

Sistemas Operativos

Kernel

Facultad de Informática
Universidad Nacional de La Plata



Facultad de Informática
UNIVERSIDAD NACIONAL DE LA PLATA

S.O.

- ✓ Versión: Marzo 2025
- ✓ Palabras Claves: Sistemas Operativos, Kernel, GNU/Linux

Los temas vistos en estas diapositivas han sido mayormente extraídos del sitio oficial del Kernel www.kernel.org y <https://linux-kernel-labs.github.io/>. Para mayor información revisar las referencias en la última diapositiva



¿Qué es el kernel Linux?

- Programa que ejecuta programas y gestiona dispositivos de hardware
- Encargado de que el software y el hardware puedan trabajar juntos
- Principales funciones
 - Administración de memoria principal
 - Administración de uso de la CPU
- Es de código abierto a los usuarios (kernel/sched.c)
- Liberado bajo licencia GPLv2
- En una misma estructura de código fuente se da soporte a todas las arquitecturas
- Escrito en:
 - Lenguaje C mayoritariamente
 - Assembler: Instrucciones especiales y de bajo nivel
 - Rust: Para implementar módulos (incluido recientemente a partir de v6.1)
- En un sentido estricto es el Sistema Operativo



¿Qué es el kernel Linux?

- Su desarrollo es colaborativo. En el mismo participan empresas, universidades y desarrolladores independientes
- Su ciclo de desarrollo se extiende generalmente en sprints de 3 a 4 meses, luego de los cuales se genera otro sprint que puede ser de 1 a 2 semanas para comitear código (merge Windows) y posteriormente una ventana de corrección de bugs
 - La discusión sobre el agregado de nuevas características ocurre solo en la merge window
 - EN la merge window van apareciendo “release candidates”, los cuáles son identificados con las siglas rc1, rc2, etc. siguiendo el nombre de la versión
- Linus Torvalds es quien mantiene el Kernel de GNU/Linux y quien aprueba o no los merge requests de cada uno de los colaboradores



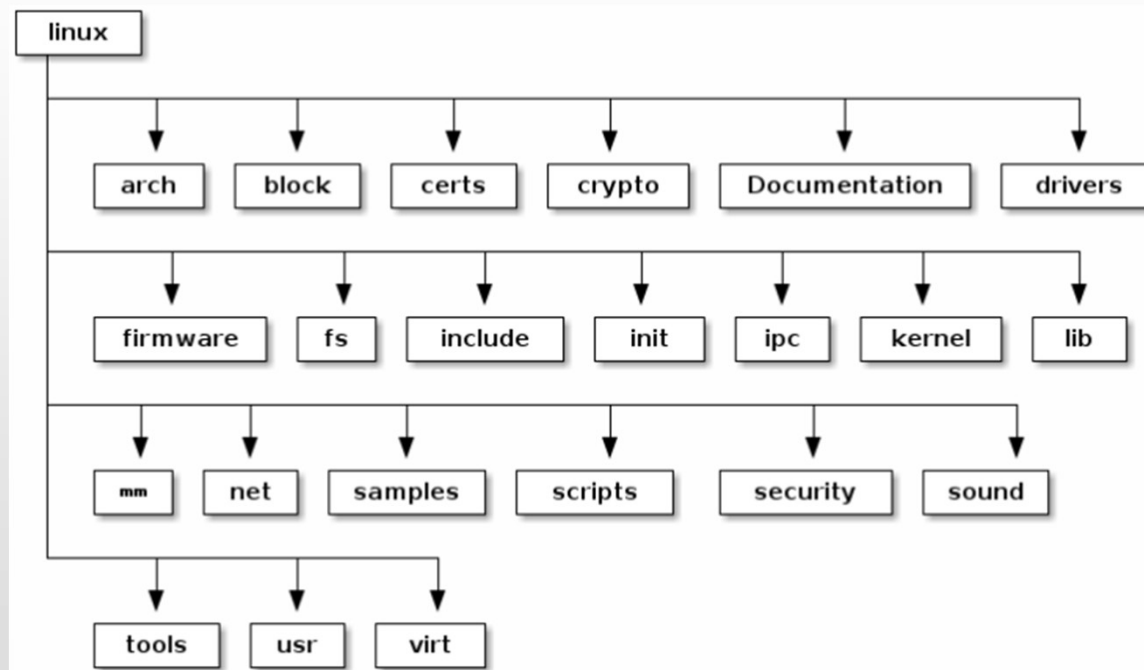
¿Qué es el kernel Linux?

