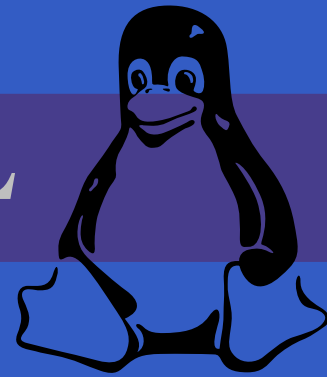


# Cómo compilar el kernel Linux

Guillermo Valdés Lozano

# Documento protegido por GFDL



Copyright (c) 2008 Guillermo Valdés Lozano.

e-mail: [guillermo\(en\)movimientolibre.com](mailto:guillermo(en)movimientolibre.com)

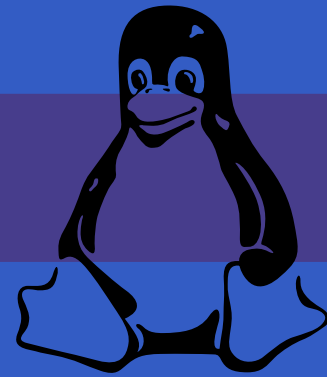
<http://www.movimientolibre.com/>

Se otorga permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre de GNU, Versión 1.2 o cualquier otra versión posterior publicada por la Free Software Foundation; sin Secciones Invariantes ni Textos de Cubierta Delantera ni Textos de Cubierta Trasera.

Una copia de la licencia está en

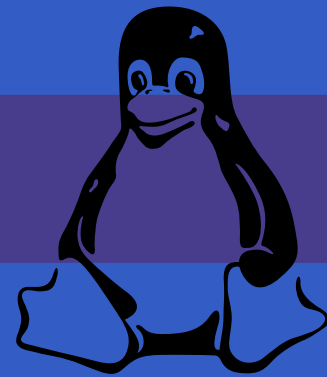
<http://www.movimientolibre.com/licencias/gfdl.html>

# Introducción



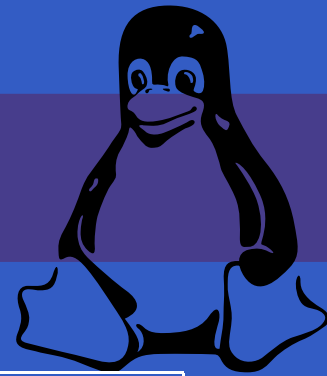
- Linux es el núcleo o kernel del sistema operativo libre denominado GNU/Linux (también llamado Linux).
- Linux fue creado por Linus Torvalds en 1991.
- Linux está protegido por la licencia GPL y está escrito en C.

# ¿Qué es el kernel?



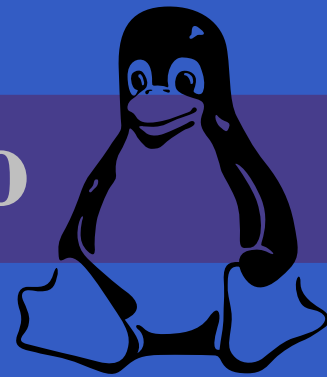
- Cada computadora, lector de discos, disco duro, tarjeta de video, etc. es diferente.
- El kernel es el programa que sabe que hacer para que esos componentes ejecuten su tarea como abrir un documento, borrarlo de un disco o mostrar una imagen en pantalla.
- El kernel debe saber cómo hablar a esos diferentes componentes materiales y controlar las labores corrientes que hacemos a diario con nuestras computadoras.

# Algunas versiones de Linux



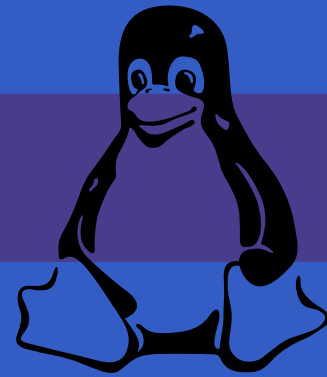
Fecha	Versión	Líneas código
septiembre de 1991	0.01	10,239
14 de marzo de 1994	1.0.0	176,250
marzo de 1995	1.2.0	310,950
25 de enero de 1999	2.2.0	1,800,847
4 de enero de 2001	2.4.0	3,377,902
17 de diciembre de 2003	2.6.0	5,929,913

# Kernel genérico vs personalizado



- Un kernel genérico es aquel capaz de funcionar en muchos tipos de equipos y que tiene los controladores de la mayoría de los componentes que se pudiera encontrar.
- Un kernel personalizado está optimizado al 100 % para nuestra máquina.
  - Añadimos los módulos que necesitamos.
  - Es más ligero, se carga más rápido.
  - Reconoce nuevos componentes y puede tener mejoras en su código.

