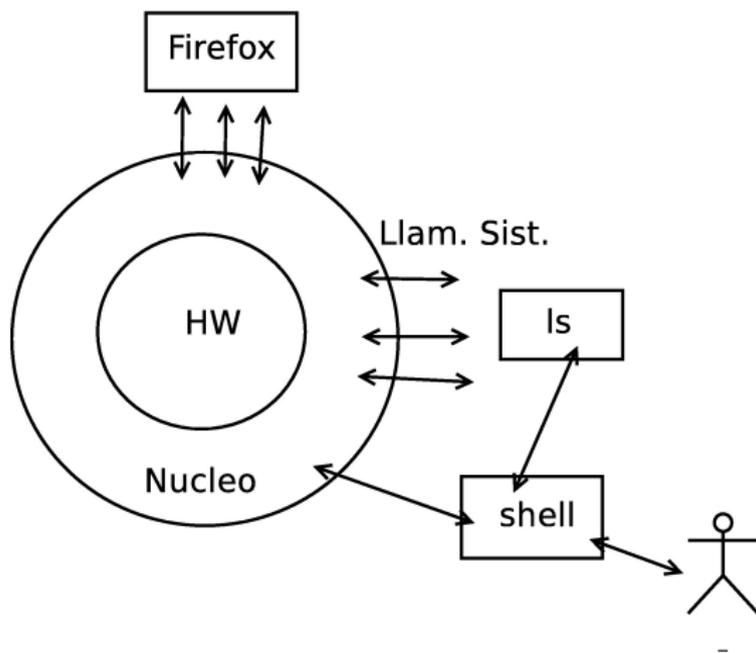


Figura: El Sistema Operativo

¿GNU/Linux ó Linux?

- 1 La Free Software Foundation (Richard Stallman) considera que:
 - Linux es estrictamente el kernel
 - Los procesos de usuario (*programas y otras utilidades básicas para el sistema*) provienen del proyecto GNU (y algunos otros).
 - Al conjunto se le debe llamar GNU/Linux.
- 2 Un número importante de personas y organismos se oponen a esta definición. La mayoría de la gente lo llama simplemente Linux

Linux es Software Libre

Linux es el *producto estrella* del Soft. Libre

- Hay software libre para cualquier S.O.
- Hay software propietario para Linux

Cuatro libertades. Quien lo recibe tiene:

- libertad de uso. Usarlo como quiera, donde quiera
- libertad de redistribución. Redistribuirlo a quien quiera, como quiera
- libertad de modificación. Modificar, adaptar, corregir, mejorar
- libertad de distribuir las modificaciones

Imprescindible: disponibilidad de código fuente.

Como cualquier modelo, puede ser criticado. Desarrollar software libre no es ni garantía de calidad ni garantía de éxito. Pero algunos argumentos en contra habituales no tienen ningún sentido:

- *Los médicos, los abogados y los fontaneros no trabajan gratis. ¿Por qué habrían de hacerlo los programadores?*

software libre \neq software gratis
desarrollar software libre \neq no cobrar

¿Qué no es software libre?

- Software gratuito
- Shareware
- Adware
- Versiones de evaluación
- Dominio Público

Tipos de licencias libres:

- Minimalistas. Permiten *cerrar* el código. Pj BSD
- *Protectoras de la libertad*. GPL.
Redistribuciones con mismos derechos que la primera distribución

Motivos para desarrollar software libre

- Ética, satisfacción personal, pertenencia a una comunidad
- Aprendizaje
- Tesis doctorales, TFG
- Empresas que se dedican a otra cosa
- Organismos públicos
- Empresas que obtienen dinero por servicios
- Empresas de Hardware
- etc etc

Inconvenientes del software libre para el usuario

- Ninguno. El modelo de software libre no tiene, por sí mismo, ningún inconveniente para el usuario

Si bien habrá ocasiones, cada vez menos, en que una solución de software propietario resulta más apropiada:

- Software libre inexistente o de calidad insuficiente
- Hardware no soportado
- Soporte insuficiente
- Otros (discutible) *Quien me rodea usa determinado software*