- El Kernel de GNU/Linux está dividido en subsistemas
- Cada subsistema es mantenido por uno o más responsables
- El responsable de cada subsistema acepta o no parches o pull requests de los desarrolladores. Luego el responsable interactúa con Linus para incluir las modificaciones en una versión candidata “rc”
- Cada responsable (mantainer) tiene su propio tree para el desarrollo:
 - Linus Torvalds: <https://web.git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git>
 - David Miller (networking): [git://git.kernel.org/pub/scm/linux/kernel/git/davem/net.git](https://git.kernel.org/pub/scm/linux/kernel/git/davem/net.git)



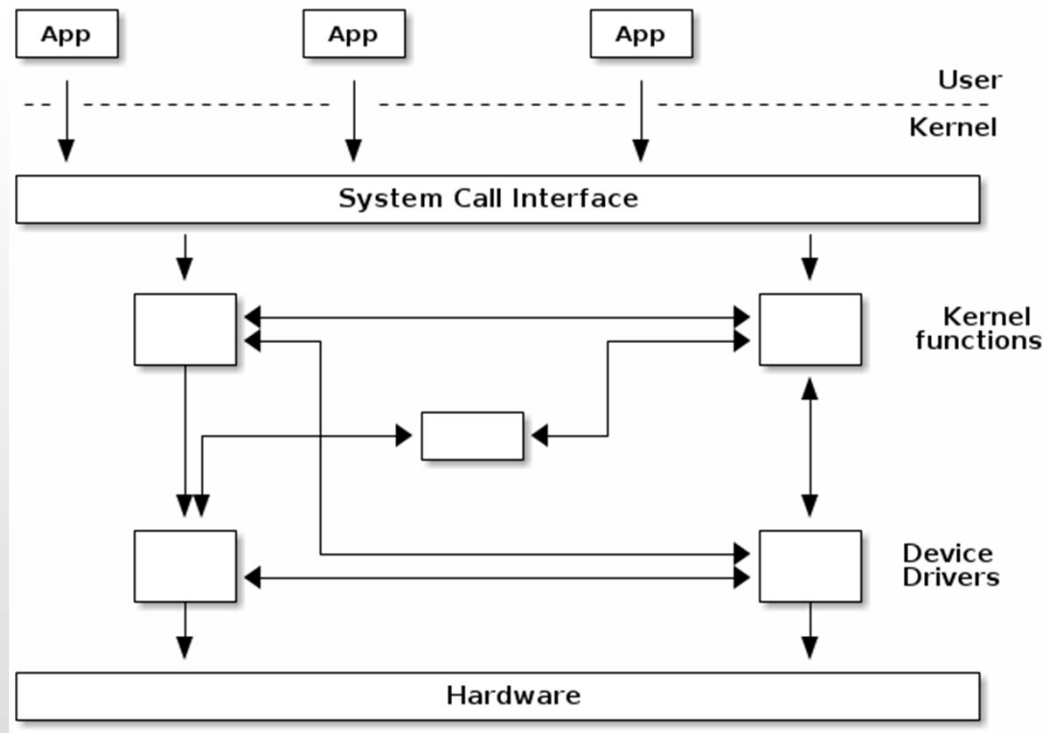
¿Qué es el kernel Linux?

- Organización/Subsistemas del código fuente del Kernel de GNU/Linux:



Funcionalidad del kernel

Arquitectura típica:



Fuente: <https://linux-kernel-labs.github.io>



¿Qué es el kernel Linux?