# Descargas en Debian



Instale estos paquetes y sus dependencias.

```
$ su
# apt-get install kernel-package
# apt-get install build-essential
# apt-get install linux-source-2.6.18
# apt-get install libncurses-dev
# apt-get install fakeroot
```

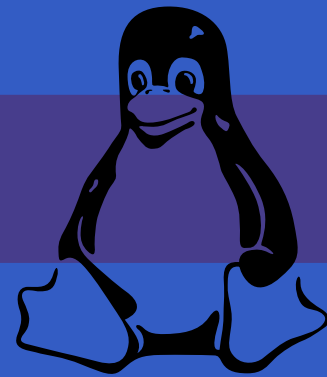
Desempaque el archivo comprimido.

```
# cd /usr/src
# tar xjf /usr/src/linux-source-2.6.18.tar.bz2
```

Haga un enlace directo.

```
# ln -s linux-source-2.6.18 linux
```

# Descargas en Gentoo



Sincronizamos el *portage* y buscamos *sources*.

```
$ su
# emerge --sync
# emerge -s sources
```

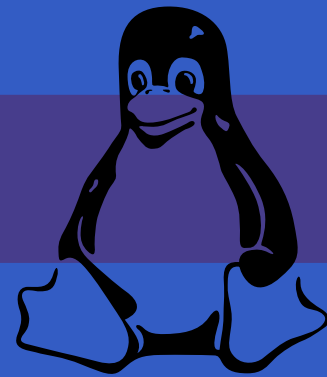
Se recomienda *gentoo-sources*.

```
# emerge -pu gentoo-sources
# rm /usr/src/linux
# emerge -u gentoo-sources
```

Las fuentes del kernel estarán en ***/usr/src/linux***.



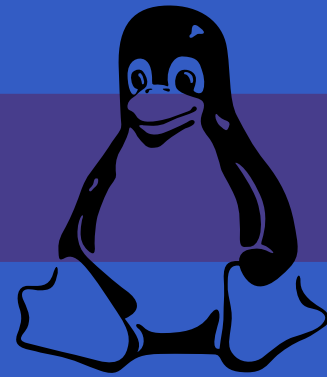
# Tome nota de los componentes



Abra el gabinete y tome nota de los componentes. Ejemplo:

- T. Madre Biostar P4TDP, Fury DDR 533
  - El disco duro y el CD-ROM se conectan por cintas IDE.
  - Tiene una ranura para tarjeta de video AGP.
  - Chip VIA VT6202 0208CD.
  - Chip de audio integrado C3DX CMI 8738/PCI-6ch-LX.
  - Chip ATA 100.
  - Tiene puertos USB por delante y por detrás.
- T. de Video AGP de 32 MB con etiqueta TAVCATG1293A
- T. de Audio Genius con chip C3DX CMI 8738/PCI-6ch-LX
- T. de Red con chip DM9102AF

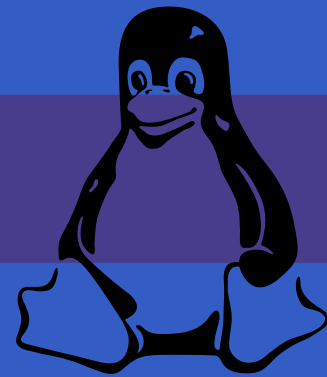
# Comando lspci



## **lspci** lista los dispositivos PCI. Ejemplo:

```
$ lspci
00:00.0 Host bridge: Intel Corporation 82845 845 (Brookdale)
Chipset Host Bridge (rev 04)
00:01.0 PCI bridge: Intel Corporation 82845 845 (Brookdale)
Chipset AGP Bridge (rev 04)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev 05)
00:1f.0 ISA bridge: Intel Corporation 82801BA ISA Bridge (LPC) (rev 05)
00:1f.1 IDE interface: Intel Corporation 82801BA IDE U100 (rev 05)
00:1f.2 USB Controller: Intel Corporation 82801BA/BAM USB (Hub #1) (rev 0)
00:1f.3 SMBus: Intel Corporation 82801BA/BAM SMBus (rev 05)
00:1f.4 USB Controller: Intel Corporation 82801BA/BAM USB (Hub #2) (rev 0)
01:00.0 VGA compatible controller: Silicon Integrated Systems
[SiS] 300/305 PCI/AGP VGA Display Adapter (rev 90)
02:00.0 Multimedia audio controller: C-Media Electronics Inc CM8738 (rev
02:04.0 Ethernet controller: Davicom Semiconductor, Inc. 21x4x
DEC-Tulip compatible 10/100 Ethernet (rev 31)
02:05.0 USB Controller: VIA Technologies, Inc. VT82xxxxx
UHCI USB 1.1 Controller (rev 50)
02:05.1 USB Controller: VIA Technologies, Inc. VT82xxxxx
UHCI USB 1.1 Controller (rev 50)
02:05.2 USB Controller: VIA Technologies, Inc. USB 2.0 (rev 51)
```