Es muy habitual usar soluciones propietarias *por inercia*, un pequeño esfuerzo en explorar las alternativas libres puede ser extraordinariamente rentable

Ventajas del software libre para el usuario

- 4 libertades
- Facilita la reutilización
- Mucho menor coste
- Nadie impone la renovación de Hw, Sw ni formación de usuarios
- Mejor interoperabilidad y escalabilidad
- Garantía de privacidad
- Permite conocer mejor el software y comprobar su calidad
- Igualdad de oportunidades: Mismas herramientas para todos.
Promoción de economía local

Más información: Estudio FLOSSImpact

El Sistema Operativo se ocupa de:

- Gestión de procesos
- Gestión de memoria
- Gestión de dispositivos
- Gestión de sistemas de ficheros
- Gestión de red

- Procesos = ejecutables + librerías dinámicas
- Identificadores asociados a cada proceso:
 - PID: Identificación única de cada proceso
 - UID: Identificación de usuario
 - GID: Identificación de grupo (posibilidad de varios grupos por proceso)
- $uid=0 \Rightarrow$ *super-usuario*, "root":
 - Control sobre el resto de procesos
 - Permiso para acceder a todos los ficheros
 - Posibilidad de realizar ciertas tareas privilegiadas

- `init`. Primer proceso, padre de todos los demás. Se encarga de arrancar y parar el sistema.
- Terminales remotas: *login* y *logout*
- `syslog`
- Ejecución periódica de órdenes: `cron` y `at`
- Entorno gráfico (X Window)
- Entorno de red (demonios)
- Correo electrónico, sistema de impresión, ...

Interfaz gráfico

- En los años 1980 y 1990, la aparición de los interfaces gráficos supuso un gran avance, al facilitar el uso de los ordenadores, especialmente para usuarios no especializados. También para tareas que se hacen eventualmente
- Para tareas exigentes, son mucho menos eficiente que los interfaces de texto: obligan a hacer las cosas *a mano* y de una en una
- Solo se puede hacer lo que el interfaz haya previsto que se haga
- No es la filosofía Unix, no son estándar
- Exigen sesión gráfica (mucho más caro que `ps` o `ssh`)
- No siempre disponibles (sistemas empujados, routers, etc)
- Hay gestores gráficos, pero no serán válidos en esta asignatura

Unix dispone de interfaz gráfico desde los 80: *X Window*, año 1984 (No confundir con Microsoft Windows).

- X Window System es un sistema gráfico utilizado fundamentalmente en sistemas Unix, aunque es multiplataforma. También llamado X11, está desarrollado por la fundación X.org
- Proporciona un mecanismo para mostrar ventanas gráficas basado en dos partes: cliente y servidor
 - Servidor X: Se ejecuta típicamente en la máquina en la que está sentado el usuario.
 - Clientes X: Aplicaciones que producen una salida gráfica que envían al Servidor X para que la presente en pantalla. Pueden ejecutarse en ordenadores remotos.

Es un sistema complejo, que requiere componentes adicionales:

- Sobre las X Window va el *gestor de ventanas* (Kwin, Enlightenment, Metacity, Xfwm, MWM...)
- Sobre el gestor de ventanas, va el *escritorio* (KDE, Gnome, Xfce...)

Desde finales de la década de 2010, se empieza a popularizar *wayland* como alternativa a X Window

- Es más moderno, ligero y seguro
- También sigue la arquitectura cliente-servidor
- No usa gestores de ventanas sino unos elementos similares llamados *compositors*
- Los escritorios (y las aplicaciones) son los mismos que con X Window, el usuario final no percibe la diferencia

Interfaz de texto: consola

El interfaz de texto resulta más difícil de aprender a manejar, pero resulta mucho más flexible y potente

Write programs that do one thing and do it well. Write programs to work together. Write programs that handle text streams, because that is a universal.

- interfaz texto: teclado
 - terminales x
 - consola: terminales virtuales (Ctrl+Alt+F1) (Ctrl+Alt+F6)
 - Vuelta a sesión X (Ctrl+Alt+F7)
- exit (EOF, Ctrl + D)

En MS Windows el interfaz de consola para la administración es una opción viable desde la aparición en 2006 de PowerShell