- Es un núcleo monolítico híbrido
 - Los drivers y el código del Kernel se ejecutan en modo privilegiado
 - Lo que lo hace híbrido es la posibilidad de cargar y descargar funcionalidad a través de módulos

Software Libre

Se dispone la **libertad** de:

1. Usar
2. Estudiar
3. Distribuir
4. Mejorar y publicar



Linux, un poco de historia

En 1991 Linus Torvalds inicia la programación del kernel Linux basado en Minix (Clon de Unix desarrollado por Tanenbaum en 1987 con el fin de crear un SO de uso educativo).

Primer anuncio (1991)

Hello everybody out there using minix -
I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu)
for 386(486) AT clones.
[. . .]
It is NOT protable (uses 386 task switching etc), and it probably never will support
anything other than AT-harddisks, as that's all I have :-(.

Linus Torvalds, 25/08/1991, comp.os.minix

<https://www.minix3.org/index.html>



Facultad de Informática
UNIVERSIDAD NACIONAL DE LA PLATA

Linux, un poco de historia

El 5 de octubre de 1991, se anuncia la primera versión “oficial” de Linux (0.02).

Primera release (1991)

[. . .]

As I mentioned a month ago, I'm working on a free version of a minix-lookalike for AT-386 computers. It has finally reached the stage where **it's even usable** (though may not be depending on what you want), and I am willing to put out the sources for wider distribution. It is just version 0.02 (+1 (very small) patch already), but **I've successfully run bash/gcc/gnu-make/gnu-sed/compress etc under it.**

[. . .]

Linus Torvalds, 05/10/1991, comp.os.minix



Linux, un poco de historia

En 1992, con la release de la versión 0.12, se decide cambiar a una licencia GNU.

Licencia GNU (1992)

[. . .]

The **Linux copyright will change**: I've had a couple of requests to make it compatible with the **GNU copyleft**, removing the “**you may not distribute it for money**” condition. I agree. I propose that the copyright be changed so that it confirms to GNU - **pending approval of the persons who have helped write code**. I assume this is going to be no problem for anybody: If you have grievances (“I wrote that code assuming the copyright would stay the same”) mail me.

Otherwise **The GNU copyleft takes effect as of the first of February**. If you do not know the gist of the GNU copyright - read it.

[. . .]

Linus Torvalds, 1992



Linux, un poco de historia

Fechas relevantes:

- En marzo de 1994 Torvalds considera que todos los componentes del kernel estaban suficientemente maduros y lanza la versión 1.0.
- En el año 1995 Linux se porta a arquitecturas DEC Alpha y Sun SPARC. Con el correr de los años se portó a otra decena de arquitecturas.
- En mayo de 1996 se decide adoptar a Tux como mascota oficial de Linux.



Linux, un poco de historia

Fechas relevantes:

- En julio de 1996 se lanza la versión 2.0 y se define un sistema de nomenclatura.
- Se desarrolló hasta febrero de 2004 y terminó con la versión 2.0.40. Esta versión comenzó a brindar soporte a **sistemas multiprocesadores**.
- En 2001 se lanza la versión 2.4 y se deja de desarrollar a fines del 2010 con la 2.4.37.11. La versión 2.4 fue la que catapultó a GNU/Linux como un sistema operativo estable y robusto.



Versionado previo a la versión 2.6

Anatomía de una versión de Linux < 2.6:

X.Y.Z

- **X** Indicaba la serie principal. Cambiaba al agregar/quitar una funcionalidad muy importante.
- **Y** Indicaba si era una versión de producción o desarrollo.
- **Z** Bugfixes.

Existían dos versiones del kernel:

- Números **Y** pares indicaban una versión en Producción (estable)
- Números **Y** impares indicaban una versión en Desarrollo



Linux, un poco de historia

Fechas relevantes:

- A fines del año 2003 se lanza la versión 2.6. Esta versión ha tenido muchas mejoras para el *kernel* dentro de las que se destacan soporte de **threads**, mejoras en la **planificación** y soporte de nuevo hardware.
- El 3 de agosto de 2011 se lanza la versión 2.6.39.4 anunciándose la misma desde meses previos como la última en su revisión.
- El 17 Julio de 2011 se lanza la versión 3.0.
 - No agrega mayores cambios. La decisión del cambio son los 20 años del SO y no superar los 40 números de revisión.
 - Totalmente compatible con Kernel 2.6.