# Observar los módulos cargados

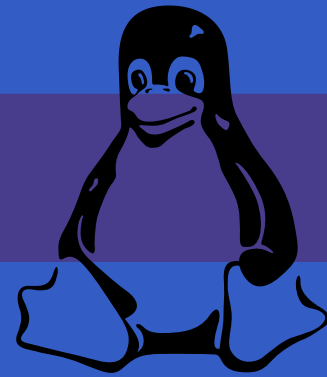


**lsmod** muestra los módulos cargados.

```
# lsmod
```

Módulo	Descripción
sis	Controlador de la tarjeta de video SIS
partport_pc	Puerto paralelo
gameport	Hay un puerto de juegos/midi
floppy	Controlador del lector de floppys
snd_cmipci	Controlador de la tarjeta de sonido Genius
i2c_i801	Tiene que ver con un chip intel
psmouse	Mouse por el puerto PS/2
intel_agp	Ranura AGP de la tarjeta madre
usbhid, ehci_hcd, uhci_hcd	Tienen que ver con el USB
ide_cd, ide_disk	Controladores del CD-ROM y del disco duro
dmfe	Controlador de la tarjeta de red Tulip/Davicom

# Bitácora del kernel



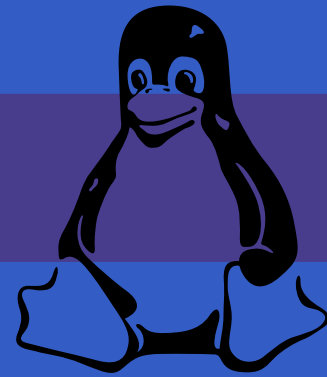
**dmesg** muestra la bitácora del kernel.

```
# dmesg | more
```

Fragmento de la salida de **dmesg**, ejemplo:

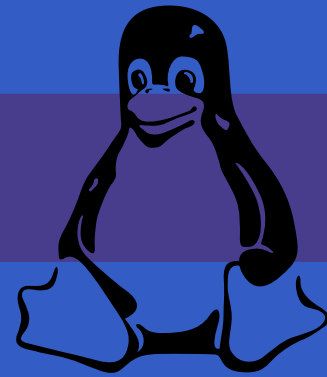
```
hda: IC35L120AVV207-0, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
Probing IDE interface ide1...
hdc: HL-DT-ST GCE-8520B, ATAPI CD/DVD-ROM drive
hdd: HL-DT-ST DVDRAM GSA-H10A, ATAPI CD/DVD-ROM drive
ide1 at 0x170-0x177,0x376 on irq 15
hda: max request size: 512KiB
hda: 241254720 sectors (123522 MB) w/1821KiB Cache,
CHS=16383/255/63, UDMA(100)
hda: cache flushes supported
hda: hda1 hda2 hda3 hda4 < hda5 hda6 hda7 hda8 hda9 hda10 >
hdc: ATAPI 40X CD-ROM CD-R/RW drive, 2048kB Cache, UDMA(33)
Uniform CD-ROM driver Revision: 3.20
hdd: ATAPI 48X DVD-ROM DVD-R-RAM CD-R/RW drive,
2048kB Cache, UDMA(33)
```

# Procesador



```
# cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 15
model         : 1
model name    : Intel(R) Pentium(R) 4 CPU 1.70GHz
stepping      : 2
cpu MHz       : 1700.335
cache size   : 256 KB
fdiv_bug     : no
hlt_bug      : no
f00f_bug     : no
coma_bug     : no
fpu          : yes
fpu_exception : yes
cpuid level  : 2
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic sep
mtrr pge mca cmov pat pse36 clflush dts acpi
mmx fxsr sse sse2 ss ht tm up
bogomips     : 3404.16
```

# Interfaz de configuración



Vaya al directorio de las fuentes.

```
# cd /usr/src/linux
```

Puede iniciar con la configuración por defecto.

```
# make defconfig
```

Interfaz basada en *ncurses* (recomendada):

```
# make menuconfig
```

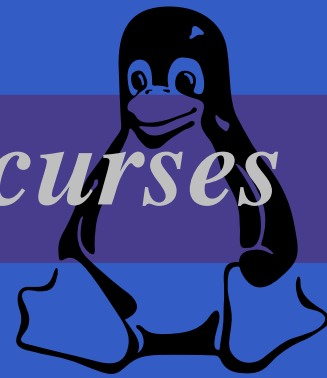
Interfaz basada en GTK:

```
# make gconfig
```

Interfaz basada en QT:

```
# make xconfig
```

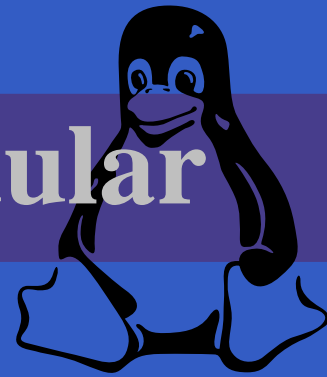
# Interfaz de configuración bajo *ncurses*



## Forma de usarla:

- Use las flechas para navegar.
- Estando sobre una opción presione **Y** para incluirla dentro del kernel, **N** para excluirla o **M** para que sea un módulo.
- Las opciones marcadas con [ \* ] serán incluidas en el kernel, las marcadas con <M> serán compiladas como módulos y las vacías no serán compiladas.
- Presione ? para mostrar la ayuda de la opción iluminada.
- Presione ESC para bajar una rama o para salir si está en el menú raíz.
- Al salir le preguntará si quiere guardar los cambios.