Linux, un poco de historia

- El 17 Julio de 2011 se lanza la versión 3.0
 - Termina con la versión 3.8.30
 - Provee mejoras en Virtualización y FileSystems

Anuncio de Linux 3.0 (2011)

[...]

I decided to just bite the bullet, and call the next version 3.0. It will get released close enough to the 20-year mark, which is excuse enough for me, although honestly, the real reason is just that I can no longer comfortably count as high as 40.

[...]

Linus Torvalds, 2011



Versionado en Linux 2.6 a 3.0

- Anatomía de una versión de Linux ≥ 2.6 y < 3.0 :

A.B.C.[D]

- **A** Denota Versión. Cambia con menor Frecuencia (cada varios años).
- **B** Denota revisión mayor.
- **C** Denota revisión menor. Solo cambia cuando hay nuevos drivers o características.
- **D** Se utiliza cuando se corrige un grave error sin agregar nueva funcionalidad.



Versionado en 3.0 y superiores

Anatomía de una versión de ≥ 3.0 :

- En el año 2011, cuando se pasó de la versión 2.6.39 a la 3.0 se volvió a un esquema de 3 números:

A.B.C[-rcX]

- **A** Denota revisión mayor. Cambia con menor Frecuencia (cada varios años).
- **B** Denota revisión menor. Solo cambia cuando hay nuevos drivers o características.
- **C** Número de revisión
- **rcX** Versiones de prueba



Linux, un poco de historia

- El 12 de Abril de 2015 se lanza la versión 4.0
 - Una de sus principales mejoras es la posibilidad de aplicar parches y actualizaciones sin necesidad de reiniciar el SO.
 - Soporte para nuevas CPU

Anuncio de Linux 4.0 (2015)

[. . .]

.. Because the people have spoken, and while most of it was complete gibberish, numbers don't lie. People preferred 4.0, and 4.0 it shall be. Unless somebody can come up with a good argument against it.

[. . .]

Linus Torvalds, 2015



Linux, un poco de historia

- El 3 de Marzo de 2019 se lanza la versión 5.0
 - Soporte de **energy-aware scheduling**: Pensado idealmente para minimizar el consumo energético en smartphones. Utiliza decisiones de consumo de energía para hacer la planificación
 - Soporte de **namespaces para binderfs** que permite ejecutar múltiples instancias de Android.
 - Soporte de **archivos swap en BTRFS**.
 - **Kernel Lockdown Mode** previene el acceso de procesos de usuario (incluso root) a memoria del kernel. (Habilitado por defecto en modo Secure Boot de EFI) - 5.4 (24 Noviembre 2019)
 - Soporte inicial de **USB 4** - 5.6 (29 de Marzo 2020)
 - Nuevo **mecanismo para manejar systemcalls de Windows** para programas como Wine - 5.11 (14 de Febrero 2021)
 - **Landlock security module**: Permite restringir las acciones que un conjunto de programas puede ejecutar en un filesystem de forma simple - 5.13 (27 de Junio de 2021)
 - Nueva system call para crear **áreas de memoria secretas** que no pueden ser accedidas ni por el usuario root (por ejemplo para guardar claves) - 5.14 (29 de Agosto de 2021)
 - Soporte opcional para **paquetes IPv6 de más de 64KB** - 5.19 (31 de Julio de 2021)