# Kernel monolítico vs kernel modular



Desde un punto de vista extremista:

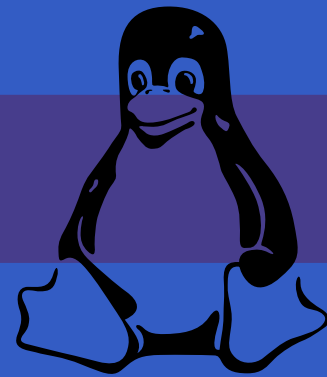
- Un kernel monolítico tiene todas las opciones marcadas con **Y**.
- Un kernel modular es aquel con la mayoría de las opciones marcadas con **M**.

Use un kernel monolítico si el hardware no cambia. En cambio un kernel modular puede:

- Tener un tamaño menor que si fuera monolítico.
- Puede cargar a voluntad o automáticamente los módulos que necesite.
- Pero **NO** marque como módulos los controladores indispensables para el arranque.



# Cambios de versión



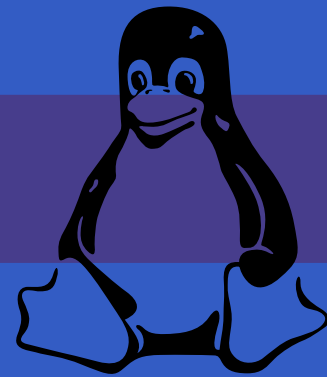
Tener la versión más reciente del kernel le brinda:

- Código mejorado y *bugs* solucionados.
- Soporte de nuevo hardware
- Posiblemente mejor desempeño.

El constante crecimiento y mejora del kernel Linux ocasiona que:

- El menú de configuración cambie de una versión a otra.
- Ocurre el aviso y luego el retiro de código (*DEPRECATED*).

# Configuraciones básicas



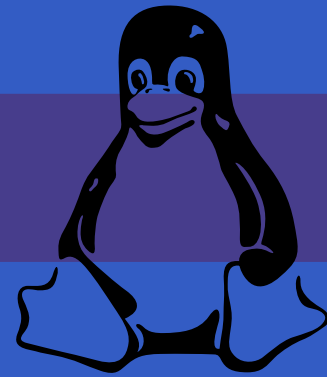
Para que aparezcan las opciones de tipo experimental:

```
Code maturity level options --->
[*] Prompt for development and/or incomplete code
```

Para que cuando conecte un dispositivo externo el módulo se cargue por sí sólo:

```
Loadable module support --->
[*] Enable loadable module support
[*]   Module unloading
[*]   Forced module unloading
[*]   Automatic kernel module loading
```

# Configuraciones básicas



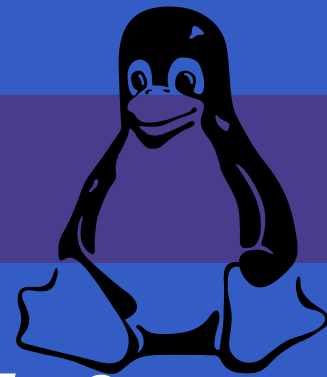
Seleccione la familia del procesador:

```
Processor type and features --->  
  Processor family (Pentium-4/Celeron(P4-based)/P
```

Si tiene un procesador *Intel* con capacidad *HyperThreading* (HT), o bien, con dos o más procesadores, debe activar:

```
Processor type and features --->  
  [*] Symmetric multi-processing support
```

# Configuraciones básicas



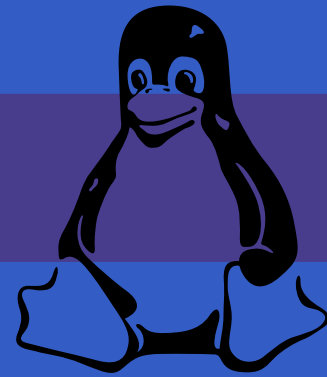
Si tiene particiones formateadas en *Ext2* y *Ext3*:

```
File systems --->
<*> Second extended fs support
[*]   Ext2 extended attributes
<*> Ext3 journalling file system support
[*]   Ext3 extended attributes
```

Para que los administradores de archivos muestren los cambios sin necesidad de refrescar:

```
File systems --->
[*] Inotify file change notification support
[*]   Inotify support for userspace
Pseudo filesystems --->
--- /proc file system support
[*]   /proc/kcore support
[*] Virtual memory file system support (former shm fs)
```

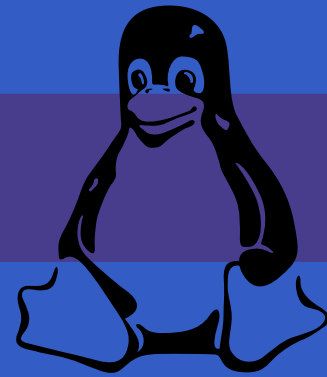
# Configuraciones básicas



## Para pertenecer a una red:

```
Networking --->
[*] Networking support
    Networking options --->
        <*> Packet socket
        [*]   Packet socket: mmaped IO
        <*> Unix domain sockets
        [*] TCP/IP networking
        [*] Network packet filtering (replaces ipchains) --->
            Core Netfilter Configuration --->
                <*> Netfilter Xtables support (required for ip_tables)
                <*>   "limit" match support
                <*>   "mac" address match support
                <*>   "state" match support
            IP: Netfilter Configuration --->
                <*> Connection tracking (required for masq/NAT)
                <*>   FTP protocol support
                <*> IP tables support (required for filtering/masq/NAT)
                <*>   Packet filtering
```

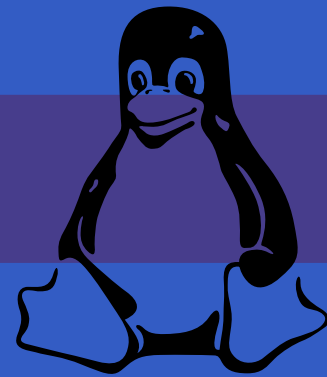
# Dispositivos de almacenamiento



Muy recomendado:

```
Device Drivers --->
  Block devices --->
    <*> Normal floppy disk support
    <*> Loopback device support
    <*> RAM disk support
    [*] Initial RAM filesystem and RAM disk (initramfs) support
    <*> Packet writing on CD/DVD media
```

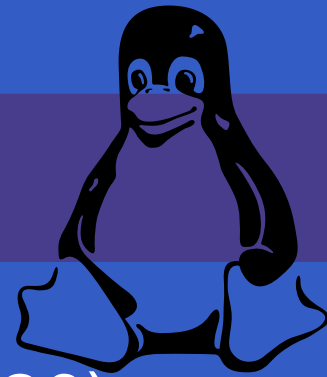
# Unidades ATA



## Discos por cinta IDE (maestro, esclavo) chipset Intel:

```
Device Drivers --->
ATA/ATAPI/MFM/RLL support --->
<*> ATA/ATAPI/MFM/RLL support
<*>   Enhanced IDE/MFM/RLL disk/cdrom/tape/floppy support
<*>   Include IDE/ATA-2 DISK support
<*>   Include IDE/ATAPI CDROM support
<*>   Include IDE/ATAPI FLOPPY support
<*>   generic/default IDE chipset support
[*]   PCI IDE chipset support
[*]   Sharing PCI IDE interrupts support
<*>   Generic PCI IDE Chipset Support
[*]   Generic PCI bus-master DMA support
[*]   Use PCI DMA by default when available
<*>   Intel PIIXn chipsets support
```

# Unidades SATA

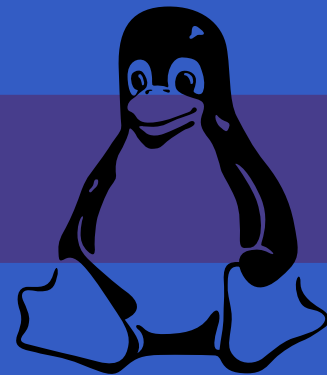


Ejemplo soporte SATA para Intel (kernel 2.6.20):

```
Device Drivers --->
  Serial ATA (prod) and Parallel ATA (experimental)
  <*> ATA device support
  <*> Intel PIIX/ICH SATA support
```



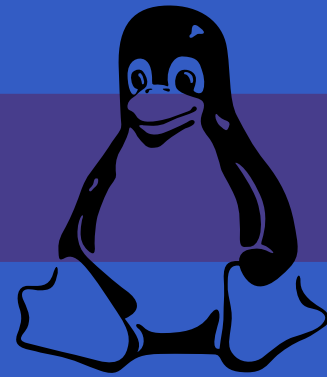
# Soporte SCSI



Algo indispensable para el montaje de los dispositivos de almacenamiento que conecte vía *USB* (por ejemplo una memoria USB) y para los quemadores de CD y DVD es el soporte *SCSI*:

```
Device Drivers  --->
  SCSI device support  --->
    [*] legacy /proc/scsi/ support
    <*> SCSI disk support
    <*> SCSI CDROM support
    <*> SCSI generic support
```