<https://www.kernel.org/doc/html/latest/>



Linux, un poco de historia

- El 2 de Octubre de 2022 se lanza la versión 6.0
 - Primer release que soporta módulos escritos en Rust - 6.1 (11 de Diciembre de 2022)
 - Mejoras en la confiabilidad de la implementación de RAID5/6 en Btrfs - 6.2 (19 de Febrero 2023)
 - Mejoras de rendimiento y fragmentación para **Btrfs** 6.3 (23 de Abril 2023)
 - **Soporte inicial para USB4.0 v2** - 6.5 (27 de Agosto 2023)
 - **Nuevo scheduler de tareas (EEVDF)** que mejora el fairness respecto al algoritmo anterior (CFS) Linux 6.6 (29 de Octubre de 2023)
 - Soporte para un **nuevo Filesystem BCachefs** con características similares a ZFS y Btrfs - 6.7 (7 de Enero de 2024)
 - Optimización de las estructuras de datos usadas para networking que mejora la performance de TCP cuando hay muchas conexiones concurrentes en hasta un 40% - 6.8 (10 de Marzo de 2023)



Linux, un poco de historia

Fuentes para los datos de 5.x y 6.x:

- <https://kernelnewbies.org/LinuxVersions>
- https://man7.org/linux/man-pages/man7/kernel_lockdown.7.html
- <https://docs.kernel.org/userspace-api/landlock.html>
- <https://lwn.net/Articles/826313/>
- <https://www.kernel.org/doc/>



¿Por qué recompilarlo?

- Soportar **nuevos dispositivos** como, por ejemplo, una placa de video (<https://github.com/NVIDIA/open-gpu-kernel-modules>)
- Agregar **mayor funcionalidad** (soporte de nuevos filesystems)
- **Optimizar** funcionamiento de acuerdo al sistema en el que corre
- **Adaptarlo al sistema** donde corre (quitar soporte de hardware no utilizado)
- **Corrección de bugs** (problemas de seguridad o errores de programación)



¿Qué necesitamos?

- gcc: Compilador de C
- make: ejecuta las directivas definidas en los Makefiles
- binutils: assembler, linker
- libc6: Archivos de encabezados y bibliotecas de desarrollo
- ncurses: bibliotecas de menú de ventanas (solo si usamos menuconfig)
- initrd-tools: Herramientas para crear discos RAM

Tip

En Debian y distribuciones derivadas, todo este software se encuentra empaquetado.

Por ejemplo, para instalar el software requerido para hacer la práctica:

```
# apt-get install build-essential libncurses-dev kbuild flex bison libelf-dev bc
```


Proceso de compilación de Linux

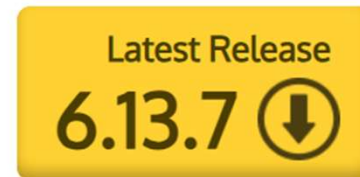
1. Obtener el código fuente.
2. Preparar el árbol de archivos del código fuente.
3. Configurar el kernel
4. Construir el kernel a partir del código fuente e instalar los módulos.
5. Reubicar el kernel.
6. Creación del initramfs
7. Configurar y ejecutar el gestor de arranque (GRUB en general).
8. Reiniciar el sistema y probar el nuevo kernel.



Obtener el código fuente

<http://www.kernel.org>

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/



mainline:	6.14-rc7	2025-03-16	[tarball]	[patch]	[inc. patch]	[view diff]	[browse]	
stable:	6.13.7	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	6.12.19	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	6.6.83	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	6.1.131	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	5.15.179	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	5.10.235	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	5.4.291	2025-03-13	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
linux-next:	next-20250317	2025-03-17						[browse]



Referencias

Linus Torvalds

Linux 3.0-rc1 <https://lkml.org/lkml/2011/5/29/204>, 2011

Andrew Tanenbaum

Minix <http://www.minix3.org/>

Linus Torvalds

Linux History <https://www.cs.cmu.edu/~awb/linux.history.html>, 1992

Linus Torvalds

Release notes for Linux v0.12 <http://ftp.funet.fi/pub/linux/historical/kernel/old-versions/RELNOTES-0.12>, 1992

Linux Kernel <https://www.kernel.org/doc/>

Debian Linux Kernel Handbook

<http://kernel-handbook.alioth.debian.org/>



¿Preguntas?



Facultad de Informática
UNIVERSIDAD NACIONAL DE LA PLATA