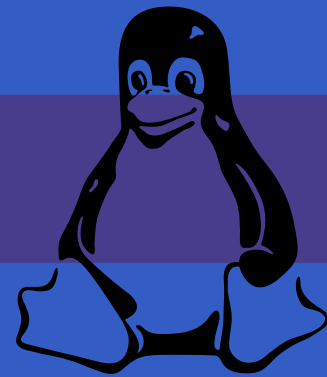
# Dispositivos USB



Si el equipo tiene puertos USB para conectar teclado, ratón, impresora, etc.:

```
Device Drivers --->
  USB support --->
    <*> Support for Host-side USB
    [*]   USB device filesystem
    <M>   EHCI HCD (USB 2.0) support
    [*]   Full speed ISO transactions (EXPERIMENTAL)
    [*]   Root Hub Transaction Translators (EXPERIMENTAL)
    <M>   OHCI HCD support
    <M>   UHCI HCD (most Intel and VIA) support
    <M>   USB Printer support
    <*>   USB Mass Storage support
    <M>   USB Human Interface Device (full HID) support
    [*]   HID input layer support
```

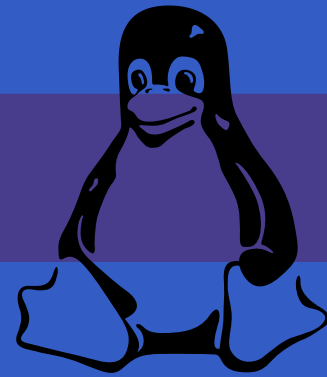
# Tarjeta de red



Marque el controlador de su tarjeta de red alámbrica. Ejemplo para la tarjeta Davicom:

```
Device Drivers --->
  Network device support --->
    [*] Network device support
      Ethernet (10 or 100Mbit) --->
        Tulip family network device support --->
          [*] "Tulip" family network device support
            <*> Davicom DM910x/DM980x support
```

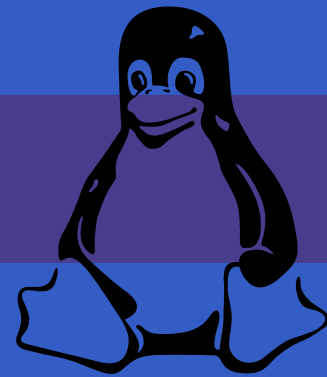
# Tarjeta de red inalámbrica



## Ejemplo para *Intel PRO/Wireless 2200BG*:

```
Networking --->
  --- Networking support
  <M>   Generic IEEE 802.11 Networking Stack
  <M>   IEEE 802.11 WEP encryption (802.1x)
  <M>   IEEE 802.11i CCMP support
  <M>   IEEE 802.11i TKIP encryption
Device Drivers --->
  Network device support --->
    [*] Network device support
    Wireless LAN (non-hamradio) --->
      [*] Wireless LAN drivers (non-hamradio) & Wireless Extensions
      <M>   Intel PRO/Wireless 2200BG and 2915ABG Network Connection
      [*]   Enable promiscuous mode
      [*]   Enable QoS support
Cryptographic options --->
  --- Cryptographic API
  <M>   Cryptographic algorithm manager
  <M>   SHA1 digest algorithm
  <M>   SHA256 digest algorithm
  <M>   ECB support
  <M>   CBC support
  <M>   AES cipher algorithms
  <M>   AES cipher algorithms (i586)
  <M>   ARC4 cipher algorithm
  <M>   Michael MIC keyed digest algorithm
```

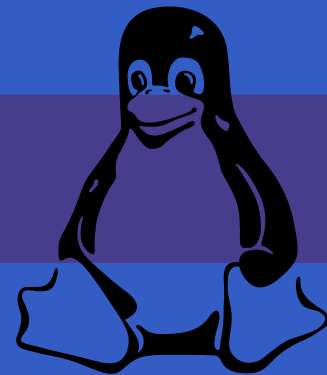
# Tarjeta de video



## Ejemplo para una tarjeta ATI Radeon:

```
Device Drivers --->
  Character devices ---->
    <*> /dev/agpgart (AGP Support)
    <M>   ATI chipset support
    <*> Direct Rendering Manager (XFree86 4.1.0 a
    <M>   ATI Radeon
    [*] HPET - High Precision Event Timer
    [*]   Allow mmap of HPET
    <*> Hangcheck timer
```

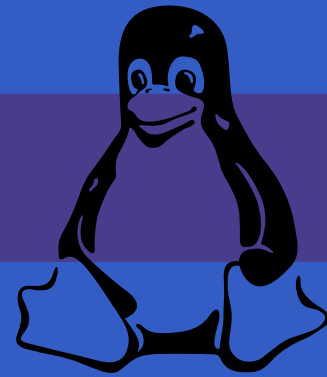
# Tarjeta de audio



## Ejemplo para una tarjeta C-Media bajo ALSA:

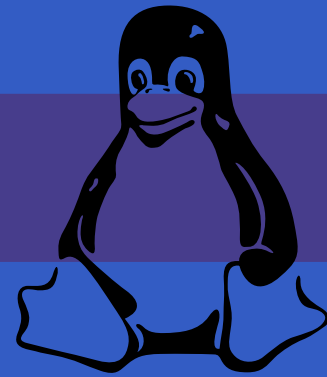
```
Device Drivers  --->
  Sound  --->
    <*> Sound card support
    Advanced Linux Sound Architecture  --->
      <M> Advanced Linux Sound Architecture
      <M>   Sequencer support
      <M>   OSS Mixer API
      <M>   OSS PCM (digital audio) API
      [*]   OSS PCM (digital audio) API - Include plugin system
      [*]   OSS Sequencer API
      <M>   RTC Timer support
      [*]   Use RTC as default sequencer timer
      [*]   Support old ALSA API
    PCI devices  --->
      <M> C-Media 8738, 8338
    Open Sound System  --->
      < > Open Sound System (DEPRECATED)
```

# Guardar la configuración



- Revise la configuración que ha elaborado.
- Guarde los cambios hechos.
- Puede respaldar el archivo **.config**

# Compilar en Debian



Puede solicitar que se eliminen las compilaciones hechas con anterioridad con:

```
# make-kpkg clean
```

Para crear un kernel *sin initrd*:

```
# fakeroot make-kpkg \  
  --revision=personal.1.0 kernel_image
```

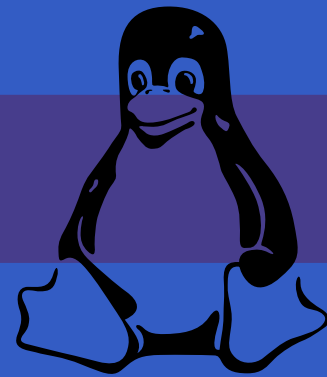
Para crear un kernel *con initrd*:

```
# fakeroot make-kpkg --initrd \  
  --revision=personal.1.0 kernel_image
```

Y tome un café en lo que su equipo compila.



# Instalar el kernel en Debian



Después de la compilación, tendrá el archivo **.deb** en **/usr/src**. Para instalar:

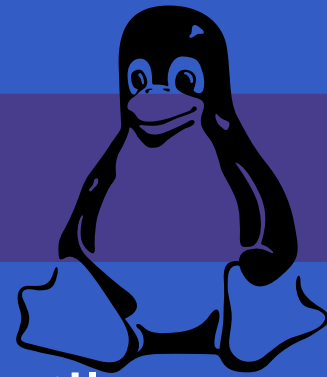
```
# cd /usr/src
```

```
# dpkg -i linux-image-2.6.18_personal.1.0_i386.de
```

Este comando hará por usted todos estos pasos:

- Instalará en **/boot** el nuevo kernel junto con el *initrd* si fue solicitado.
- Copiará los módulos al directorio **/lib/modules/version-del-kernel**
- Agregará la opción para seleccionarlo en el gestor de arranque *Grub* o *Lilo*.

# Compilar en Gentoo



En Gentoo Linux se hace la compilación al estilo tradicional. Puede hacer una *limpia* con el comando:

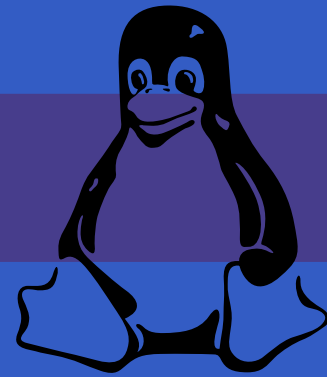
```
# make clean
```

Para compilarlo simplemente ejecute:

```
# make
```

Tome un descanso en lo que su equipo compila.

# Instalar el kernel en Gentoo



Monte la partición **/boot** y copie el kernel a la misma.

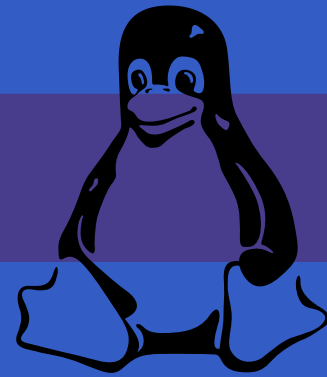
```
# mount /boot
```

```
# cp arch/i386/boot/bzImage /boot/kernel-2.6.20-r8
```

Una buena sugerencia es hacer una copia de la configuración.

```
# cp .config /boot/config-2.6.20-r8
```

# Instalar los módulos



Instale los módulos compilados:

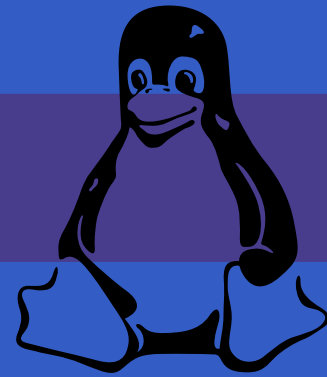
```
# make modules_install
```

Si desea que se carguen siempre algunos módulos en el arranque:

```
# nano -w /etc/modules.autoload.d/kernel-2.6
```

```
# update-modules
```

# Modificar GRUB



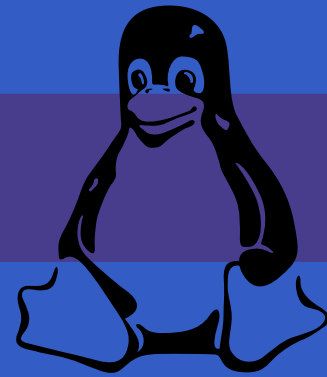
Modifique el gestor de arranque:

```
# nano -w /boot/grub/menu.lst
```

Ejemplo:

```
title  Mi kernel Linux personalizado 2.6.20-r8
root   (hd0,0)
kernel /kernel-2.6.20-r8 root=/dev/hda3
```

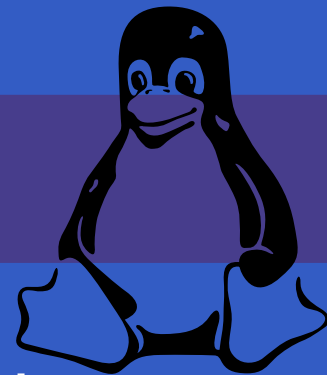
# Primer arranque



Para probar su nuevo kernel, reinicie y selecciónelo en su gestor de arranque.

- La probabilidad de que funcione perfectamente al primer arranque es poca.
- Si falla, tendrá que revisar de nuevo la configuración y volver a compilar, instalar y probar.
- Haga una búsqueda en internet, puede que encuentre la solución a su problema.

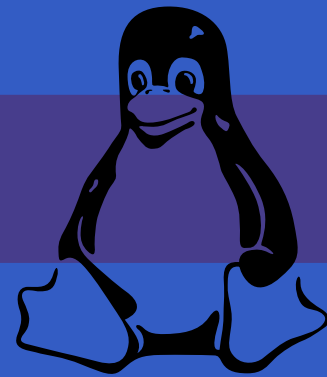
# Revisión



Si el arranque es satisfactorio, no deje de revisar:

- La bitácora con **dmesg | more**, busque mensajes de error.
- Que se hayan cargado los módulos que necesite con el comando **lsmod**.
- Revise que funcionen correctamente los principales dispositivos (video, red, audio, teclado, ratón).
- Pruebe los dispositivos que no estén siempre conectados al equipo, como memorias USB.

# Eliminar un kernel anterior



Ejemplo para remover un kernel en Debian:

```
# dpkg -r linux-image-2.6.17
```

Ejemplo para remover un kernel en Gentoo:

```
# mount /boot
```

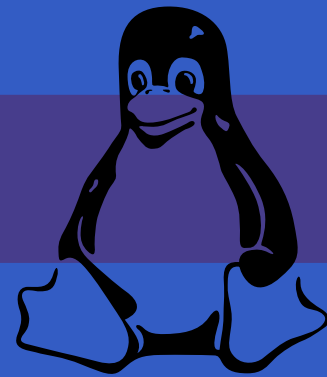
```
# rm /boot/kernel-2.6.17-r4
```

```
# nano -w /boot/grub/menu.lst
```

```
# rm -fr /lib/modules/2.6.17-gentoo-r4
```



# Remover las fuentes



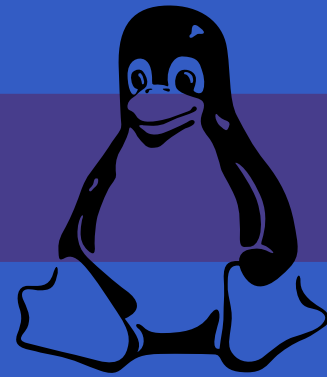
Las fuentes del kernel consumen mucho espacio:

- El directorio `/usr/src/linux-2.6.20-gentoo-r8` me consume **364 MB**.
- Después del comando `make clean` el espacio ocupado descendió a **282 MB**.

Para eliminar las fuentes de kernel no usado (ejemplo):

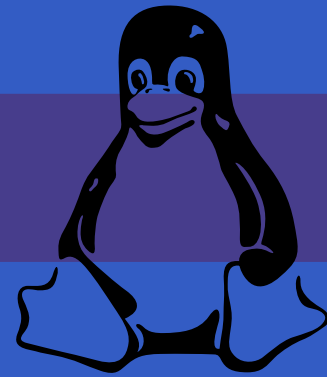
```
rm -rf /usr/src/linux-2.6.17-gentoo-r4
```

# Conclusiones



- Compilar su propio kernel puede llegar a ser una de las cosas más fantásticas que haga con GNU/Linux.
- O una de las más frustrantes.
- Los kernels más recientes soportan más hardware, tienen más *bugs* reparados y mejor desempeño.
- Una de las bondades del Software Libre es que usted puede ESTUDIARLO y MODIFICARLO.

# Referencias



- **Debian Kernel Handbook**  
<http://kernel-handbook.alioth.debian.org/>
- **initrd From Wikipedia, the free encyclopedia**  
<http://en.wikipedia.org/wiki/Initramfs>
- **Gentoo Handbook**  
<http://www.gentoo.org/doc/en/handbook/index.xml>
- **Guía de instalación de Debian GNU/Linux**  
<http://www.debian.org/releases/stable/i386/index.html.es>